

CMT2380F32 User Guide (Microcontroller Part)

Overview

Thank you for your support and trust in CMOSTEK products. Please read this manual and CMT2380F32 data sheet before using the CMT2380F32.

This manual mainly discusses the functions, operations and usages of the CMT2380F32 controller. The manual targets for readers engaging in the development of this product series.

For peripheral functions, this manual discusses their components and operations, however the related specifications are not covered. For detailed chip specifications, please refer to the data sheet document.

The product models covered in this document are shown in the table below.

Table 1. Product Models Covered in This Document

Product Model	Operating Frequency	Modulation Method	Transmit Power	Sensitivity	Chip Properties	Package
CMT2380F32	127 – 1020 MHz	OOK/(G)FSK	+20 dBm	-120 dBm	Wireless MCU	QFN40

Table of Content

1 FUNCTION MODULES.....	16
1.1 32-BIT CORTEX-M0+ CORE.....	18
1.2 32 K BYTE FLASH	18
1.3 4 K BYTE RAM.....	18
1.4 CLOCK SYSTEM	18
1.5 OPERATING MODE.....	19
1.6 HARDWARE REAL TIME CLOCK (RTC).....	19
1.7 GENERAL PURPOSE IO PORT.....	19
1.8 INTERRUPT CONTROLLER	19
1.9 CONTROLLER RESET	20
1.10 TIMER/COUNTER	21
1.11 WDT (WATCH DOG TIMER).....	22
1.12 UNIVERSAL ASYNCHRONOUS RECEIVER UART0, UART1 AND UART2.....	22
1.13 SYNCHRONOUS SERIAL PERIPHERAL INTERFACE (SPI).....	22
1.14 I2C BUS	22
1.15 BUZZER.....	23
1.16 CLOCK CALIBRATION CIRCUIT	23
1.17 UNIQUE ID NUMBER	23
1.18 CRC16 HARDWARE CYCLIC REDUNDANCY CHECK CODE	23
1.19 12-BIT SARADC.....	23
1.20 VOLTAGE COMPARATOR (VC)	23
1.21 LOW VOLTAGE DETECTOR (LVD).....	23
1.22 EMBEDDED DEBUG SYSTEM	24
1.23 HIGH SECURITY.....	24
1.24 PIN SCHEMATIC	24
1.25 PIN DESCRIPTION	24
1.26 MODULE SIGNAL DESCRIPTION.....	30
2 SYSTEM AND MEMORY.....	34
2.1 OVERVIEW.....	34
2.2 SYSTEM ADDRESS MAP.....	34
3 OPERATING MODE.....	37
3.1 ACTIVE MODE	38
3.2 SLEEP MODE	40
3.3 DEEP SLEEP MODE	41
4 SYSTEM CONTROLLER.....	43
4.1 SYSTEM CLOCK INTRODUCTION.....	43
4.1.1 Internal High Speed RC Clock (RCH).....	44
4.1.2 Internal Low Speed RC Clock (RCL).....	45
4.1.3 External Low Speed Crystal Oscillator Clock XTL.....	45
4.1.4 External High Speed Crystal Clock (XTH)	45
4.1.5 Clock Startup Process.....	45

4.2 SYSTEM CLOCK SWITCHING.....	45
4.2.1 Switching from Internal High Speed Clock to External Low Speed One	45
4.2.2 Switching from Internal High Speed Clock to External High Speed One	46
4.2.3 Switching from Internal Low Speed Clock to External High Speed One	47
4.3 CLOCK CALIBRATION MODULE	47
4.4 INTERRUPT WAKEUP CONTROL (WIC).....	48
4.4.1 NVIC Settings for Waking-up from Deep Sleep Mode to ISR Execution.....	48
4.4.2 Settings for NVIC Waking-up from Deep Sleep Mode without ISR Execution	49
4.4.3 Sleep-on-exit Function Usage.....	49
4.5 REGISTER	50
4.5.1 System Control Register 0 (SYSCTRL0)	51
4.5.2 System Control Register 1 (SYSCTRL1)	52
4.5.3 System Control Register 2 (SYSCTRL2)	54
4.5.4 Oscillation RCH Control Register (RCH_CR).....	54
4.5.5 Oscillation XTH Control Register (XTH_CR)	55
4.5.6 Oscillation RCL Control Register (RCL_CR)	56
4.5.7 Oscillation XTL Control Register (XTL_CR).....	58
4.5.8 Peripheral Module Clock Control Register (PERI_CLKEN)	59
4.5.9 Systick Clock Control (SYSTICK_CR).....	60
5 RESET CONTROLLER	62
5.1 RESET CONTROLLER INTRODUCTION	62
5.1.1 Power-on and Power-off Reset (POR)	62
5.1.2 External Reset Pin Reset.....	62
5.1.3 WDT Reset	63
5.1.4 PCA Reset	63
5.1.5 LVD Low Voltage Reset	63
5.1.6 Cortex-M0+ SYSRESETREQ Reset.....	63
5.1.7 Cortex-M0+ LOCKUP Reset.....	63
5.2 REGISTER	63
5.2.1 Reset Flag Register (Reset_flag)	63
5.2.2 Peripheral Module Reset Control Register (PREI_RESET)	64
6 INTERRUPT CONTROLLER	66
6.1 OVERVIEW.....	66
6.2 INTERRUPT PRIORITY	66
6.3 INTERRUPT VECTOR TABLE	67
6.4 INTERRUPT INPUT AND ACTIONS IN PENDING STATE	68
6.5 INTERRUPT LATENCY	70
6.6 INTERRUPT SOURCE	71
6.7 INTERRUPT STRUCTURE	73
6.8 REGISTER	74
6.8.1 Interrupt Enabling Setting Register (SCS_SETENA)	74
6.8.2 Interrupt Enabling Clearing Register (SCS_CLRENA).....	75
6.8.3 Interrupt Pending State Setting Register (SCS_SETPEND).....	75
6.8.4 Interrupt Pending State Clearing Register (SCS_CLRPEND)	76
6.8.5 Interrupt Priority Register (SCS_IPR0).....	77

6.8.6	Interrupt Priority Register (SCS_IPR1).....	77
6.8.7	Interrupt Priority Register (SCS_IPR2).....	78
6.8.8	Interrupt Priority Register (SCS_IPR3)	79
6.8.9	Interrupt Priority Register (SCS_IPR4).....	79
6.8.10	Interrupt Priority Register (SCS_IPR5).....	80
6.8.11	Interrupt Priority Register (SCS_IPR6).....	81
6.8.12	Interrupt Priority Register (SCS_IPR7).....	81
6.8.13	Interrupt Mask Special Register (SCS_PRIMASK).....	82
6.9	BASIC SOFTWARE OPERATION.....	82
6.9.1	External Interrupt Enabling.....	83
6.9.2	NVIC Interrupt Enabling and Clearing Enabling.....	83
6.9.3	NVIC Interrupt Pending and Clearing	83
6.9.4	NVIC Interrupt Priority	83
6.9.5	NVIC Interrupt Mask.....	83
7	PORT CONTROLLER.....	84
7.1	PORT CONTROLLER INTRODUCTION	84
7.2	MAJOR FEATURES FOR PORT CONTROLLER	84
7.3	PORT CONTROLLER FUNCTION DESCRIPTION.....	84
7.3.1	Port Multiplexing Function	84
7.3.2	Port Interrupt Function	85
7.3.3	Port Configuration Function.....	86
7.4	OPERATING MODE	86
7.4.1	Port Multiplexing Operation Process.....	86
7.4.2	Port Interrupt Operating Process.....	87
7.4.3	Port Configuration Operating Process.....	88
7.5	PORT CONTROLLER REGISTER DESCRIPTION	88
7.5.1	Port P01 Function Configuration (P01_SEL).....	90
7.5.2	Port P02 Function Configuration (P02_SEL).....	91
7.5.3	Port P03 Function Configuration (P03_SEL).....	92
7.5.4	Port P14 Function Configuration (P14_SEL).....	92
7.5.5	Port P15 Function Configuration (P15_SEL).....	93
7.5.6	Port P23 Function Configuration (P23_SEL).....	94
7.5.7	Port P24 Function Configuration (P24_SEL).....	94
7.5.8	Port P25 Function Configuration (P25_SEL).....	95
7.5.9	Port P26 Function Configuration (P26_SEL).....	96
7.5.10	Port P27 Function Configuration (P27_SEL).....	96
7.5.11	Port P31 Function Configuration (P31_SEL).....	97
7.5.12	Port P32 Function Configuration (P32_SEL).....	98
7.5.13	Port P33 Function Configuration (P33_SEL).....	98
7.5.14	Port P34 Function Configuration (P34_SEL).....	99
7.5.15	Port P35 Function Configuration (P35_SEL).....	100
7.5.16	Port P36 Function Configuration (P36_SEL).....	100
7.5.17	Port P0 Input/Output Configuration Register (P0DIR).....	101
7.5.18	Port P1 Input/Output Configuration Register (P1DIR).....	102
7.5.19	Port P2 Input/Output Configuration Register (P2DIR).....	102
7.5.20	Port P3 Input/Output Configuration Register(P3DIR).....	103

7.5.21	Port P0 Input Value Register (P0IN)	104
7.5.22	Port P1 Input Value Register(P1IN)	105
7.5.23	Port P2 Input Value Register (P2IN)	106
7.5.24	Port P3 Input Value Register (P3IN)	106
7.5.25	Port P0 Output Value Register (P0OUT).....	107
7.5.26	Port P1 Output Value Register (P1OUT).....	108
7.5.27	Port P2 Output Value Register (P2OUT).....	109
7.5.28	Port P3 Output Value Register (P3OUT).....	109
7.5.29	Port P0 Digital Analog Register (P0ADS).....	110
7.5.30	Port P1 Digital Analog Register (P1ADS).....	111
7.5.31	Port P2 Digital Analog Register (P2ADS).....	112
7.5.32	Port P3 Digital Analog Register (P3ADS).....	112
7.5.33	Port P0 Drive Capability Configuration Register (P0DR).....	113
7.5.34	Port P1 Drive Capability Configuration Register (P1DR).....	114
7.5.35	Port P2 Drive Capability Configuration Register (P2DR).....	115
7.5.36	Port P3 Drive Capability Configuration Register (P3DR).....	115
7.5.37	Port P0 Pulling-up Enabling Configuration Register (P0PU).....	116
7.5.38	Port P01 Pulling-up Enabling Configuration Register (P1PU)	117
7.5.39	Port P02 Pulling-up Enabling Configuration Register (P2PU)	117
7.5.40	Port P03 Pulling-up Enabling Configuration Register (P3PU)	118
7.5.41	Port P0 Pulling-down Enabling Configuration Register (P0PD).....	119
7.5.42	Port P1 Pulling-down Enabling Configuration Register (P1PD).....	120
7.5.43	Port P2 Pulling-up Enabling Configuration Register (P2PD).....	121
7.5.44	Port P3 Pulling-up Enabling Configuration Register (P3PD).....	121
7.5.45	Port P0 Open Drain Output Configuration Register (P0OD)	122
7.5.46	Port P1 Open Drain Output Configuration Register (P1OD)	123
7.5.47	Port P2 Open Drain Output Configuration Register (P2OD)	124
7.5.48	Port P3 Open Drain Output Configuration Register (P3OD)	124
7.5.49	Port P0 High Level Interrupt Enabling Configuration Register (P0HIE).....	126
7.5.50	Port P1 High Level Interrupt Enabling Configuration Register (P1HIE).....	126
7.5.51	Port P2 High Level Interrupt Enabling Configuration Register (P2HIE).....	127
7.5.52	Port P3 High Level Interrupt Enabling Configuration Register (P3HIE).....	129
7.5.53	Port P0 Low Level Interrupt Enabling Configuration Register (P0LIE)	129
7.5.54	Port P1 Low Level Interrupt Enabling Configuration Register (P1LIE)	130
7.5.55	Port P2 Low Interrupt Enabling Configuration Register (P2LIE).....	131
7.5.56	Port P3 Low Interrupt Enabling Configuration Register(P3LIE)	132
7.5.57	Port P0 Low Rising Edge Interrupt Enabling Configuration Register (P0RIE).....	132
7.5.58	Port P1 Rising Edge Interrupt Enabling Configuration Register(P1RIE)	133
7.5.59	Port P2 Rising Edge Interrupt Enabling Configuration Register(P2RIE)	134
7.5.60	Port P3 Rising Edge Interrupt Enabling Configuration Register (P3RIE)	135
7.5.61	Port P0 Falling Edge Interrupt Enabling Configuration Register (P0FIE).....	135
7.5.62	Port P1 Falling Edge Interrupt Enabling Configuration Register(P1FIE)	136
7.5.63	Port P2 Falling Edge Interrupt Enabling Configuration Register (P2FIE).....	137
7.5.64	Port P3 Falling Edge Interrupt Enabling Configuration Register (P3FIE).....	137
7.5.65	Port P0 Interrupt State Register (P0_STAT)	139
7.5.66	Port P1 Interrupt State Register (P1_STAT)	139

7.5.67	Port P2 interrupt state register(P2_STAT).....	140
7.5.68	Port P3 Interrupt State Register (P3_STAT)	141
7.5.69	Port P0 Interrupt Clearing Register (P0_ICLR)	142
7.5.70	Port P1 Interrupt State Register (P1_ICLR).....	142
7.5.71	Port P2 Interrupt Clearing Register(P2_ICLR)	143
7.5.72	Port P3 Interrupt Clearing Register(P3_ICLR)	144
7.5.73	Port Additional Function Configuration Register 0 (GPIO_CTRL0).....	145
7.5.74	Port Additional Function Configuration Register 1 (GPIO_CTRL1).....	146
7.5.75	Port Additional Function Configuration Register 2(GPIO_CTRL2).....	147
7.5.76	Port Additional Function Configuration Register 3(GPIO_CTRL3).....	148
7.5.77	Port Additional Function Configuration Register 4(GPIO_CTRL4).....	150
8	FLASH CONTROLLER	151
8.1	INTRODUCTION.....	151
8.2	STRUCTURE DIAGRAM.....	151
8.3	FUNCTION DESCRIPTION	151
8.3.1	Erasing Operation.....	151
8.3.2	Writing Operation	152
8.3.3	Reading Operation.....	153
8.4	ERASE TIME	153
8.5	READING WAITING CYCLE	154
8.6	ERASE PROTECTION.....	154
8.6.1	Erase Protection Bit.....	154
8.6.2	PC Address Erasing Protection	154
8.7	REGISTER WRITING PROTECTION.....	154
8.8	REGISTER	155
9	RAM CONTROLLER	163
9.1	INTRODUCTION.....	163
9.2	FUNCTIONAL DESCRIPTION	163
9.2.1	Bit Width for Reading and Writing	163
9.2.2	Parity.....	163
9.3	REGISTER	163
9.3.1	Controlling Register (RAM_CR)	163
9.3.2	Parity Error Address Register (RAM_ERRADDR)	164
9.3.3	Error Interrupt Flag Register (RAM_IFR)	165
9.3.4	Error Interrupt Flag Clearing Register(RAM_ICLR)	165
10	BASE TIMER.....	166
10.1	BASE TIMER INTRODUCTION.....	166
10.2	BASE TIMER FUNCTION DESCRIPTION	166
10.2.1	Counting Function.....	168
10.2.2	Timing Function.....	169
10.2.3	Buzzer Function	169
10.3	BASE TIMER INTERCONNECT.....	169
10.3.1	GATE Interconnect	169
10.3.2	Toggle Output Interconnect	170

10.4	BASE TIMER REGISTER DESCRIPTION	170
10.4.1	16-bit Mode Counting Register (TIMx_ARR).....	170
10.4.2	16-bit Mode Counting Register (TIMx_CNT).....	171
10.4.3	32-bit Mode Count Register (TIMx_CNT32).....	171
10.4.4	Control Register (TIMx_CR).....	172
10.4.5	Interrupt Flag Register (TIMx_IFR)	172
10.4.6	Interrupt Flag Clearing Register (TIMx_ICLR).....	173
11	LOW POWER TIMER (LPTIMER)	174
11.1	LPTIMER INTRODUCTION.....	174
11.2	LPTIMER FUNCTION DESCRIPTION	174
11.2.1	Timing function	175
11.2.2	Count function	176
11.3	LPTIMER INTERCONNECT.....	176
11.3.1	GATE Interconnect	176
11.3.2	EXT Interconnect	176
11.3.3	Toggle Output Interconnect	176
11.4	LPTIMER REGISTER DESCRIPTION	176
11.4.1	Counter Value Register (TIM3_CNT).....	176
11.4.2	Reload Register (TIM3_ARR).....	177
11.4.3	Control Register (TIM3_CR)	177
11.4.4	Interrupt Flag Register (TIM3_IFR)	178
11.4.5	Interrupt Flag Clearing Register (TIM3_ICLR).....	179
12	PROGRAMMABLE COUNTING ARRAY (PCA).....	180
12.1	PCA INTRODUCTION.....	180
12.2	PCA FUNCTION DESCRIPTION	180
12.2.1	PCA Timer/Counter.....	181
12.2.2	PCA Capture Function	182
12.2.3	PCA Compare Function	183
12.3	INTERCONNECTS AND CONTROLS BETWEEN PCA MODULE AND OTHER MODULES	187
12.3.1	ECI Interconnects	187
12.3.2	PCACAP0	187
12.3.3	PCACAP1	187
12.3.4	PCACAP [4:2].....	188
12.4	PCA REGISTER DESCRIPTION	188
12.4.1	Control Register (PCA_CCON)	189
12.4.2	Mode Register (PCA_CMOD).....	191
12.4.3	Count Register (PCA_CNT)	191
12.4.4	Interrupt Clearing Register (PCA_ICLR)	192
12.4.5	Compare/capture Mode Register (PCA_CCAPM0~4)	193
12.4.6	Higher 8-bit of Compare/Capture Data Register (PCA_CCAP0~4H)	194
12.4.7	Lower 8-bit of the Compare/capture Data Registe (PCA_CCAP0~4L)	194
12.4.8	16-bit Register for Compare/Capture (PCA_CCAP0~4).....	195
12.4.9	High Speed Output Flag Register for Comparator (PCA_CCAPO)	195
13	ADVANCED TIMER.....	197

13.1	ADVANCED TIMER INTRODUCTION.....	197
13.2	ADVANCED TIMER DESCRIPTION.....	198
13.2.1	Basic Actions	198
13.2.1.1	Basic Waveform Mode	198
13.2.1.2	Compare Output.....	199
13.2.1.3	Capture Inut.....	200
13.2.2	Clock Source Selection.....	201
13.2.3	Counting Direction	201
13.2.3.1	Sawtooth Counting Direction.....	201
13.2.3.2	Triangle wave counting direction	201
13.2.4	Digital Filtering.....	201
13.2.5	Software Synchronization.....	202
13.2.5.1	Software Synchronous Start.....	202
13.2.5.2	Software Synchronous Stop.....	203
13.2.5.3	Software Synchronous Clearing	203
13.2.6	Hardware Synchronization	203
13.2.6.1	Synchronous Startup by Hardware	203
13.2.6.2	Hardware Synchronous Stop	203
13.2.6.3	Hardware Synchronous Clearing.....	203
13.2.6.4	Hardware Synchronous Capture	203
13.2.6.5	Synchronous Count by Hardware	203
13.2.7	Buffer Function	204
13.2.7.1	Buffer Transfer Time	205
13.2.8	General Purpose PWM Output.....	205
13.2.8.1	PWM Spread Frequency Output	205
13.2.8.2	Independent PWM Output.....	206
13.2.8.3	Complementary PWM output.....	206
13.2.8.3.1	Setting GCMBR Complementary PWM Output by Software.....	206
13.2.8.3.2	Setting GCMBR Complementary PWM Output by Hardware	207
13.2.8.4	Multiphase PWM Output.....	208
13.2.9	Quadrature Encoding Count	209
13.2.9.1	Position Count Mode	210
13.2.9.1.1	Basic Count.....	210
13.2.9.1.2	Phase Difference Count.....	210
13.2.9.1.3	Direction Count.....	211
13.2.9.2	Revolution Mode	212
13.2.9.2.1	Z-phase Count.....	212
13.2.9.2.2	Position Overflow Count	212
13.2.9.2.3	Mixed Count.....	213
13.2.9.2.4	Z-phase Action Disabling	213
13.2.10	Response per Interval Cycles	214
13.2.11	Protection Mechanism.....	215
13.2.12	Interrupt Description	215
13.2.12.1	Count Compare Match Interrupt.....	215
13.2.12.2	Count Cycle Match Interrupt.....	215
13.2.12.3	Dead Time Error Interrupt.....	215

13.2.13	Brake Protection.....	215
13.2.13.1	Port Brake and Software Brake	215
13.2.13.2	Low Power Mode Automatic Braking	216
13.2.13.3	Brake Per Output Level Both High or Both Low Case	216
13.2.13.4	VC Brake	217
13.2.14	Internal Interconnect.....	217
13.2.14.1	Interrupt Triggering Output	217
13.2.14.2	AOS Triggering.....	217
13.2.14.3	Port Triggering TRIGA-TRIGD	218
13.2.14.4	Interconnect Between Compare Output VC and Advanced Timer	219
13.2.14.5	Interconnect Between UART and Advanced Timer	219
13.3	REGISTER DESCRIPTION	219
13.3.1	General Count Reference Value Register (TIMx_CNTER)	221
13.3.2	General Purpose Period Reference Value Register (TIMx_PERAR).....	221
13.3.3	General Purpose Period Reference Value Buffer Register (TIMx_PERBR).....	221
13.3.4	General Purpose Compare Reference Value Register (TIMx_GCMAR-GCMDR)	222
13.3.5	Dead Time Reference Value Register (TIMx_DTUAR- DTDAR).....	222
13.3.6	General Purpose Control Register (TIMx_GCONR).....	223
13.3.7	Interrupt Control Register (TIMx_ICONR).....	224
13.3.8	Port Control Register (TIMx_PCONR).....	225
13.3.9	Buffer Control Register (TIMx_BCONR).....	227
13.3.10	Dead Time Control Register (TIMx_DCONR).....	229
13.3.11	Filtering Control Register (TIMx_FCONR)	229
13.3.12	Valid Period Register (TIMx_VPERR)	231
13.3.13	State Flag Register (TIMx_STFLR).....	232
13.3.14	Hardware Start Event Selection Register (TIMx_HSTAR).....	233
13.3.15	Hardware Stopping Event Selection Register (TIMx_HSTPR)	235
13.3.16	Hardware Clearing Event Selection Register (TIMx_HCELRL)	236
13.3.17	Hardware Capture A Event Selection Register (TIMx_HCPAR)	238
13.3.18	Hardware Capture B Event Selection Register (TIMx_HCPBR).....	240
13.3.19	Hardware Count UP Register (TIMx_HCUPR)	241
13.3.20	Hardware Count Down Register (TIMx_HCDOR).....	243
13.3.21	Software Synchronous Start Register (TIMx_SSTAR).....	245
13.3.22	Software Synchronous Stopping Register (TIMx_SSTPR)	246
13.3.23	Software Synchronous Clearing Register (TIMx_SCLRR)	247
13.3.24	Interrupt Flag Register (TIMx_IFR)	247
13.3.25	Interrupt Flag Clearing Register (TIMx_ICLR)	248
13.3.26	Spread Frequency and Interrupt Triggering Selection (TIMx_CR)	249
13.3.27	AOS Selection Control Register (TIMx_AOSSR).....	250
13.3.28	AOS Selection Control Register Flag Clearing (TIMx_AOSCL)	251
13.3.29	Port Brake Control Register (TIMx_PTBK).....	252
13.3.30	Port Triggering Control Register (TIMx_TTRIG)	253
13.3.31	AOS Triggering Control Register (TIMx_ITRIG)	254
13.3.32	Port Brake Polarity Control Register (TIMx_PTBK)	254
14	REAL-TIME CLOCK (RTC).....	256
14.1	REAL-TIME CLOCK INTRODUCTION.....	256

14.2 REAL-TIME CLOCK FUNCTION DESCRIPTION	257
14.2.1 Power-on Settings	257
14.2.2 RTC Count Start Setting	257
14.2.3 System Low Power Mode Switching.....	257
14.2.4 Reading Count Register.....	257
14.2.5 Writing Count Register.....	258
14.2.6 Alarm Settings	258
14.2.7 1 Hz Output.....	258
14.2.8 Clock Deviation Compensation.....	259
14.3 RTC INTERRUPT	260
14.3.1 RTC Alarm Interrupt.....	260
14.3.2 RTC Period Interrupt	261
14.4 RTC REGISTER DESCRIPTION	261
14.4.1 Control Register 0 (RTC_CR0).....	261
14.4.2 Control Register 1(RTC_CR1).....	263
14.4.3 Second Count Register (RTC_SEC).....	264
14.4.4 Minute Count Register (RTC_MIN)	264
14.4.5 Hour Count Register (RTC_HOUR)	265
14.4.6 Day Count Register (RTC_DAY)	266
14.4.7 Week Count Register (RTC_WEEK).....	267
14.4.8 Month Count Register (RTC_MON)	268
14.4.9 Year Count Register (RTC_YEAR)	268
14.4.10 Minute Alarm Register (RTC_ALMMIN)	269
14.4.11 Hour Alarm Register (RTC_ALMHOUR)	269
14.4.12 Week Alarm Register (RTC_ALM WEEK).....	270
14.4.13 Clock Deviation Compensation Register (RTC_COMPEN)	271
15 WATCHDOG WDT	273
15.1 WDT INTRODUCTION.....	273
15.2 WDT FUNCTION DESCRIPTION	273
15.2.1 Interrupt after WDT Overflow	273
15.2.2 Reset after WDT overflow.....	274
15.3 WDT REGISTER DESCRIPTION	274
15.3.1 WDT Clearing Control Register (WDT_RST)	274
15.3.2 WDT_CON Register.....	275
16 UART	276
16.1 OVERVIEW.....	276
16.2 BLOCK DIAGRAM.....	276
16.3 OPERATING MODE	276
16.3.1 Mode 0 (Synchronous Mode, Half Duplex)	276
16.3.1.1 Transmitting data	277
16.3.1.2 Receiving Data.....	277
16.3.2 Mode1 (Asynchronous Mode, Full-duplex)	277
16.3.2.1 Transmitting Data.....	278
16.3.2.2 Receiving Data.....	278
16.3.3 Mode 2 (Asynchronous Mode, Full-duplex)	278

16.3.3.1	Transmitting Data.....	278
16.3.3.2	Receiving Data.....	279
16.3.4	Mode 3 (Asynchronous Mode, Full-duplex)	279
16.4	BAUD RATE PROGRAMMING	279
16.4.1	Mode 0.....	279
16.4.2	Mode 1/3	279
16.4.3	Mode 2.....	280
16.5	FRAME ERROR DETECTION.....	280
16.6	MULTIPROCESSOR COMMUNICATION	280
16.7	AUTOMATIC ADDRESS RECOGNITION.....	280
16.7.1	Given Address	281
16.7.2	Broadcast Address.....	281
16.8	TRANSCEIVER BUFFER	281
16.8.1	Receiving Buffer.....	281
16.8.2	Transmitting Buffer.....	282
16.9	FREQUENCY DIVIDING SETTINGS FOR DIFFERENT BAUD RATES	282
16.10	REGSITER	287
16.10.1	Data Register (UARTx_SBUF)	287
16.10.2	Control Register (UARTx_SCON)	289
16.10.3	Address Register (UARTx_SADDR).....	289
16.10.4	Address Mask Register (UARTx_SADEN).....	290
16.10.5	Flag Register (UARTx_ISR).....	290
16.10.6	Flag Clearing Register (UARTx_ICR).....	291
17	LPUART	292
17.1	OVERVIEW.....	292
17.2	STRUCTURE DIAGRAM.....	292
17.3	OPERATING MODE	292
17.3.1	Mode 0 (Synchronous Mode, Half Duplex).....	293
17.3.1.1	Transmitting Data.....	293
17.3.1.2	Receiving Data.....	293
17.3.2	Mode 1 (Asynchronous Mode, Full-duplex)	294
17.3.2.1	Transmitting Data.....	294
17.3.2.2	Receiving Data.....	294
17.3.3	Mode 2 (Asynchronous Mode, Full-duplex)	294
17.3.3.1	Transmitting Data.....	295
17.3.3.2	Receiving Data.....	295
17.3.4	Mode 3 (Asynchronous Mode, Full-duplex)	295
17.4	BAUD RATE PROGRAMMING	295
17.4.1	Mode 0.....	295
17.4.2	Mode 1/3	296
17.4.3	Mode2.....	296
17.5	FRAME ERROR DETECTION.....	297
17.6	MULTIPROCESSOR COMMUNICATION	297
17.7	AUTOMATIC ADDRESS RECOGNITION.....	297
17.7.1	Given Address	297
17.7.2	Broadcast Address.....	297

17.7.3	Example	298
17.8	TRANCEIVER BUFFER	298
17.8.1	Receiving Buffer.....	298
17.8.2	Transmitting Buffer.....	298
17.9	REGISTER	299
17.9.1	Data Register (LPUART_SBUF).....	299
17.9.2	Control Register (LPUART_SCON)	300
17.9.3	Address Register (LPUART_SADDR)	301
17.9.4	Address Mask Register (LPUART_SADEN).....	301
17.9.5	Interrupt Flag Register (LPUART_ISR)	302
17.9.6	Interrupt Flag Clearing Register (LPUART_ICR)	303
18	I2C	304
18.1	I2C INTRODUCTION.....	304
18.2	I2C MAJOR FEATURES	304
18.3	I2C PROTOCOL DESCRIPTION.....	304
18.3.1	Data Transmission on the I2C Bus.....	305
18.3.2	START Signal or Repeated START Signal	305
18.3.3	Slave Address Transfer.....	306
18.3.4	Data Transfer.....	306
18.4	I2C FUNCTION DESCRIPTION.....	307
18.4.1	I2C Operating Mode	308
18.4.2	Arbitration and Synchronization Logic	308
18.4.3	Serial Clock Generator.....	309
18.4.4	Input Filter	310
18.4.5	Address Comparator	310
18.4.6	Interrupt Generator	310
18.4.7	I2C Master Transmitting Mode.....	310
18.4.8	I2C Master Receiving Mode	314
18.4.9	I2C Slave Receiving Mode	317
18.4.10	I2C Slave Transmitting Mode.....	320
18.4.11	I2C Other Miscellaneous State.....	323
18.5	I2C OPERATION MODE	323
18.5.1	Initialization Procedure.....	323
18.5.2	Port Configuration Procedure.....	324
18.5.3	Start the Master Transmission Function.....	324
18.5.4	Start the Master Receiving Function.....	324
18.5.5	I2C Interrupt Procedure	324
18.5.6	State with No Specified Mode.....	325
18.5.7	Master Transmitter State	325
18.5.8	Master Receiving States	326
18.5.9	Slave Receiver State	327
18.5.10	Slave Transmitter State.....	329
18.6	I2C REGISTER DESCRIPTION	330
18.6.1	I2C Baud Rate Counter Enabling Register (I2C_TMRUN).....	331
18.6.2	I2C Baud Rate Counter Configuration Register (I2C_TM)	331
18.6.3	I2C Configuration Register (I2C_CR)	332

18.6.4	I2C Data Register (I2C_DATA)	332
18.6.5	I2C Address Register (I2C_ADDR)	333
18.6.6	I2C State Register (I2C_STAT).....	333
19	SPI	334
19.1	INTRODUCTION TO SPI	334
19.2	SPI MAJOR FEATURES	334
19.3	SPI FUNCTION DESCRIPTION.....	334
19.3.1	SPI Master Mode	334
19.3.1.1	Operation Flow	334
19.3.1.2	Timing	335
19.3.2	SPI Slave Mode	335
19.3.2.1	Operation Flow	336
19.3.2.2	Timing	336
19.4	SPI INTERRUPT.....	337
19.5	SPI MULTI-MASTER/MULTI-SLAVE MODE.....	337
19.6	SPI REGISTER DESCRIPTION.....	337
19.6.1	SPI0 Configuration Register (SPI0_CR)	338
19.6.2	SPI0 Chip Selection Configuration Register(SPI0_SSN)	339
19.6.3	SPI0 State Register (SPI0_STAT).....	339
19.6.4	SPI0 Data Register (SPI0_DATA)	340
20	CLK_TRIM.....	341
20.1	INTRODUCTION TO CLK_TRIM	341
20.2	CLK_TRIM FEATURES	341
20.3	CLK_TRIM FUNCTION DESCRIPTION.....	341
20.3.1	CLK_TRIM Calibration Mode	341
20.3.1.1	Operation Flow	341
20.3.2	CLK_TRIM Monitoring Mode.....	342
20.3.2.1	Operation Flow	342
20.4	CLK_TRIM REGISTER DESCRIPTION.....	343
20.4.1	Configuration Register(CLKTRIM_CR)	343
20.4.2	Reference Counter Initial Value Configuration Register (CLKTRIM_REFCON)	344
20.4.3	Reference Counter Value Configuration Register (CLKTRIM_REFCNT)	344
20.4.4	Calibration Counter Value Register (CLKTRIM_CALCNT).....	345
20.4.5	Interrupt Flag Register (CLKTRIM_IFR).....	345
20.4.6	Interrupt Flag Clearing Register (CLKTRIM_ICLR)	346
20.4.7	Calibration Counter Overflow Value Configuration Register (CLKTRIM_CALCON)	346
21	CRC16	347
21.1	OVERVIEW.....	347
21.2	FUNCTIONAL DESCRIPTION	347
21.2.1	CRC Encoding Mode.....	347
21.2.2	CRC Verification Mode.....	348
21.3	REGISTER	348
21.3.1	CRC Result Register (CRC_RESULT)	348
21.3.2	CRC Data Register (CRC_DATA)	349

22 ADC MODULE.....	350
22.1 MODULE INTRODUCTION.....	350
22.2 ADC BLOCK DIAGRAM	351
22.3 CONVERSION TIMING AND SPEED	351
22.4 SINGLE CONVERSION MODE	352
22.5 CONTINUOUS CONVERSION MODE	353
22.6 CONTINUOUS CONVERSION ACCUMULATION MODE	354
22.7 ADC CONVERSION RESULTS COMPARISON	356
22.8 ADC INTERRUPT.....	356
22.9 MEASURING AMBIENT TEMPERATURE VIA TEMPERATURE SENSOR	357
22.10 ADC MODULE REGISTER.....	358
22.10.1 ADC Configuration Register 0 (ADC_CR0)	358
22.10.2 ADC Configuration Register 1 (ADC_CR1)	360
22.10.3 ADC Configuration Register 2 (ADC_CR2)	364
22.10.4 ADC Conversion Result for Channel 0 (ADC_result0).....	365
22.10.5 ADC Conversion Result for Channel 1 (ADC_result1).....	365
22.10.6 ADC Conversion Result for Channel 2 (ADC_result2).....	365
22.10.7 ADC Conversion Result for Channel 3 (ADC_result3).....	366
22.10.8 ADC Conversion Result for Channel 4 (ADC_result4).....	366
22.10.9 ADC Conversion Result for Channel 5 (ADC_result5).....	367
22.10.10 ADC Conversion Result for Channel 6 (ADC_result6).....	367
22.10.11 ADC Conversion Result for Channel 7 (ADC_result7).....	368
22.10.12 ADC Conversion Result for Channel 8 (ADC_result8).....	368
22.10.13 ADC Conversion Result Accumulated Value (ADC_result_acc).....	369
22.10.14 ADC Comparison Higher Threshold (ADC_HT).....	369
22.10.15 ADC Comparison Lower Threshold (ADC_LT).....	369
22.10.16 Interrupt Flag Register (ADC_IFR).....	370
22.10.17 Interrupt Flag Clearing Register (ADC_ICLR)	371
22.10.18 ADC Restult (ADC_result).....	372
23 VC MODULE	373
23.1 INTRODUCTION TO ANALOG VOLTAGE COMPARATOR (VC)	373
23.2 VOLTAGE COMPARATOR BLOCK DIAGRAM	373
23.3 SETUP/RESPONSE TIME.....	374
23.4 FILTERING TIME	374
23.5 HYSTERESIS FUNCTION.....	374
23.6 VC REGISTER.....	375
23.6.1 VC Configuration Register (VC_CR).....	375
23.6.2 VC0 Configuration Register (VC0_CR)	376
23.6.3 VC1 Configuration Register (VC1_CR)	378
23.6.4 VC0 Output Configuration Register (VC0_OUT_CFG).....	379
23.6.5 VC1 Configuration Register (VC1_OUT_CFG)	381
23.6.6 VC Interrupt Register (VC_IFR).....	382
24 LVD MODULE	383
24.1 INTRODUCTION TO LVD	383

24.2	LVD BLOCK DIAGRAM	383
24.3	DIGITAL FILTERING	383
24.4	HYSTeresis FUNCTION.....	384
24.5	CONFIGURATION EXAMPLES.....	384
24.5.1	LVD Configured as Low Voltage Reset	384
24.5.2	LVD Configured as Voltage Change Interrupt	385
24.6	LVD REGISTER	385
24.6.1	LVD Configuration Register (LVD_CR).....	385
24.6.2	LVD Interrupt Register (LVD_IFR).....	387
25	OTHER ANALOG REGISTER	389
25.1	BGR CONFIGURATION REGISTER (BGR_CR).....	389
26	SWD DEBUG INTERFACE	390
26.1	SWD AUTHORIZATION KEY OPERATING PROCESS	390
26.2	ADDITIONAL FUNCTION FOR SWD DEBUG	390
26.3	DEBUG MODE MODULE OPERATING STATE CONTROL (DEBUG_ACTIVE)	390
27	APPENDIX A: SYSTICK TIMER	393
27.1	INTRODUCTION TO SysTICK TIMER.....	393
27.2	SYS TICK SETUP.....	393
27.3	SYS TICK REGISTER.....	394
27.3.1	SysTick Control and State Register (CTRL)	394
27.3.2	SysTick Reload Register (LOAD)	395
27.3.3	SysTick Current Value Register (VAL)	395
28	REVISE HISTORY	396
29	CONTACTS	397

1 Function Modules

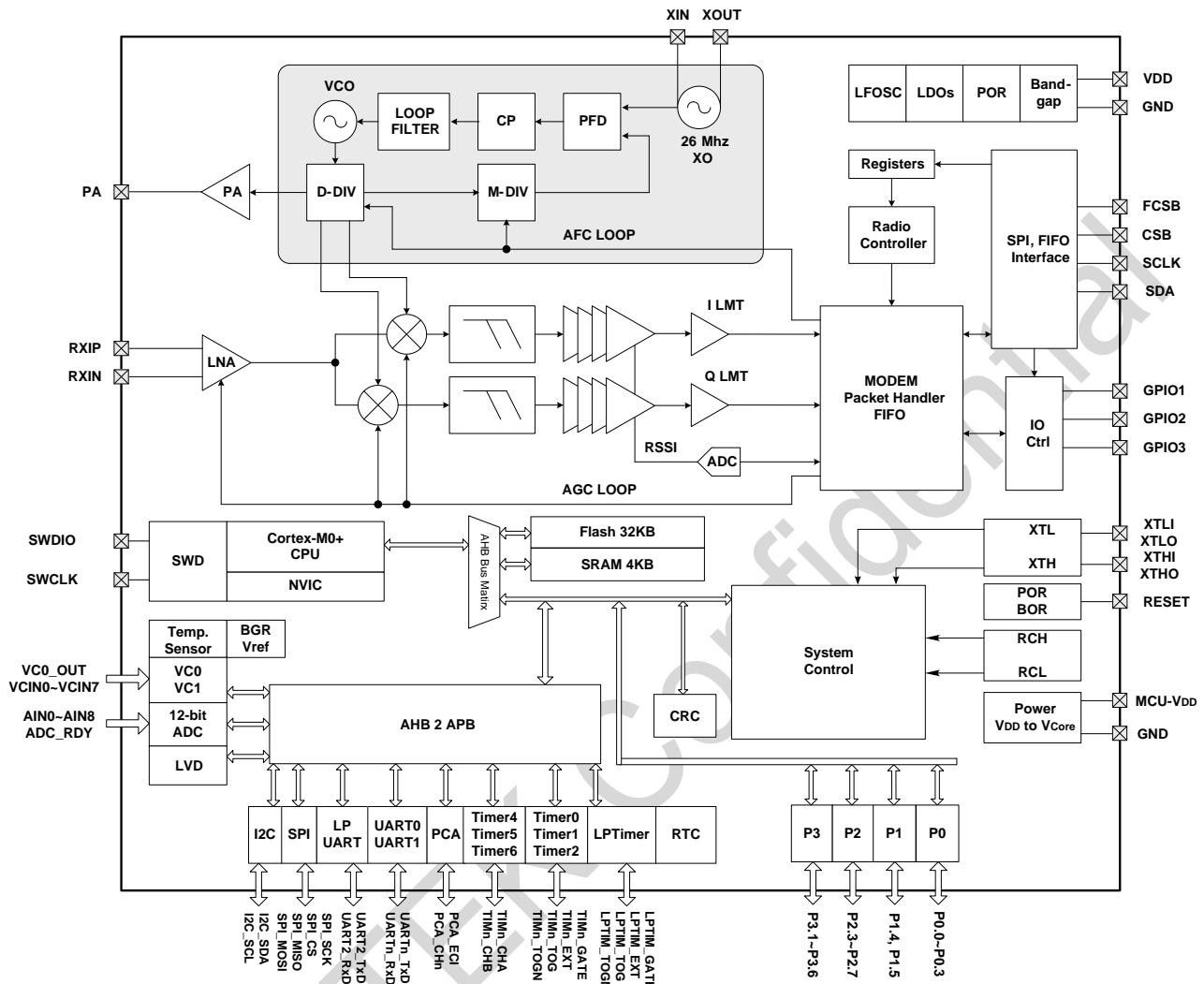


Figure 1. Chip Block Diagram

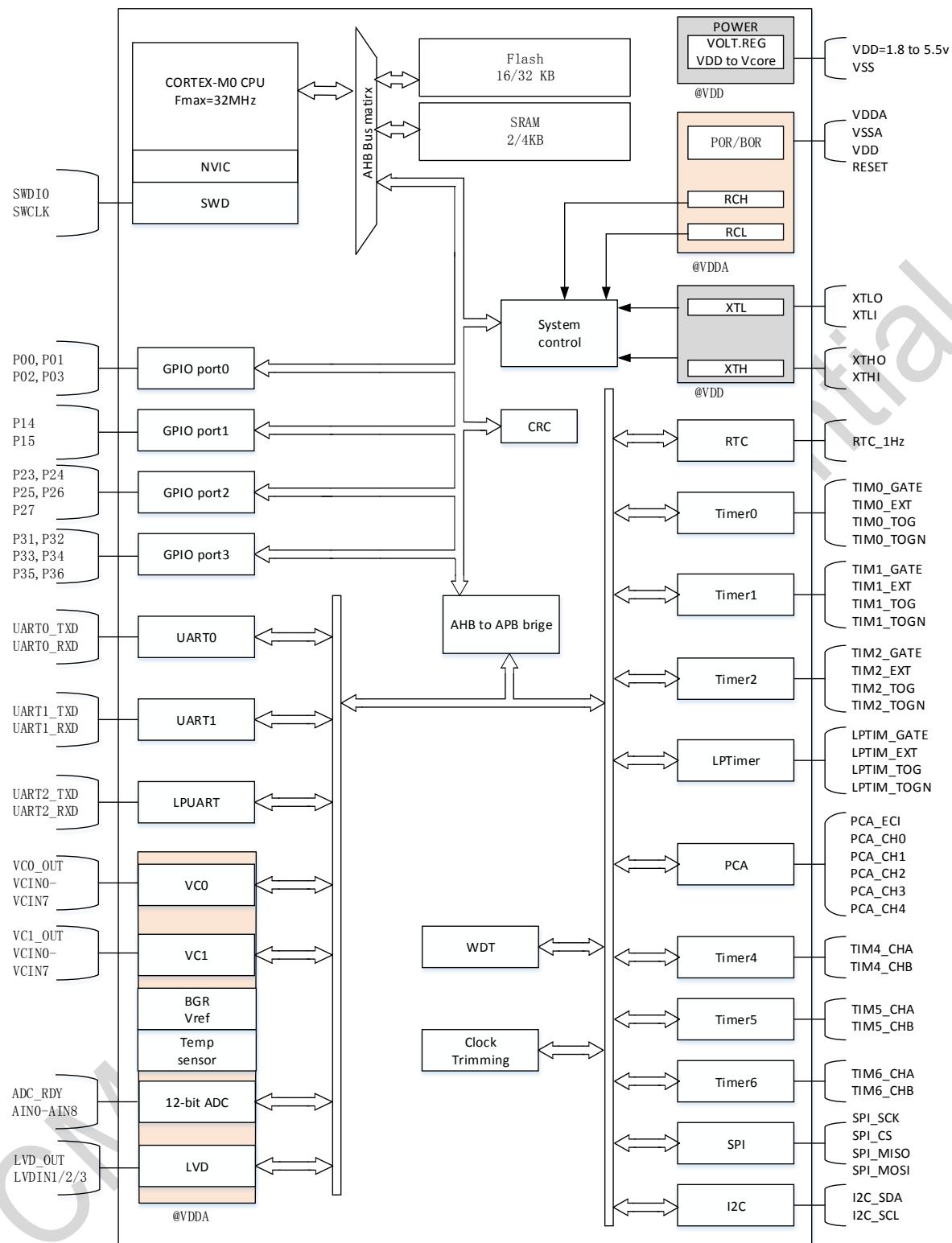


Figure 2. Function Modules of Controller Part

1.1 32-Bit Cortex-M0+ Core

The ARM® Cortex®-M0+ processor is based on the Cortex-M0 and includes a 32-bit RISC processor with computing power of 0.95 Dhrystone MIPS/MHz. It applies new designs which improve debug and tracking capabilities, reduce the number of inter-process communication, enhance two-stage pipeline for Flash access and employ energy saving techniques. In addition, the Cortex-M0+ processor fully supports the integrated Keil & IAR debugger.

The Cortex-M0+ includes a hardware debug circuit that supports the 2-pin SWD debug interface.

ARM Cortex-M0+ features are listed in the below table.

Instruction Set	Thumb / Thumb-2
Pipeline	Two-stage pipeline
Performance Efficiency	2.46 CoreMark / MHz
Performance Efficiency	0.95 DMIPS / MHz in Dhrystone
Interrupt	32 fast Interrupts
Interrupt Priority	Configurable 4-level interrupt priority
Enhanced Instruction	Single cycle 32-bit multiplier
Debug	Serial-wire debug port with 4 hard break points and 2 watch points

1.2 32 k Byte Flash

Built in with a fully-integrated Flash controller, it eliminates the need for external high-voltage inputs, supporting being programmed by high-voltage internal circuits. More over, it offers ISP, IAP and ICP functions as well.

1.3 4 k Byte RAM

RAM data is retained based on users' choices of different ultra-low power modes. With hardware parity bit, if the data is accidentally destroyed, the hardware circuit will immediately generate an interrupt once the data is read, which helps ensuring the system reliability.

1.4 Clock System

- A high-precision internal clock RCH with a frequency of 4 ~ 24 MHz. The wake-up time from low power mode to active mode is 3 us at 16 MHz. The frequency error over the full voltage and temperature range is less than $\pm 2.5\%$, eliminating the need for an expensive high frequency crystal.
- An external crystal oscillator XTH with a frequency range of 4 ~ 32 MHz.
- An external crystal oscillator XTL at 32.768 KHz, mainly for providing RTC real-time clock.
- An internal clock RCL at 32.768/38.4 KHz.

1.5 Operating mode

- **Active Mode:** CPU runs and peripheral function modules run.
- **Sleep Mode:** CPU stops running and peripheral function modules run.
- **Deep Sleep Mode:** CPU stops running, the high speed clock stops running, and low power function modules run.

1.6 Hardware Real Time Clock (RTC)

RTC (Real Time Counter) is a register supporting BCD data. It uses a 32,768 Hz crystal oscillator as its clock to implement the perpetual calendar function. The interrupt period can be configured per year/month/day/hour/minute/second with supports of 24/12 hour modes and automatic correction of leap years by hardware. With support of precision compensation function (using the internal temperature sensor or external temperature sensor to implement precision compensation), it can offer an optimal precision of 0.96 ppm. It provides adjusting the year/month/day/hour/minute/second by the +1/-1 operation in software with a minimum adjustable precision of 1 second.

The RTC calendar recorder, which indicates the time and date, does not clear the reserved value in case the MCU is reset caused by external factors, which makes it the best choice for measuring instruments that requires a permanent high-precision real-time clock.

1.7 General Purpose IO Port

Up to 16 GPIO ports are available with some of them multiplexed with analog ports. Each port is controlled by an independent control register bit. It supports features such as edge-triggered / level-triggered interrupts, waking up the MCU from a variety of ultra-low power modes to active mode, CMOS push-pull output and open-drain output, and built-in pull-up resistor and pull-down resistor and the Schmitt trigger input filtering function. The output drive capability is configurable and supports up to 12 mA of current drive capability. All the 16 GPIO ports support external asynchronous interrupts.

1.8 Interrupt controller

The Cortex-M0+ processor provides a built-in nested vectored interrupt controller (NVIC) that supports up to 32 interrupt request (IRQ) inputs. It supports 4 interrupt priorities to handle complex logic using real-time control and interrupt handling.

The 32 interrupt entry vector addresses are as follows.

Table 2. Interrupt Entry Vector Addresses

interrupt vector#	Interrupt Source
0	GPIO_P0
1	GPIO_P1
2	GPIO_P2
3	GPIO_P3
4	-
5	-
6	UART0
7	UART1
8	UART2

interrupt vector#	Interrupt Source
9	-
10	SPI
11	-
12	I2C
13	-
14	Timer 0
15	Timer 1
16	Timer 2
17	LPTimer
18	Timer 4
19	Timer 5
20	Timer 6
21	PCA
22	WDT
23	RTC
24	ADC
25	-
26	VC0
27	VC1
28	LVD
29	-
30	RAM FLASH fault
31	Clock trim

1.9 Controller Reset

This product has 7 reset signal sources. Each reset signal allows the CPU to re-run with most registers being reset and the program counter - PC being reset and pointing to 00000000.

Table 3. Reset Signal Sources

#	Interrupt Source
0	Power-on and power-off reset POR, BOR
1	External reset pin reset

2	WDT reset
3	PCA reset
4	Hardware reset by Cortex-M0+ LOCKUP
5	Software reset by Cortex-M0+ SYSRESETREQ
6	LVD reset

1.10 Timer/Counter

Table 4. Timer/Counter

Type	Timer	Bit Width	Pre Divided Frequency	Counting Direction	PWM	Capture	Complementary Output
Base Timer	Timer0	16/32	1/2/4/8/16/ 32/64/256	Count up	NA	NA	NA
	Timer1	16/32	1/2/4/8/16/ 32/64/256	Count up	NA	NA	NA
	Timer2	16/32	1/2/4/8/16/ 32/64/256	Count up	NA	NA	NA
LP Timer	LPTimer	16	NA	Count up	NA	NA	NA
PCA	PCA	16	2/4/8/16/32	Count up	5	5	NA
Advanced Timer	Timer4	16	1/2/4/8/16/ 64/256/1024	Count up Count down Count up and down	2	2	1
	Timer5	16	1/2/4/8/16/ 64/256/1024	Count up Count down Count up and down	2	2	1
	Timer6	16	1/2/4/8/16/ 64/256/1024	Count up Count down Count up and down	2	2	1

- The Base Timer contains 3 timers, Timer 0/1/2. Timer 0/1/2 function exactly the same. Timer 0/1/2 are synchronous timers/counters that can be used as timers/counters for 16-bit auto-reload function or as 32-bit timers/counters without reload function. Timer 0/1/2 can count external pulses or implement system timing.
- LPTimer is an asynchronous 16-bit timer/counter that can be clocked/countered by internal low speed RC or external low speed crystal oscillator after the system clock is closed. It wakes up the system from low power mode by interrupt.
- PCA (Programmable Counter Array) supports up to five 16-bit capture/compare modules. This timer/counter can be used as capture/compare function for a general purpose clock count / event counter. Each module of the PCA can be

independently programmed to provide functions of input capture, output compare or pulse width modulation. In addition, module 4 has an additional watchdog timer mode.

- Advanced Timer are 3 timers, timer 4/5/6. Timer 4/5/6 are high-performance counters, which function exactly the same. They can be used to count different clock waveforms. Each timer can generate a complementary pair of PWMs or 2 separated PWM outputs that capture external inputs for pulse width or cycle measuring.

The basic function and features of Advanced Timer are as follows.

Table 5. Basic Function and Features of Advanced Timer

Waveform	Sawtooth wave, triangle wave
Basic functions	2 count directions, count-up and count-down
	Software synchronization
	Hardware synchronization
	Buffer function
	Orthogonal code count
	General purpose PWM output
	Protection mechanism
	AOS related actions
Interrupt Type	Count compare match interrupt
	Count cycle match interrupt
	Dead time error interrupt
	Short circuit monitoring interrupt

1.11 WDT (Watch Dog Timer)

WDT (Watch Dog Timer) is a configurable 20-bit timer that provides reset function in case of MCU exception. A built-in 10 k low speed clock input is used as the counter clock. It can be paused per users' desire in debug mode. WDT restart can be performed by writing a specific sequence.

1.12 Universal Asynchronous Receiver UART0, UART1 and UART2

- 2 universal asynchronous transceivers (Universal Asynchronous Receiver/Transmitter)
- One asynchronous transceiver (Low Power Universal Asynchronous Receiver/Transmitter)

1.13 Synchronous Serial Peripheral Interface (SPI)

The synchronous serial interface SPI (Serial Peripheral Interface) supports master-slave mode.

1.14 I2C Bus

It supports one I2C (Inter-Integrated Circuit) with master-slave mode. With serial synchronous clock, data can be transmitted between devices at different rates with the serial 8-bit bidirectional data transmission reaching a maximum speed of 1 Mbps.

1.15 Buzzer

3 Base Timers along with 1 LPTimer co-function to output programmable frequency that can drive Buzzer. The buzzer port provides 16 mA sink current and complementary output without additional transistors required.

1.16 Clock Calibration Circuit

The built-in clock calibration circuit calibrates the internal RC clock using an external precision crystal oscillator. On the other hand, the internal RC clock can also be used to verify whether the external crystal clock is working properly.

1.17 Unique ID Number

Each chip has a unique 16-Byte device identification number, including wafer lot information, and chip coordinate information. The ID address is 0X0010_0E70-0X0010_0E7F.

1.18 CRC16 Hardware Cyclic Redundancy Check Code

It conforms to the polynomial given in ISO/IEC13239, that is $F(x) = X^{16} + X^{12} + X^5 + 1$.

1.19 12-Bit SARADC

The 12-bit successive approximation analog-to-digital converter, which is monotonous without loss of code, has a sampling rate of 1 Msps when operating with a 24 M ADC clock. The reference voltage can be selected from the on-chip precision voltage (1.5v or 2.5v), an external input or supply voltage. The 12 input channels includes 9 external pin inputs, one internal temperature sensor voltage, one 1/3 supply voltage and one built-in BGR 1.2V voltage. A configurable input signal amplifier is built in to detect weak signals.

1.20 Voltage Comparator (VC)

It's a chip pin-voltage monitoring / comparison circuit, providing 8 configurable positive/negative external input channels, 5 internal input channels including one internal temperature sensor voltage, one built-in BGR 2.5V reference voltage, one built-in BGR 1.2V voltage and one 64-step resistor partial pressure. The VC output can be used by Timer 0/1/2, LPTimer, Advanced Timer and programmable count array (PCA) capture, gating, and external count clocks. An asynchronous interrupt can be generated based on the rising/falling edge to wakeup the MCU from low power mode. It supports configurable software anti-shake function as well.

1.21 Low Voltage Detector (LVD)

It detects the chip supply voltage or chip pin voltage with sixteen levels of voltage monitoring values (1.8 ~ 3.3 V). An asynchronous interrupt or reset can be generated based on the rising/falling edge. It provides hardware hysteresis circuit and configurable software anti-shake function.

1.22 Embedded Debug System

The embedded debug solution provides a full-featured real-time debugger co-working with sophisticated debug software such as Keil and IAR. It supports 4 hard breakpoints and multiple soft breakpoints.

1.23 High security

It's powered with encrypted embedded debug solution providing full-featured real-time debugger.

1.24 Pin Schematic

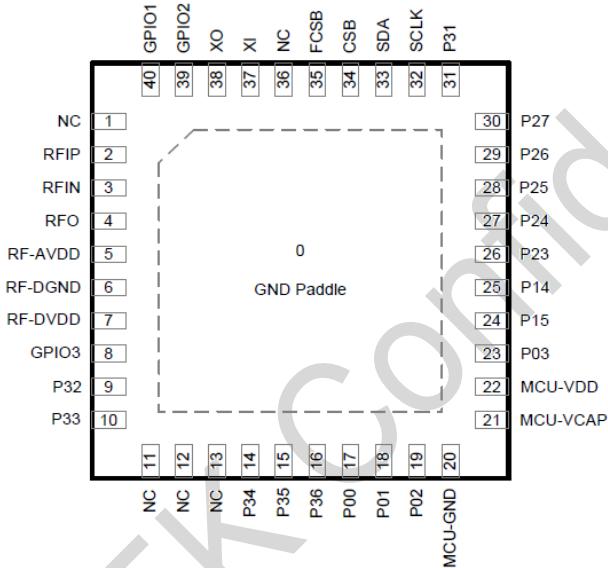


Figure 3. Chip PIN Schematic

1.25 PIN Description

Table 6. CMT2380F32 Pin Description

Pin #	Pin Name	I/O	Description
0	GND	I	Chip substrate, must connect to ground
1	NC	-	No connection
2 - 3	RFIP/RFIN	I	Differential RF signal input port
4	FRO	O	PA output
5	RF-AVDD	I	RF circuit VDD, requires to connect to a 1.8-3.6 V power supply
6	RF-DGND	I	GND, must connect to ground
7	RF-DVDD	I	Digital VDD, requires to connect to a 1.8-3.6 V power supply
8 ^[1]	GPIO3	IO	Can be configured as: CLKO, DOUT/DIN, INT2 and DCLK (TX/RX)
9	P32	IO	General purpose digital input/output pin
	TIM3_TOGN	O	LPtimer opposite reverse output output
	PCA_CH2	IO	PCA capture input/compare output 2

Pin #	Pin Name	I/O	Description
9	TIM6_CHB	IO	Timer6 capture input/compare output B
	VC1OUT	O	VC1 output
	UART1_TXD	O	UART1 TXD
	PCA_CH4	IO	PCA capture input/compare output 4
	RTC_1HZ	O	RTC1HZ output
	AIN2/VC2	I	Analog input
10	P33	IO	General purpose digital input/output pin
	UART2_RXD	I	UART2 RXD
	PCA_CH1	IO	PCA capture input / compare output 1
	TIM5_CHB	IO	Timer5 capture input / compare output B
	PCA_ECI	I	PCA external clock
	UART1_RXD	I	UART1 RXD
	XTL_OUT	O	32K Oscillating output
	TIM1_TOGN	O	Timer1 opposite reverse output
11	NC	-	Not connect
12	NC	-	Not connect
13	NC	-	Not connect
14	P34	IO	General purpose digital input/output pin
	PCA_CH0	IO	PCA capture input / compare output 0
	UART2_TXD	O	UART2 TXD
	TIM5_CHA	IO	Timer5 capture input / compare output A
	TIM0_EXT	I	Timer0 external clock input
	TIM4_CHA	IO	Timer4 capture input / compare output A
	RTC_1HZ	O	RTC1HZ output
	TIM1_TOG	O	Timer1 reverse output
15	AIN4/VC4	I	Analog input
15	P35	IO	General purpose digital input/output pin
	UART1_TXD	O	UART1 TXD
	TIM6_CHB	IO	Timer6 capture input / compare output B
	UART0_TXD	O	UART0 TXD
	TIM0_GATE	I	Timer0 gating
	TIM4_CHB	IO	Timer4 capture input / compare output B
	SPI_MISO	I	SPI Module master input slave output data signal
	I2C_SDA	IO	I2Cdata
16	AIN5/VC5	I	Analog input
16	P36	IO	General purpose digital input/output pin
	UART1_RXD		UART1 RXD
	TIM6_CHA		Timer6 capture input / compare output A
	UART0_RXD		UART0 RXD

Pin #	Pin Name	I/O	Description
	PCA_CH4		PCA capture input / compare output 4
	TIM5_CHA		Timer5 capture input / compare output A
	SPI_MOSI		SPI module master output slave input data signal
	I2C_SCL	I	I2C Clock
	AIN6/VC6/AVREF	O	Analog input
17	P00	I	Digital input
	ResetB	I	Reset input port, active low, chip reset
18	P01	IO	General purpose digital input/output pin
	UART0_RXD	I	UART0 RXD
	I2C_SDA	IO	I2C data
	UART1_TXD	O	UART1 TXD
	TIM0_TOG	O	Timer0 reverse output
	TIM5_CHB	IO	Timer5 capture input / compare output B
	SPI_SCK	O	SPI clock
	TIM2_EXT	O	Timer2 external clock
	AIN7/VC7	I	Analog input
	XTHI	I	External XTH crystal oscillator clock input
19	P02	IO	General purpose digital input/output pin
	UART0_TXD	O	UART0 TXD
	I2C_SCL	O	I2C clock
	UART1_RXD	I	UART1 RXD
	TIM0_TOGN	O	Timer0 opposite reverse output
	TIM6_CHA	IO	Timer6 capture input / compare outputA
	SPI_CS	O	SPI CS
	TIM2_GATE	I	Timer2 gating
	AIN8	I	Analog Input
	XTHO	O	External XTH crystal oscillator clock output
20	MCU-GND	I	Digital Ground
21	MCU-VCAP	O	LDO core power supply output (internal circuit only, connect 4.7uF capacitor)
22	MCU-VDD	I	Digital Power Supply
23	P03	IO	General purpose digital input/output pin
	PCA_CH3	O	PCA capture input / compare output 3
	SPI_CS	O	SPI CS
	TIM6_CHB	IO	Timer6 capture input / compare output B
	LPTIM_EXT	I	LPTimer external clock input
	RTC_1HZ	O	RTC 1Hz output
	PCA_ECI	I	PCA external clock input
	VC0_OUT	O	VC0 output
	LVDIN1	I	Analog input

Pin #	Pin Name	I/O	Description
24	P15	IO	General purpose digital input/output pin
	I2C_SDA	IO	I2C data
	TIM2_TOG	O	Timer2 reverse output
	TIM4_CHB	IO	Timer4 capture input / compare output B
	LPTIM_GATE	I	LPTimer gating
	SPI_SCK	O	SPI clock
	UART0_RXD	I	UART0 RXD
	LVD_OUT	O	LVD output
	XTLO	O	External XTL crystal oscillator clock input
25	P14	IO	General purpose digital input/output pin
	I2C_SCL	O	I2C clock
	TIM2_TOGN	O	Timer2 opposite reverse output
	ECI	I	PCA external clock input
	ADC_RDY	O	ADC ready
	SPI_CS	O	SPI CS
	UART0_TXD	O	UART0 TXD
	XTLI	I	External XTL crystal oscillator clock input
26	P23	IO	General purpose digital input/output pin
	TIM6_CHA	IO	Timer6 capture input / compare output A
	TIM4_CHB	IO	Timer4 capture input / compare output B
	TIM4_CHA	IO	Timer4 capture input / compare output A
	PCA_CH0	IO	PCA capture input / compare output0
	SPI_MISO	IO	SPI Module master input slave output data signal
	UART1_TXD	O	UART1 TXD
	IR_OUT	O	38k carrier output
	LVDIN2/VC0	I	Analog input
27	P24	IO	General purpose digital input/output pin
	TIM4_CHB	IO	Timer4 capture input / compare output B
	TIM5_CHB	IO	Timer5 capture input / compare output B
	HCLK_OUT	O	HCLK output
	PCA_CH1	IO	PCA capture input / compare output 1
	SPI_MOSI	O	SPI module master output slave input data signal
	UART1_RXD	I	UART1 RXD
	VC1_OUT	O	VC1 output
	AIN0	I	Analog input
28	P25	IO	General purpose digital input/output pin
	SPI_SCK	O	SPI Clock
	PCA_CH0	IO	PCA capture input / compare output 0
	TIM5_CHA	IO	Timer5 capture input / compare output A
	LVD_OUT	O	LVD output

Pin #	Pin Name	I/O	Description
29	UART2_RXD	I	UART2 RXD
	I2C_SDA	IO	I2C data
	TIM1_GATE	I	Timer1 gating
	LVDIN3/VC1	I	Analog input
30	P26	IO	P26 general purpose digital input/output pin
	SPI_MOSI	O	SPI Module master output slave input data signal
	TIM4_CHA	IO	Timer4 capture input / compare output A
	TIM5_CHB	IO	Timer5 capture input / compare output B
	PCA_CH2	IO	PCA capture input / compare output 2
	UART2_TXD	O	UART2 TXD
	I2C_SCL	O	I2C Clock
	TIM1_EXT	I	Timer1 external clock input
	AIN1	I	Analog input
31	P27	IO	General purpose digital input/output pin
	SPI_MISO	IO	SPI Module master input slave output data signal
	TIM5_CHA	IO	Timer5 capture input / compare output A
	TIM6_CHA	IO	Timer6 capture input / compare output A
	PCA_CH3	IO	PCA capture input / compare output 3
	UART0_RXD	I	UART0 RXD
	RCH_OUT	O	24 M oscillating output
	XTH_OUT	O	32 M oscillating output
	SWDIO	IO	Debug interface, SWDIO
32	P31	IO	General purpose digital input/output pin
	TIM3_TOG	O	Timer 3 reverse output
	PCA_ECI	I	PCA external clock
	PCLK_OUT	O	PCLK output
	VC0OUT	O	VC0 output
	UART0_TXD	O	UART0 TXD
	RCL_OUT	O	RCL oscillating output
	HCLK_OUT	O	HCLK output
33	SWCLK	I	Debug interface, SWCLK
	SCLK	I	RF SPI clock
	SDA	IO	RF SPI data input/output, connect to 10 kΩ pull-up resistor externally
	CSB	I	RF SPI chip selection for register access
	FCSB	I	RF SPI chip selection for FIFO access
	NC	-	Not connect
	XI	I	26 MHz crystal circuit input
	XO	O	26 MHz crystal circuit output
	GPIO2	IO	Can configure as: INT1,INT2,DOUT/DIN,DCLK (TX/RX),RF_SWT
	GPIO1	IO	Can configure as: DOUT/DIN,INT1,INT2,DCLK (TX/RX),RF_SWT

Pin #	Pin Name	I/O	Description
Notes:			
[1]. INT1 and INT2 refer to RF interrupts. DOUT refers to the demodulated data output. DIN refers to the modulation data input. DCLK refers to the modulation or demodulation data rate synchronous clock, which is switched automatically according to operation mode switch between TX and RX.			

CMOSTEK Confidential

1.26 Module Signal Description

Table 7. Module Signal Description

Module	PIN Name	Description
ADC	AIN0	ADC input channel 0
	AIN1	ADC input channel 1
	AIN2	ADC input channel 2
	AIN3	ADC input channel 3
	AIN4	ADC input channel 4
	AIN5	ADC input channel 5
	AIN6	ADC input channel 6
	AIN7	ADC input channel 7
	AIN8	ADC input channel 8
	ADC_VREF	ADC external reference voltage
	ADC_RDY	ADC Ready output signal
Voltage Compare VC	VCIN0	VC input 0
	VCIN1	VC input 1
	VCIN2	VC input 2
	VCIN3	VC input 3
	VCIN4	VC input 4
	VCIN5	VC input 5
	VCIN6	VC input 6
	VCIN7	VC input 7
	VC0_OUT	VC0 compare output
	VC1_OUT	VC1 compare output
LVD	LVDIN1	voltage detection input 1
	LVDIN2	voltage detection input 2
	LVDIN3	voltage detection input 3
	LVD_OUT	voltage detection input
UART	UART0_TXD	UART0 data transmit end
	UART0_RXD	UART0 data receive end
	UART1_TXD	UART1 data transmit end
	UART1_RXD	UART1 data receive end

Module	PIN Name	Description
LPUART	UART2_TXD	LPUART data transmit end
	UART2_RXD	LPUART data receive end
SPI	SPI_MISO	SPI module master input slave output data signal
	SPI_MOSI	SPI module master output slave input data signal
	SPI_SCK	SPI module clock Signal
	SPI_CS	SPI CS
I2C	I2C_SDA	I2C module data signal
	I2C_SCL	I2C module clock signal
Base Timer	TIM0_TOG	Timer 0 reverse output signal
	TIM0_TOGN	Timer 0 opposite reverse output signal
	TIM0_EXT	Timer 0 external count input signal
	TIM0_GATE	Timer 0 gate control signal
	TIM1_TOG	Timer 1 reverse output signal
	TIM1_TOGN	Timer 1 opposite reverse output signal
	TIM1_EXT	Timer 1 external count input signal
	TIM1_GATE	Timer1 gate control signal
	TIM2_TOG	Timer 2 reverse output signal
	TIM2_TOGN	Timer 2 opposite reverse output signal
	TIM2_EXT	Timer 2 external count input signal
	TIM2_GATE	Timer 2 gating signal
LPTimer	LPTIM_TOG	LPTimer reverse output signal
	LPTIM_TOGN	LPTimer opposite reverse output signal
	LPTIM_EXT	LPTimer external count input signal
	LPTIM_GATE	LPTimer gate control signal
PCA	PCA_ECI	external clock input signal
	PCA_CH0	capture input / compare output /PWM output 0
	PCA_CH1	capture input / compare output /PWM output 1
	PCA_CH2	capture input / compare output /PWM output 2
	PCA_CH3	capture input / compare output /PWM output 3
	PCA_CH4	capture input / compare output /PWM output 4
Advanced Timer	TIM4_CHA	Advanced Timer 4 compare output / capture input end A
	TIM4_CHB	Advanced Timer 4 compare output / capture input end B

Module	PIN Name	Description
	TIM5_CHA	Advanced Timer 5 compare output / capture input end A
	TIM5_CHB	Advanced Timer 5 compare output / capture input end B
	TIM6_CHA	Advanced Timer 6 compare output / capture input end A
	TIM6_CHB	Advanced Timer 6 compare output / capture input end B
	P00-P36	Advanced Timer external trigger input and internal register configuration

Table 8. Analog and Digital Pins

Analog		Digital							
PxADS = 1		PxADS = 0							
		Px_sel = 0	Px_sel = 1	Px_sel = 2	Px_sel = 3	Px_sel = 4	Px_sel = 5	Px_sel = 6	Px_sel = 7
AIN4	VCIN4	P34	PCA_CH0	UART2_TXD	TIM5_CHA	TIM0_EXT	TIM4_CHA	RTC_1Hz	TIM1_TOG
AIN5	VCIN5	P35	UART1_TXD	TIM6_CHB	UART0_RXD	TIM0_GATE	TIM4_CHB	SPI_MISO	I2C_SDA
AIN6 ADC_VREF	VCIN6	P36	UART1_RXD	TIM6_CHA	UART0_RXD	PCA_CH4	TIM5_CHA	SPI_MOSI	I2C_SCL
		P00 Reset							
AIN7 XTHI	VCIN7	P01	UART0_RXD	I2C_SDA	UART1_RXD	TIM0_TOG	TIM5_CHB	SPI_SCK	TIM2_EXT
AIN8 XTHO		P02	UART0_TXD	I2C_SCL	UART1_RXD	TIM0_TOGN	TIM6_CHA	SPI_CS	TIM2_GATE
		VSS							
		VCAP							
		VDD							
LVDIN1		P03	PCA_CH3	SPI_CS	TIM6_CHB	LPTIM_EXT	RTC_1Hz	PCA_ECI	VC0_OUT
XTLO		P15	I2C_SDA	TIM2_TOG	TIM4_CHB	LPTIM_GATE	SPI_SCK	UART0_RXD	LVD_OUT
XTLI		P14	I2C_SCL	TIM2_TOGN	PCA_ECI	ADC_RDY	SPI_CS	UART0_RXD	NC
LVDIN2	VCIN0	P23	TIM6_CHA	TIM4_CHB	TIM4_CHA	PCA_CH0	SPI_MISO	UART1_RXD	IR_OUT
AIN0		P24	TIM4_CHB	TIM5_CHB	HCLK_OUT	PCA_CH1	SPI_MOSI	UART1_RXD	VC1_OUT
LVDIN3	VCIN1	P25	SPI_SCK	PCA_CH0	TIM5_CHA	LVD_OUT	UART2_RXD	I2C_SDA	TIM1_GATE
AIN1		P26	SPI_MOSI	TIM4_CHA	TIM5_CHB	PCA_CH2	UART2_RXD	I2C_SCL	TIM1_EXT
		P27 SWDIO	SPI_MISO	TIM5_CHA	TIM6_CHA	PCA_CH3	UART0_RXD	RCH_OUT	XTH_OUT

		P31 SWCLK	LPTIM_TOG	PCA_ECI	PCLK_OUT	VC0_OUT	UART0_TXD	RCL_OUT	HCLK_OUT
AIN2	VCIN2	P32	LPTIM_TOGN	PCA_CH2	TIM6_CHB	VC1_OUT	UART1_TXD	PCA_CH4	RTC_1Hz
AIN3	VCIN3	P33	UART2_RXD	PCA_CH1	TIM5_CHB	PCA_ECI	UART1_RXD	XTL_OUT	TIM1_TOGN
Base Timer		LPTimer	ADVTimer	PCA	UART	SPI	I2C	RTC	Other

Note: the IO port is in the input high-impedance state after reset. It maintains the previous port state when entering sleep mode and deep sleep mode.

2 System and Memory

2.1 Overview

The system of this product consists of the following components:

- **One AHB bus master**
 - Cortex-M0+
- **4 AHB bus slaves**
 - FLASH memory
 - SRAM memory
 - AHB0 and AHB to APB bridge, including all APB interface peripherals
 - AHB1, including all AHB interface peripherals

Multi-level AHB-lite bus interconnection is adopted for the entire system as shown in the below figure.

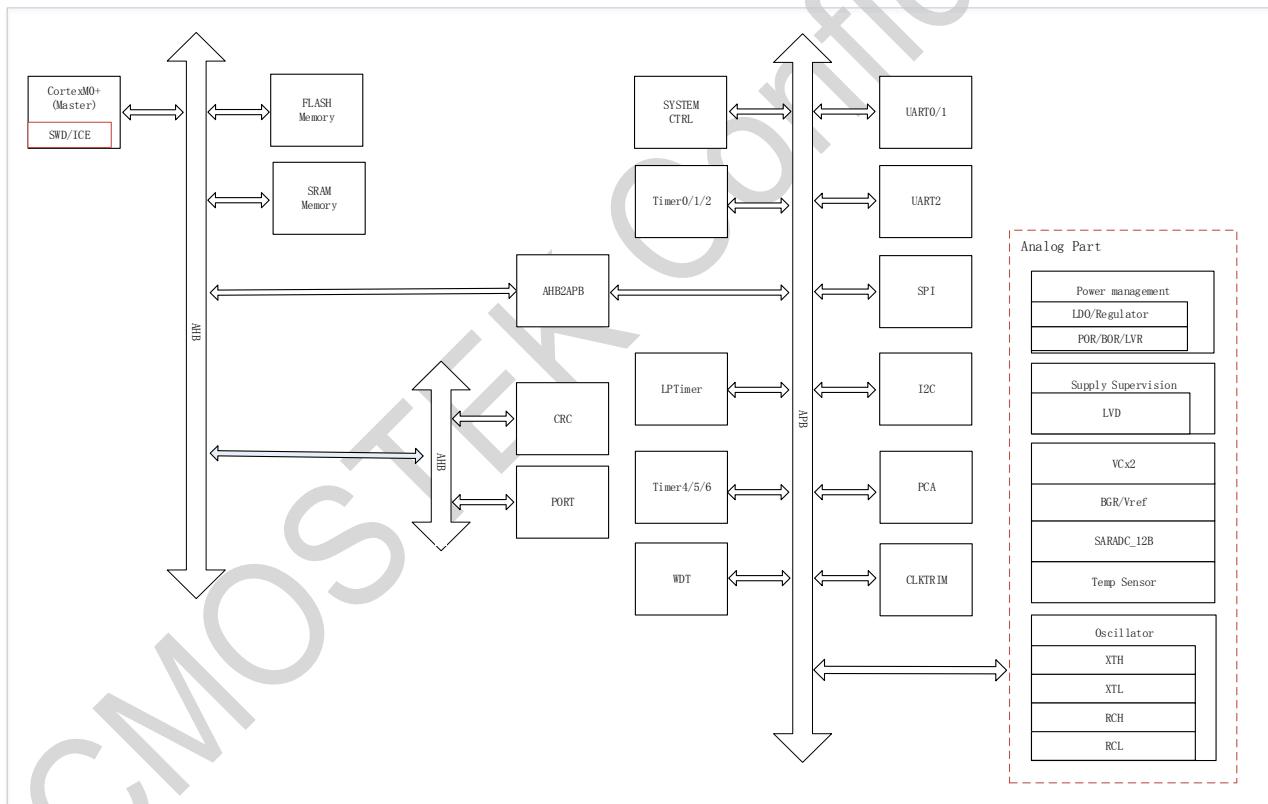


Figure 4. System Architecture Diagram

2.2 System Address Map

The entire system address map is shown in the below figure.

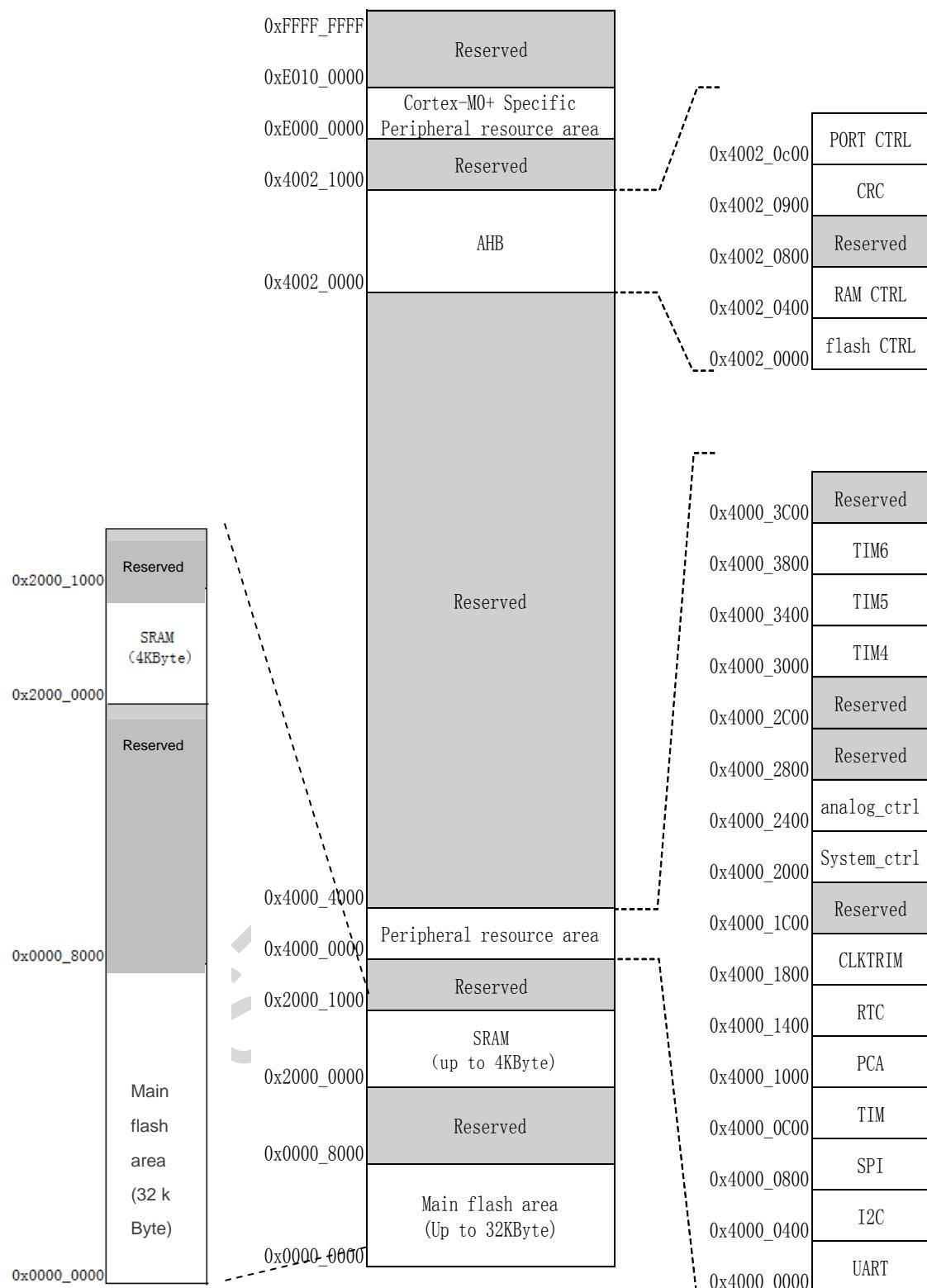


Figure 5. Schematic Diagram of System Address Map

Table 9. Memory and Module Address Map

Boundary Address	Size	Memory Area
0x0000_0000 – 0x0000_7FFF	32 kbyte	FLASH Memory
0x0000_8000 – 0x0010_0BFF	-	Reserved
0x0010_0C00 – 0x0010_0C3B	60 byte	Trim Data
0x0010_0C3C – 0x0010_0E6F	-	Reserved
0x0010_0E70 – 0x0010_0E7F	16 byte	UID
0x0010_0E80 – 0x1FFF_FFFF	-	Reserved
0x2000_0000 – 0x2000_0FFF	4 kbyte	SRAM Memory
0x2000_1000 – 0x3FFF_FFFF	-	Reserved
0x4000_0000 – 0x4000_00FF	256 byte	UART0
0x4000_0100 – 0x4000_01FF	256 byte	UART1
0x4000_0200 – 0x4000_02FF	256 byte	LPUART
0x4000_0300 – 0x4000_03FF	-	Reserved
0x4000_0400 – 0x4000_07FF	1 kbyte	I2C
0x4000_0800 – 0x4000_0BFF	1 kbyte	SPI
0x4000_0C00 – 0x4000_0FFF	1 kbyte	Timer0/1/2/WDT/LPTimer
0x4000_1000 – 0x4000_13FF	1 kbyte	PCA
0x4000_1400 – 0x4000_17FF	1 kbyte	RTC
0x4000_1800 – 0x4000_1BFF	1 kbyte	CLKTRIM
0x4000_1C00 – 0x4000_1FFF	-	Reserved
0x4000_2000 – 0x4000_23FF	1 kbyte	SYSTEMCTRL
0x4000_2400 – 0x4000_27FF	1 kbyte	ANALOGCTRL
0x4000_2800 – 0x4000_2FFF	-	Reserved
0x4000_3000 – 0x4000_33FF	1 kbyte	Timer4
0x4000_3400 – 0x4000_37FF	1 kbyte	Timer5
0x4000_3800 – 0x4000_3BFF	1 kbyte	Timer6
0x4000_3C00 - 0x4001_FFFF	-	Reserved
0x4002_0000 - 0x4002_03FF	1 kbyte	FLASH CTRL
0x4002_0400 - 0x4002_07FF	1 kbyte	RAM CTRL
0x4002_0800 - 0x4002_08FF	256 byte	Reserved
0x4002_0900 - 0x4002_0BFF	768 byte	CRC
0x4002_0C00 - 0x4002_0FFF	1 kbyte	PORT CTRL

3 Operating Mode

The power management module of the chip is responsible for managing the switching between various operating modes, and controlling the operating state of each functional module in each operating mode. The supply voltage (VCC) of this product is from 1.8 to 3.6 V.

The operating modes are as follows:

1. **Active Mode:** CPU runs and peripheral function modules run.
2. **Sleep Mode:** CPU stops running and the peripheral function modules run.
3. **Deep Sleep Mode:** CPU stops running and the high speed clock stops running.

It can enter the low power modes from active mode by software program execution or interrupt triggering.

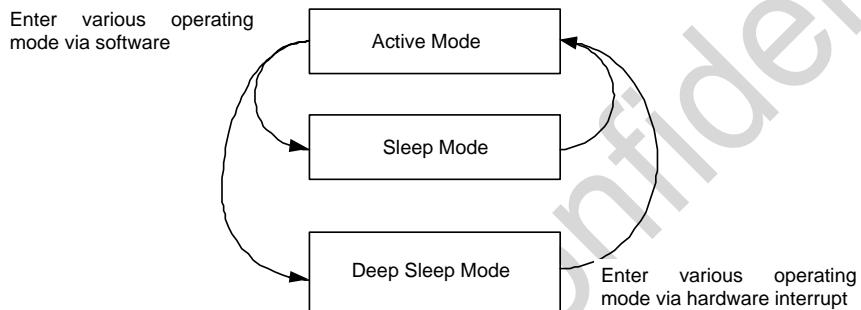


Figure 6. Modes Control Block Diagram

In each mode, the CPU can respond to all types of interrupt.

Table 10. Interrupts Responded in Each Operating Mode

#	Interrupt Source	Operating Mode	Sleep Mode	Deep Sleep Mode
0	GPIO_P0	✓	✓	✓
1	GPIO_P1	✓	✓	✓
2	GPIO_P2	✓	✓	✓
3	GPIO_P3	✓	✓	✓
6	UART0	✓	✓	
7	UART1	✓	✓	
8	LPUART	✓	✓	✓
10	SPI	✓	✓	
12	I2C	✓	✓	
14	Timer0	✓	✓	
15	Timer1	✓	✓	
16	Timer2	✓	✓	
17	LPTimer	✓	✓	✓
18	Timer4	✓	✓	

19	Timer5	✓	✓	
[20]	Timer6	✓	✓	
[21]	PCA	✓	✓	
[22]	WDT	✓	✓	✓
[23]	RTC	✓	✓	✓
[24]	ADC	✓	✓	
[26]	VC0	✓	✓	✓
[27]	VC1	✓	✓	✓
[28]	LVD	✓	✓	✓
[30]	RAM FLASH fault	✓	✓	
[31]	Clock trim	✓	✓	✓

Notes:

1. It responds all types of reset in each operating mode.

Table 11. Resets Responded in Each Operating Mode

	Reset Source	Operating Mode	Sleep Mode	Deep Sleep Mode
[0]	Power-on (POR)	✓	✓	✓
[1]	External reset pin reset	✓	✓	✓
[2]	LVD reset	✓	✓	✓
[3]	WDT reset	✓	✓	✓
[4]	PCA reset	✓	✓	
[5]	Cortex-M0+ LOCKUP hardware reset	✓		
[6]	Cortex-M0+ SYSRESETREQ software reset	✓		

3.1 Active Mode

In active mode, the microcontroller MCU is running after power-on reset or the system wakes up from low power modes. The system stays in low power modes for energy saving in the conditions where it's not necessary for CPU to run, such as waiting for some external event. Users need to select an optimal low power mode based on conditions such as minimum power consumption, shortest startup time, available wakeup sources, and so on.

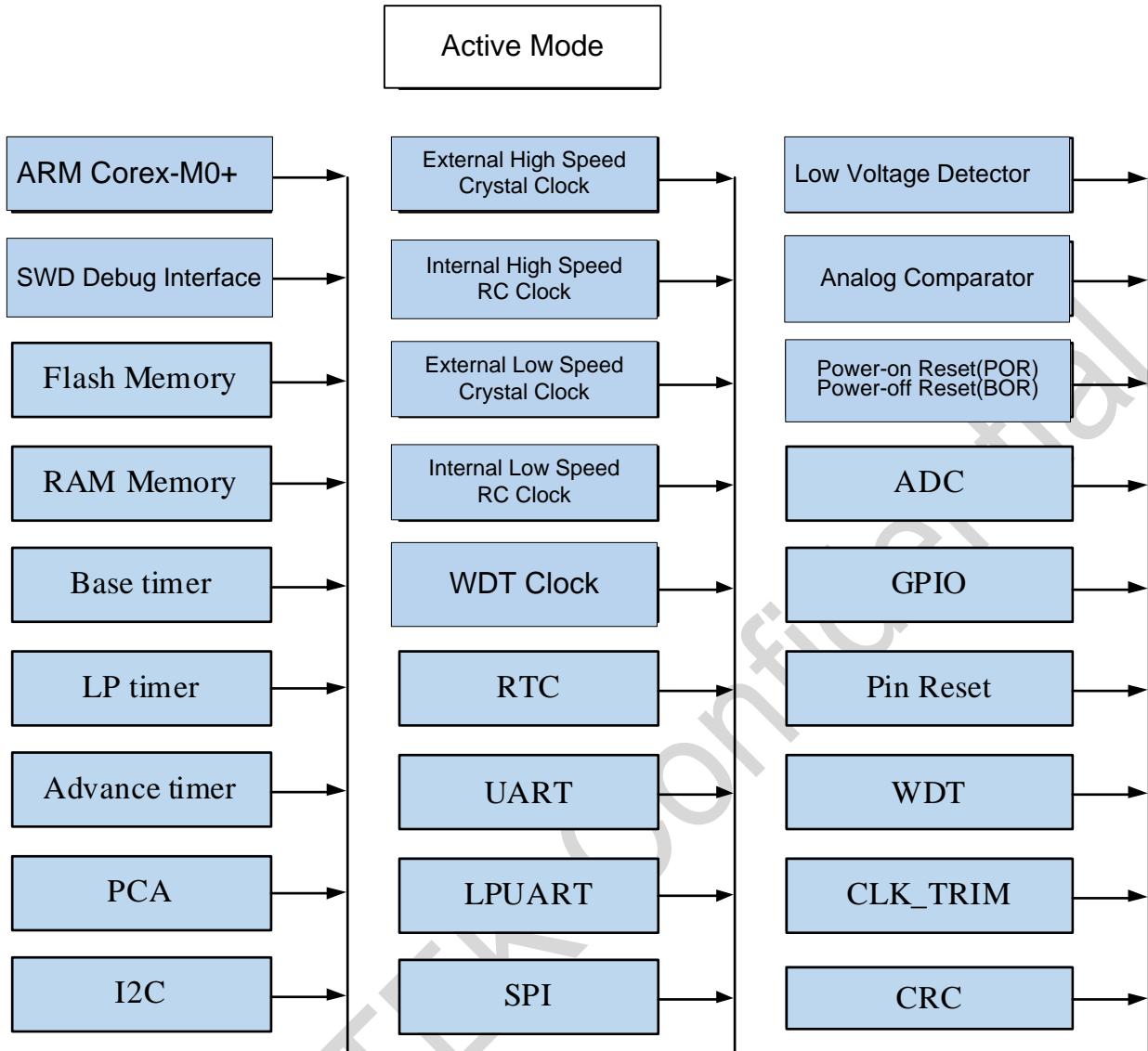


Figure 7. Operational Modules in Active Mode

The ways to reduce the power consumption of the chip in active mode are as follows.

- In active mode, it can reduce the speed of any system clock (HCLK, PCLK) by programming the prescaler registers (SYSCLK0.AHB_CLK_DIV, SYSCLK0.APB_CLK_DIV). It can also use the prescaler to reduce peripherals' clock speeds before the system enters Sleep Mode.
- In active Mode, turn off the clocks of unused peripherals (PERI_CLKx) to reduce power consumption.
- In active mode, turn off the clocks of unused peripherals (PERI_CLKx) to reduce power consumption, which ensures more energy saving when entering Sleep Mode as well. In addition, turn off the clocks of unused peripherals (PERI_CLKx) before executing the WFI instruction.
- Users can choose either low power usage in active mode or sleep mode for energy saving based on actual conditions. As the wakeup time of this product is very short (about 3 us), either way can satisfy real time response requirements.

3.2 Sleep Mode

It can enter sleep mode by executing the WFI instruction. In sleep mode, the CPU stops running and the clock module, system clock, NVIC interrupt processing module and peripheral function modules can still work.

- **How to enter sleep mode**

Execute the WFI instruction to enter sleep mode. According to the value of the SLEEPONEXIT bit in the CortexTM-M0+ system control register, 2 options are available to enter sleep mode.

1. **SLEEP-NOW:** If the SLEEPONEXIT bit is cleared, the microcontroller enters sleep mode immediately when WFI or WFE is executed.
2. **SLEEP-ON-EXIT:** If the SLEEPONEXIT bit is set, the microcontroller enters sleep mode immediately after the system exits from the lowest priority interrupt handler.

- **How to exit Sleep Mode**

Any peripheral interrupt responded by any high-priority nested vectored interrupt controller can wakeup the system from sleep mode if it enters sleep mode via WFI instruction execution.

- **Usage Tips**

1. If the bit SLEEP-ON-EXIT is set to 1, it enters sleep mode automatically after the interrupt is executed with no need to write __wfi() in the program.
2. If the bit SLEEP-ON-EXIT is cleared to 0, main() enters sleep mode after executing __wfi(). Once an interrupt is triggered, it executes the interrupt program and returns main(). Then by executing the WFI instruction, it enters sleep mode waiting for subsequent interrupt triggering.
3. The bit SLEEP-ON-EXIT does not affect __wfi() instruction execution. If SLEEP-ON-EXIT =0, main() enters sleep mode after executing wfi(). It returns to main() after an interrupt triggering and the interrupt program execution, then continues further execution.
4. If it enters sleep mode via an interrupt, only the interrupt with higher priority than that of the current interrupt can wake it up based on the principle that higher priority interrupt is executed first.

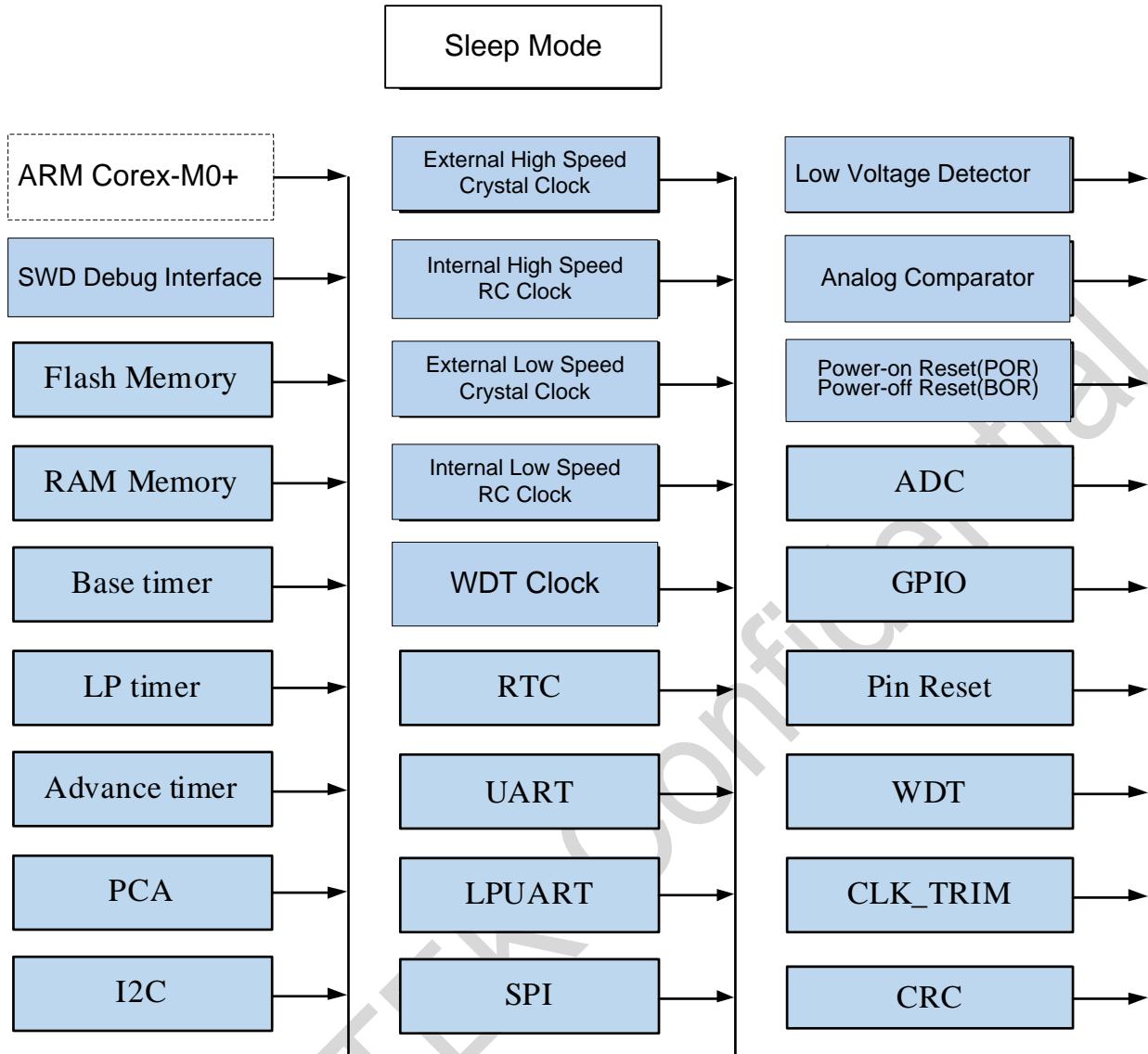


Figure 8. Operational Module Diagram in Sleep Mode

3.3 Deep Sleep Mode

It uses the SLEEPDEEP instruction and the WFI instruction to enter deep sleep mode. In deep sleep mode, CPU stops running, high speed clocks are off, low speed clocks and some low power peripheral modules run optionally per configuration. NVIC interrupt processing can work.

- **Entering deep sleep mode from high speed clocks**

High speed clocks are automatically turned off, and low speed clocks keep the state before entering sleep mode.

- **Entering deep sleep mode from low speed clocks**

Low speed clock keep running, while the rest modules are turned off automatically.

If the system clock is switching when entering deep sleep mode, no clock is turned off automatically. It needs the software to turn-on or turn-off the desired clocks according to the specific power consumption and system requirements.

Entering deep sleep mode does not change the port state, therefore it needs to change the IO state according to IO state

requirements in deep sleep mode.

- **How to enter deep sleep mode**

Set the SLEEPDEEP bit in the Cortex-M0+ system control register first, enter deep sleep mode by executing WFI instruction then. There are two options corresponding to two mechanisms to access deep sleep mode depending on the value of SLEEPONEXIT bit in the CortexTM-M0+ system control register.

1. **SLEEP-NOW:** If the SLEEPONEXIT bit is cleared, the microcontroller enters deep sleep mode immediately once WFI or WFE is executed.
2. **SLEEP-ON-EXIT:** If the SLEEPONEXIT bit is set and the system exits from the lowest priority interrupt handler, the microcontroller enters deep sleep mode immediately.

- **How to exit deep sleep mode**

If it enters deep sleep mode via WFI instruction execution, any peripheral interrupt (the interrupt from operational peripheral modules in deep sleep mode) responded by the nested vectored interrupt controller can wakeup the system from deep sleep mode.

See section 4.4 *Interrupt Wakeup Control WIC* for more details on wakeup settings.

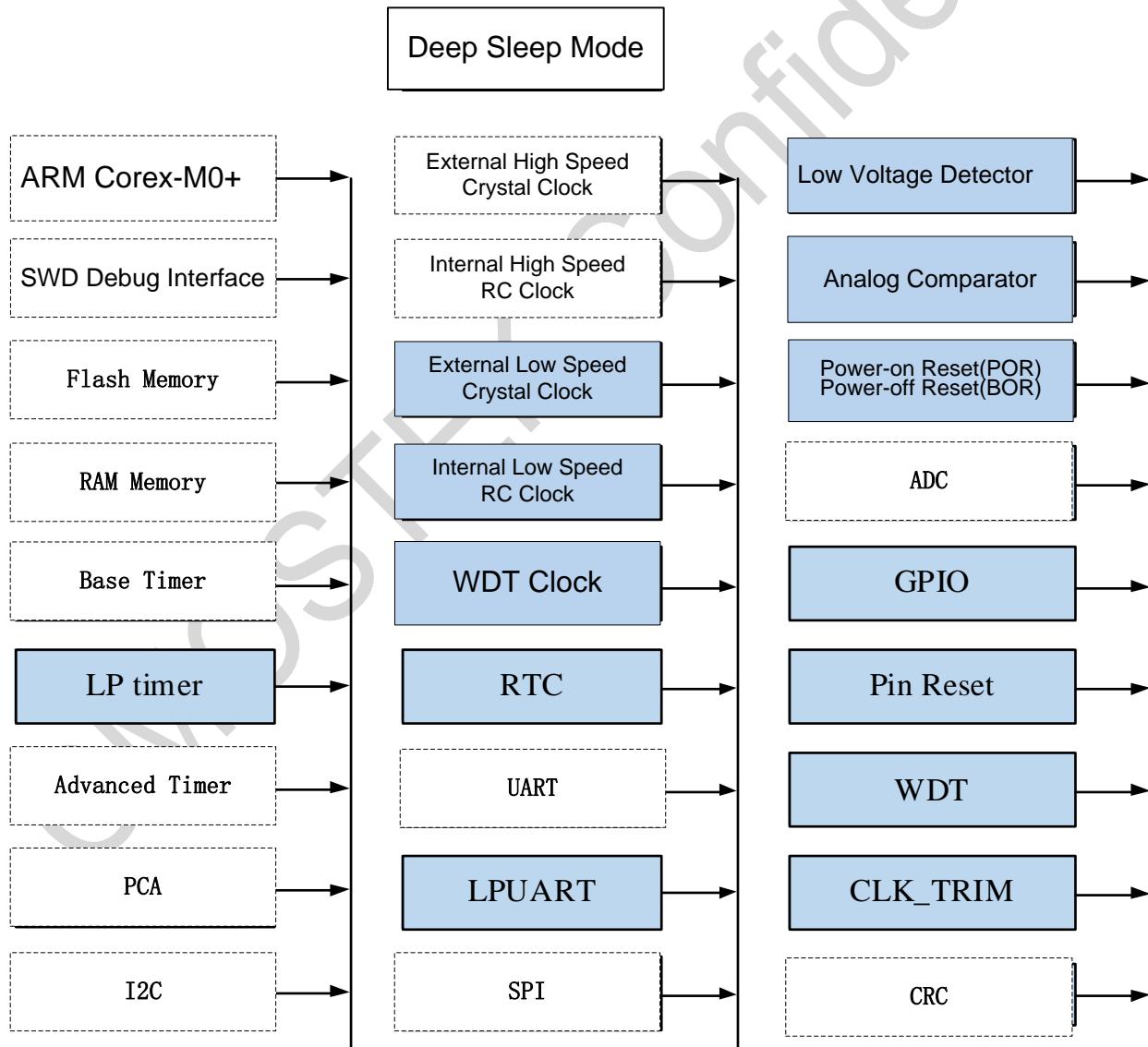


Figure 9. Operational Moudles in Deep Sleep Mode

System Control Register (Cortex-M0+ Core System Control Register)

Address: 0xE000ED10

Reset value: 0x0000 0000

Table 12. System Control Register (Cortex-M0+ Core System Control Register)

Bit	Flag	Function Description	Read/Write
31:5	RESERVED	Reserved	
4	SEVONPEND	When it's set to 1, an event is generated each time a new interrupt is triggered. If it enters sleep mode via WFE, SEVONPEND can be used to wakeup CPU.	RW
3	RESERVED	Reserved	
2	SLEEPDEEP	When it is set to 1, it enters deep sleep mode after WFI execution. When it is set to 0, it enters sleep mode after WFI execution.	RW
1	SLEEPONEXIT	When it is set to 1, the processor automatically enters deep sleep mode (WFI) when exiting exception handling and returning to the program thread. When it is set to 0, this feature is automatically disabled.	RW
0	RESERVED	Reserved	

When it is waken-up from deep sleep mode, there are two system clock choices. By default, it chooses the clock used when entering deep sleep mode. However, if the configuration register YSCTRL0.wakeup_byRCH is 1, it uses the internal high speed clock RCH regardless of the clock used before entering deep sleep mode. Such settings can speed up system wakeup if an external crystal oscillation is used.

4 System Controller

4.1 System Clock Introduction

The clock control module majorly controls system clocks and the peripheral clocks. Different clock sources can be configured as the system clock. It supports different system clock dividing configuration and the peripheral clock enabling/disabling. To ensure high precision, all internal clocks enable calibration function.

The product supports using the following 4 clock sources as the system clock.

- Internal high speed RC clock RCH (4 M as default frequency)
- External low speed crystal clock XTL
- Internal low speed RC clock RCL (38.4 k and 32.768 k, which is configurable)
- External high speed crystal clock XTH

- RCH, XTL, and XTH can also be input from the outside through IO P31, P14, and P01. When using an external oscillator input, the corresponding oscillation needs to be enabled. External oscillator selection control is configured in in SYSCTRL1 register.
- In addition, the product supports the following 2 clocks as well:
- Internal low speed 10 K watchdog clock. Enabling this clock is controlled by watchdog enabling, meaning, to enable this clock, the watchdog must be enabled. meaning .
- Internal 150 K debounce clock

It should be noted it should follow the correct process when switching the system clock source. See chapter *System Clock Switching* for details.

The clock architecture of the product is shown in the below figure.

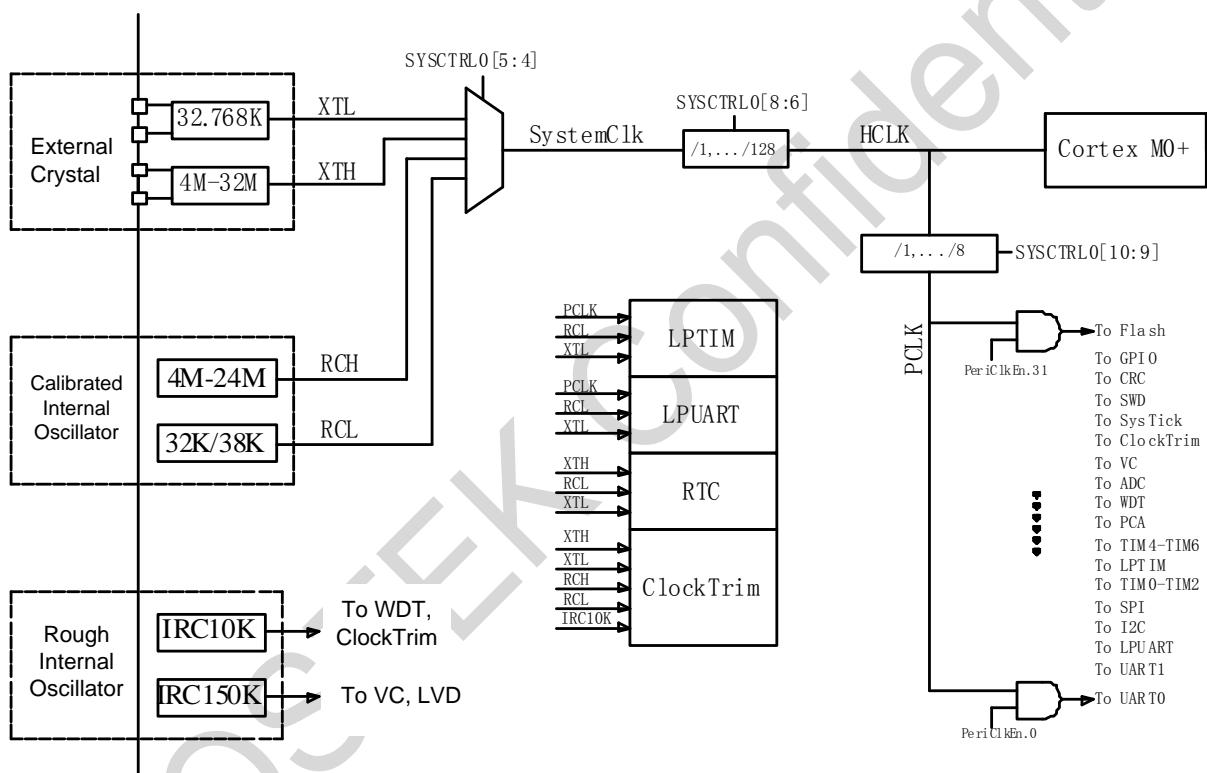


Figure 10. Clock Control Module Block Diagram

4.1.1 Internal High Speed RC Clock (RCH)

The default system clock is the internal high speed clock. It starts working after the chip powers on or resets. The internal high speed clock frequency is configured by the register RCH_CR[10:0], delivering accurate frequencies at 4 M, 8 M, 16 M, 22.12 M or 24 M. As the internal high speed clock startups fast in about 3 us, the system uses this clock to wakeup more promptly from deep sleep mode to a state able to respond external interrupts. To ensure real-time response, it needs to switch the system clock to the internal high speed RC clock before entering deep sleep mode.

4.1.2 Internal Low Speed RC Clock (RCL)

The internal low speed clock frequency can be configured as 38.4 K or 32.768 K. In low speed and low precision applications, this clock source can be selected as the system clock.

4.1.3 External Low Speed Crystal Oscillator Clock XTL

The external low speed crystal oscillator clock requires an external 32.768 K low power crystal oscillator with ultra-high precision and ultra-low power consumption. The operational modules in ultra-low power mode use this clock source as the clock signal.

4.1.4 External High Speed Crystal Clock (XTH)

The external 4~32 M crystal oscillator clock needs to be connected to a 4 ~ 32 M high speed crystal oscillator according to users' system requirements.

4.1.5 Clock Startup Process

All of the above 4 clock sources undergo a stabilization time during startup. It uses an external 32.768 K crystal oscillator as an example in the below figure to illustrate the startup stabilization process of the clock.

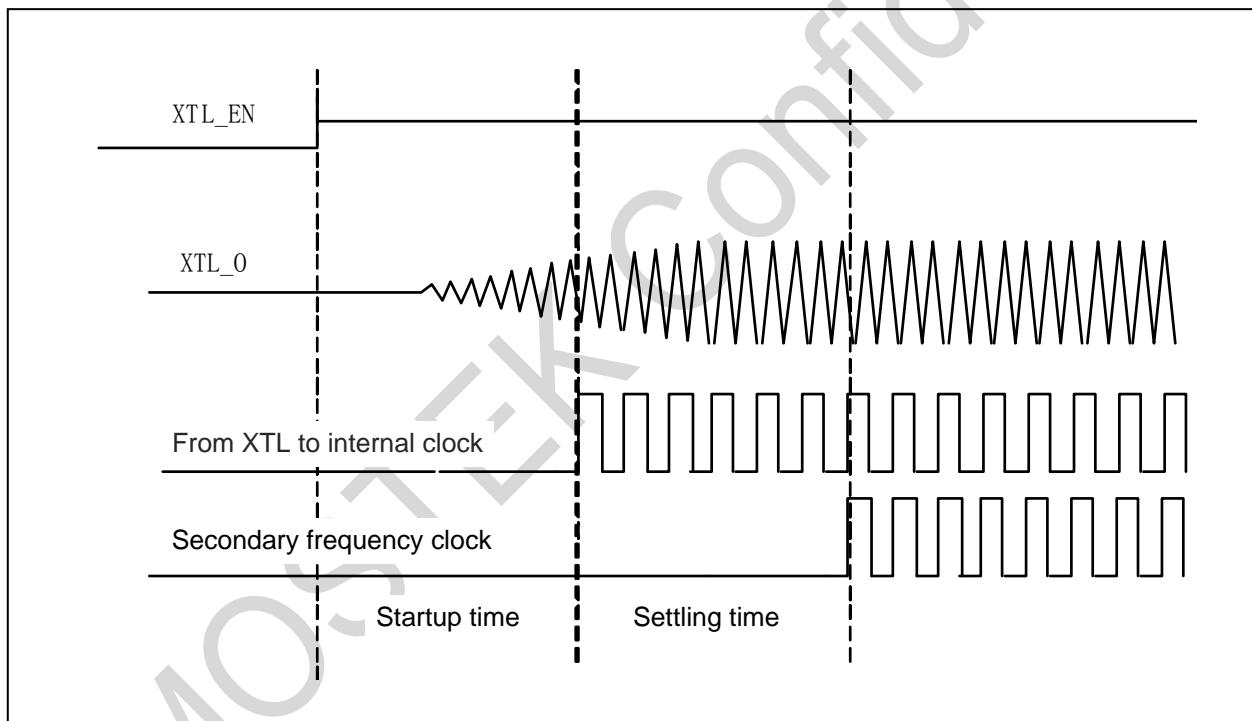


Figure 11. Crystal Clock Startup Schematic

4.2 System Clock Switching

The switching of the clock source is controlled by the register SYSCTRL0[5:0]. In dual clock mode, when the system clock is switched from the current clock to the target one, the implementation must follow a certain process, otherwise an exception will occur.

4.2.1 Switching from Internal High Speed Clock to External Low Speed One

For example, the process of switching from RCH (internal high speed clock) to XTL (external 32.768K crystal oscillator) is as follows.

1. Configure the pin used by the XTL clock to be switched as an analog pin.
2. Make the XTL clock enabled. Write the register SYSCTRL0.XTL_EN and write the protection sequence to the register SYSCTRL2.
3. Wait until the clock XTL is stable.
4. Switch clock, write the register SYSCTRL0.clk_sw4_sel, change the register SYSCTRL0 again, and write protection sequence to register SYSCTRL2 again.
5. Turn off the RCH clock as desired.

4.2.2 Switching from Internal High Speed Clock to External High Speed One

For example, the process of switching from RCH (internal high speed clock) to XTH (external 32M crystal oscillator) is as follows.

1. For the XTH clock to be switched, configure the pin it uses as an analog pin.
2. Make the XTH clock enabled. Write the register SYSCTRL0.XTL_EN and write the protection sequence to the register SYSCTRL2.
3. Wait until the clock XTH is stable.
4. Switch clock, write the register SYSCTRL0.clk_sw4_sel, change the register SYSCTRL0 again, and write protection sequence to register SYSCTRL2 again.
5. Turn off the RCH clock as desired.

Notes:

1. When switching from a low speed clock to a high speed 32 M one, it needs to configure the flash control register and turn on FLASH_CR.wait by setting to 1, otherwise the system may not work properly due to the limitation of flash speed.
2. When using an external high speed 32 M crystal oscillator, set the stabilization time control bit XTH_CR.Startup to 0x3 since the default configuration value of the stabilization time, 0x2, is not enough.

The clock switching timing diagram is shown in the below figure.

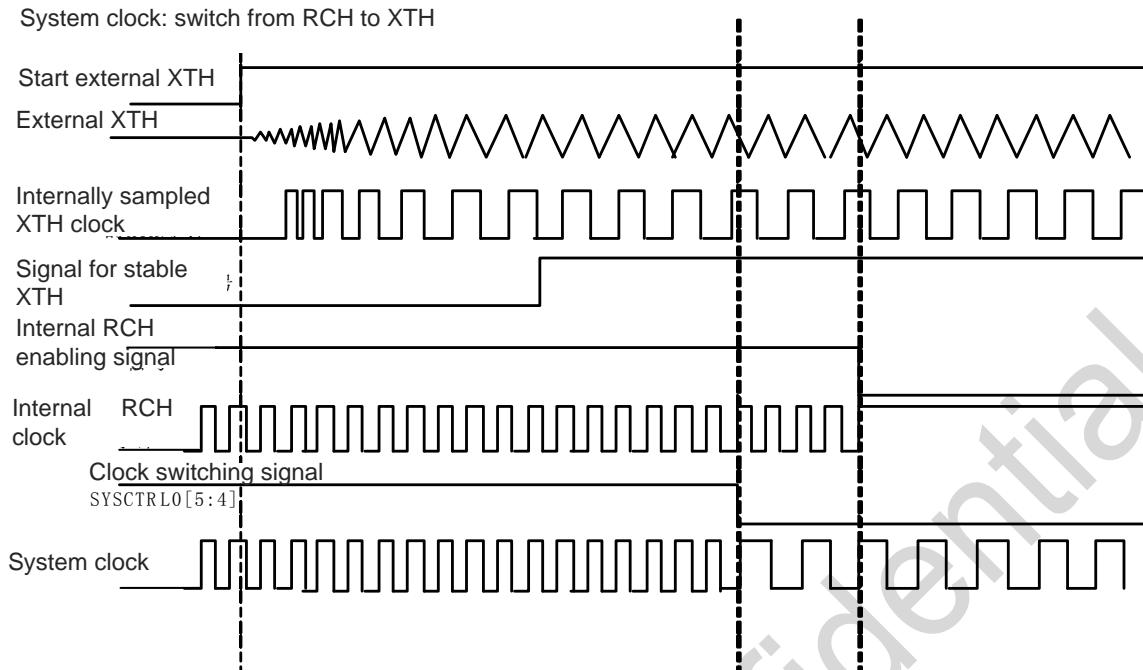


Figure 12. Clock Switching Schematic

4.2.3 Switching from Internal Low Speed Clock to External High Speed One

For example, the process of switching from RCL (internal low speed clock) to XTH (external 32M crystal oscillator) is as follows.

1. Configure the pin used by the clock to be switched as an analog pin.
2. Make the XTH clock enabled. Write the register SYSCTRL0.XTL_EN and write the protection sequence to the register SYSCTRL2.
3. Wait until the clock XTH is stable.
4. Switch clock, write the register SYSCTRL0.clk_sw4_sel, change the register SYSCTRL0 again, and write protection sequence to register SYSCTRL2 again.
5. Turn off the RCL clock as desired.

4.3 Clock Calibration Module

This product is built with clock calibration circuit. As shown in the figure below, the 4 sources of the system clock can calibrate each other. After selecting the reference clock and the calibrated clock, set the register REFCNT and set cali.start to start the clock calibration circuit. At this point, the two 32-bit counters (up and down) work simultaneously. When the down counter is equal to 0, cali.finish is set, indicating that the calibration is over. At this point, the software can read the CALCNT value to figure out the correlation between the reference clock and the clock being calibrated.

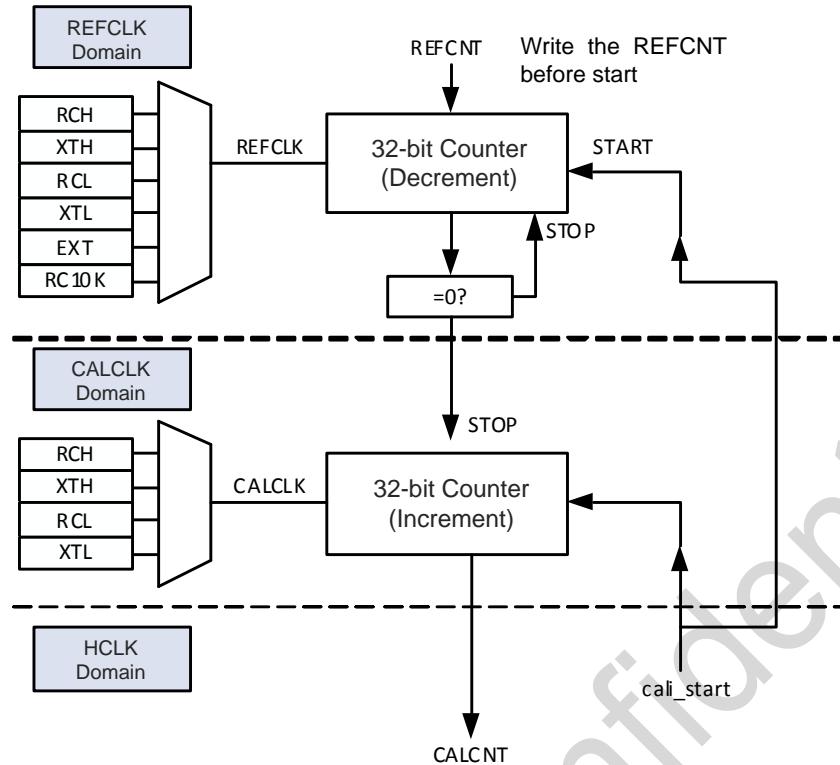


Figure 13. Clock Calibration Schematic

4.4 Interrupt Wakeup Control (WIC)

When the processor enters sleep mode through the sleep-on-exit function or the WFI instruction, it stops instruction execution and the processor is waken-up when interrupt request (higher priority) occurs with processing required.

Table 13. List of Interrupt Wakeup Actions

WFI Action	Wakeup	ISR Execution
PRIMASK Clear	Y	Y
IRQ Priority > Current Priority	N	N
IRQ Priority > Current Priority		
PRIMASK set (interrupt disabled)		
IRQ Priority > Current Priority	Y	N
IRQ Priority \leq Current Priority	N	N

4.4.1 NVIC Settings for Waking-up from Deep Sleep Mode to ISR Execution

1. Enable NVIC to make it operational in deep sleep mode
2. Enable module interrupt
3. Set SCB—>SCR.SLEEPDEEP to 1

4. Enter the deep sleep mode using the WFI instruction
5. The system enters deep sleep mode and waits for an interrupt to wake it up. After wakeup, the next instruction is executed.

An example routine is as follows.

```
SCB_SCR |= 0x00000004u;
    __asm("nop");
    __asm("nop");
    __asm("nop");
    While(1
    {
        __asm("WFI");
    }
```

4.4.2 Settings for NVIC Waking-up from Deep Sleep Mode without ISR Execution

1. Enable NVIC to make it operational in deep sleep mode
2. Mask interrupts using the PRIMASK register
3. Enable module interrupt
4. Set SCB—>SCR.SLEEPDEEP to 1
5. Enter the deep sleep mode by the WFI instruction
6. The system enters deep sleep mode and waits for an interrupt to wake it up. After wakeup, the next instruction is executed.
7. Clear interrupt flag and clear interrupt pending state

An example routine is as follows.

```
__asm("CPSID I"); //disable interrupt enable
SCB_SCR |= 0x00000004u;
    __asm("nop");
    __asm("nop");
    __asm("nop");
while(1
{
    __asm("WFI");
    BTIMERLP_REG->TFCR_f.TFC=0;           //Clear LPTIMER interrupt flag
    NVIC_ClearPendingIRQ(BASE_TIMER3_IRQn); // ClearLPTIMER interrupt pending
}
```

4.4.3 Sleep-on-exit Function Usage

Sleep-on-exit is ideal for interrupt-driven applications. When this feature is enabled, the processor enters sleep mode as soon as the exception handling is completed and returns to thread mode. Using the sleep-on-exit feature, it can make the processor stay in sleep mode as long as possible.

The Cortex-M0 uses the sleep-on-exit feature to enter sleep mode, similar to the effect of performing WFI immediately after an exception handling and exits. However, to avoid pushing the stack when entering the exception next time, the processor will not perform the stacking process.

1. Enable NVIC to make it operational in deep sleep mode
2. Mask interrupts using the PRIMASK register
3. Enable module interrupt
4. Set SCB—>SCR.SLEEPDEEP to 1
5. Enter the Deep Sleep Mode using the WFI instruction
6. The system enters Deep Sleep Mode and waits for an interrupt to wake it up. After waking up, execute the next instruction.
7. Enter sleep mode automatically when exit from interrupt service routine.

An example routine is as follows.

```
SCB_SCR |= 0x00000004u;
SCB_SCR |= 0x00000002u;
    __asm("nop");
    __asm("nop");
    __asm("nop");
while(1)
{ __asm("WFI"); }
```

4.5 Register

Base Address: 0x40002000

Oscillation RCH Control Register

Table 14. System Control Register

Register	Offset Address	Description
SYSCTRL0	0x000	System control register 0
SYSCTRL1	0x004	System control register 1
SYSCTRL2	0x008	System control register 2
RCH_CR	0x00C	Oscillation RCH control register
XTH_CR	0x010	Oscillation XTH control register
RCL_CR	0x014	Oscillation RCL control register

XTL_CR	0x018	Oscillation XTL control register
PERI_CLKEN	0x020	Peripheral module clock control register
SYSTICK_CR	0x034	Systick clock control

4.5.1 System Control Register 0 (SYSCTRL0)

Offset address: 0x000

Reset value: 0x0000 0001

Table 15. System Control Register 0 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 16. System Control Register 0 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wakeup_byRCH	Reserved			PCLK_PRS	HCLK_PRS	clk_sw4_sel	XTL_EN	RCL_EN	XTH_EN	RCH_EN					
R/W	-			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17. System Control Register 0 (3)

Bit	Flag	Description
31:16	Reserved	
15	Wakeup_byRCH	1: when waking-up from deep sleep mode, hardware automatically enables RCH first, and system clock automatically switches to RCH with the original clock keeps on. 0: wake up from deep sleep mode with no change on system clock source.
14:11	Reserved	
10:9	PCLK_PRS	Pclk frequency division selections: 00: hclk 01: hclk/2 10: hclk/4 11: hclk/8
8:6	HCLK_PRS	Hclk frequency division selections(system_clk refers to system clock): 000: system_clk 001: system_clk/2 010: system_clk/4 011: system_clk/8 100: system_clk/16 101: system_clk/32 110: system_clk/64

		111: system_clk/128
5:4	Clk_sw4_sel	<p>System clock selections:</p> <p>00: internal high speed clock RCH</p> <p>01: external 4 ~ 32 M oscillation XTH</p> <p>10: internal low speed clock RCL</p> <p>11: external 32 k oscillation XTL</p>
3	XTL_EN	<p>External 32 k oscillation XTLOSC enabling signal</p> <p>0: disable</p> <p>1: enable</p> <p>Notes: the two external ports connected to the oscillation should be configured as analog ports when it is used.</p>
2	RCL_EN	<p>Internal low speed clock RCL enabling signal</p> <p>0: disable</p> <p>1: enable</p>
1	XTH_EN	<p>External 4 ~ 32 M oscillation XTHOSC enabling signal</p> <p>0: disable</p> <p>1: enable</p> <p>Notes: the 2 external ports connected to the oscillation should be configured as analog ports. This high speed clock is turned off automatically when the system enters deep sleep mode.</p>
0	RCH_EN	<p>Internal high speed clock RCH enabling signal</p> <p>0: disable</p> <p>1: enable</p> <p>Notes: this high speed clock is turned off automatically when the system enters deep sleep mode.</p>

Notes:

- When write protection control register SYSCTRL0 and SYSCTRL1, write 0x5A5A to SYSCTRL2 first, then 0xA5A5 to start write operation. This protection bit resumes to protection status when the write operation completes. Therefore this sequence is used for the protection of the 2 registers.

4.5.2 System Control Register 1 (SYSCTRL1)

Offset address: 0x004

Reset value: 0x00000008

Table 18. System Control Register 1 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 19. System Control Register 1 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			RTC_FREQ_ADJUST			SWD_UIO	RES_UIO	LOCK_EN	RTC_LPW	Clock_ft_en	XTL_awayson	EXTL_EN	EXTH_EN	Res	
			R/W			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Table 20. System Control Register 1 (3)

Bit	Flag	Description
31:12	Reserved	
11:9	RTC_FREQ_ADJUST	Compensation clock frequency selections for high speed clock. 000: 4 M, 001: 6 M, 010: 8 M, 011: 12 M, 100: 16 M, 101: 20 M, 110: 24 M, 111: 32 M.
8	SWD_USE_IO	SWD is used as gpio. 0: SWD port 1: gpio port
7	RESET_USE_IO	Reset is used as gpio. 0: reset port 1: gpio port
6	LOCKUP_EN	Cortex-M0+ lookUp function enabling. 0: disable 1: enable Notes: when the Cortex-M0+ reads wrong instructions, the MCU will reset to ensure system reliability.
5	RTC_LPW	The RTC module has low power control. When it's enabled, the RTC module will enter a low power state with the RTC register unreadable and un-writable. 1: low power enabled 0: low power disabled
4	CLOCK_FAULT_EN	Clock failure detection enabling control. 1: clock failure detection is enabled. When it detects clock failure, the system clock is automatically switched to the RCH. 0: clock failure detection is disabled.
3	XTL_ALWAYS_ON	XTL can be enabled, and can not be disabled 1: SYSCTRL0.XTL_EN, can be enabled and can not be disabled 0: SYSCTRL0.XTL_EN, allow to disable
2	EXTL_EN	External XTL input enabling. 1: enable XTL external P1.4 input 0: disable XTL external P1.4 input, using external oscillation module to output Notes: the enabling bit SYSCTRL0.XTL_EN for low speed oscillation needs to be enabled when using external low speed oscillation.
1	EXTH_EN	External XTH input enabling. 1: enable XTH external P0.1 input

Bit	Flag	Description
		0: desable XTH external P0.1 input, using external oscillation module to output Notes: The enable bit SYSCTRL0.XTL_EN for high speed oscillation needs to be enabled when using external low speed oscillation.
0	Reserved	

4.5.3 System Control Register 2 (SYSCTRL2)

Offset address: 0x008

Reset value: 0x00000000

Table 21. System Control Register 2 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 22. System Control Register 2 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SYSCTRL2															
W															

Table 23. System Control Register 2 (3)

Bit	Flag	Description
31:16	Reserved	
15:0	SYSCTRL2	When write protection control register SYSCTRL0 and SYSCTRL1, write 0x5A5A to SYSCTRL2 first, then 0xA5A5 to start write operation. This protection bit resumes to protection status when the write operation completes. Therefore this sequence is used for the protection of the 2 registers.

4.5.4 Oscillation RCH Control Register (RCH_CR)

Offset address: 0x00C

Reset value: 0x00000126

Table 24. Oscillation RCH Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 25. Oscillation RCH Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Stable	TRIM										
-				R	R/W										

Table 26. Oscillation RCH Control Register (3)

Bit	Flag	Description
31:12	Reserved	
11	Stable	<p>RCH clock stabilization flag.</p> <p>1: RCH is stable and can be used by internal circuits.</p> <p>0: RCH is not stable and cannot be used by internal circuits</p>
10:0	TRIM	<p>The frequency adjustment of the main frequency clock (process variation)</p> <p>Note: the frequency adjustment value is stored in Flash in factory. Users need to write the Flash value to RCH_CR.TRIM to configure the accurate main frequency clock.</p> <p>24 M calibration value address: 0X0010_0C00</p> <p>22.12 M calibration value address: 0X0010_0C02</p> <p>16 M calibration value address: 0X0010_0C04</p> <p>8 M calibration value address: 0X0010_0C06</p> <p>4 M calibration value address: 0X0010_0C08</p>

4.5.5 Oscillation XTH Control Register (XTH_CR)

Offset address: 0x010

Reset value: 0x00000022

Table 27. Oscillation XTH Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 28. Oscillation XTH Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							Stable	Startup	Driver						
-							R	R/W	R/W						

Table 29. Oscillation XTH Control Register (3)

Bit	Flag	Description
31:6	Reserved	
6	stable	<p>External 4 ~ 32 M crystal stabilization flag.</p> <p>1: the external high speed crystal oscillator clock is stable and can be used by internal circuits.</p> <p>0: the external high speed crystal oscillator clock is not stable and cannot be used by internal circuits.</p>

Bit	Flag	Description
5:4	Startup	<p>External 4~32M crystal oscillator stabilization time selections.</p> <p>00: 256 cycles 01: 1024 cycles 10: 4096 cycles 11: 16384 cycles</p> <p>When using a high speed crystal clock, the settling time needs to be set to 11, otherwise the lack of settling time may cause system instability when the system clock is switched or when the high speed crystal clock is used to wakeup from deep sleep mode.</p>
3:0	Driver	<p>External 4 ~ 32 M crystal driving selections.</p> <p>1111: maximum drive (recommended value) 0000: minimum drive</p> <p>For this register, bits 3:2 Xtal_fsel represents external 4 ~ 32 M crystal frequency setting. Set the bits to the corresponding value as below according to the external oscillation frequency.</p> <p>11: 20 ~ 32 M 10: 12 ~ 20 M 01: 6 ~ 12 M 00: 4 ~ 6 M</p> <p>For this register, bitd 1:0 represents external oscillation driving capability selections. Select the corresponding driving capability according to the required driving capacity of the crystal oscillator, capacitive load and parasitic load of the board. The power consumption will raise as the driving ability raises and vice versa.</p> <p>11: the strongest driving ability 10: default drive capability (recommended value) 01: weak drive capability 00: the weakest drive capability</p>

4.5.6 Oscillation RCL Control Register (RCL_CR)

Offset address: 0x014

Reset value: 0x0000033Fh

Table 30. Oscillation RCL Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 31. Oscillation RCL Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			Stable	Startup		TRIM									
-			R	R/W		R/W									

Table 32. Oscillation RCL Control Register (3)

Bit	Flag	Description
31:13	Reserved	
12	Stable	<p>Internal low speed clock stabilization flag.</p> <p>1: the external low speed crystal oscillator clock is stable and can be used by internal circuits.</p> <p>0: the external low speed crystal oscillator clock is not stable and cannot be used by internal circuits.</p>
11:10	Startup	<p>Internal low speed clock stabilization time selections.</p> <p>11: 256 cycles</p> <p>10: 64 cycles</p> <p>01: 16 cycles</p> <p>00: 4 cycles</p>
9:0	TRIM	<p>The frequency adjustment of the internal low speed clock.</p> <p>Note: the frequency variation value is stored in Flash in factory. Users need to write the Flash value to RCH_CR.TRIM to configure the calibrated 38.4K/32.768K internal low speed clock. (process variation)</p> <p>38 k calibration value address: 0X0010_0C20</p> <p>32 k calibration value address: 0X0010_0C22</p>

4.5.7 Oscillation XTL Control Register (XTL_CR)

Offset address: 0x018

Reset value: 0x00000021

Table 33. Oscillation XTL Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 34. Oscillation XTL Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							Stable	Startup	Driver						
-							R	R/W	R/W						

Table 35. Oscillation XTL Control Register (3)

Bit	Flag	Description
31:6	Reserved	
6	Stable	External 32 k crystal stabilization flag. 1: the external 32K oscillator clock is stable and can be used by internal circuits. 0: the external 32K oscillator clock is not stable and cannot be used by internal circuits.
5:4	Startup	External 32 k oscillator stabilization time selections 00: 256 cycles 01: 1024 cycles 10: 4096 cycles 11: 16384 cycles
3:0	Driver	External 32 k oscillator driving capability selections. 1111: Maximum drive 0000: minimum drive 3:2 bits are for Amp_control, representing the selections for the fine adjustment of the oscillation amplitude of the 32KHz oscillator. 11: maximum amplitude 00: minimum amplitude (recommended value) 1:0 bits are for driver, representing the external crystal drive capability selections. Please select the corresponding driving capability according to the required driving capacity of the crystal oscillator, capacitive load and parasitic load of the board. The power consumption will raise as the driving ability raises and vice versa. 11: The strongest driving ability 10: Strong driving ability 01: Default drive capability (recommended value) 00: the weakest drive capability

4.5.8 Peripheral Module Clock Control Register (PERI_CLKEN)

Offset address: 0xC080_0000

Reset value: 0x020

Table 36. Peripheral Module Clock Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Flash	Reserved	GPIO	Res.	CRC	Res.	TICK	Reserved	Reserved	Trim	RTC	Reserved	VC	ADC	-	-
R/W	-	R/W	-	R/W	-	R/W	-	-	R/W	R/W	-	R/W	R/W	R/W	R/W

Table 37. Peripheral Module Clock Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDT	PCA	Reserved	ADVTIM	LPTIM	BASETIM	Res.	SPI	Res.	I2C	Res.	LPUART	UART1	UART0	-	-
R/W	R/W	-	R/W	R/W	R/W	-	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W

Table 38. Peripheral Module Clock Control Register (3)

Bit	Flag	Description
31	Flash	The flash controller module clock is enabled. If it's disabled, the flash configuration register is non-writable and the program in Flash can still run. 1:enable, 0:disable.
30:29	Res.	Reserved bit
28	GPIO	The GPIO module clock enabling. 1:enable, 0:disable.
27	Res.	Reserved bit
26	CRC	CRC module clock enabling. 1:enable, 0:disable.
25	Res.	Reserved bit
24	TICK	SYSTICK control register clock enabling. 1:enable, 0:disable.
23:22	Res.	Reserved bit
21	Trim	clk_trim module clock enabling. 1: enable; 0: disable
20	RTC	RTC module clock enabling. 1:enable, 0:disable.
19:18	Res.	Reserved bit
17	VC	VC and LVD module clock enabling. 1:enable, 0:disable.
16	ADC	ADC module clock enabling. 1:enable, 0:disable.

Bit	Flag	Description
15	WDT	WDT module clock enabling. 1: enable; 0: disable
14	PCA	PCA module clock enabling. 1:enable, 0: disable.
13:11	Res.	Reserved bit
10	ADVTIM	Timer 456 module clock enabling. 1:enable, 0: disable.
9	LPTIM	LPTimer module clock enabling. 1:enable, 0: disable.
8	BASETIM	Timer 0/1/2 module clock enabling. 1:enable, 0: disable.
7	Res.	Reserved bit
6	SPI	SPI module clock enabling. 1:enable, 0: disable.
5	Res.	Reserved bit
4	I2C	I2C module clock enabling. 1:enable, 0: disable.
3	Res.	Reserved bit
2	LPUART	LPUART module clock enabling. 1: enable; 0: disable
1	UART1	UART1 module clock enabling. 1:enable, 0: disable.
0	UART0	UART0 module clock enabling. 1:enable, 0: disable.

4.5.9 Systick Clock Control (SYSTICK_CR)

Offset address: 0x0100_0147

Reset value: 0x034

Table 39. Systick Clock Control (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
Reserved				CLK_SEL	NOREF	SKEW	STCALIB[23:16]									
-	R/W	R/W	R/W	R/W												

Table 40. Systick Clock Control (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STCALIB[15:0]															
R/W															

Bit	Flag	Description
31 - 28	Reserved	
27 - 26	CLK_SEL	<p>External reference clock selections for Systick.</p> <p>00: external 32 k oscillator</p> <p>01: internal 38.4 k low speed clock</p> <p>10: SCLK/8, which divides the system clock (instead of HCLK clock) by 8. Changing the division ratio HCLK does not change this external reference clock.</p> <p>11: external 4 ~ 32 M clock</p>
25	NOREF	<p>Whether systick timer uses an external reference clock.</p> <p>1: use the core clock</p> <p>0: use external reference clock</p> <p>Notes:</p> <p>The core clock is used if either one of the two registers, meaning system register SysTick->CTRL.CLKSOURCE (0XE000E010) and this register, is set to 1.</p> <p>When using the reference clock as the systick clock, the reference clock frequency is not allowed to be higher than that of the system clock HCLK.</p>
24	SKEW	<p>Whether the 10 ms STCALIB value is accurate.</p> <p>1: inaccurate</p> <p>0: accurate</p>
23:0	STCALIB	Systick 10 ms calibration value, which is a 10 ms calibration value using an external 32 k oscillator reference clock.

5 Reset Controller

5.1 Reset Controller Introduction

This product has 7 reset signal sources. Each reset signal allows the CPU to re-run with most registers being reset and the program counter - PC being reset and pointing to 00000000.

- power-on, power-off reset (POR) for the digital part
- External Reset PAD, the reset signal is low level
- WDT reset
- PCA reset
- LVD low voltage reset
- Cortex-M0+ SYSRESETREQ software reset
- Cortex-M0+ LOCKUP hardware reset

Each reset source is represented by a dedicated reset flag bit, which is set to 1 by hardware and cleared by software. All reset flag bits can be cleared by the POR of the digital part except POR reset flag bit.

The various reset sources are shown in the below figure.

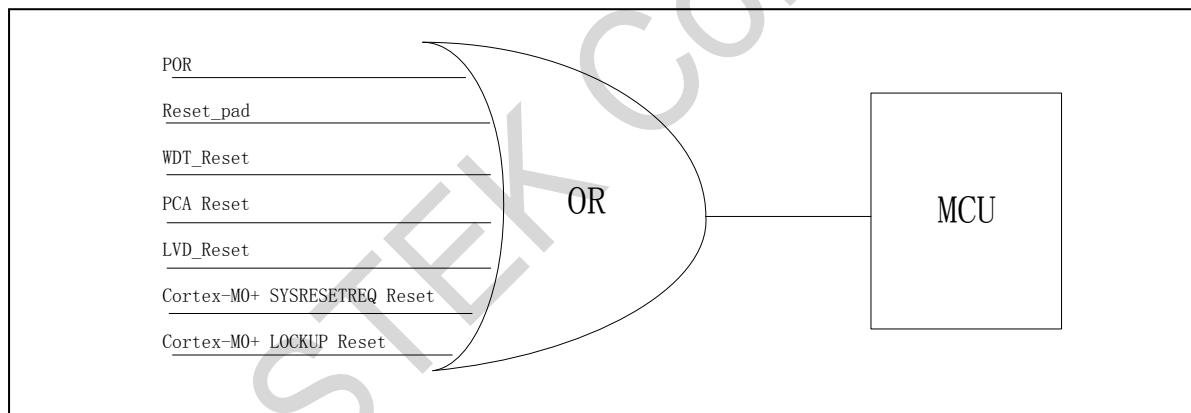


Figure 14. Reset Source Schematic Diagram

5.1.1 Power-on and Power-off Reset (POR)

When the chip is powered-on and powered-off, if the power supply is below a threshold voltage (1.6 V), a POR signal is generated internally. Once the power supply is above the threshold voltage, the POR signal is released. The POR signal resets the SFR and the control signals of the chip. This product has two domains, the VCC domain and the Vcore domain, so there are two PORs, the POR of the VCC and the POR of the Vcore.

5.1.2 External Reset Pin Reset

Pulling the external reset pin to ground will generate a system reset. This reset pin has a pull-up resistor inside to make it high in normal case. In addition, a glitch filter circuit is integrated internally, and the system will automatically filter the glitch signal less

than 20 us (typ.). Therefore, when using the reset pin to generate the reset signal, users need to generate a low level for more than 20 us to ensure the chip can receive the reset signal.

5.1.3 WDT Reset

See the chapter *WDT* for details.

5.1.4 PCA Reset

See the chapter *PCA* for details.

5.1.5 LVD Low Voltage Reset

See the chapter *LVD* for details.

5.1.6 Cortex-M0+ SYSRESETREQ Reset

It is the software reset of the Cortex-M0+.

5.1.7 Cortex-M0+ LOCKUP Reset

When the Cortex-M0+ encounters a critical exception, it stops its PC pointer at the current address and locks itself, then reset the entire CORE area after a delay of several clock cycles.

5.2 Register

5.2.1 Reset Flag Register (Reset_flag)

Offset address: 00000000_00000000_00000000_xxxxxx11b

Reset value: 0x4000201C

Table 41. Reset Flag Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 42. Reset Flag Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								RSTB	Sysreq	Lockup	PCA	WDT	LVD	Por15	Por5v
-								R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 43. Reset Flag Register (3)

Bit	Flag	Description
31:8	Reserved	
7	RSTB	RESETB port reset flag. It is initialized and cleared by the software. The status is uncertain during power-on. 1: port reset occurs 0: no port reset occurs

Bit	Flag	Description
6	Sysreq	Cortex-M0+ CPU software reset flag. It is initialized and cleared by the software. The status is uncertain during power-up. 1: Cortex-M0+ CPU software reset occurs 0: no Cortex-M0+ CPU software reset occurs
5	Lockup	Cortex-M0+ CPU lockup reset flag. It is initialized and cleared by the software. The status is uncertain during power-up. 1: Cortex-M0+ CPU lockup reset occurs 0: no Cortex-M0+ CPU lockup reset occurs
4	PCA	PCA reset flag. It is initialized and cleared by the software. 1: PCA reset occurs 0: no PCA reset occurs
3	WDT	WDT reset flag. It is initialized and cleared by the software. 1: WDT reset occurs 0: no WDT reset occurs
2	LVD	LVD reset flag. It is initialized and cleared by the software. 1: LVD reset occurs 0: no LVD reset occurs
1	POR15V	Vcore area reset flag 1: Vcore area reset occurs 0: no Vcore area reset occurs
0	POR5V	VCC supply area reset flag 1: VCC supply area reset occurs 0: no VCC supply area reset occurs

5.2.2 Peripheral Module Reset Control Register (PREI_RESET)

Offset address: 0xD7B3C757

Reset value: 0x40002028

Table 44. Peripheral Module Reset Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved	GPIO	Res.	CRC	Res.	TICK	Reserved	TRIM	RTC	Reserved	VC	ADC				
-	R/W	-	R/W	-	R/W	-	R/W	R/W	-	R/W	R/W	-	R/W	R/W	R/W

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PCA	Reserved	ADVTIM	LPTIM	BASETIM	Res.	SPI	Res.	I2C	Res.	LPUART	UART1	UART0		
-	R/W	-	R/W	R/W	R/W	-	R/W	-	R/W	-	R/W	R/W	R/W	R/W	R/W

Table 45. Peripheral Module Reset Control Register (2)

Bit	Flag	Description
31:29	Res.	Reserved bit
28	GPIO	GPIO module reset enabling. 1: operate normally, 0: module is in reset status.
27	Res.	Reserved bit
26	CRC	CRC module reset enabling. 1: operate normally, 0: module is in reset status.
25	Res.	Reserved bit
24	TICK	SYSTICK module reset enabling. 1: operate normally, 0: module is in reset status.
23:22	Res.	Reserved bit
21	TRIM	clk_trim module reset enabling. 1: operate normally, 0: module is in reset status.
20	RTC	RTC module reset enabling 1: operate normally, 0: module is in reset status.
19-18	Res.	Reserved bit
17	VC	VC, LVD module reset enabling 1: operate normally, 0: module is in reset status.
16	ADC	ADC module reset enabling. 1: operate normally, 0: module is in reset status.
15	Res.	Reserved bit
14	PCA	PCA module reset enabling. 1: operate normally, 0: module is in reset status.
13:11	Res.	Reserved bit
10	ADVTIM	Timer456 module reset enabling. 1: operate normally, 0: module is in reset status.
9	LPTIM	LPTimer module reset enabling 1: operate normally, 0: module is in reset status.
8	BASETIM	Timer 0/1/2 module reset enabling. 1: operate normally, 0: module is in reset status.
7	Res.	Reserved bit
6	SPI	SPI module reset enabling. 1: operate normally, 0: module is in reset status.
5	Res.	Reserved bit
4	I2C	I2C module reset enabling. 1: operate normally, 0: module is in reset status.
3	Res.	Reserved bit
2	LPUART	LPUART module reset enabling. 1: operate normally, 0: module is in reset status.
1	UART1	UART1 module reset enabling.

		1: operate normally, 0: module is in reset status.
0	UART0	UART0 module reset enabling. 1: operate normally, 0: module is in reset status.

6 Interrupt Controller

6.1 Overview

The Cortex-M0+ processor has a built-in nested vectored interrupt controller (NVIC) that supports up to 32 interrupt request (IRQ) inputs and 1 non-maskable interrupt (NMI) input (not used in this product). In addition, the processor supports multiple internal exceptions.

Each exception source has a separate exception number with each exception type corresponding to a priority. Some exceptions have fixed priorities and some are programmable. The details are as follows.

Table 46. Cortex-M0+ Processor Exceptions

#	Exception No.	Priority	Description
1	Reset	-3 (the highest)	Reset
2	NMI	-2	Non-maskable interrupt (not used in this system)
3	Hardware Error	-1	Error handling exception
4-10	Reserved	NA	...
11	SVC	Programmable	Call the management program via the SVC instruction
12-13	Reserved	NA	...
14	PendSV	Programmable	Pending system service request
15	SysTick	Programmable	SysTickTimer
16	Interrupt #0	Programmable	External interrupt #0
17	Interrupt #1	Programmable	External interrupt #1
...
47	Interrupt #31	Programmable	External interrupt #31

This chapter details the 32 external interrupt requests (interrupt #0 to interrupt #31) of the processor. Please refer to other related documents for details of internal processor exceptions. This chapter discusses the interrupt handling mechanism of NVIC in the processor core only. The interrupt generation mechanism of the peripheral module itself is not covered.

6.2 Interrupt Priority

Each external interrupt corresponds to a 2-bit priority register. The highest two bits of the interrupt priority register (b-bit) are used. Therefore, the available priorities are 0x00 (the highest), 0x40, 0x80, and 0xc0 (the lowest).

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Used	Not Used, read as 0						

Figure 6-1. Priority Register with Only the Higher 2 Bits Used

Preemption occurs if the new interrupt has higher priority than that of the current executed one. The system will turn to execute the new interrupt and halt the current one, called nesting. After the new interrupt is executed, the previous interrupt processing continues and it returns to the program thread when the present one completes.

If the currently processed interrupt has the same or higher priority, the new interrupt will wait and enter pending state. The pending interrupt waits until the current interrupt level changes to a lower level. For example, after the currently processed interrupt completes, the current priority is reduced to a level which is lower than that of the pending interrupt.

If 2 interrupts occur at the same time with the same priority, the interrupt with the smaller interrupt number will be executed first. For example, if Interrupt #0 and Interrupt #1 are enabled and have the same priority, Interrupt #0 will be executed first when they are triggered simultaneously.

6.3 Interrupt Vector Table

When the Cortex-M0+ processor is to process an interrupt service request, it needs to determine the starting address of the exception handling first, which is stored at the vector table as shown in the below figure. The vector table is stored at the beginning of the memory space and it contains the exception (interrupt) vectors of all the exceptions (interrupts) available in the system, as well as the initial value of the main stack pointer (MSP).

Register Address	Exception #
0x0000004C	19
0x00000048	18
0x00000044	17
0x00000040	16
0x0000003C	15
0x00000038	14
0x00000034	13
0x00000030	12
0x0000002C	11
0x00000028	10
0x00000024	9
0x00000020	8
0x0000001C	7
0x00000018	6
0x00000014	5
0x00000010	4
0x0000000C	3
0x00000008	2
0x00000004	1
0x00000000	0

Figure 15. Interrupt Vector Table

The interrupt vectors are stored in order of interrupt number. As each vector occupies 1 word (4 bytes), the address of the interrupt vector is 4 times of the interrupt number. Each interrupt vector is the address where the corresponding interrupt processing starts.

6.4 Interrupt Input and Action in Pending State

In the NVIC module of the Cortex-M0+ processor, each interrupt input corresponds to a pending status register occupying 1 bit where the interrupt request information is stored no matter the request is acknowledged or not. When the processor begins processing this interrupt, the hardware then clears the pending status bit automatically.

The peripherals of this system use the level-triggered interrupt output. As the peripherals are connected to the NVIC, when an

interrupt event occurs, the interrupt signal will be acknowledged. This signal remains high until the processor executes the interrupt service and clears it. Inside the NVIC, when an interrupt is detected, the pending state of the interrupt is set. When the processor receives the interrupt and begins executing the interrupt service routine, the pending state is cleared. The process is shown in the below figure.

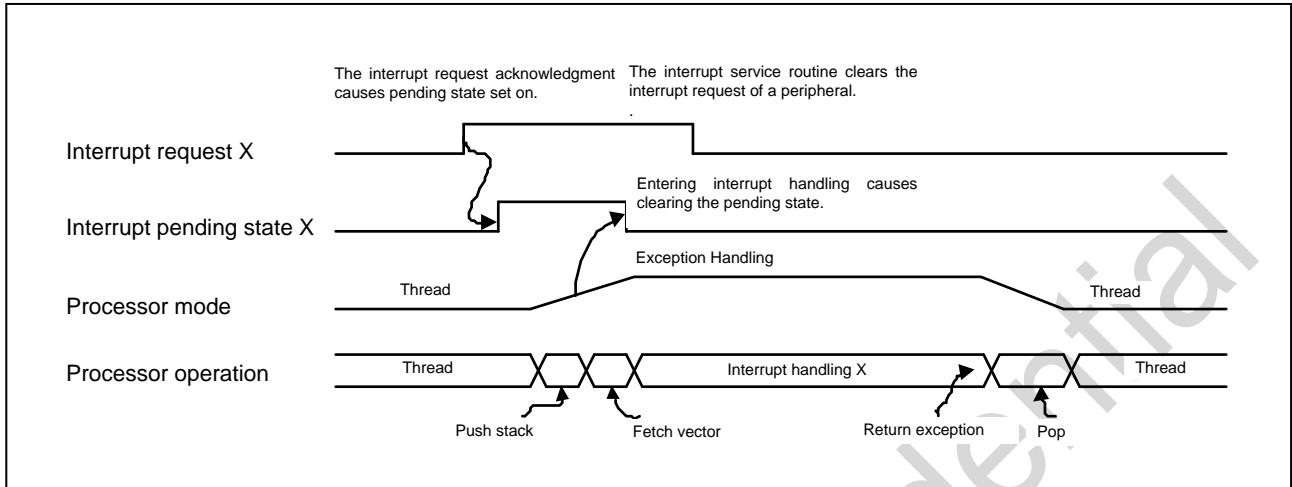


Figure 16. Interrupt Activating and Pending Status

If the interrupt request is not executed immediately and cleared by software before the acknowledgment, the processor will ignore the request and will not perform interrupt processing. The interrupt pending state can be cleared by writing to the NVIC_CLRPEND register, which is useful when setting up peripherals since the peripheral may have generated an interrupt request before setup.

If the peripheral still keeps an interrupt request while the software is clearing the pending state, the pending state takes in effect immediately. The process is shown in the below figure.

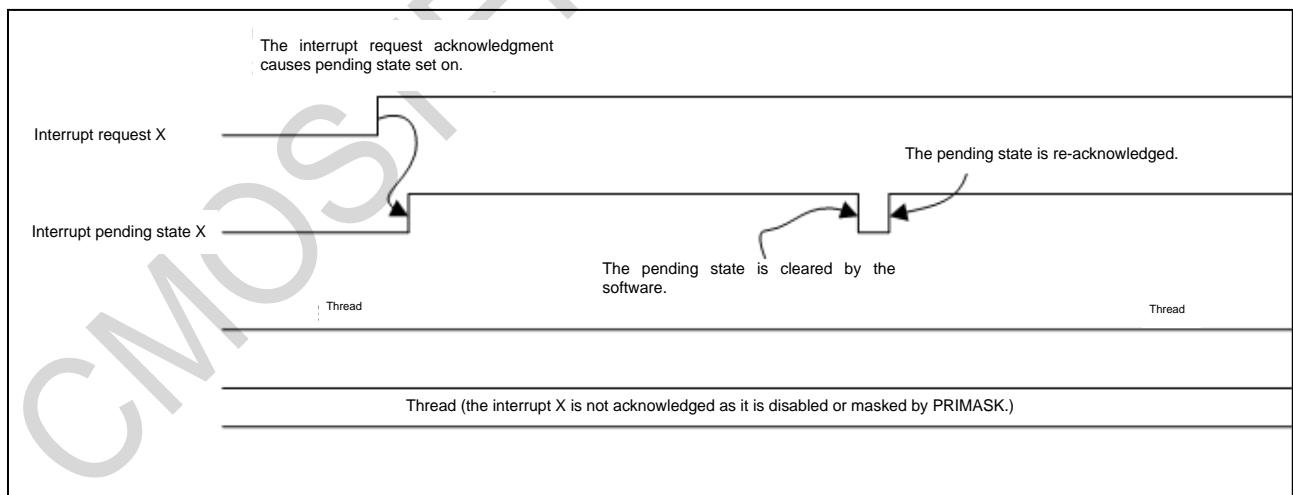


Figure 6-4. Interrupt Pending Status Is Cleared and Then Re-acknowledged

If the interrupt request generated by the peripheral is not cleared during exception processing, the pending state will be activated again after the exception is returned, so that the interrupt service routine will be executed again. The process is shown in the below figure.

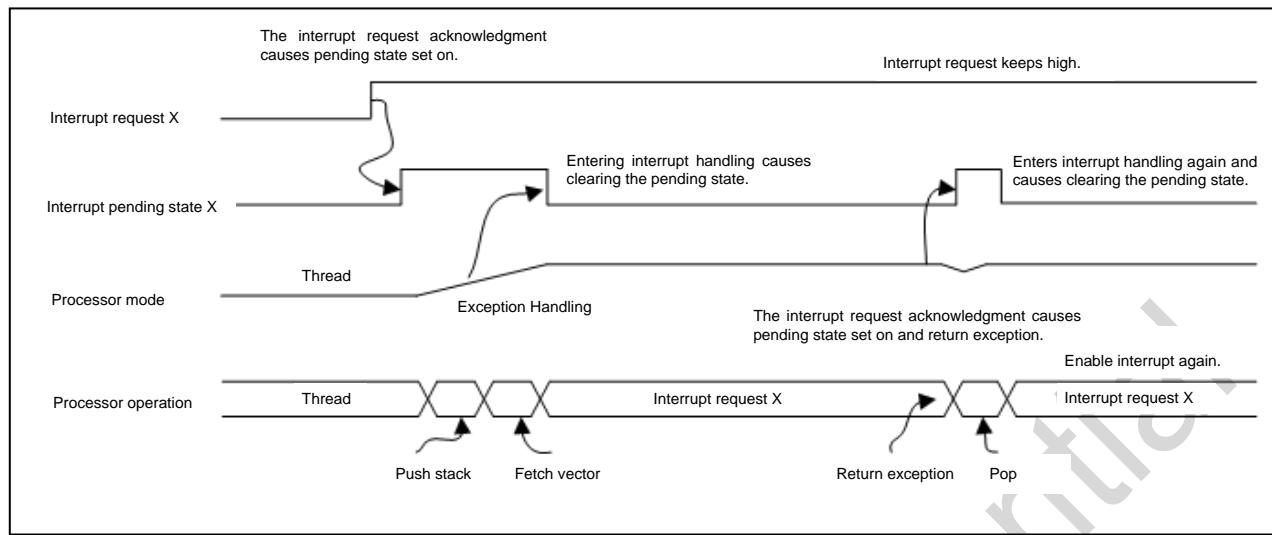


Figure 17. Execution of Interrupt Processing When the Interrupt Request Remains High While the Interrupt Exits

If a peripheral interrupt request is generated during the execution of the interrupt service routine, the request will be treated as a new interrupt request and will cause the interrupt service routine to be executed again after the interrupt is exited. The process is shown in the below figure.

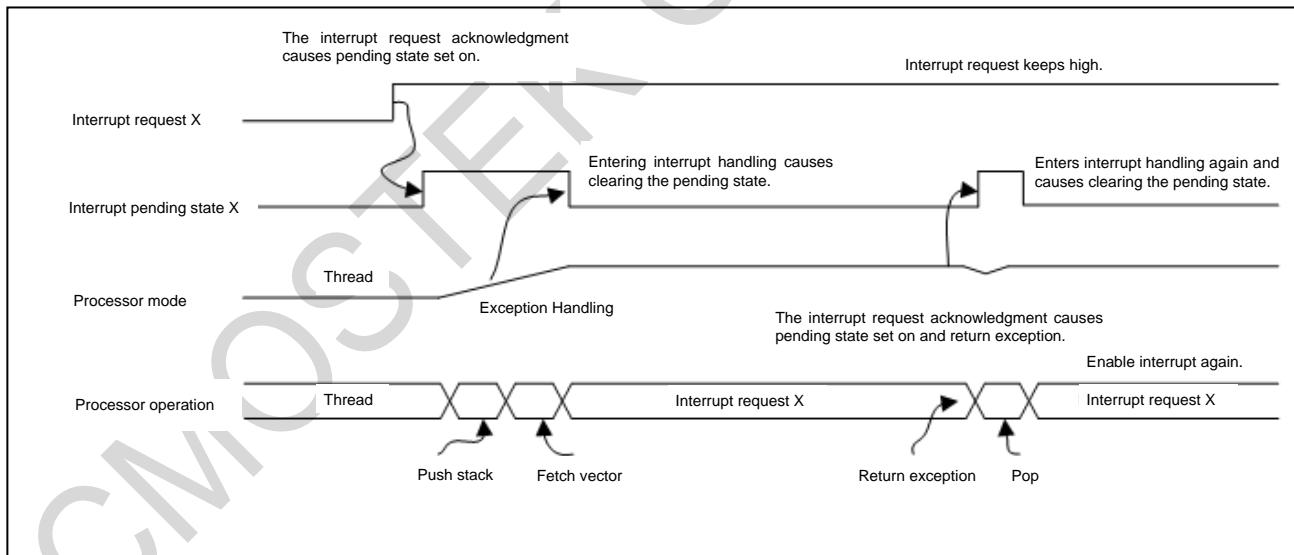


Figure 18. Interrupt Pending is Acknowledged During Interrupt Processing

6.5 Interrupt Latency

The general NVIC interrupt latency is 16 cycles. This interrupt latency time starts from the processor clock cycle of the interrupt acknowledgement and ends when the interrupt processing begins. The general interrupt latency is calculated depending on the following pre-conditions:

- The interrupt is enabled and not masked by SCS_PRIMASK or other ongoing exception handling.

- The memory system does not have any latency. Otherwise, due to the bus transmission using during interrupt processing, pushing stack, vector fetching and interrupt processing starting, if the memory system requires waiting, the latency occurring in bus transmission may cause more interrupt latency.

It may have different interrupt latency in the following cases.

- The end of the interrupt is interlocked. If another interrupt request is generated when an interrupt returns, the processor skips the stacking and pushing process, which results in less interrupt latency.
- Late arrival. If an interrupt occurs while another low priority interrupt is in pushing stack processing, due to the late arrival mechanism, the high-priority interrupt will be executed first, which results in the less interrupt latency of the high priority one.

6.6 Interrupt Source

As the NVIC of the Cortex-M0+ processor supports up to 32 external interrupts, and the system supports more than 32 external interrupt sources, some external interrupts share the same NVIC interrupt input. In addition, NMI (non-maskable interrupt) is not used in the system. The corresponding relationship between all external interrupt sources and NVIC interrupt inputs of this system is shown in the below table.

Table 47. Corresponding Relationship Between External Interrupt and NVIC Interrupt Input

NVIC Interrupt Input	External Interrupt Source	Active Mode	Sleep Mode	Deep Sleep Mode
Interrupt #0	PORT0	v	v	v
Interrupt #1	PORT1	v	v	v
Interrupt #2	PORT2	v	v	v
Interrupt #3	PORT3	v	v	v
Interrupt #4	Reserved	-	-	-
Interrupt #5	Reserved	-	-	-
Interrupt #6	UART0	v	v	-
Interrupt #7	UART1	v	v	-
Interrupt #8	LPUART	v	v	v
Interrupt #9	Reserved	-	-	-
Interrupt #10	SPI	v	v	-
Interrupt #11	Reserved	-	-	-
Interrupt #12	I2C	v	v	-
Interrupt #13	Reserved	-	-	-
Interrupt #14	TIM0	v	v	-
Interrupt #15	TIM1	v	v	-
Interrupt #16	TIM2	v	v	-
Interrupt #17	LPTIM	v	v	v
Interrupt #18	TIM4	v	v	-

NVIC Interrupt Input	External Interrupt Source	Active Mode	Sleep Mode	Deep Sleep Mode
Interrupt #19	TIM5	v	v	-
Interrupt #20	TIM6		v	-
Interrupt #21	PCA	v	v	-
Interrupt #22	WDT	v	v	v
Interrupt #23	RTC	v	v	v
Interrupt #24	ADC	v	v	-
Interrupt #25	Reserved	-	-	-
Interrupt #26	VC0	v	v	v
Interrupt #27	VC1	v	v	v
Interrupt #28	LVD	v	v	v
Interrupt #29	Reserved	-	-	-
Interrupt #30	EFCTRL/RAMCTRL	v	v	-
Interrupt #31	CLK_TRIM	v	v	-

Notes:

- As an IRQ interrupt source may multiplexed by multiple module interrupts, when CPU enters the interrupt operation, it needs to determine which module generating the interrupt, and then perform the corresponding interrupt operation.

6.7 Interrupt Structure

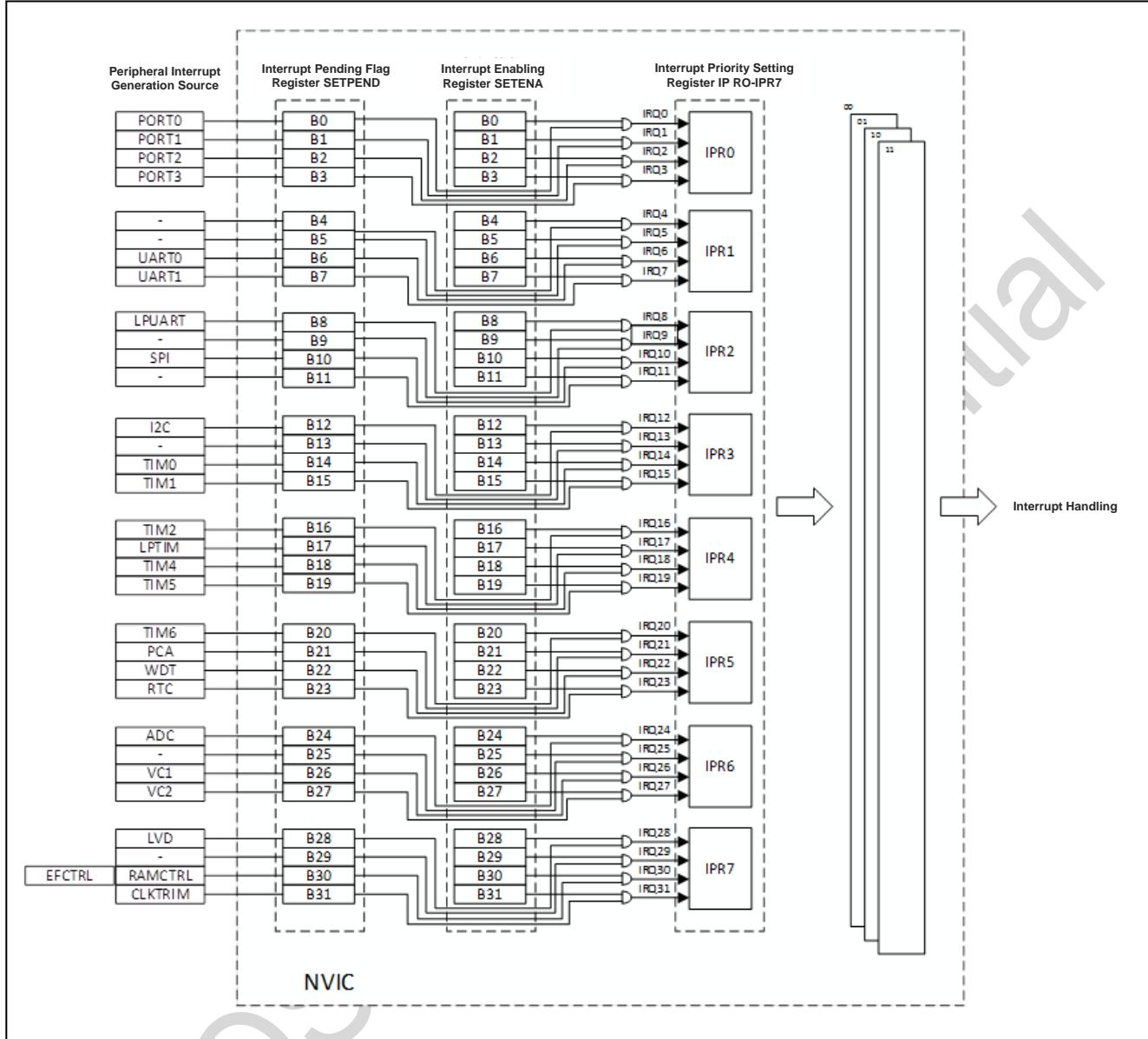


Figure 19. Interrupt Structure Diagram

See

Figure 19 for the system interrupt structure diagram. Please be noted the following indications.

- The interrupt enabling of each peripheral interrupt source is not marked in the figure. In this figure, it covers only the logic block diagram related to the interrupt signal after the peripheral interrupt generation.
- IRQ30 is shared by 2 peripheral interrupt inputs. The interrupt flags of these 2 peripherals must be read separately to identify the specific peripheral interrupt.
- If the peripheral interrupt source is high, the interrupt pending register SCS_SEPEND will be set (no matter whether the NVIC interrupt enabling register SCS_SETENA is set or not), which indicates the corresponding peripheral interrupt source has an interrupt.
- Only if the interrupt enabling register SCS_SETENA is set, the corresponding interrupt IRQ acknowledges the processor to execute the interrupt service routine.
- The peripheral interrupt source high level interrupt signal must be cleared in the interrupt program. The interrupt

- pending register SCS_SETPEND is automatically cleared by the hardware.
- Interrupt priority register SCS_IPR0- SCS_IPR7 sets the priority of the 32 interrupt sources with *00* as the highest priority and *11* as the lowest priority. When the priorities are the same, it judges the priority by the interrupt number. The less the number, the higher the priority.

6.8 Register

Base address: 0xE000 E000

Table 48. Interrupt Register

Register	Offset Address	Description
SCS_SETENA	0x100	Interrupt request enabling register
SCS_CLRENA	0x180	Interrupt request clearing enabling register
SCS_SETPEND	0x200	Interrupt setting pending register
SCS_CLRPEND	0x280	Interrupt clearing pending register
SCS_IPR0	0x400	Interrupt #0 - interrupt#3 priority register
SCS_IPR1	0x404	Interrupt #4 - interrupt #7 priority register
SCS_IPR2	0x408	Interrupt #8 - interrupt #11 priority register
SCS_IPR3	0x40C	Interrupt #12 - interrupt #15 priority register
SCS_IPR4	0x410	Interrupt #16 - interrupt #19 priority register
SCS_IPR5	0x414	Interrupt #20 -interrupt #23 priority register
SCS_IPR6	0x418	Interrupt #24 -interrupt #27 priority register
SCS_IPR7	0x41C	Interrupt #28 - interrupt #31 priority register
SCS_PRIMASK	-	Interrupt mask special register

6.8.1 Interrupt Enabling Setting Register (SCS_SETENA)

Offset address: 0x100

Reset value: 0x0000 0000

Table 49. Interrupt Enabling Setting Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SETENA[31:16]															
R/W															

Table 50. Interrupt Enabling Setting Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SETENA[15:0]															

R/W

Table 51. Interrupt Enabling Setting Register (3)

Bit	Flag	Description
31:0	SETENA [31:0]	Setting enabling interrupt #0 ~ #31. Set to 1. 0 is invalid. [0]: IRQ0 [1]: IRQ1 [2]: IRQ2 [31]: IRQ31

6.8.2 Interrupt Enabling Clearing Register (SCS_CLRENA)

Offset address: 0x180

Reset value: 0x0000 0000

Table 52. Interrupt Enabling Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLRENA															
R/W															

Table 53. Interrupt Enabling Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CLRENA															
R/W															

Table 54. Interrupt Enabling Clearing Register (3)

Bit	Flag	Description
31:0	CLRENA	Clearing enabling interrupt #0 ~ #31. Set to 1 to clear. 0 is invalid.

6.8.3 Interrupt Pending State Setting Register (SCS_SETPEND)

Offset address: 0x200

Reset value: 0x0000 0000

Table 55. Interrupt Pending State Setting Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SETPEND[31:16]															

R/W

Table 56. Interrupt Pending State Setting Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SETPEND[15:0]															
R/W															

Table 57. Interrupt Pending State Setting Register (3)

Bit	Flag	Description
31:0	SETPEND	Setting the pending state of interrupt #0 ~ #31. Set to 1. 0 is invalid. [0]:IRQ0 [1]:IRQ1 [2]:IRQ2 [31]:IRQ31

6.8.4 Interrupt Pending State Clearing Register (SCS_CLRPEND)

Offset address: 0x280

Reset value: 0x0000 0000

Table 58. Interrupt Pending State Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLRPEND[31:16]															
R/W															

Table 59. Interrupt Pending State Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CLRPEND[15:0]															
R/W															

Table 60. Interrupt Pending State Clearing Register (3)

Bit	Flag	Description
31:0	CLRPEND	Clearing the pending state of interrupt #0 ~ #31. Set to 1 to clear. 0 is invalid. [0]:IRQ0

		[1]:IRQ1 [2]:IRQ2 [31]:IRQ31
--	--	---------------------------------------------

6.8.5 Interrupt Priority Register (SCS_IPR0)

Offset address: 0x400

Reset value: 0x0000 0000

Table 61. Interrupt Priority Register SCS_IPR0 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR0[31:16]															
R/W															

Table 62. Interrupt Priority Register SCS_IPR0 (1)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR0[15:0]															
R/W															

Table 63. Interrupt Priority Register SCS_IPR0 (3)

Bit	Flag	Description
31:0	IPR0[31:0]	The priority of interrupt #0 ~ interrupt #3 [31:30]: the priority of interrupt #3 [23:22]: the priority of interrupt #2 [15:14]: the priority of interrupt #1 [7:6]: the priority of interrupt #0 00 is the highest priority, 11 the lowest.

6.8.6 Interrupt Priority Register (SCS_IPR1)

Offset address: 0x404

Reset value: 0x0000 0000

Table 64. Interrupt Priority Register SCS_IPR1 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR1[31:16]															

R/W

Table 65. Interrupt Priority Register SCS_IPR1 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR1[15:0]															
R/W															

Table 66. Interrupt Priority Register SCS_IPR1 (3)

Bit	Flag	Description
31:0	IPR1[31:0]	The priority of interrupt #4 ~ interrupt #7 [31:30]: the priority of interrupt #7 [23:22]: the priority of interrupt #6 [15:14]: the priority of interrupt #5 [7:6]: the priority of interrupt #4 00 is the highest priority, 11 the lowest.

6.8.7 Interrupt Priority Register (SCS_IPR2)

Offset address: 0x408

Reset value: 0x0000 0000

Table 67. Interrupt Priority Register SCS_IPR2 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR2[31:16]															
R/W															

Table 68. Interrupt Priority Register SCS_IPR2 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR2[15:0]															
R/W															

Table 69. Interrupt Priority Register SCS_IPR2 (3)

Bit	Flag	Description
31:0	IPR2[31:0]	The priority of interrupt #8 ~ interrupt #11. [31:30]: the priority of interrupt #11 [23:22]: the priority of interrupt #10

		[15:14]: the priority of interrupt #9 [7:6]: the priority of interrupt #8 00 is the highest priority, 11 the lowest.
--	--	------------------------------------------------------------------------------------------------------------------------------------

6.8.8 Interrupt Priority Register (SCS_IPR3)

Offset address: 0x40C

Reset value: 0x0000 0000

Table 70. Interrupt Priority Register SCS_IPR3 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR3[31:16]															
R/W															

Table 71. Interrupt Priority Register SCS_IPR3 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR3[15:0]															
R/W															

Table 72. Interrupt Priority Register SCS_IPR3 (3)

Bit	Flag	Description
31:0	IPR3[31:0]	the priority of interrupt #12 ~ interrupt #15 [31:30]: the priority of interrupt #15 [23:22]: the priority of interrupt #14 [15:14]: the priority of interrupt #13 [7:6]: the priority of interrupt #12 00 is the highest priority, 11 the lowest.

6.8.9 Interrupt Priority Register (SCS_IPR4)

Offset address: 0x410

Reset value: 0x0000 0000

Table 73. Interrupt Priority Register SCS_IPR4 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR4[31:16]															
R/W															

Table 74. Interrupt Priority Register SCS_IPR4 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR4[15:0]															
R/W															

Bit	Flag	Description
31:0	IPR4[31:0]	The priority of interrupt #16 ~ interrupt #19 [31:30]: the priority of interrupt #16 [23:22]: the priority of interrupt #17 [15:14]: the priority of interrupt #18 [7:6]: the priority of interrupt #19 00 is the highest priority, 11 the lowest.

6.8.10 Interrupt Priority Register (SCS_IPR5)

Offset address: 0x414

Reset value: 0x0000 0000

Table 75. Interrupt Priority Register SCS_IPR5 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR5[31:16]															
R/W															

Table 76. Interrupt Priority Register SCS_IPR5 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR5[15:0]															
R/W															

Table 77. Interrupt Priority Register SCS_IPR5 (3)

Bit	Flag	Description
31:0	IPR5[31:0]	The priority of interrupt #20 ~ interrupt #23. [31:30]: the priority of interrupt#20 [23:22]: the priority of interrupt#21

		[15:14]: the priority of interrupt#22 [7:6]: the priority of interrupt#23 00 is the highest priority, 11 the lowest.
--	--	----------------------------------------------------------------------------------------------------------------------------

6.8.11 Interrupt Priority Register (SCS_IPR6)

Offset address: 0x418

Reset value: 0x0000 0000

Table 78. Interrupt Priority Register SCS_IPR6 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR6[31:16]															
R/W															

Table 79. Interrupt Priority Register SCS_IPR6 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR6[15:0]															
R/W															

Table 80. Interrupt Priority Register SCS_IPR6 (3)

Bit	Flag	Description
31:0	IPR6[31:0]	The priority of interrupt #24 ~ interrupt #27. [31:30]: the priority of interrupt #24 [23:22]: the priority of interrupt #25 [15:14]: the priority of interrupt #26 [7:6]: the priority of interrupt #27 00 is the highest priority, 11 the lowest.

6.8.12 Interrupt Priority Register (SCS_IPR7)

Offset address: 0x41C

Reset value: 0x0000 0000

Table 81. Interrupt Priority Register SCS_IPR7 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR7[31:16]															
R/W															

Table 82. Interrupt Priority Register SCS_IPR7 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR7[15:0]															
R/W															

Table 83. Interrupt Priority Register SCS_IPR7 (3)

Bit	Flag	Description
31:0	IPR7[31:0]	The priority of interrupt #28 ~ interrupt #31 [31:30]: the priority of interrupt#28 [23:22]: the priority of interrupt#29 [15:14]: the priority of interrupt#30 [7:6]: the priority of interrupt#31 00 is the highest priority, 11 the lowest.

6.8.13 Interrupt Mask Special Register (SCS_PRIMASK)

Offset address: -

Reset value: 0x0000 0000

Table 84. Interrupt Mask Special Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 85. Interrupt Mask Special Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														PRIMASK	
R														R/W	

Table 86. Interrupt Mask Special Register (3)

Bit	Flag	Description
31:1	Reserved	
0	PRIMASK	When it's set, all interrupts except NMI and hardware error exceptions are masked After it's cleared, no exceptions and interrupts are blocked. This special register can be accessed through either the MSR and MRS special register operation instructions or the CPS(changing processor status) instruction. Operations on the PRIMASK register are required for time-sensitive applications.

6.9 Basic Software Operation

6.9.1 External Interrupt Enabling

Each peripheral module has its individual interrupt enabling register. Set on this interrupt enabling register first when performing interrupt operations. The detail information of this register is not covered in this chapter. Please refer to the chapters specific for each peripheral module for more details.

6.9.2 NVIC Interrupt Enabling and Clearing Enabling

The Cortex-M0+ processor supports up to 32 interrupt sources, with an interrupt enabling bit and a clear enabling bit available for each one. Therefore, it forms a 32-bit interrupt enabling register, SCS_SETENA, and a 32-bit clear enabling register, SCS_CLRENA. Users can set the corresponding bit of the SCS_SETENA register and the SCS_CLRENA register for interrupt enabling and clearing respectively.

Notes:

1. The interrupt enabling method mentioned here is specific for the processor NVIC. The interrupt generation of each peripheral depends on the interrupt control register of the corresponding peripheral, which is irrelevant to SCS_SETENA and SCS_CLRENA.

6.9.3 NVIC Interrupt Pending and Clearing

If an interrupt occurs but cannot be processed immediately, the interrupt request will be in pending state. The pending state is stored in a register and it retains the state if neither the current priority of the processor reduces lower enough to handle the pending request nor it is manually cleared.

When the processor begins to enter interrupt processing, the hardware will clear the pending state automatically.

The interrupt pending state can be accessed or modified through operating the 2 registers, interrupt pending setting register SCS_SETPEND and interrupt clearing pending register SCS_CLRPEND. The interrupt pending state register allows software to trigger an interrupt.

6.9.4 NVIC Interrupt Priority

The setting of SCS_IPR0- SCS_IPR7 register determines the priority of SCS_IRQ0 - SCS_IRQ32. The interrupt priority register should be programmed before the interrupt is enabled, which is usually done at the beginning of the program. Modifying Interrupt priority should be avoided after the interrupt is enabled. Such operation is not supported in Cortex-M0+ processor and the consequence is unpredictable.

6.9.5 NVIC Interrupt Mask

Some time-sensitive applications require all interrupts disabled for a short period of time, which can be implemented via the interrupt mask register SCS_PRIMASK. The special register SCS_PRIMASK has only 1 bit valid which defaults to 0 after reset. When this register is 0, all interrupts and exceptions are enabled. When it is set to 1, only NMI (not available in this system) and hardware error exceptions are enabled. In other words, when SCS_PRIMASK is set to 1, the processor's current priority value drops to 0 (representing the highest priority).

The SCS_PRIMASK register can be programmed in a variety of ways. In assembly language, the *MS R* instruction can be used to set and clear the SCS_PRIMASK register. In C language with CMSIS device driver library, users can use the following functions to set and clear PRIMASK

```
void __enable_irq(void); //Clear PRIMASK
void __disable_irq(void); //Set PRIMASK
```

7 Port Controller

7.1 Port Controller Introduction

The CMT2380F32 microcontroller supports 16 digital general purpose input and output ports including P01-P03, P14-P15, P23-P27, P31-P36 and 1 digital general purpose input port P00. The input and output signals of the analog signal ADC/VC/LVD, the input and output signals of each functional module (such as SPI, UART, I2C, Timer, etc.) and the input and output signals of the test and debug function can be multiplexed on the digital ports.

Each port can be configured as internal pull-up/pull-down input, high-impedance input, CMOS output, open drain output, or output with enhanced drive capability. After the chip reset, the ports are high-impedance inputs to prevent the abnormal actions of external devices caused by abnormal reset of the chip. However, to avoid leakage caused by high-impedance input, the related port configuration (configured as internal pull-up input or output) is required after chip startup. If a digital port is configured as an analog port, the digital function is blocked, no 1 and 0 output allowed. The result is 0 when CPU reads the port in the case.

All the ports can provide external interrupts, and each one can be configured as any of the 4 types, high level trigger, low level trigger, rising edge trigger or falling edge trigger. It can detect the corresponding port with interrupt triggered via querying the interrupt flag of Px_STAT[n]. In addition, each port interrupt can wakeup the chip from sleep mode/deep sleep mode to active mode.

7.2 Major Features of Port Controller

The port controller supports the following features.

- Support multiplex of analog function pin, debug pin, digital general purpose pin and digital function pin.
- Support pull-up / pull-down / normal drive / enhanced drive / push-pull output and open drain output functions
- Support interrupt in active mode / sleep mode / deep sleep mode.
- Support interrupt at high level / low level / rising edge / falling edge.

7.3 Port Controller Function Description

7.3.1 Port Multiplex Function

Port multiplex is one of the major functionalities of the port controller, which supports using a port as an analog port / test debug port / digital general port / digital function port flexibly via configuring registers.

As shown in the below table, the PxADS register is used for digital port and analog port switching. When the port is configured as an analog port, the digital function is blocked with no 1 and 0 output allowed. The result is 0 when CPU reads the port. The Px_sel registers are used for digital general purpose port and digital function port switching. Each port can be individually configured as a desired functional port based on system requirements. For the configuration information of the test and debug port, please refer to the relevant chapters.

Table 87. Port Multiplex Table

Analog		Digital							
PxADS=1		PxADS=0							
		Px_sel=0	Px_sel=1	Px_sel=2	Px_sel=3	Px_sel=4	Px_sel=5	Px_sel=6	Px_sel=7
AIN4	VCIN4	P34	PCA_CH0	UART2_TXD	TIM5_CHA	TIM0_EXT	TIM4_CHA	RTC_1Hz	TIM1_TOG
AIN5	VCIN5	P35	UART1_RXD	TIM6_CHB	UART0_RXD	TIM0_GATE	TIM4_CHB	SPI_MISO	I2C_SDA
AIN6 ADC_VREF	VCIN6	P36	UART1_RXD	TIM6_CHA	UART0_RXD	PCA_CH4	TIM5_CHA	SPI_MOSI	I2C_SCL
		P00 Reset							
AIN7 XTHI	VCIN7	P01	UART0_RXD	I2C_SDA	UART1_RXD	TIM0_TOG	TIM5_CHB	SPI_SCK	TIM2_EXT
AIN8 XTHO		P02	UART0_RXD	I2C_SCL	UART1_RXD	TIM0_TOGN	TIM6_CHA	SPI_CS	TIM2_GATE
		VSS							
		VCAP							
		VDD							
LVDIN1		P03	PCA_CH3	SPI_CS	TIM6_CHB	LPTIM_EXT	RTC_1Hz	PCA_ECI	VC0_OUT
XTLO		P15	I2C_SDA	TIM2_TOG	TIM4_CHB	LPTIM_GATE	SPI_SCK	UART0_RXD	LVD_OUT
XTLI		P14	I2C_SCL	TIM2_TOGN	PCA_ECI	ADC_RDY	SPI_CS	UART0_RXD	NC
LVDIN2	VCIN0	P23	TIM6_CHA	TIM4_CHB	TIM4_CHA	PCA_CH0	SPI_MISO	UART1_RXD	IR_OUT
AIN0		P24	TIM4_CHB	TIM5_CHB	HCLK_OUT	PCA_CH1	SPI_MOSI	UART1_RXD	VC1_OUT
LVDIN3	VCIN1	P25	SPI_SCK	PCA_CH0	TIM5_CHA	LVD_OUT	UART2_RXD	I2C_SDA	TIM1_GATE
AIN1		P26	SPI_MOSI	TIM4_CHA	TIM5_CHB	PCA_CH2	UART2_RXD	I2C_SCL	TIM1_EXT
		P27/SWDIO	SPI_MISO	TIM5_CHA	TIM6_CHA	PCA_CH3	UART0_RXD	RCH_OUT	XTH_OUT
		P31/SWCLK	LPTIM_TOG	PCA_ECI	PCLK_OUT	VC0_OUT	UART0_RXD	RCL_OUT	HCLK_OUT
AIN2	VCIN2	P32	LPTIM_TOGN	PCA_CH2	TIM6_CHB	VC1_OUT	UART1_RXD	PCA_CH4	RTC_1Hz
AIN3	VCIN3	P33	UART2_RXD	PCA_CH1	TIM5_CHB	PCA_ECI	UART1_RXD	XTL_OUT	TIM1_TOGN
Bastimer		LPTIMER	advtimer	PCA	UART	SPI	I2C	RTC	Other

7.3.2 Port Interrupt Function

Each digital general purpose port can be interrupted by an external signal source. The external signal source can be 4 types of signals, high level, low level, rising edge or falling edge signals. The corresponding interrupt enabling registers are high level interrupt enabling register, low level interrupt enabling register, rising edge interrupt enabling register and falling edge interrupt

enabling register. When the interrupts are triggered, It can detect the corresponding ports with interrupt triggered by querying the interrupt state register. The corresponding interrupt state flag can be cleared by clearing the interrupt clearing register.

7.3.3 Port Configuration Function

Each port can be configured as an internal pull-up/pull-down input, enhanced drive output, CMOS output or open-drain output based on system requirements. After the pull-up/pull-down of the port is enabled, the pull-up/pull-down is always valid regardless of the port state (input/output).

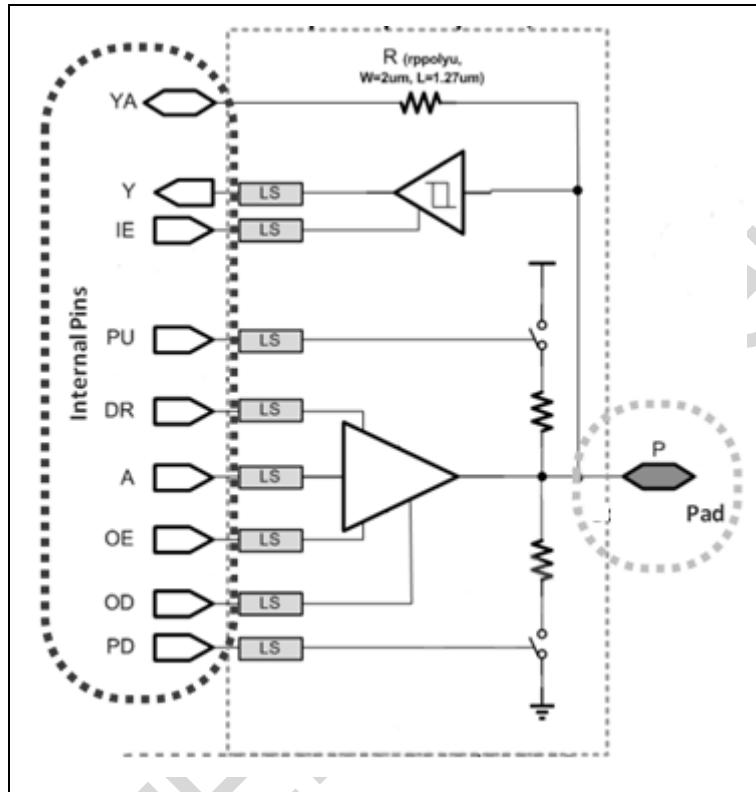


Figure 20. Port Circuit Diagram

7.4 Operating Mode

7.4.1 Port Multiplex Operation Process

The port multiplexing operation process is as follows.

1. Configure a port as an analog port
 - Set register Px_ADS to 1.
2. Configure a port as a general purpose digital port.
 - Set register Px_ADS to 0.
 - Set register Px_sel to 1.
 - Set register PxDIR to 1, representing the port direction is input. CPU can read port status at PxIN.
 - Set register PxDIR to 0, representing the port direction is output.
 - Set register PxOUT to 1, representing port output is high.

- Set register PxOUT to 0, representing port output is low.
3. Configure a port as a digital functional port.
 - Set register Px_ADS to 0.
 - Set register Px_sel to 1~7 (depends on system requirement, refer to port multiplex table).
 - Set register PxDIR (depends on system requirement).
 4. Configure a port as a test debug port.
 - Refer to the related chapters for test and debug.
 5. Configure a port with infrared output signal.

Port P23 can be configured with an infrared output signal at a frequency of 38 k.

- Set register P23_ADS to 0.
- Set register P23_sel to 7.
- Set register P2DIR[3] to 0. Port direction is output.
- Set register GPIO_CTRL1 bit14 to select the infrared signal output polarity.
- Set register P2OUT[3], gated infrared signal output.

7.4.2 Port Interrupt Operating Process

1. High level interrupt
 - Set register Px_ADS to 0
 - Set register Px_sel to 0
 - Set register PxDIR to 1
 - Set register PxHIE to 1
 - Read interrupt status register Px_STAT after interrupt triggering
 - Set register Px_ICLR to 0, clear interrupt state register
2. Low Level Interrupt
 - Set register Px_ADS to 0
 - Set register Px_sel to 0
 - Set register PxDIR to 1
 - Set register PxLIE to 1
 - Read interrupt status register Px_STAT after interrupt triggering
 - Set register Px_ICLR to 0, clear interrupt state register
3. Interrupt at rising Edge
 - Set register Px_ADS to 0
 - Set register Px_sel to 0
 - Set register PxDIR to 1
 - Set register PxRIE to 1
 - Read interrupt status register Px_STAT after interrupt triggering
 - Set register Px_ICLR to 0, clear interrupt state register Px_STAT
4. Interrupt at falling Edge
 - Set register Px_ADS to 0
 - Set register Px_sel to 0
 - Set register PxDIR to 1
 - Set register PxFIE to 1
 - Read interrupt status register Px_STAT after interrupt triggering

- Set register Px_ICLR to 0, clear interrupt state register Px_STAT.

7.4.3 Port Configuration Operating Process

1. Pull-up enabling
 - Set register PxPU to 1
2. Pull-down enabling
 - Set register PxPU to 0
 - Set register PxPD to 1
3. Enhanced drive
 - Set register PxDR to 0
4. Open drain output
 - Set register PxOD to 1

7.5 Port Controller Register Description

Base address: 0x40020C00

Table 88. Port Controller Register Description

Offset	Register Name	Access	Register Description
0x04	P01_SEL	RW	Port P01 function configuration register
0x08	P02_SEL	RW	Port P02 function configuration register
0x0c	P03_SEL	RW	Port P03 function configuration register
0x50	P14_SEL	RW	Port P14 function configuration register
0x54	P15_SEL	RW	Port P15 function configuration register
0x8c	P23_SEL	RW	Port P23 function configuration register
0x90	P24_SEL	RW	Port P24 function configuration register
0x94	P25_SEL	RW	Port P25 function configuration register
0x98	P26_SEL	RW	Port P26 function configuration register
0x9c	P27_SEL	RW	Port P27 function configuration register
0xc4	P31_SEL	RW	Port P31 function configuration register
0xc8	P32_SEL	RW	Port P32 function configuration register
0xcc	P33_SEL	RW	Port P33 function configuration register
0xd0	P34_SEL	RW	Port P34 function configuration register
0xd4	P35_SEL	RW	Port P35 function configuration register
0xd8	P36_SEL	RW	Port P36 function configuration register
0x100	P0DIR	RW	Port P0 input/output configuration register
0x104	P0IN	RO	Port P0 input value register
0x108	P0OUT	RW	Port P0 output value configuration register

Offset	Register Name	Access	Register Description
0x10c	P0ADS	RW	Port P0 analog digital configuration
0x11c	P0DR	RW	Port P0 drive capability configuration register
0x120	P0PU	RW	Port P0 pull-up enabling configuration register
0x124	P0PD	RW	Port P0 pull-down enabling configuration register
0x12c	P0OD	RW	Port P0 open drain configuration register
0x130	P0HIE	RW	Port P0 high level interrupt enabling configuration register
0x134	P0LIE	RW	Port P0 low level interrupt enabling configuration register
0x138	P0RIE	RW	Port P0 rising edge interrupt enabling configuration register
0x13c	P0FIE	RW	Port P0 falling edge interrupt enabling configuration register
0x200	P0_STAT	RO	Port P0 interrupt state register
0x210	P0_ICLR	RW	Port P0 interrupt clearing register
0x140	P1DIR	RW	Port P1 input/output configuration register
0x144	P1IN	RO	Port P1 input value register
0x148	P1OUT	RW	Port P1 output value configuration register
0x14c	P1ADS	RW	Port P1 analog digital configuration register
0x15c	P1DR	RW	Port P1 drive capability configuration register
0x160	P1PU	RW	Port P1 pull-up enabling configuration register
0x164	P1PD	RW	Port P1 pull-down enabling configuration register
0x16c	P1OD	RW	Port P1 open drain configuration register
0x170	P1HIE	RW	Port P1 high level interrupt enabling configuration register
0x174	P1LIE	RW	Port P1 low level interrupt enabling configuration register
0x178	P1RIE	RW	Port P1 rising edge interrupt enabling configuration register
0x17c	P1FIE	RW	Port P1 falling edge interrupt enabling configuration register
0x240	P1_STAT	RO	Port P1 interrupt state register
0x250	P1_ICLR	RW	Port P1 interrupt clearing register
0x180	P2DIR	RW	Port P2 input/output configuration register
0x184	P2IN	RO	Port P2 input value register
0x188	P2OUT	RW	Port P2 output value configuration register
0x18c	P2ADS	RW	Port P2 digital analog configuration register
0x19c	P2DR	RW	Port P2 drive capability configuration register
0x1a0	P2PU	RW	Port P2 pull-up enabling configuration register
0x1a4	P2PD	RW	Port P2 pull-down enabling configuration register

Offset	Register Name	Access	Register Description
0x1ac	P2OD	RW	Port P2 open drain output configuration register
0x1b0	P2HIE	RW	Port P2 high level interrupt enabling configuration register
0x1b4	P2LIE	RW	Port P2 low level interrupt enabling configuration register
0x1b8	P2RIE	RW	Port P2 rising edge interrupt enabling configuration register
0x1bc	P2FIE	RW	Port P2 falling edge interrupt enabling configuration register
0x280	P2_STAT	RO	Port P2 interrupt state register
0x290	P2_ICLR	RW	Port P2 interrupt clearing register
0x1c0	P3DIR	RW	Port P3 input/output configuration register
0x1c4	P3IN	RO	Port P3 input value register
0x1c8	P3OUT	RW	Port P3 output value register
0x1cc	P3ADS	RW	Port P3 digital analog configuration register
0x1dc	P3DR	RW	Port P3 drive capability configuration register
0x1e0	P3PU	RW	Port P3 pull-up enabling configuration register
0x1e4	P3PD	RW	Port P3 pull-down enabling configuration register
0x1ec	P3OD	RW	Port P3 open drain output configuration register
0x1f0	P3HIE	RW	Port P3 high level interrupt enabling configuration register
0x1f4	P3LIE	RW	Port P3 low level interrupt enabling configuration register
0x1f8	P3RIE	RW	Port P3 rising edge interrupt enabling configuration register
0x1fc	P3FIE	RW	Port P3 falling edge interrupt enabling configuration register
0x2c0	P3_STAT	RO	Port P3 interrupt state register
0x2d0	P3_ICLR	RW	Port P3 interrupt state register
0x300	GPIO_CTRL0	RW	Port additional function configuration register 0
0x304	GPIO_CTRL1	RW	Port additional function configuration register 1
0x308	GPIO_CTRL2	RW	Port additional function configuration register 2
0x30c	GPIO_CTRL3	RW	Port additional function configuration register 3
0x310	GPIO_CTRL4	RW	Port additional function configuration register 4

7.5.1 Port P01 Function Configuration (P01_SEL)

Offset address: 0x04

Reset value: 0x0000 0000

Table 89. Port P01 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 90. Port P01 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P01_sel			
-												R/W			

Table 91. Port P01 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P01_sel	Port P01 function selections. 000: GPIO P01 001 : UART0_RXD, UART0 module RXD signal 010: I2C_SDA, I2C module data signal 011: UART1_TXD, UART1 module TXD signal 100 : TIM0_TOG, Timer 0 module reverse signal 101: TIM5_CHB, Advanced Timer module channel 1 B signal 110 : SPI_SCK, SPI module clock signal 111 : TIM2_EXT, Timer2 module external clock input signal

7.5.2 Port P02 Function Configuration (P02_SEL)

Offset address: 0x08

Reset value: 0x0000 0000

Table 92. Port P02 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 93. Port P02 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P02_sel			
-												R/W			

Table 94. Port P02 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P02_sel	Port2 P02 function selections.

		000: GPIO P02 001: UART0_TXD, UART0 module TXD signal 010: I2C_SCL, I2C module clock signal 011: UART1_RXD, UART1 module RXD signal 100: TIM0_TOGN, Timer0 module opposite reverse output signal 101: TIM6_CHA, Advanced Timer module channel 2 A signal 110: SPI_CS, SPI module master mode chip selection signal 111: TIM2_GATE, Timer2 module gating signal
--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

7.5.3 Port P03 Function Configuration (P03_SEL)

Offset address: 0x0C

Reset value: 0x0000 0000

Table 95. Port P03 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 96. Port P03 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P03_sel		
-													R/W		

Table 97. Port P03 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P03_sel	Port P03 function selections. 000: GPIO P03 001: PCA_CH3, PCA module channel 3 capture/compare signal 010: SPI_CS, SPI module master mode chip selection signal 011: TIM6_CHB, Advanced Timer module channel 2 B signal 100: LPTIM_EXT, Timer3 module external clck input signal 101: RTC_1Hz, RTC module 1Hz input signal 110: PCA_ECI, PCA module external clck input signal 111: VC0_OUT, VC0 module output

7.5.4 Port P14 Function Configuration (P14_SEL)

Offset address: 0x50

Reset value: 0x0000 0000

Table 98. Port P14 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 99. Port P14 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P14_sel		
-													R/W		

Table 100. Port P14 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P14_sel	Port P14 function selections. 000: GPIO P14 001: I2C_SCL, I2C module clkok signal 010: TIM2_TOGN, Timer2 module opposite reverse output signal 011: PCA_ECI, PCA module external clkok input signal 100: ADC_RDY, ADC module RDY signal 101: SPI_CS, SPI module master mode chip selection signal 110: UART0_TXD, UART0 module TXD signal 111: NC

7.5.5 Port P15 Function Configuration (P15_SEL)

Offset address: 0x54

Reset value: 0x0000 0000

Table 101. Port P15 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 102. Port P15 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P15_sel		
-													R/W		

Table 103. Port P15 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P15_sel	Port P15 function selections

		000: GPIO P15 001: I2C_SDA, I2C module data signal 010: TIM2_TOG, Timer2 module reverse output signal 011: TIM4_CHB, Advanced Timer module channel 0 B signal 100: LPTIM_GATE, Timer 3 module gating signal 101: SPI_SCK, SPI module clock signal 110: UART0_RXD, UART0 module RXD signal 111: LVD_OUT, LVD module input signal
--	--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

7.5.6 Port P23 Function Configuration (P23_SEL)

Offset address: 0x8C

Reset value: 0x0000 0000

Table 104. Port P23 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 105. Port P23 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P23_sel			
-												R/W			

Table 106. Port P23 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P23_sel	Port P23 function selections. 000: GPIO P23 001: TIM6_CHA, Advanced Timer module channel 2 A signal 010: TIM4_CHB, Advanced Timer module channel 0 B signal 011: TIM4_CHA, Advanced Timer module channel 0 A signal 100: PCA_CH0, PCA module channel 0 capture/compare signal 101: SPI_MOSI, SPI module master input slave output mode chip selection signal 110: UART1_TXD, UART1 module TXD signal 111: IR_OUT, infrared output signal

7.5.7 Port P24 Function Configuration (P24_SEL)

Offset address: 0x90

Reset value: 0x0000 0000

Table 107. Port P24 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 108. Port P24 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P24_sel			
-												R/W			

Table 109. Port P24 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P24_sel	Port P24 function selections. 000: GPIO P24 001: TIM4_CHB, Advanced Timer module channel 0 B signal 010: TIM5_CHB, Advanced Timer module channel 1 B signal 011: HCLK_OUT, AHB bus clock output signal 100: PCA_CH1, PCA module channel 1 capture/compare signal 101: SPI_MOSI, SPI module master output slave input mode chip selection signal 110: UART1_RXD, UART1 module RXD signal 111: VC1_OUT, VC1 module output

7.5.8 Port P25 Function Configuration (P25_SEL)

Offset address: 0x94

Reset value: 0x0000 0000

Table 110. Port P25 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 111. Port P25 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P25_sel			
-												R/W			

Table 112. Port P25 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	

2:0	P25_sel	Port P25 function selections	
		000: GPIO P25	
001: SPI_SCK, SPI module clock signal			
010: PCA_CH0, PCA module channel 0 capture/compare signal			
011: TIM5_CHA, Advanced Timer module channel 1 A signal			
100: LVD_OUT, LVD module output signal			
101: UART2_RXD, UART2 module RXD signal			
110: I2C_SDA, I2C module data signal			
111: TIM1_GATE, Timer 1 module gating signal			

7.5.9 Port P26 Function Configuration (P26_SEL)

Offset address: 0x98

Reset value: 0x0000 0000

Table 113. Port P26 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 114. Port P26 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P26_sel			
-												R/W			

Table 115. Port P26 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P26_sel	Port P26 function selections. 000: GPIO P26 001: SPI_MOSI, SPI module master output slave input mode chip selection signal 010: TIM4_CHA, Advanced Timer module channel 0 A signal 011: TIM5_CHB, Advanced Timer module channel 1 B signal 100: PCA_CH2, PCA module channel 2 capture/compare signal 101: UART2_TXD, UART2 module TXD signal 110: I2C_SCL, I2C module clock signal 111: TIM1_EXT, Timer 1 module external clock input signal

7.5.10 Port P27 Function Configuration (P27_SEL)

Offset address: 0x9C

Reset value: 0x0000 0000

Table 116. Port P27 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 117. Port P27 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P27_sel		
-													R/W		

Table 118. Port P27 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P27_sel	Port P27 function selections. 000: GPIO P27 001: SPI_MOSI, SP I module master input slave output mode chip selection signal 010: TIM5_CHA, Advanced Timer module channel 1 A signal 011: TIM6_CHA, Advanced Timer module channel 2 A signal 100: PCA_CH3, PCA module channel 3 capture/compare signal 101: UART0_RXD, UART0 module RXD signal 110: RCH_OUT, internal 24M RC clock output signal 111: XTH_OUT, external 32M oscillator output signal

7.5.11 Port P31 Function Configuration (P31_SEL)

Offset address: 0xC4

Reset value: 0x0000 0000

Table 119. Port P31 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 120. Port P31 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P31_sel		
-													R/W		

Table 121. Port P31 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	

2:0	P31_sel	Port P31 function selections. 000: GPIO P31 001: LPTIM_TOG, Timer 3 module reverse signal 010: PCA_ECI, PCA module external clock input signal 011: PCLK_OUT, APB bus clock output signal 100: VC0_OUT, VC0 module output 101: UART0_RXD, UART0 module RXD signal 110: RCL_OUT, internal 38K RC clock output signal 111: HCLK_OUT, AHB bus clock output signal
-----	---------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

7.5.12 Port P32 Function Configuration (P32_SEL)

Offset address: 0xC8

Reset value: 0x0000 0000

Table 122. Port P32 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 123. Port P32 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P32_sel		
													R/W		

Table 124. Port P32 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P32_sel	Port P32 function selections. 000: GPIO P32 001: LPTIM_TOGN, Timer 3 module opposite reverse output signal 010: PCA_CH2, PCA module channel 2 capture/compare signal 011: TIM6_CHB, Advanced Timer module channel 1 B signal 100: VC1_OUT, VC1 module output 101: UART1_RXD, UART1 module RXD signal 110: PCA_CH4, PCA module channel 4 capture/compare signal 111: RTC_1Hz, RTC module 1Hz output signal

7.5.13 Port P33 Function Configuration (P33_SEL)

Offset address: 0xCC

Reset value: 0x0000 0000

Table 125. Port P33 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 126. Port P33 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P33_sel		
-													R/W		

Table 127. Port P33 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P33_sel	Port P33 function selections 000: GPIO P33 001: UART2_RXD, UART2 module RXD signal 010: PCA_CH1, PCA module channel 1 capture/compare signal 011: TIM5_CHB, Advanced Timer module channel 1 B signal 100: PCA_ECI, PCA module external clock input signal 101: UART1_RXD, UART1 module RXD signal 110: XTL_OUT, external 32K oscillator output signal 111: TIM1_TOGN, Timer 1 module opposite reverse output signal

7.5.14 Port P34 Function Configuration (P34_SEL)

Offset address: 0xD0

Reset value: 0x0000 0000

Table 128. Port P34 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 129. Port P34 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P34_sel		
-													R/W		

Table 130. Port P34 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	

2:0	P34_sel	Port P34 function selections 000: GPIO P34 001: PCA_CH0, PCA module channel 0 capture/compare signal 010: UART2_TXD, UART2 module TXD signal 011: TIM5_CHA, Advanced Timer module channel 1 A signal 100: TIM0_EXT, Timer 0 module external clock input signal 101: TIM4_CHA, Advanced Timer module channel 0 A signal 110: RTC_1Hz, RTC module 1Hz output signal 111: TIM1_TOG, Timer 1 module reverse signal
-----	---------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

7.5.15 Port P35 Function Configuration (P35_SEL)

Offset address: 0xD4

Reset value: 0x0000 0000

Table 131. Port P35 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 132. Port P35 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P35_sel			
-												R/W			

Table 133. Port P35 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P35_sel	Port P35 function selections. 000: GPIO P35 001: UART1_TXD, UART1 module TXD signal 010: TIM6_CHB, Advanced Timer module channel 2 B signal 011: UART0_TXD, UART0 module TXD signal 100: TIM0_GATE, Timer 0 module gating signal 101: TIM4_CHB, Advanced Timer module channel 0 B signal 110: SPI_MOSI, SPI module master input slave output mode chip selection signal 111: I2C_SDA, I2C module data signal

7.5.16 Port P36 Function Configuration (P36_SEL)

Offset address: 0xD8

Reset value: 0x0000 0000

Table 134. Port P36 Function Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 135. Port P36 Function Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													P36_sel		
-													R/W		

Table 136. Port P36 Function Configuration Register (3)

Bit	Flag	Description
31:3	Reserved	
2:0	P36_sel	Port P36 function selections. 000: GPIO P36 001: UART1_RXD, UART1 module TXD signal 010: TIM6_CHA, Advanced Timer module channel 2 A signal 011: UART0_RXD, UART0 module RXD signal 100: PCA_CH4, PCA module channel 4 capture/compare signal 101: TIM5_CHA, Advanced Timer module channel 1 A signal 110: SPI_MOSI, SPI module master output slave input mode chip selection signal 111: I2C_SCL, I2C module clock signal

7.5.17 Port P0 Input/Output Configuration Register (P0DIR)

Offset address: 0x100

Reset value: 0xffff ffff

Table 137. Port P0 Input/Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 138. Port P0 Input/Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										P0DIR3	P0DIR2	P0DIR1	Res.		
-										R/W	R/W	R/W			

Table 139. Port P0 Input/Output Configuration Register (3)

Bit	Flag	Description
31:4	Reserved	

Bit	Flag	Description
3	P0DIR3	Port P03 Input/Output configuration register. 1: configure as input 0: configure as output
2	P0DIR2	Port P02 input/output configuration register 1: configure as input 0: configure as output
1	P0DIR1	Port P01 input/output configuration register 1: configure as input 0: configure as output
0	Reserved	

7.5.18 Port P1 Input/Output Configuration Register (P1DIR)

Offset address: 0x140

Reset value: 0xffff ffff

Table 140. Port P1 Input/Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 141. Port P1 Input/Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1DIR5	P1DIR4	Reserved					
-								R/W	R/W	-					

Table 142. Port P1 Input/Output Configuration Register (3)

Bit	Flag	Description
31:6	Reserved	
5	P1DIR5	Port P15 input/output configuration register 1: configure as input 0: configure as output
4	P1DIR4	Port P14 input/output configuration register 1: configure as input 0: configure as output
3:0	Reserved	

7.5.19 Port P2 Input/Output Configuration Register (P2DIR)

Offset address: 0x180

Reset value: 0xffff ffff

Table 143. Port P2 Input/Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 144. Port P2 Input/Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2DIR7	P2DIR6	P2DIR5	P2DIR4	P2DIR3	Reserved		
-								R/W	R/W	R/W	R/W	R/W	-		

Table 145. Port P2 Input/Output Configuration Register (3)

Bit	Flag	Description
31:8	Reserved	
7	P2DIR7	Port P27 input/output configuration register 1: configure as input 0: configure as output
6	P2DIR6	Port P26 Input/Output Configuration Register 1: configure as input 0: configure as output
5	P2DIR5	Port P25 input/output configuration register 1: configure as input 0: configure as output
4	P2DIR4	Port P24 input/output configuration register 1: configure as input 0: configure as output
3	P2DIR3	Port P23 input/output configuration register 1: configure as input 0: configure as output
2:0	Reserved	

7.5.20 Port P3 Input/Output Configuration Register(P3DIR)

Offset address: 0x1C0

Reset value: 0xffff ffff

Table 146. Port P3 Input/Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 147. Port P3 Input/Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

Reserved	P3DIR6	P3DIR5	P3DIR4	P3DIR3	P3DIR2	P3DIR1	Res.
-	R/W	R/W	R/W	R/W	R/W	R/W	-

Table 148. Port P3 Input/Output Configuration Register (3)

Bit	Flag	Description
31:7	Reserved	
6	P3DIR6	Port P36 input/output configuration register 1: configure as input 0: configure as output
5	P3DIR5	Port P35 input/output configuration register 1: configure as input 0: configure as output
4	P3DIR4	Port P34 input/output configuration register 1: configure as input 0: configure as output
3	P3DIR3	Port P33 input/output configuration register 1: configure as input 0: configure as output
2	P3DIR2	Port P32 input/output configuration register 1: configure as input 0: configure as output
1	P3DIR1	Port P31 input/output configuration register 1: configure as input 0: configure as output
0	Reserved	

7.5.21 Port P0 Input Value Register (P0IN)

Offset address: 0x104

Reset value: NA

Table 149. Port P0 Input Value Registe (01)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 150. Port P0 Input Value Registe (02)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										P0IN3	P0IN2	P0IN1	P0IN0		
-										RO	RO	RO	RO		

Table 151. Port P0 Input Value Register (03)

Bit	Flag	Description
31:4	Reserved	
3	P0IN3	Port P03 input value register 1: input high 0: input low
2	P0IN2	Port P02 input value register 1: input high 0: input low
1	P0IN1	Port P01 input value register 1: input high 0: input low
0	P0IN0	Port P00 input value register 1: input high 0: input low

7.5.22 Port P1 Input Value Register(P1IN)

Offset address: 0x144

Reset value: NA

Table 152. Port P1 Input Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 153. Port P1 Input Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1IN5	P1IN4	Reserved					
-								RO	RO	-					

Table 154. Port P1 Input Value Register (3)

Bit	Flag	Description
31:6	Reserved	
5	P1IN5	Port P15 input value register 1: input high 0: input low
4	P1IN4	Port P14 input value register 1: input high 0: input low
3:0	Reserved	

7.5.23 Port P2 Input Value Register (P2IN)

Offset address: 0x184

Reset value: NA

Table 155. Port P2 Input Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 156. Port P2 Input Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2IN7	P2IN6	P2IN5	P2IN4	P2IN3	Reserved		
-		RO		RO		RO		RO		RO		-			

Table 157. Port P2 Input Value Register (3)

Bit	Flag	Description
31:8	Reserved	
7	P2IN7	Port P27 input value register 1: input high 0: input low
6	P2IN6	Port P26 input value register 1: input high 0: input low
5	P2IN5	Port P25 input value register 1: input high 0: input low
4	P2IN4	Port P24 input value register 1: input high 0: input low
3	P2IN3	Port P23 input value register 1: input high 0: input low
2:0	Reserved	

7.5.24 Port P3 Input Value Register (P3IN)

Offset address: 0x1C4

Reset value: NA

Table 158. Port P3 Input Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 159. Port P3 Input Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3IN6	P3IN5	P3IN4	P3IN3	P3IN2	P3IN1	Res.	
-								RO	RO	RO	RO	RO	RO	RO	-

Table 160. Port P3 Input Value Register (3)

Bit	Flag	Description
31:7	Reserved	
6	P3IN6	Port P36 input value register 1: input high 0: input low
5	P3IN5	Port P35 input value register 1: input high 0: input low
4	P3IN4	Port P34 input value register 1: input high 0: input low
3	P3IN3	Port P33 input value register 1: input high 0: input low
2	P3IN2	Port P32 input value register 1: input high 0: input low
1	P3IN1	Port P31 input value register 1: input high 0: input low
0	Reserved	

7.5.25 Port P0 Output Value Register (P0OUT)

Offset address: 0x108

Reset value: NA

Table 161. Port P0 Input Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 162. Port P0 Input Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P0OUT3	P0OUT2	P0OUT1	Res.				

-	R/W	R/W	R/W	-
---	-----	-----	-----	---

Bit	Flag	Description
31:4	Reserved	
3	P0OUT3	Port P03 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: input low
2	P0OUT2	Port P02 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: input low
1	P0OUT1	Port P01 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
0	Reserved	

7.5.26 Port P1 Output Value Register (P1OUT)

Offset address: 0x148

Reset value: NA

Table 163. Port P1 Input Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 164. Port P1 Input Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1OUT5	P1OUT4	Reserved					
-								R/W	R/W	-					

Table 165. Port P1 Input Value Register (3)

Bit	Flag	Description
31:6	Reserved	
5	P1OUT5	Port P15 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
4	P1OUT4	Port P14 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low

3:0	Reserved	
-----	----------	--

7.5.27 Port P2 Output Value Register (P2OUT)

Offset address: 0x188

Reset value: NA

Table 166. Port P2 Output Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2OUT7	P2OUT6	P2OUT5	P2OUT4	P2OUT3	Reserved		
-								R/W	R/W	R/W	R/W	R/W	-		

Table 167. Port P2 Output Value Register (2)

Bit	Flag	Description
31:8	Reserved	
7	P2OUT7	Port P27 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
6	P2OUT6	Port P26 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
5	P2OUT5	Port P25 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
4	P2OUT4	Port P24 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
3	P2OUT3	Port P23 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
2:0	Reserved	

7.5.28 Port P3 Output Value Register (P3OUT)

Offset address: 0x1C8

Reset value: NA

Table 168. Port P3 Output Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 169. Port P3 Output Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3OUT6	P3OUT5	P3OUT4	P3OUT3	P3OUT2	P3OUT1	Res.	
-								R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Table 170. Port P3 Input Value Register

Bit	Flag	Description
31:7	Reserved	
6	P3OUT6	Port P36 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
5	P3OUT5	Port P35 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
4	P3OUT4	Port P34 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
3	P3OUT3	Port P33 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
2	P3OUT2	Port P32 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
1	P3OUT1	Port P31 output value register 1: output high, if configured as an open-drain output, an external pull-up resistor is required to pull high. 0: output low
0	Reserved	

7.5.29 Port P0 Digital Analog Register (P0ADS)

Offset address: 0x10C

Reset value: 0x0000 0000

Table 171. Port P0 Digital Analog Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 172. Port P0 Digital Analog Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P0ADS3	P0ADS2	P0ADS1	Res.				

-	R/W	R/W	R/W	-
---	-----	-----	-----	---

Table 173. Port P0 Digital Analog Register (3)

Bit	Flag	Description
31:4	Reserved	
3	P0ADS3	Port P03 digital analog configuration register 1: configure as analog port 0: configure as digital port
2	P0ADS2	Port P02 digital analog configuration register 1: configure as analog port 0: configure as digital port
1	P0ADS1	Port P01 digital analog configuration register 1: configure as analog port 0: configure as digital port
0	Reserved	

7.5.30 Port P1 Digital Analog Register (P1ADS)

Offset address: 0x14C

Reset value: 0x0000 0000

Table 174. Port P1 Digital Analog Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 175. Port P1 Digital Analog Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										P1ADS5	P1ADS4				
-										R/W	R/W				

Table 176. Port P1 Digital Analog Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1ADS5	Port P15 digital analog configuration register 1: configure as analog port 0: configure as digital port
4	P1ADS4	Port P14 digital analog configuration register 1: configure as analog port 0: configure as digital port
3:0	Reserved	

7.5.31 Port P2 Digital Analog Register (P2ADS)

Offset address: 0x18C

Reset value: 0x0000 0000

Table 177. Port P2 Digital Analog Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 178. Port P2 Digital Analog Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2ADS7	P2ADS6	P2ADS5	P2ADS4	P2ADS3	Reserved		
-								R/W	R/W	R/W	R/W	R/W	-		

Table 179. Port P2 Digital Analog Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2ADS7	Port P27 digital analog configuration register 1: configure as analog port 0: configure as digital port
6	P2ADS6	Port P26 digital analog configuration register 1: configure as analog port 0: configure as digital port
5	P2ADS5	Port P25 digital analog configuration register 1: configure as analog port 0: configure as digital port
4	P2ADS4	Port P24 digital analog configuration register 1: configure as analog port 0: configure as digital port
3	P2ADS3	Port P23 digital analog configuration register 1: configure as analog port 0: configure as digital port
2:0	Reserved	

7.5.32 Port P3 Digital Analog Register (P3ADS)

Offset address: 0x1CC

Reset value: 0x0000 0000

Table 180. Port P3 Input Value Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 181. Port P3 Digital Analog Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3ADS6	P3ADS5	P3ADS4	P3ADS3	P3ADS2	P3ADS1	Res.	
-								R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Table 182. Port P3 Digital Analog Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3ADS6	Port P36 digital analog configuration register 1: configure as analog port 0: configure as digital port
5	P3ADS5	Port P35 digital analog configuration register 1: configure as analog port 0: configure as digital port
4	P3ADS4	Port P34 digital analog configuration register 1: configure as analog port 0: configure as digital port
3	P3ADS3	Port P33 digital analog configuration register 1: configure as analog port 0: configure as digital port
2	P3ADS2	Port P32 digital analog configuration register 1: configure as analog port 0: configure as digital port
1	P3ADS1	Port P31 digital analog configuration register 1: configure as analog port 0: configure as digital port
0	Reserved	

7.5.33 Port P0 Drive Capability Configuration Register (P0DR)

Offset address: 0x11C

Reset value: 0x0000 0000

Table 183. Port P0 Drive Capability Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 184. Port P0 Drive Capability Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P0DR3	P0DR2	P0DR1	Res.				

-	R/W	R/W	R/W	-
---	-----	-----	-----	---

Table 185. Port P0 Drive Capability Configuration Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0DR3	Port P03 drive capability configuration register 1: low drive capability 0: high drive capability
2	P0DR2	Port P02 drive capability configuration register 1: low drive capability 0: high drive capability
1	P0DR1	Port P01 drive capability configuration register 1: low drive capability 0: high drive capability
0	Reserved	

7.5.34 Port P1 Drive Capability Configuration Register (P1DR)

Offset address: 0x15C

Reset value: 0x0000 0000

Table 186. Port P1 Drive Capability Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 187. Port P1 Drive Capability Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1DR5	P1DR4	Reserved					
-								R/W	R/W	-					

Table 188. Port P1 Drive Capability Configuration Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1DR5	Port P15 drive capability configuration register 1: low drive capability 0: high drive capability
4	P1DR4	Port P14 drive capability configuration register 1: low drive capability 0: high drive capability
3:0	Reserved	

7.5.35 Port P2 Drive Capability Configuration Register (P2DR)

Offset address: 0x19C

Reset value: 0x0000 0000

Table 189. Port P2 Drive Capability Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 190. Port P2 Drive Capability Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2DR7	P2DR6	P2DR5	P2DR4	P2DR3	Reserved		
-								R/W	R/W	R/W	R/W	R/W	-		

Table 191. Port P2 Drive Capability Configuration Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2DR7	Port P27 drive capability configuration register 1: low drive capability 0: high drive capability
6	P2DR6	Port P26 drive capability configuration register 1: low drive capability 0: high drive capability
5	P2DR5	Port P25 drive capability configuration register 1: low drive capability 0: high drive capability
4	P2DR4	Port P24 drive capability configuration register 1: low drive capability 0: high drive capability
3	P2DR3	Port P23 drive capability configuration register 1: low drive capability 0: high drive capability
2:0	Reserved	

7.5.36 Port P3 Drive Capability Configuration Register (P3DR)

Offset address: 0x1DC

Reset value: 0x0000 0000

Table 192. Port P3 Drive Capability Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 193. Port P3 Drive Capability Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3DR6	P3DR5	P3DR4	P3DR3	P3DR2	P3DR1	Res.	
-								R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Table 194. Port P3 Drive Capability Configuration Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3DR6	Port P36 drive capability configuration register 1: low drive capability 0: high drive capability
5	P3DR5	Port P35 drive capability configuration register 1: low drive capability 0: high drive capability
4	P3DR4	Port P34 drive capability configuration register 1: low drive capability 0: high drive capability
3	P3DR3	Port P33 drive capability configuration register 1: low drive capability 0: high drive capability
2	P3DR2	Port P32 drive capability configuration register 1: low drive capability 0: high drive capability
1	P3DR1	Port P31 drive capability configuration register 1: low drive capability 0: high drive capability
0	Reserved	

7.5.37 Port P0 Pulling-up Enabling Configuration Register (P0PU)

Offset address: 0x120

Reset value: 0x0000 0000

Table 195. Port P0 Pulling-up Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 196. Port P0 Pulling-up Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P0PU3	P0PU2	P0PU1	Res.				
-								R/W	R/W	R/W	-				

Table 197. Port P0 Pulling-up Enabling Configuration Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0PU3	Port P03 pulling-up enabling configuration register 1: enable 0: disable
2	P0PU2	Port P02 pulling-up enabling configuration register 1: enable 0: disable
1	P0PU1	Port P01 pulling-up enabling configuration register 1: enable 0: disable
0	Reserved	

7.5.38 Port P01 Pulling-up Enabling Configuration Register (P1PU)

Offset address: 0x160

Reset value: 0x0000 0000

Table 198. Port P01 Pulling-up Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 199. Port P01 Pulling-up Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1PU5	P1PU4	Reserved					
-								R/W	R/W	-					

Table 200. Port P01 Pulling-up Enabling Configuration Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1PU5	Port P15 pulling-up enabling configuration register 1: enable 0: disable
4	P1PU4	Port P14 pulling-up enabling configuration register 1: enable 0: disable
3:0	Reserved	

7.5.39 Port P02 Pulling-up Enabling Configuration Register (P2PU)

Offset address: 0x1A0

Reset value: 0x0000 0000

Table 201. Port P02 Pulling-up Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 202. Port P02 Pulling-up Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2PU7	P2PU6	P2PU5	P2PU4	P2PU3	Reserved		
-								R/W	R/W	R/W	R/W	R/W	-		

Table 203. Port P02 Pulling-up Enabling Configuration Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2PU7	Port P27 pulling-up enabling configuration register 1: enable 0: disable
6	P2PU6	Port P26 pulling-up enabling configuration register 1: enable 0: disable
5	P2PU5	Port P25 pulling-up enabling configuration register 1: enable 0: disable
4	P2PU4	Port P24 pulling-up enabling configuration register 1: enable 0: disable
3	P2PU3	Port P23 pulling-up enabling configuration register 1: enable 0: disable
2:0	Reserved	

7.5.40 Port P03 Pulling-up Enabling Configuration Register (P3PU)

Offset address: 0x1E0

Reset value: 0x0000 0000

Table 204. Port P03 Pulling-up Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 205. Port P03 Pulling-up Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3PU6	P3PU5	P3PU4	P3PU3	P3PU2	P3PU1	Res.	
-								R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Table 206. Port P03 Pulling-up Enabling Configuration Register (3)

位	标记	功能描述
31: 7	Reserved	
6	P3PU6	Port P36 pulling-up enabling configuration register 1: enable 0: disable
5	P3PU5	Port P35 pulling-up enabling configuration register 1: enable 0: disable
4	P3PU4	Port P34 pulling-up enabling configuration register 1: enable 0: disable
3	P3PU3	Port P33 pulling-up enabling configuration register 1: enable 0: disable
2	P3PU2	Port P32 pulling-up enabling configuration register 1: enable 0: disable
1	P3PU1	Port P31 pulling-up enabling configuration register 1: enable 0: disable
0	Reserved	

7.5.41 Port P0 Pulling-down Enabling Configuration Register (P0PD)

Offset address: 0x124

Reset value: 0x0000 0000

Table 207. Port P0 Pulling-down Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 208. Port P0 Pulling-down Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P0PD3	P0PD2	P0PD1	Res.				

-	R/W	R/W	R/W	-
---	-----	-----	-----	---

Table 209. Port P0 Pulling-down Enabling Configuration Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0PD3	Port P03 pulling-down enabling configuration register 1: enable 0: disable
2	P0PD2	Port P02 pulling-down enabling configuration register 1: enable 0: disable
1	P0PD1	Port P01 pulling-down enabling configuration register 1: enable 0: disable
0	Reserved	

7.5.42 Port P1 Pulling-down Enabling Configuration Register (P1PD)

Offset address: 0x164

Reset value: 0x0000 0000

Table 210. Port P2 Pulling-down Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 211. Port P2 Pulling-down Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1PD5	P1PD4	Reserved					
-								R/W	R/W	-					

Table 212. Port P2 Pulling-down Enabling Configuration Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1PD5	Port P15 pulling-down enabling configuration register 1: enable 0: disable
4	P1PD4	Port P14 pulling-down enabling configuration register 1: enable 0: disable
3:0	Reserved	

7.5.43 Port P2 Pulling-up Enabling Configuration Register (P2PD)

Offset address: 0x1A4

Reset value: 0x0000 0000

Table 213. Port P2 Pulling-up Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 214. Port P2 Pulling-up Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2PD7	P2PD6	P2PD5	P2PD4	P2PD3	Reserved		
-								R/W	R/W	R/W	R/W	R/W	-		

Table 215. Port P2 Pulling-up Enabling Configuration Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2PD7	Port P27 pulling-down enabling configuration register 1: enable 0: disable
6	P2PD6	Port P26 pulling-down enabling configuration register 1: enable 0: disable
5	P2PD5	Port P25 pulling-down enabling configuration register 1: enable 0: disable
4	P2PD4	Port P24 pulling-down enabling configuration register 1: enable 0: disable
3	P2PD3	Port P23 pulling-down enabling configuration register 1: enable 0: disable
2:0	Reserved	

7.5.44 Port P3 Pulling-up Enabling Configuration Register (P3PD)

Offset address: 0x1E4

Reset value: 0x0000 0000

Table 216. Port P3 Pulling-up Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Reserved

Table 217. Port P3 Pulling-up Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3PD6	P3PD5	P3PD4	P3PD3	P3PD2	P3PD1	Res.	
-								R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Table 218. Port P3 Pulling-up Enabling Configuration Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3PD6	Port P36 pulling-down enabling configuration register 1: enable 0: disable
5	P3PD5	Port P35 pulling-down enabling configuration register 1: enable 0: disable
4	P3PD4	Port P34 pulling-down enabling configuration register 1: enable 0: disable
3	P3PD3	Port P33 pulling-down enabling configuration register 1: enable 0: disable
2	P3PD2	Port P32 pulling-down enabling configuration register 1: enable 0: disable
1	P3PD1	Port P31 pulling-down enabling configuration register 1: enable 0: disable
0	Reserved	

7.5.45 Port P0 Open Drain Output Configuration Register (P0OD)

Offset address: 0x12C

Reset value: 0x0000 0000

Table 219. Port P0 Open Drain Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 220. Port P0 Open Drain Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P0OD3	P0OD2	P0OD1	Res.
-												R/W	R/W	R/W	-

Table 221. Port P0 Open Drain Output Configuration Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0OD3	Port P03 open drain output configuration register 1: enable 0: disable
2	P0OD2	Port P02 open drain output configuration register 1: enable 0: disable
1	P0OD1	Port P01 open drain output configuration register 1: enable 0: disable
0	Reserved	

7.5.46 Port P1 Open Drain Output Configuration Register (P1OD)

Offset address: 0x16C

Reset value: 0x0000 0000

Table 222. Port P1 Open Drain Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 223. Port P1 Open Drain Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1OD5	P1OD4	Reserved					
-								R/W	R/W	-					

Table 224. Port P1 Open Drain Output Configuration Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1OD5	Port P15 open drain output configuration register 1: enable 0: disable
4	P1OD4	Port P14 open drain output configuration register 1: enable

		0: disable
3:0	Reserved	

7.5.47 Port P2 Open Drain Output Configuration Register (P2OD)

Offset address: 0x1AC

Reset value: 0x0000 0000

Table 225. Port P2 Open Drain Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 226. Port P2 Open Drain Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2OD7	P2OD6	P2OD5	P2OD4	P2OD3	Reserved		
-				R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-		

Table 227. Port P2 Open Drain Output Configuration Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2OD7	Port P27 open drain output configuration register 1: enable 0: disable
6	P2OD6	Port P26 open drain output configuration register 1: enable 0: disable
5	P2OD5	Port P25 open drain output configuration register 1: enable 0: disable
4	P2OD4	Port P24 open drain output configuration register 1: enable 0: disable
3	P2OD3	Port P23 open drain output configuration register 1: enable 0: disable
2:0	Reserved	

7.5.48 Port P3 Open Drain Output Configuration Register (P3OD)

Offset address: 0x1EC

Reset value: 0x0000 0000

Table 228. Port P3 Open Drain Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 229. Port P3 Open Drain Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							P3OD6	P3OD5	P3OD4	P3OD3	P3OD2	P3OD1	Res.		
-							R/W	R/W	R/W	R/W	R/W	R/W	R/W	-	

Table 230. Port P3 Open Drain Output Configuration Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3OD6	Port P36 open drain output configuration register 1: enable 0: disable
5	P3OD5	Port P35 open drain output configuration register 1: enable 0: disable
4	P3OD4	Port P34 open drain output configuration register 1: enable 0: disable
3	P3OD3	Port P33 open drain output configuration register 1: enable 0: disable
2	P3OD2	Port P32 open drain output configuration register 1: enable 0: disable
1	P3OD1	Port P31 open drain output configuration register 1: enable 0: disable
0	Reserved	

7.5.49 Port P0 High Level Interrupt Enabling Configuration Register (P0HIE)

Offset address: 0x130

Reset value: 0x0000 0000

Table 231. Port P0 High Level Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 232. Port P0 High Level Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P0HIE3	P0HIE2	P0HIE1	P0HIE0
-												R/W	R/W	R/W	R/W

Table 233. Port P0 High Level Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0HIE3	Port P03 high level interrupt enabling configuration register 1: enable 0: disable
2	P0HIE2	Port P02 high level interrupt enabling configuration register 1: enable 0: disable
1	P0HIE1	Port P01 high level interrupt enabling configuration register 1: enable 0: disable
0	P0HIE0	Port P00 high level interrupt enabling configuration register 1: enable 0: disable

7.5.50 Port P1 High Level Interrupt Enabling Configuration Register (P1HIE)

Offset address: 0x170

Reset value: 0x0000 0000

Table 234. Port P1 High Level Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 235. Port P1 High Level Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

Reserved		P1HIE5	P1HIE4	Reserved
-		R/W	R/W	-

Table 236. Port P1 High Level Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1HIE5	Port P15 high level interrupt enabling configuration register 1: enable 0: disable
4	P1HIE4	Port P14 high level interrupt enabling configuration register 1: enable 0: disable
3:0	Reserved	

7.5.51 Port P2 High Level Interrupt Enabling Configuration Register (P2HIE)

Offset address: 0x1B0

Reset value: 0x0000 0000

Table 237. Port P2 High Level Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 238. Port P2 High Level Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				P2HIE7	P2HIE6	P2HIE5	P2HIE4	P2HIE3	Reserved						
-				R/W	R/W	R/W	R/W	R/W	-						

Table 239. Port P2 High Level Interrupt Enabling Configuration Register (3)

位	标记	功能描述
31: 8	Reserved	
7	P2HIE7	Port P27 high level interrupt enabling configuration register 1: enable 0: disable
6	P2HIE6	Port P26 high level interrupt enabling configuration register 1: enable 0: disable
5	P2HIE5	Port P25 high level interrupt enabling configuration register 1: enable 0: disable
4	P2HIE4	Port P24 high level interrupt enabling configuration register

位	标记	功能描述
		1: enable 0: disable
3	P2HIE3	Port P23 high level interrupt enabling configuration register 1: enable 0: disable
2:0	Reserved	

7.5.52 Port P3 High Level Interrupt Enabling Configuration Register (P3HIE)

Offset address: 0x1F0

Reset value: 0x0000 0000

Table 240. Port P3 High Level Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 241. Port P3 High Level Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							P3HIE6	P3HIE5	P3HIE4	P3HIE3	P3HIE2	P3HIE1	Res.		
-							R/W	R/W	R/W	R/W	R/W	R/W	R/W	-	

Table 242. Port P3 High Level Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3HIE6	Port P36 high level interrupt enabling configuration register 1: enable 0: disable
5	P3HIE5	Port P35 high level interrupt enabling configuration register 1: enable 0: disable
4	P3HIE4	Port P34 high level interrupt enabling configuration register 1: enable 0: disable
3	P3HIE3	Port P33 high level interrupt enabling configuration register 1: enable 0: disable
2	P3HIE2	Port P32 high level interrupt enabling configuration register 1: enable 0: disable
1	P3HIE1	Port P31 high level interrupt enabling configuration register 1: enable 0: disable
0	Reserved	

7.5.53 Port P0 Low Level Interrupt Enabling Configuration Register (P0LIE)

Offset address: 0x134

Reset value: 0x0000 0000

Table 243. Port P0 Low Level Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 244. Port P0 Low Level Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												POLIE3	POLIE2	POLIE1	POLIE0
-												R/W	R/W	R/W	R/W

Table 245. Port P0 Low Level Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	POLIE3	Port P03 low level interrupt enabling configuration register 1: enable 0: disable
2	POLIE2	Port P02 low level interrupt enabling configuration register 1: enable 0: disable
1	POLIE1	Port P01 low level interrupt enabling configuration register 1: enable 0: disable
0	POLIE0	Port P01 low level interrupt enabling configuration register 1: enable 0: disable

7.5.54 Port P1 Low Level Interrupt Enabling Configuration Register (P1LIE)

Offset address: 0x174

Reset value: 0x0000 0000

Table 246. Port P1 Low Level Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 247. Port P1 Low Level Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P1LIE5	P1LIE4	Reserved	
-												R/W	R/W	-	

Table 248. Port P1 Low Level Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1LIE5	Port P15 low level interrupt enabling configuration register 1: enable 0: disable
4	P1LIE4	Port P14 low level interrupt enabling configuration register 1: enable 0: disable
3:0	Reserved	

7.5.55 Port P2 Low Level Interrupt Enabling Configuration Register (P2LIE)

Offset address: 0x1B4

Reset value: 0x0000 0000

Table 249. Port P2 Low Level Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 250. Port P2 Low Level Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2LIE7	P2LIE6	P2LIE5	P2LIE4	P2LIE3	Reserved		
-								R/W	R/W	R/W	R/W	R/W	-		

Table 251. Port P2 Low Level Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2LIE7	Port P27 low level interrupt enabling configuration register 1: enable 0: disable
6	P2LIE6	Port P26 low level interrupt enabling configuration register 1: enable 0: disable
5	P2LIE5	Port P25 low level interrupt enabling configuration register 1: enable 0: disable
4	P2LIE4	Port P24 low level interrupt enabling configuration register 1: enable 0: disable
3	P2LIE3	Port P23 low level interrupt enabling configuration register 1: enable 0: disable

2:0	Reserved	
-----	----------	--

7.5.56 Port P3 Low Level Interrupt Enabling Configuration Register(P3LIE)

Offset address: 0x1F4

Reset value: 0x0000 0000

Table 252. Port P3 Low Level Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 253. Port P3 Low Level Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3LIE6	P3LIE5	P3LIE4	P3LIE3	P3LIE2	P3LIE1	Res.	
-								R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Table 254. Port P3 Low Level Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3LIE6	Port P36 low level interrupt enabling configuration register 1: enable 0: disable
5	P3LIE5	Port P35 low level interrupt enabling configuration register 1: enable 0: disable
4	P3LIE4	Port P34 low level interrupt enabling configuration register 1: enable 0: disable
3	P3LIE3	Port P33 low level interrupt enabling configuration register 1: enable 0: disable
2	P3LIE2	Port P32 low level interrupt enabling configuration register 1: enable 0: disable
1	P3LIE1	Port P31 low level interrupt enabling configuration register 1: enable 0: disable
0	Reserved	

7.5.57 Port P0 Low Rising Edge Interrupt Enabling Configuration Register (P0RIE)

Offset address: 0x138

Reset value: 0x0000 0000

Table 255. Port P0 Rising Edge Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 256. Port P0 Rising Edge Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P0RIE3	P0RIE2	P0RIE1	P0RIE0
-												R/W	R/W	R/W	R/W

Table 257. Port P0 Rising Edge Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0RIE3	Port P03 rising edge interrupt enabling configuration register 1: enable 0: disable
2	P0RIE2	Port P02 rising edge interrupt enabling configuration register 1: enable 0: disable
1	P0RIE1	Port P01 rising edge interrupt enabling configuration register 1: enable 0: disable
0	P0RIE0	Port P00 rising edge interrupt enabling configuration register 1: enable 0: disable

7.5.58 Port P1 Rising Edge Interrupt Enabling Configuration Register(P1RIE)

Offset address: 0x178

Reset value: 0x0000 0000

Table 258. Port P1 Rising Edge Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 259. Port P1 Rising Edge Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1RIE5	P1RIE4	Reserved					
-								R/W	R/W	-					

Table 260. Port P1 Rising Edge Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 6	Reserved	

5	P1RIE5	Port P15rising edge interrupt enabling configuration register 1: enable 0: disable
4	P1RIE4	Port P14rising edge interrupt enabling configuration register 1: enable 0: disable
3:0	Reserved	

7.5.59 Port P2 Rising Edge Interrupt Enabling Configuration Register(P2RIE)

Offset address: 0x1B8

Reset value: 0x0000 0000

Table 261. Port P2 Rising Edge Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 262. Port P2 Rising Edge Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					P2RIE7	P2RIE6	P2RIE5	P2RIE4	P2RIE3	Reserved					
-					R/W	R/W	R/W	R/W	R/W	-					

Table 263. Port P2 Rising Edge Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2RIE7	Port P27 rising edge interrupt enabling configuration register 1: enable 0: disable
6	P2RIE6	Port P26 rising edge interrupt enabling configuration register 1: enable 0: disable
5	P2RIE5	Port P25 rising edge interrupt enabling configuration register 1: enable 0: disable
4	P2RIE4	Port P24 rising edge interrupt enabling configuration register 1: enable 0: disable
3	P2RIE3	Port P23 rising edge interrupt enabling configuration register 1: enable 0: disable
2:0	Reserved	

7.5.60 Port P3 Rising Edge Interrupt Enabling Configuration Register (P3RIE)

Offset address: 0x1F8

Reset value: 0x0000 0000

Table 264. Port P3 Rising Edge Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 265. Port P3 Rising Edge Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							P3RIE6	P3RIE5	P3RIE4	P3RIE3	P3RIE2	P3RIE1	Res.		
-							R/W	R/W	R/W	R/W	R/W	R/W	R/W	-	

Table 266. Port P3 Rising Edge Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3RIE6	Port P36 rising edge interrupt enabling configuration register 1: enable 0: disable
5	P3RIE5	Port P35 rising edge interrupt enabling configuration register 1: enable 0: disable
4	P3RIE4	Port P34 rising edge interrupt enabling configuration register 1: enable 0: disable
3	P3RIE3	Port P33 rising edge interrupt enabling configuration register 1: enable 0: disable
2	P3RIE2	Port P32 rising edge interrupt enabling configuration register 1: enable 0: disable
1	P3RIE1	Port P31 rising edge interrupt enabling configuration register 1: enable 0: disable
0	Reserved	

7.5.61 Port P0 Falling Edge Interrupt Enabling Configuration Register (P0FIE)

Offset address: 0x13C

Reset value: 0x0000 0000

Table 267. Port P0 Falling Edge Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

\Table 268. Port P0 Falling Edge Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										P0FIE3	P0FIE2	P0FIE1	P0FIE0		
-										R/W	R/W	R/W	R/W		

Table 269. Port P0 Falling Edge Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0FIE3	Port P03 falling edge interrupt enabling configuration register 1: enable 0: disable
2	P0FIE2	Port P02 falling edge interrupt enabling configuration register 1: enable 0: disable
1	P0FIE1	Port P01 falling edge interrupt enabling configuration register 1: enable 0: disable
0	P0FIE0	Port P00 falling edge interrupt enabling configuration register 1: enable 0: disable

7.5.62 Port P1 Falling Edge Interrupt Enabling Configuration Register(P1FIE)

Offset address: 0x17C

Reset value: 0x0000 0000

Table 270. Port P1 Falling Edge Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 271. Port P1 Falling Edge Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										P1FIE5	P1FIE4	Reserved			
-										R/W	R/W	-			

Table 272. Port P1 Falling Edge Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1FIE5	Port P15 falling edge interrupt enabling configuration register 1: enable 0: disable
4	P1FIE4	Port P14 falling edge interrupt enabling configuration register 1: enable 0: disable

3:0	Reserved	
-----	----------	--

7.5.63 Port P2 Falling Edge Interrupt Enabling Configuration Register (P2FIE)

Offset address: 0x1BC

Reset value: 0x0000 0000

Table 273. Port P2 Falling Edge Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 274. Port P2 Falling Edge Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					P2FIE7	P2FIE6	P2FIE5	P2FIE4	P2FIE3	Reserved					
-					R/W	R/W	R/W	R/W	R/W	-					

Table 275. Port P2 Falling Edge Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2FIE7	Port P27 falling edge interrupt enabling configuration register 1: enable 0: disable
6	P2FIE6	Port P26 falling edge interrupt enabling configuration register 1: enable 0: disable
5	P2FIE5	Port P25 falling edge interrupt enabling configuration register 1: enable 0: disable
4	P2FIE4	Port P24 falling edge interrupt enabling configuration register 1: enable 0: disable
3	P2FIE3	Port P23 falling edge interrupt enabling configuration register 1: enable 0: disable
2:0	Reserved	

7.5.64 Port P3 Falling Edge Interrupt Enabling Configuration Register (P3FIE)

Offset address: 0x1FC

Reset value: 0x0000 0000

Table 276. Port P3 Falling Edge Interrupt Enabling Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Reserved

Table 277. Port P3 Falling Edge Interrupt Enabling Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3FIE6	P3FIE5	P3FIE4	P3FIE3	P3FIE2	P3FIE1	Res.	
-								R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Table 278. Port P3 Falling Edge Interrupt Enabling Configuration Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3FIE6	Port P36 falling edge interrupt enabling configuration register 1: enable 0: disable
5	P3FIE5	Port P35 falling edge interrupt enabling configuration register 1: enable 0: disable
4	P3FIE4	Port P34 falling edge interrupt enabling configuration register 1: enable 0: disable
3	P3FIE3	Port P33 falling edge interrupt enabling configuration register 1: enable 0: disable
2	P3FIE2	Port P32 falling edge interrupt enabling configuration register 1: enable 0: disable
1	P3FIE1	Port P31 falling edge interrupt enabling configuration register 1: enable 0: disable
0	Reserved	

7.5.65 Port P0 Interrupt State Register (P0_STAT)

Offset address: 0x200

Reset value: 0x0000 0000

Table 279. Port P0 Interrupt State Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 280. Port P0 Interrupt State Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										P0STA3	P0STA2	P0STA1	P0STA0		
-										RO	RO	RO	RO		

Table 281. Port P0 Interrupt State Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0STA3	Port P03 interrupt state register 1: interrupt triggering 0: no interrupt triggering
2	P0STA2	Port P02 interrupt state register 1: interrupt triggering 0: no interrupt triggering
1	P0STA1	Port P01 interrupt state register 1: interrupt triggering 0: no interrupt triggering
0	P0STA0	Port P00 interrupt state register 1: interrupt triggering 0: no interrupt triggering

7.5.66 Port P1 Interrupt State Register (P1_STAT)

Offset address: 0x240

Reset value: 0x0000 0000

Table 282. Port P1 Interrupt State Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 283. Port P1 Interrupt State Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1STA5	P1STA4	Reserved					
-								RO	RO	-					

Table 284. Port P1 Interrupt State Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1STA5	Port P15 interrupt state register 1: interrupt triggering 0: no interrupt triggering
4	P1STA4	Port P14 interrupt state register 1: interrupt triggering 0: no interrupt triggering
3:0	Reserved	

7.5.67 Port P2 Interrupt State Register(P2_STAT)

Offset address: 0x280

Reset value: 0x0000 0000

Table 285. Port P2 Interrupt State Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 286. Port P2 Interrupt State Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				P2STA7	P2STA6	P2STA5	P2STA4	P2STA3	Reserved						
-				RO	RO	RO	RO	RO	-						

Table 287. Port P2 Interrupt State Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2STA7	Port P27 interrupt state register 1: interrupt triggering 0: no interrupt triggering
6	P2STA6	Port P26 interrupt state register 1: interrupt triggering 0: no interrupt triggering
5	P2STA5	Port P25 interrupt state register 1: interrupt triggering 0: no interrupt triggering
4	P2STA4	Port P24 interrupt state register 1: interrupt triggering 0: no interrupt triggering

3	P2STA3	Port P23 interrupt state register 1: interrupt triggering 0: no interrupt triggering
2:0	Reserved	

7.5.68 Port P3 Interrupt State Register (P3_STAT)

Offset address: 0x2C0

Reset value: 0x0000 0000

Table 288. Port P3 Interrupt State Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 289. Port P3 Interrupt State Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3STA6	P3STA5	P3STA4	P3STA3	P3STA2	P3STA1	Res.	
-								RO	RO	RO	RO	RO	RO	-	

Table 290. Port P3 Interrupt State Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3STA6	Port P36 interrupt state register 1: interrupt triggering 0: no interrupt triggering
5	P3STA5	Port P35 interrupt state register 1: interrupt triggering 0: no interrupt triggering
4	P3STA4	Port P34 interrupt state register 1: interrupt triggering 0: no interrupt triggering
3	P3STA3	Port P33 interrupt state register 1: interrupt triggering 0: no interrupt triggering
2	P3STA2	Port P32 interrupt state register 1: interrupt triggering 0: no interrupt triggering
1	P3STA1	Port P31 interrupt state register 1: interrupt triggering

		0: no interrupt triggering
0	Reserved	

7.5.69 Port P0 Interrupt Clearing Register (P0_ICLR)

Offset address: 0x210

Reset value: 0xffff ffff

Table 291. Port P0 Interrupt Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 292. Port P0 Interrupt Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												P0CLR3	P0CLR2	P0CLR1	P0CLR0
-												R/W	R/W	R/W	R/W

Table 293. Port P0 Interrupt Clearing Register (3)

Bit	Flag	Description
31: 4	Reserved	
3	P0CLR3	Port P03 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
2	P0CLR2	Port P02 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
1	P0CLR1	Port P01 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
0	P0CLR0	Port P00 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag

7.5.70 Port P1 Interrupt State Register (P1_ICLR)

Offset address: 0x250

Reset value: 0xffff ffff

Table 294. Port P1 Interrupt Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 295. Port P1 Interrupt Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P1CLR5	P1CLR4	Reserved					
-								R/W	R/W	-					

Table 296. Port P1 Interrupt Clearing Register (3)

Bit	Flag	Description
31: 6	Reserved	
5	P1CLR5	Port P15 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
4	P1CLR4	Port P14 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
3:0	Reserved	

7.5.71 Port P2 Interrupt Clearing Register(P2_ICLR)

Offset address: 0x290

Reset value: 0xffff ffff

Table 297. Port P2 Interrupt Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 298. Port P2 Interrupt Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P2CLR7	P2CLR6	P2CLR5	P2CLR4	P2CLR3	Reserved		
-								R/W	R/W	R/W	R/W	R/W	-		

Table 299. Port P2 Interrupt Clearing Register (3)

Bit	Flag	Description
31: 8	Reserved	
7	P2CLR7	Port P27 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
6	P2CLR6	Port P26 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
5	P2CLR5	Port P25 interrupt clearing register

		1: keep interrupt flag 0: clear interrupt flag
4	P2CLR4	Port P24 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
3	P2CLR3	Port P23 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
2:0	Reserved	

7.5.72 Port P3 Interrupt Clearing Register(P3_ICLR)

Offset address: 0x2D0

Reset value: 0xffff ffff

Table 300. Port P3 Interrupt Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 301. Port P3 Interrupt Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								P3CLR6	P3CLR5	P3CLR4	P3CLR3	P3CLR2	P3CLR1	Res.	
-								R/W	R/W	R/W	R/W	R/W	R/W	-	

Table 302. Port P3 Interrupt Clearing Register (3)

Bit	Flag	Description
31: 7	Reserved	
6	P3CLR6	Port P36 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
5	P3CLR5	Port P35 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
4	P3CLR4	Port P34 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
3	P3CLR3	Port P33 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
2	P3CLR2	Port P32 interrupt clearing register 1: keep interrupt flag 0: clear interrupt flag
1	P3CLR1	Port P31 interrupt clearing register

		1: keep interrupt flag 0: clear interrupt flag
0	Reserved	

7.5.73 Port Additional Function Configuration Register 0 (GPIO_CTRL0)

Offset address: 0x300

Reset value: 0x0000 0001

Table 303. Port Additional Function Configuration Register 0 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 304. Port Additional Function Configuration Register 0 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															IESEL
-															R/W

Table 305. Port Additional Function Configuration Register 0 (3)

Bit	Flag	Description
31: 1	Reserved	
0	IESEL	Port interrupt mode selection register 1: DEEPSLEEP mode 0: ACTIVE/SLEEP mode

7.5.74 Port Additional Function Configuration Register 1 (GPIO_CTRL1)

Offset address: 0x304

Reset value: 0x0000 0000

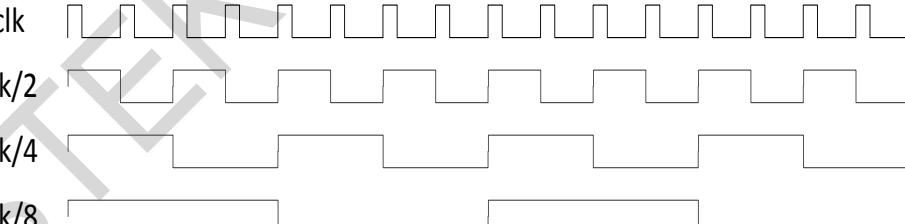
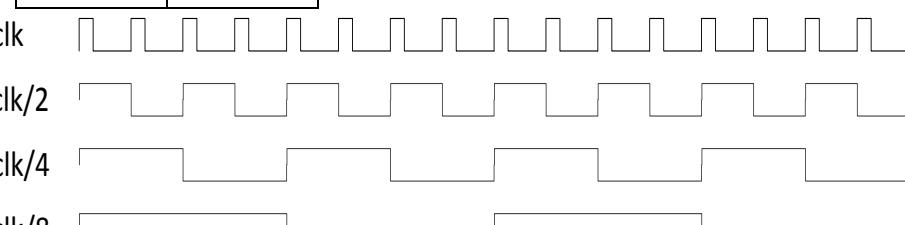
Table 306. Port Additional Function Configuration Register 1 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 307. Port Additional Function Configuration Register 1 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ir_pol	hclk_en	pclk_en	hclk_sel		pclk_sel	ssn_sel				ext_clk_sel				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W				R/W				

Table 308. Port Additional Function Configuration Register 1 (3)

Bit	Flag	Description									
31: 15	Reserved										
14	ir_pol	IR output polarity selection.	0 – positive output 1 – negative output								
13	hclk_en	Hclk output gating control.	0 – gating control 1 – output								
12	pclk_en	Pclk output gating control.	0 –gating control 1 –output								
11:10	hclk_sel	Hclk output dividing selections	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0 0</td><td>hclk</td></tr> <tr><td>0 1</td><td>hclk/2</td></tr> <tr><td>1 0</td><td>hclk/4</td></tr> <tr><td>1 1</td><td>hclk/8</td></tr> </table> 	0 0	hclk	0 1	hclk/2	1 0	hclk/4	1 1	hclk/8
0 0	hclk										
0 1	hclk/2										
1 0	hclk/4										
1 1	hclk/8										
9:8	pclk_sel	pclk output dividing selection	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0 0</td><td>pclk</td></tr> <tr><td>0 1</td><td>pclk/2</td></tr> <tr><td>1 0</td><td>pclk/4</td></tr> <tr><td>1 1</td><td>pclk/8</td></tr> </table> 	0 0	pclk	0 1	pclk/2	1 0	pclk/4	1 1	pclk/8
0 0	pclk										
0 1	pclk/2										
1 0	pclk/4										
1 1	pclk/8										
7:4	ssn_sel	SPI SSN signal source selections									

Bit	Flag	Description	
		0000	high level
		0001	P35
		0010	P36
		0011	P01
		0100	P02
		0101	P03
		0110	P15
		0111	P14
		1000	P23
		1001	P24
		1010	P25
		1011	P26
		1100	P27
		1101	P31
		1110	P32
		1111	P33
3:0	ext_clk_sel	external clock signal source selections	
		0000	high level
		0001	P35
		0010	P36
		0011	P01
		0100	P02
		0101	P03
		0110	P15
		0111	P14
		1000	P23
		1001	P24
		1010	P25
		1011	P26
		1100	P27
		1101	P31
		1110	P32
		1111	P33

7.5.75 Port Additional Function Configuration Register 2 (GPIO_CTRL2)

Offset address: 0x308

Reset value: 0x0000 0000

Table 309. Port Additional Function Configuration Register 2 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 310. Port Additional Function Configuration Register 2 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					pca_cap4_sel	pca_cap3_sel		pca_cap2_sel	pca_cap1_sel		pca_cap0_sel				
-					R/W	R/W		R/W	R/W		R/W				

Table 311. Port Additional Function Configuration Register 2 (3)

Bit	Flag	Description			
31:10	Reserved				
9:8	pca_cap4_sel	PCA capture channel 4 signal source selection			
		0 0	PCA_CH4		
		0 1	UART0_RXD		
		1 0	UART1_RXD		
		1 1	UART2_RXD		
7:6	pca_cap3_sel	PCA capture channel 3 signal source selection			
		0 0	PCA_CH3		
		0 1	UART0_RXD		
		1 0	UART1_RXD		
		1 1	UART2_RXD		
5:4	pca_cap2_sel	PCA capture channel 2 signal source selection			
		0 0	PCA_CH2		
		0 1	UART0_RXD		
		1 0	UART1_RXD		
		1 1	UART2_RXD		
3:2	pca_cap1_sel	PCA capture channel 1 signal source selection			
		0 0	PCA_CH1		
		0 1	UART0_RXD		
		1 0	UART1_RXD		
		1 1	UART2_RXD		
1:0	pca_cap0_sel	PCA capture channel 0 signal source selection			
		0 0	PCA_CH0		
		0 1	UART0_RXD		
		1 0	UART1_RXD		
		1 1	UART2_RXD		

7.5.76 Port Additional Function Configuration Register 3 (GPIO_CTRL3)

Offset address: 0x30C

Reset value: 0x0000 0000

Table 312. Port Additional Function Configuration Register 3 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 313. Port Additional Function Configuration Register 3 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				tm6_a_sel		tm5_a_sel		tm4_a_sel		tm6_b_sel		tm5_b_sel		tm4_b_sel	
-						R/W		R/W		R/W		R/W		R/W	

Table 314. Port Additional Function Configuration Register 3 (3)

Bit	Flag	Description	
31:12	Reserved		
11:10	tm6_a_sel	Timer6 A channel signal source selection	
		0 0	TIM6_CHA
		0 1	UART0_RXD
		1 0	UART1_RXD
		1 1	UART2_RXD
9:8	tm5_a_sel	Timer5 A channel signal source selection	
		0 0	TIM5_CHA
		0 1	UART0_RXD
		1 0	UART1_RXD
		1 1	UART2_RXD
7:6	tm4_a_sel	Timer4 A channel signal source selection	
		0 0	TIM4_CHA
		0 1	UART0_RXD
		1 0	UART1_RXD
		1 1	UART2_RXD
5:4	tm6_b_sel	Timer6 B channel signal source selection	
		0 0	TIM6_CHB
		0 1	UART0_RXD
		1 0	UART1_RXD
		1 1	UART2_RXD
3:2	tm5_b_sel	Timer5 B channel signal source selection	
		0 0	TIM5_CHB
		0 1	UART0_RXD
		1 0	UART1_RXD
		1 1	UART2_RXD
1:0	tm4_b_sel	Timer4 B channel signal source selection	
		0 0	TIM4_CHB
		0 1	UART0_RXD
		1 0	UART1_RXD
		1 1	UART2_RXD

7.5.77 Port Additional Function Configuration Register 4 (GPIO_CTRL4)

Offset address: 0x310

Reset value: 0x0000 0000

Table 315. Port Additional Function Configuration Register 4 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 316. Port Additional Function Configuration Register 4 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								tm3_gate_sel	tm2_gate_sel	tm1_gate_sel	tm0_gate_sel				
-								R/W	R/W	R/W	R/W				

Table 317. Port Additional Function Configuration Register 4 (3)

Bit	Flag	Description								
31:8	Reserved									
7:6	tm3_gate_sel	Timer3 gate input signal source selection <table border="1" style="margin-left: 20px;"> <tr> <td>0 0</td><td>TIM3_GATE</td></tr> <tr> <td>0 1</td><td>UART0_RXD</td></tr> <tr> <td>1 0</td><td>UART1_RXD</td></tr> <tr> <td>1 1</td><td>UART2_RXD</td></tr> </table>	0 0	TIM3_GATE	0 1	UART0_RXD	1 0	UART1_RXD	1 1	UART2_RXD
0 0	TIM3_GATE									
0 1	UART0_RXD									
1 0	UART1_RXD									
1 1	UART2_RXD									
5:4	tm2_gate_sel	Timer2 gate input signal source selection <table border="1" style="margin-left: 20px;"> <tr> <td>0 0</td><td>TIM2_GATE</td></tr> <tr> <td>0 1</td><td>UART0_RXD</td></tr> <tr> <td>1 0</td><td>UART1_RXD</td></tr> <tr> <td>1 1</td><td>UART2_RXD</td></tr> </table>	0 0	TIM2_GATE	0 1	UART0_RXD	1 0	UART1_RXD	1 1	UART2_RXD
0 0	TIM2_GATE									
0 1	UART0_RXD									
1 0	UART1_RXD									
1 1	UART2_RXD									
3:2	tm1_gate_sel	Timer1 gate input signal source selection <table border="1" style="margin-left: 20px;"> <tr> <td>0 0</td><td>TIM1_GATE</td></tr> <tr> <td>0 1</td><td>UART0_RXD</td></tr> <tr> <td>1 0</td><td>UART1_RXD</td></tr> <tr> <td>1 1</td><td>UART2_RXD</td></tr> </table>	0 0	TIM1_GATE	0 1	UART0_RXD	1 0	UART1_RXD	1 1	UART2_RXD
0 0	TIM1_GATE									
0 1	UART0_RXD									
1 0	UART1_RXD									
1 1	UART2_RXD									
1:0	tm0_gate_sel	Timer0 gate input signal source selection <table border="1" style="margin-left: 20px;"> <tr> <td>0 0</td><td>TIM0_GATE</td></tr> <tr> <td>0 1</td><td>UART0_RXD</td></tr> <tr> <td>1 0</td><td>UART1_RXD</td></tr> <tr> <td>1 1</td><td>UART2_RXD</td></tr> </table>	0 0	TIM0_GATE	0 1	UART0_RXD	1 0	UART1_RXD	1 1	UART2_RXD
0 0	TIM0_GATE									
0 1	UART0_RXD									
1 0	UART1_RXD									
1 1	UART2_RXD									

8 FLASH Controller

8.1 Introduction

This device contains a 32 kbyte FLASH memory, which is divided into 64 sectors with a capacity of 512 bytes for each. This module supports erasing, programming and reading operations on the memory. Moreover, it enables erasing protection of FLASH memory and write protection of control registers.

8.2 Structure Diagram

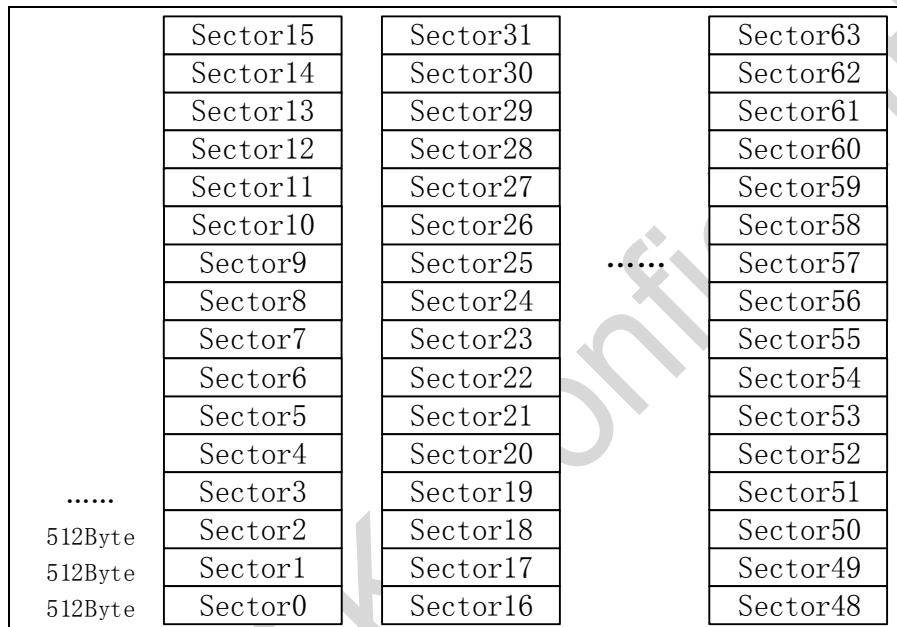


Figure 21. Memory Sector Partition

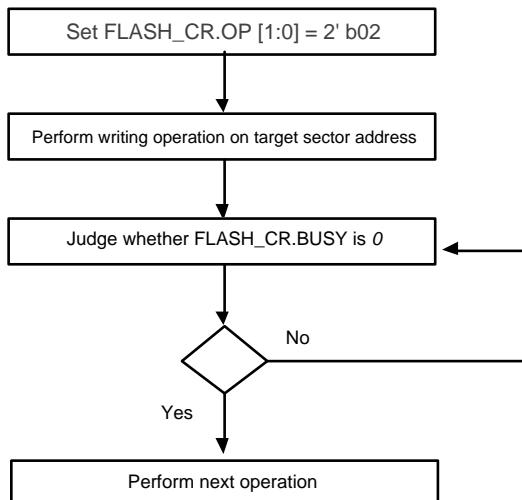
8.3 Function Description

The controller supports read and write operations on FLASH with 3 types of bit width, byte (8 bits), half-word (16 bits), and word (32 bits). Note that the address of the byte operation must be aligned by byte, the target address of the Half-word operation must be aligned by half-word (the lowest bit of the address is 1'b0), and the address of the Word operation must be aligned by word (the lowest two bits of the address are 2' B00). If the address of the read or write operation is not aligned according to the bit width specification, the operation is invalid and the system enters hard fault error interrupt.

8.3.1 Erasing Operation

- **Sector Erasing**

Sector erasing operations follow steps as shown in the below figure.

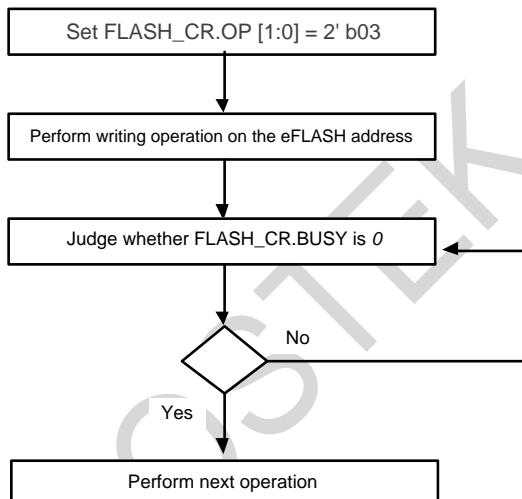
**Notes:**

1. The controller ignores the lower 9 bits of the target address as long as it is within the address range of the eFLASH.
2. This writing operation is for triggering sector erasing operation so that the written data is ignored by the controller.
3. If the current erasing instruction is executed in eFLASH, CPU fetching will pause with hardware automatically staying in waiting status until the eFLASH exits from eFLASH BUSY status.
4. If the current erasing instruction is executed in RAM, CPU fetching will not pause. The software should judge whether eFLASH is in BUSY status.

Figure 22..Sector Erasing Operation Steps

- **Chip Erasing**

Chip erasing operations follow steps as shown in the below figure.

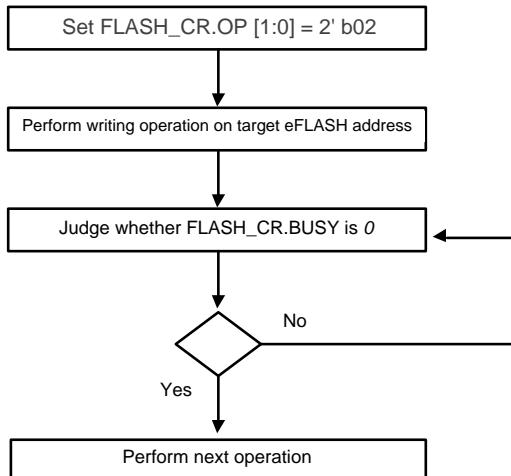
**Notes:**

1. The controller ignores the lower 15 bits of the target address as long as it is within the address range of the eFLASH.
2. This writing operation is for triggering sector erasing operation so that the written data is ignored by the controller.
3. If the current erasing instruction is executed in eFLASH, CPU fetching will pause with hardware automatically staying in waiting status until the eFLASH exits from eFLASH BUSY status.
4. If the current erasing instruction is executed in RAM, CPU fetching will not pause. The software should judge whether eFLASH is in BUSY status.

Figure 23. Chip Erasing Operation Steps

8.3.2 Writing Operation

Writing operations follow steps as shown in the below figure.



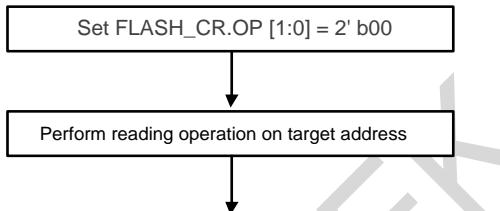
Notes:

- 1 . If the current erasing instruction is executed in eFLASH, CPU fetching will pause with hardware automatically staying in waiting status until the eFLASH exits from eFLASH BUSY status.
2. If the current erasing instruction is executed in RAM, CPU fetching will not pause. The software should judge whether eFLASH is in BUSY status.

Figure 24. Writing Operation Steps

8.3.3 Reading Operation

Reading operations follow steps as shown in the below figure.



Notes:

- 1.The first step, setting FLASH_CR.OP [1:0], can be omitted actually and reading operations can be performed regardless of the value of FLASH_CR.OP [1:0].

Figure 25. Reading Operation Steps

8.4 Erase Time

The FLASH memory has strict requirements against the time of control signals for erasing and programming operations. A erasing and programming operation failure will happen if the effective time of the control signal exceeds the specific design requirements. Such signal control in this module is designed according to the default system clock of 4 MHz. If the system clock frequency is changed, the time parameters of such control signals should be changed accordingly.

The corresponding time parameter registers include, FLASH_TNVS, FLASH_TPGS, FLASH_TPROG, FLASH_TSERASE, FLASH_TMERASE, FLASH_TPRCV, FLASH_TSRCV and FLASH_TMRCV. For example, if the system clock is increased from the default 4 MHz to 8 MHz, the value of the above FLASH_Tx register should be set to twice the default value, meaning that it remains the result of the current Tsysclk*FLASH_Tx equal to the default value.

The below table lists the corresponding flash erase time parameters at different frequencies.

Table 318. Corresponding Flash Erase Time Parameters at Different Frequencies

Parameter	4 M (default)	8 M	16 M	24 M	32 M
Tnvs	0x20	0x40	0x80	0xC0	0x100
Tpgs	0x17	0x2E	0x5C	0x8A	0xB8
Tprog	0x1B	0x36	0x6C	0xA2	0xD8
Tserase	0x4650	0x8CA0	0x11940	0x1A5E0	0x23280
Tmerase	0x222E0	0x445C0	0x88B80	0xCD140	0x111700
Tprcv	0x18	0x30	0x60	0x90	0xC0
Tsrcv	0xF0	0x1E0	0x3C0	0x5A0	0x780
Tmrcv	0x3E8	0x7D0	0xFA0	0x1770	0x1F40

8.5 Reading Waiting Cycle

The CPU reads one instruction per cycle, so the CPU operating frequency (meaning the system clock frequency) cannot handle clock frequency higher than 24 MHz. When the system clock frequency exceeds 24 MHz, it must insert the waiting cycles for CPU instruction fetch.

In this device, by setting the register bit FLASH_CR.WAIT to 1, it can insert one waiting cycle for CPU instruction fetch. This way, the system can operate at a clock frequency higher than 24 MHz. The CPU reads instruction code in the FLASH memory every two cycles. The operating frequency of this system is up to 32 MHz and there is no design for inserting more than 1 CPU waiting cycle.

8.6 Erase Protection

8.6.1 Erase Protection Bit

As discussed above, the entire 32 kbyte FLASH memory is divided into 64 sectors with corresponding erase protection bit available for each one. The protection bit register FLASH_SLOCK.MAINPL[15:0] defaults to 0000, meaning that erasing is not allowed. A sector can only be erased only if the corresponding protection bit is changed to 1.

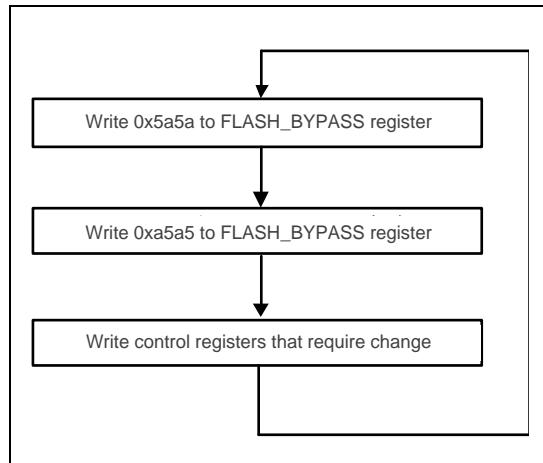
When any sector in the FLASH memory is erase-protected, the chip erasing of the FLASH is automatically disabled by the controller with the alarm flag set and the interrupt signal generated. To perform chip erasing, the value of the protection bit register FLASH_SLOCK.SLOCK[15:0] must be modified to FFFF.

8.6.2 PC Address Erasing Protection

When the CPU runs a program in FLASH, if it happened that the current PC pointer falls in the sector address range of the area currently being erased by software, the erasing operation will be also automatically disabled by the controller with the alarm flag set and the interrupt signal generated.

8.7 Register Writing Protection

Normally, the controller of this module disables write operations, which can be changed only by writing a specific sequence. The specific steps are as follows.

**Figure 26. Writing Register Bypass Sequence**

Notes:

1. Write 0x5a5a followed by 0xa5a5 successively with no other write operations in-between, otherwise the Bypass sequence will be invalid and it needs rewriting the 0x5a5a-0xa5a5 sequence in the case.

8.8 Register

Base address: 0x4002 0000

Table 319. Flash Controller Register

Register	Offset Address	Description
FLASH_TNVS	0x00	Tnvs time parameter
FLASH_TPGS	0x04	Tpgs time parameter
FLASH_TPROG	0x08	Tprog time parameter
FLASH_TSERASE	0x0C	Tserase time parameter
FLASH_TMERASE	0x10	Tmerase time parameter
FLASH_TPRCV	0x14	Tprcv time parameter
FLASH_TSRCV	0x18	Tsrcv time parameter
FLASH_TMRCV	0x1C	Tmrcv time parameter
FLASH_CR	0x20	Controlling register
FLASH_IFR	0x24	Interrupt flag register
FLASH_ICLR	0x28	Interrupt flag clearing register
FLASH_BYPASS	0x2C	0x5a5a-0xa5a5 Bypass sequence register
FLASH_SLOCK	0x30	Sector erasing protection register

- TNVS parameter register (FLASH_TNVS)

Offset address: 0x00

Reset value: 0x0000 0020

Table 320. TNVS Parameter Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 321. TNVS Parameter Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								TNVS							
R								R/W							

Table 322. TNVS Parameter Register (3)

Bit	Flag	Description
31:9	Reserved	
8:0	TNVS	Calculation formula: Tnvs = TNVS * Tsysclk = 32 * 250 ns = 8 us (reset value)

- TPGS Parameter Register (FLASH_TPGS)

Offset address: 0x04

Reset value: 0x0000 0017

Table 323. TPGS Parameter Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 324. TPGS Parameter Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								TPGS							
R								R/W							

Table 325. TPGS Parameter Register (3)

Bit	Flag	Description
31:8	RESERVED	
7:0	TPGS	Calculation formula: Tpgs = TPGS * Tsysclk = 23 * 250 ns = 5.75 us (reset value)

- TPROG Parameter Register (FLASH_TPROG)

Offset address: 0x08

Reset value: 0x0000 001B

Table 326. TPROG Parameter Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 327. TPROG Parameter Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								TPROG							
R								R/W							

Table 328. TPROG Parameter Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	TPROG	Calculation formula: Tpgs = TPROG * Tsysclk = 27 * 250 ns = 6.75 us (reset value)

- **TSERASE Register (FLASH_TSERASE)**

Offset address: 0x0C

Reset value: 0x0000 4650

Table 329. TSERASE Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 330. TSERASE Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								TSERASE							
R								R/W							

Table 331. TSERASE Register (3)

Bit	Flag	Description
31:18	RESERVED	
17:0	TSERASE	Calculation formula: Tserase = TSERASE * Tsysclk = 18000 * 250 ns = 4.5ms (reset value)

- **TMERASE Parameter Register (FLASH_TMERASE)**

Offset address: 0x10

Reset value: 0x000222E0

Table 332. TMERASE Parameter Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										TMERASE[20:16]					
R										R/W					

Table 333. TMERASE Parameter Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMERASE[15:0]															
R/W															

Table 334. TMERASE Parameter Register (3)

Bit	Flag	Description
31:21	Reserved	
20:0	TMERASE	calculation formula: Tmerase=TMERASE*Tsysclk=140000*250ns=35ms (Reset value)

- **TPRCV Parameter Register (FLASH_TPRCV)**

Offset address: 0x14

Reset value: 0x0000 0018

Table 335. TPRCV Parameter Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 336. TPRCV Parameter Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										TPRCV					
R										R/W					

Table 337. TPRCV Parameter Register (3)

Bit	Flag	Description
31:12	Reserved	
11:0	TPRCV	calculation formula: Tprcv=TPRCV*Tsysclk=24*250ns=6us (reset value)

- **TSRCV Parameter Register (FLASH_TSRCV)**

Offset address: 0x18

Reset value: 0x0000 00F0

Table 338. TSRCV Parameter Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 339. TSRCV Parameter Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				TSRCV											
R				R/W											

Table 340. TSRCV Parameter Register (3)

Bit	Flag	Description
31:12	Reserved	
11:0	TSRCV	calculation formula: Tsrcv=TSRCV*Tsysclk=240*250ns=60us (reset value)

- **TMRCV Parameter Register (FLASH_TMRCV)**

Offset address: 0x1C

Reset value: 0x0000 03E8

Table 341. TMRCV Parameter Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 342. TMRCV Parameter Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				TMRCV											
R				R/W											

Table 343. TMRCV Parameter Register (3)

Bit	Flag	Description
31:13	RESERVED	
12:0	TMRCV	calculation formula: Tmrcv=TMRCV*Tsysclk=1000*250ns=250us (Reset value)

- CR Register (FLASH_CR)

Offset address: 0x20

Reset value: 0x0000 0000

Table 344. CR Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 345. CR Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved							IE		BUSY		Reserved		WAIT		OP	
R							R/W		R		R		R/W		R/W	

Table 346. CR Register (3)

Bit	Flag	Description
31:7	Reserved	
6:5	IE	IE[6]: interrupt for erasing protected address in FLASH. 0: disabled, 1: enabled. IE[5]: interrupt for erasing PC address in FLASH. 0: disabled, 1: enabled.
4	BUSY	Idle/busy flag. 0: idle, 1: busy.
3	Reserved	
2	WAIT	Reading FLASH waiting period. 0: zero waiting cycle, 1: one zero waiting cycle.
1:0	OP	FLASH operations. 00: reading, 01: program, 10: sector erasing , 11: chip erasing.

- IFR Register (FLASH_IFR)

Offset address: 0x24

Reset value: 0x0000 0000

Table 347. IFR Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 348. IFR Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												IF1		IF0	
												R		R	

Table 349. IFR Register (3)

Bit	Flag	Description
31:2	Reserved	
1	IF1	Erasing protection alarm interrupt flag.
0	IF0	Erasing PC address alarm interrupt flag.

- **ICLR Register (FLASH_ICLR)**

Offset address: 0x28

Reset value: 0x0000 000F

Table 350. ICLR Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 351. ICLR Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
														ICLR1	ICLR0
R														W	W

Table 352. ICLR Register (3)

Bit	Flag	Description
31:4	Reserved	
3:2	Reserved	Writing is invalid. It is read as 0x3.
1	ICLR1	Clearing protection alarm interrupt flag. Writing 0 is for clearing. Writing 1 is invalid.
0	ICLR0	Clearing PC address interrupt flag. Writing 0 is for clearing. Writing 1 is invalid.

- **BYPASS Register (FLASH_BYPASS)**

Offset address: 0x2C

Reset value: 0x0000 0000

Table 353. BYPASS Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 354. BYPASS Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

BYSEQ
W

Table 355. BYPASS Register (3)

Bit	Flag	Description
31:16	Reserved	
15:0	BYSEQ	It must write the sequence 0x5a5a-0xa5a5 to the BYSEQ[15:0] register before modifying this module register, that is the register can only be modified once each time the Bypass sequence is written. To modify the register again, must write the 0x5a5a-0xa5a5 sequence again.

- **SLOCK Register (FLASH_SLOCK)**

Offset address: 0x30

Reset value: 0x0000 0000

Table 356. SLOCK Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 357. SLOCK Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLOCK															
R/W															

Table 358. SLOCK Register (3)

Bit	Flag	Description
31:16	Reserved	
15:0	SLOCK	<p>Sector erasing protection bit. 0: erasing is disabled, 1: erasing is enabled.</p> <p>SLOCK[0] is corresponding to sector 0-1-2-3.</p> <p>SLOCK[1] is corresponding to sector 4-5-6-7.</p> <p>SLOCK[2] is corresponding to sector 8-9-10-11.</p> <p>SLOCK[3] is corresponding to sector 12-13-14-15.</p> <p>.....</p> <p>SLOCK[15] is corresponding to sector 60-61-62-63.</p>

9 RAM Controller

9.1 Introduction

This product contains a static SRAM with a capacity of 2 /4 kbyte. It supports 3 types of read and write operations, byte (8 bits), half-word (16 bits), and word (32 bits). Reading and writing operations can be performed at the system clock frequency with no need waiting for cycles. In addition, the controller enables parity function, with the supports of parity check on each byte of SRAM data and parity error interrupt generation.

9.2 Functional Description

9.2.1 Bit Width for Reading and Writing

The controller supports read and write operations on FLASH with 3 types of bit width, byte (8 bits), half-word (16 bits), and word (32 bits). Note that the address of the byte operation must be aligned by byte, the target address of the half-word operation must be aligned by half-word (the lowest bit of the address is 1'b0), and the address of the word operation must be aligned by word (the lowest two bits of the address are 2' B00). If the address of the read or write operation is not aligned according to the bit width specification, the operation is invalid and the system enters the hard fault error interrupt.

9.2.2 Parity

This controller supports the parity of SRAM data. This function is disabled by default after reset. When the parity function is enabled, during writing to the SRAM, each byte the data is parity-checked with the 1-bit check value along with the 8-bit data stored in the SRAM. When reading data from the SRAM, the controller reads 8-bit data and 1-bit check value and performs parity check. The parity error flag is set in case of check error with the interrupt generated if the interrupt is enabled.

Notes:

- When parity is enabled, the SRAM must be initialized before reading the SRAM data, otherwise the parity alarm flag or interrupt may be triggered by mistake.

9.3 Register

Base address: 0x4002 0400

Table 359. RAM Controller Register

Register	Offset address	Description
RAM_CR	0x00	Controlling register
RAM_ERRADDR	0x04	Error address register
RAM_IFR	0x08	Error interrupt flag register
RAM_ICLR	0x0C	Error interrupt flag clearing register

9.3.1 Controlling Register (RAM_CR)

Offset address: 0x00

Reset value: 0x0000 0000

Table 360. Controlling Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 361. Controlling Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															
R															

Table 362. Controlling Register (3)

Bit	Flag	Description
31:2	RESERVED	
1	IE	Error alarm interrupt enabling signal. 1: enable alarm interrupt, 0: disable alarm interrupt.
0	CHKEN	Parity enable signal. 1: enable parity, 0: disable parity.

9.3.2 Parity Error Address Register (RAM_ERRADDR)

Offset address: 0x04

Reset value: 0x0000 0000

Table 363. Parity Error Address Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 364. Parity Error Address Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				ERRADDR											
R				R											

Table 365. Parity Error Address Register (3)

Bit	Flag	Description
31:12	Reserved	
11:0	ERRADDR	12-bit parity error byte address

9.3.3 Error Interrupt Flag Register (RAM_IFR)

Offset address: 0x08

Reset value: 0x0000 0000

Table 366. Error Interrupt Flag Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 367. Error Interrupt Flag Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															
R															

Table 368. Error Interrupt Flag Register (3)

Bit	Flag	Description
31:1	RESERVED	
0	ERR	Parity error flag

9.3.4 Error Interrupt Flag Clearing Register (RAM_ICLR)

Offset address: 0x0C

Reset value: 0x0000 0001

Table 369. Error Interrupt Flag Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 370. Error Interrupt Flag Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															
R															

Table 371. Error Interrupt Flag Clearing Register (3)

Bit	Flag	Description
31:1	Reserved	
0	ERRCLR	Error interrupt flag clearing bit. 1: invalid 0: clear

10 Base Timer

10.1 Base Timer Introduction

The base timer contains 3 timers, Timer 0/1/2. Timer 0/1/2 function exactly the same. Timer 0/1/2 are synchronous timers/counters that can be used as timers/counters for 16-bit auto-reload function or as 32-bit timers/counters without reload function. Timer 0/1/2 can count external pulses or implement system timing.

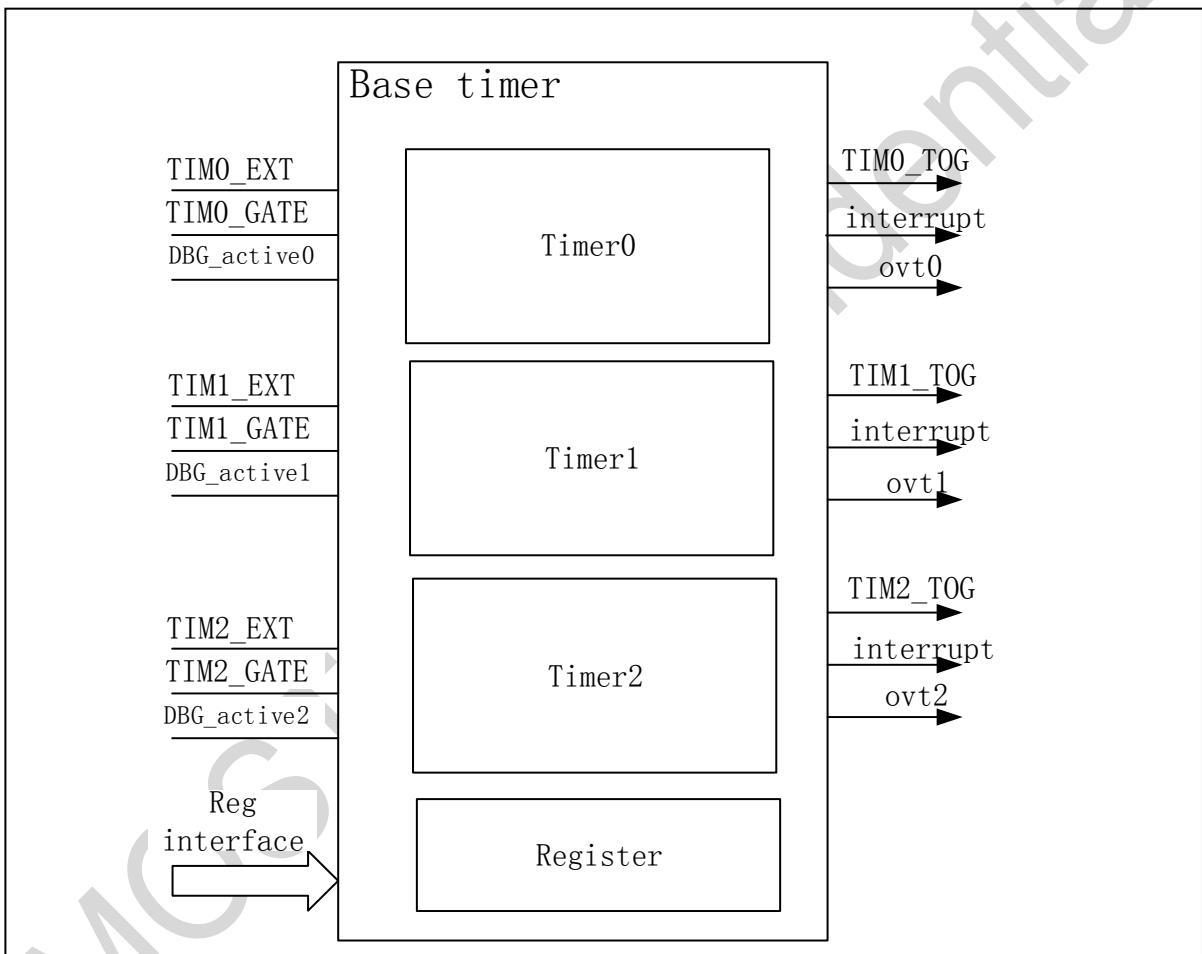


Figure 27. Base Timer Block Diagram

10.2 Base Timer Function Description

Each of Timer 0/1/2 has independent control start signal, external input clock and gate signal.

Timer 0/1/2 uses EXT, GATE for counting functions with EXT used for the counter's external input clock signal and GATE for the effective level counting enabling signal. When the gate function is enabled, the counter counts only if the external input GATE level is effective, otherwise the counter does not count. Gating enables the use of CR.gate control. The Gating feature is disabled by default. The gate level selection is controlled using CR.GATE_P. The gate is active high by default and can be canged to

active low by setting CR.GATE_P to 1.

TIM 0/1/2 uses PCLK and GATE for timing functions with PCLK applied as the internal input clock signal of the timer, and GATE as the effective level counting enabling signal. When the gate function is enabled, the timer counts only if the external input GATE level is effective, otherwise the timer stops counting. Gating enabling uses CR.Gate for controlling. The default gating feature is disabled. The gate level selection is controlled using CR.Gate_P. The gate is active high by default and can be changed to active low by setting CR.GATE_P to 1. The timing function can be configured with pre-division using CR.PRS to control the division ratio.

Table 372. Use CR.PRS to Control Division Ratio

PRS [2:0]	000	001	010	011	100	101	110	111
Frequency dividing ratio	1	2	4	8	16	32	64	256

The TIM0 /1/2 timer/counter supports 2 operation modes, which are selected by setting the MD in the timer control register (CR) with mode 1 referring to a 32-bit free counting mode and mode 2 referring to a 16-bit reload mode.

In 32-bit free counting mode, it counts up to the maximum value 0xFFFFFFFF and generates an interrupt in case of overflow with the timer/counter value becoming 0X00000000 and counting continued. In 16-bit reload mode, it counts up to the maximum value 0xFFFF and generates an interrupt in case of overflow with the timer/event counter reloaded with ARR value and counting up continued then.

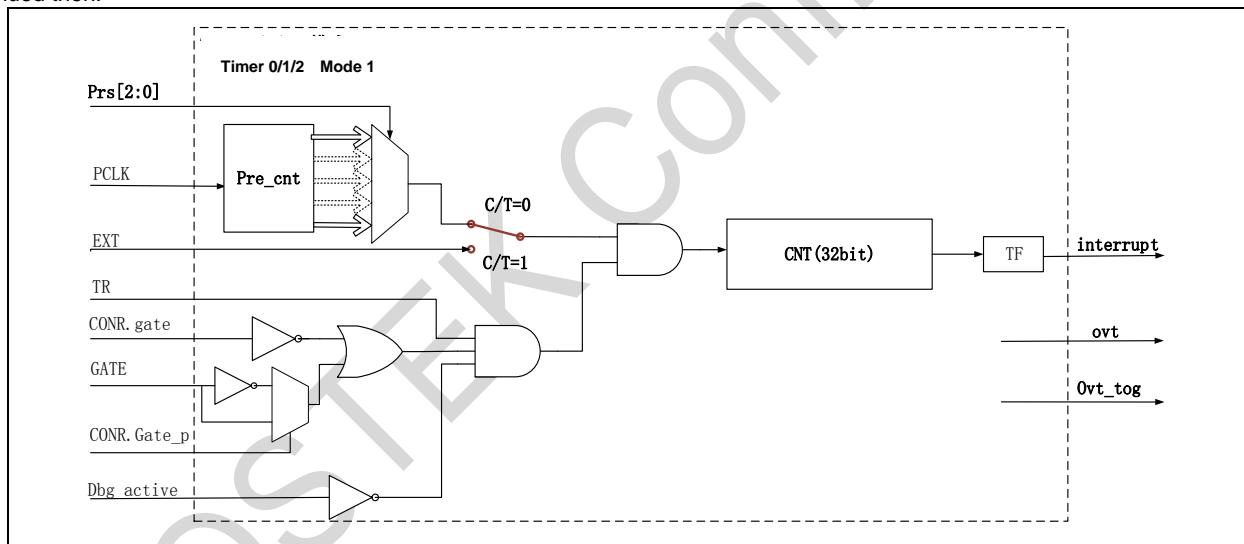


Figure 28. Timer Mode 1 Block Diagram

In mode 2, reload mode, needs to pay consideration that, if a too short timing time is set, it may not have enough time for interrupt processing and may result in interrupt loss.

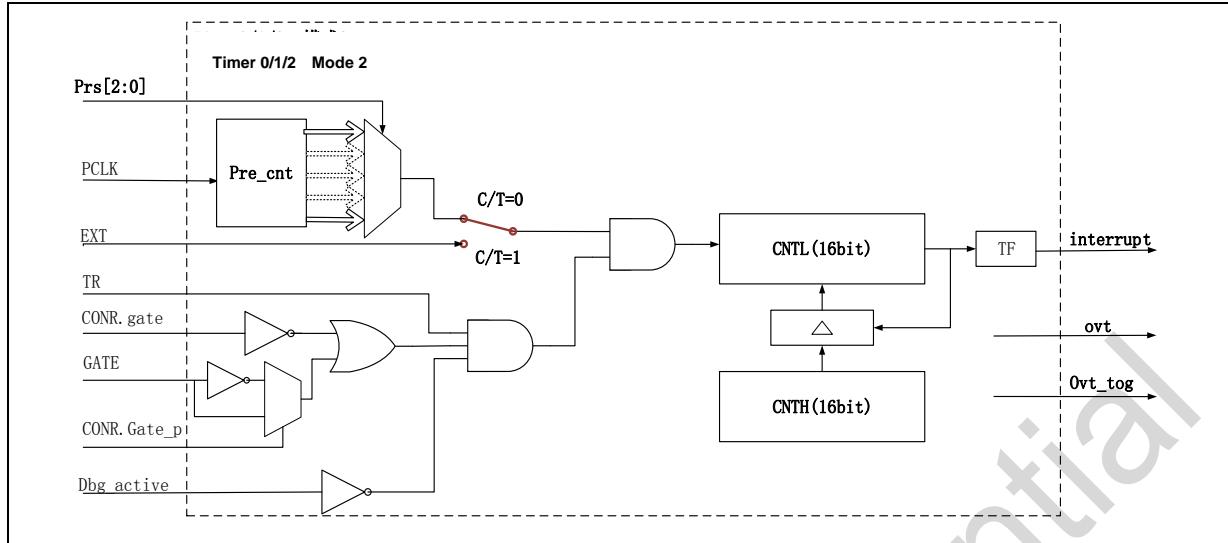


Figure 29. Timer Mode 2 Block Diagram

Mode 1 is a 32-bit timer/counter. After startup, if the corresponding timer TR is set to 1, the timer starts running. It starts counting from the initial value T_{set} by the register and generates an interrupt in case of overflow, then the timer counter continues counting from 0. Mode 2 is a 16-bit reload timer/counter. After startup, it starts counting up from the initial value of the register CNT and generates an interrupt in case of counting up to the maximum value of 0xFFFF with the counter CNT reloaded with ARR value and counting up continued then.

10.2.1 Counting Function

The counting function is used to measure the number of times an event has occurred. In the counting function, the counter is increased by 1 on the falling edge of each corresponding input clock. The input signal is sampled by the internal Pclk clock, so the external input clock frequency cannot exceed the system's Pclk clock. When counting to the maximum, it overflows and generates an interrupt. The interrupt flag requires software removal.

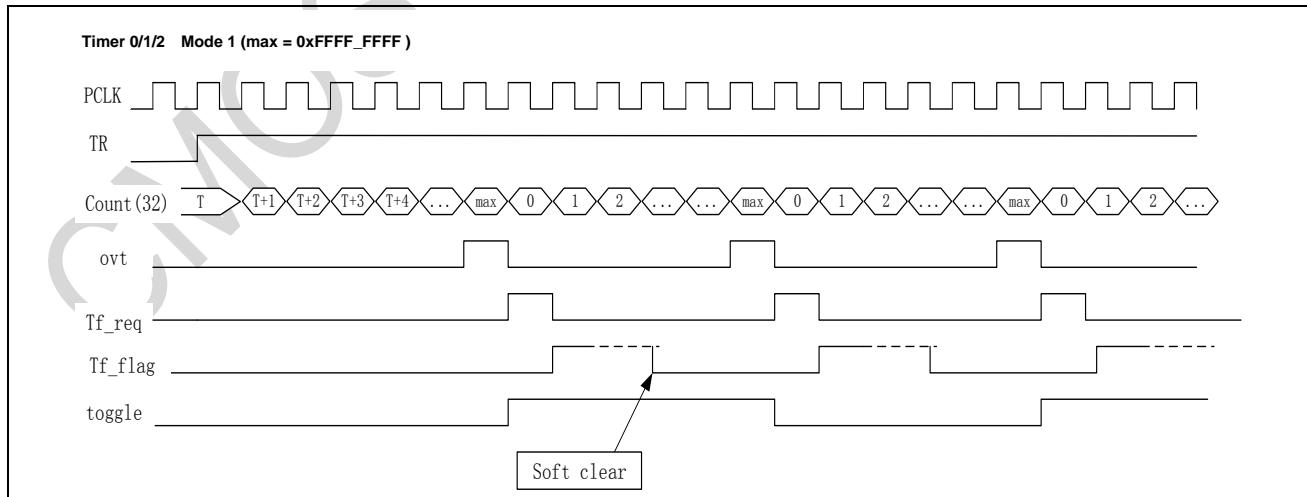


Figure 30. Mode 1 Timing Diagram

10.2.2 Timing Function

The timing function is to generate periodic timing. In the timing function, the timer is pre-divided, the timer is accumulated once in each clock cycle per pre-divided frequency. When counting to the maximum, it will overflow and generate an interrupt. The interrupt flag requires software removal.

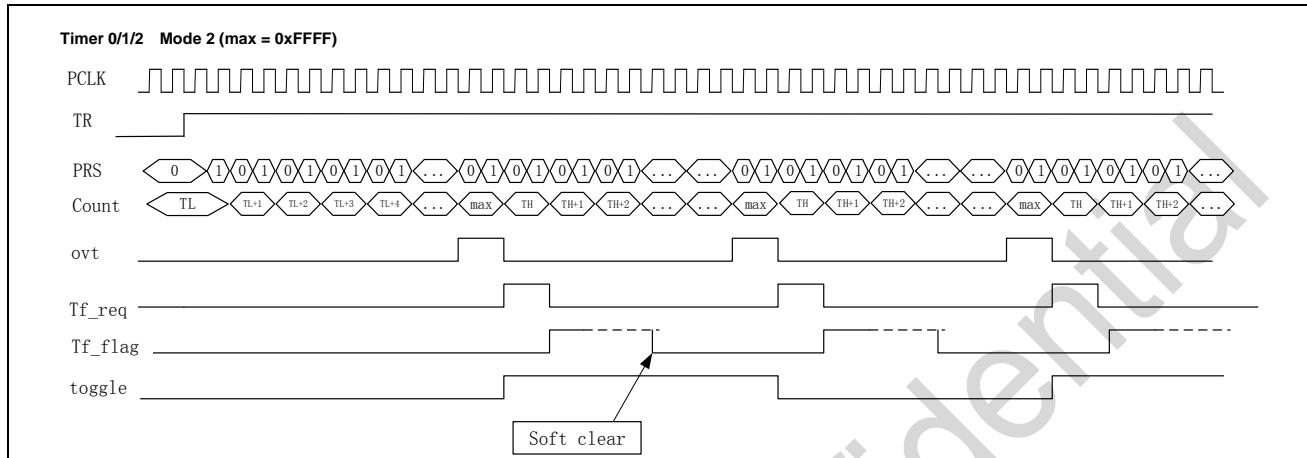


Figure 31. Mode 2 Timing Diagram (Prescaler is set to 2)

10.2.3 Buzzer Function

The Buzzer driving function can be implemented by the reverse output function of the timer. When CR.TOG_EN is set to 1, TOG and TOGN have reverse outputs, whereas when CR.TOG_EN is set to 0, TOG and TOGN outputs are 0. In timer reload mode, the configuration for different buzzer output frequencies at 4M counting clock is listed as follows.

Table 373. Configuration for Different Buzzer Output Frequencies

Buzzer Frequency	Counter Cycle	Counting Value	Counter Reload Value	CNT Initial Value	ARR Reload Value
1000 Hz	0.5 ms	2000	63536	0xF830	0xF830
2000 Hz	0.25 ms	1000	64536	0xFC18	0xFC18
4000 Hz	0.125 ms	500	65036	0xFE0C	0xFE0C

10.3 Base Timer Interconnect

10.3.1 GATE Interconnect

The GATE input can come from a direct port input or the RX signal of the UART. VC input can be configured as a GATE signal as well, which is available for Timer 0/1/2.

By the internal interconnect configuration, it can support the automatic identification of the UART baud rate, the measurement of the pulse width of the VC comparison output and the externally controlled count function.

Inputting from the RX is controlled in the GPIO_CTRL register and inputting from VC controlled in the VC Control Register. Among above 3 gate input selections, port selection, UART input and VC input, only one of them is effective.

10.3.2 Toggle Output Interconnect

TIM0 has reverse output (tog0) to the internal module UART0 controlling the baud rate of UART0. TIM1 has reverse output (tog1) to the internal module UART1 controlling the baud rate of UART1. TIM0/1/2 supports reverse outputs to ports as well, driving Buzzer control.

10.4 Base Timer Register Description

The base address of Base Timer is 0X40000C00.

$x = 0, 1, 2$.

Table 374. Base Timer Register List (1)

Timer	Offset address	Description
TIM0	0x00	TIM0 offset address
TIM1	0x20	TIM1 offset address
TIM2	0x40	TIM2 offset address

Table 375. Base Timer Register List (2)

Register	Offset Address	Description
TIMx_ARR	0X000	TIM0/1/2 reload register
TIMx_CNT	0X004	TIM0/1/2 16-bit mode counting register
TIMx_CNT32	0X008	TIM0/1/2 32-bit mode counting register
TIMx_CR	0X00C	TIM0/1/2 controlling register
TIMx_IFR	0X010	TIM0/1/2 interrupt flag
TIMx_ICLR	0X014	TIM0/1/2 interrupt clearing register

10.4.1 16-bit Mode Counting Register (TIMx_ARR)

Offset address: 0x0000

Reset value: 0x0000 0000

Table 376. 16-bit Mode Counting Register (TIMx_ARR) (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 377. 16-bit Mode Counting Register (TIMx_ARR) (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
R/W															

Table 378. 16-bit Mode Counting Register (TIMx_ARR) (3)

Bit	Flag	Description
31:16	Reserved	Reserved bit, read as 0
15:0	ARR	Timer mode 2 reload register

10.4.2 16-bit Mode Counting Register (TIMx_CNT)

Offset address: 0x004

Reset value: 0x0000 0000

Table 379. 16-bit Mode Counting Register (TIMx_CNT) (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 380. 16-bit Mode Counting Register (TIMx_CNT) (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
R/W															

Table 381. 16-bit Mode Counting Register (TIMx_CNT) (3)

Bit	Flag	Description
31:16	Reserved	Reserved bit, read as 0
15:0	CNT	timer mode 2 counting value register

10.4.3 32-bit Mode Count Register (TIMx_CNT32)

Offset address: 0x008

Reset value: 0x0000 0000

Table 382. 32-bit Mode Count Register (TIMx_CNT32) (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNT32[31:16]															
R/W															

Table 383. 32-bit Mode Count Register (TIMx_CNT32) (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT32[15:0]															
R/W															

Table 384. 32-bit Mode Count Register (TIMx_CNT32) (3)

Bit	Flag	Description
31:0	CNT32	Timer mode 1 count value register

10.4.4 Control Register (TIMx_CR)

Offset address: 0x00C

Reset value: 0x0000 0008

Table 385. Control Register (TIMx_CR) (1)

31:11	10	9	8	7	6:4	3	2	1	0
Reserved	IE	GATE_P	GATE	Reserved	PRS	TOG_EN	CT	MD	TR
-	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W

Table 386. Control Register (TIMx_CR) (2)

Bit	Flag	Description
31:11	Reserved	Reserved bit, read as 0.
10	IE	Interrupt enabling control. Set to 1 to enable interrupt.
9	GATE_P	Port GATE polarity control. The high level is effective by default. If set to 1, it is active low.
8	GATE	Timer gate. 0: no gate, timer starts operation if TR=1 1: timer starts operation if both the port gate is effective and TR=1
7	Reserved	Reserved bit, read as 0
6:4	PRS	TIM prescaler: 000:1, 001:2, 010:4, 011:8, 100:16, 101:32, 110:64, 111:256.
3	TOG_EN	TOG output enabling 0: both TOG and TOGN output 0 1: both TOG and TOGN output reverse-phased signal. Can be used by buzzer.
2	CT	Counter/timer function selection. 0: timer function, the timer is counted by PCLK. 1: counter function, the counter is counted by the falling edge of the external input. The external input is sampled by PCLK and the external input clock frequency is less than 1/2 sample clock.
1	MD	Timer operating mode. 0: mode 1, 32-bit counter/timer 1: mode 2, automatically reloads 16-bit counter/timer
0	TR	Timer operation control. 0: timer stops 1: timer works

10.4.5 Interrupt Flag Register (TIMx_IFR)

Offset address: 0x010

Reset value: 0x0000 0000

Table 387. Interrupt Flag Register (1)

31:8	7	6	5	4	3	2	1	0
								TF

	RO
--	----

Table 388. Interrupt Flag Register (2)

Bit	Flag	Description
31:1	REV	Reserved bit, read as 0,
0	TF	Interrupt flag, it is set by hardware and cleared by writing clearing register.

10.4.6 Interrupt Flag Clearing Register (TIMx_ICLR)

Offset address: 0x014

Reset value: 0x0000 0001

Table 389. Interrupt Flag Clearing Register (1)

31:8	7	6	5	4	3	2	1	0
								TFC
								WO

Table 390. Interrupt Flag Clearing Register (2)

Bit	Flag	Description
31:1	Reserved	Reserved bit, read as 0
0	TFC	Interrupt clearing flag. Setting to 0 for clearing. It is invalid if set to 1

11 Low Power Timer (LPTimer)

11.1 LPTimer Introduction

The LPTimer is an asynchronous 16-bit timer/counter that can perform clocking/counting by internal low speed RC or external low speed crystal oscillator even the system clock is off. It supports waking up the system in low power mode by interrupt.

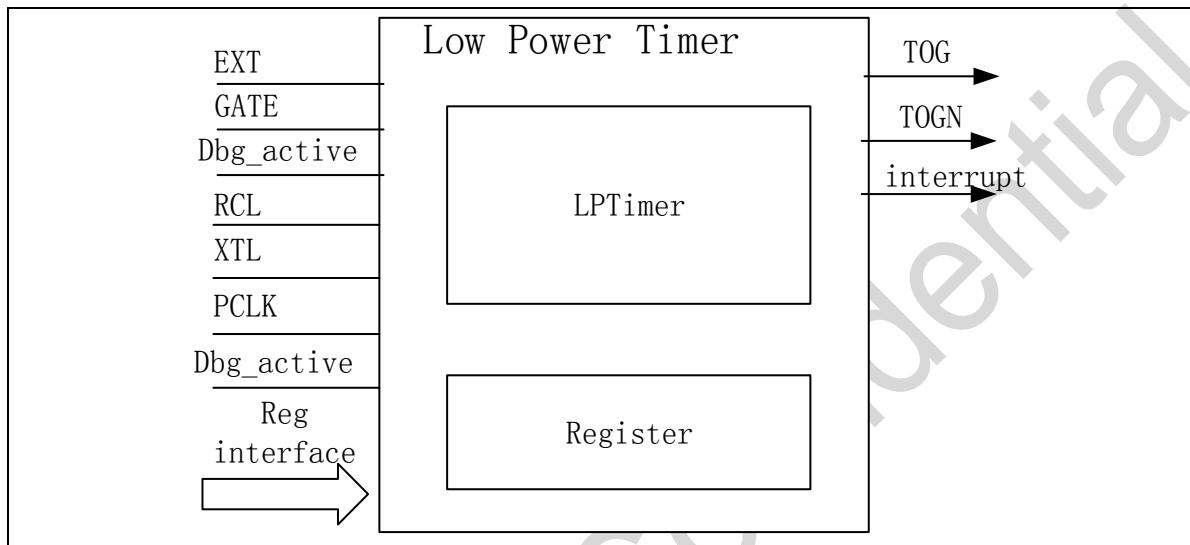


Figure 32. LPTimer Block Diagram

11.2 LPTimer Function Description

LPTimer supports two operational modes with each timer/counter having the independent startup controlling signal, external input clock and gate signal.

LPTimer uses EXT and GATE performing counting function. EXT is used for the counter's external input clock signal and GATE for the effective level counting enabling signal.

The LPTimer's timer supports two operational modes, mode 1 - 16-bit free counting mode and mode 2 - 16-bit reload mode, which can be selected by setting the MD in the control register (CR).

When LPTimer starts, it automatically loads the value of the reload register ARR into the counter.

LPTimer supports selecting 1 of the 3 clock options as the timer clock, which is controlled by the control register CR.TCK_SEL with PCLK as default. The clock selection is as follows.

Table 391. Clock Selection

TCK_SEL	00	01	10	11
Timer Clock	PCLK	PCLK	LXI	LRC
Read Timer Counting Value	Read with synchronization	No synchronization	Read with synchronization	Read with synchronization

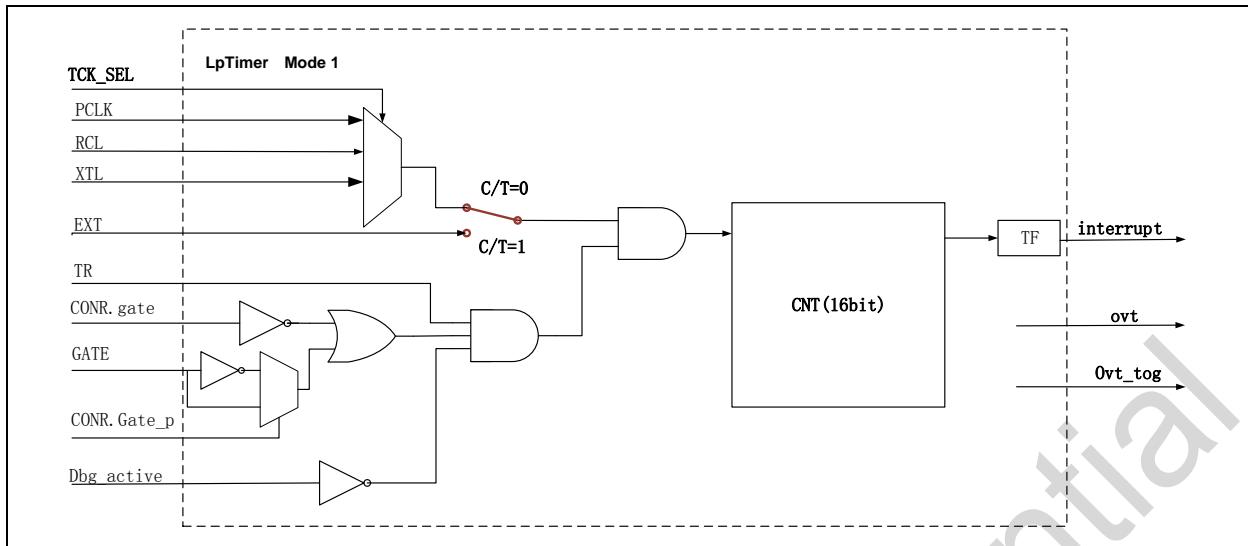


Figure 33. LPTimer Mode 1

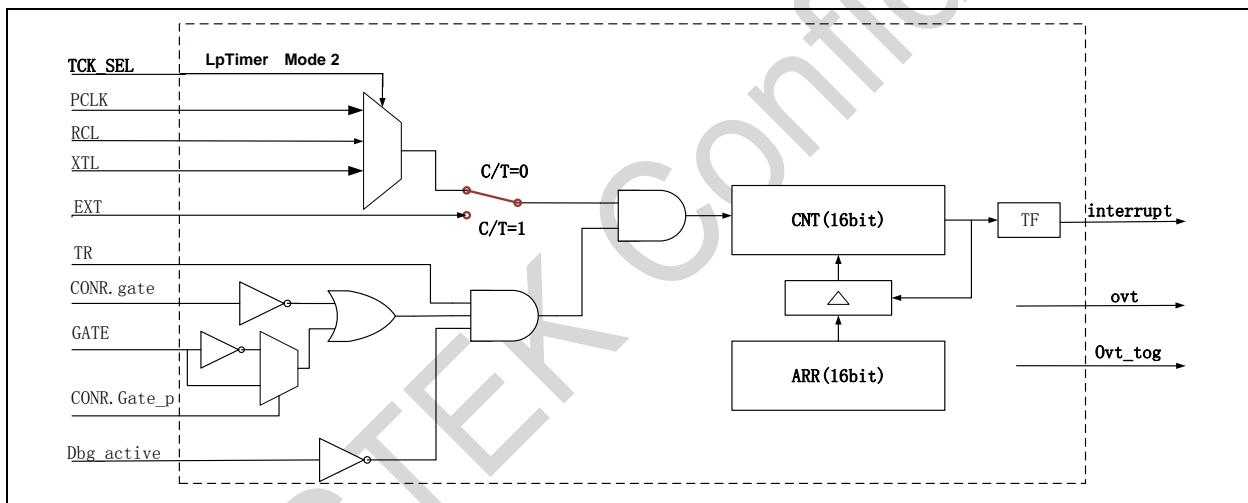


Figure 34. LPTimer Mode 2

The timer runs when the corresponding timer TR is set to 1. The counter starts counting from the set value, and an overflow interrupt is generated after counting to the maximum value, 0xFFFF. After the interrupt, in mode 1, the counter is cleared to continue counting up whereas it is reloaded with the value of the ARR register and continues counting up in mode 2. The initial value of the counter CNT is automatically loaded by the ARR during the startup of the timer.

Since the LPTimer is an asynchronous timer, the timer interrupt is synchronized from the timer clock domain to the pclk domain. When interrupt function is enabled for reload mode timer, interrupt will be lost if the reload mode timer value is set greater than 0Xffffb. Therefore It is recommended to set the value of the reload register less than 0XFFFC. There is no such restriction if interrupt function is disabled.

11.2.1 Timing Function

The timing function is used to generate periodic timing. In the timing function, the timer is increased by one on the falling edge of input clock. It overflows when counting to the maximum with interrupt generation.

11.2.2 Count function

The count function is used to measure the number of times an event has occurred. In the count function, the counter is increased once on the falling edge of each corresponding input clock. The input signal is sampled by the internal count clock, so the external input clock frequency cannot exceed the system's count clock. When counting to the maximum, it overflows with interrupt generation.

11.3 LPTimer Interconnect

11.3.1 GATE Interconnect

The GATE input can come from a direct port input or the RX signal of the UART. VC input can be configured as a GATE signal as well.

By the internal interconnect configuration, the GATE of TIM3 can fulfill the automatic identification of the UART baud rate, measurement of the pulse width of the VC comparison output and the externally controlled count.

The UART selection control register is in the port control register GPIO_CTRL4, and the VC output control register is in the VC control module.

11.3.2 EXT Interconnect

The EXT input can come from a direct port input or VC's output as configured.

By the internal interconnect configuration, the EXT of TIM3 can fulfill the measurement of the VC pulse count. The VC output control register is in VC control module.

11.3.3 Toggle Output Interconnect

The LPTimer's reverse outputs to ports can drive buzzer control.

11.4 LPTimer Register Description

Base address: 0X40000C00

Table 392. LPTimer Register List

Register	Offset Address	Description
CNT	0X060	LPTimer count value read-only register
ARR	0X064	LPTimer reload register
CR	0X06C	LPTimer control register
IFR	0X070	LPTimer interrupt flag
ICLR	0X074	LPTimer interrupt clearing register

11.4.1 Counter Value Register (TIM3_CNT)

Offset address: 0x060

Reset value: 0x0000 0000

Table 393. Counter Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 394. Counter Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
RO															

Table 395. Counter Value Register (3)

Bit	Flag	Description
31:16	Reserved	Reserved bit, read as 0
15:0	CNT	Low power timer count value read-only register

11.4.2 Reload Register (TIM3_ARR)

Offset address: 0x064

Reset value: 0x0000 0000

Table 396. Reload Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 397. Reload Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
R/W															

Table 398. Reload Register (3)

Bit	Flag	Description
31:16	Reserved	Reserved bit, read as 0
15:0	ARR	Low power timer reload register. Need read CR.WT_FLAG before write ARR. Only when WT_FLAG is 1, the data can be written during the write operation. WT2_FLAG becomes low after writing the ARR register.

11.4.3 Control Register (TIM3_CR)

Offset address: 0x06C

Reset value: 0x0000 0008

Table 399. Control Register (1)

31:11	10	9	8	7	6	5:4	3	2	1	0
Reserved	IE	GATE_P	GATE	WT_FLAG	Reserved	TCK_SEL	TOG_EN	CT	MD	TR
-	R/W	R/W	R/W	R	-	R/W	R/W	R/W	R/W	R/W

Table 400. Control Register (2)

Bit	Flag	Description
31:11	Reserved	Reserved bit, read as 0
10	IE	Interrupt enabling control. Interrupt is enabled when set as 1.
9	GATE_P	GATE polarity control. High gate active by default. Low gate active when it's set to.
8	GATE	Timer gating 0: no gating, timer works when TR=1; 1: timer works only work if the gating is 1 and TR=1.
7	WT_FLAG	WT, write synchronization flag 0: synchronizing, writing ARR is invalid at this time 1: synchronization completes, and ARR can be changed at this time.
6	Reserved	Reserved bit, read as 0
5:4	TCK_SEL	LPTimer clock selection, 00: PCLK, 10: XIL, 11: RCL
3	TOG_EN	TOG output enabling 0: TOG and TOGN output 0 at the same time 1: TOG and TOGN output signals with opposite phases, can be used by buzzer.
2	CT	Counter/timer function selection 0: timer function, the timer counts using the clock selected by TCK_SEL. 1: counter function, the counter counts using the falling edge of the external input. The sampling clock uses the clock selected by TCK_SEL and the external input clock should be lower than a half of the sampling clock.
1	MD	Timer Mode 0: mode 1, 16-bit counter/timer without reload 1: mode 2, 16-bit counter/timer with automatic reload
0	TR	Timer control bit 0: timer stops 1: timer runs

11.4.4 Interrupt Flag Register (TIM3_IFR)

Offset address: 0x070

Reset value: 0x0000 0000

Table 401. Interrupt Flag Register (1)

31:10	9	8	7	6	5	4	3	2	1	0
Reserved										TF
-										RO

Table 402. Interrupt Flag Register (2)

Bit	Flag	Description
31:1	Reserved	Reserved bit, read as 0
0	TF	Interrupt flag, set by hardware. Write clearing register is cleared.

11.4.5 Interrupt Flag Clearing Register (TIM3_ICLR)

Offset address: 0x074

Reset value: 0x0000 0001

Table 403. Interrupt Flag Clearing Register (1)

31:10	9	8	7	6	5	4	3	2	1	0
Reserved										TFC
-										WO

Table 404. Interrupt Flag Clearing Register (2)

Bit	Flag	Description
31:1	Reserved	Reserved bit, read as 0
0	TFC	Interrupt clearing flag. It is cleared when set to 0. It is invalid when set to 1.

12 Programmable Counting Array (PCA)

12.1 PCA Introduction

The PCA (programmable counter array) supports up to five 16-bit capture/compare modules. This timer/counter can be used to perform capture/compare function for a general purpose clock count/event counter. Each module of the PCA can be independently programmed to provide input capture, output compare, or pulse width modulation. In addition, module 4 has an additional watchdog timer mode.

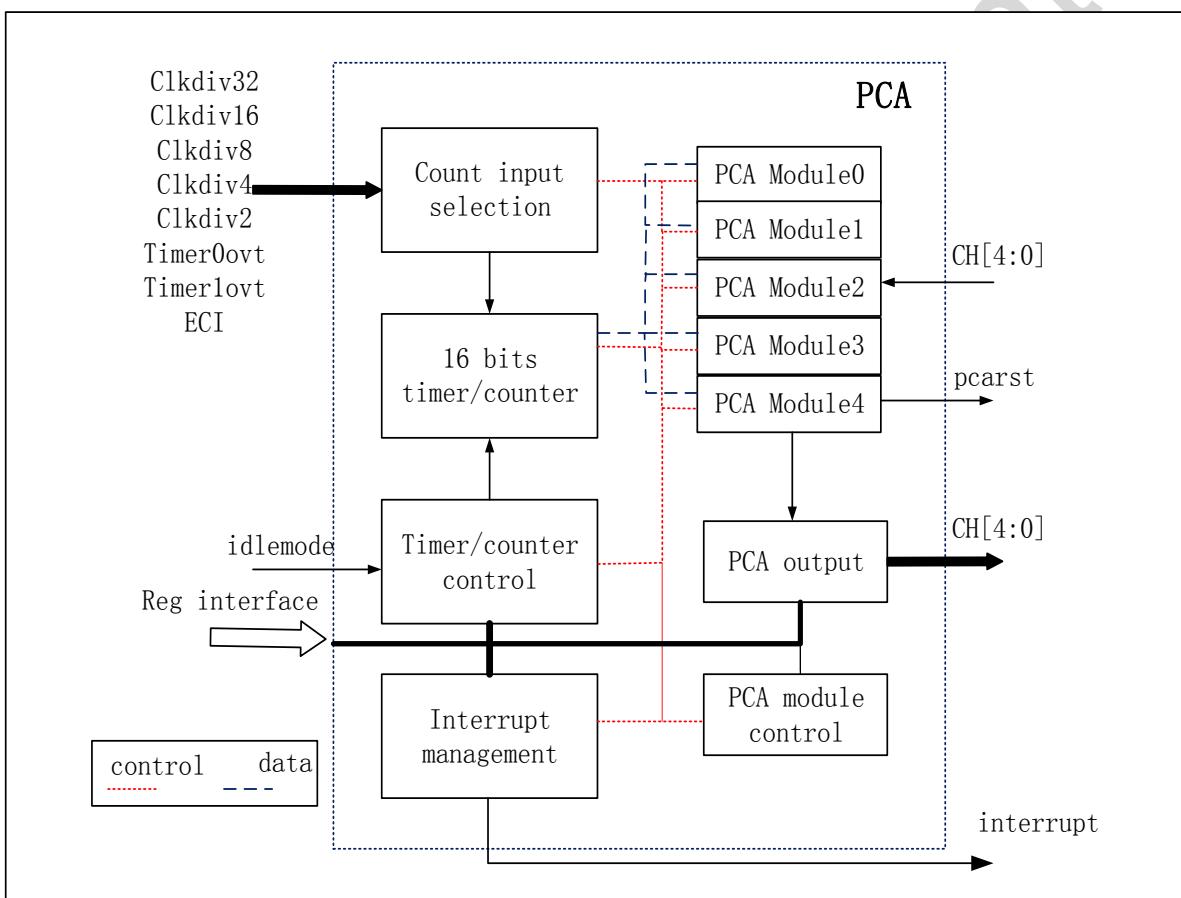


Figure 35. Overall Block Diagram of PCA

12.2 PCA Function Description

Each module can be configured to operate independently with 3 modes: edge triggered capture, output compare, and 8-bit pulse width modulation. Each module has its own function registers in the system controller. These registers are used to configure how the module works and exchanges data.

Each set of compare/capture module consists of a compare/capture register set (CCAPx), a 16-bit comparator, and various logic gates. The register set is used to store time or number information, capture conditions based on external triggering and compare conditions based on internal triggering. In PWM mode, the register (CCAPxL) is used to control the duty cycle of the output waveform.

Each module can be independently programmed to operate in any of the following modes,

- **16-bit capture mode:** triggered at rising edge, falling edge or arbitrary edge.
- **Compare mode:** for 16-bit software timer, 16-bit high speed output, 16-bit watchdog timer (module 4) or 8-bit pulse width modulation
- **Disabled mode**

The corresponding operating mode is determined by the compare/capture module mode register (CCAPMx). All compare/capture modules share the same time base for count when being programmed. It controls the on/off state of the timer/counter through the CCON.CR bit. When a compare/capture module operating in capture mode, software timer mode or high speed output mode, if a matching occurs, it sets compare/capture flag (CCON.CCFx), with a PCA interrupt request generated if the corresponding interrupt enabling flag is set in the CCAPMx register . The CPU can read and write CCAPx registers at any time.

12.2.1 PCA Timer/Counter

This set of special function registers for CNT can be used as a 16-bit timer/counter. This is a 16-bit counter which counts up. If the CMOD.CFIE bit is set to 1, in case of CNT overflow, the hardware automatically sets the PCA overflow flag (CCON.CF) and generates a PCA interrupt request. The 3 bits, CMOD.CPS[2:0], select 8 signals as input of the timer/counter.

- The system clock PCLK divided by 32.
- The system clock PCLK divided by 16.
- The system clock PCLK divided by 8.
- The system clock PCLK divided by 4.
- The system clock PCLK divided by 2.
- Timer 0 overflow, the CNT is increased each time the Timer 0 count overflows, providing a variable programming frequency input to the PCA.
- Timer 1 overflow, the CNT is increased each time the Timer 1 count overflows, providing a variable programming frequency input to the PCA.
- ECI, the CPU samples the PCA ECI every 4 PCLK clock cycles. When each sample goes from high to low, CL automatically increases by 1, so the highest ECI input frequency cannot be higher than 1/8 of the system clock PCLK to meet the sampling requirements.

The controller (CCON.CR) starts the PCA timer/counter. When CMOD.CIDL is set to 1, the PCA timer/counter can continue to run in Idle mode. CPU can read CNT at any time, however, when the count starts (CCON.CR=1), CNT is prohibited from writing to prevent counting errors.

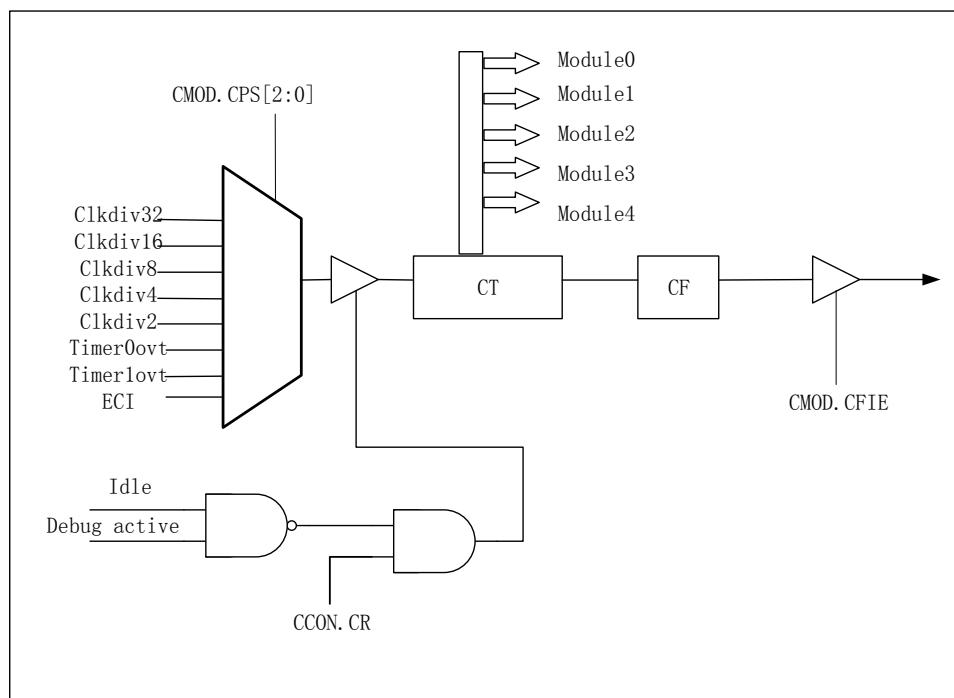


Figure 36. PCA Counter Block Diagram

12.2.2 PCA Capture Function

The PCA capture mode provides 5 PCA channels for measuring pulse period, pulse width, duty cycle and phase difference.

A level transition on the pin triggers PCA capturing the value of the PCA counter/timer and loading it into the corresponding module's 16-bit capture/compare register (CCAPx). The CCAPMx.CAPP and CCAPMx.CAPN bits are used to select the type of level change that triggers the capture, including low to high (positive edge), high to low (negative edge), or any change (positive or negative). When a capture occurs, the capture/compare flag (CCFn) in CCON is set to logic 1 and an interrupt request is generated (if the CCF interrupt is enabled). The CCFn bit is not cleared by hardware when CPU goes to the interrupt service routine. Therefore it needs user software to clear it via writing the INTCL register instead. If the CCPMx.CAPP and CCAPMx.CAPN bits are both set to logic 1, by reading the state of the corresponding port pin, it can determine whether by a rising edge or a falling edge this capture is triggered. The measurement resolution is equal to the timer/counter clock. The input signal must last for at least 2 clock cycles during a high or low period to ensure that the input signal can be recognized by the hardware. CPU can read or write to the CCAPx registers at any time.

The capture settings are as follows.

- When it is required to capture on the external rising edge, set CCPMx.CAPP to 1 and CCAPMx.CAPN to 0.
- When it is required to capture on the external falling edge, set CCPMx.CAPP to 0 and CCAPMx.CAPN to 1.
- When it is required to capture on the external rising and falling edge, set CCPMx.CAPP to 1 and CCAPMx.CAPN to 1.

Notes:

1. In a certain module, the values captured later will overwrite the currently captured values. The captured value is saved in RAM in the interrupt service routine to prevent overwriting. This operation must be completed before the next event occurs, otherwise the previous capture sampling value will be lost.

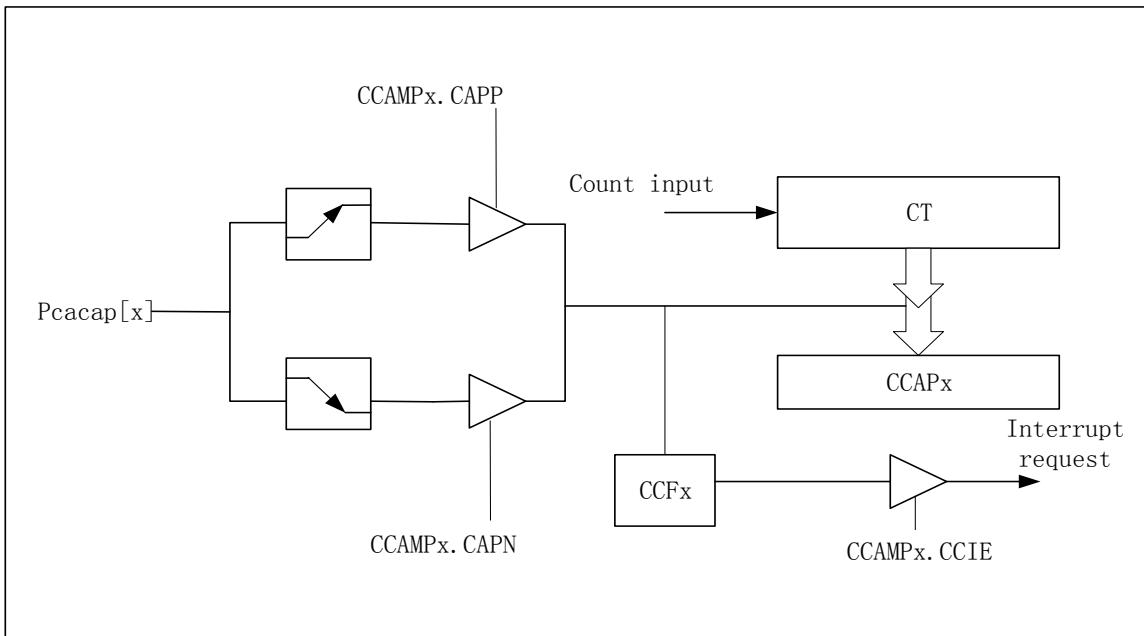


Figure 37. PCA Capture Function Block Diagram

12.2.3 PCA Compare Function

The compare function supports 5 modules, providing the functions of timer, event counter, and pulse width modulation. It supports 4 compare function modes including 16-bit software timer mode, high speed output mode, WDT mode and PWM mode. In the first 3 ones, the compare/capture module compares the value of the 16-bit PCA timer/counter with the 16-bit value preloaded into the module's CCAPx register. In PWM mode, the PCA module continuously compares the PCA timer/counter low byte register (CNT) with a value of 8 bits in the CCAPxL module register. It compares once every 4 clock cycles, meaning that it matches the clock rate of the fastest PCA timer/counter.

Set the CCAPMx.ECOM bit to select the compare function of the module.

To use the module in compare mode correctly, please follow the general procedure below.

1. Select the operating mode of the PCA module.
2. Select the input signal of the PCA timer/counter.
3. Load the compare value into the module's compare/capture register pair.
4. Set the PCA timer/counter operation control bit.
5. Generate interrupt after the match and clear the module's compare/capture flag.

- **16-bit software counting mode**

To set a 16-bit software timer mode for the compare/capture module, the CCAPMx.ECOM and CCAPMx.MAT bits need to be set. Once a match occurs between the PCA timer/counter and the compare/capture register (CCAPx), it sets the module's compare/capture flag (CCON.CCFx). An interrupt request is generated if the corresponding interrupt enabling bit (CCAPMx.CCIE) is set. Users must clear the compare/capture flag by software since the hardware does not clear it automatically. A new 16-bit compare value can be written to the compare/capture register (CCAPx) in the interrupt service routine.

Notes:

1. To prevent invalid matches, when updating these registers, user software should write CCAPxL first and then CCAPxH. Once the CCAPxL is written, it will clear the ECOMx bit, which disables the compare function. Writing CCAPxH will set the ECOMx bit as well, which enables compare function again, meaning that, when writing a 16-bit value to the

capture/compare register of PCA0, the low byte should be written first.

- **High speed output mode**

In high speed output mode, the logic level on the CAP/CMP[x] pin of module PCA changes once the value in the PCA counter matches the module's 16-bit capture/compare register (CCAPx). It can provide higher precision than the method of switching IO output, as this high speed output is not responded by interrupt, which avoids potential affection of the output frequency. Moreover, the method of switching IO output by CPU is not ideal both in power consumption and precision.

Set the CCAPMx.ECOM, CCAPMx.MAT and CCAPMx.TOG bits to setup the high speed output mode of the compare/capture module. The match between the PCA timer/counter and the compare/capture register (CCAPx) switches the PCA's CAP/CMP[x] signal and sets the module's compare/capture flag (CCON.CCFx). Users can choose to match the switching signal from low to high or high to low by setting or clearing the PCA's CAP/CMP[x] signal.

Users can also choose to generate an interrupt request by setting the corresponding interrupt enabling bit (CCAPMx.CCIE) for interrupt generation when a match occurs. The flag must be cleared in users' software as the hardware does not clear the compare/capture interrupt flag. If users do not change the compare/capture register in the interrupt routine, the PCA recounts the compare value and the next reverse output will occur if it matches. In the interrupt service routine, a new 16-bit compare value can be written to the compare/capture register (CCAPx).

Notes:

1. To prevent invalid matches, user software should write CCAPxL first followed by CCAPxH when updating these registers. Writing CCAPxL will clear the ECOM bit to clear the function disabling, then writing CCAPxH will set the ECOM bit to re-enable the compare function.

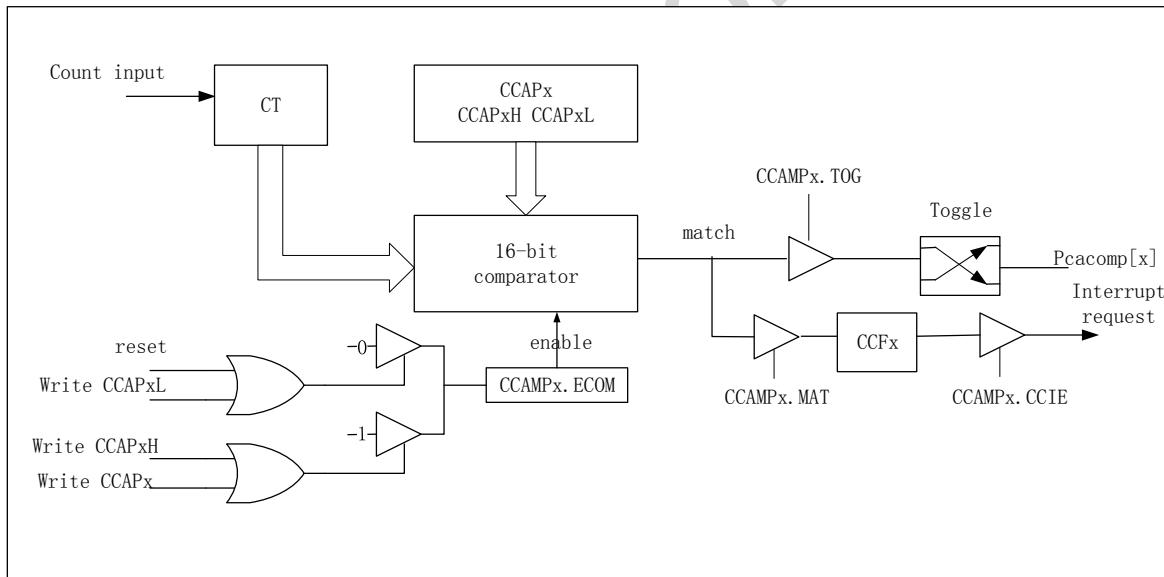


Figure 38. PCA Compare Functional Block Diagram

- **WDT function of PCA module 4**

In addition to the WDT hardware module, PCA module 4 provides a 16-bit WDT with programmable frequency as well. This mode produces a reset signal when the count of the PCA timer/counter matches the value stored in the compare/capture register (CCAP4) of the module 4. The WDT reset signal of PCA is an independent reset signal besides external reset (RST), hardware watchdog reset (WDTRST), low voltage reset (LVD), power-on (POR) and power-down reset. Users can either combine them freely or use alone. Module 4 is the only one which has WDT mode among the PCA modules. When it is not set to WDT, it can be used in other modes independently.

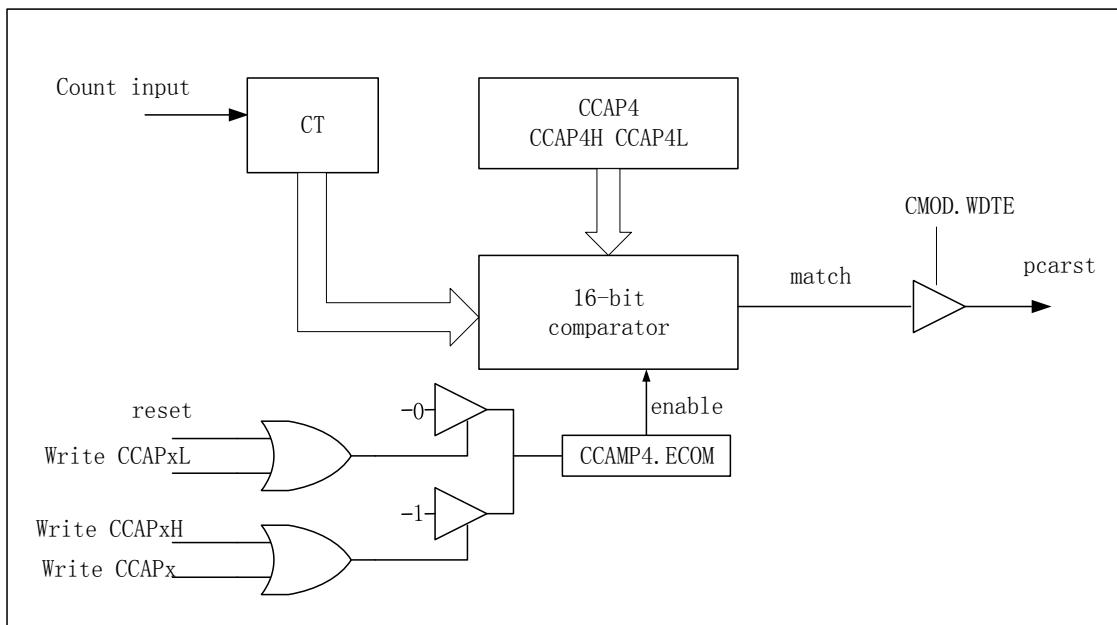


Figure 39. PCA WDT Functional Block Diagram

When PCA module 4 is used as WDT, CCAPM4.ECOM4, CCAPM4.MAT4 and CMOD.WDTE must be set. In addition, the PCA timer/counter can set CMOD.CPS to select different input counting frequencies.

In the compare/capture register (CCAP4), input a 16-bit comparison value, which can be a 16-bit initial value or the reset value (0x0000) in a PCA timer/counter (CNT). The WDT reset time (counter running time till the matching occurs) is calculated by the time difference between the comparison value and the initial/reset value as well as the pulse rate of the PCA input. Set the timer/counter running control bit (CCON.CR) to start PCA WDT. At each match, the WDT of PCA generates a reset signal. There're 3 options available for users to prevent a PCA WDT reset.

1. Change CCAP4 comparison value periodically, so a matching never occurs.
2. Change the PCA timer/counter value (CNT) periodically, so a matching never occurs.
3. Disable the module reset output signal by clearing the CMOD.WDTE bit before the matching, and enable it later then.

The first two options are more robust than the third one since WDT matching is not blocked in the third option.

The second option is not recommended if all other PCA modules are in use since the 5 modules share a common time base (using the same counter). In short, the first option is the best in most applications.

The PCA WDT configuration is as flows.

1. Configure WDT compare/capture register PCA_CCAP4.
2. Configure PCA count register PCA_CNT.
3. Configure PCA_CCAMP4 to select the matching function.
4. Configure PCA_CMOD to select input clock for WDT function enabling.
5. Start PCA.
6. Select PCA WDT clearing method and clear PCA WDT before PCA WDT reset.

- **PCA 8-bit Pulse Width Modulation Function**

Pulse width modulation (PWM) is a technique used to control the duty cycle, cycle and phase of the waveform. Each of the 5 PCA modules can be used independently to generate pulse width modulation (PWM) outputs at the CAP/CMP [x] pins of the corresponding PCA with a 8-bit resolution pulse width. The frequency of the PWM output depends on the time base of the PCA counter/timer. Use the module's capture/compare register CCAPxL to change the duty cycle of the PWM output signal. When the low byte (CL) of the PCA counter/timer is equal to the value in the CAPxL, the output on the CAP/CMP [x] pin of the PCA is set to 1. When the count value in the CL overflows, the CAP/CMP [x] output of the PCA is reset to 0. When the counter/timer low byte CL overflows (from 0xFF to 0x00), the values stored in CCAPxH are automatically loaded into CCAPxL with no software intervention required.

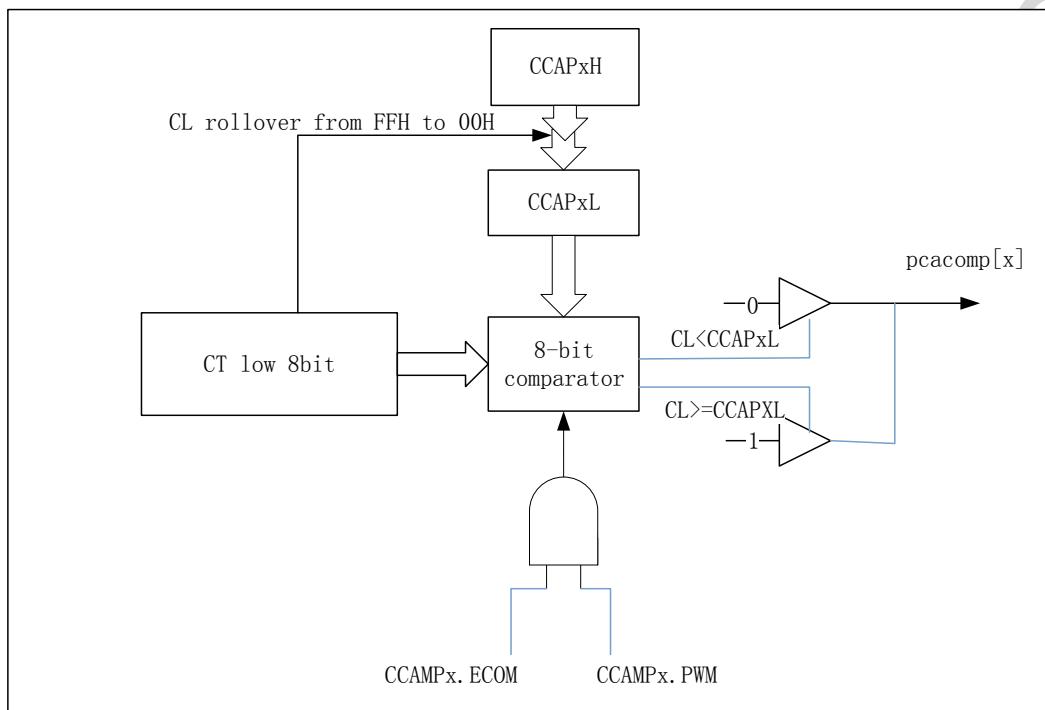


Figure 40. PCA PWM Functional Block Diagram

In this mode, the value in the low bytes of the PCA timer/counter (CL) is continuously compared to the value in the low byte compare/capture register (CCAPxL). When CL is less than CCAPxL, the output waveform is low. When the two match ($CL = CCAPxL$), the output waveform goes to high and remains (remains high till the end) until the CL overflows from FFH to 00H. When overflow occurs, the value of CCAPxH is automatically loaded into CCAPxL and a new cycle starts.

The duty cycle of the current waveform depends on the value of CCAPxL. The duty cycle of the next waveform depends on the value of CCAPxH. The pulse width modulation can be changed through changing the value in CCAPxL. As shown in the below figure, the 8-bit value in CCAPxL can range from 0 (100% duty cycle) to 255 (0.4% duty cycle). To change the CCAPxL value without glitch, a new value needs to be written to the high byte register (CCAPxH). When CL exceeds 0xFF and scrolls to 0x00, this value is automatically loaded to CCAPxL by hardware.

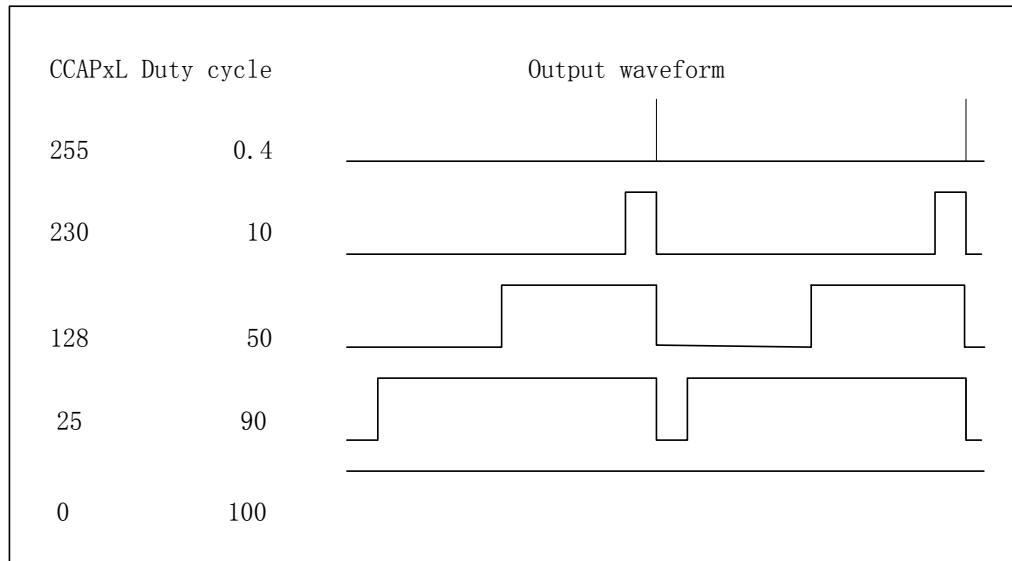


Figure 41. PCA PWM Output Waveform

To set the compare/capture module operating in PWM mode, it needs to set CCAPMx.ECOM and CCAPMx bits. In addition, the PCA timer/counter can select input count signal frequency via programming CMOD.CSP[2:0]. Input an 8-bit value in CCAPxL to specify the duty cycle of the first PWM waveform, and input an 8-bit value in CCAPxH to specify the duty cycle of the second PWM waveform. Set the control bit (CCON.CR) of the timer/counter to start the PCA timer/counter.

12.3 Interconnects and Controls between PCA Module and Other Modules

12.3.1 ECI Interconnects

The ECI input comes from either different input ports selected via IO MUX externally or the filtered comparison output of the internal VC. The VC output control register is in the VC control module.

12.3.2 PCACAP0

Channel 0's capture input can be the followings.

- External IO MUX input port.
- The MUX input of the external UART's RX.
- Internal VC1's filtered output after comparison.

UART selection and control register is in the port control register - GPIO_CTRL2 and VC output control register is in the VC control module.

12.3.3 PCACAP1

Channel 1's capture input can be the followings.

- External IO MUX input port
- The MUX input of the external UART's RX

- Internal VC2's filtered output after comparison

UART selection and control register is in the port control register - GPIO_CTRL2 and VC output control register is in the VC control module.

12.3.4 PCACAP [4:2]

Channel 2, 3, 4's capture input can be the followings.

- External IO MUX input port
- The MUX input of the external UART's RX

UART selection and control register is in the port control register - GPIO_CTRL2.

12.4 PCA Register Description

Base address: 0X40001000

Table 12-1. PCA Register List

Register	Offset Address	Description
CCON	0X000	PCA control register
CMOD	0X004	PCA mode register
CNT	0X008	PCA count register
INT_CL	0X00C	PCA interrupt clearing register
CCAPM0	0x010	PCA compare/capture module 0's mode register
CCAPM1	0x014	PCA compare/capture module1's mode register
CCAPM2	0x018	PCA compare/capture module2's mode register
CCAPM3	0x01C	PCA compare/capture module3's mode register
CCAPM4	0x020	PCA compare/capture module 4's mode register
CCAP0H	0X024	PCA compare/capture module's 0 high 8-bit register
CCAP0L	0X028	PCA compare/capture module's 0 low 8-bit register
CCAP1H	0X02C	PCA compare/capture module 1's high 8-bit register
CCAP1L	0X030	PCA compare/capture module 1's low 8-bit register
CCAP2H	0X034	PCA compare/capture module 2's high 8-bit register
CCAP2L	0X038	PCA compare/capture module 2's low 8-bit register
CCAP3H	0X03C	PCA compare/capture module 3's high 8-bit register
CCAP3L	0X040	PCA compare/capture module 3's low 8-bit register
CCAP4H	0X044	PCA compare/capture module 4's high 8-bit register
CCAP4L	0X048	PCA compare/capture module 4's low 8-bit register
CCAPO	0X04C	PCA PWM and high speed output flag register
CCAP0	0X050	PCA compare/capture module 0's 16-bit register

Register	Offset Address	Description
CCAP1	0X054	PCA compare/capture module 1's 16-bit register
CCAP2	0X058	PCA compare/capture module 2's 16-bit register
CCAP3	0X05C	PCA compare/capture module 3's 16-bit register
CCAP4	0X060	PCA compare/capture module 4's 16-bit register

12.4.1 Control Register (PCA_CCON)

Offset address: 0x000

Reset value: 0x0000 0000

Table 405. Control Register (1)

31:8	7	6	5	4	31	2	1	0
Reserved	CF	CR	Reserved	CCF4	CCF3	CCF2	CCF1	CCF0
-	RO	R/W	-	RO	RO	RO	RO	RO

Table 406. Control Register (2)

Bit	Flag	Description
31:8	Reserved	Reserved bit
7	CF	PCA counter overflow flag (writing is invalid). When the PCA count overflows, the CF is set by the hardware, and if the CFIE bit of the CMOD register is 1, the CF flag can generate interrupt. 1: counter overflow 0: no overflow
6	CR	PCA counter control bit. 1: enable the PCA counter count. 0: disable the PCA counter count.
5	Reserved	Reserved bit
4	CCF4	PCA counter module 4 compare/capture flag bit. When matching or capturing occurs, the bit is set by hardware (writing is invalid) When CCAPM4.CCIE is set, it generates PCA interrupt on the flag.
3	CCF3	PCA counter module 3 compare/capture flag bit. When matching or capturing occurs, the bit is set by hardware (writing is invalid) When CCAPM3.CCIE is set, it generates PCA interrupt on the flag.
2	CCF2	PCA counter module 2 compare/capture flag bit When matching or capturing occurs, the bit is set by hardware (writing is invalid) When CCAPM2.CCIE is set, it generates PCA interrupt on the flag.
1	CCF1	PCA counter module 1 compare/capture flag bit When matching or capturing occurs, the bit is set by hardware (writing is invalid) When CCAPM1.CCIE is set, it generates PCA interrupt on the flag.
0	CCF0	PCA counter module 0 compare/capture flag bit

		When matching or capturing occurs, the bit is set by hardware (writing is invalid) When CCAPM0.CCIE is set, it generates PCA interrupt on the flag.
--	--	--------------------------------------------------------------------------------------------------------------------------------------------------------

CMOSTEK Confidential

12.4.2 Mode Register (PCA_CMOD)

Offset address: 0x004

Reset value: 0x0000 0000

Table 407. Mode Register (1)

31:8	7	6	5	4	3	2	1	0
Reserved	CIDL	WDTE	Reserved		CPS[2:0]		CFIE	
-	R/W	R/W	-		R/W		R/W	

Table 408. Mode Register (2)

Bit	Flag	Description
31:8	Reserved	Reserved bit
7	CIDL	Whether PCA stops running in sleep mode. 1: PCA stops running in sleep mode. 0: PCA is running in sleep mode.
6	WDTE	PCA WDT function enabling control bit. 1: enable the WDT function of the PCA module 0: disable the WDT function of the PCA module
5:4	Reserved	Reserved bit
3:1	CPS	Clock frequency dividing selection and clock source selection. 000: PCLK/32 001: PCLK/16 010: PCLK/8 011: PCLK/4 100: PCLK/2 101: Timer0 overflow 110: Timer1 overflow 111: ECI external clock and the frequency-quadrant sampling of PCLK clock.
0	CFIE	PCA counter interrupt enabling control signal. 1: enable interrupt 0: disable interrupt

12.4.3 Count Register (PCA_CNT)

Offset address: 0x008

Reset value: 0x0000 0000

Table 409. Count Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 410. Count Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT															
R/W															

Table 411. Count Register (3)

Bit	Flag	Description
31:16	Reserved	Reserved bit
15:0	CNT	Timer/counter value. Writing is valid only when PCA stops running, otherwise writing is invalid.

12.4.4 Interrupt Clearing Register (PCA_ICLR)

Offset address: 0x000C

Reset value: 0x0000 009Fh

Table 412. Interrupt Clearing Register (1)

31:8	7	6	5	4	3	2	1	0
Reserved	CF	Reserved		CCF4	CCF3	CCF2	CCF1	CCF0
-	WO	-		WO	WO	WO	WO	WO

Table 413. Interrupt Clearing Register (2)

Bit	Flag	Description
31:8	Reserved	Reserved bit
7	CF	PCA counter overflow flag clearing. Set to 0 by software for clearing. Setting to 1 is invalid. Reading value is 1.
6:5	Res.	Reserved bit
4	CCF4	PCA counter module 4 compare/capture flag clearing. Set to 0 by software for clearing. Setting to 1 is invalid. Reading value is 1.
3	CCF3	PCA counter module 3 compare/capture flag clearing. Set to 0 by software for clearing. Setting to 1 is invalid. Reading value is 1.
2	CCF2	PCA counter module 2 compare/capture flag clearing. Set to 0 by software for clearing. Setting to 1 is invalid. Reading value is 1.
1	CCF1	PCA counter module 1 compare/capture flag clearing. Set to 0 by software for clearing. Setting to 1 is invalid. Reading value is 1.

0	CCF0	PCA counter module 0 compare/capture flag clearing. Set to 0 by software for clearing. Setting to 1 is invalid. Reading value is 1.
---	------	-------------------------------------------------------------------------------------------------------------------------------------

12.4.5 Compare/capture Mode Register (PCA_CCAPM0~4)

Offset address:

CCAPM0: 0x010 CCAPM1: 0x014 CCAPM2: 0x018

CCAPM3: 0x01C CCAPM4: 0x020

Reset value: 0x0000 0000

Table 414. Compare/capture Mode Register (1)

31:8	7	6	5	4	3	2	1	0
Reserved	ECOM	CAPP	CAPN	MAT	TOG	PWM	CCIE	
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 415. Compare/capture Mode Register (2)

Bit	Flag	Description
31:7	Reserved	Reserved bit
6	ECOM	Comparator function control bits. 1: enable comparator function 0: disable comparator function It needs to set ECOM bit when PCA is used as software counter, high speed output, PWM mode or WDT mode. It sets ECOM bit automatically when writing to the CCAMPPh or CCAMPx registers. It clears the ECOM bit automatically when writing to the CCAMPLx register.
5	CAPP	Rising edge capture control bit. 1: enable rising edge capture 0: disable rising edge capture
4	CAPN	Falling edge capture control bit 1: enable falling edge capture; 0: disable falling edge capture
3	MAT	Matching function control bits. 1: when the PCA count value matches the module's compare/capture register value, the interrupt flag CCFx (x=0-4) in the CCON register is set. 0: matching function is disabled
2	TOG	Toggle control bit. 1: operate in PCA high speed output mode, the value of PCA counter matches the value of the module's compare/capture register, CCPx pin has reverse output. 0: disable reverse output function
1	PWM	Pulse width modulation (PWM) control bit 1: enable CCPx pin as PWM output 0: disable PWM PWM function is valid only when CCAPMx[6:0] = 100_0010

0	CCIE	PCA interrupt enabling. 1: enable compare/capture interrupt 0: disable PCA compare/capture interrupt
---	------	------------------------------------------------------------------------------------------------------------

12.4.6 Higher 8-bit of Compare/Capture Data Register (PCA_CCAP0~4H)

Offset address

CCAP0H: 0x024 CCAP1H: 0x02C CCAP2H: 0x034

CCAP3H: 0x03C CCAP4H: 0x044

Reset value: 0x0000 0000

Table 416. Higher 8-bit of Compare/Capture Data Register (1)

31:8	7	6	5	4	3	2	1	0
Reserved	CCAPx[15:8]							
-	R/W							

Table 417. Higher 8-bit of Compare/Capture Data Register (2)

Bit	Flag	Description
31:8	Reserved	Reserved bit
7:0	CCAPx[15:8]	Higher 8-bit register in compare/capture mode. When the PCA mode is used for compare/capture, it is to store the high 8 bits of the 16-bit capture count value. Writing to the CCAPxH register sets the ECOM bit of the CCAPMx register automatically. When the PCA mode is used for PWM, it is used to control the loading register of the output duty cycle. When the low 8-bit of the counter overflows, the loading register is automatically updated to the PWM compare register.

12.4.7 Lower 8-bit of the Compare/capture Data Registe (PCA_CCAP0~4L)

Offset address

CCAP0L: 0x028 CCAP1L: 0x030 CCAP2L: 0x038;

CCAP3L: 0x040 CCAP4L: 0x048

Reset value: 0x0000 0000

Table 418. Lower 8-bit of the Compare/capture Data Register (1)

31:8	7	6	5	4	3	2	1	0
Reserved	CCAPx[7:0]							
-	R/W							

Table 419. Lower 8-bit of the Compare/capture Data Register (2)

Bit	Flag	Description
31:8	Reserved	Reserved bit

7:0	CCAPx[7:0]	<p>Lower 8-bit Register for compare/capture mode.</p> <p>When PCA is in compare/capture mode, it is used to store the lower 8 bits of the 16-bit capture count value. Writing to the CCAPxH register clears the ECOM bit of register CCAPMx automatically.</p> <p>When the PCA mode is in PWM mode, it is used to control the output duty cycle comparison register. In PWM mode, if the lower 8 bits of the counter are less than the value of CCAPx[7:0], the PWM output is low, otherwise the PWM output is high.</p>
-----	------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

12.4.8 16-bit Register for Compare/Capture (PCA_CCAP0~4)

Offset address

CCAP0: 0x050 CCAP1: 0x054 CCAP2: 0x058

CCAP3: 0x05C CCAP4: 0x060

Reset value: 0x0000 0000

Table 420. 16-bit Register for Compare/Capture (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 421. 16-bit Register for Compare/Capture (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCAPx[15:0]															
R/W															

Table 422. 16-bit Register for Compare/Capture (3)

Bit	Flag	Description
31:16	Reserved	Reserved bit
15:0	CCAPx	<p>The 16-bit register for compare/capture mode.</p> <p>Used to save the 16-bit capture count value when PCA is in compare/capture mode. Writing the CCAPx register will set the ECOM bit of the CCAPMx register.</p> <p>Writing the CCAPX register is equivalent to writing the two 8-bit registers, CCAPxL and CCAPxH. This register can be read and written directly in compare/capture mode. In PWM mode, the CCAPxL and CCAPxH registers are used.</p>

12.4.9 High Speed Output Flag Register for Comparator (PCA_CCAPO)

Offset address: 0x04C

Reset value: 0x0000 0000

Table 423. High Speed Output Flag Register for Comparator (1)

31:8	7	6	5	4	3	2	1	0
Reserved				CCAPO4	CCAPO3	CCAPO2	CCAPO1	CCAPO0

-	R/W	R/W	R/W	R/W	R/W
---	-----	-----	-----	-----	-----

Table 424. High Speed Output Flag Register for Comparator (2)

Bit	Flag	Description
31:5	Reserved	Reserved bit
4	CCAPO4	The output value of compare module 4
3	CCAPO3	The output value of compare module 3
2	CCAPO2	The output value of compare module 2
1	CCAPO1	The output value of compare module 1
0	CCAPO0	The output value of compare module 0

13 Advanced Timer

13.1 Advanced Timer Introduction

Advanced Timer consists of 3 high performance timers, timer 4/5/6, with same functionalities.

It can be used to count and generate different forms of clock waveforms. One timer can generate a complementary pair of PWM outputs or 2 independent PWM outputs, which can capture external input for pulse width or period measurement.

The basic functions and features of the advanced timer are shown in the below table.

Table 425. Advanced Timer Basic Features

Waveform Mode	Sawtooth wave and triangle wave
Basic Function	Up and down counting directions
	Software synchronization
	Hardware synchronization
	Buffer function
	Quadrature encoding count
	General purpose PWM output
	Protection mechanism
	AOS related actions
	Count compare matching Interrupt
Interrupt Type	Count cycle matching interrupt
	Dead time error interrupt
	Short circuit monitoring interrupt

Table 426. Advanced Timer Port List

Bit	Flag	Description
TIMx_CHA	Input/output	Input/output quadrature encoding count clock input port, capture
TIMx_CHB		Input port or compare output port (x = 4~6)
TIMTRIA	Input	Port for hardware count clock input or capture input.
TIMTRIB		
TIMTRIC		Port for hardware start/stop/clear conditions input.
TIMTRID		

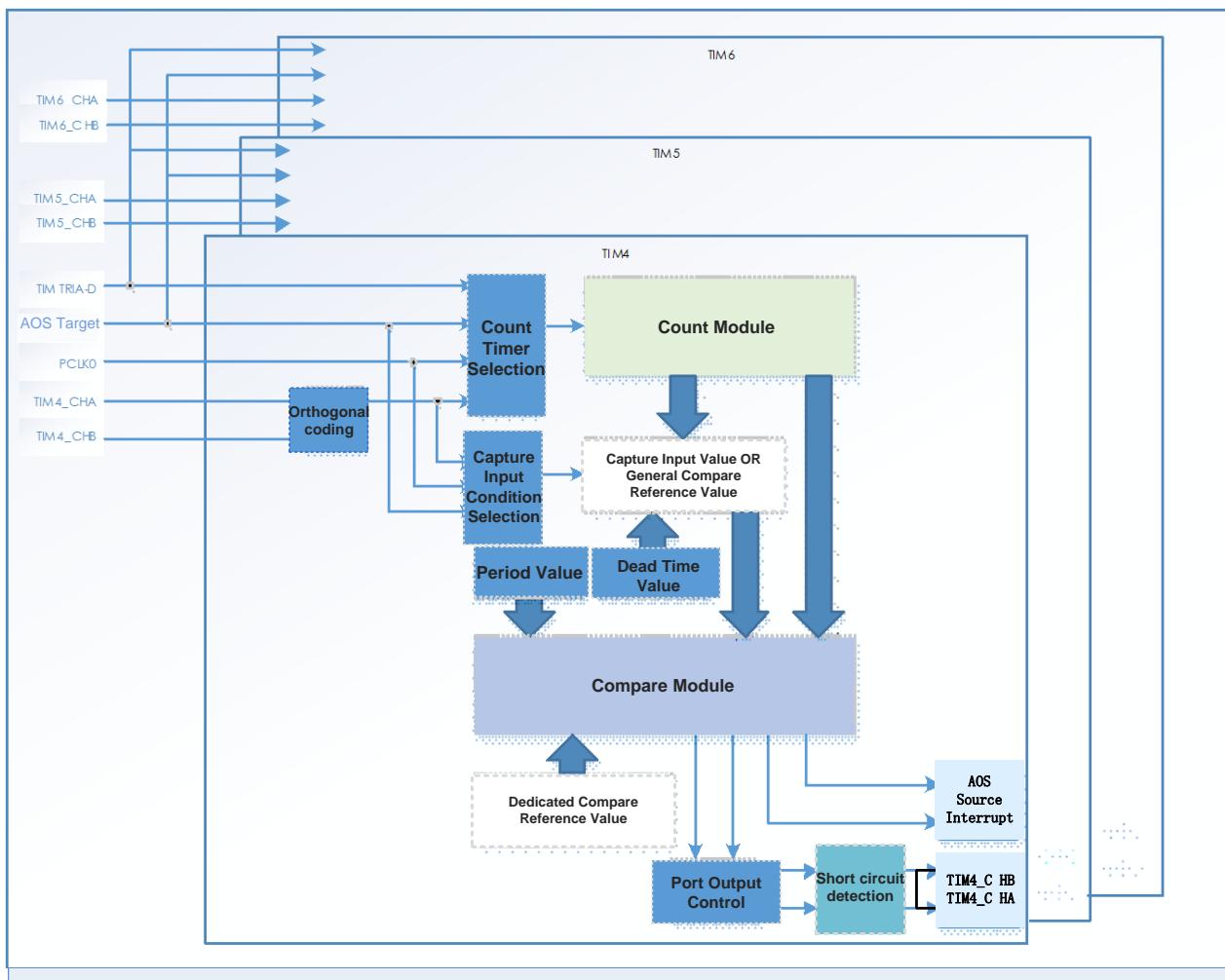


Figure 42. Advanced Timer Block Diagram

13.2 Advanced Timer Description

13.2.1 Basic Actions

13.2.1.1 Basic Waveform Mode

Timer 4/5/6 has two basic count waveform modes, sawtooth mode and triangular wave mode. The waveform modes are further classified according to different internal counting actions. The triangular wave mode is classified into a triangular wave A mode and a triangular wave B mode. The basic waveforms of the sawtooth and triangle waves are shown in Figure 43 and Figure 44. The triangular wave A mode and the triangular wave B mode differ in buffer transmission. The triangular wave A mode only has one buffer-transmission (valley point) in one cycle, whereas the triangular wave B mode has two buffer-transmissions (peak point and valley point) in one cycle.

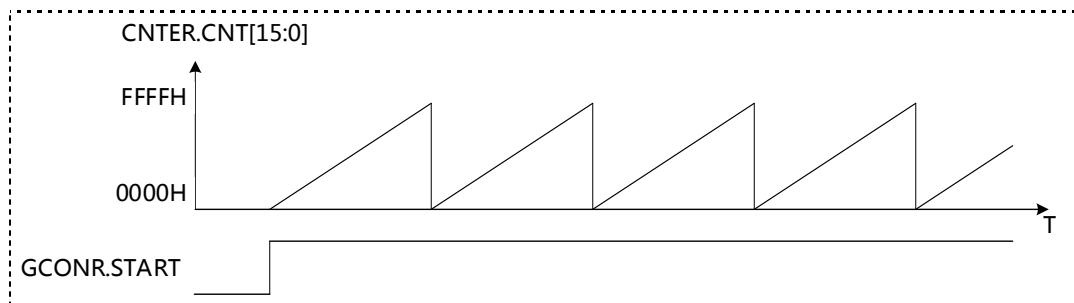


Figure 43. Sawtooth Waveform (counting up)

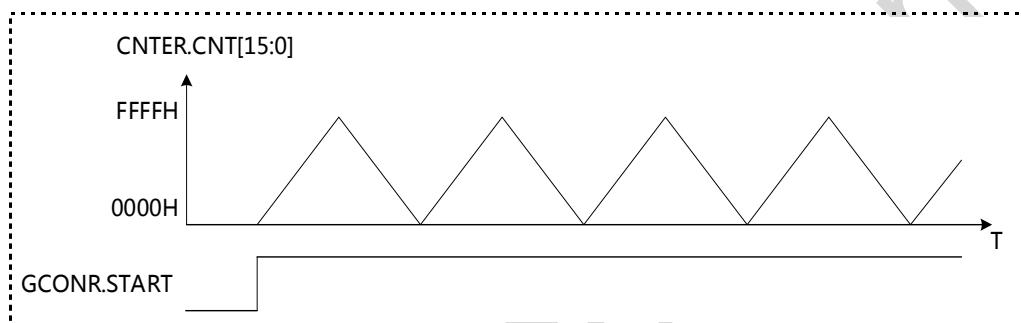


Figure 44. Triangle Waveform

13.2.1.2 Compare Output

There are 2 compare output ports (CHxA, CHxB) in each of timer 4/5/6. They can output the specified level when the count value matches the count reference value. The GCMAR and GCMBR registers correspond to the count compare reference values of CHxA and CHxB respectively. When the count value of the counter is equal to GCMAR, CHxA outputs the specified level. When the count value of the counter is equal to GCMBR, CHxB outputs the specified level.

Such levels as the start level of the CHxA and CHxB ports, the stop level, the level at which the compare is matched, etc., can be configured by setting the PCONR.STACA, PCONR.STPCA, PCONR.STASTPSA, PCONR.CMPCA[1:0] of the port control register (PCONR.), PCONR.PERCA[1:0] and PCONR.STACB, PCONR.STPCB, PCONR.STASTPSB, PCONR.CMPCB[1:0] and PCONR.PERCB[1:0]. An example of the compare output operation is shown in the below figure.

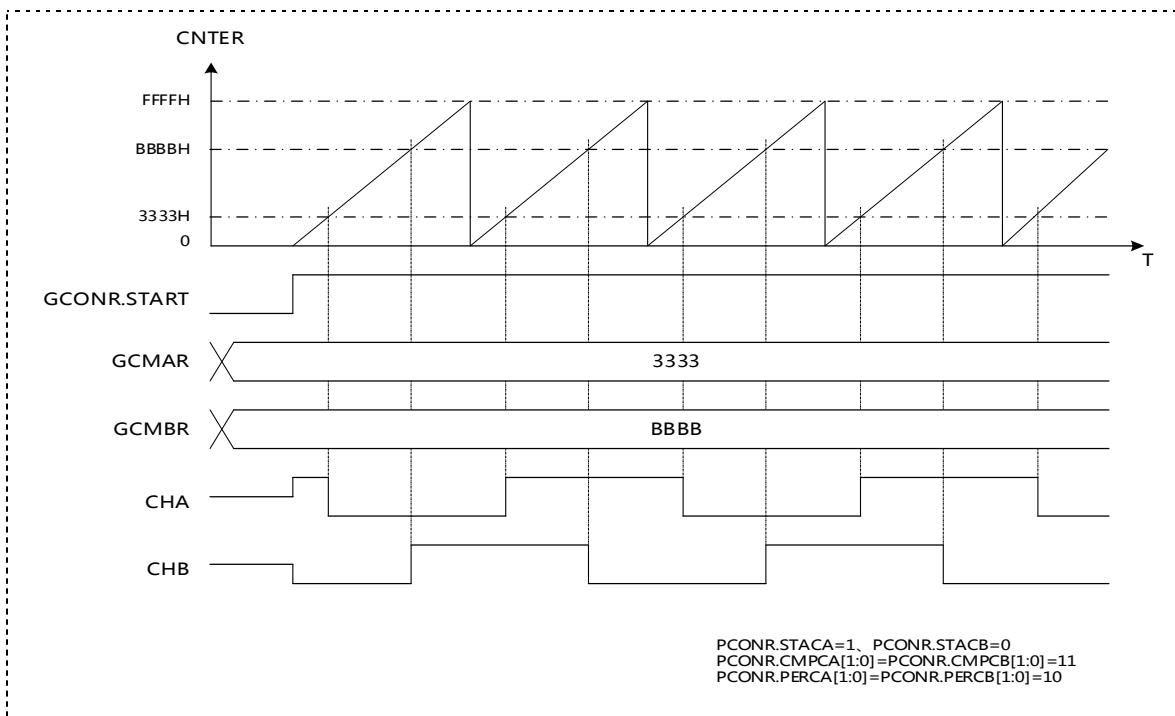


Figure 45. Compare Output Action

13.2.1.3 Capture Inut

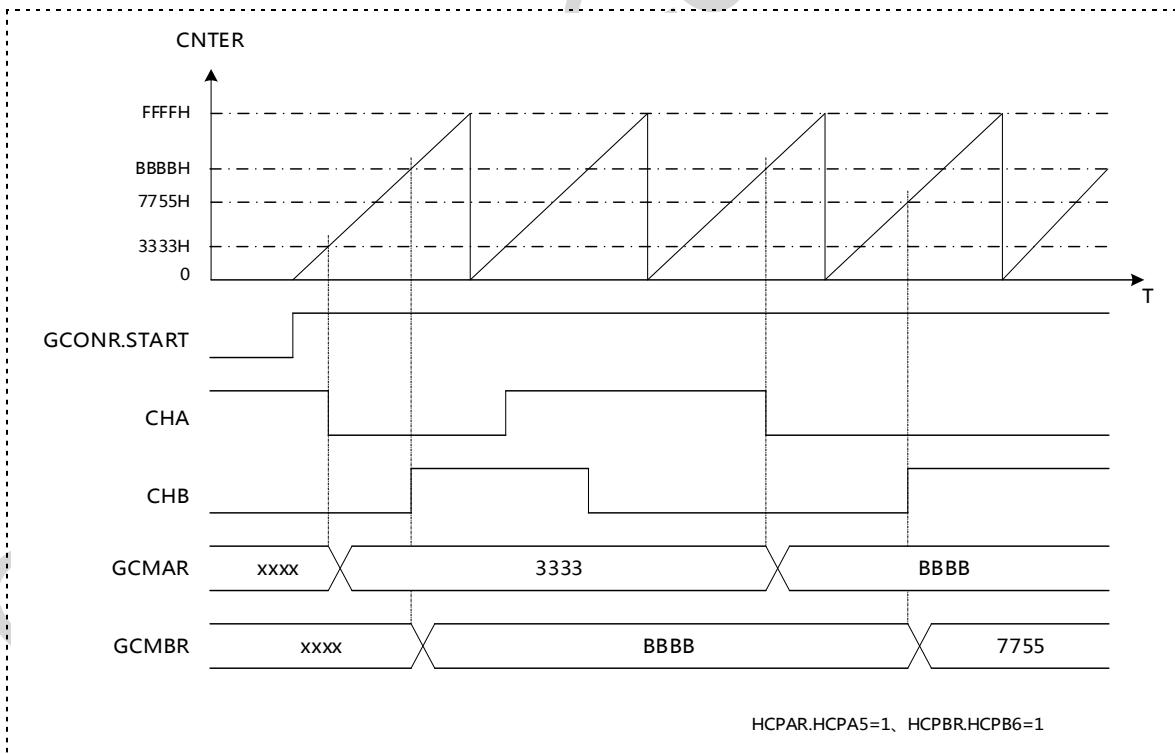


Figure 46. Capture Input Action

Timer 4/5/6 supports the capture input function. They provide 2 sets of capture input registers (GCMAR, GCMBR) for saving the captured count value. The capture input function is enabled by setting the PCONR.CAPCA and PCONR.CAPCB bits of the port

control register (PCONR) to 1. When the corresponding capture input conditions are set and the conditions are valid, the current count value is saved to the corresponding register (GCMAR, GCMBR).

The conditions for each set of capture inputs can be AOS event trigger, TIMTRIA-TIMTRID input, CHxA or input, etc. The specific conditions can be set using the hardware capture event selection register (HCPAR, HCPBR). An example of input capture action is shown in Figure 46.

13.2.2 Clock Source Selection

The timer 4/5/6 count clock supports the following options.

- a) PCLK and the 2, 4, 8, 16, 64, 256, 1024 frequency divisions of PCLK (configured by setting GCONR.CKDIV[2:0])
- b) AOS event trigger input (configured by setting HCUPR.HCUP [19:16] or HCDOR.HCDO [19:16])
- c) Quadrature encoding input of CHxA and CHxB (configured by setting HCUPR.HCUP[7:0] or HCDOR.HCDO [7:0])
- d) Port input for TIMTRIA-TIMTRID (configured by setting HCUPR.HCUP [15:8] or HCDOR.HCDO [15:8])

As seen from the above description, the clocks of b, c, and d are independent of each other, and can be set to be enabled and disabled respectively. When the clocks of b, c, and d are selected, the clock of a is disabled automatically.

13.2.3 Counting Direction

The counting direction of timer 4/5/6 can be changed by software. The method of changing the counting direction is slightly different for different waveform modes.

13.2.3.1 Sawtooth Counting Direction

In the sawtooth mode, it supports setting counting direction whenever the counter runs or stops..

In the count-up case, when setting GCONR.DIR=0 (count down), the counter will count up until overflow occurs, then turns to count-down mode. In the count-down case, when setting GCONR.DIR=1 (count up) the counter counts down until overflow occurs, then turns to count-up mode.

13.2.3.2 Triangle wave counting direction

In the triangular wave mode, the counting direction can only be set when the counter stops. Setting the counting direction during counting is invalid.

Set the GCONR.DIR bit when the count stops. After the counter starts, it counts until overflow occurs and the GCONR.DIR setting takes in effect then.

13.2.4 Digital Filtering

The CHxA, CHxB, and TIMTRIA~D port inputs of timer 4/5/6 support digital filtering. The filtering function of the corresponding port can be enabled by setting the relevant enabling bit of the filter control register (FCONR). The reference clock for filtering is also set by the filter control register (FCONR).

When the filtered sampling reference clock samples 3 identical levels, the sampled level is transmitted to the module as an effective level. When the number of sampled identical levels is less than 3, it will be filtered out as external interferences and not transmitted into the module. An example is shown in the below figure.

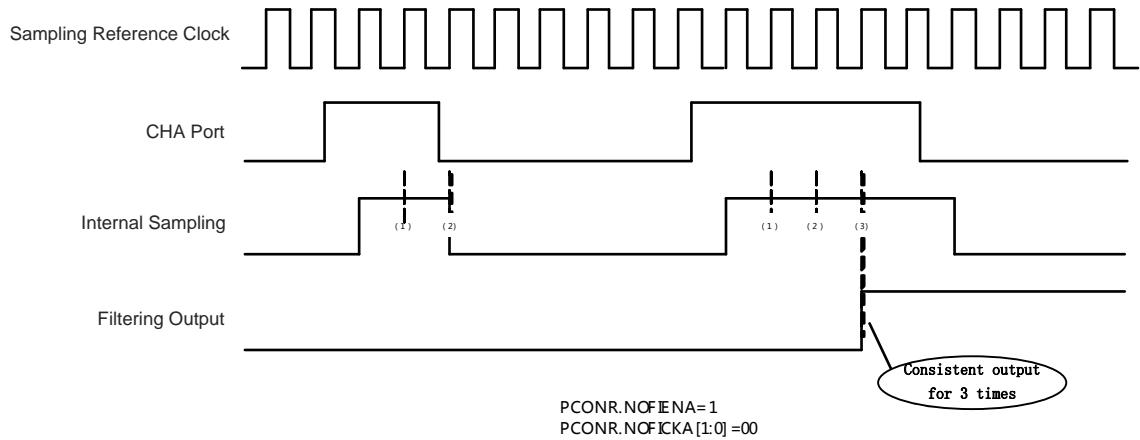


Figure 47. Filtering Function of the Capture Input Port

The TIMTRIA~D port is a group of ports shared by timer 4/5/6. The digital filtering function of this group of ports is implemented only in timer 4. For the other timers, timer 5/6, the digital filtering function related setting is invalid.

13.2.5 Software Synchronization

13.2.5.1 Software Synchronous Start

The synchronous startup of the target timer 4/5/6 can be fulfilled by setting the relevant bit of the software synchronization start register (SSTAR).

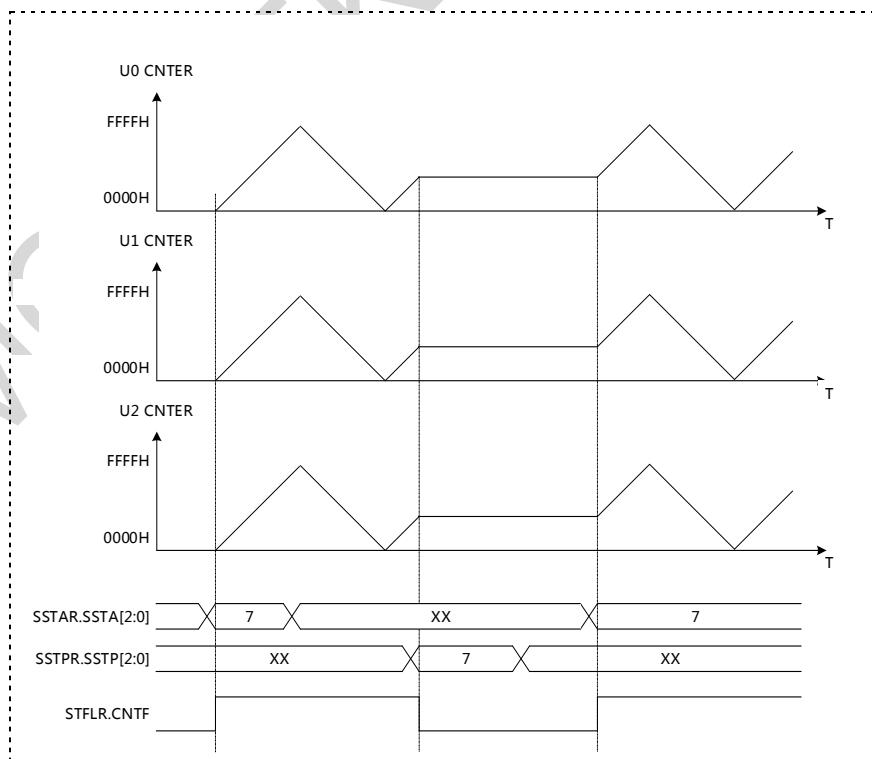


Figure 48. Action of Software Synchronization

13.2.5.2 Software Synchronous Stop

The synchronous stop of the target timer 4/5/6 can be fulfilled by setting the relevant bit of the software synchronization stop register (SSTPR).

13.2.5.3 Software Synchronous Clearing

The synchronous clearing of the target timer 4/5/6 can be fulfilled by setting the relevant bit of the software synchronization stop register (SCLRR).

As shown in Figure 48, the synchronous clearing of the timer 4/5/6 can be implemented by setting of SSTAR.SSTA0 = SSTAR.SSTA1 = SSTAR.SSTA2 on timer 4.

The software synchronous action related registers (SSTAR, SSTPR, SCLRR) are a set of registers shared by all TIMs and independent with timer 4/5/6. For each bit of this register set, setting to 1 is valid and setting to 0 is invalid. It can read the counter status of each timer by reading the SSTAR register. It is read as 0 when reading SSTPR or SCLRR.

13.2.6 Hardware Synchronization

In addition to the 2 universal input ports (CHxA and CHxB), each timer supports 4 external universal input ports (TIMTRIA, TIMTRIB, TIMTRIC and TIMTRID) and 4 AOS as well, targeting for the hardware synchronization function among timers.

13.2.6.1 Synchronous Startup by Hardware

It can choose to start the counter by hardware for each one of timer 4/5/6. Timers selecting the same hardware start conditions can achieve synchronous startup if the startup conditions are valid. The specific hardware startup conditions depend on the setting of the hardware startup event selection register (HSTAR).

13.2.6.2 Hardware Synchronous Stop

It can choose to stop the counter by hardware for each of timer 4/5/6. Timers selecting the same hardware stop conditions can achieve synchronous stop if the stop conditions are valid. The specific hardware stop conditions depend on the setting of the hardware stop event selection register (HSTPR).

13.2.6.3 Hardware Synchronous Clearing

It can choose to clear the counter by hardware for each of timer 4/5/6. Timers selecting the same hardware clearing conditions can achieve synchronous clear when the clearing conditions are valid. The specific hardware clearing conditions depend on the setting of the hardware clearing event selection register (HCLR).

13.2.6.4 Hardware Synchronous Capture

It can choose to fulfill capture input function by hardware for each of timer 4/5/6. Timers selecting the same hardware capture input conditions can achieve synchronous capture input when the capture conditions are valid. The specific hardware capture input conditions depend on the setting of the hardware capture event selection register (HCPAR, HCPBR).

13.2.6.5 Synchronous Count by Hardware

It can choose to use hardware input as clock for count for each of timer 4/5/6. Timers selecting the same hardware count conditions can achieve synchronous count when the count conditions are valid. The specific hardware count conditions depend on the setting of the hardware count up selection register (HCUPR) and the hardware count down selection register (HCDOR).

When the hardware synchronous count function is selected, it just selects the external input clock source, and the startup, stop, and clearing actions of the counter are not affected. It still needs separate settings per the startup, stop, and clearing actions of the counter.

Please see the below figure for the example of the hardware synchronization actions of the timer 4/5/6.

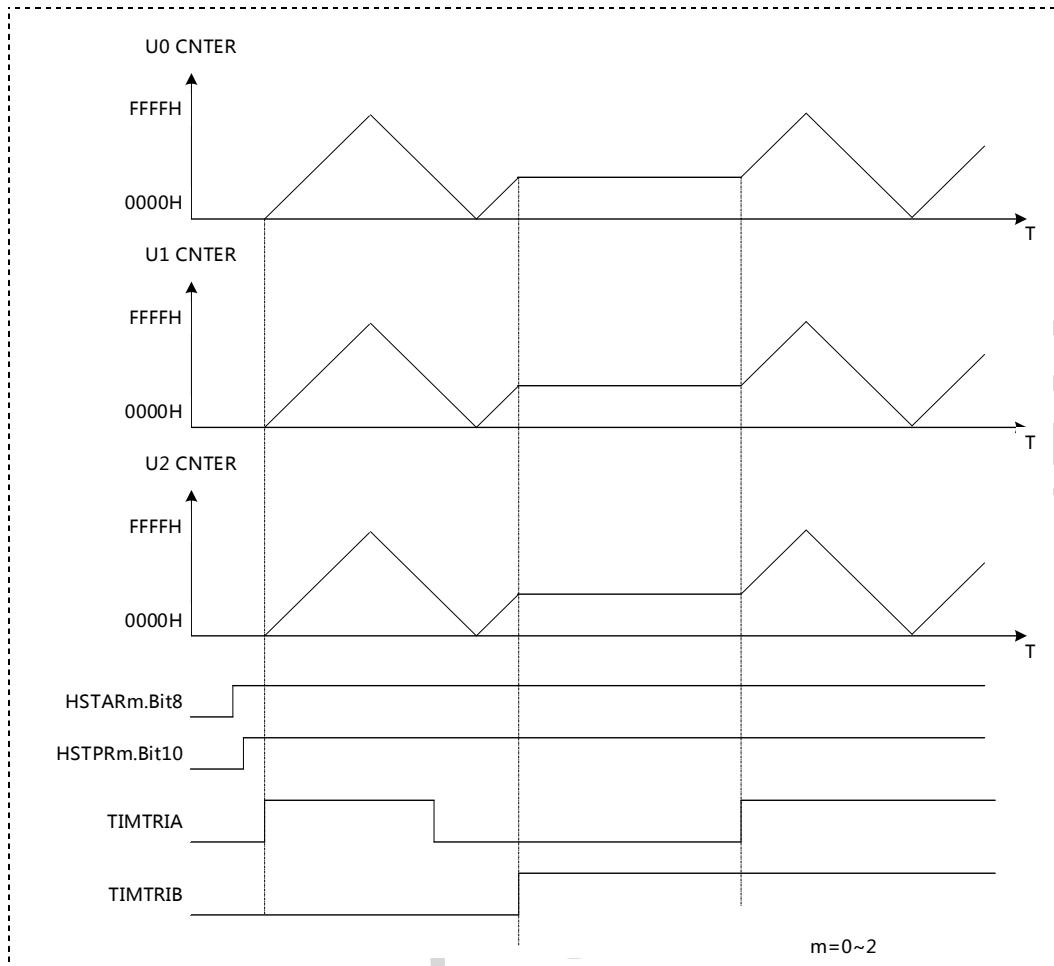


Figure 49. Hardware Synchronization Actions

13.2.7 Buffer Function

The buffer function selects the following events at the buffer transfer time point by setting the buffer control register (BCONR):

- Transfer the value of the general purpose period reference value buffer register (PERBR) to the general purpose period reference value register (PERAR) automatically.
- Transfer the value of the general purpose compare reference buffer register (GCMCR, GCMDR) to the general purpose compare reference register (GCMAR, GCMBR) automatically (in compare output case).
- Transfer the value of the general purpose compare reference value buffer register (GCMAR, GCMBR) to the general purpose compare reference value register (GCMCR, GCMDR) automatically (in capture input case).

The below figure shows the timing chart of the single buffer mode of the general compare reference value register in compare output case. As seen from the figure, changing the value of the general purpose compare reference register (GCMAR) during the count can adjust the output duty cycle and changing the value of the general purpose period reference register (PERAR) can adjust the output period.

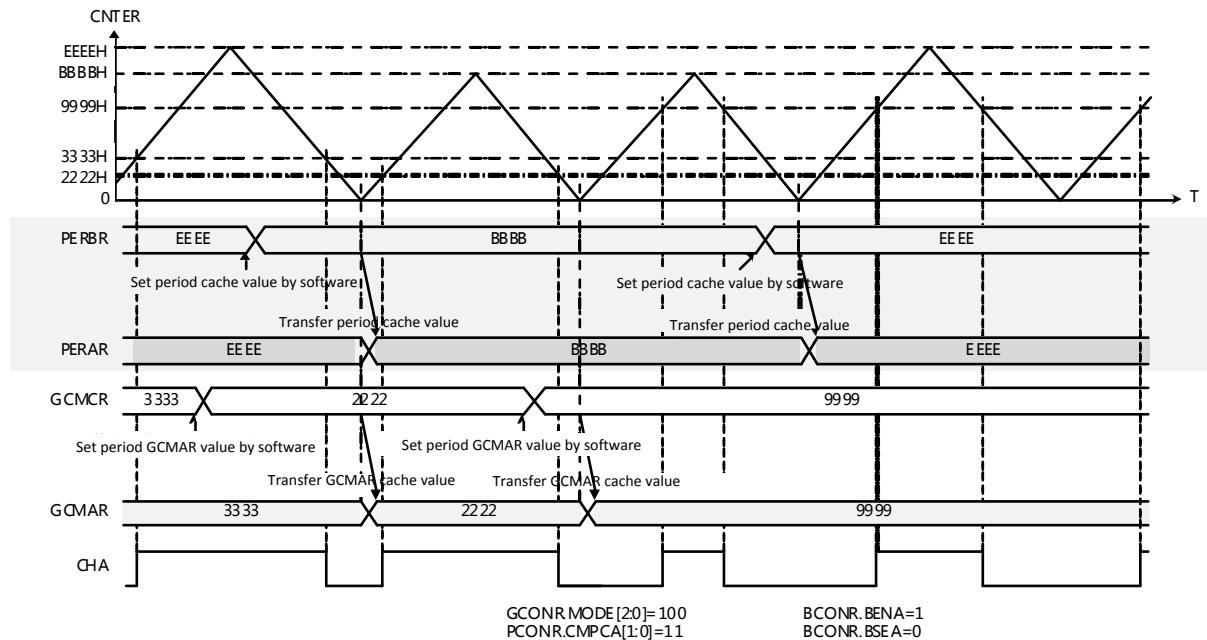


Figure 50. Timing Chart of the Single Buffer Mode

13.2.7.1 Buffer Transfer Time

- **General Purpose Period Reference Value Buffer Transfer Time**

The general purpose period reference value buffer transfer time points are at the overflow point of the up count and the underflow point of the down count of the sawtooth wave and the count valley point of the triangle wave.

- **General Purpose Compare Reference Value Buffer Transfer Time**

In the sawtooth A mode, when setting BCONR.BENA=1 or BCONR.BNEB=1, the buffer action is valid. Buffer transfers occur at overflow points or underflow points.

In the triangular A mode, when setting BCONR.BENA=1 or BCONR.BNEB=1, the buffer action is valid. Buffer transfers occur at the count valley.

In the triangular B mode, when setting BCONR.BENA=1 or BCONR.BNEB=1, the buffer action is valid. Buffer transfers occur at the count valley point and the count peak point.

- **Capture Input Value Buffer Transfer Time**

The capture input buffer is transferred at the time when the input action is captured.

- **Buffer Transfer When Clearing Action Occurs**

In the sawtooth wave count mode or the hardware count mode, if a clear action occurs during the normal compare output period, a buffer-transfer is performed per the general purpose period reference value, the general purpose compare reference value, and the like, according to the corresponding buffer action setting conditions.

13.2.8 General Purpose PWM Output

13.2.8.1 PWM Spread Frequency Output

To reduce the external interference of the PWM output, it supports a spread frequency configuration at the PWM output level. Each PWM output cycle trims the phase of the PWM output.

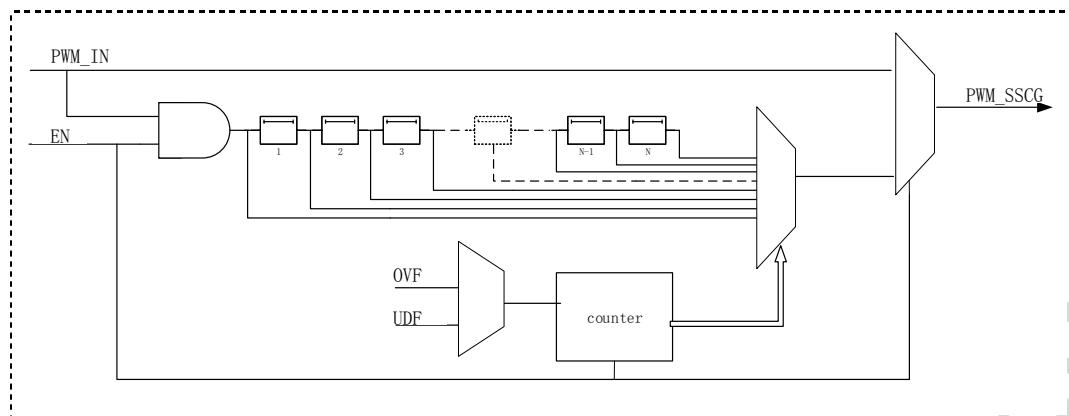


Figure 51. PWM Spread Frequency Output Diagram

13.2.8.2 Independent PWM Output

The two ports of each timer, CHxA and CHxB, can independently output PWM waves. As shown in the below figure, the CHA port of timer 6 outputs a PWM wave.

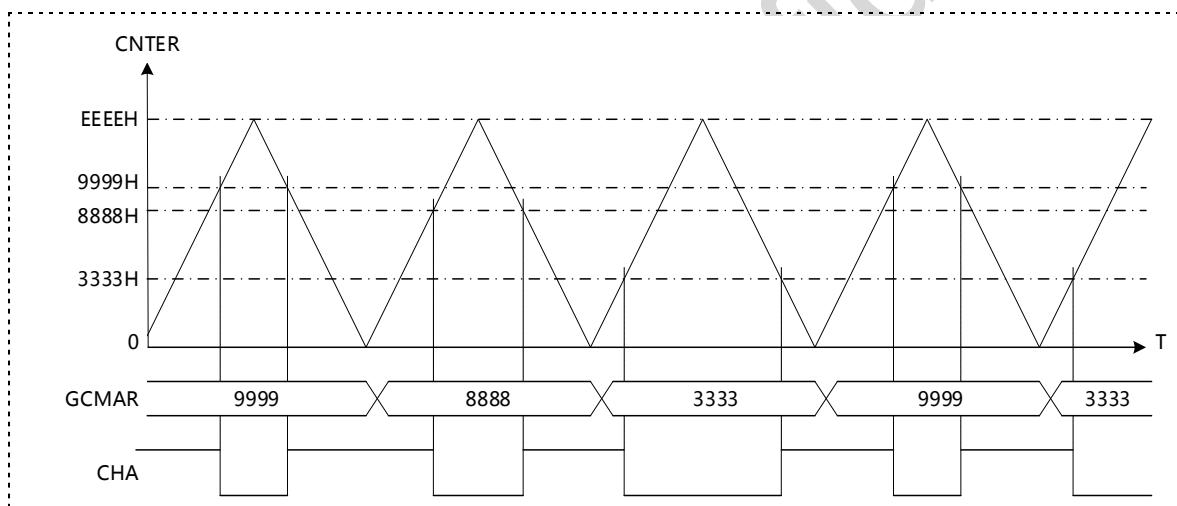


Figure 52.. CHA Output PWM Wave

13.2.8.3 Complementary PWM output

The CHxA port and CHxB port can be combined to output complementary PWM waveforms in different modes.

13.2.8.3.1 Setting GCMBR Complementary PWM Output by Software

This function means that, in the sawtooth mode, the triangular wave A mode and the triangular wave B mode, the value of the general purpose compare reference register GCMBR is used for CHxB port waveform output, which is set via register GCMBR directly, with no direct association with register GCMAR.

The below figure shows an example of the output of the GCMBR complementary PWM wave set by the software.

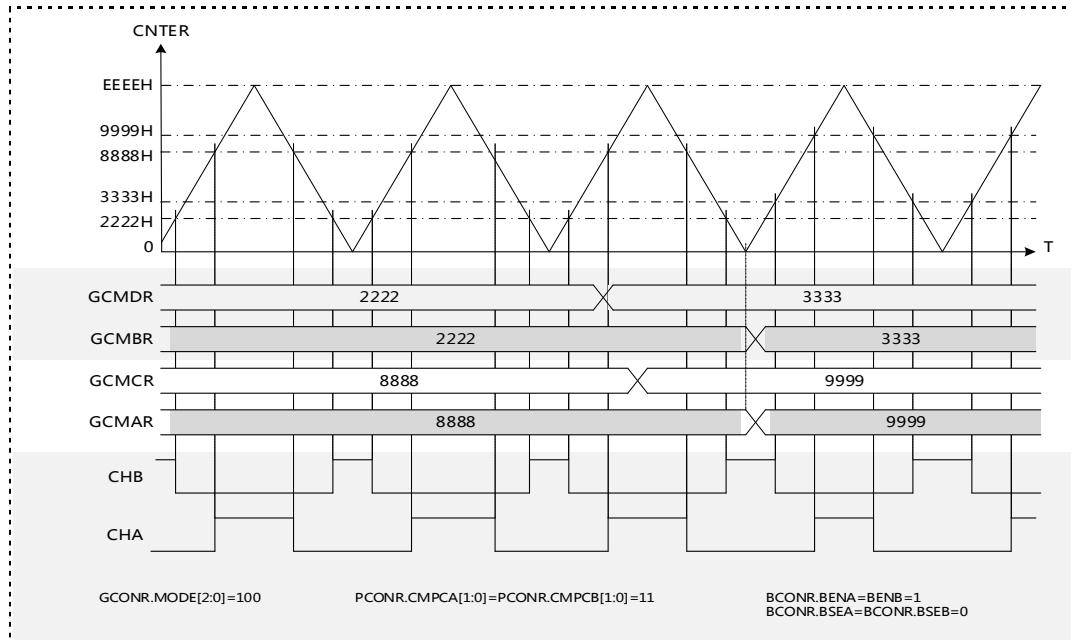


Figure 53. Setting GCMBR complementary PWM wave Output by Software in Triangular Wave A Mode

13.2.8.3.2 Setting GCMBR Complementary PWM Output by Hardware

This function means that, in the triangular wave A mode and the triangular wave B mode, the value of the general purpose compare reference value register (GCMBR) for the CHxB port waveform output depends on the calculation result of the general purpose compare reference value register GCMAR value and the dead time reference value register (DTUAR and DTDAR) value.

The below figure shows an example of setting GCMBR complementary PWM output by hardware.

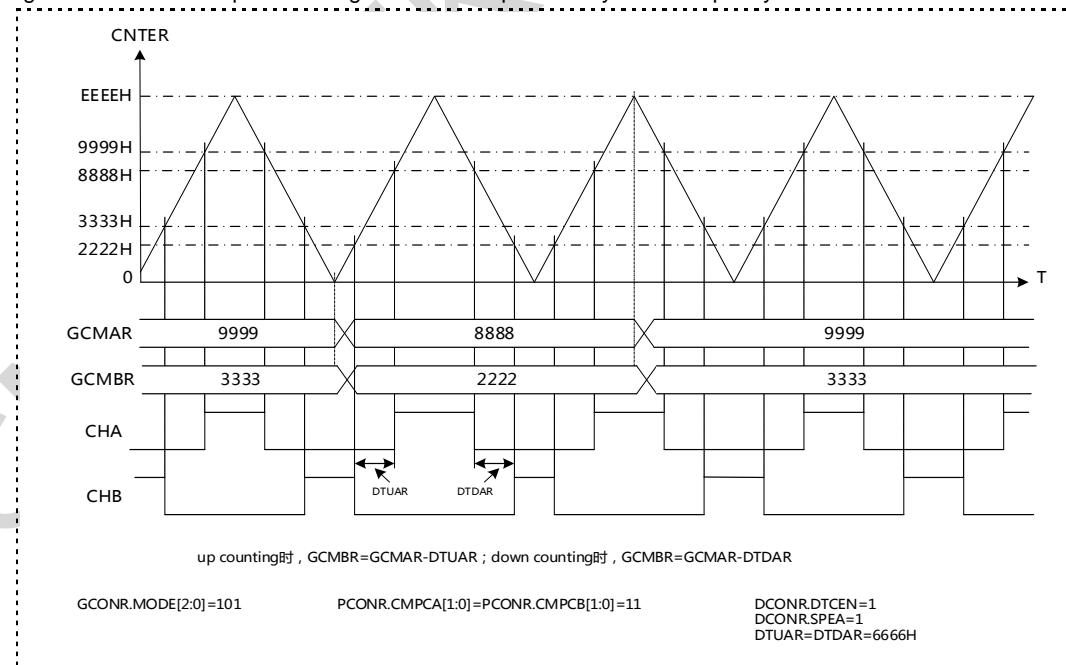


Figure 54. Setting GCMBR Complementary PWM Wave Output (Symmetric Dead Time) by Hardware in Triangle Wave B Mode

13.2.8.4 Multiphase PWM Output

Each timer's CHxA and CHxB ports can output independent 2-phase PWM waves or a set of complementary PWM waves. The multi-phase PWM output can be fulfilled via multiple timers combined along with software and hardware synchronization. As shown in Figure 55, the combine of timer 4, timer 5, and timer 6 outputs 6-phase PWM waves. As shown in Figure 56, the combine of timer 4, timer 5, and timer 6 outputs 3-phase complementary PWM waves.

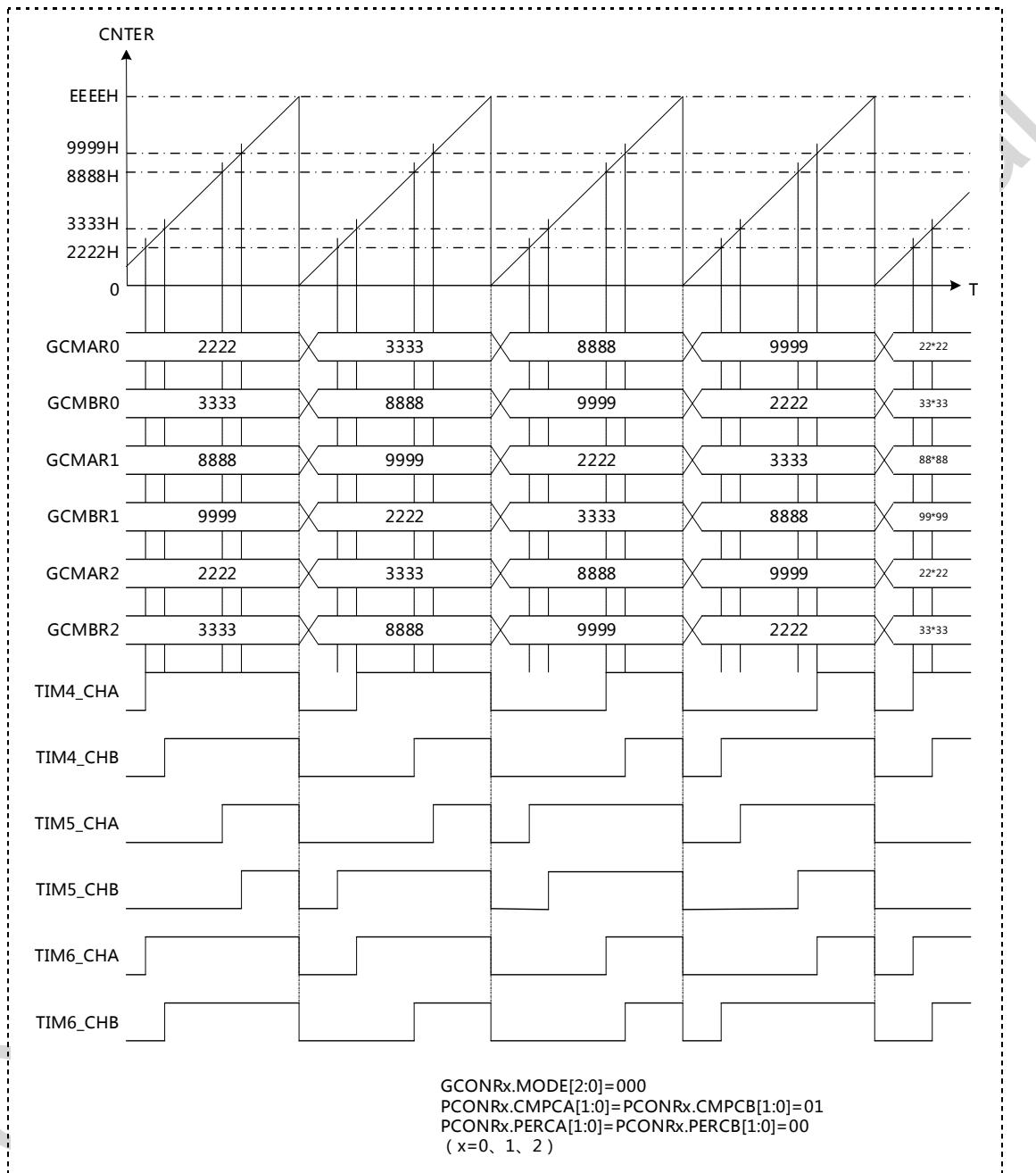


Figure 55. 6-phase PWM Wave

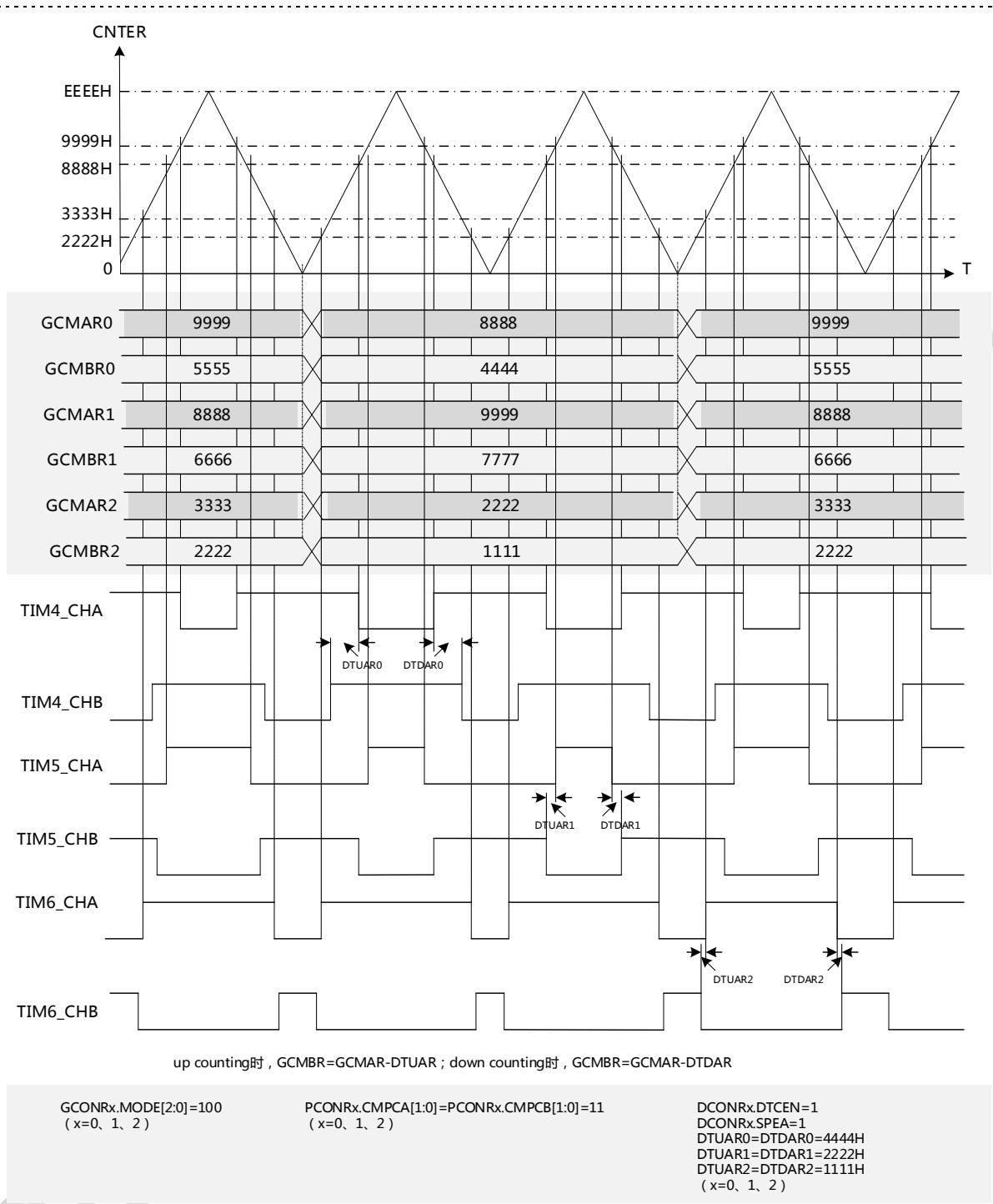


Figure 56. 3-phase Complementary PWM Output with Dead Time in Triangular Wave A Mode

13.2.9 Quadrature Encoding Count

By using the CHxA input as the AIN input, the CHxB input as the BIN input, and any of the TIMTRIA-D inputs as the ZIN input, the advanced timer can support the quadrature encoding count with the 3 inputs.

A single individual timer with independent action of AIN and BIN can support the position count mode. 2 timers with the combination action of AIN, BIN and ZIN can support the revolution count mode. Among them, one timer is used for position count,

another one for revolution count.

In the revolution count mode, every combination of 2 timers (the combination of timer 4, 5, timer 4 as the position count unit, and timer 5 as the revolution count unit) can support position count and revolution count respectively.

The count conditions of AIN and BIN are implemented by setting the quadrature relationship between CHxA and CHxB in the hardware count up selection register (HCUPR) and the hardware count down selection register (HCDOR). The ZIN input clears the position counter in the position count unit by setting the hardware clearing event selection register (HCLRR) in the position count unit. It implements the revolution count in the revolution count unit by setting the hardware count up selection register (HCUPR) in the revolution count unit.

13.2.9.1 Position Count Mode

The quadrature encoding position mode refers to the basic count function, the phase difference count function, and the direction counting function according to the input of AIN and BIN.

13.2.9.1.1 Basic Count

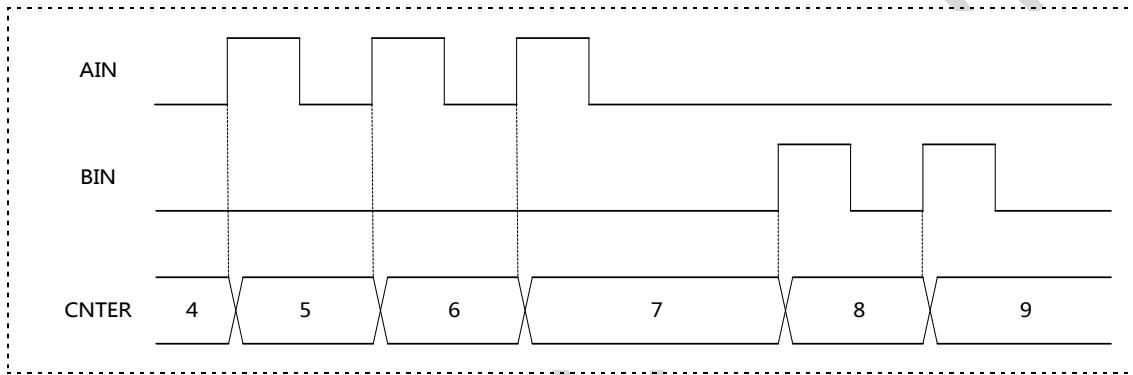


Figure 57. Basic Count Action in Position Mode

By setting the HCUPR and HCDOR registers, users can flexibly implement phase difference counting in various ways.

13.2.9.1.2 Phase Difference Count

The phase difference count performs count based on the phase relationship between AIN and BIN. Depending on the setting, it supports 1x count, 2x count, 4x count, etc. as shown in the following Figure 58 ~ Figure 60.

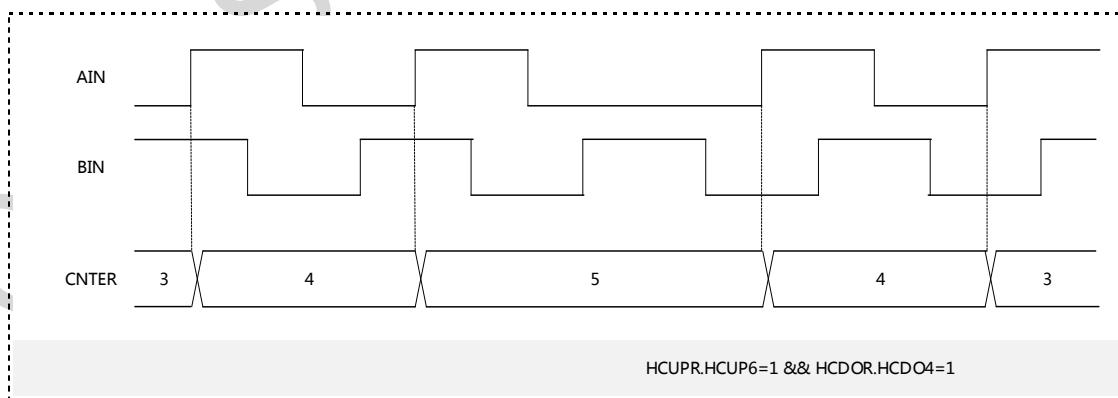


Figure 58. Phase Difference Count Action Setting in Position Mode (1x)

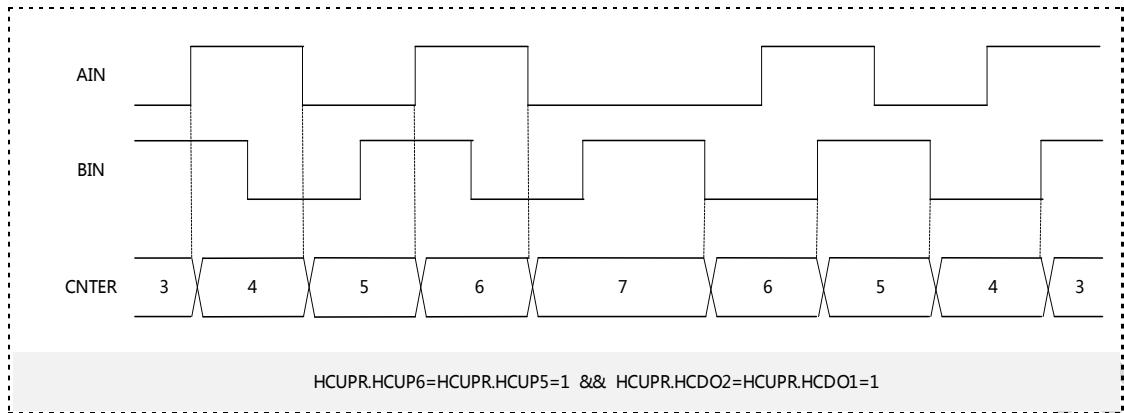


Figure 59. Phase Difference Count Action Setting in Position Mode (2x)

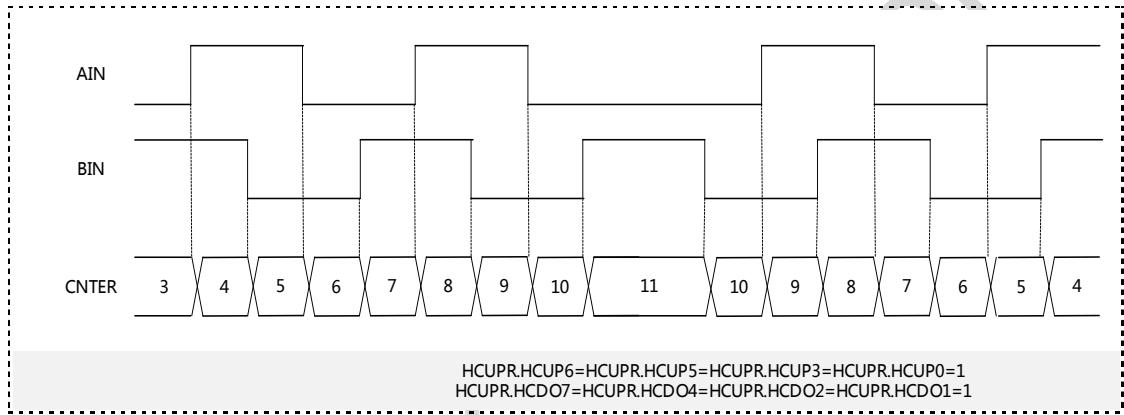


Figure 60. Phase Difference Count Action Setting in Position Mode (4x)

13.2.9.1.3 Direction Count

Direction count sets the input state of AIN as direction control. BIN input is used as clock count, as shown in the below figure.

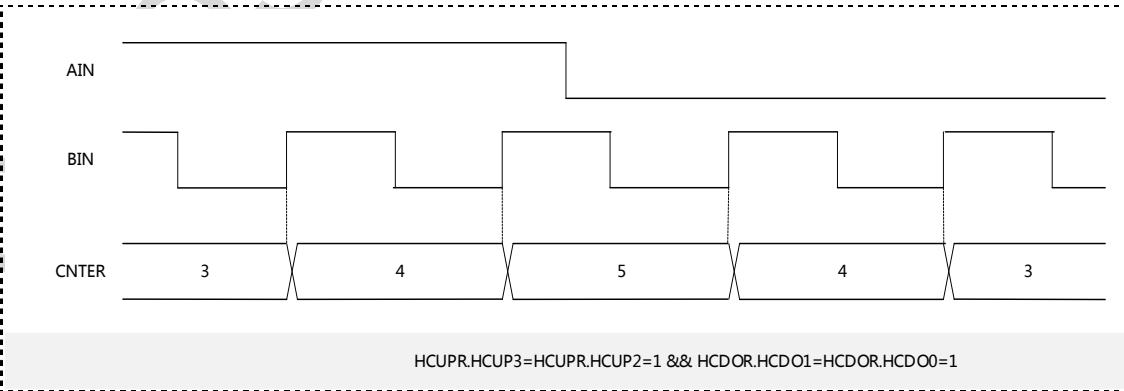


Figure 61. Direction Count Action in Posision Mode

13.2.9.2 Revolution Mode

The quadrature encoding revolution mode refers to using AIN and BIN counts plus ZIN input events to determine the number of revolutions. In the revolution mode, it can fulfill the Z-phase counting function, position counter output counting function, and Z-phase counting and position counter output mixing counting function according to the counting modes of the revolution counter. This is achieved using two Advanced Timers.

13.2.9.2.1 Z-phase Count

The Z-phase count is a counting operation in which the revolution counting unit counts and the position counting unit is cleared according to the input of ZIN.

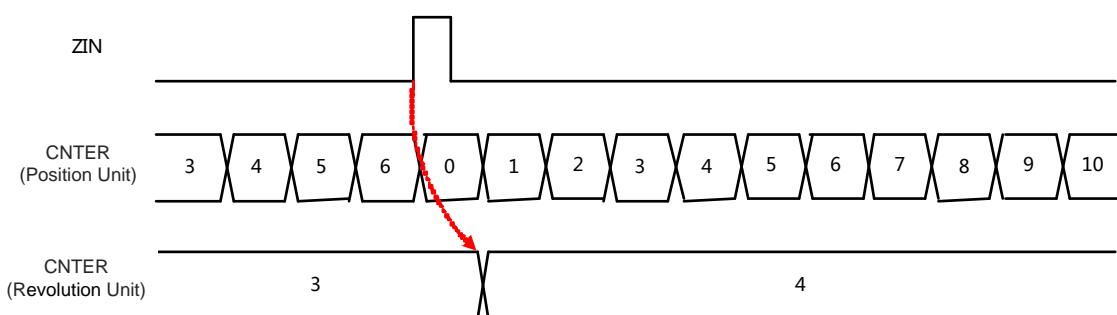


Figure 62. Z-phase Count Action in Revolution Mode

13.2.9.2.2 Position Overflow Count

The position overflow count means that, an overflow event, generated when overflow or underflow occurs, triggers the revolution count unit counting once (in this count mode, the ZIN input performs neither the count action of the revolution count unit nor the clearing action of the position counting unit).

It fulfills the position count by the overflow event of the of the position count unit co-functioning with the associated selection mechanism of the AOS module. In the revolution counter unit, the count up (or count down) event of the hardware count up (or count down) selection register (HCUPR or HCDOR) selects 1 bit in Bit16:19, and the AOS module sets the event source of the corresponding count up (or count down) event as the count overflow event of the position count unit. Please refer to the AOS related chapter for more details. See the below figure for position overflow count action.

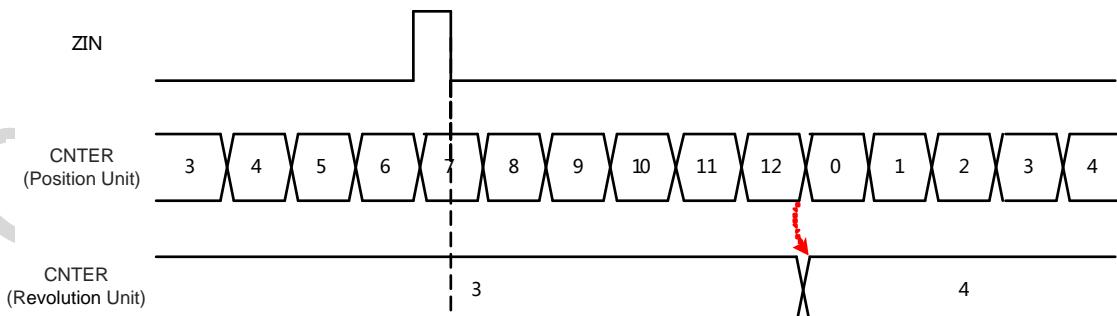


Figure 63. Output Count Action of the Position Counter in Revolution Mode

13.2.9.2.3 Mixed Count

The mixed count refers to combined action of the above-mentioned Z-phase count action and position overflow count action. The implementation is also the combination of the above two count methods accordingly, as shown in the below figure.

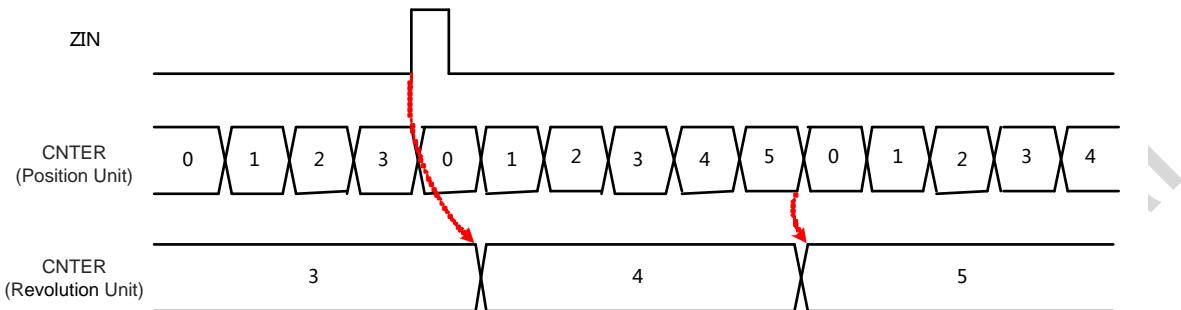


Figure 64. Mixed Output Count Action of Z-phase Count and Position Counter in Revolution Mode

13.2.9.2.4 Z-phase Action Disabling

In the Z-phase count function or the mixed count function of the revolution count mode, it supports disabling the effective ZIN input several cycles after the overflow or underflow time point of the position counter (disable by setting GCONR.ZMSK[0:1]), thus the count function of the revolution count unit and the clearing function of the position count unit are not performed.

When the GCONR.ZMSKPOS of the general control register (GCONR) in the position count unit is 1, the Z-phase disabling function of the position count unit is enabled. The number of cycles of the Z-phase disabling function is set by GCONR.ZMSK. When GCONR.ZMSKREV of GCONR in the general control register in the revolution count unit is 1, the Z-phase disabling function of the revolution count unit is enabled.

As shown in the below figure, in the mixed count mode, 4 counting cycles after the position count unit overflows, there is a ZIN phase input, however it is ineffective, meaning that neither the revolution count unit counts nor the position count unit clears. The further ZIN phase input resumes normal action.

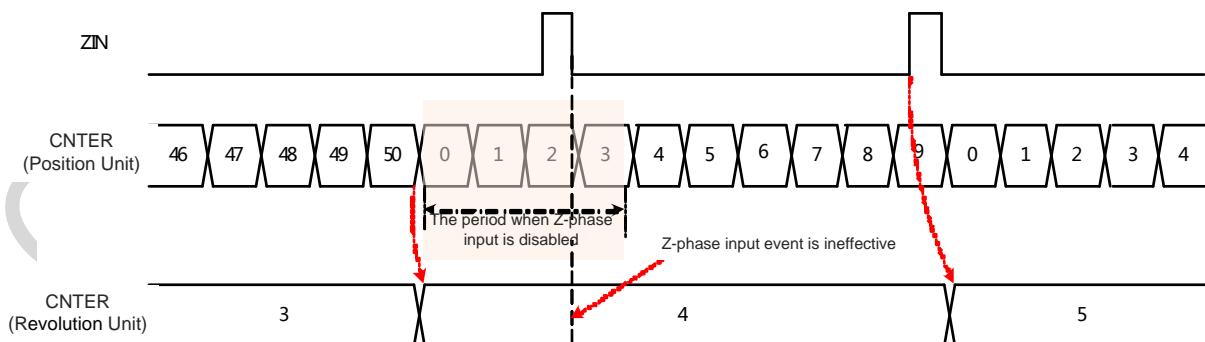


Figure 65. Mixed Revolution Count Mode - Z-phase Disabling Example 1

As shown in the below figure, in the mixed revolution count mode, during the 3rd cycle after the position count unit overflows, the count direction changes. The disabling period which is set as 4 cycles becomes invalid (the actual ZIN phase disabling time lasts for only 3 cycles, and starts counting down). After the count overflow occurs in the position count unit, the ZIN phase disabling

function resumes and lasts for 4 cycles then becomes ineffective. During the ZIN phase disabling period, ZIN phase input function is ineffective, that is, neither the revolution count unit counts nor the position count unit clears. Afterwards, the futher ZIN phase input resumes normal behavior.

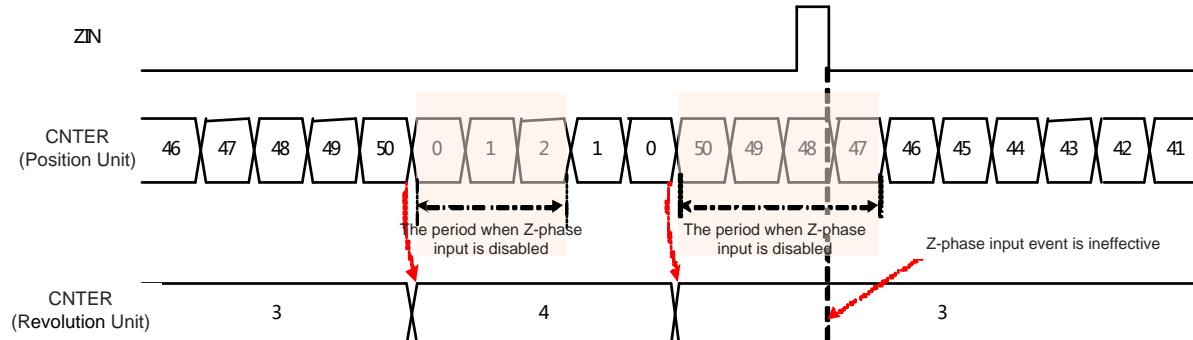


Figure 66. Mixed Revolution Count Mode - Z-phase Disabling Example 2

13.2.10 Response per Interval Cycles

The general purpose compare reference value register (GCMAR ~ GCMDR) for timer 4/5/6 can generate a dedicated valid request signal when count and compare is matched. The signal is sent to the AOS module to associate actions with other modules.

It can generate a valid request signal every few cycles. By setting the VPERR.PCNTS bit of the valid period register (VPERR), it can specify how many cycles the request signal is valid once. In other cycles, even if the count value is equal to the value of the compare reference value register GCMAR or GCMBR, no valid signal outputs. The below figure shows an example of the valid request signal occurred every few cycles.

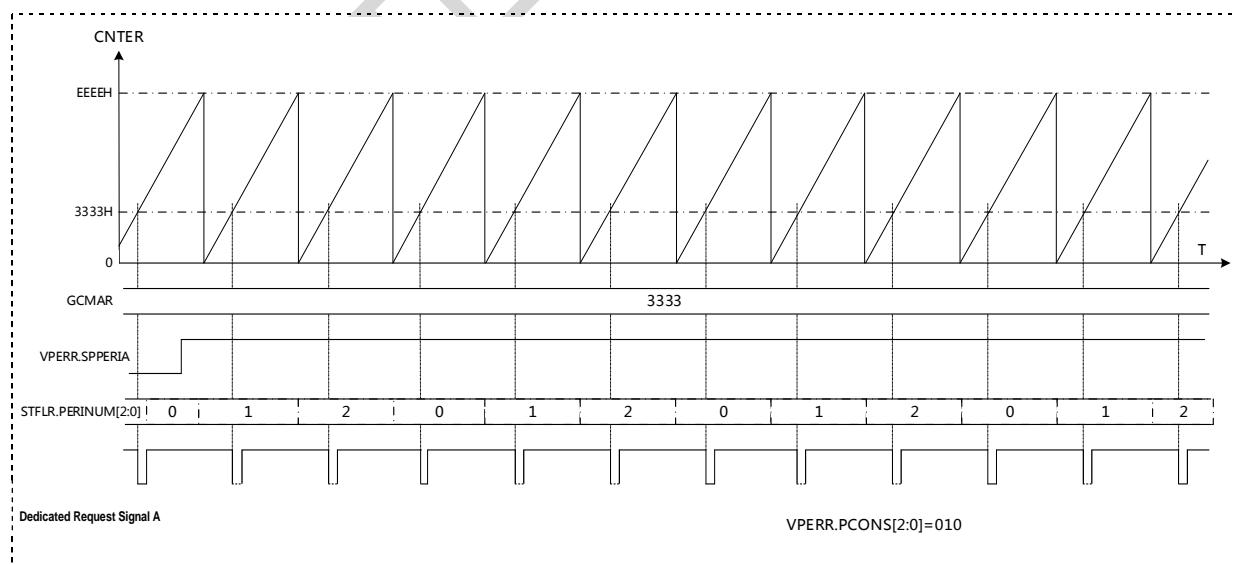


Figure 67. Action of Valid Request Signal Per Interval Cycles

13.2.11 Protection Mechanism

The advanced timer provides protection control over port output state.

The advanced timer supports 4 shared port output invalid event interfaces connecting to the 4 sets of brake events output by the brake control module. The abnormal condition events selected on each interface is set in brake control modules. The abnormal condition events detected on these interfaces can be used to control the general PWM output.

In normal output state, if a brake event from brake control module is detected, the port output state changes to a preset state. When an abnormal event for brake control occurs, the general purpose PWM port output state changes to high-impedance output state, low level output, or high level output (depending on the setting of PCONR.DISVALA and PCONR.DISVALB).

For example, during normal output period of the CHxA port, if PCONR.DISSELA[1:0]=01&PCONR.DISVALA=01 is set, the output on the CHxA port becomes high-impedance if a brake event occurs corresponding to the output invalid condition 1.

13.2.12 Interrupt Description

Each of timer 4/5/6 contains a total of 9 interrupts with 3 types, including 4 general purpose count compare match interrupts (2 capture input interrupts among them), 2 count cycle match interrupts, and 1 dead time error interrupt.

13.2.12.1 Count Compare Match Interrupt

A total of 4 common compare reference value registers (GCMAR-GCMDR) can be compared with the count values to generate compare match valid signal. When the count compare matches, the STFLR.CMAF~STFLR.CMDF bits in the status flag register (STFLR) are set to 1 respectively. At this time, if the corresponding bit in ICONR.INTENA~ICONR.INTEND of the interrupt control register (ICONR) is set to enable the interrupt, the corresponding interrupt request will be triggered as well.

The capture input action occurs when the capture input valid condition selected by the hardware capture event selection register (HCPAR, HCPBR) is generated. At this time, if the ICONR.INTENA or ICONR.INTENB bit of the interrupt control register (ICONR) is set to 1 to enable the interrupt, the corresponding interrupt request is triggered.

13.2.12.2 Count Cycle Match Interrupt

When the sawtooth wave counts up to the overflow point, or counts down to underflow point, or the triangle wave counts to the valley point, the STFLR.OVFF or STFLR.UDFF bit of the status flag register (STFLR) is set to 1. If the interrupt is enabled by setting the ICONR.INTENOVF bit and the ICONR.INTENUFD bit in the interrupt control register (ICONR), the count period match interrupt can be triggered at the corresponding time.

13.2.12.3 Dead Time Error Interrupt

When the value of the dead time reference value register (DTUAR, DTDAR) is loaded into the general compare reference value register (GCMBR). If the period limit is exceeded, a dead time error will occur, and the STFLR.DTEF of the status flag register (STFLR) is set to 1. At this time, if the interrupt is enabled by setting the ICONR.INTENDE bit in the interrupt control register (ICONR), the dead time error interrupt will be triggered.

13.2.13 Brake Protection

With the invalid condition 0~3 setting and the configure of PCONR.DISVALA, PCONR.DISVALB ready, when the invalid condition is valid, the hardware automatically changes the port status to the preset state (high level, low level, high-impedance state or keeping normal output).

13.2.13.1 Port Brake and Software Brake

In this case, after the polarity selection control and enabling, it performs digital filtering and synchronized to generate the port brake flag. The port brake flag is used as the invalid condition 3 of the advanced timer. The port brake flag requires software removal.

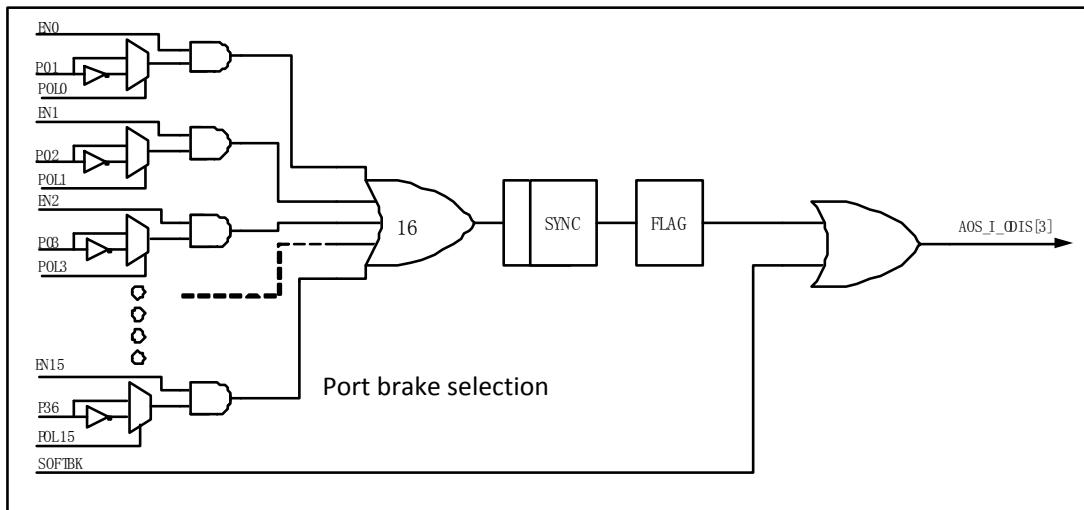


Figure 68. Port Brake and Software Brake Schematic

13.2.13.2 Low Power Mode Automatic Braking

In this case, the system enters low power mode and the PWM cannot work normally after the clock stops. The low power mode, as the invalid condition 2 of the advanced timer, is used to control the PWM brake.

13.2.13.3 Brake Per Output Level Both High or Both Low Case

In this case, after the output level monitoring and enabling, it is synchronized to generate the brake flag per output level both high or both low. The port brake flag is used as the invalid condition 1 of the advanced timer. This flag requires software removal.

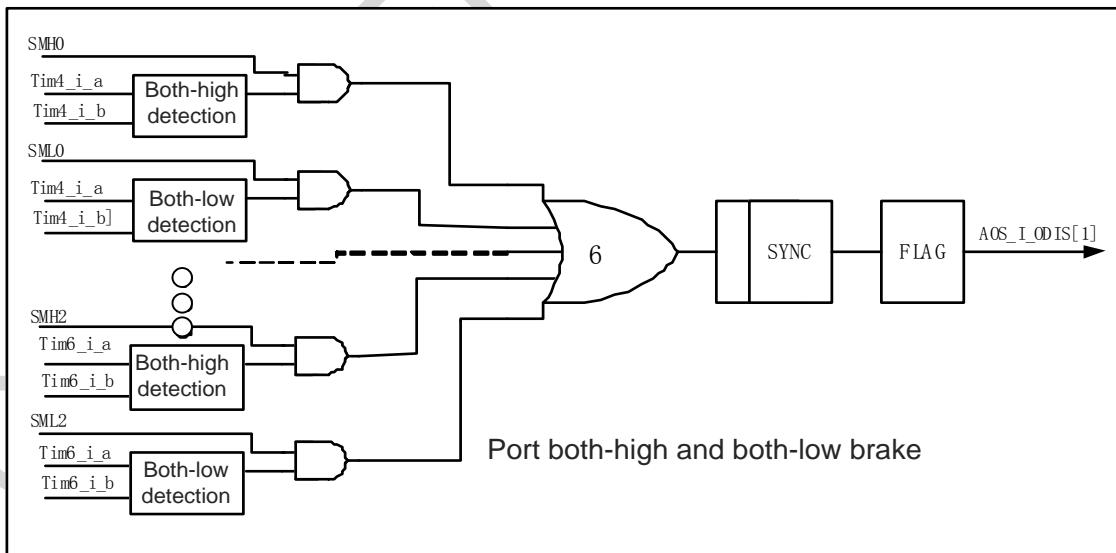


Figure 69. Schematic of The Brake Per Output Both High or Both Low Case

13.2.13.4 VC Brake

The VC1 and VC2 interrupt flags are enabled as the invalid condition 0 of the advanced timer.

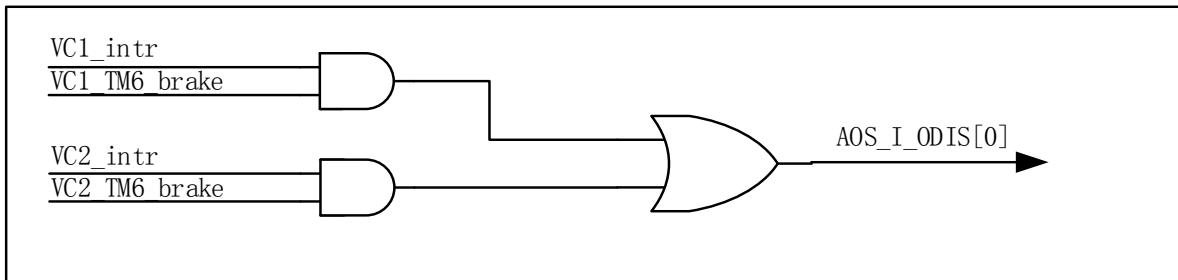


Figure 70. VC Brake Control Diagram

13.2.14 Internal Interconnect

13.2.14.1 Interrupt Triggering Output

A single interrupt of timer 4/5/6 can select multiple interrupt sources. The interrupt signal source controlling per ADC triggering and AOS is independent. The following interrupt sources can be selected as triggering conditions, including overflow, underflow, or any one of the interrupt sources of the 6 TIMxs among the 4 compare matches.

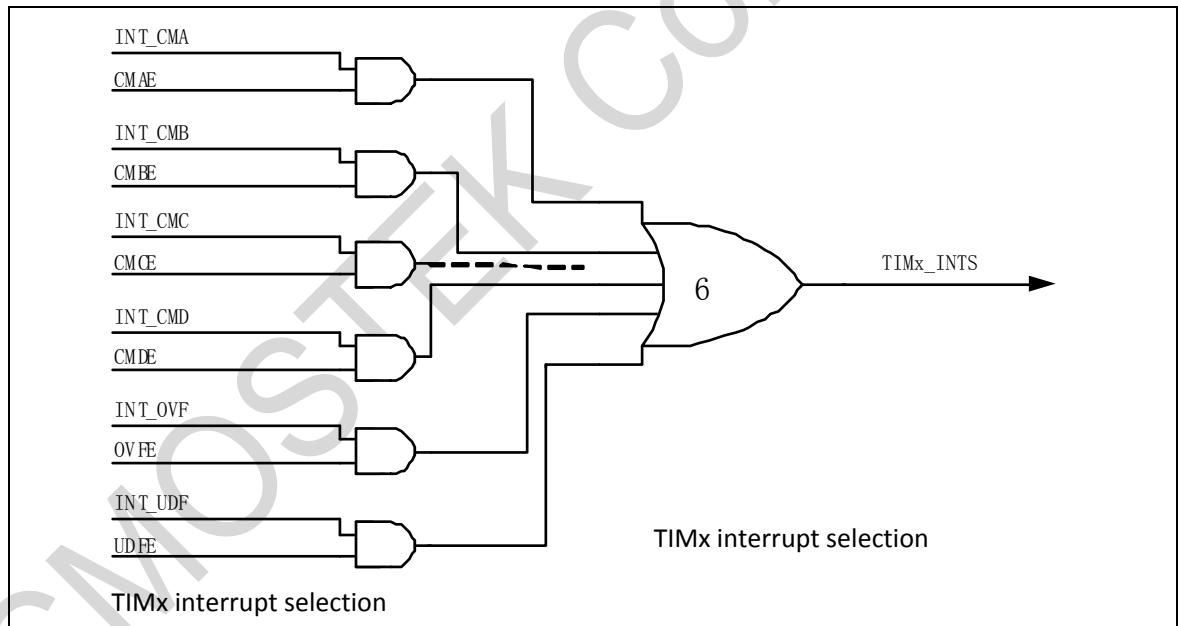


Figure 71. Timer 4/5/6 Interrupt Selection

13.2.14.2 AOS Triggering

AOS, an internal signal of the system, can trigger functions of advanced timer such as starting, stopping, clearing, increasing by 1, decreasing by 1, etc. The advanced timer contains 4 AOS triggers with each one supporting to select different modules as interrupt source. The selected signal generates single-pulse triggering input to the advanced timer to control the starting, stopping, and clearing of advanced timer counters.

Timer 4/5/6 internally uses registers to select different AOS_I_TRIGs as its own triggering signal. If the HSTAR register is available, it can use an interrupt to trigger the hardware start of the corresponding timer.

Table 427. AOS Source Selection

	AOS_i_trig0	AOS_i_trig1	AOS_i_trig2	AOS_i_trig3
Select Control Signal	ITRIG.IAOS0S	ITRIG.IAOS1S	ITRIG.IAOS2S	ITRIG.IAOS3S
0000	TIM0_INT	TIM0_INT	TIM0_INT	TIM0_INT
0001	TIM1_INT	TIM1_INT	TIM1_INT	TIM1_INT
0010	TIM2_INT	TIM2_INT	TIM2_INT	TIM2_INT
0011	LPTIMER_INT	LPTIMER_INT	LPTIMER_INT	LPTIMER_INT
0100	TIM4_INTS	TIM4_INTS	TIM4_INTS	TIM4_INTS
0101	TIM5_INTS	TIM5_INTS	TIM5_INTS	TIM5_INTS
0110	TIM6_INTS	TIM6_INTS	TIM6_INTS	TIM6_INTS
0111	UART0_INT	UART0_INT	UART0_INT	UART0_INT
1000	UART1_INT	UART1_INT	UART1_INT	UART1_INT
1001	UART2_INT	UART2_INT	UART2_INT	UART2_INT
1010	VC1_INT	VC1_INT	VC1_INT	VC1_INT
1011	VC2_INT	VC2_INT	VC2_INT	VC2_INT
1100	RTC_INT	RTC_INT	RTC_INT	RTC_INT
1101	PCA_INT	PCA_INT	PCA_INT	PCA_INT
1110	SPI_INT	SPI_INT	SPI_INT	SPI_INT
1111	ADC_INT	ADC_INT	ADC_INT	ADC_INT

13.2.14.3 Port Triggering TRIGA-TRIGD

For advanced timer, the port triggering can control the hardware starting, stopping, clearing, capture, counting-up and counting-down of counters. It has optional digital filtering function, and can be configured on any port of the chip.

Table 428. Port Triggering Selection

Control Signal Selection (independent control)	TRIGA	TRIGB	TRIGC	TRIGD
0000	P01	P01	P01	P01
0001	P02	P02	P02	P02
0010	P03	P03	P03	P03
0011	P15	P15	P15	P15
0100	P14	P14	P14	P14
0101	P23	P23	P23	P23

0110	P24	P24	P24	P24
0111	P25	P25	P25	P25
1000	P26	P26	P26	P26
1001	P27	P27	P27	P27
1010	P31	P31	P31	P31
1011	P32	P32	P32	P32
1100	P33	P33	P33	P33
1101	P34	P34	P34	P34
1110	P35	P35	P35	P35
1111	P36	P36	P36	P36

13.2.14.4 Interconnect Between Compare Output VC and Advanced Timer

The VC can be interconnected internally to the capture input of the advanced timer to capture the edge of the VC output.

13.2.14.5 Interconnect Between UART and Advanced Timer

UART0/1/2_RX can be internally interconnected to base timer, LPTimer, PCA and advanced timer. It can fulfill the automatic recognition of baud rate by software.

The UART selection control register is in the port control register GPIO_CTRL3, and the VC output control register is in the VC control module.

13.3 Register Description

CH0 base address: 0x40003000

CH1 base address: 0x40003400

CH2 base address: 0x40003800

Table 429. Advanced Timer Register List

Register	Offset Address	Description
TIMx_CNTER	0x000	General purpose count reference value register
TIMx_PERAR	0x004	General purpose period reference value register
TIMx_PERBR	0x008	General purpose period reference value buffer register
TIMx_GCMAR	0x010	General purpose compare A reference value register
TIMx_GCMBR	0x014	General purpose compare B reference value register
TIMx_GCMCR	0x018	General purpose compare C reference value register
TIMx_GCMDR	0x01C	General purpose compare D reference value register
TIMx_DTUAR	0x040	Dead time reference value register

Register	Offset Address	Description
TIMx_DTDAR	0x044	Dead time reference value register
TIMx_GCONR	0x050	General purpose control register
TIMx_ICONR	0x054	Interrupt control register
TIMx_PCONR	0x058	Port control register
TIMx_BCONR	0x05C	Buffer control register
TIMx_DCONR	0x060	Dead time control register
TIMx_FCONR	0x068	Filter control register
TIMx_VPERR	0x06C	Valid period register
TIMx_STFLR	0x070	Status flag register
TIMx_HSTAR	0x074	Hardware starting event selection register
TIMx_HSTPR	0x078	Hardware stopping event selection register
TIMx_HCLRR	0x07C	Hardware clearing event selection register
TIMx_HCPAR	0x080	Hardware capture event selection register
TIMx_HCPBR	0x084	Hardware capture event selection register
TIMx_HCUPR	0x088	Hardware count down event selection register
TIMx_HCDOR	0x08C	Hardware count down event selection register
TIMx_IFR	0x100	Interrupt flag register
TIMx_ICLR	0x104	Interrupt clearing register
TIMx_CR	0x108	Spread frequency and interrupt triggering selection register
TIMx_AOSSR	0x110	AOS selection register, shared by 3 channels
TIMx_AOSCL	0x114	AOS brake flag clearing register, shared by 3 channels
TIMx_PTBKSL	0x118	Port brake control register, shared by 3 channels
TIMx_TTRIG	0x11C	Port triggering control register, shared by 3 channels
TIMx_ITRIG	0x120	AOS triggering control register, shared by 3 channels
TIMx_PTBKPL	0x124	Port brake polarity control register, shared by 3 channels
TIMx_SSTAR	0x3F4	Software synchronization start register
TIMx_SSTPR	0x3F8	Software synchronization stop register
TIMx_SCLRR	0x3FC	Software synchronization clearing register

13.3.1 General Count Reference Value Register (TIMx_CNTER)

Offset address: 0x000

Reset value: 0x0000 0000

Table 430. General Count Reference Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 431. General Count Reference Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
R/W															

Table 432. General Count Reference Value Register (3)

Bit	Flag	Description
31:16	Reserved	-
15:0	CNT[15:0]	The count value of the current register

13.3.2 General Purpose Period Reference Value Register (TIMx_PERAR)

Offset address: 0x004

Reset value: 0x0000 FFFF

Table 433. General Purpose Period Reference Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 434. General Purpose Period Reference Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PERA[15:0]															
R/W															

Table 435. General Purpose Period Reference Value Register (3)

Bit	Flag	Description
31:16	Reserved	-
15:0	PERA[15:0]	Count period value. Set count period value for each count.

13.3.3 General Purpose Period Reference Value Buffer Register (TIMx_PERBR)

Offset address: 0x008

Reset value: 0x0000 FFFF

Table 436. General Purpose Period Reference Value Buffer Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 437. General Purpose Period Reference Value Buffer Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PERB[15:0]															
R/W															

Table 438. General Purpose Period Reference Value Buffer Register (3)

Bit	Flag	Description
31:16	Reserved	-
15:0	PERB[15:0]	Count period buffer value

13.3.4 General Purpose Compare Reference Value Register (TIMx_GCMAR-GCMDR)

Offset address: 0x0010, 0x0014, 0x0018, 0x001C

Reset value: 0x0000 FFFF

Table 439. General Purpose Compare Reference Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 440. General Purpose Compare Reference Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GCMA/GCMB/GCMC/GCMD[15:0]															
R/W															

Table 441. General Purpose Compare Reference Value Register (3)

Bit	Flag	Description
31:16	Reserved	-
15:0	GCMA-D [15:0]	Settings for count compare reference value and compare reference value. The matching signal is valid when the compare reference value is equal to the count value.

13.3.5 Dead Time Reference Value Register (TIMx_DTUAR- DTDAR)

Offset address: 0x040, 0x044

Reset value: 0x0000 FFFF

Table 442. Dead Time Reference Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 443. Dead Time Reference Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DTUA/DTDA[15:0]															
R/W															

Table 444. Dead Time Reference Value Register (3)

Bit	Flag	Description
31:16	Reserved	-
15:0	DTUA/DTDA [15:0]	Dead time setting value.

13.3.6 General Purpose Control Register (TIMx_GCONR)

Offset address: 0x050

Reset value: 0x000000100

Table 445. General Purpose Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										ZMSK	ZMSKPOS			ZMSKREV	
-										R/W	R/W			R/W	

Table 446. General Purpose Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				DIR		Res.		CKDIV			MODE			START	
-				R/W		-		R/W			R/W			R/W	

Table 447. General Purpose Control Register (3)

Bit	Flag	Description
31:20	Reserved	-
19:18	ZMSK	Z-phase input disabling period. The disabled count period value per quadrature encoding Z-phase input. 00: Z-phase input disabling function is invalid 01: Z-phase input is disabled 4 count cycles after the position count overflows or underflows. 10: Z-phase input is disabled 8 count cycles after the position count overflows or underflows. 11: Z-phase input is disabled 16 count cycles after the position count overflows or underflows.
17	ZMSKPOS	Z-phase input position counter selection. 0: this timer is used as the position counter during Z-phase input. The position counter clearing function operates normally during the disabling period. 1: this timer is used as the position counter during Z-phase input. The position counter clearing function operates is disabled during the disabling period.

Bit	Flag	Description
16	ZMSKREV	Z-phase input revolution counter selection. 0: this timer is used as the revolution counter during Z-phase input. The revolution counter count function operates normally during the disabling period. 1: this timer is used as the revolution counter during Z-phase input. The revolution counter count function operates is disabled during the disabling period.
15:9	Reserved	-
8	DIR	Count direction. 0: count down, 1: count up.
7	Reserved	-
6:4	CKDIV	Count clock selection. 000: PCLK0 001: PCLK0/2 010: PCLK0/4 100: PCLK0/16 101: PCLK0/64 110: PCLK0/256 111: PCLK0/1024
3:1	MODE	Count mode. 000: sawtooth wave mode A, 100: triangular wave mode A, 101: triangular wave mode B. Please do not set to other values.
0	START	Counter start. 0: the counter stops 1: counter starts Notes: this bit will automatically change to 0 when the software stopping condition or hardware stopping condition is valid.

13.3.7 Interrupt Control Register (TIMx_ICNR)

Offset address: 0x054

Reset value: 0x0000 0000

Table 448. Interrupt Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 449. Interrupt Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		INTENDE		INTENUDF		INTENOVF		Res.		INTED		INTEC		INTEB	
-		R/W		R/W		R/W		-		R/W		R/W		R/W	

Table 450. Interrupt Control Register (3)

Bit	Flag	Description
31:9	Reserved	-
8	INTENDE	Dead time error interrupt enabling. 0: The interrupt is disabled in case of dead time error. 1: The interrupt is enabled in case of dead time error.

Bit	Flag	Description
7	INTENUDF	Underflow interrupt enabling. 0: the interrupt is disabled when sawtooth wave underflows or triangle wave counts to valley point. 1: the interrupt is enabled when sawtooth wave underflows or triangle wave counts to valley point.
6	INTENOVF	Overflow interrupt enabling. 0: the interrupt is disabled when sawtooth wave overflows or triangle wave counts to valley point. 1: the interrupt is enabled when sawtooth wave overflows or triangle wave counts to valley point.
5:4	Reserved	-
3	INTEND	Count matching interrupt enabling D 0: The interrupt is disabled when the GCMDR register is equal to the count value. 0: The interrupt is enabled when the GCMDR register is equal to the count value.
2	INTENC	Count matching interrupt enabling C 0: The interrupt is disabled when the GCMDR register is equal to the count value. 0: The interrupt is enabled when the GCMDR register is equal to the count value.
1	INTENB	Count matching interrupt enabling B 0: The interrupt is disabled when the GCMDR register is equal to the count value. 0: The interrupt is enabled when the GCMDR register is equal to the count value.
0	INTENA	Count matching interrupt enabling A 0: The interrupt is disabled when the GCMDR register is equal to the count value. 0: The interrupt is enabled when the GCMDR register is equal to the count value.

13.3.8 Port Control Register (TIMx_PCONR)

Offset address: 0x058

Reset value: 0x0000 0000

Table 451. Port Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
Reserved			DISVALB		DISSELB		OUTENB		PERCB		CMPCB		STASTPS	STP	STAC	CAP
-			R/W		R/W		R/W		R/W		R/W		R/W	R/W	R/W	R/W

Table 452. Port Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved			DISVALA		DISSELLA		OUTENA		PERCA		CMPCA		STASTPS	STP	STAC	CAP
-			R/W		R/W		R/W		R/W		R/W		R/W	R/W	R/W	R/W

Table 453. Port Control Register (3)

Bit	Flag	Description
31:29	Reserved	-

Bit	Flag	Description
28:27	DISVALB	CHxB output status control 00: the CHxB port outputs normally if the selected forced output invalid condition 0~3 is satisfied. 01: the CHxB port outputs high impedance state if the selected forced output invalid condition 0~3 is satisfied. 10: the CHxB port outputs low level if the selected forced output invalid condition 0~3 is satisfied. 11: the CHxB port outputs high level if the selected forced output invalid condition 0~3 is satisfied.
26:25	DISSELB	Forced output invalid condition selection B. 00: select forced output invalid condition 0. 01: select forced output invalid condition 1 10: select forced output invalid condition 2. 11: select forced output invalid condition 3.
24	OUTENB	Output enabling B 0: CHxB port output is disabled when the advanced timer function is enabled. 1: CHxB port output is enabled when the advanced timer function is enabled.
23:22	PERCB	Port status setting B when the period value matches. 00: CHxB port output remains low level when the counter count value is equal to the period value. 01: CHxB port output is set as high level when the counter count value is equal to the period value. 10: CHxB port output is set as the previous state when the counter count value is equal to the period value. 11: CHxB port output is set as the reverse level when the counter count value is equal to the period value.
21:20	CMPCB	Port status setting B when the compare value matches. 00: CHxB port output remains low level when the counter count value is equal to GCMBR. 01: CHxB port output is set as high level when the counter count value is equal to GCMBR. 10: CHxB port output is set as the previous state when the counter count value is equal to GCMBR. 11: CHxB port output is set as the reverse level when the counter count value is equal to GCMBR.
19	STASTPSB	Port status selection B when count starts/stops. 0: When the count starts or stops, the CHxB port output is determined by STACB and STPCB. 1: When the count starts or stops, the CHxB port output is set to the previous state. Note: The start of the count here means that the count starts initially or starts again after stopping. The count stopping means that the count stops initially or stops after the count starts.
18	STPCB	Port status setting B when count stops. 0: CHxB port output is set to low level when count stops. 1: CHxB port output is set to high level when count stops
17	STACB	Port status setting B when count starts. 0: CHxB port output is set to low level when count starts. 1: CHxB port output is set to high level when count starts.
16	CAPCB	Function mode selection B. 0: compare output function 1: capture input function
15:13	Reserved	-

Bit	Flag	Description
12:11	DISVALA	CHxA output status control. 00: the CHxA port outputs normally if the selected forced output invalid condition 0~3 is satisfied. 01: the CHxA port outputs high impedance state if the selected forced output invalid condition 0~3 is satisfied. 10: the CHxA port outputs low level if the selected forced output invalid condition 0~3 is satisfied. 11: the CHxA port outputs high level if the selected forced output invalid condition 0~3 is satisfied.
10:9	DISSELA	Forced output invalid condition selection A. 00: select forced output invalid condition 0. 01: select forced output invalid condition 1 10: select forced output invalid condition 2. 11: select forced output invalid condition 3.
8	OUTENA	Output enabling A. 0: CHxB port output is disabled when the advanced timer function is enabled 1: CHxB port output is enabled when the advanced timer function is enabled
7:6	PERCA	Port status setting A when the period value matches. 00: CHxB port output remains low level when the counter count value is equal to the period value. 01: CHxB port output is set as high level when the counter count value is equal to the period value. 10: CHxB port output is set as the previous state when the counter count value is equal to the period value. 11: CHxB port output is set as the reverse level when the counter count value is equal to the period value.
5:4	CMPCA	Port status setting A when the compare value matches. 00: CHxB port output remains low level when the counter count value is equal to GCMBR. 01: CHxB port output is set as high level when the counter count value is equal to GCMBR. 10: CHxB port output is set as the previous state when the counter count value is equal to GCMBR. 11: CHxB port output is set as the reverse level when the counter count value is equal to GCMBR.
3	STASTPSA	Port status selection A when count starts/stops. 0: When the count starts or stops, the CHxA port output is determined by STACA and STPCA. 1: When the count starts or stops, the CHxA port output is set to the previous state. Note: The start of the count here means that the count starts initially or starts again after stopping. The count stopping means that the count stops initially or stops after the count starts.
2	STPCA	Port status setting A when count stops. 0: CHxA port output is set to low level when count stops. 1: CHxA port output is set to high level when count stops
1	STACA	Port status setting A when count starts. 0: CHxA port output is set to low level when count starts. 1: CHxA port output is set to high level when count starts.
0	CAPCA	Function mode selection A. 0: compare output function 1: capture input function

13.3.9 Buffer Control Register (TIMx_BCONR)

Offset address: 0x05C

Reset value: 0x0000 0000

Table 454. Buffer Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 455. Buffer Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				BENP		Reserved				BENB		Res.		BENA	
-				R/W		-				R/W		-		R/W	

Table 456. Buffer Control Register (3)

Bit	Flag	Description
31:9	Reserved	-
8	BENP	Period value buffer transfer. 0: buffer transfer is disabled 1: buffer transfer is enabled (PERBR->PERAR)
7:3	Reserved	-
2	BENB	General purpose compare value buffer transfer B. 0: buffer transfer is disabled 1: buffer transfer is enabled For compare output function: GCMDR->GCMBR. For capture input function: GCMBR->GCMDR.
1	Reserved	-
0	BENA	General purpose compare value buffer transfer B. 0: buffer transfer is disabled 1: buffer transfer is enabled For compare output function: GCMCR->GCMAR. For capture input function: GCMAR->GCMCR.

13.3.10 Dead Time Control Register (TIMx_DCONR)

Offset address: 0x060

Reset value: 0x0000 0000

Table 457. Dead Time Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 458. Dead Time Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							SEPA	Reserved							DTCEN
-							R/W	-							R/W

Table 459. Dead Time Control Register (3)

Bit	Flag	Description
31:9	Reserved	-
8	SEPA	Separate setting 0: set DTUAR and DTDAR separately 1: let the values of DTDAR and DTUAR be equal automatically
7:1	Reserved	-
0	DTCEN	Dead time function enabling. 0: dead time function is disabled 1: dead time function is enabled

13.3.11 Filtering Control Register (TIMx_FCONR)

Offset address: 0x068

Reset value: 0x0000 0000

Table 460. Filtering Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	NOFICKTD	NOFIENTD	Res.	NOFICKTC	NOFIENTC	Res.	NOFICKTB	NOFIENTB	Res.	NOFICKTA	Res.	NOFICKTA	NOFIENTA		
-	R/W	R/W	-	R/W	R/W	-	R/W	R/W	-	R/W	R/W	-	R/W	R/W	

Table 461. Filtering Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NOFICKTD	NOFIENTD	Res.	NOFICKTC	NOFIENTC	Res.	NOFICKGB	NOFIENGB	Res.	NOFICKGA	Res.	NOFICKGA	NOFIENGA		
-	R/W	R/W	-	R/W	R/W	-	R/W	R/W	-	R/W	R/W	-	R/W	R/W	

Table 462. Filtering Control Register (3)

Bit	Flag	Description
15	14	13
12	11	10
9	8	7
8	7	6
7	6	5
6	5	4
5	4	3
4	3	2
3	2	1
2	1	0

Bit	Flag	Description
31	Reserved	-
30:29	NOFICKTD[1:0]	TRID port filtering and sampling reference clock selection 00: PCLK0, 01: PCLK0/4, 10: PCLK0/16, 11: PCLK0/64.
28	NOFIENTD	TRID port capture input and filtering enabling, 0: disable 1: enable.
27	Reserved	-
26:25	NOFICKTC[1:0]	TRIC port filtering and sampling reference clock selection 00: PCLK0, 01: PCLK0/4, 10: PCLK0/16, 11: PCLK0/64.
24	NOFIENTC	TRIC port capture input and filtering enabling, 0: disable 1: enable
23	Reserved	-
22:21	NOFICKTB[1:0]	TRIB port filtering and sampling reference clock selection. 00: PCLK0, 01: PCLK0/4, 10: PCLK0/16, 11: PCLK0/64.
20	NOFIENTB	TRIB port capture input and filtering enabling, 0: disable 1: enable
19	Reserved	-
18:17	NOFICKTA[1:0]	TRIA port filtering and sampling reference clock selection. 00: PCLK0, 01: PCLK0/4, 10: PCLK0/16, 11: PCLK0/64.
16	NOFIENTA	TRIA port capture input and filtering enabling, 0: disable 1: enable.
15:7	Reserved	-
6:5	NOFICKGB[1:0]	CHxIB port filtering and sampling reference clock selection. 00: PCLK0, 01: PCLK0/4, 10: PCLK0/16, 11: PCLK0/64.
4	NOFIENGB	CHxIB port capture input and filtering enabling, 0: disable 1: enable
3	Reserved	-
2:1	NOFICKGA[1:0]	CHxIA port filtering and sampling reference clock selection 00: PCLK0, 01: PCLK0/4, 10: PCLK0/16, 11: PCLK0/64.
0	NOFIENGA	CHxIA port capture input and filtering enabling, 0: disable 1: enable.

Notes:

1. The TRIGA-D filtering setting is valid only in TIM4. It is invalid in the timer 5/6.

13.3.12 Valid Period Register (TIMx_VPERR)

Offset address: 0x06C

Reset value: 0x0000 0000

Table 463. Valid Period Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												PCNTS	PCNTE		
-												R/W	R/W		

Table 464. Valid Period Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												GEPERID	GEPERIC	GEPERIB	GEPERIA
-												R/W	R/W	R/W	R/W

Table 465. Valid Period Register (3)

Bit	Flag	Description
31:21	Reserved	-
20:18	PCNTS	<p>Valid period selection.</p> <p>000: valid period selection is disabled</p> <p>001: be valid every 1 cycle.</p> <p>010: be valid every 2 cycles.</p> <p>011: be valid every 3 cycles.</p> <p>100: be valid every 4 cycles.</p> <p>101: be valid every 5 cycles.</p> <p>110: be valid every 6 cycles.</p> <p>111: be valid every 7 cycles.</p>
17:16	PCNTE	<p>Valid periodical count condition selection.</p> <p>00: valid periodical count selection function is invalid.</p> <p>01: set the count condition as the overflow, underflow points of awtooth wave or the valley points of triangle wave</p> <p>10: set the count condition as the overflow, underflow points of awtooth wave or the peak points of triangle wave</p> <p>11: set the count condition as the overflow, underflow points of awtooth wave or the valley, peak points of triangle wave</p>
15:4	Reserved	-

3	GEPERID	General purpose signal valid period selection D 0: valid period selection function is disabled. 1: valid period selection function is enabled
2	GEPERIC	General purpose signal valid period selection C 0: valid period selection function is disabled. 1: valid period selection function is enabled
1	GEPERIB	General purpose signal valid period selection B 0: valid period selection function is disabled. 1: valid period selection function is enabled
0	GEPERIA	General purpose signal valid period selection A 0: valid period selection function is disabled. 1: valid period selection function is enabled

13.3.13 State Flag Register (TIMx_STFLR)

Offset address: 0x070

Reset value: 0x0000 0000

Table 466. State Flag Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DIR	Reserved										VPERNUM Reserved				
R	-										R				

Table 467. State Flag Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								DTEF	UDEF	OVFF	Reserved		CMD	CMC	CMB
-								R/W	R/W	R/W	-		R/W	R/W	R/W

Table 468. State Flag Register (3)

Bit	Flag	Description
31	DIRF	Count direction. 0: count-down 1: count-up
30:24	Reserved	-
23:21	VPERNUM	The number of periods. It means how many periods after the count, the valid period selection function is enabled.
20:9	Reserved	-
8	DTEF	Dead time error 0: no dead time error occurs, 1: dead time error occurs.
7	UDFF	Underflow matching. 0: no sawtooth wave underflow or triangle wave count to valley point occurs 1: sawtooth wave underflow or triangle wave count to valley point occurs

Bit	Flag	Description
6	OVFF	Overflow matching. 0: no sawtooth wave overflow or triangle wave peak point occurs 1: sawtooth wave overflow or triangle wave peak point occurs
27:26	Reserved	-
3	CMDF	Count matching D 0: the value of GCMDR register is not equal to the count value. 1: the value of GCMDR register is equal to the count value.
2	CMCF	Count matching C 0: the value of GCMCR register is not equal to the count value. 1: the value of GCMCR register is equal to the count value.
1	CMBF	Count matching B 0: the value of GCMBR register is not equal to the count value and no CHxB capture action completes. 1: the value of GCMBR register is equal to the count value or the CHxB capture action completes.
0	CMAF	Count matching A 0: the value of GCMAR register is not equal to the count value and no CHxA capture action completes. 1: the value of GCMAR register is equal to the count value or the CHxA capture action completes.

13.3.14 Hardware Start Event Selection Register (TIMx_HSTAR)

Offset address: 0x074

Reset value: 0x0000 0000

Table 469. Hardware Start Event Selection Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STARTS	Reserved														
R/W	-														

Table 470. Hardware Start Event Selection Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSTA15	HSTA14	HSTA13	HSTA12	HSTA11	HSTA10	HSTA9	HSTA8	HSTA7	HSTA6	HSTA5	HSTA4	HSTA3	HSTA2	HSTA1	HSTA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 471. Hardware Start Event Selection Register (3)

Bit	Flag	Description
31	STARTS	Hardware start enabling. 0: hardware start is disabled 1: hardware start is enabled Note: the SSTAR setting is disabled when hardware start is enabled.
30:16	Reserved	-

Bit	Flag	Description
15	HSTA15	Hardware start condition 15, sampling on the TIMTRID port at the falling edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
14	HSTA14	Hardware start condition 14, sampling on the TIMTRID port at the rising edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
13	HSTA13	Hardware start condition 13, sampling on the TIMTRIC port at the falling edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
12	HSTA12	Hardware start condition 12, sampling on the TIMTRIC port at the rising edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
11	HSTA11	Hardware start condition 11, sampling on the TIMTRIB port at the falling edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
10	HSTA10	Hardware start condition 10, sampling on the TIMTRIB port at the rising edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
9	HSTA9	Hardware start condition 9, sampling on the TIMTRIA port at the falling edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
8	HSTA8	Hardware start condition 8, sampling on the TIMTRIA port at the rising edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
7	HSTA7	Hardware start condition 7, sampling on the CHxB port at the falling edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
6	HSTA6	Hardware start condition 6, sampling on the CHxB port at the rising edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
5	HSTA5	Hardware start condition 5, sampling on the CHxA port at the falling edge. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
4	HSTA4	Hardware start condition 4, sampling on the CHxA port at the rising edge 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
3	HSTA3	Hardware start condition 3. Event triggering 3 from AOS is valid. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
2	HSTA2	Hardware start condition 2. Event triggering 2 from AOS is valid. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.

Bit	Flag	Description
1	HSTA1	Hardware start condition 1. Event triggering 1 from AOS is valid. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.
0	HSTA0	Hardware start condition 0. Event triggering 0 from AOS is valid. 0: hardware start is disabled when the condition is matched. 1: hardware start is enabled when the condition is matched.

13.3.15 Hardware Stopping Event Selection Register (TIMx_HSTPR)

Offset address: 0x078

Reset value: 0x0000 0000

Table 472. Hardware Stopping Event Selection Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STOP	Reserved														
R/W	-														

Table 473. Hardware Stopping Event Selection Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSTP15	HSTP14	HSTP13	HSTP12	HSTP11	HSTP10	HSTP9	HSTP8	HSTP7	HSTP6	HSTP5	HSTP4	HSTP3	HSTP2	HSTP1	HSTP0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 474. Hardware Stopping Event Selection Register (3)

Bit	Flag	Description
31	STOPS	Hardware stopping enabling. 0: hardware stopping is disabled. 1: hardware stopping is enabled. Note: the software stopping is disabled when the hardware stopping is enabled.
30:16	Reserved	-
15	HSTP15	Hardware stopping condition 15, sampling on the TIMTRID port at the falling edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
14	HSTP14	Hardware stopping condition 14, sampling on the TIMTRID port at the rising edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
13	HSTP13	Hardware stopping condition 13, sampling on the TIMTRIC port at the falling edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
12	HSTP12	Hardware stopping condition 12, sampling on the TIMTRIC port at the rising edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.

Bit	Flag	Description
11	HSTP11	Hardware stopping condition 11, sampling on the TIMTRIB port at the falling edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
10	HSTP10	Hardware stopping condition 10, sampling on the TIMTRIB port at the rising edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
9	HSTP9	Hardware stopping condition 9, sampling on the TIMTRIA port at the falling edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
8	HSTP8	Hardware stopping condition 8, sampling on the TIMTRIA port at the rising edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
7	HSTP7	Hardware stopping condition 7, sampling on the CHxB port at the falling edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
6	HSTP6	Hardware stopping condition 6, sampling on the CHxB port at the rising edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
5	HSTP5	Hardware stopping condition 5, sampling on the CHxA port at the falling edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
4	HSTP4	Hardware stopping condition 4, sampling on the CHxA port at the rising edge. 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
3	HSTP3	Hardware stopping condition 3. Event triggering 3 from AOS is valid 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
2	HSTP2	Hardware stopping condition 2. Event triggering 2 from AOS is valid 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
1	HSTP1	Hardware stopping condition 1. Event triggering 1 from AOS is valid 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.
0	HSTP0	Hardware stopping condition 0. Event triggering 0 from AOS is valid 0: hardware stopping is disabled when the condition is matched. 1: hardware stopping is enabled when the condition is matched.

13.3.16 Hardware Clearing Event Selection Register (TIMx_HCELR)

Offset address: 0x07C

Reset value: 0x0000 0000

Table 475. Hardware Clearing Event Selection Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLEARS	Reserved														
R/W	-														

Table 476. Hardware Clearing Event Selection Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HCEL15	HCEL14	HCEL13	HCEL12	HCEL11	HCEL10	HCEL9	HCEL8	HCEL7	HCEL6	HCEL5	HCEL4	HCEL3	HCEL2	HCEL1	HCEL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 477. Hardware Clearing Event Selection Register (3)

Bit	Flag	Description
31	STARTS	Hardware clearing enabling 0: hardware clearing is disabled 1: hardware clearing is enabled Note: the software clearing is disabled when the hardware clearing is enabled.
30:16	Reserved	-
15	HCEL15	Hardware clearing condition 15, sampling on the TIMTRID port at the falling edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
14	HCEL14	Hardware clearing condition 14, sampling on the TIMTRID port at the rising edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
13	HCEL13	Hardware clearing condition 13, sampling on the TIMTRIC port at the falling edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
12	HCEL12	Hardware clearing condition 12, sampling on the TIMTRIC port at the rising edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
11	HCEL11	Hardware clearing condition 11, sampling on the TIMTRIB port at the falling edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
10	HCEL10	Hardware clearing condition 10, sampling on the TIMTRIB port at the rising edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
9	HCEL9	Hardware clearing condition 9, sampling on the TIMTRIA port at the falling edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
8	HCEL8	Hardware clearing condition 8, sampling on the TIMTRIA port at the rising edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
7	HCEL7	Hardware clearing condition 7, sampling on the CHxB port at the falling edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.

Bit	Flag	Description
6	HCEL6	Hardware clearing condition 6, sampling on the CHxB port at the rising edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
5	HCEL5	Hardware clearing condition 5, sampling on the CHxA port at the falling edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
4	HCEL4	Hardware clearing condition 4, sampling on the CHxA port at the rising edge. 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
3	HCEL3	Hardware clearing condition 3. Event triggering 3 from AOS is valid 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
2	HCEL2	Hardware clearing condition 2. Event triggering 2 from AOS is valid 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
1	HCEL1	Hardware clearing condition 1. Event triggering 1 from AOS is valid 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.
0	HCELO	Hardware clearing condition 0. Event triggering 0 from AOS is valid 0: hardware clearing is disabled when the condition is matched. 1: hardware clearing is enabled when the condition is matched.

13.3.17 Hardware Capture A Event Selection Register (TIMx_HCPAR)

Offset address: 0x080

Reset value: 0x0000 0000

Table 478. Hardware Capture A Event Selection Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 479. Hardware Capture A Event Selection Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HCPA15	HCPA14	HCPA13	HCPA12	HCPA11	HCPA10	HCPA9	HCPA8	HCPA7	HCPA6	HCPA5	HCPA4	HCPA3	HCPA2	HCPA1	HCPA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 480. Hardware Capture A Event Selection Register (3)

Bit	Flag	Description
31:16	Reserved	-
15	HCPA15	Hardware capture A enabling. 0: hardware capture A is disabled. 1: hardware capture A is enabled. Note: the software capture A is disabled when the hardware capture A is enabled.

Bit	Flag	Description
14	HCPA14	Hardware capture A condition 15, sampling on the TIMTRID port at the falling edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
13	HCPA13	Hardware capture A condition 14, sampling on the TIMTRID port at the rising edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
12	HCPA12	Hardware capture A condition 13, sampling on the TIMTRIC port at the falling edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
11	HCPA11	Hardware capture A condition 12, sampling on the TIMTRIC port at the rising edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
10	HCPA10	Hardware capture A condition 11, sampling on the TIMTRIB port at the falling edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
9	HCPA9	Hardware capture A condition 10, sampling on the TIMTRIB port at the rising edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
8	HCPA8	Hardware capture A condition 9, sampling on the TIMTRIA port at the falling edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
7	HCPA7	Hardware capture A condition 8, sampling on the TIMTRIA port at the rising edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
6	HCPA6	Hardware capture A condition 7, sampling on the CHxB port at the falling edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
5	HCPA5	Hardware capture A condition 6, sampling on the CHxB port at the rising edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
4	HCPA4	Hardware capture A condition 5, sampling on the CHxA port at the falling edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
3	HCPA3	Hardware capture A condition 4, sampling on the CHxA port at the rising edge. 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
2	HCPA2	Hardware capture A condition 3. Event triggering 3 from AOS is valid 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.
1	HCPA1	Hardware capture A condition 2. Event triggering 2 from AOS is valid 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.

Bit	Flag	Description
0	HCPA0	Hardware capture A condition 1. Event triggering 1 from AOS is valid 0: hardware capture A is disabled when the condition is matched. 1: hardware capture A is enabled when the condition is matched.

13.3.18 Hardware Capture B Event Selection Register (TIMx_HCPBR)

Offset address: 0x084

Reset value: 0x0000 0000

Table 481. Hardware Capture B Event Selection Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 482. Hardware Capture B Event Selection Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HCPB15	HCPB14	HCPB13	HCPB12	HCPB11	HCPB10	HCPB9	HCPB8	HCPB7	HCPB6	HCPB5	HCPB4	HCPB3	HCPB2	HCPB1	HCPB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 483. Hardware Capture B Event Selection Register (3)

Bit	Flag	Description
31:16	Reserved	Hardware capture B enabling. 0: hardware capture B is disabled. 1: hardware capture B is enabled. Note: the software capture B is disabled when the hardware capture B is enabled.
15	HCPB15	-
14	HCPB14	Hardware capture B condition 15, sampling on the TIMTRID port at the falling edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
13	HCPB13	Hardware capture B condition 14, sampling on the TIMTRID port at the rising edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
12	HCPB12	Hardware capture B condition 13, sampling on the TIMTRIC port at the falling edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
11	HCPB11	Hardware capture B condition 12, sampling on the TIMTRIC port at the rising edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
10	HCPB10	Hardware capture B condition 11, sampling on the TIMTRIB port at the falling edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
9	HCPB9	Hardware capture B condition 10, sampling on the TIMTRIB port at the rising edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.

Bit	Flag	Description
8	HCPB8	Hardware capture B condition 9, sampling on the TIMTRIA port at the falling edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
7	HCPB7	Hardware capture B condition 8, sampling on the TIMTRIA port at the rising edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
6	HCPB6	Hardware capture B condition 7, sampling on the CHxB port at the falling edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
5	HCPB5	Hardware capture B condition 6, sampling on the CHxA port at the rising edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
4	HCPB4	Hardware capture B condition 5, sampling on the CHxA port at the falling edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
3	HCPB3	Hardware capture B condition 4, sampling on the CHxA port at the rising edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
2	HCPB2	Hardware capture B condition 3. Event triggering 3 from AOS is valid 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
1	HCPB1	Hardware capture B condition 2. Event triggering 2 from AOS is valid 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
0	HCPB0	Hardware capture B condition 1. Event triggering 1 from AOS is valid 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.

13.3.19 Hardware Count UP Register (TIMx_HCUPR)

Offset address: 0x088

Reset value: 0x0000 0000

Table 484. Hardware Count UP Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										HCUP19	HCUP18	HCUP17	HCUP16		
-										R/W	R/W	R/W	R/W		

Table 485. Hardware Count UP Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HCUP15	HCUP14	HCUP13	HCUP12	HCUP11	HCUP10	HCUP9	HCUP8	HCUP7	HCUP6	HCUP5	HCUP4	HCUP3	HCUP2	HCUP1	HCUP0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 486. Hardware Count UP Register (3)

Bit	Flag	Description
31:20	Reserved	-
19	HCUP19	Hardware count up condition. Event triggering 3 from AOS is valid 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
18	HCUP18	Hardware count up condition. Event triggering 2 from AOS is valid 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
17	HCUP17	Hardware count up condition. Event triggering 1 from AOS is valid 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
16	HCUP16	Hardware count up condition. Event triggering 0 from AOS is valid 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
15	HCUP15	Hardware count up condition, sampling on the TIMTRID port at the falling edge. 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
14	HCUP14	Hardware count up condition, sampling on the TIMTRID port at the rising edge. 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
13	HCUP13	Hardware count up condition, sampling on the TIMTRC port at the falling edge. 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
12	HCUP12	Hardware count up condition, sampling on the TIMTRIC port at the rising edge. 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
11	HCUP11	Hardware count up condition, sampling on the TIMTRIB port at the falling edge. 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
10	HCUP10	Hardware count up condition, sampling on the TIMTRIB port at the rising edge. 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
9	HCUP9	Hardware count up condition, sampling on the TIMTRIA port at the falling edge. 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
8	HCUP8	Hardware count up condition, sampling on the TIMTRIA port at the rising edge. 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
7	HCUP7	Hardware count up condition, sampling on the CHxA port at the falling edge when the CHxB port is high level 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.

Bit	Flag	Description
6	HCUP6	Hardware count up condition, sampling on the CHxA port at the rising edge when the CHxB port is high level 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
5	HCUP5	Hardware count up condition, sampling on the CHxA port at the falling edge when the CHxB port is low level 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
4	HCUP4	Hardware count up condition, sampling on the CHxA port at the rising edge when the CHxB port is low level 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
3	HCUP3	Hardware count up condition, sampling on the CHxB port at the falling edge when the CHxA port is high level 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
2	HCUP2	Hardware count up condition, sampling on the CHxB port at the rising edge when the CHxA port is high level 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
1	HCUP1	Hardware count up condition, sampling on the CHxB port at the falling edge when the CHxA port is low level 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.
0	HCUP0	Hardware count up condition, sampling on the CHxB port at the rising edge when the CHxA port is low level 0: hardware count up is disabled when the condition is matched. 1: hardware count up is enabled when the condition is matched.

13.3.20 Hardware Count Down Register (TIMx_HCDOR)

Offset address: 0x08C

Reset value: 0x0000 0000

Table 487. Hardware Count Down Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										HCDO19	HCDO18	HCDO17	HCDO16		
-										R/W	R/W	R/W	R/W		

Table 488. Hardware Count Down Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HCD01 5	HCD01 4	HCD01 3	HCD01 2	HCD01 1	HCD01 0	HCD0 9	HCD0 8	HCD0 7	HCD0 6	HCD0 5	HCD0 4	HCD0 3	HCD0 2	HCD0 1	HCD0 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 489. Hardware Count Down Register (3)

Bit	Flag	Description
31:20	Reserved	-
19	HCDO19	Hardware count down condition. Event triggering 3 from AOS is valid 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
18	HCDO18	Hardware count down condition. Event triggering 2 from AOS is valid 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
17	HCDO17	Hardware count down condition. Event triggering 1 from AOS is valid 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
16	HCDO16	Hardware count down condition. Event triggering 0 from AOS is valid 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
15	HCDO15	Hardware count down condition, sampling on the TIMTRID port at the falling edge 0: hardware capture B is disabled when the condition is matched. 1: hardware capture B is enabled when the condition is matched.
14	HCDO14	Hardware count down condition, sampling on the TIMTRID port at the rising edge 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
13	HCDO13	Hardware count down condition, sampling on the TIMTRC port at the falling edge 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
12	HCDO12	Hardware count down condition, sampling on the TIMTRIC port at the rising edge 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
11	HCDO11	Hardware count down condition, sampling on the TIMTRIB port at the falling edge 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
10	HCDO10	Hardware count down condition, sampling on the TIMTRIB port at the rising edge 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
9	HCDO9	Hardware count down condition, sampling on the TIMTRIA port at the falling edge 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.

Bit	Flag	Description
8	HCDO8	Hardware count down condition, sampling on the TIMTRIA port at the rising edge 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
7	HCDO7	Hardware count down condition, sampling on the CHxA port at the falling edge when the CHxB port is high level. 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
6	HCDO6	Hardware count down condition, sampling on the CHxA port at the rising edge when the CHxB port is high level. 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
5	HCDO5	Hardware count down condition, sampling on the CHxA port at the falling edge when the CHxB port is low level 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
4	HCDO4	Hardware count down condition, sampling on the CHxA port at the rising edge when the CHxB port is low level. 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
3	HCDO3	Hardware count down condition, sampling on the CHxB port at the falling edge when the CHxA port is high level 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
2	HCDO2	Hardware count down condition, sampling on the CHxB port at the rising edge when the CHxA port is high level. 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
1	HCDO1	Hardware count down condition, sampling on the CHxB port at the falling edge when the CHxA port is low level. 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.
0	HCDO0	Hardware count down condition, sampling on the CHxB port at the rising edge when the CHxA port is low level. 0: hardware count down is disabled when the condition is matched. 1: hardware count down is enabled when the condition is matched.

13.3.21 Software Synchronous Start Register (TIMx_SSTAR)

Offset address: 0x3F4

Reset value: 0x0000 0000

Table 490. Software Synchronous Start Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 491. Software Synchronous Start Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												SSTA2	SSTA1	SSTA0	
-												R/W	R/W	R/W	

Table 492. Software Synchronous Start Register (3)

Bit	Flag	Description
31:3	Reserved	
2	SSTA2	Timer 6 software start. 0: software start is disabled. 1: software start is enabled.
1	SSTA1	Timer 5 software start. 0: software start is disabled. 1: software start is enabled.
0	SSTA0	Timer 4 software start. 0: software start is disabled. 1: software start is enabled.

13.3.22 Software Synchronous Stopping Register (TIMx_SSTPR)

Offset address: 0x3F8

Reset value: 0x0000 0000

Table 493. Software Synchronous Stopping Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 494. Software Synchronous Stopping Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												SSTP2	SSTP1	SSTP0	
-												R/W	R/W	R/W	

Table 495. Software Synchronous Stopping Register (3)

Bit	Flag	Description
31:3	Reserved	
2	SSTP2	Timer 6 software start. 0: software stopping is disabled. 1: software stopping is enabled.
1	SSTP1	Timer 5 software start .

		0: software stopping is disabled. 1: software stopping is enabled.
0	SSTP0	Timer 4 software start. 0: software stopping is disabled. 1: software stopping is enabled.

13.3.23 Software Synchronous Clearing Register (TIMx_SCLRR)

Offset address: 0x3FC

Reset value: 0x0000 0000

Table 496. Software Synchronous Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 497. Software Synchronous Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												SCLR2	SCLR1	SCLR0	
-												R/W	R/W	R/W	

Table 498. Software Synchronous Clearing Register (3)

Bit	Flag	Description
31:3	Reserved	
2	SCLR2	Timer 6 software clearing. 0: software clearing is disabled. 1: software clearing is enabled.
1	SCLR1	Timer 5 software clearing. 0: software clearing is disabled. 1: software clearing is enabled.
0	SCLR0	Timer 4 software clearing. 0: software clearing is disabled. 1: software clearing is enabled.

13.3.24 Interrupt Flag Register (TIMx_IFR)

Offset address: 0x100

Reset value: 0x0000 0000

Table 499. Interrupt Flag Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 500. Interrupt Flag Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

SAMHF	SAMLF	Reserved	DTEF	UDFF	OVFF	Reserved	CMDF	CMCF	CMBF	CMAF
RO	RO	-	RO	RO	RO	-	RO	RO	RO	RO

Table 501. Interrupt Flag Register (3)

Bit	Flag	Description
31:16	Reserved	-
15	SAMHF	CHxA/B port high level state interrupt flag. 0: no high level occurs on both CHxA and CHxB ports. 1: high level occurs on both CHxA and CHxB ports.
14	SAMLF	CHxA/B port low level state interrupt flag. 0: no low level occurs on both CHxA and CHxB ports. 1: low level occurs on both CHxA and CHxB ports.
13:9	Reserved	-
8	DTEF	Dead time error interrupt flag. 0: no dead time error occurs, 1: dead time error occurs.
7	UDFF	Underflow matching interrupt flag. 0: no sawtooth wave overflow or triangle wave count to valley point occurs. 1: sawtooth wave overflow or triangle wave count to valley point occurs.
6	OVFF	Overflow matching interrupt flag. 0: no sawtooth wave overflow or triangle wave count to peak point occurs. 1: sawtooth wave overflow or triangle wave count to peak point occurs.
5:4	Reserved	-
3	CMDF	Count matching D interrupt flag. 0: the value of the GCMDR register is not equal to the count value. 1: the value of the GCMDR register is equal to the count value.
2	CMCF	Count matching C interrupt flag. 0: the value of the GCMCR register is not equal to the count value. 1: the value of the GCMCR register is equal to the count value.
1	CMBF	Count matching B interrupt flag. 0: the value of the GCMBR register is not equal to the count value. 1: the value of the GCMBR register is equal to the count value.
0	CMAF	Count matching A interrupt flag. 0: the value of the GCMAR register is not equal to the count value and no CHxA capture action completes. 1: the value of the GCMAR register is equal to the count value and the CHxA capture action completes.

13.3.25 Interrupt Flag Clearing Register (TIMx_ICLR)

Offset address: 0x104

Reset value: 0x0000 0000

Table 502. Interrupt Flag Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 503. Interrupt Flag Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SAMHC	SAMLC	Reserved				DTEC	UDFC	OVFC	Reserved		CMDC	CMCC	CMBC	CMAC	
WO	WO	-				WO	WO	WO	-		WO	WO	WO	WO	

Table 504. Interrupt Flag Clearing Register (3)

Bit	Flag	Description
31:16	Reserved	-
15	SAMHC	CHxA/B port high level status interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.
14	SAMLC	CHxA/B port low level status interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.
13:9	Reserved	-
8	DTEC	Dead time error interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.
7	UDFC	Underflow interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.
6	OVFC	Overflow interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.
5:4	Reserved	-
3	CMDC	Count matching D interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.
2	CMCC	Count matching C interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.
1	CMBC	Count matching B interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.
0	CMAC	Count matching A interrupt flag clearing. Setting to 1 is invalid. Setting to 0 is for clearing the corresponding interrupt.

13.3.26 Spread Frequency and Interrupt Triggering Selection (TIMx_CR)

Offset address: 0x108

Reset value: 0x0000 0000

Table 505. Spread Frequency and Interrupt Triggering Selection (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 506. Spread Frequency and Interrupt Triggering Selection (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					DITENS	DITENB	DITENA	UDFE	OVFE	Reserved	CMDE	CMCE	CMBE	CMAE	
-					R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W

Table 507. Spread Frequency and Interrupt Triggering Selection (3)

Bit	Flag	Description
31:11	Reserved	-
10	DITENS	PWM spread frequency count selection. 0: select underflow, 1: select overflow.
9	DITENB	PWM channel B spread frequency enabling. 0: the enabling is enabled, 1: the enabling is disabled and PWM output delay changes each cycle.
9	DITENA	PWM channel A spread frequency enabling. 0: the enabling is enabled, 1: the enabling is disabled and PWM output delay changes each cycle.
7	UDFE	Underflow matching enabling triggering ADC. 0: the enabling is enabled, 1: enabling is disabled and the interrupt can control ADC/AOS_i_tirg.
6	OVFE	Overflow matching enabling triggering ADC. 0: the enabling is enabled, 1: enabling is disabled and the interrupt can control ADC/AOS_i_tirg.
5:4	Reserved	-
3	CMDE	Count matching D enabling triggering ADC. 0: the enabling is enabled, 1: enabling is disabled and the interrupt can control ADC/AOS_i_tirg.
2	CMCE	Count matching C enabling triggering ADC. 0: the enabling is enabled, 1: enabling is disabled and the interrupt can control ADC/AOS_i_tirg.
1	CMBE	Count matching B enabling triggering ADC. 0: the enabling is enabled 1: enabling is disabled and the interrupt can control ADC/AOS_i_tirg.
0	CMAE	Count matching A enabling triggering ADC. 0: the enabling is enabled, 1: enabling is disabled and the interrupt can control ADC/AOS_i_tirg.

13.3.27 AOS Selection Control Register (TIMx_AOSSR)

Offset address: 0x110

Reset value: 0x0000 0000

Timer 4/5/6 share the same physical register. When anyone of them is changed, the values of the other two timers will be changed at the same time.

Table 508. AOS Selection Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 509. AOS Selection Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	SMH2	SMH1	SMH0	SML2	SML1	SML0	SOFTB	Reserved	BFILTEN	BFILTS	FSAME	FBRAKE			

-	R/W	-	R/W	R/W	R	R						
---	-----	-----	-----	-----	-----	-----	-----	---	-----	-----	---	---

Table 510. AOS Selection Control Register (3)

Bit	Flag	Description
31:14	Reserved	-
13	SMH2	Channel 2 both high level selection. 0: the selection is disabled, 1: the selection is enabled. Set accurs AOS_i_odis[1] when both high.
12	SMH1	Channel 1 both high level selection. 0: the selection is disabled 1: the selection is enabled. Set accurs AOS_i_odis[1] when both high.
11	SMH0	Channel 0 both high level selection. 0: the selection is disabled 1: the selection is enabled. Set accurs AOS_i_odis[1] when both high.
10	SML2	Channel 2 both low level selection. 0: the selection is disabled 1: the selection is enabled. Set accurs AOS_i_odis[1] when both high.
9	SML1	Channel 1 both low level selection. 0: the selection is disabled 1: the selection is enabled. Set accurs AOS_i_odis[1] when both high.
8	SML0	Channel 0 both low level selection. 0: the selection is disabled 1: the selection is enabled. Set accurs AOS_i_odis[1] when both high.
7	SOFTBK	Software brake. Setting to 1 is for software brake.
13	Reserved	-
4	BFILTEN	Port brake filtering enabling.
3:2	BFILTS	Port brake filtering clock selection.
1	FSAME	Both high level and both low level brake flag, read only.
0	FBRAKE	Port brake flag, read only.

13.3.28 AOS Selection Control Register Flag Clearing (TIMx_AOSCL)

Offset address: 0x114

Reset value: 0x0000 0000

Timer 4/5/6 share the same physical register. When anyone of them is changed, the values of the other two timers will be changed as well.

Table 511. AOS Selection Control Register Flag Clearing (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 512. AOS Selection Control Register Flag Clearing (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														FSAME	F BRAKE
-														R	R

Table 513. AOS Selection Control Register Flag Clearing (3)

Bit	Flag	Description
31:2	Reserved	-
1	FSAME	Both high level and both low level brake flag clearing. Set to 0 for clearing. It's invalid when set to 1. It's read as 1 always.
0	F BRAKE	The port brake flag clearing. Set to 0 for clearing. It's invalid when set to 1. It's read as 1 always.

13.3.29 Port Brake Control Register (TIMx_PTAKS)

Offset address: 0x118

Reset value: 0x0000 0000

Timer 4/5/6 share the same physical register. When anyone of them is changed, the values of the other two timers will be changed as well.

Table 514. Port Brake Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 515. Port Brake Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 516. Port Brake Control Register (3)

Bit	Flag	Description
31:16	Reserved	-
15	EN15	P36 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
14	EN14	P35 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
13	EN13	P34 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
12	EN12	P33 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
11	EN11	P32 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
10	EN10	P31 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
9	EN9	P27 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
8	EN8	P26 brake port enabling. 1: the selection is enabled 0: the selection is disabled.

7	EN7	P26 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
6	EN6	P24 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
5	EN5	P23 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
4	EN4	P14 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
3	EN3	P15 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
2	EN2	P03 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
1	EN1	P02 brake port enabling. 1: the selection is enabled 0: the selection is disabled.
0	EN0	P01 brake port enabling. 1: the selection is enabled 0: the selection is disabled.

13.3.30 Port Triggering Control Register (TIMx_TTRIG)

Offset address: 0x11C

Reset value: 0x0000 0000

Timer 4/5/6 share the same physical register. When anyone of them is changed, the values of the other two timers will be changed as well.

Table 517. Port Triggering Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 518. Port Triggering Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGDS				TRIGCS				TRIGBS				TRIGAS			
R/W				R/W				R/W				R/W			

Table 519. Port Triggering Control Register (3)

Bit	Flag	Description
31:16	Reserved	-
15:12	TRIGDS	TIMx triggering D port selection
11:8	TRIGCS	TIMx triggering C port selection
7:4	TRIGBS	TIMx triggering B port selection
3:0	TRIGAS	TIMx triggering A port selection

The control signal and port selection is listed in the below table.

Table 520. Control Signal and Port Selection

0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
P01	P02	P03	P15	P14	P23	P24	P25	P26	P27	P31	P32	P33	P34	P35	P36

13.3.31 AOS Triggering Control Register (TIMx_ITRIG)

Offset address: 0x120

Reset value: 0x0000 0000

Timer 4/5/6 share the same physical register. When anyone of them is changed The values of the other two timers will be changed as well.

Table 521. AOS Triggering Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 522. AOS Triggering Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IAOS3S				IAOS2S				IAOS1S				IAOS0S			
R/W				R/W				R/W				R/W			

Table 523. AOS Triggering Control Register (3)

Bit	Flag	Description
31:16	Reserved	-
15:12	IAOS3S	TIMx AOS3 triggering source selection.
11:8	IAOS2S	TIMx AOS2 triggering source selection.
7:4	IAOS1S	TIMx AOS1 triggering source selection.
3:0	IAOS0S	TIMx AOS0 triggering source selection.

Control signal (IAOSxS, x = 0,1,2,3) and interrupt source selection is listed in the below table.

Table 524. Interrupt Source Selection

0000	0001	0010	0011	0100	0101	0110	0111
TIM0_INT	TIM1_INT	TIM2_INT	LPTIMER_INT	TIM4_INTS	TIM5_INTS	TIM6_INTS	UART0_INT
1000	1001	1010	1011	1100	1101	1110	1111
UART1_INT	UART2_INT	VC1_INT	VC2_INT	RTC_INT	PCA_INT	SPI_INT	ADC_INT

13.3.32 Port Brake Polarity Control Register (TIMx_PTBKPx)

Offset address: 0x124

Reset value: 0x0000 0000

Timer 4/5/6 share the same physical register. When anyone of them is changed, the values of the other two timers will be changed as well.

Table 525. Port Brake Polarity Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 526. Port Brake Polarity Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POL1 5	POL1 4	POL1 3	POL1 2	POL1 1	POL1 0	POL 9	POL 8	POL 7	POL 6	POL 5	POL 4	POL 3	POL 2	POL 1	POL 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 527. Port Brake Polarity Control Register (3)

Bit	Flag	Description
31:16	Reserved	-
15	POL15	P36 brake port polarity control. 1: active low, 0: active high.
14	POL14	P35 brake port polarity control. 1: active low, 0: active high.
13	POL13	P34 brake port polarity control. 1: active low, 0: active high.
12	POL12	P33 brake port polarity control. 1: active low, 0: active high.
11	POL11	P32 brake port polarity control. 1: active low, 0: active high.
10	POL10	P31 brake port polarity control. 1: active low, 0: active high.
9	POL9	P27 brake port polarity control. 1: active low, 0: active high.
8	POL8	P26 brake port polarity control. 1: active low, 0: active high.
7	POL7	P26 brake port polarity control. 1: active low, 0: active high.
6	POL6	P24 brake port polarity control. 1: active low, 0: active high.
5	POL5	P23 brake port polarity control. 1: active low, 0: active high.
4	POL4	P14 brake port polarity control. 1: active low, 0: active high.
3	POL3	P15 brake port polarity control. 1: active low, 0: active high.
2	POL2	P03 brake port polarity control. 1: active low, 0: active high.
1	POL1	P02 brake port polarity control. 1: active low, 0: active high.
0	POL0	P01 brake port polarity control. 1: active low, 0: active high.

14 Real-time Clock (RTC)

14.1 Real-time Clock Introduction

The real-time clock/calendar provides a time of day information on seconds, minutes, hours, days, weeks, months, and years. The number of days per month and the number of days in leap years are automatically adjusted. The clock determines 24- or 12-hour format by the AM/PM register. The basic features are listed in the below table.

Table 528. RTC Basic Features

Clock Source	Off-chip low speed crystal LXT (32.768 kHz). On-chip low speed oscillator LRC (32 kHz with an precision of 1%). Off-chip high speed crystal oscillator HXT.
Basic Function	Support calculating the seconds, minutes, hours, days, weeks, months, years in 00 ~ 99 years.
	Support automatic leap year adjustment.
	Support configurable in 24- or 12-hour format.
	Support programmable control of start or stopping.
	Support alarm clock function.
	Support high precision 1 Hz square wave signal output.
Interrupt	Support interrupt with periodic interrupt.
	Support alarm interrupt.

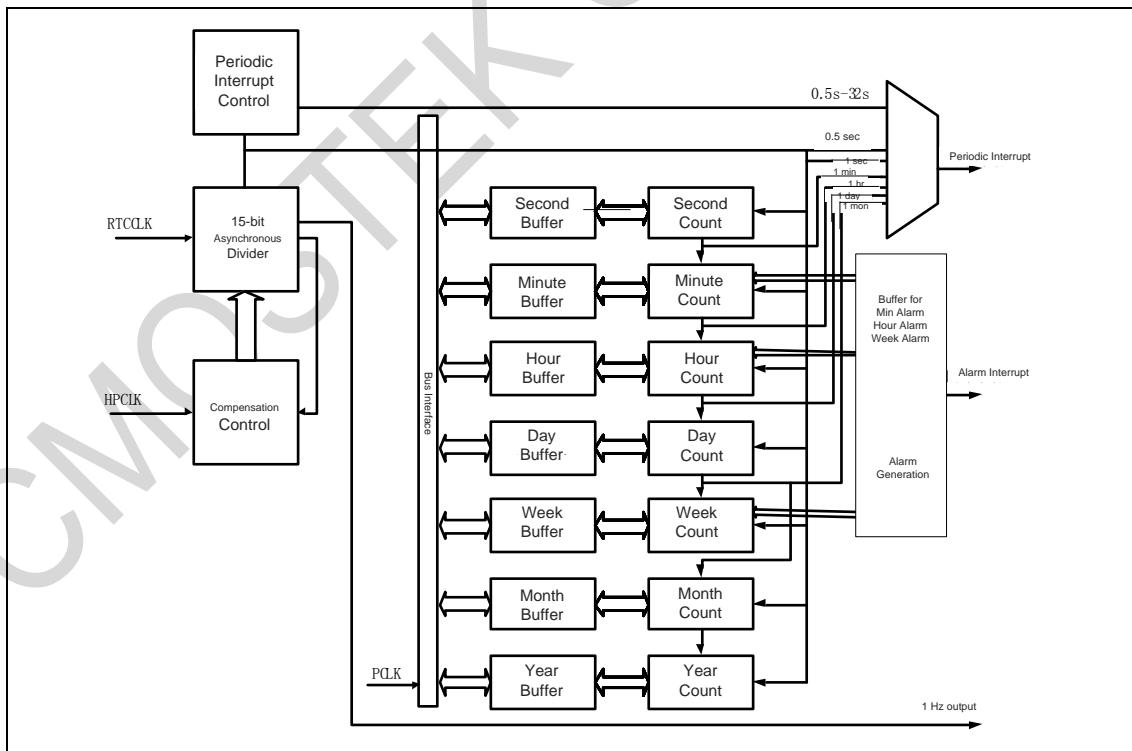


Figure 72. RTC Block Diagram Block Diagram

14.2 Real-time Clock Function Description

The clock source of the real-time clock can be configured as external low speed crystal, external high speed crystal, or internal low speed RC (as default). Control registers CR0, CR1, and COMPEN are controlled by power-on reset only, meaning that other reset sources cannot reset these 3 control registers. The state of other data registers during power-up is uncertain, therefore an Initialization process is required after power-up. These registers are not affected by reset.

All the date time values written and read by software are BCD codes , no need for conversion from hexadecimal to decimal.

Any invalid date time can not be written in, like writing days as 32, hours as 25, seconds as 70, months as B, etc.

14.2.1 Power-on Settings

RTC is reset once after power-on. RCT cannot be reset by various external reset requests and keeps counting as long as the system is not powered down. After power-on, RTC starts when the settings of calendar initial value, alarm, deviation compensation, interrupt, etc. complete.

14.2.2 RTC Count Start Setting

1. Set CR0.START as 0, then the count stops.
2. Set CR0.AMPM and CR0.PRDS, and set CR0.PRDX for time format and interrupt period configuration.
3. Set CR1.CKSEL to select the RTC clock.
4. Set the calendar count register for seconds, minutes, hours, weeks, days, months, and years.
5. When clock deviation compensation is required, set the count clock deviation compensation register COMPEN.
6. Clear the interrupt flag bits CR1.ALMF and CR1.PRDF, and enable the interrupt.
7. Set CR0.START as 1, then the count starts.

14.2.3 System Low Power Mode Switching

After the RTC count starts, if the system needs to switch to the low power mode immediately, please ensure to perform any of the following items before switching (the control register is in the system control register SYSCTRL1.RTC_LPW).

1. After set CR0.START to 1, perform switching after two or more RTC count clocks.
2. After set CR0.START to 1, set CR1.WAIT to 1 and query CR1.WAITF=1. Then set CR1.WAIT to 0 and query CR1.WAITF=0. Perform switching after then.

In RTC low power mode, the RTC register cannot be read or written. In low power mode, the RTC consumes less current.

It does not require waiting when switching to low power mode during the runtime of RTC.

14.2.4 Reading Count Register

It supports 3 modes to read the count register.

- **Mode 1 (reading at any time)**

1. Set CR1.WAIT to 1, stop the calendar register count and enter the read/write mode.
2. Keep querying until it meets CR1.WAITF=1.
3. Read the seconds, minutes, hours, weeks, days, months, and year count register values.
4. Set CR1.WAIT to 0, the counter counts.
5. Keep querying until it meets CR1.WAITF=0.

- **Mode 2 (reading at any time)**

1. Read the minute, hour, week, day, month, year count register value.
2. Read the second count register value.
3. Read the second count register value again.
4. Judge whether the values of seconds obtained by the 2 read operations are the same. If the values are different, repeat step 1, otherwise the read operation ends.

- **Interrupt reading mode**

Read the seconds, minutes, hours, weeks, days, months, and year count register values in the RTC period interrupt service. The data retains for at least a half second between the interrupt occurring time to the time when the next data change occurs.

14.2.5 Writing Count Register

1. Set CR1.WAIT to 1, stop the calendar register count and enter the read/write mode.
2. Keep querying until it meets CR1.WAITF=1.
3. Read the seconds, minutes, hours, weeks, days, months, and year count register values.
4. Set CR1.WAIT to 0, the counter counts again. Please be noted that all writing operation must be completed within 1 second
5. Keep querying until it meets CR1.WAITF = 0.

It's not necessary to wait for WAIT when writing the seconds, minutes, hours, weeks, days, months, and years count registers if RTC does not run.

Note:

1. Changing the seconds register in count mode will reset the seconds count. Writing the value of count registers for minutes, hours, weeks, days, months, and years will not affect RTC count though.

14.2.6 Alarm Settings

1. Set CR1.ALMEN to 0, the alarm is disabled then.
2. Set CR1.ALMIIE to 1, the alarm interrupt is enabled then.
3. Set the alarm clock per minutes (ALMMIN), alarm clock per hours (ALMHOUR) and alarm clock per weeks (ALMWEEK).
4. Set CR1.ALMEN to 1, the alarm interrupt is enabled then.
5. Wait for an interrupt.
6. Since the alarm interrupt and the fixed periodic interrupt share the interrupt request signal, when it meets CR1.ALMF = 1, the alarm interrupt processing is performed, otherwise, the fixed periodic interrupt processing is performed.

14.2.7 1 Hz Output

RTC can choose to output 3 kinds of 1 Hz clocks with general precision, higher precision and high precision. A higher precision 1 Hz clock is output when the clock deviation compensation function is enabled. A high precision 1 Hz clock is output when PCLKs with different frequencies are used. The system control register needs to be configured according to the PCLK frequency as follows.

- **Settings for 1 Hz output with general precision (no clock compensation)**

1. Set CR0.START to 0, the count stops then.
2. Set the RTC output pin.
3. Set CR0.1HZOE to 1, clock output is enabled then.
4. Set CR0.START to 1, the count starts then.
5. Wait for 2 counting cycles or more.
6. The 1 Hz output starts.

- **Settings for 1 Hz output with higher precision (low speed compensation)**

1. Set CR0.START to 0, the count stops.
2. Set the RTC output pin.
3. Set CR0.1HZOE to 1, clock output is enabled.
4. Set the compensation value to the clock deviation compensation register COMPEN.CR.
5. Set the clock deviation compensation register COMPEN.EN to 1, the error compensation is enabled.
6. Set CR0.START to 1, the count starts.
7. Wait for 2 counting cycles or more.
8. The 1 Hz output starts.

- **Settings for 1 Hz output with high precision** (providing 4/6/8/12/16/20/24/32 MHz high speed PCLK clock for RTC with higher precision output)

1. Set CR0.START to 0, the count stops.
2. Set the RTC output pin.
3. Set CR0.1HZOE to 1, clock output is enabled.
4. Set CR0.1HZSEL to 1, choose to output high precision 1 Hz clock.
5. Set high speed compensation clock, SYSCTRL1.RTC_FREQ_ADJUST
6. Set the compensation value to the clock deviation compensation register COMPEN.CR[8:0].
7. Set the clock deviation compensation register COMPEN.EN to 1, the deviation compensation is enabled.
8. Set CR0.START to 1, the count starts.
9. Wait for 2 counting cycles or more.
10. The 1 Hz output starts.

14.2.8 Clock Deviation Compensation

Since there is deviation in the external crystal oscillator, it is necessary to have deviation compensation in case of high-precision requirements. 2 compensation methods are available: 1) the deviation compensation based on the clock of itself. 2) the deviation compensation based on the high speed clock.

- **Method 1, the principle and calculation to compensate deviation of the clock itself**

Since the counter uses 32.768 kHz clock count, if it requires compensation per second based accuracy, it can only compensate according to the integer part of 32.768 kHz, thus the minimum unit of compensation per second is $(1/32768)*10^6 = 30.5$ ppm, which fails in satisfying high-precision requirements.

To achieve higher-precision clock compensation on the 32.768 kHz count clock, an algorithm adjustment is required to enlarge the maximum compensation period by 32 times. Thus, in the case of a minimum compensation unit of 30.5 ppm, the averaged

compensation unit per second becomes $30.5 \text{ ppm} / 32 = 0.96 \text{ ppm}$, which can meet the high-precision clock compensation requirements well. Moreover, the compensations are evenly distributed in every 32 seconds. Therefore, a setting of 5 decimal places is introduced in this register.

Example 1:

When the 1 Hz clock is directly output by default, the target compensation value is calculated by measuring the accuracy of the clock. Assuming the actual measured value is 0.9999888 Hz, then the calculation is as follows.

$$\text{Actual vibration frequency} = 32768 \times 0.9999888 \approx 32767.63$$

$$\begin{aligned}\text{Target compensation value} &= (\text{actual vibration frequency} - \text{target frequency}) / \text{target frequency} \times 10^6 \\ &= (32767.96 - 32768) / 32768 \times 10^6 \\ &= -11.29 \text{ ppm}\end{aligned}$$

According to the formula below,

$$\text{CR[8:0]} = \left(\frac{\text{target compensation value [ppm]} \times 2^{15}}{10^6} \right) \text{ taking 2's complement} + 0001.00000B$$

As an example, If the target compensation value is -11.29 ppm, the calculation of the corresponding register value is as follows.

$$\begin{aligned}\text{CR[8:0]} &= (-11.29 \times 2^{15}/10^6) \text{ taking 2's complement} + 0001.00000B \\ &= (-0.37) \text{ taking 2's complement} + 0001.00000B \\ &= 1111.10101B + 0001.00000B \\ &= 0000.10101B\end{aligned}$$

- Method 2, the principle and calculation to compensate deviation of the high speed 24 MHz clock**

The calculation method is the same as the deviation compensation of clock itself. Due to the introduction of a 4 - 32 MHz high speed clock, the original deviation of 1/32768 second distributed in every 32 seconds in method 1 spreads in every one second in method 2 with a minimum of 0.96 ppm (23 clock cycles at 24 MHz) compensation per 1 second, to fulfill an average high-precision (per each second) 1 Hz clock output.

14.3 RTC Interrupt

RTC supports 2 types of interrupts, the alarm interrupt and fixed periodic interrupt. They share the same interrupt signal.

14.3.1 RTC Alarm Interrupt

When CR1.ALMIE = 1, the alarm interrupt is triggered if the current calendar time is equal to the alarm clock register ALMMIN, the alarm clock register ALMHOUR, and the weekly alarm register (ALMWEEK).

14.3.2 RTC Periodic Interrupt

If ALMIE = 1 in the control register 1 (CR1), when the selected period being met, the fixed periodic wakeup interrupt is triggered. The alarm function and the fixed periodic function share the interrupt, which are distinguished by the flag register bits.

14.4 RTC Register Description

Base address: 0X40001400

Table 529. Register List

Register	Offset Address	Description
RTC_CR0	0X000	Control register 0
RTC_CR1	0X004	Control register 1
RTC_SEC	0X008	Second counter register
RTC_MIN	0X00C	Minute counter register
RTC_HOUR	0X010	Hour counter register
RTC_WEEK	0X014	Week counter register
RTC_DAY	0X018	Day counter register
RTC_MON	0X01C	Month counter register
RTC_YEAR	0X020	Year counter register
RTC_ALMMIN	0X024	Minute alarm register
RTC_ALMHOUR	0X028	Hour alarm register
RTC_ALM WEEK	0X02C	Week alarm register
RTC_COMPEN	0X030	Clock deviation compensation register

14.4.1 Control Register 0 (RTC_CR0)

* The register reset is valid only in power-on case.

Offset address: 0x000

Reset value0x0000 0000

Table 530. Control Register 0 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 531. Control Register 0 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PRDSEL				PRDX			START	HZ1SEL	HZ1OE	Res.	AMPM		PRDS	
-	R/W				R/W			R/W	R/W	R/W	-	R/W			

Table 532. Control Register 0 (3)

Bit	Flag	Description
31:15	Reserved	-
14	PRDSEL	0: use the periodic interrupt interval set by PRDS. 1: use the periodic interrupt interval set by PRDX.
13:8	PRDX	Set the interval for generating interrupt periodically ranging from 0.5 seconds to 32 seconds with the step of 0.5 seconds. 000000: 0.5 sec 000001: 1 sec 111110: 31.5 sec 111111: 32 sec
7	START	0: stop RTC counter 1: start RTC counter
6	1HZSEL	0: general-precision 1 Hz output 1: high-precision 1 Hz output
5	1HZOE	0: disable 1 Hz output 1: enable 1 Hz output
4	Reserved	-
3	AMPM	0: 12-hour mode 1: 24-hour mode
2:0	PRDS	Set the interval for generating interrupt periodically. 000: not generate period interrupt 001: 0.5 sec 010: 1 sec 011: 1 minute 100: 1 hour 101: 1 day 11x: 1 month Notes: to change the periodic interrupt interval during the time when START=1, please follow the steps below. Step 1. close the RTC interrupt in the NVIC. Step 2. change the periodic interrupt interval. Step 3. clear the RTC interrupt flag. Step 4. enable the RTC interrupt.

14.4.2 Control Register 1(RTC_CR1)

* The register reset is valid only in power-on case.

Offset address: 0x004

Reset value0X00000000

Table 533. Control Register 1 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 534. Control Register 1 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		CKSEL		ALMEN		ALMIE		Res.		ALMF		PRDF		Res.	
-		R/W		R/W		R/W		-		R/W		R/W		R/W	

Table 535. Control Register 1 (3)

Bit	Flag	Description
31:11	Reserved	-
10:8	CKSEL	RTC clock selection. 00x: LXT 32.768 k 01x: LRC 32 k 100: HXT/128 (select this item in 4 M oscillator case) 101: HXT/256 (select this item in 8 M oscillator case) 110: HXT/512 (select this item in 16 M oscillator case) 111: HXT/1024 (select this item in 32 M oscillator case)
7	ALMEN	0: alarm is disabled 1: alarm is enabled Note: to prevent malfunction, please close the system interrupt in the calendar count when enabling ALMEN if START = 1 and the interrupt for ALMIE = 1 is enabled. Clear the ALMF flag after the enabling.
6	ALMIE	0: alarm interrupt is disabled 1: alarm interrupt is enabled
5	Reserved	-
4	ALMF	0: alarm interrupt does not occur 1: alarm interrupt occurs Notes: this bit is valid only when ALMEN = 1. When the alarm is matched, 32.768 kHz is set to 1 after one clock. Writing 0 is for clearing the flag and writing 1 is invalid.
3	PRDF	0: alarm interrupt does not occur 1: alarm interrupt occurs Notes: when period interrupt occurs, set the bit to 1. Writing 0 is for clearing. Writing 1 is invalid.
2	Reserved	-
1	WAITF	0: Non-write/read status 1: Write/read status Note: This flag indicates whether the WAIT bit is valid. Before writing/reading, please check

		whether the bit value is 1. During the counting process, make sure clear this bit to 0 only after the WAIT bit is cleared upon writing completion.
0	WAIT	0: normal count mode 1: write/read mode Notes: set this bit to 1 during writing/reading. Since the counter is continuously counting, please make sure completing the write/read operation in 1 second and clear this bit to 0.

14.4.3 Second Count Register (RTC_SEC)

Offset address: 0x008

Reset value: uncertain

Table 536. Second Count Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 537. Second Count Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							SECH				SECL				
-							R/W				R/W				

Table 538. Second Count Register (3)

Bit	Flag	Description
31:7	Reserved	-
6:4	SECH	The ten-digit of the second count
3:0	SECL	The digit of the second count

It represents 0 - 59 minutes in decimal count. Please write the BCD code of decimal 0 - 59. The written value will be ignored in error value case.

14.4.4 Minute Count Register (RTC_MIN)

Offset address: 0x00C

Reset value: uncertain

Table 539. Minute Count Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 540. Minute Count Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							MINH				MINL				

-	R/W	R/W
---	-----	-----

Table 541. Minute Count Register (3)

Bit	Flag	Description
31:7	Reserved	-
6:4	MINH	The ten-digit of the minute count
3:0	MINL	The digit of the minute count

It represents 0 - 59 seconds in decimal count. Please write the BCD code of decimal 0 - 59. The written value will be ignored in error value case.

14.4.5 Hour Count Register (RTC_HOUR)

Offset address: 0x010

Reset value: uncertain

Table 542. Hour Count Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 543. Hour Count Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								HOURH				HOURL			
-								R/W				R/W			

Table 544. Hour Count Register (3)

Bit	Flag	Description
31:6	Reserved	-
5:4	HOURH	The ten-digit of the hour count
3:0	HOURL	The digit of the hour count

In 24-hour mode, it represents 0 - 23 hours. In 12-hour mode, it represents AM when b5 = 0, meaning 01:12 representing morning. It represents PM when b5 = 1, meaning 21:32 representing afternoon.

Please setup the correct decimal (for instance, 0:23, 01:12 or 21:32) BCD code according to the AM/PM control. The written value will be ignored if it is out-of-range.

The specific time expression is shown in the below table.

Table 545. Specific Time Expression

24-hour Mode		AMPM = 1		12-hour Mode		AMPM = 0	
Time		Register Value		Time		Register Value	
00 hour		00H		AM 12 hour		12H	
01 hour		01H		AM 01 hour		01H	

24-hour Mode		AMPM = 1	12-hour Mode		AMPM = 0
Time		Register Value	Time		Register Value
02 hour		02H	AM 02 hour		02H
03 hour		03H	AM 03 hour		03H
04 hour		04H	AM 04 hour		04H
05 hour		05H	AM 05 hour		05H
06 hour		06H	AM 06 hour		06H
07 hour		07H	AM 07 hour		07H
08 hour		08H	AM 08 hour		08H
09 hour		09H	AM 09 hour		09H
10 hour		10H	AM 10 hour		10H
11 hour		11H	AM 11 hour		11H
12 hour		12H	PM 12 hour		32H
13 hour		13H	PM 01 hour		21H
14 hour		14H	PM 02 hour		22H
15 hour		15H	PM 03 hour		23H
16 hour		16H	PM 04 hour		24H
17 hour		17H	PM 05 hour		25H
18 hour		18H	PM 06 hour		26H
19 hour		19H	PM 07 hour		27H
20 hour		20H	PM 08 hour		28H
21 hour		21H	PM 09 hour		29H
22 hour		22H	PM 10 hour		30H
23 hour		23H	PM 11 hour		31H

14.4.6 Day Count Register (RTC_DAY)

Offset address: 0x018

Reset value: uncertain

Table 546. Day Count Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 547. Day Count Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										DAYH	DAYL				
-										R/W	R/W				

Table 548. Day Count Register (3)

Bit	Flag	Description
31:6	Reserved	-
5:4	DAYH	The ten-digit of the day count
3:0	DAYL	The digit of the day count

In decimal expression, it represents 1:31 with the auto-calculation of leap year and month. The specific expression is as follows.

Table 549. Month Expression

Month	Day Count Value
February (ordinary year)	01:28
February (leap year)	01:29
April, June, September, November	01:30
January, March, May, July, August, October, December	01:31

14.4.7 Week Count Register (RTC_WEEK)

Offset address: 0x014

Reset value: uncertain

Table 550. Week Count Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 551. Week Count Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										WEEK					
-										R/W					

Table 552. Week Count Register (3)

Bit	Flag	Description
31:3	Reserved	-
2:0	WEEK	Week count value

Decimal 0:6 represents Sunday:Saturday. Please write the correct BCD code of the decimal 0:6. The written value will be ignored if it is out-of-range. The correspondence of the week count values is shown in the below table.

Table 553. Week Count Value

Day of Week	Week count value
Sunday	00H
Monday	01H
Tuesday	02H
Wednesday	03H
Thursday	04H
Friday	05H
Saturday	06H

14.4.8 Month Count Register (RTC_MON)

Offset address: 0x01C

Reset value: uncertain

Table 554. Month Count Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 555. Month Count Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										MON					
-										R/W					

Table 556. Month Count Register (3)

Bit	Flag	Description
31:5	Reserved	-
4:0	MON	Month count value

Decimal 0:12 represents January:December. Please write the correct BCD code of the decimal 0:12. The written value will be ignored if it is out-of-range.

14.4.9 Year Count Register (RTC_YEAR)

Offset address: 0x020

Reset value: uncertain

Table 557. Year Count Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 558. Year Count Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								YEARH				YEARL			
-								R/W				R/W			

Table 559. Year Count Register (3)

Bit	Flag	Description
31:8	Reserved	-
7:4	YEARH	The ten-digit of the year count
3:0	YEARL	The digit of the year count

In decimal expression, it represents 0:99. The year is counted based on month value increasing with auto-calculation of leap year (for instance 00, 04, 08, ..., 92, 96). Please write the correct decimal year count value. The written value will be ignored in error value case.

14.4.10 Minute Alarm Register (RTC_ALMMIN)

Offset address: 0x024

Reset value: uncertain

Table 560. Minute Alarm Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 561. Minute Alarm Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								ALMMINH				ALMMINL			
-								R/W				R/W			

Table 562. Minute Alarm Register (3)

Bit	Flag	Description
31:6	Reserved	-
5:4	ALMMINH	The ten-digit of the minute alarm matching value
3:0	ALMMINL	The digit of the minute alarm matching value

Please write the correct BCD code of the decimal 0:59. Alarm matching will not occur if it is out-of-range.

14.4.11 Hour Alarm Register (RTC_ALMHOUR)

Offset address: 0x028

Reset value: uncertain

Table 563. Hour Alarm Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 564. Hour Alarm Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										ALMHOURH	ALMHOURL				
-										R/W	R/W				

Table 565. Hour Alarm Register (3)

Bit	Flag	Description
31:6	Reserved	-
5:4	ALMHOURH	The ten-digit of the hour alarm matching value
3:0	ALMHOURL	The digit of the hour alarm matching value

Please write the correct alarm matching value, otherwise, alarm matching will not occur.

14.4.12 Week Alarm Register (RTC_ALMWEEK)

Offset address: 0x02C

Reset value: uncertain

Table 566. Week Alarm Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 567. Week Alarm Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										WEEK					
-										R/W					

Table 568. Week Alarm Register (3)

Bit	Flag	Description
31:7	Reserved	-
6:0	ALMWEEK	Week alarm matching value. B0:b6 represents Sunday:Saturday. When the corresponding bit is set to 1, it means that the alarm is enabled on the specific day of a week. For example, when b0=1 and b5=1, it means that the alarm are enabled on Sunday and Friday.

Please write the correct alarm matching value, otherwise, alarm matching will not occur.

14.4.13 Clock Deviation Compensation Register (RTC_COMPEN)

Offset address: 0x030

* The register reset is valid only in power-on case. Reset value: 0x00000020.

Table 569. Clock Deviation Compensation Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 570. Clock Deviation Compensation Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN	Reserved										CR				
R/W	-										R/W				

Table 571. Clock Deviation Compensation Register (3)

Bit	Flag	Description																																																																																																																								
31:16	Reserved	-																																																																																																																								
15	EN	Compensation enabling. 0: the clock deviation compensation is disabled. 1: the clock deviation compensation is enabled.																																																																																																																								
14:b	Reserved	-																																																																																																																								
8:0	CR	<p>Compensation value</p> <p>By the compensation value setting, it can reach an compensation accuracy of +/-0.96 ppm per second. The compensation value is 2's complement, a 9-bit decimal value with the last 5 bits being decimal parts, a compensation range of 274.6:212.6 ppm, a minimum differential error of +/- 0.48 ppm and a minimum resolution of 0.96 ppm. Please refer to the following table for specific compensation accuracy.</p> <table border="1"> <thead> <tr> <th colspan="10">Compensation Value Setting</th> <th rowspan="2">Compensation Value</th> </tr> <tr> <th>EN</th> <th colspan="9">CR[8:0]</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td> <td>- 274.6 ppm</td> </tr> <tr> <td>1</td> <td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td> <td>- 273.7 ppm</td> </tr> <tr> <td>:</td> <td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td> <td>:</td> </tr> <tr> <td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td> <td>- 0.95 ppm</td> </tr> <tr> <td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td> <td>0 ppm</td> </tr> <tr> <td>:</td> <td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td> <td>:</td> </tr> <tr> <td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td> <td>+ 211.7 ppm</td> </tr> <tr> <td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td> <td>+ 212.6 ppm</td> </tr> <tr> <td>0</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td> <td>No compensation</td> </tr> </tbody> </table>	Compensation Value Setting										Compensation Value	EN	CR[8:0]									1	1	0	0	0	0	0	0	0	0	- 274.6 ppm	1	0	0	0	0	0	0	0	0	1	- 273.7 ppm	:	:	:	:	:	:	:	:	:	:	:	0	0	0	0	1	1	1	1	1	1	- 0.95 ppm	0	0	0	1	0	0	0	0	0	0	0 ppm	:	:	:	:	:	:	:	:	:	:	:	0	1	1	1	1	1	1	1	1	0	+ 211.7 ppm	0	1	1	1	1	1	1	1	1	1	+ 212.6 ppm	0	X	X	X	X	X	X	X	X	X	No compensation
Compensation Value Setting										Compensation Value																																																																																																																
EN	CR[8:0]																																																																																																																									
1	1	0	0	0	0	0	0	0	0	- 274.6 ppm																																																																																																																
1	0	0	0	0	0	0	0	0	1	- 273.7 ppm																																																																																																																
:	:	:	:	:	:	:	:	:	:	:																																																																																																																
0	0	0	0	1	1	1	1	1	1	- 0.95 ppm																																																																																																																
0	0	0	1	0	0	0	0	0	0	0 ppm																																																																																																																
:	:	:	:	:	:	:	:	:	:	:																																																																																																																
0	1	1	1	1	1	1	1	1	0	+ 211.7 ppm																																																																																																																
0	1	1	1	1	1	1	1	1	1	+ 212.6 ppm																																																																																																																
0	X	X	X	X	X	X	X	X	X	No compensation																																																																																																																

The principle and calculation for compensation is as follows.

Since the counter uses 32.768 kHz clock count, if it requires compensation per second based accuracy, it can only compensate according to the integer part of 32.768 kHz, thus the minimum unit of compensation per second is $(1/32768)*106 = 30.5$ ppm, fail to satisfy high-precision requirements.

To achieve higher-precision clock compensation on the 32.768 kHz count clock, an algorithm adjustment is required to enlarge the maximum compensation period by 32 times. Thus, in the case of a minimum compensation unit of 30.5 ppm, the averaged compensation unit per second becomes $30.5 \text{ ppm} / 32 = 0.96$ ppm, which can meet the high-precision clock compensation requirements well. Moreover, the compensations are evenly distributed in every 32 seconds. Therefore, a setting of 5 decimal places is introduced in this register.

Example 1:

When the 1 Hz clock is directly output by default, the target compensation value is calculated by measuring the accuracy of the clock. Assuming the actual measured value is 0.9999888 Hz, then the calculation is as follows.

$$\text{Actual vibration frequency} = 32768 \times 0.9999888 \approx 32767.63$$

$$\begin{aligned}\text{Target compensation value} &= (\text{actual vibration frequency} - \text{target frequency}) / \text{target frequency} \times 106 \\ &= (32767.96 - 32768) / 32768 \times 106 \\ &= -11.29\text{ppm}\end{aligned}$$

According to the formula below,

$$\text{CR[8:0]} = \left(\frac{\text{target compensation value [ppm]} \times 2^{15}}{10^6} \right) \text{taking 2's complement} + 0001.00000B$$

As an example, If the target compensation value is + 20.6 ppm, the calculation of the corresponding register value is as follows.

$$\begin{aligned}\text{CR[8:0]} &= (20.6 \times 2^{15}/10^6) \text{ taking 2's complement} + 0001.00000B \\ &= (0.6651904) \text{ taking 2's complement} + 0001.00000B \\ &= 0000.10101B + 0001.00000B \\ &= 0000.10101B\end{aligned}$$

As another example, If the target compensation value is - 20.6 ppm, the calculation of the corresponding register value is as follows.

$$\begin{aligned}\text{CR[8:0]} &= (- 20.6 \times 2^{15}/10^6) \text{ taking 2's complement} + 0001.00000B \\ &= (- 0.6651904) \text{ taking 2's complement} + 0001.00000B \\ &= 1111.01011B + 0001.00000B \\ &= 0000.01011B\end{aligned}$$

15 Watchdog WDT

15.1 WDT Introduction

WDT can be used to detect and resolve faults caused by software errors. When the WDT counter reaches the set overflow time, an interrupt is triggered or a system reset is generated. WDT is driven by a dedicated 10 kHz on-chip oscillator.

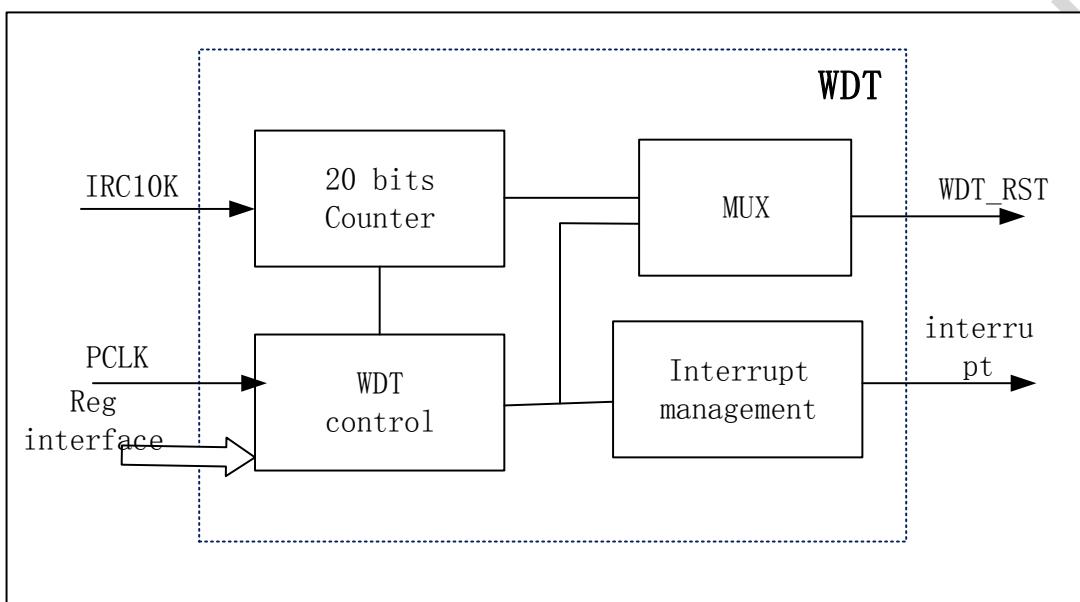


Figure 73. WDT Overall Block Diagram

15.2 WDT Function Description

- 20-bit freely-operating counter, with the overflow time set ranging from 1.6 ms to 50 s.
- The action after overflow can be configured as interrupt or reset.
- The WDT clock is provided by a separate RC oscillator, which can operate in sleep and deep sleep modes.
- The WDTCON register can only be modified when the WDT is not started yet, to prevent unintentional modification of the WDT configuration after startup.

15.2.1 Interrupt after WDT Overflow

In this mode, WDT will generate interrupt periodically based on the set time. The WDT interrupt flag needs to be cleared in the interrupt service routine.

The configuration steps are as follows.

1. Configure WDT_CON.WOV to select the WDT overflow time.
2. Set WDT_CON.WINT_EN to 1 to choose generating interrupt after the WDT overflow.
3. Enable the WDT interrupt in the NVIC interrupt vector table.
4. Write 0x1E and 0xE1 to the WDT_RST register in sequence to start the WDT timer.

5. Write 0x1E and 0xE1 to the WDT_RST register in the interrupt service routine to clear the interrupt flag.

15.2.2 Reset after WDT Overflow

In this mode, WDT will generate reset signal which resets MCU. It needs to clear the WDT counter before the WDT overflow in user program to prevent WDT reset.

The configuration steps are as follows.

1. Configure WDT_CON.WOV to select the WDT overflow time.
2. Set WDT_CON.WINT_EN to 0 to choose generating reset signal after the WDT overflow.
3. Write 0x1E and 0xE1 to the WDT_RST register in sequence to start the WDT timer.
4. Write 0x1E and 0xE1 to the WDT_RST register in sequence before the WDT overflow to clear the WDT counter.

Note:

1. Since the WDT oscillator is a low precision RC oscillator, it is strongly recommended to clear WDT before the WDT counter count reaches half of the overflow value.

15.3 WDT Register Description

Base address: 0X40000C00

Table 572. WDT Register List

Register	Offset Address	Description
WDT_RST	0X080	WDT clearing control register
WDT_CON	0X084	WDT control register

15.3.1 WDT Clearing Control Register (WDT_RST)

Offset address: 0x080

Reset value: 0x0000 0000

Table 573. WDT Clearing Control Register (1)

31:8	7	6	5	4	3	2	1	0
Reserved	WDTRST							
-	W							

Table 574. WDT Clearing Control Register (2)

Bit	Flag	Description
31:8	Reserved	Reserved bit, read as 0
7:0	WDTRST	Watchdog start/clearing control When the watchdog does not start yet, write 0x1E and 0xE1 to this register in sequence to start the

		<p>WDT timer.</p> <p>When the watchdog starts, write 0x1E and 0xE1 to this register in sequence to clear the WDT timer and interrupt flag.</p>
--	--	------------------------------------------------------------------------------------------------------------------------------------------------

15.3.2 WDT_CON Register

Offset address: 0x084

Reset value: 0x0000 000F

Notes: this register can be written only when WDT does not start.

Table 575. WDT_CON Register (1)

31:16	15:8	7	6	5	4	3	2	1	0
Reserved	WCNTL	WDINT	Res.	WINT_EN	WDTR			WOV	
-	R	R	-	R/W	R			R/W	

Table 576. WDT_CON Register (3)

Bit	Flag	Description																
31:16	Reserved	Reserved bit, read as 0.																
15:8	WCNTL	The low 8-bit of the WDT count.																
7	WDTINT	WDT interrupt flag 1: interrupt occurs, write 0x1E, 0xE1 to the WDT_RST register in sequence to the interrupt flag. 0: interrupt does not occur																
5	WINT_EN	Setting the action after the WDT overflow 1: generate interrupt after the overflow. 0: generate reset signal after the overflow.																
4	WDTR	WDT running flag 1: WDT runs 0: WDT stops																
3:0	WOV[3:0]	WDT count overflow time setting. <table> <tr> <td>0000: 1.6 ms</td> <td>1000: 500 ms</td> </tr> <tr> <td>001: 3.2 ms</td> <td>1001: 820 ms</td> </tr> <tr> <td>0010: 6.4 ms</td> <td>1010: 1.64 s</td> </tr> <tr> <td>0011: 13 ms</td> <td>1011: 3.28 s</td> </tr> <tr> <td>0100: 26 ms</td> <td>1100: 6.55 s</td> </tr> <tr> <td>0101: 51 ms</td> <td>1101: 13.1 s</td> </tr> <tr> <td>0110: 102 ms</td> <td>1110: 26.2 s</td> </tr> <tr> <td>0111: 205 ms</td> <td>1111: 52.4 s</td> </tr> </table>	0000: 1.6 ms	1000: 500 ms	001: 3.2 ms	1001: 820 ms	0010: 6.4 ms	1010: 1.64 s	0011: 13 ms	1011: 3.28 s	0100: 26 ms	1100: 6.55 s	0101: 51 ms	1101: 13.1 s	0110: 102 ms	1110: 26.2 s	0111: 205 ms	1111: 52.4 s
0000: 1.6 ms	1000: 500 ms																	
001: 3.2 ms	1001: 820 ms																	
0010: 6.4 ms	1010: 1.64 s																	
0011: 13 ms	1011: 3.28 s																	
0100: 26 ms	1100: 6.55 s																	
0101: 51 ms	1101: 13.1 s																	
0110: 102 ms	1110: 26.2 s																	
0111: 205 ms	1111: 52.4 s																	

16 UART

16.1 Overview

This product provides 2 general purpose UART modules (UART0/1) with supports of half-duplex and full-duplex transmission, 8-bit and 9-bit data formats and 4 different transmission modes - mode 0/1/2/3. The UART0 baud rate is generated by TIMER0 and UART1 by TIMER1. It features multiprocessor communication mode, automatic address recognition and support of given address and broadcast address.

The general purpose UART (UART0/1) shares only 1 clock input PCLK and both the register configuration logic and the data transceiving logic operate in this clock domain.

16.2 Block Diagram

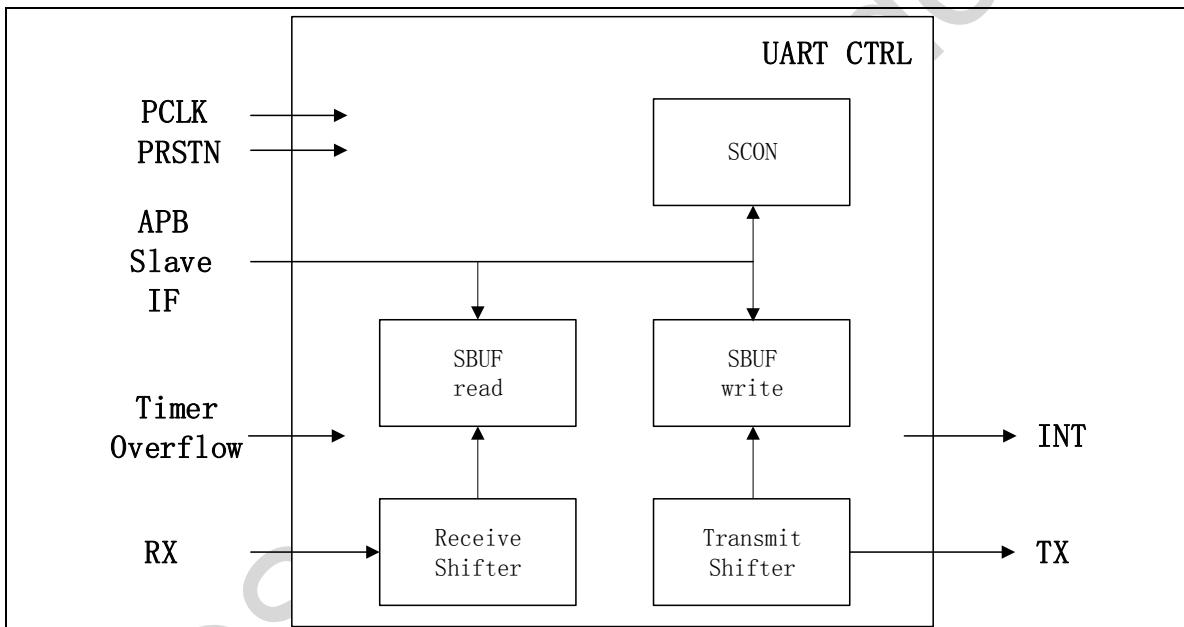


Figure 74. UART Block Diagram

16.3 Operating Mode

16.3.1 Mode 0 (Synchronous Mode, Half Duplex)

When operating in mode 0, the UART operates in synchronous mode with a fixed baud rate as 1/12 of the PCLK clock. The UART receiving data is input through RXD and the transmitting data is output through TXD output as well. Thus RXD operates as input and output port. The UART synchronous shift clock is output by the TXD which operates as output here. Note that this mode can only be used as a master to send a synchronous shift clock, and it cannot receive such clock as a slave from outside. In this mode, the transmitted data bit width can only be 8-bit without start and end bit.

Clear UARTx_SCON.SM0 and UARTx_SCON.SM1 to enter mode 0 operation.

16.3.1.1 Transmitting data

When transmitting data, clear the `UARTx_SCON.REN` bit and write the data to the `UARTx_SBUF` register. In the case, the transmitting data will be output from RXD (lower bit first) and the synchronous shift clock will be output from TXD.

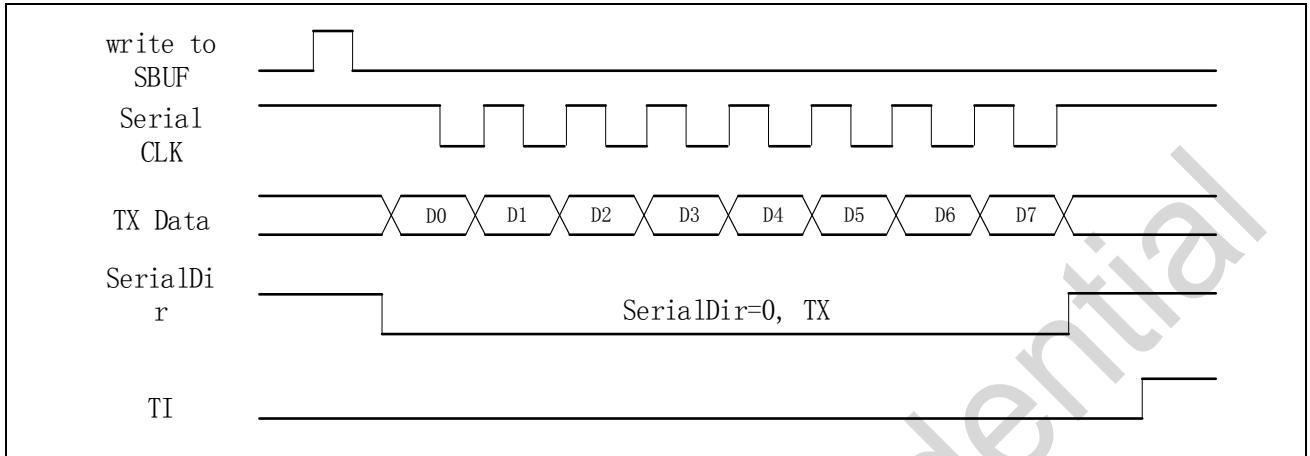


Figure 75. Transmitting Data in Mode 0

16.3.1.2 Receiving Data

When receiving data, set the `UARTx_SCON.REN` bit and clear the `UARTx_ISR.RI` bit. When receiving completes, data can be read from the `UARTx_SBUF` register. In the case, the received data is input from RXD (lower bit first), and the synchronous shift clock is output from TXD.

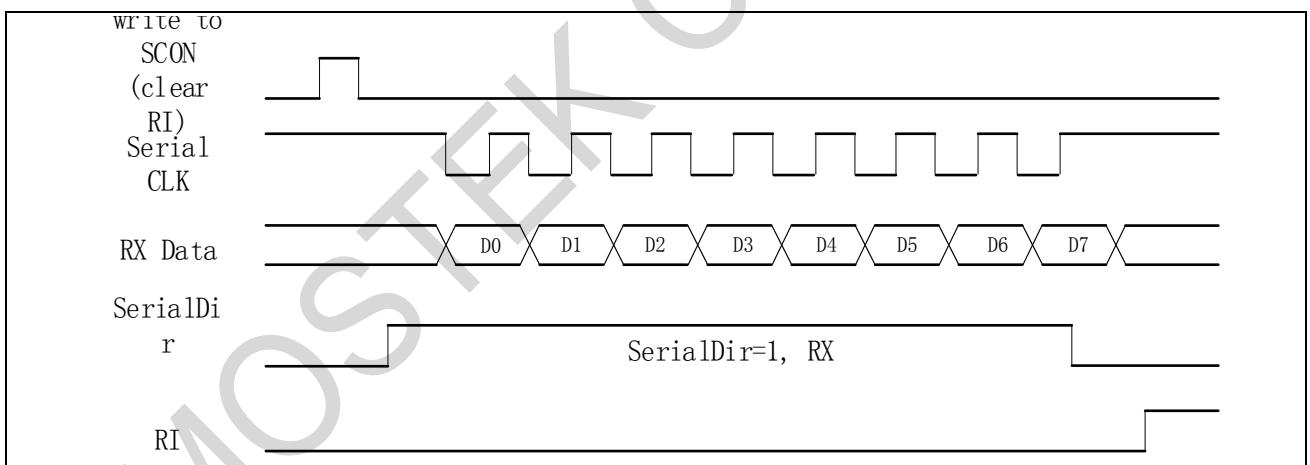


Figure 76. Receiving Data in Mode 0

16.3.2 Mode1 (Asynchronous Mode, Full-duplex)

When operating in mode1, the transmitted data is transmitted through TXD, and the received data is received through RXD. The data consists of 10 bits including the start bit as 0, followed by the 8 data bits (lower bit first), and the end bit as 1.

In this mode, the baud rate is generated by the timer module, which is programmable. The baud rate of UART0 is generated by TIMER0, UART1 by TIMER1.

Clear `UARTx_SCON.SM0` to 0 and `UARTx_SCON.SM1` to 1 to enter mode 1.

16.3.2.1 Transmitting Data

When transmitting data, the transmitted data is written to the `UARTx_SBUF` register and transmitted from `TXD` (lower bit first), which is irrelevant to the value of `UARTx_SCON.REN`.

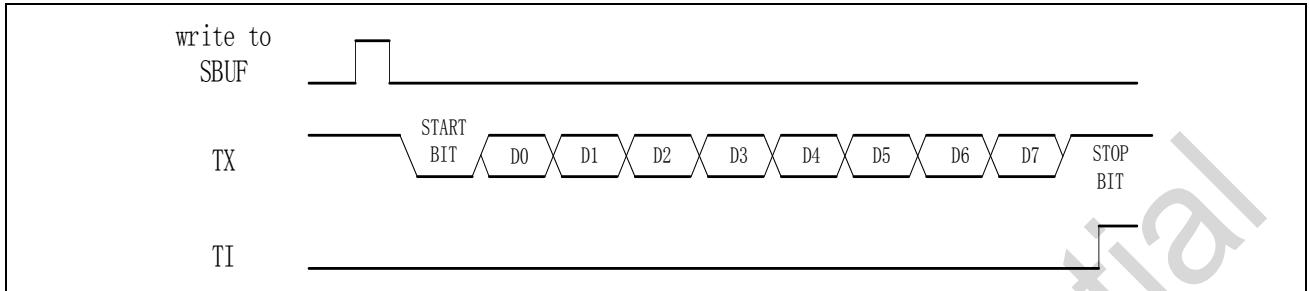


Figure 77. Transmitting Data in Mode1

16.3.2.2 Receiving Data

When receiving data, set the `UARTx_SCON.REN` bit and clear the `UARTx_ISR.RI` bit to 0 then start receiving data on `RXD` (lower bit first). When receiving completes, the data can be read from the `UARTx_SBUF` register.

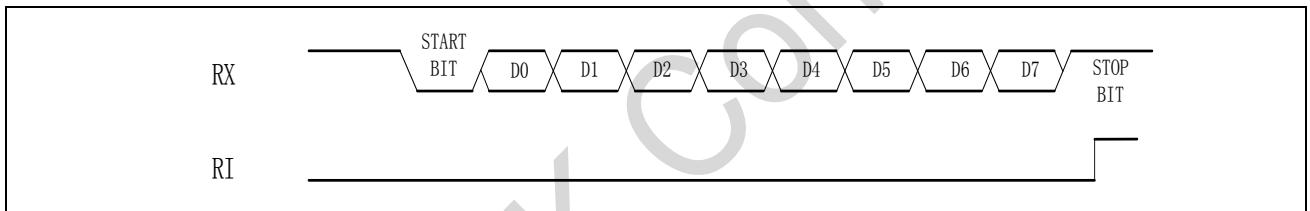


Figure 78. Receiving Data in Mode 1

16.3.3 Mode 2 (Asynchronous Mode, Full-duplex)

When operating in mode 2, the transmitted data is transmitted through `TXD`, and the received data is received through `RXD`. The data consists of 11 bits including the start bit as 0, followed by the 8 data bits (lower bit first), then 1 `TB8` bit and the end bit. The extra `TB8` bit is used in multiprocessor communication environments. When `TB8 = 1`, it indicates that a address frame is received. When `TB8 = 0`, it indicates a data frame is received. This bit can also be used as a parity bit when multiprocessor communication is not required.

In this mode, the baud rate can be generated independently with no external TIMER required.

Set `UARTx_SCON.SM0` to 1 and `UARTx_SCON.SM1` to 0 to enter mode 2.

16.3.3.1 Transmitting Data

When transmitting data, the transmitted data is written to the `UARTx_SBUF` register and transmitted from `TXD` (lower bit first), which is irrelevant to the value of `UARTx_SCON.REN`.

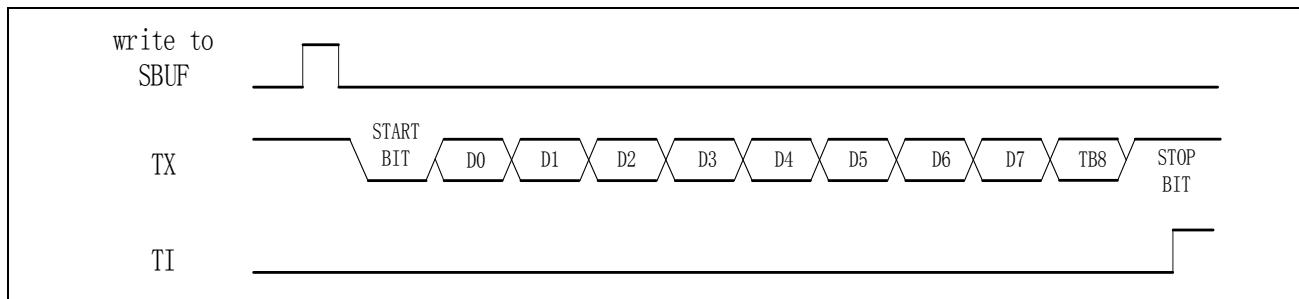


Figure 79. Transmitting Data in Mode 2

16.3.3.2 Receiving Data

When receiving data, set the `UARTx_SCON.REN` bit to 1 and clear the `UARTx_ISR.RI` bit to 0 then start receiving data on `RXD` (lower bit first). When receiving completes, the data can be read from the `UARTx_SBUF` register.

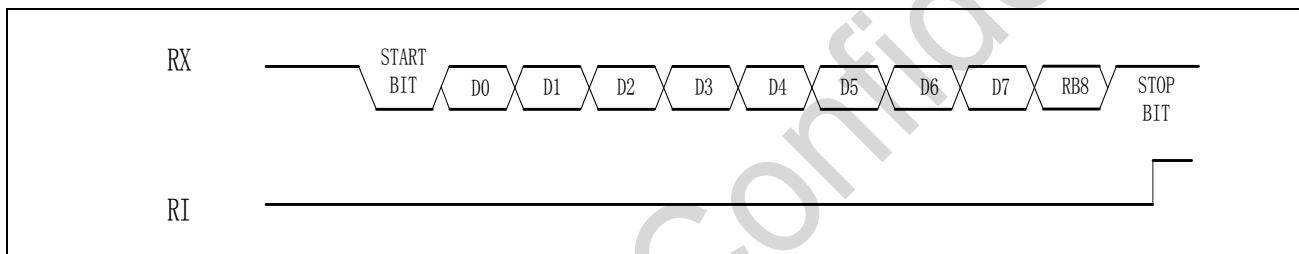


Figure 80. Receiving Data in Mode 2

16.3.4 Mode 3 (Asynchronous Mode, Full-duplex)

The data format, transmission timing and operation mode of mode 3 are the same as those of mode 2. However, unlike mode 2 with the baud rate being generated by the device itself independently, the baud rate in mode 3 is generated by TIMER instead, which is the only difference between them. The baud rate of mode 3 is programmable and it is generated in the same way as mode 1. In this product, the baud rate of UART0 is generated by TIMER0, UART1 by TIMER1.

Set `UARTx_SCON.SM0` to 1 and `UARTx_SCON.SM1` to 1 to enter mode 3.

16.4 Baud Rate Programming

16.4.1 Mode 0

When operating in mode 0, the baud rate is fixed at 1/12 of PCLK without TIMER support required.

16.4.2 Mode 1/3

When operating in mode 1 or mode 3, the baud rate depends on the TIMER overflow time. The specific formula is shown as follows.

$$\text{BaudRate} = \frac{(\text{UARTx_SCON.DBAUD} + 1) * \text{Freq}}{32 * (65536 - \text{TM})}$$

Among them, `UARTx_SCON.DBAUD` represents double baud rate, `Freq` is PCLK clock frequency, and `TM` is TIMER count value. Note that TIMER must be configured as 16-bit auto-reload mode, and both the count register and the reload register must be written with the `TM` value.

16.4.3 Mode 2

When operating in mode 2, the baud rate is fixed to the value calculated out via the specific formula as follows.

$$\text{BaudRate} = \frac{(\text{UARTx_SCON.DBAUD} + 1) * \text{Freq}}{64}$$

Among them, `UARTx_SCON.DBAUD` represents double baud rate and `Freq` is PCLK clock frequency.

16.5 Frame Error Detection

Mode 1/2/3 supports the frame error detection function with hardware automatically detecting whether the received frame data contains a valid *stop* bit. If the valid *stop* bit is not received, `UARTx_ISR.FE` is set. The `UARTx_ISR.FE` bit is set to 1 by hardware and cleared to 0 by software. If it is not cleared by software in time, even the subsequent received data contains the valid *stop* bit, it cannot be cleared to 0.

16.6 Multiprocessor Communication

To supports multiprocessor communication function in mode 2/3, the 1-bit `TB8/RB8` is added into its frame format. Set `UARTx_SCON.SM2` to 1 to enable the multiprocessor communication bit.

When the multiprocessor communication bit is enabled, the master can use `UARTx_SCON.TB8` to distinguish whether the current frame is an address frame (`UARTx_SCON.TB8 = 1`) or a data frame (`UARTx_SCON.TB8=0`). When receiving data, the slave ignores the current received frame with the `RB8` bit (bit 9) being 0. The `RB8` bit (bit 9) of the received frame being 1 indicates an address frame received. The slave then judges whether the received address is equal to its own address. If there is a match, the slave sets both `UARTx_SCON.RB8` and `UARTx_ISR.R` to 1, indicating an address frame received with address matched. When the slave software checks out both `UARTx_SCON.RB8` and `UARTx_ISR.RI` being 1, it clears the `UARTx_SCON.SM2` bit to 0 and then prepares receiving data frame. If the address is not equal, it indicates that the master is not addressing the slave. `UARTx_SCON.RB8` and `UARTx_ISR.RI` is retained as 0 by hardware. `UARTx_SCON.SM2` bit is retained as 1 by software and the slave keeps on the address listening.

16.7 Automatic Address Recognition

When the multiprocessor communication bit is enabled (`UARTx_SCON.SM2` is set to 1), the automatic address recognition function will be enabled then. This function is implemented in hardware level, so that the slave can detect each address frame received and have address matching then. If it succeeds in matching the slave address, the receiver will set the receiving flag `UARTx_ISR.RI`, otherwise, no receiving flag is set if it fails.

If necessary, the multiprocessor communication bit can also be enabled in mode 1, and the `TB8` bit is replaced by the *stop* bit in the case. When the slave receives a matching address frame and a valid *stop* bit, `UARTx_ISR.RI` is set to 1.

To support automatic address recognition, the concepts of broadcast address and given address are introduced.

16.7.1 Given Address

The UARTx_SADDR register of the UART device is to contain the given device-address of its own. The UARTx_SADEN register is to contain an address mask representing the bitmasks irrelevant to the specific address. When a certain bit in UARTx_SADEN is 0, it indicates this address bit is an irrelevant bit, meaning that this address bit does not participate in address matching. These irrelevant bits help increase the flexibility of addressing so that the master can address one or more slave devices simultaneously. Note that if a single matching address is required, the UARTx_SADEN register must be set as 8'hFF.

$$\text{GivenAddr} = \text{SADDR} \& \text{ SADEN}$$

16.7.2 Broadcast Address

The broadcast address is used to address all slave devices at the same time. Generally, the broadcast address is 8'hFF.

$$\text{BroadcastAddr} = \text{SADDR} | \text{SADEN}$$

Take the below case as an example.

Suppose a slave's UARTx_SADDR and UARTx_SADEN are configured as follows.

SADDR: 8'b01101001

SADEN: 8'b11111011

Then its given address and broadcast address are as follows.

Given: 8'b01101x01

Broadcast: 8'b11111x11

It can be seen that the master can address the slave with 4 addresses as follows.

8'b01101001 and 8'b01101101 (given address)

8'b11111011 and 8'b11111111 (broadcast address).

16.8 Transceiver Buffer

16.8.1 Receiving Buffer

The general purpose UART (UART0/1) receiver supports a receiving buffer with 1 frame length (8/9-bit), meaning that when a frame of data is received, the data in the receive buffer retains until it finishes receiving the *stop* bit of the next frame, thereafter the receiving buffer is updated with the new data frame.

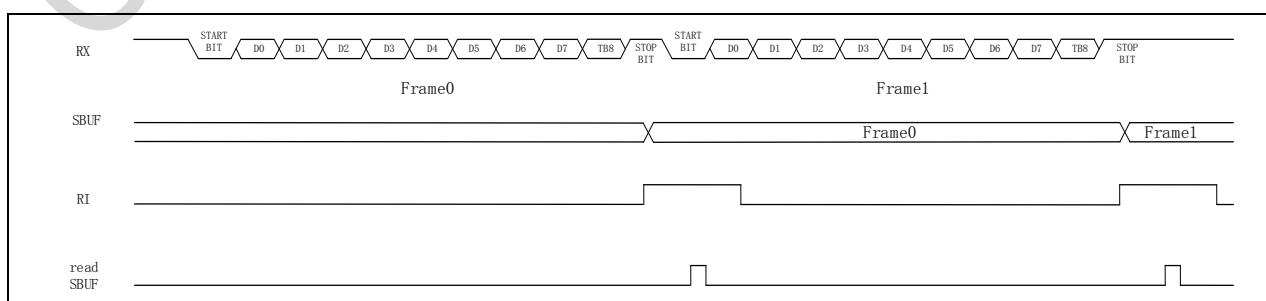


Figure 81. Receiving Buffer

16.8.2 Transmitting Buffer

The general purpose UART (UART0/1) transmitter does not support transmitting buffer. If the `UARTx_SBUF` register is filled during data transmission, the data currently being transmitted will be corrupted. Such operation should be prevented in software.

16.9 Frequency Dividing Settings for Different Baud Rates

Table 577. Frequency Dividing Settings for PCLK = 1 MHz

Baud Rate	PCLK = 1 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	26	2403.85	0.16%	13	2403.85	0.16%
4800	13	4807.69	0.16%	7	4464.29	- 6.99%
9600	7	8928.57	- 6.99%	3	10416.67	8.51%
19200	3	20833.33	8.51%	2	15625.00	-18.62%
38400	2	31250.00	- 18.62%	1	31250.00	-18.62%
57600	1	62500.00	8.51%	1	31250.00	- 45.75%
76800	1	62500.00	-18.62%	0	#DIV/0!	#DIV/0!
115200	1	62500.00	- 45.75%	0	#DIV/0!	#DIV/0!

Table 578. Frequency Dividing Settings for PCLK = 4 MHz

Baud Rate	PCLK = 4 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	104	2403.85	0.16%	52	2403.85	0.16%
4800	52	4807.69	0.16%	26	4807.69	0.16%
9600	26	9615.38	0.16%	13	9615.38	0.16%
19200	13	19230.77	0.16%	7	17857.14	- 6.99%
38400	7	35714.29	- 6.99%	3	41666.67	8.51%
57600	4	62500.00	8.51%	2	62500.00	8.51%
76800	3	83333.33	8.51%	2	62500.00	- 18.62%
115200	2	125000.00	8.51%	1	125000.00	8.51%

Table 579. Frequency Dividing Settings for PCLK = 10 MHz

Baud Rate	PCLK = 10 MHz
-----------	---------------

	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	260	2403.85	0.16%	130	2403.85	0.16%
4800	130	4807.69	0.16%	65	4807.69	0.16%
9600	65	9615.38	0.16%	33	9469.70	-1.36%
19200	33	18939.39	-1.36%	16	19531.25	1.73%
38400	16	39062.50	1.73%	8	39062.50	1.73%
57600	11	56818.18	-1.36%	5	62500.00	8.51%
76800	8	78125.00	1.73%	4	78125.00	1.73%
115200	5	125000.00	8.51%	3	104166.67	-9.58%

Table 580. Frequency Dividing Settings for PCLK = 14 MHz

Baud Rate	PCLK = 14 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	365	2397.26	-0.11%	182	2403.85	0.16%
4800	182	4807.69	0.16%	91	4807.69	0.16%
9600	91	9615.38	0.16%	46	9510.87	-0.93%
19200	46	19021.74	-0.93%	23	19021.74	-0.93%
38400	23	38043.48	-0.93%	11	39772.73	3.57%
57600	15	58333.33	1.27%	8	54687.50	-5.06%
76800	11	79545.45	3.57%	6	72916.67	-5.06%
115200	8	109375.00	-5.06%	4	109375.00	-5.06%

Table 581. Frequency Dividing Settings for PCLK = 20 MHz

Baud Rate	PCLK = 20 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	521	2399.23	-0.03%	260	2403.85	0.16%
4800	260	4807.69	0.16%	130	4807.69	0.16%
9600	130	9615.38	0.16%	65	9615.38	0.16%
19200	65	19230.77	0.16%	33	18939.39	-1.36%
38400	33	37878.79	-1.36%	16	39062.50	1.73%

57600	22	56818.18	- 1.36%	11	56818.18	- 1.36%
76800	16	78125.00	1.73%	8	78125.00	1.73%
115200	11	113636.36	- 1.36%	5	125000.00	8.51%

Table 582. Frequency Dividing Settings for PCLK = 24 MHz

Baud Rate	PCLK = 24 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	625	2400.00	0.00%	313	2396.17	- 0.16%
4800	313	4792.33	- 0.16%	156	4807.69	0.16%
9600	156	9615.38	0.16%	78	9615.38	0.16%
19200	78	19230.77	0.16%	39	19230.77	0.16%
38400	39	38461.54	0.16%	20	37500.00	- 2.34%
57600	26	57692.31	0.16%	13	57692.31	0.16%
76800	20	75000.00	- 2.34%	10	75000.00	- 2.34%
115200	13	115384.62	0.16%	7	107142.86	- 6.99%

Table 583. Frequency Dividing Settings for PCLK = 2 MHz

Baud Rate	PCLK = 2 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	52	2403.85	0.16%	26	2403.85	0.16%
4800	26	4807.69	0.16%	13	4807.69	0.16%
9600	13	9615.38	0.16%	7	8928.57	- 6.99%
19200	7	17857.14	- 6.99%	3	20833.33	8.51%
38400	3	41666.67	8.51%	2	31250.00	- 18.62%
57600	2	62500.00	8.51%	1	62500.00	8.51%
76800	2	62500.00	- 18.62%	1	62500.00	- 18.62%
115200	1	125000.00	8.51%	1	62500.00	- 45.75%

Table 584. Frequency Dividing Settings for PCLK = 3 MHz

Baud Rate	PCLK = 8 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	208	2403.85	0.16%	104	2403.85	0.16%
4800	104	4807.69	0.16%	52	4807.69	0.16%
9600	52	9615.38	0.16%	26	9615.38	0.16%
19200	26	19230.77	0.16%	13	19230.77	0.16%
38400	13	38461.54	0.16%	7	35714.29	- 6.99%
57600	9	55555.56	- 3.55%	4	62500.00	8.51%
76800	7	71428.57	- 6.99%	3	83333.33	8.51%
115200	4	125000.00	8.51%	2	125000.00	8.51%

Table 585. Frequency Dividing Settings for PCLK = 11.0592 MHz

Baud Rate	PCLK = 11.0592 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	288	2400.00	0.00%	144	2400.00	0.00%
4800	144	4800.00	0.00%	72	4800.00	0.00%
9600	72	9600.00	0.00%	36	9600.00	0.00%
19200	36	19200.00	0.00%	18	19200.00	0.00%

38400	18	38400.00	0.00%	9	38400.00	0.00%
57600	12	57600.00	0.00%	6	57600.00	0.00%
76800	9	76800.00	0.00%	5	69120.00	-10.00%
115200	6	115200.00	0.00%	3	115200.00	0.00%

Table 586. Frequency Dividing Settings for PCLK = 18 MHz

Baud Rate	PCLK = 16 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	417	2398.08	-0.08%	208	2403.85	0.16%
4800	208	4807.69	0.16%	104	4807.69	0.16%
9600	104	9615.38	0.16%	52	9615.38	0.16%
19200	52	19230.77	0.16%	26	19230.77	0.16%
38400	26	38461.54	0.16%	13	38461.54	0.16%
57600	17	58823.53	2.12%	9	55555.56	-3.55%
76800	13	76923.08	0.16%	7	71428.57	-6.99%
115200	9	111111.11	-3.55%	4	125000.00	8.51%

Table 587. Frequency Dividing Settings for PCLK = 32 MHz

Baud Rate	PCLK = 32 MHz					
	Dual Baud Rate			Single Baud Rate		
	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	833	2400.96	0.04%	417	2398.08	-0.08%
4800	417	4796.16	-0.08%	208	4807.69	0.16%
9600	208	9615.38	0.16%	104	9615.38	0.16%
19200	104	19230.77	0.16%	52	19230.77	0.16%
38400	52	38461.54	0.16%	26	38461.54	0.16%
57600	35	57142.86	-0.79%	17	58823.53	2.12%
76800	26	76923.08	0.16%	13	76923.08	0.16%
115200	17	117647.06	2.12%	9	111111.11	-3.55%

Table 588. Frequency Dividing Settings for PCLK = 22.12 MHz

Baud Rate	PCLK = 22.12 MHz					
	Dual Baud Rate			Single Baud Rate		

	CNT	Actual Baud Rate	Error%	CNT	Actual Baud Rate	Error%
2400	576	2400.17	0.01%	288	2400.17	0.01%
4800	288	4800.35	0.01%	144	4800.35	0.01%
9600	144	9600.69	0.01%	72	9600.69	0.01%
19200	72	19201.39	0.01%	36	19201.39	0.01%
38400	36	38402.78	0.01%	18	38402.78	0.01%
57600	24	57604.17	0.01%	12	57604.17	0.01%
76800	18	76805.56	0.01%	9	76805.56	0.01%
115200	12	115208.33	0.01%	6	115208.33	0.01%

16.10 Registers

UART0 base address: 0x4000 0000

UART1 base address: 0x4000 0100

Table 589. UART Register

Register	Offset address	Description
UARTx_SBUF	0x00	Data register
UARTx_SCON	0x04	Control register
UARTx_SADDR	0x08	Address register
UARTx_SADEN	0x0C	Address mask register
UARTx_ISR	0x10	Interrupt flag bit register
UARTx_ICR	0x14	Interrupt flag bit clearing register

16.10.1 Data Register (UARTx_SBUF)

Offset address: 0x00

Reset value: 0x0000 0000

Table 590. Data Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 591. Data Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								SBUF							

R	R/W
---	-----

Table 592. Data Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	SBUF	The transmitted data is written to the register when transmitting data. The received data is read from the register after receiving completes when receiving data. Note that, for this register, the actual reading value is read from RXBuffer and the actual writing value is written to TXShifter.

16.10.2 Control Register (UARTx_SCON)

Offset address: 0x04

Reset value: 0x0000 0000

Table 593. Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 594. Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				DBAUD		Res.		SM01		SM2		REN		TB8	
R				R/W		R		R/W		R/W		R/W		R/W	

Table 595. Control Register (3)

Bit	Flag	Description
31:10	Reserved	
9	DBAUD	Dual baud rate 0: single baud rate 1: dual baud rate
8	Reserved	
7:6	SM01	Operating mode. 00: mode 0, 01: mode 1, 10: mode 2, 11: mode 3.
5	SM2	Multiprocessor communication. 0: disable, 1: enable.
4	REN	Receiving enabling. Mode 0, 0: transmit, 1: receive. Other modes, 0: transmit, 1: transmit/receive.
3	TB8	Transmit TB8 bit.
2	RB8	Receive TB8 bit.
1	TIEN	Transmission completion interrupt enabling. 0: disable, 1: enable.
0	RIEN	Receiving completion interrupt enabling. 0: disable, 1: enable.

16.10.3 Address Register (UARTx_SADDR)

Offset address: 0x08

Reset value: 0x0000 0000

Table 596. Address Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 597. Address Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								SADDR							
R								R/W							

Table 598. Address Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	SADDR	Slave device address register.

16.10.4 Address Mask Register (UARTx_SADEN)

Offset address: 0x0C

Reset value: 0x0000 0000

Table 599. Address Mask Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 600. Address Mask Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								SADEN							
R								R/W							

Table 601. Address Mask Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	SADEN	Slave device-address mask register

16.10.5 Flag Register (UARTx_ISR)

Offset address: 0x10

Reset value: 0x0000 0000

Table 602. Flag Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 603. Flag Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												TE	FE	TI	RI
R												R	R	R	R

Table 604. Flag Register (3)

Bit	Flag	Description
31:4	Reserved	
3	TE	Transmission buffer empty interrupt flag. It is set and cleared by hardware. 1: TE interrupt is enabled 0: TE interrupt is disabled
2	FE	Received frame error interrupt flag. It is set by hardware and cleared by software. 1: FE interrupt is enabled 0: FE interrupt is disabled
1	TI	Transmission completion interrupt flag. It is set by hardware and cleared by software. 1: TI interrupt is enabled 0: TI interrupt is disabled
0	RI	Receiving completion interrupt flag. It is set by hardware and cleared by software. 1: RI interrupt is enabled 0: RI interrupt is disabled

16.10.6 Flag Clearing Register (UARTx_ICR)

Offset address: 0x14

Reset value: 0x0000 0000

Table 605. Flag Clearing Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 606. Flag Clearing Register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												FECLR	TICLR	RICLR	
R												R	R	R	

Table 607. Flag Clearing Register

Bit	Flag	Description
31:3	RESERVED	
2	FECLR	Clearing received frame error flag. Writing with 0 for clearing. Writing 1 is invalid.
1	TICLR	Clearing transmission completion error flag. Writing with 0 for clearing. Writing 1 is invalid.
0	RICLR	Clearing receiving completion flag. Writing with 0 for clearing. Writing 1 is invalid.

17 LPUART

17.1 Overview

This product contains 1 LPUART module with supports of half-duplex and full-duplex transmission, 8-bit/9-bit data formats, 4 different transmission modes - mode 0/1/2/3, baud rate generated by either external TIMER2 or the internal logic of the module, multiprocessor communication mode, automatic address recognition and given address & broadcast address.

To support low power applications, the LPUART is enhanced with one additional SCLK clock besides the original PCLK clock. The internal register configuration logic of LPUART module operates in the PCLK clock domain, the data transceiver logic in the SCLK clock domain. When the system enters the low power mode, the high frequency PCLK clock is disabled and the low frequency SCLK clock is enabled instead with data transmission and reception operating normally.

It can select PCLK, external low speed clock (XTL) or internal low speed clock (RCL) as the SCLK clock source. The SCLK clock supports 1/2/4/8/16/32/64/128 frequency pre-division when LPMODE = 1.

Notes:

- When LPMODE = 0, the LPUART receives the toggle output signal of the TIMER2 clock instead of the overflow signal, so it's required to enable the toggle output of TIMER2.

17.2 Structure Diagram

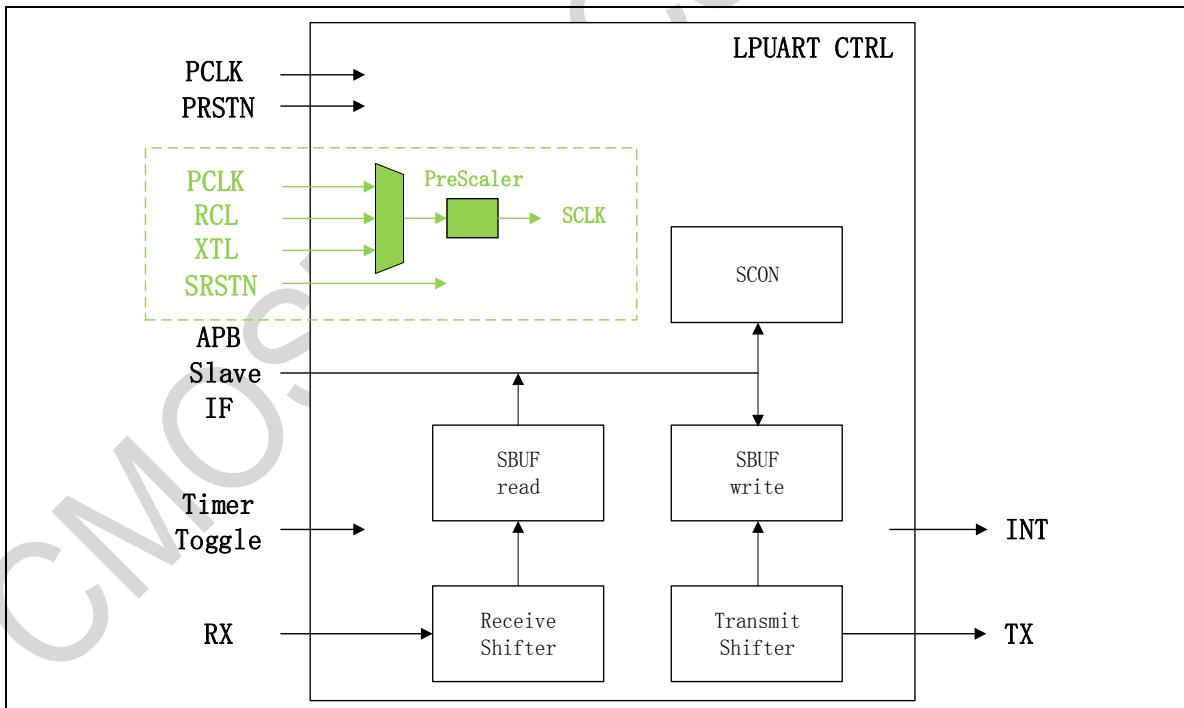


Figure 82 LPUART Structure Diagram

17.3 Operating Mode

The LPUART is enhanced with the LPMODE control bit compared with the general purpose UART (UART0/1). When this bit is

set to 1, only the operating mode 1/3 is supported, and the baud rate generation mechanism changes as well. Please refer to the following chapters for details.

17.3.1 Mode 0 (Synchronous Mode, Half Duplex)

When operating in mode 0, the UART operates in synchronous mode with a baud rate as 1/12 of the fixed SCLK clock. The transmitted data is transmitted through TXD, and the received data is received through RXD, thus RXD operates as input and output port at this point. The UART sync shift clock is output by the TXD, which is an output port in the case. Note that this mode can only be used as a master to send a synchronous shift clock, and it cannot be received as a slave from the outside. In this mode, the transmitted data bit width can only be 8 bits with neither start bit nor end bit.

Clear the LPUART_SCON.SM0 and LPUART_SCON.SM1 to enter mode 0.

When LPMODE = 1, it does not support mode 0.

17.3.1.1 Transmitting Data

When transmitting data, clear the LPUART_SCON.REN bit and write the data to the SBUF register. At this time, the transmitting data will be output from RXD (lower bit first) and the synchronous shift clock will be output from TXD.

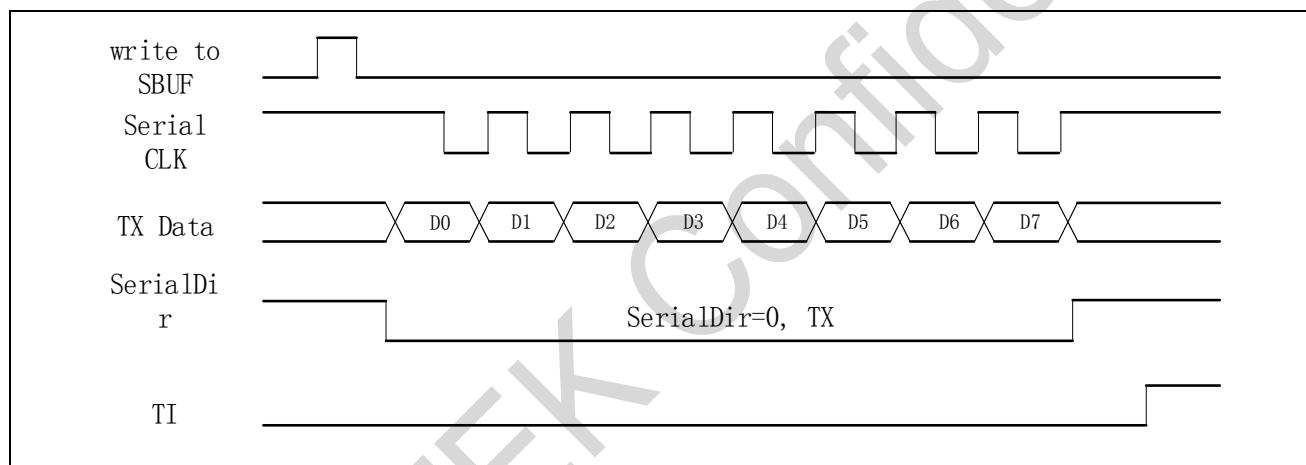


Figure 83. Transmitting Data in Mode 0

17.3.1.2 Receiving Data

When receiving data, set the LPUART_SCON.REN bit and clear the LPUART_ISR.RI bit to 0 then start receiving data on RXD (lower bit first). When receiving completes, the data can be read from the LPUART_SBUF register.

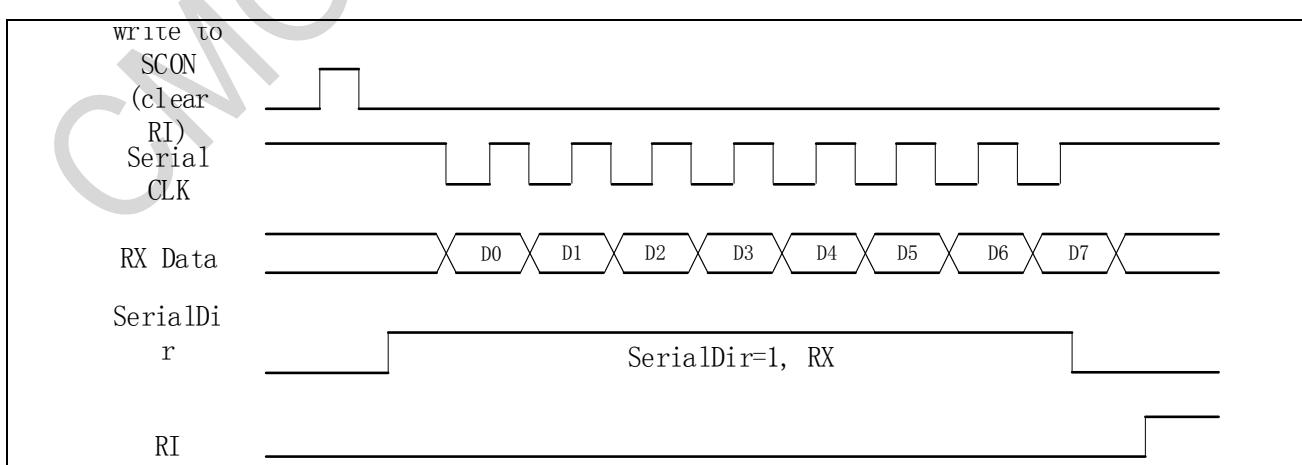


Figure 84. Receiving Data in Mode 0

17.3.2 Mode 1 (Asynchronous Mode, Full-duplex)

When operating in mode 1, the transmitted data is transmitted through TXD, and the received data is received through RXD. The data consists of 10 bits including the start bit as 0, followed by the 8 data bits (lower bit first), and the end bit as 1.

In this mode, the baud rate is generated by the timer module - TIMER2, which is programmable.

Clear UARTx_SCON.SM0 to 0 and UARTx_SCON.SM1 to 1 to enter mode 1.

When LPMODE = 1, it supports operating in mode 1, however the baud rate calculation changes. Please see more details in the chapters specific for baud rate programming.

17.3.2.1 Transmitting Data

When transmitting data, the transmitted data is written to the LPUART_SBUF register and transmitted from TXD (lower bit first), which is irrelevant to the value of LPUART_SCON.REN.

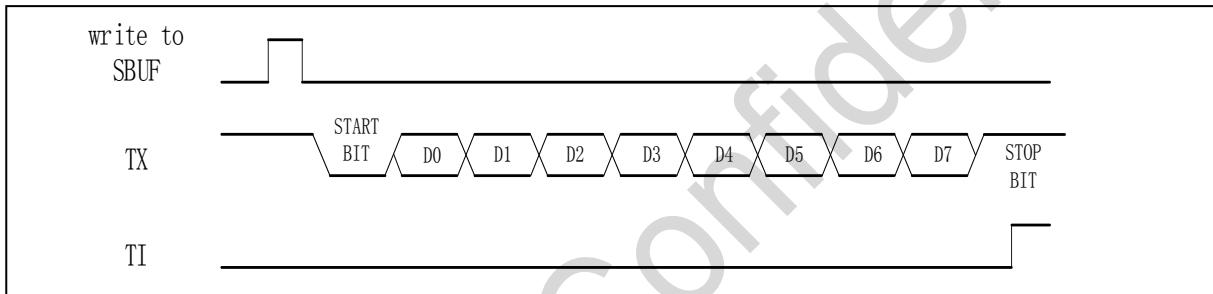


Figure 85. Transmitting Data in Mode 1

17.3.2.2 Receiving Data

When receiving data, set the LPUART_SCON.REN bit and clear the LPUART_ISR.RI bit to 0 then start receiving data on RXD (lower bit first). When receiving completes, the data can be read from the LPUART_SBUF register.

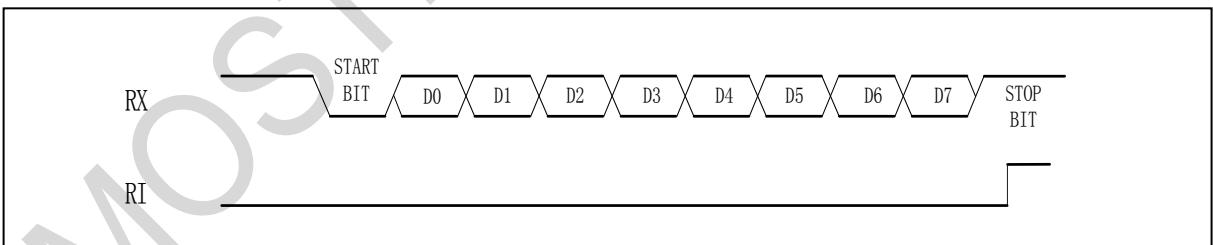


Figure 86. Receiving Data in Mode 1

17.3.3 Mode 2 (Asynchronous Mode, Full-duplex)

When operating in mode 2, the transmitted data is transmitted through TXD, and the received data is received through RXD. The data consists of 11 bits including the start bit as 0, followed by the 8 data bits (lower bit first), then 1 TB8 bit and the end bit. The extra TB8 bit is used in multiprocessor communication environments. When TB8 = 1, it indicates that a address frame is received. When TB8 = 0, it indicates a data frame is received. This bit can also be used as a parity bit when multiprocessor communication is not required.

In this mode, the baud rate can be generated independently with no external TIMER required.

Set UARTx_SCON.SM0 to 1 and UARTx_SCON.SM1 to 0 to enter mode 2.

Mode 2 is not supported when LPMODE = 1.

17.3.3.1 Transmitting Data

When transmitting data, write the data to the LPUART_SBUF register. The data will be transmitted from TXD then (lower bit first).

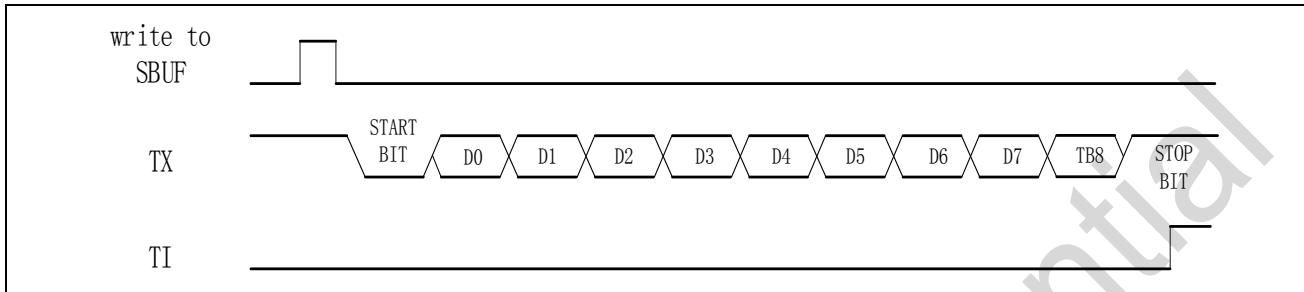


Figure 87. Transmitting Data in Mode 2

17.3.3.2 Receiving Data

When receiving data, set the LPUART_SCON.REN bit and clear the LPUART_ISR.RI bit to 0 then start receiving data on RXD (lower bit first). When receiving completes, the data can be read from the LPUART_SBUF register.

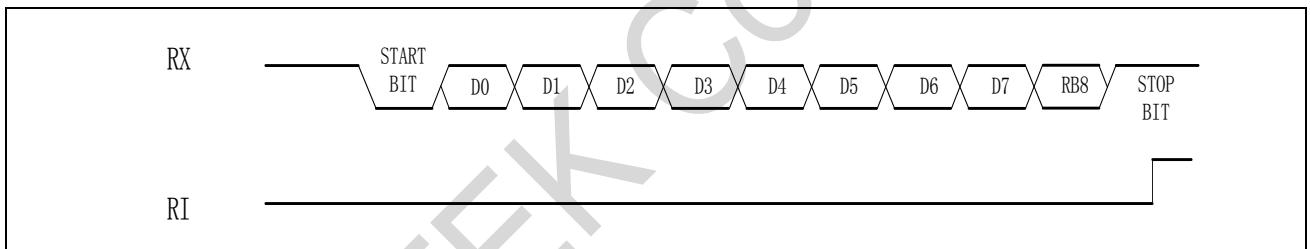


Figure 88. Receiving Data in Mode 2

17.3.4 Mode 3 (Asynchronous Mode, Full-duplex)

The data format, transmission timing and operation mode of mode 3 are the same as those of mode 2. However, unlike mode 2 with the baud rate being generated by the device itself independently, the baud rate in mode 3 is generated by TIMER instead, which is the only difference between them. The baud rate in mode 3 is programmable and it is generated in the same way as in mode 1.

Set UARTx_SCON.SM0 to 1 and UARTx_SCON.SM1 to 1 to enter mode 3.

When LPMODE = 1, it supports operating in mode 3, however the baud rate calculation changes. Please see more details in the chapters specific for baud rate programming.

17.4 Baud Rate Programming

17.4.1 Mode 0

- LPMODE = 0

When operating in mode 0, the baud rate is fixed at 1/12 of PCLK without timer support required.

- LPMODE=1

The mode is not supported when LPMODE = 1.

17.4.2 Mode 1/3

- LPMODE = 0

When operating in mode 1 or mode 3, the baud rate depends on the TIMER overflow time. The specific formula is shown below.

$$\text{Baud Rate} = \frac{(\text{SCON.DBAUD} + 1) * \text{Fsclk}}{32 * (65536 - \text{TM})}$$

Among them, LPUART_SCON.DBAUD represents double baud rate, Fsclk is SCLK clock frequency, and TM is TIMER count value. Note that TIMER must be configured as 16-bit auto-reload mode, and both the count register and the reload register must be written with the TM value.

- LPMODE = 1

When LPMODE is set to 1, the baud rate calculation is simplified as follows.

$$\text{BaudRate} = \frac{\text{Fsclk}}{\text{PreScale} * 4}$$

Among them, Fsclk represents SCLK clock frequency and PreScale represents pre-division factor.

17.4.3 Mode2

- LPMODE = 0

When operating in mode 2, the baud rate is fixed to the calculation result of the below formula.

$$\text{BaudRate} = \frac{(\text{SCON.DBAUD} + 1) * \text{Freq}}{64}$$

Among them, LPUART_SCON.DBAUD represents double baud rate and Freq represents the clock frequency of PCLK.

- LPMODE=1

This mode is not supported when LPMODE = 1.

17.5 Frame Error Detection

Mode 1/2/3 supports the frame error detection function with hardware automatically detecting whether the received frame data contains a valid stop bit. If the valid stop bit is not received, LPUART_ISR.FE is set. The LPUART_ISR.FE bit is set to 1 by hardware and cleared to 0 by software. If it is not cleared by software in time, even the subsequent received data contains the valid stop bit, it will not make this bit cleared to 0.

17.6 Multiprocessor Communication

To support multiprocessor communication function in mode 2/3, the 1-bit TB8/RB8 is added into its frame format. Set SCON.SM2 to 1 to enable the multiprocessor communication bit.

When the multiprocessor communication bit is enabled, the master can use LPUART_SCON.TB8 to distinguish whether the current frame is an address frame (LPUART_SCON.TB8 = 1) or a data frame (LPUART_SCON.TB8 = 0). When receiving data, the slave ignores the current received frame with the RB8 bit (bit 9) being 0. With the RB8 bit (bit 9) of the received frame being 1, it indicates an address frame received. The slave then judges whether the received address is equal to its own address. If there is a match, the slave sets both LPUART_SCON.RB8 to 1 and LPUART_ISR.RI to 1, indicating an address frame received with address matched. When the slave software checks out both LPUART_SCON.RB8 and LPUART_ISR.RI being 1, it clears the LPUART_SCON.SM2 bit to 0 and prepares the data frame receiving. If the address is not equal, it indicates that the master is not addressing the slave. UARTx_SCON.RB8 and UARTx_ISR.RI is retained as 0 by hardware. UARTx_SCON.SM2 bit is retained as 1 by software and the slave keeps on the address listening.

17.7 Automatic Address Recognition

When the multiprocessor communication bit is enabled (LPUART_SCON.SM2 is set to 1), the automatic address recognition function will be enabled then. This function is implemented in hardware level, so that the slave can detect each address frame received and have address matching. If it succeeds in matching the slave address, the receiver will set the receiving flag LPUART_ISR.RI, otherwise, no receiving flag set if it fails.

If necessary, the multiprocessor communication bit can also be enabled in mode 1, and the TB8 bit is replaced with the stop bit in the case. When the slave receives a matching address frame and a valid stop bit, LPUART_ISR.RI is set to 1.

To support automatic address recognition, the concepts of broadcast address and given address are introduced.

17.7.1 Given Address

The LPUART_SADDR register of the UART device is used for the given device-address of its own. The LPUART_SADEN register is for an address mask representing the bitmasks irrelevant to the specific address. When a certain bit in LPUART_SADEN is 0, it indicates this address bit is an irrelevant bit, meaning that this address does not participate in address matching. These irrelevant bits help increase the flexibility of addressing so that the master can address one or more slave devices simultaneously. Note that if a single matching address is required, the LPUART_SADEN register must be set to 8'hFF.

$$\text{GivenAddr} = \text{SADDR} \& \text{SADEN}$$

17.7.2 Broadcast Address

The broadcast address is used to address all slave devices at the same time. Generally, the broadcast address is 8'hFF.

$$\text{BroadCastAddr} = \text{SADDR} | \text{SADEN}$$

17.7.3 Example

Take the below case as an example.

Suppose a slave's UARTx_SADDR and UARTx_SADEN are configured as follows.

SADDR: 8'b01101001

SADEN: 8'b11111011

Then its given address and broadcast address are as follows:

Given: 8'b01101x01

Broadcast: 8'b11111x11

It can be seen that the master can address the slave with 4 addresses as follows:

8'b01101001 and 8'b01101101 (given address)

8'b11111011 and 8'b11111111 (broadcast address)

17.8 Tranceiver Buffer

17.8.1 Receiving Buffer

The LPUART receiver supports a receiving buffer with 1 frame length (8/9-bit), meaning that when a frame of data is received, the data in the receive buffer retains until it finishes receiving the *stop bit* of the next frame, thereafter the receiving buffer is updated with the new data frame.

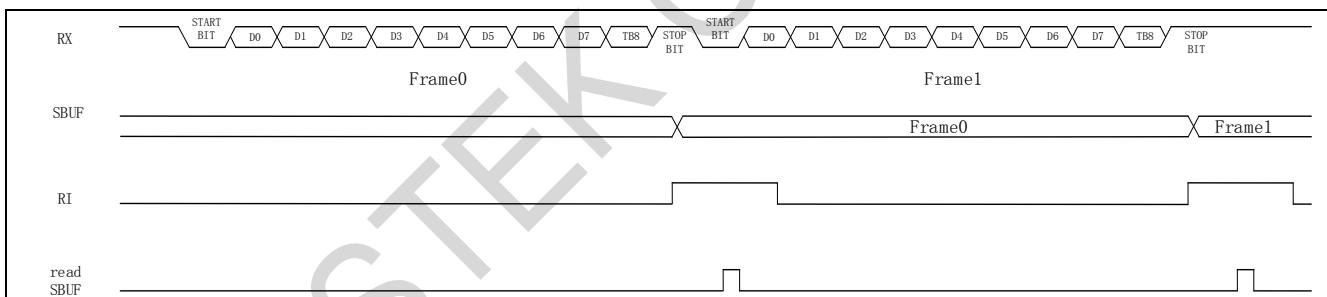
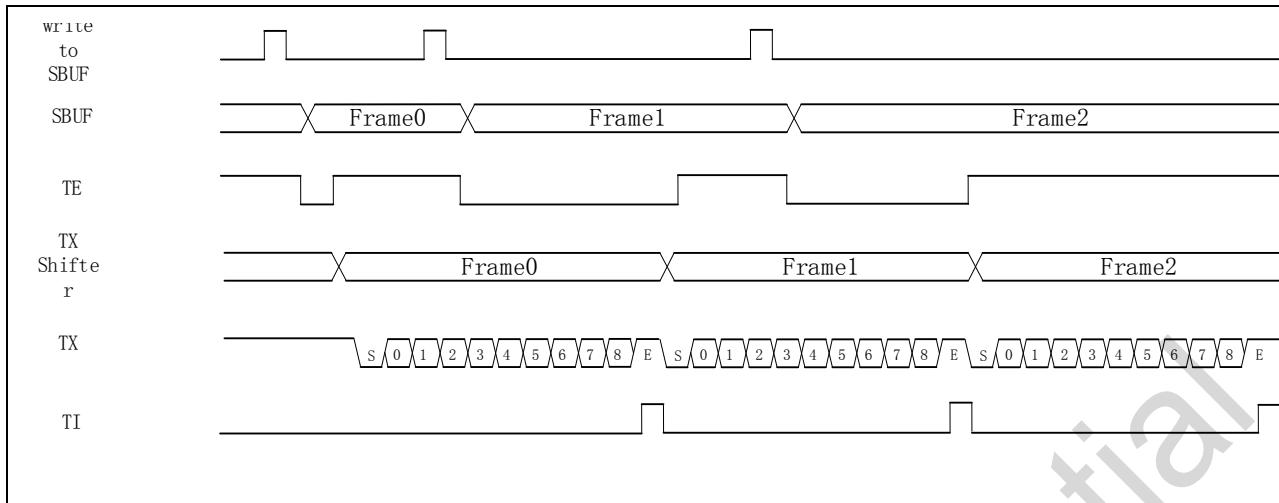


Figure 89. Receiving Buffer

17.8.2 Transmitting Buffer

The LPUART (UART2) transmitter supports a transmitting buffer with 1 frame length (8/9-bit), meaning that when the current data frame is transmitted from the transmitting shift register, CPU can write the next data frame to the transmitting buffer.

**Figure 90. Transmitting Buffer**

Among them, the register bit LPUART_ISR.TE is the transmitting buffer empty flag. The LPUART module contains only one frame (8/9bits) for transmitting buffer, so the hardware automatically masks software writing to the LPUART_SBUF register if the LPUART_ISR.TE bit is 1 unless the LPUART_ISR.TE bit changes to 0. The software must judge the state (0, or 1) of the LPUART_ISR.TE bit before filling the transmitting data into the LPUART_SBUF register, otherwise the transmitting data will be lost.

17.9 Register

LPUART base address: 0x4000 0200

Table 608. LPUART Register

Register	Offset address	Description
LPUART_SBUF	0x00	Data register
LPUART_SCON	0x04	Control register
LPUART_SADDR	0x08	Address register
LPUART_SADEN	0x0C	Address mask register
LPUART_ISR	0x10	Interrupt flag register
LPUART_ICR	0x14	Interrupt flag clearing register

17.9.1 Data Register (LPUART_SBUF)

Offset address: 0x00

Reset value: 0x0000 0000

Table 609. Data Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 610. Data Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

Reserved	SBUF
R	R/W

Table 611. Data Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	SBUF	The transmitted data is written to the register when transmitting data. When receiving data, the received data is read from the register after receiving completes. Note that, for this register, the actual read value is read from RXBuffer and the actual writing value is written to TXShifter.

17.9.2 Control Register (LPUART_SCON)

Offset address: 0x04

Reset value: 0x0000 E000

Table 612. Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 613. Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRS	SCLKSEL	LPMODE	DBAUD	TEEN	SM01	SM2	REN	TB8	RB8	TIEN	RIEN				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 614. Control Register (3)

Bit	Flag	Description
31:16	Reserved	
15:13	PRS	Transmitting clock - SCLK pre-division selection. 000: div128, 001: div64, 010: div32, 110: div2, 111: div1. PRS[2:0] is enabled only if LPMODE = 1. PRS[2:0] is disabled when LPMODE = 0, meaning that pre-division is not performed.
12:11	SCLKSEL	Transmission clock - SCLK selection. 00,01: PCLK, 10: XTL, 11: RCL.
10	LPMODE	Low power mode. 0: normal operating mode, 1: low power mode.
9	DBAUD	Double baud rate. 0: single baud rate, 1: double baud rate.
8	TEEN	Empty transmission buffer enabling. 0: disable, 1: enable.
7:6	SM01	Operating mode. 00: mode 0, 01: mode 1, 10: mode 2, 11: mode3.
5	SM2	Multiprocessor communication. 0: disable, 1: enable.

Bit	Flag	Description
4	REN	Receiving enabling. In mode 0, 0: transmitting, 1: receiving. In other mode, 0: transmitting, 1: receiving/transmitting.
3	TB8	Transmission TB8 bit.
2	RB8	Receiving RB8 bit.
1	TIEN	Transmitting completion interrupt enabling. 0: disable, 1: enable.
0	RIEN	Receiving completion interrupt enabling. 0: disable, 1: enable.

17.9.3 Address Register (LPUART_SADDR)

Offset address: 0x08

Reset value: 0x0000 0000

Table 615. Address Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 616. Address Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								SADDR							
R								R/W							

Table 617. Address Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	SADDR	Slave device-address register

17.9.4 Address Mask Register (LPUART_SADEN)

Offset address: 0x0C

Reset value: 0x0000 0000

Table 618. Address Mask Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 619. Address Mask Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

Reserved	SADEN
R	R/W

Table 620. Address Mask Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	SADEN	Slave device-address mask register

17.9.5 Interrupt Flag Register (LPUART_ISR)

Offset address: 0x10

Reset value: 0x0000 0008

Table 621. Interrupt Flag Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 622. Interrupt Flag Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRS												TE	FE	TI	RI
R												R	R	R	R

Table 623. Interrupt Flag Register (3)

Bit	Flag	Description
31:4	Reserved	
3	TE	Empty transmission buffer interrupt flag. It is set and cleared by hardware. Notes: when the bit value is 0, the hardware disables the software write operation of SBUF automatically. 1: TE interrupt is enabled. 0: TE interrupt is disabled.
2	FE	Frame error received flag. It is set by hardware and cleared by software. 1: FE interrupt is disabled. 0: FE interrupt is enabled
1	TI	Transmitting completion interrupt flag. It is set by hardware and cleared by software. 1: TI interrupt is disabled. 0: TI interrupt is enabled.
0	RI	Receiving completion interrupt flag. It is set by hardware and cleared by software. 1: RI interrupt is enabled. 0: RI interrupt is disabled.

17.9.6 Interrupt Flag Clearing Register (LPUART_ICR)

Offset address: 0x14

Reset value: 0x0000 0007

Table 624. Interrupt Flag Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R															

Table 625. Interrupt Flag Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRS															
R															

Table 626. Interrupt Flag Clearing Register (3)

Bit	Flag	Description
31:3	Reserved	
2	FECLR	Clearing frame error received flag. Write 0 for clearing and writing 1 is disabled.
1	TICLR	Clearing transmission completion interrupt flag. Write 0 for clearing and writing 1 is disabled.
0	RICLR	Clearing receiving completion interrupt flag. Write 0 for clearing and writing 1 is disabled.

18 I2C

18.1 I2C Introduction

I2C is a two-wire bidirectional serial bus providing a simple and efficient way for data exchange between devices. The I2C standard is a true multiprocessor bus with a collision detection mechanism and an arbitration mechanism. It prevents data collisions when two or more processors requesting the same control bus at the same time.

The I2C bus controller can satisfy the various specifications of the I2C bus and supports all transmission modes of I2C bus communication.

The I2C bus uses SCL (serial clock bus) and SDA (serial data bus) of the connected device for information transfer.

Data is transferred byte and byte in synchronous mode between the master and the slave controlled by the SCL clock line with 8 bits contained in each byte. One data bit is transferred in 1 SCL clock pulse. The data transfer starts from the highest MSB bit. Each byte is transferred followed by an acknowledging bit. Since each bit is sampled when SCL is high, the SDA line can be changed only when SCL is low, other words, SDA remains stable when SCL is high. If a transition on the SDA line occurs in case of SCL high, it will be regarded as a command interrupt (START or STOP), which ensures the I2C logic handling the transfer of bytes autonomously. It supports the continuous track of the serial transfer, and offers a state register (I2C_STAT) as well which reflects the state of the I2C bus controller and the I2C bus.

18.2 I2C Major Features

The I2C controller supports the following features.

- Support 4 operating modes: master transmitting, master receiving, slave transmitting and slave receiving.
- Support 3 operating rates: standard (100 Kbps), fast (400 Kbps) and high speed (1 Mbps).
- Support 7-bit addressing function.
- Support noise filtering.
- Support broadcast address.
- Support interrupt state query function.

18.3 I2C Protocol Description

Usually the standard I2C transmission protocol consists of the following 4 parts.

1. START signal or repeated START signal
2. Slave address transfer and R/W bit transfer
3. Data transmission
4. STOP signal

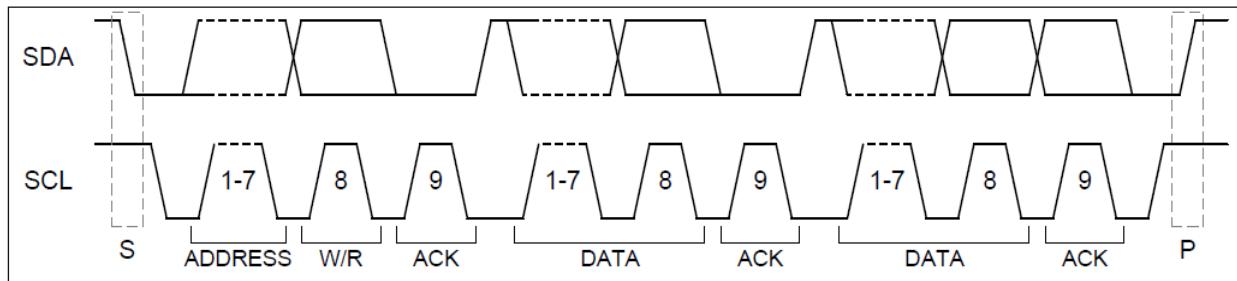


Figure 91. I2C Transmission Protocol

18.3.1 Data Transmission on the I2C Bus

The master transmits a 7-bit address (one byte) and the slave receive it. The transfer direction does not change yet.

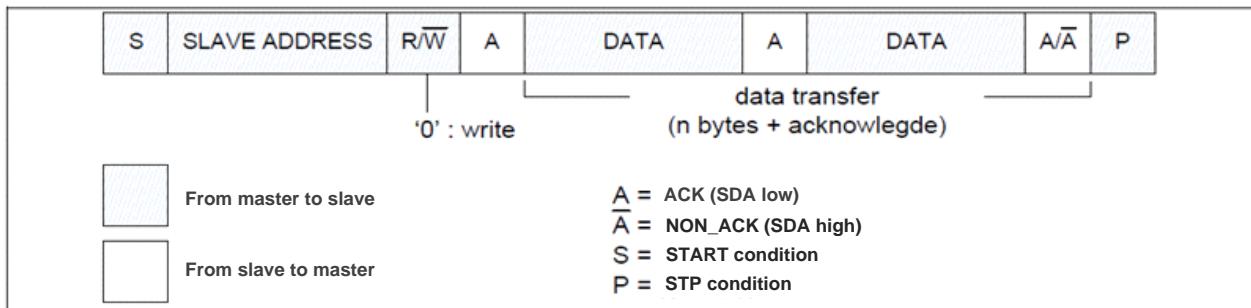


Figure 92. Master Transmitting Data to Slave

After the first byte transmission, the master reads the data from the slave (the content is the slave address) and the transfer direction changes.

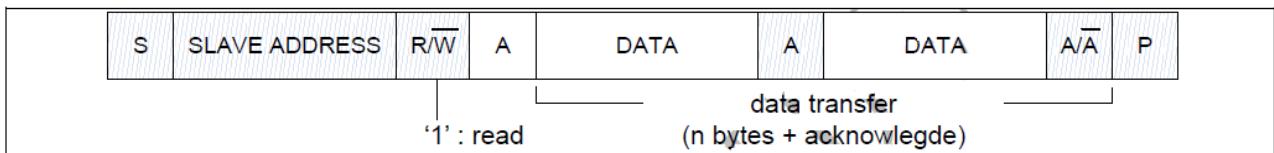


Figure 93. Master Read Address From Slave

18.3.2 START Signal or Repeated START Signal

When the bus is idle, meaning that no master initiates a transfer request on the bus (SCL and SDA lines are both high), the master can initiate a transfer request by sending a START signal then.

- **START signal**

START signal is usually represented as S-bit. When the SCL line is high, the signal on the SDA line is from high to low, indicating that the START signal is generated on the bus, and the new transmission starts.

- **Repeated START signal**

Repeated START signal (Sr) represents that there is no STOP signal between two START signals. The master uses this method

to communicate with another slave or the same slave in different directions of transmission (eg, a transition from the writing device to the reading slave) without releasing the bus.

- **STOP signal**

The master sends a STOP signal to the bus to end the data transfer. The STOP signal, represented as P-bit, is defined as a low-to-high signal which appears on the SDA line when the SCL line is high.

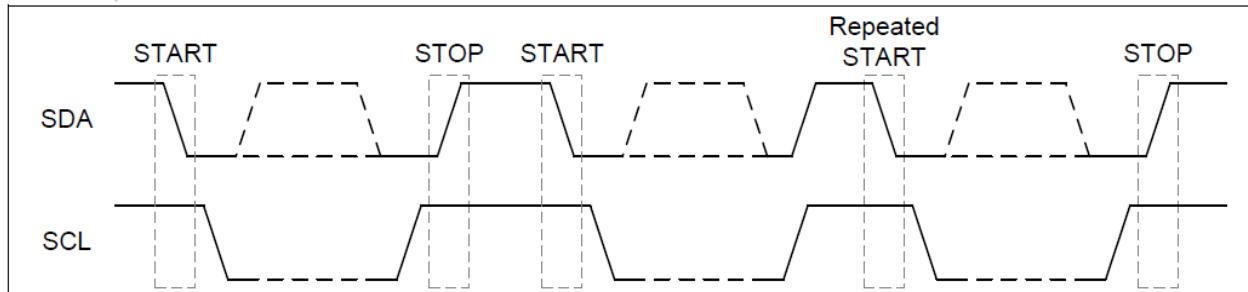


Figure 94. START and STOP Conditions

18.3.3 Slave Address Transfer

When a START signal is followed by a slave address, the master immediately transmits the first bit of the data. This is a 7-bit calling address followed by a RW bit, which controls the direction of the slave's signal transmission. As no 2 slaves share the same address, thus only the slave addressed by the master acknowledges it by setting SDA to low in the 9th SCL clock cycle.

18.3.4 Data Transfer

When the slave address is identified successfully, it can perform byte-by-byte data transfer according to the transmission direction determined via RW. Each transmitted byte contains a acknowledging signal in the 9th clock cycle. If a slave generates non-acknowledgment signal (NON-ACK), the master can either generate a STOP signal to exit the data transmission or a repeated START signal to start a new round of data transmission.

When the master acts as a receiving device, if a non-acknowledgment signal (NON-ACK) occurs, the slave releases the SDA line so that the master can generate a STOP signal or repeated START signal.

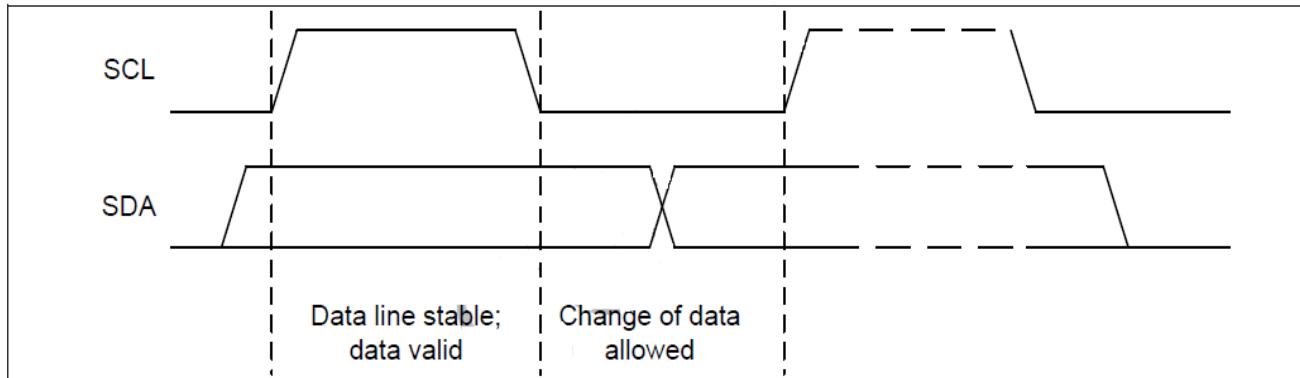


Figure 95. I2C Bit Transfer on the Bus

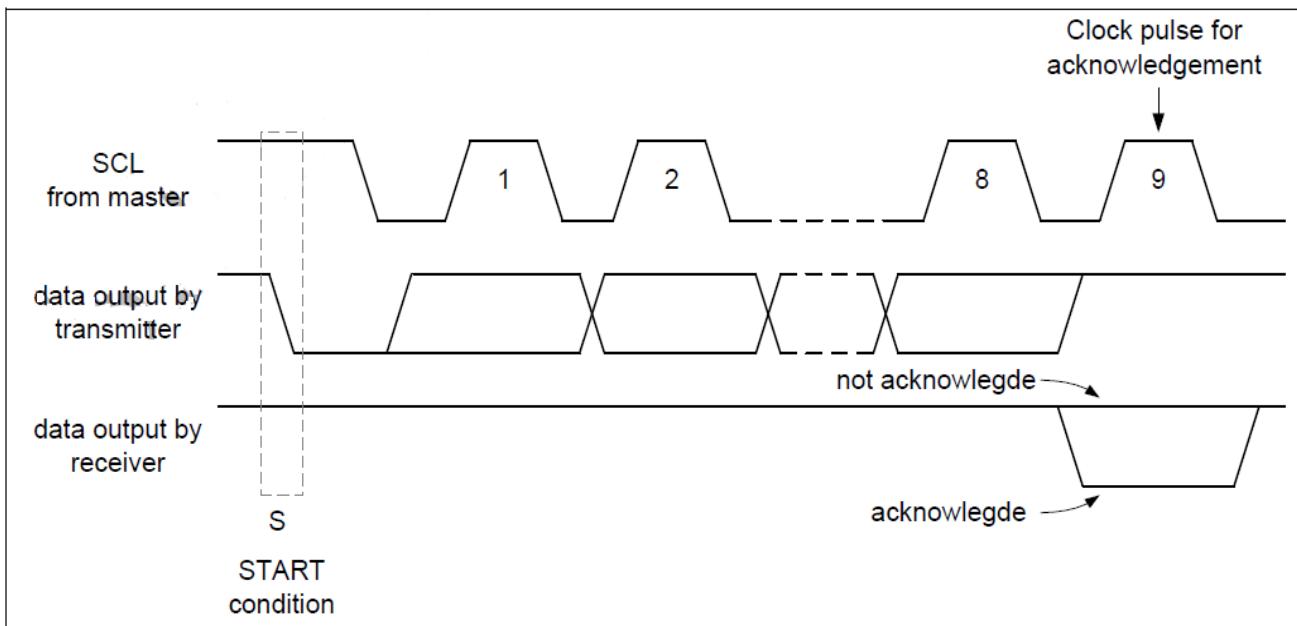


Figure 96. I2C Acknowledge Signal on the Bus

18.4 I2C Function Description

The I2C bus uses 2 wires to transfer information between devices connected to the buses - SCL (serial clock line) and SDA (serial data line). Since there are only non-directional ports, the I2C component needs to use the open-drain buffers of pins. Each device connected to the bus can be addressed by software via a specific address. The I2C standard is a true multiprocessor bus with both collision detection and arbitration mechanism available. It prevents data collisions when two or more processors start transmitting data at the same time. Filtering logic can filter glitches on the data bus to protect data integrity.

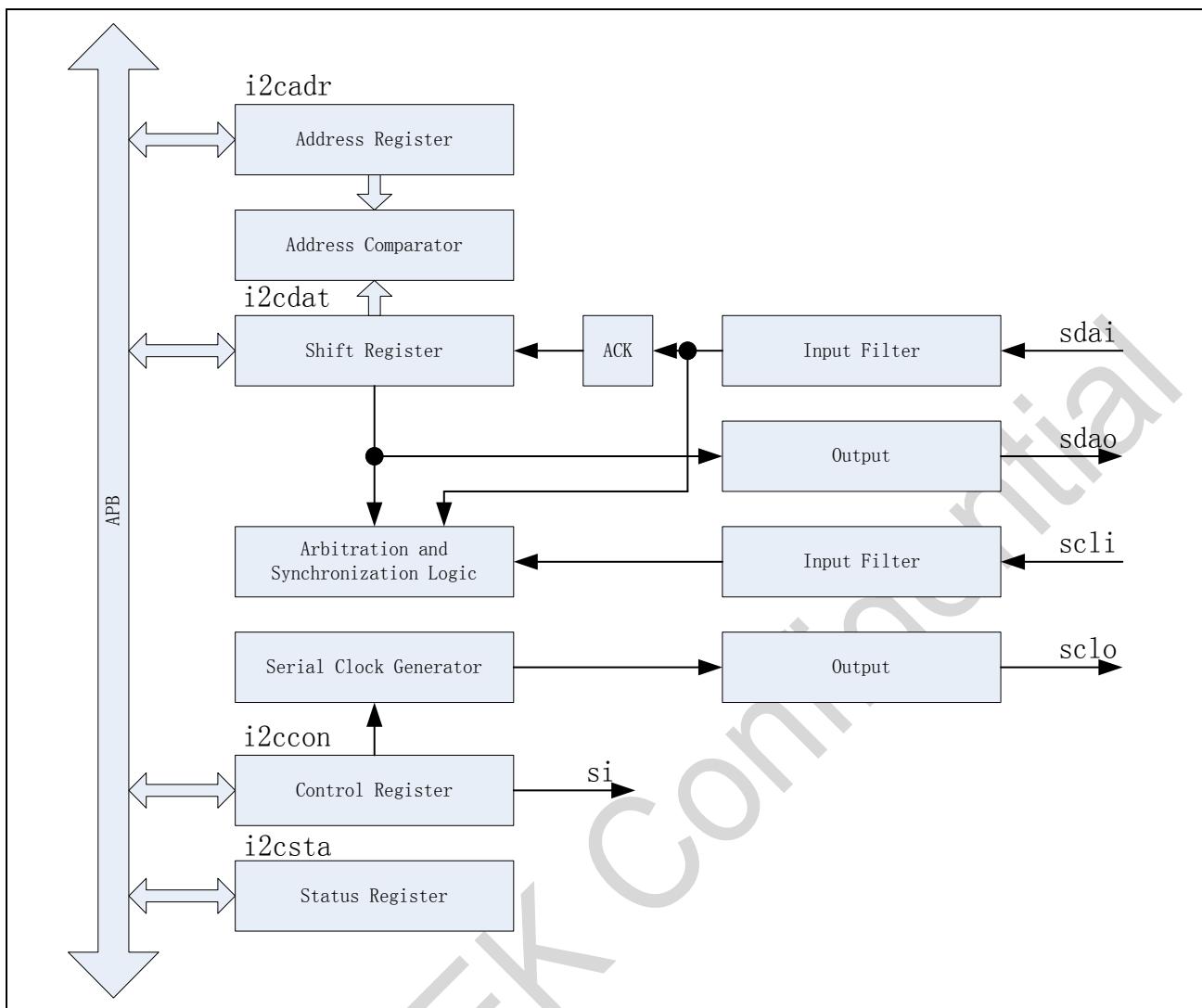


Figure 97. I2C Function Block Diagram

18.4.1 I2C Operating Mode

I2C components can fulfill 8-bit bidirectional data transmission, with transmission rates of 100 kbit/s in standard mode, 400 kbit/s in high speed mode and 1 Mbit/s in ultra-high speed mode. The operating modes are as follows.

- **Master transmitting mode**

Serial data is sent through the SDA port and when serial clock signal is sent through the SCL port.

- **Master receiving mode:**

Serial data is received through the SDA port when serial clock signal is sent through the SCL port.

- **Slave receiving mode:**

Serial data and serial clock are received through the SDA port and SCL port respectively.

- **Slave transmitting mode:**

Serial data is sent through SDA port when serial clock is received through the SCL port.

18.4.2 Arbitration and Synchronization Logic

In the master transmitting mode, the arbitration logic checks whether each transmitted logic 1 actually appears on the bus. If another device on the bus undoes a logic 1 and pulls the SDA line low, the arbitration is lost and the I2C module changes from the master transmitting mode to the slave receiving mode immediately, however the I2C module continues outputting clock pulses (on SCL) until the current serial bytes transmission completes.

The arbitration may also be lost in the master receiving mode. This case occurs only when the I2C module is returning a non-acknowledgement (logic 1) to the bus. In this case, the arbitration is lost when another device on the bus pulls the signal low. As it only appears at the end of the serial bytes, the I2C module will no longer generate clock pulses.

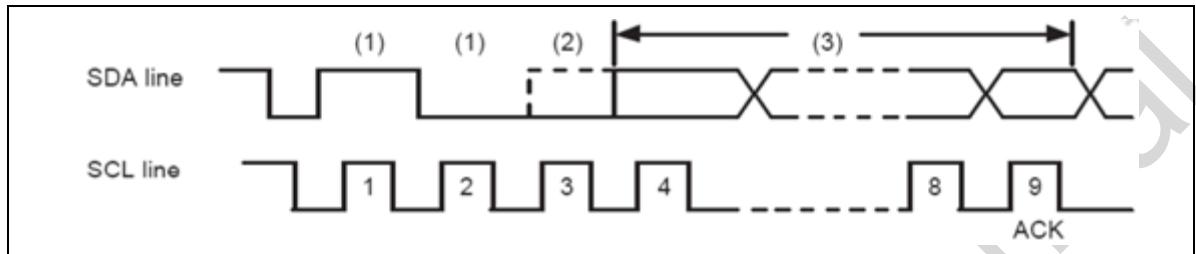


Figure 98. Arbitration on the I2C Bus

The arbitration flow is as follows.

1. Another device transmits serial data.
2. Another device undoes a logic 1 (dashed line) sent by the I2C master by pulling SDA low. The arbitration is lost and I2C enters the slave receiving mode.
3. At this time, I2C is in the slave receiving mode, however the I2C module continues outputting clock pulse until the current byte transmission completes. I2C will not generate clock pulse for the next byte transmission. Once wins the arbitration, the data transmission on the SDA is initiated by the new master.

Synchronization logic synchronizes the serial clock generator with the clock pulse of another device on the SCL line. When two or more masters generating clock pulses, the high level period is subject to the device that produces the shortest high level and the low period is subject to the device that produces the longest low level.

18.4.3 Serial Clock Generator

The serial clock generator uses an 8-bit counter as the baud rate generator. The frequency relationship between the SCL signal and the pclk signal is $F_{SCL} = F_{PCLK}/8*(N+1)$.

The table below shows the frequency value of the SCL signal when the pclk is at various frequencies with a division factor range of 1~7.

Table 627. I2C I2C Clock Signal Baud Rate

Frequency (kHz)	1	2	3	4	5	6	7
1000	62	41	31	25	20	17	15
2000	125	83	62	50	41	35	31
4000	250	166	125	100	83	71	62
6000	375	250	187	150	125	107	93

8000	500	333	250	200	166	142	125
10000	625	416	312	250	208	178	156
12000	750	500	375	300	250	214	187
14000	875	583	437	350	291	250	218
16000	1000	666	500	400	333	285	250

18.4.4 Input Filter

The input signal is synchronized with the clock signal (clk), and spike signals shorter than 3 clock cycles are filtered out. Each filter consists of 3 triggers. The first trigger is used to latch the input signal directly and load the data into the shift register formed by the other two triggers. When the status of the second and third triggers is 11 or 00, their internal filter signals are set to 1 or 0 correspondingly.

18.4.5 Address Comparator

The I2C comparator compares its own slave address to the received 7-bit slave address. It supports programming its own slave address using the I2C_ADDR register, which is compared to both the first received 8-bit byte and the general call address (0x00) according to the GC bit of the I2C_ADDR register. If either of the comparing result is the same, the *si* bit of the I2C_CR register will be set with an interrupt request generated.

18.4.6 Interrupt Generator

When all the 4 modes of the I2C module are used, there are 26 possible bus states. When I2C enters the 25 states among the 26 ones, the *si* flag of the I2C_CR register is set by hardware. The only state in which the *si* bit is not set is F8h, which indicates that there is no valid relevant state information. The *si* flag must be cleared by software. In order to clear the *si* bit, it must write 0 to this bit. Writing 1 in *si* does not change the value of *si*. To determine the actual interrupt source of the interrupt, the interrupt service routine will query the I2C state register before clearing the *si* flag.

18.4.7 I2C Master Transmitting Mode

It must set *ens* 1 to enable the I2C module. If the *aa* bit is reset, the I2C module will not acknowledge its own slave address or general call address when another device is becoming the bus master. In other words, if the *aa* bit is reset, the I2C interface cannot enter slave mode. *Sta*, *sto* and *si* must be reset to enter slave mode in the case.

In this case, it can enter the master transmitting mode by setting the *sta* bit. Once the bus is idle, the I2C logic will test the I2C bus immediately and generate a START signal. When the START signal is transmitted, the serial interrupt flag (*si*) is set and the state code in the state register (I2C_STAT) is 0x08. The interrupt service routine uses this state code to enter the corresponding state service program, loading the slave address and data direction bits (SLA+W) into I2C_DATA. The *si* bit of I2C_CR must be reset before the serial transfer continues.

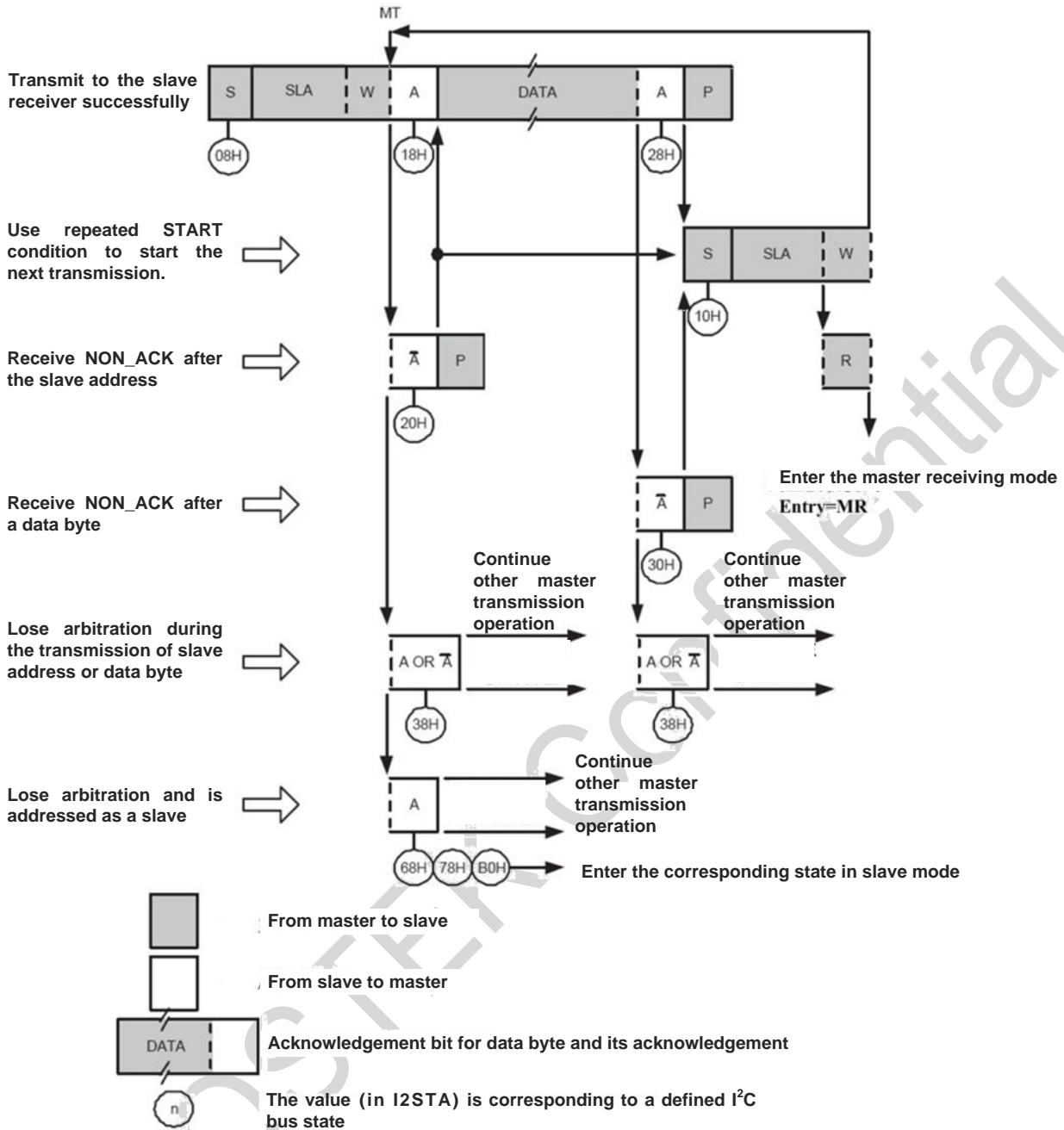
When the slave address and direction bits are transmitted and an acknowledge bit is received, the serial interrupt flag (*si*) is set again. I2C_STAT contains a series of different state codes, with 0x18, 0x20 or 0x38 for the master mode and 0x68, 0x78 or 0xB0 for the slave mode (meaning *aa* is logic 1). The operation corresponding to each state code is detailed in the following table. After the repeated START signal (state 0x10) is transmitted, the I2C module switches to the master receiving mode by loading SLA+R into I2C_DATA.

Table 628. I2C Master Transmitting Mode State Table

State code	I2C Bus and Hardware State	Application Software Respond				Next Action of I2C Hardware	
		Write/Read I2C_DATA	Write I2C_CR				
			sta	sto	si	aa	
08H	START is sent.	Load SLA+W	X	0	0	X	Send SLA+W and receive ACK
10H	Repeated START is sent	Load SLA+W	X	0	0	X	Same as above
		Load SLA+R	X	0	0	X	Send SLA+R and I2C switches to master receiving mode automatically
18H	SLA+W is sent and ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset.
20H	SLA+W is sent and NON-ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset.
28H	Data in I2C_DATA is sent and ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA action	1	0	0	X	Send repeated START signal
		related	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset
30H	Data in I2C_DATA is sent	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset

State code	I2C Bus and Hardware State	Application Software Respond				Next Action of I2C Hardware	
		Write/Read I2C_DATA	Write I2C_CR				
			sta	sto	si	aa	
08H	START is sent.	Load SLA+W	X	0	0	X	Send SLA+W and receive ACK
10H	Repeated START is sent	Load SLA+W	X	0	0	X	Same as above
		Load SLA+R	X	0	0	X	Send SLA+R and I2C switches to master receiving mode automatically
18H	SLA+W is sent and ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset.
20H	SLA+W is sent and NON-ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset.
28H	Data in I2C_DATA is sent and ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA action	1	0	0	X	Send repeated START signal
		related	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset
30H	Data in I2C_DATA is sent	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset
38H	The arbitration is lost when SLA+R/W or	No I2C_DATA related action	0	0	0	X	The I2C bus is released and it enters the mode in which slave addressing is not available.

State code	I2C Bus and Hardware State	Application Software Respond				Next Action of I2C Hardware	
		Write/Read I2C_DATA	Write I2C_CR				
			sta	sto	si	aa	
08H	START is sent.	Load SLA+W	X	0	0	X	Send SLA+W and receive ACK
10H	Repeated START is sent	Load SLA+W	X	0	0	X	Same as above
		Load SLA+R	X	0	0	X	Send SLA+R and I2C switches to master receiving mode automatically
18H	SLA+W is sent and ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset.
20H	SLA+W is sent and NON-ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset.
28H	Data in I2C_DATA is sent and ACK is received	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA action	1	0	0	X	Send repeated START signal
		related	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset
30H	Data in I2C_DATA is sent data bytes are sent	Load data bytes	0	0	0	X	Send data bytes and receive ACK
		No I2C_DATA related action	1	0	0	X	Send repeated START signal
		No I2C_DATA related action	0	1	0	X	Send STOP signal and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP signal first and send START signal then. Sto flag is reset
		No I2C_DATA related action	1	0	0	X	Send START signal when the I2C is idle

Figure 99. I²C Master Transmitting State Diagram

18.4.8 I²C Master Receiving Mode

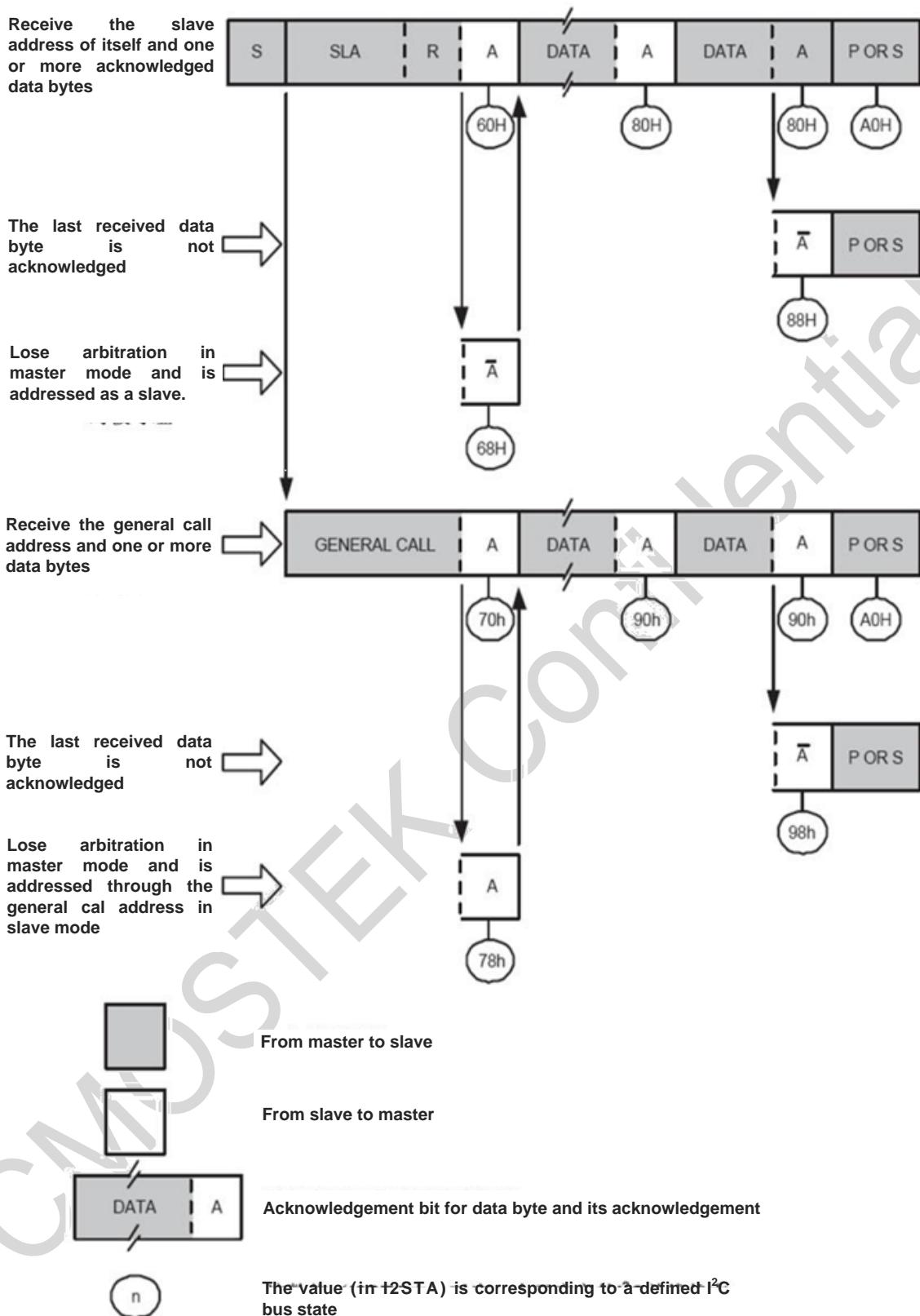
In the master receiving mode, the data bytes received by the master are from the slave transmitter. The transfer is initiated in the same way as in the master transmitting mode. When the START signal is transmitted, the interrupt service routine must load the 7-bit slave address and data direction bits (SLA+R) into I²C_DATA. The *si* bit in I²C_CR must be cleared before continuing the serial transfer.

When the slave address and data direction bits are transmitted and an ACK bit is received, the serial interrupt flag *si* is set again. In this case, I²C_STAT contains one of a series of different state codes, with 0x18, 0x20 or 0x38 for the master mode and 0x68, 0x78 or 0xB0 for the slave mode (meaning aa is logic 1). The operation corresponding to each state code is detailed in the table

below. After the repeated START signal (state 0x10) is sent, the I2C module switches to the master transmitting mode by loading SLA+W into I2C_DATA.

Table 629. I2C Master Receiving Mode State Table

State code	I2C Bus and Hardware State	Application Software Respond				Next Action of I2C Hardware	
		Write/Read I2C_DATA	Write I2C_CR				
			sta	sto	si	aa	
08H	START is sent	Load SLA+R	X	0	0	X	Receive SLA + R and receive ACK
10H	Repeated START is sent	Load SLA+R	X	0	0	X	Same as above
		Load SLA+W	X	0	0	X	Receive SLA + W and I2C switches to master receiving mode automatically
38H	The arbitration is lost in NON_ACK	No I2C_DATA related action	0	0	0	X	The I2C bus is released. Enter slave mode.
		No I2C_DATA related action	1	0	0	X	Initiate the START condition when the bus is idle.
40H	SLA+R is sent and ACK is received	No I2C_DATA related action	0	0	0	0	Receive data bytes and return NON_ACK
		No I2C_DATA related action	0	0	0	1	Receive data bytes and return ACK
48H	SLA+R is sent and NON_ACK is received	No I2C_DATA related action	1	0	0	X	Send repeated START condition
		No I2C_DATA related action	0	1	0	X	Send STOP condition and sto flag is reset
		No I2C_DATA related action	1	1	0	X	Send STOP condition first and send START condition then. Sto flag is reset.
50H	Data bytes is received and ACK returns	Read data byte	0	0	0	0	Receive data bytes and return NON_ACK
		Read data byte	0	0	0	1	Receive data bytes and return ACK
58H	Data bytes is received and NON_ACK returns	Read data byte	1	0	0	X	Send repeated START condition
		Read data byte	0	1	0	X	Send STOP condition and sto flag is reset

Figure 100. I²C Master Receiving State Diagram

18.4.9 I2C Slave Receiving Mode

In slave receiving mode, the data bytes received by the slave is from the master transmitter. The higher 7 bits are the address the I2C module responds to during the master addressing process. If LSB(GC) is set, the I2C module responds to the general call address (0x00), otherwise the general call address is ignored.

The setting of the I2C bus rate does not affect the I2C module in slave mode. I2CEN must be set to enable the I2C module. The aa bit must be set to enable the I2C module to acknowledge its own slave address or general call address. Sta, sto and si must be reset.

When I2C_ADDR and I2C_CR are Initialized, the I2C module waits until it is addressed by the slave address, followed by the data direction bit judgement. In order to operate in the slave receiving mode, the data direction bit must be 0 (W). After receiving its own slave address and W bit, the serial interrupt flag (si) is set and a valid state code can be read from I2C_STAT. This state code is used as a vector for the state server. The corresponding operation of each state code is shown in the table below. The slave receiving mode can also be entered if the I2C module loses arbitration in master mode (refer to the descriptions of states 0x68 and 0x78 for more details).

If the aa bit is reset during transmission, the I2C module returns a NON_ACK (logic 1) to SDA after the next data byte is received. When aa is reset, the I2C module does not respond to its own slave address or general call address. However, the I2C bus is still being monitored and address recognition can be recovered at any time by setting aa, meaning that the aa bit can separates the I2C module from the I2C bus temporarily.

Table 630. I2C Slave Receiving Mode State Table

State code	I2C Bus and Hardware State	Application Software Respond				Next Action of I2C Hardware	
		Read/Write I2C_DATA	Write I2C_CR				
			sta	sto	si	aa	
60H	The SLA + W of its own is received and ACK is returned	No I2C_DATA related action	X	0	0	0	Receive data bytes and return NON_ACK
		No I2C_DATA related action	X	0	0	1	Receive data bytes and return ACK
68H	The arbitration is lost in the SLA + R/W of master mode. The SLA + W of its own is received. ACK is returned.	No I2C_DATA related action	X	0	0	0	Receive data bytes and return NON_ACK
		No I2C_DATA related action	X	0	0	1	Receive data bytes and return ACK
70H	The general call address (0x00) is received and ACK is returned.	No I2C_DATA related action	X	0	0	0	Receive data bytes and return NON_ACK
		No I2C_DATA related action	X	0	0	1	Receive data bytes and return ACK
78H	The arbitration is lost in the SLA + R/W of master mode. The general call address (0x00) is received and	No I2C_DATA related action	X	0	0	0	Receive data bytes and return NON_ACK
		No I2C_DATA related action	X	0	0	1	Receive data bytes and return ACK

	ACK is returned.					
80H	The previous addressing uses its own slave address. Data bytes are received and ACK is returned.	No I2C_DATA related action	X	0	0	0 Receive data bytes and return NON_ACK
		No I2C_DATA related action	X	1	0	1 Receive data bytes and return ACK
88H	The previous addressing uses its own slave address. Data bytes are received and NON_ACK is returned.	Read data byte	0	0	0	0 Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.
		Read data byte	0	0	0	1 Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.
		Read data byte	1	0	0	0 Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.
		Read data byte	1	0	0	1 Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the START condition when the bus is idle.
90H	The previous addressing uses the general call address. Data bytes are received and ACK is returned.	Read data byte	1	0	0	X Receive data bytes and return NON_ACK
		Read data byte	0	1	0	X Receive data bytes and return ACK
98H	The previous addressing uses the general call address. Data bytes are received and NON_ACK is returned.	Read data byte	0	0	0	0 Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.
		Read data byte	0	0	0	1 Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.
		Read data byte	1	0	0	0 Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the

						START condition when the bus is idle.	
		Read data byte	1	0	0	1	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the START condition when the bus is idle.
A0H	Received a STOP condition or a repeated START condition when using static addressing from the receive/slave transmitting mode	No I2C_DATA related action	0	0	0	0	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.
		No I2C_DATA related action	0	0	0	1	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.
		No I2C_DATA related action	1	0	0	0	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the START condition when the bus is idle.
		No I2C_DATA related action	1	0	0	1	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the START condition when the bus is idle.

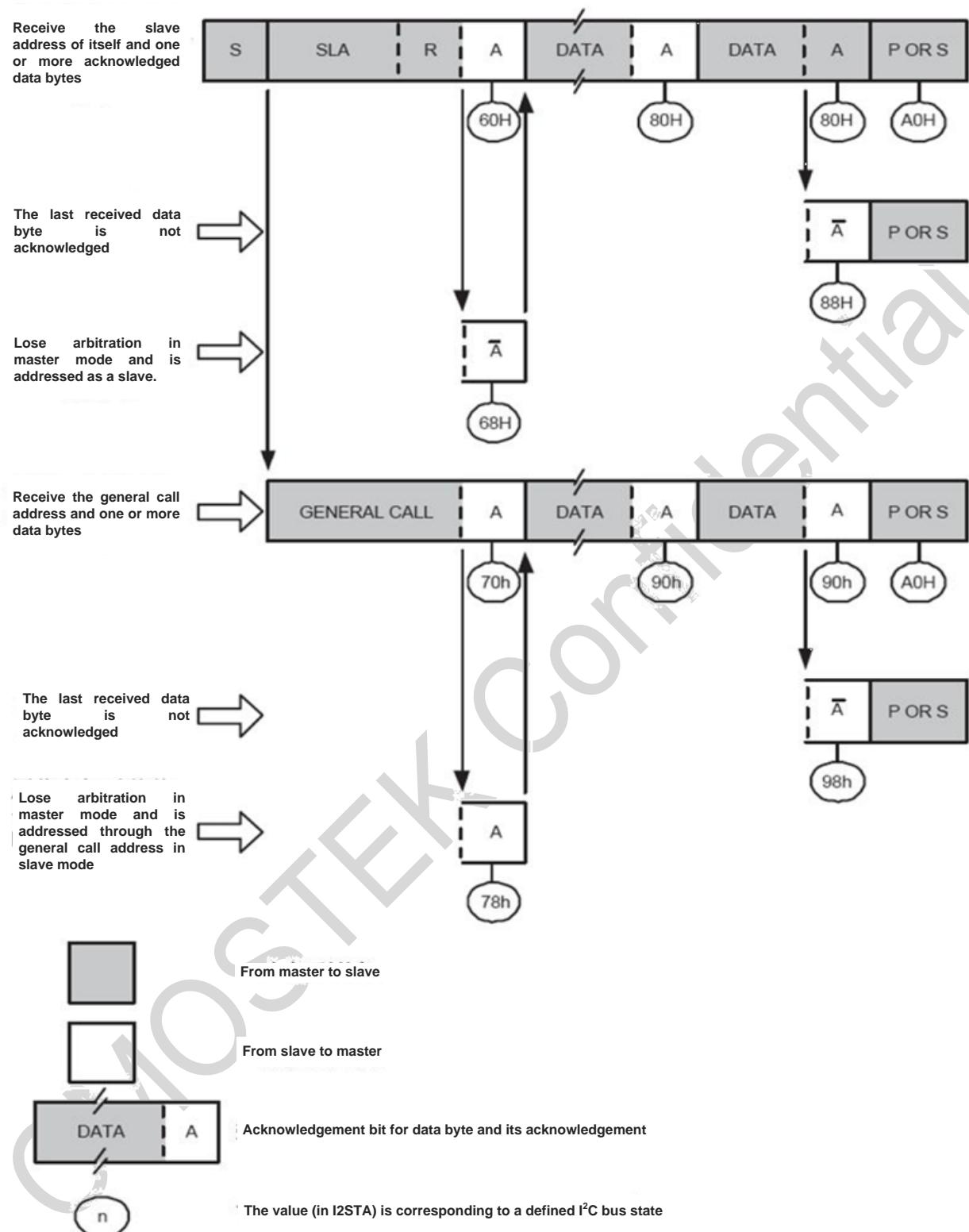


Figure 101. Master Receiving State Diagram

18.4.10 I²C Slave Transmitting Mode

In the slave transmitting mode, the data byte is sent to the master receiver. The data transfer is initialized as it is in the slave receiving mode. After initializing I²C_ADDR and I²C_CR, the I²C module waits until it is addressed by its own slave address

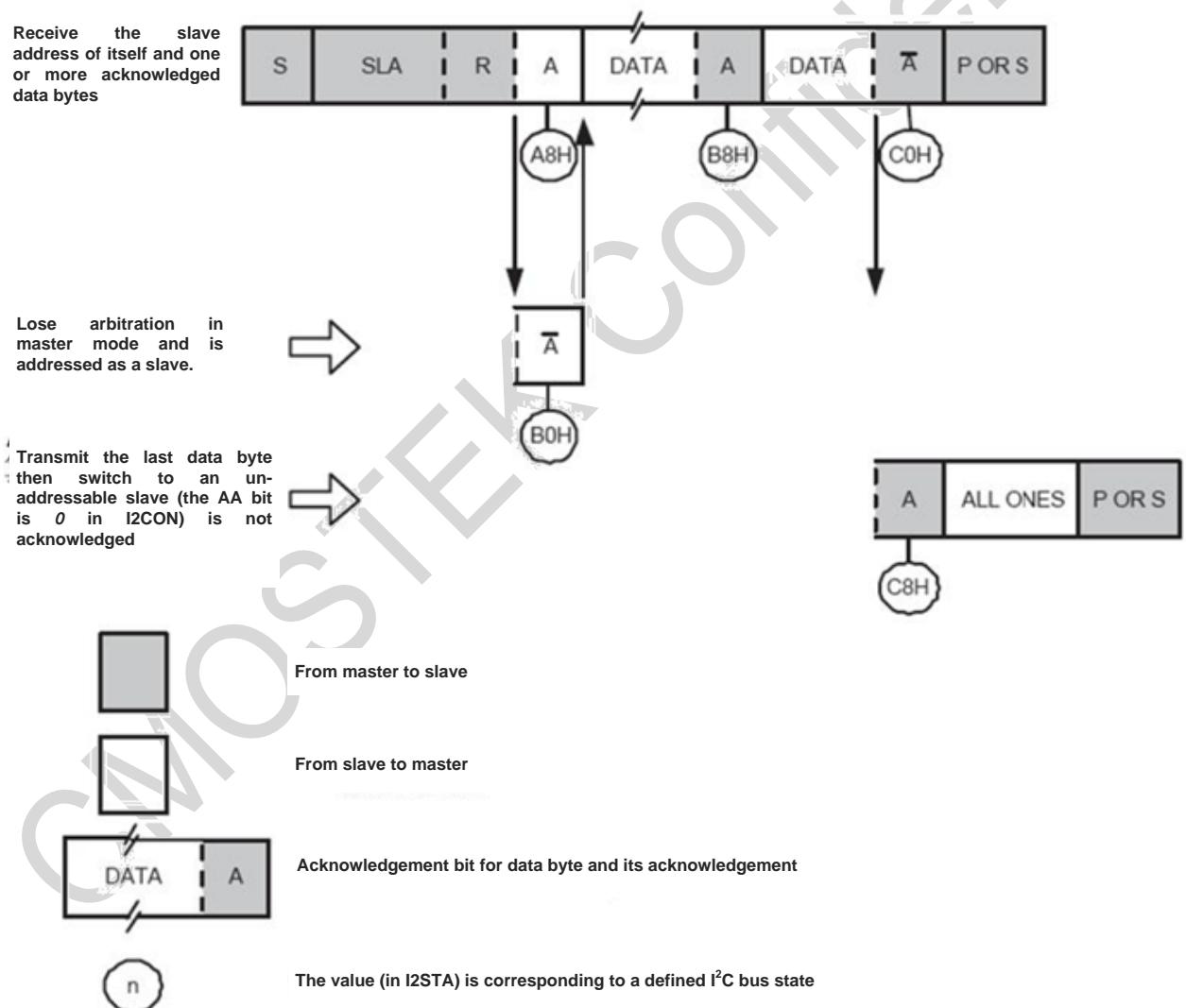
followed by the data direction bit. The data direction bit must be 1 (R) to ensure the I2C module operates in slave transmitting mode. After receiving its own slave address and R bit, the serial interrupt flag (si) is set and a valid state code can be read from I2C_STAT. This state code is used as a vector for the state service program. The corresponding operation for each state code is shown in the following table. If the arbitration of the I2C module is lost in master mode, it enters the slave transmitting mode (see state 0xB0).

If the aa bit is reset during transmission, the I2C module will transmit the last byte and enter state 0xC0 or 0xC8. The I2C module switches to the non-addressable slave mode and ignores the master receiver if it continues transmitting. Therefore the master receiver will receive serial data with all the bits being 1. When aa is reset, the I2C module does not respond to its own slave address or general call address. However, the I2C bus is still being monitored and address recognition can be recovered at any time by setting aa, meaning that the aa bit can separate the I2C module from the I2C bus temporarily.

Table 631. Master Transmitting Mode State Table

State code	I2C Bus and Hardware State	Application Software Respond					Next Action of I2C Hardware	
		Read/Write I2C_DATA	Write I2C_CR					
			sta	sto	si	aa		
A8H	The SLA + R of its own is received and ACK is returned	Load data bytes	X	0	0	0	Send the last data byte and receive ACK.	
		Load data bytes	X	0	0	1	Send the last data byte and receive ACK.	
B0H	The arbitration is lost in the SLA + R/W of master mode. The SLA + R of its own is received. ACK is returned.	Load data bytes	X	0	0	0	Send the last data byte and receive ACK.	
		Load data bytes	X	0	0	1	Send the last data byte and receive ACK.	
B8H	Data bytes is sent and ACK is received.	Load data bytes	X	0	0	0	Send the last data byte and receive ACK.	
		Load data bytes	X	0	0	1	Send the last data byte and receive ACK.	
C0H	Data bytes is sent and NON_ACK is received.	No I2C_DATA related action	0	0	0	0	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.	
		No I2C_DATA related action	0	0	0	1	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.	
		No I2C_DATA related action	1	0	0	0	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the START condition when the bus is idle.	
		No I2C_DATA related action	1	0	0	1	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the START condition when the bus is idle.	
C8H	The loaded data bytes is sent and	No I2C_DATA related action	0	0	0	0	Switch to the un-addressable slave mode. Does not recognize its own slave address	

	ACK is received.					or general address.
	No I2C_DATA related action	0	0	0	1	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address.
	No I2C_DATA related action	1	0	0	0	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the START condition when the bus is idle.
	No I2C_DATA related action	1	0	0	1	Switch to the un-addressable slave mode. Does not recognize its own slave address or general address. Send the START condition when the bus is idle.

Figure 102. I²C Slave Transmitting State Diagram

18.4.11 I2C Other Miscellaneous State

I2C_STAT= 0xF8, this state code indicates that there is no relevant information available because the serial interrupt flag *si* is not set yet. This situation occurs between the state where I2C module has not yet started serial transmission and other possible states.

I2C_STAT= 0x00, this state code indicates that a bus error occurs during the I2C serial transfer. A bus error occurs when a START or STOP condition occurs on an illegal location of a format frame. These illegal locations refer to the address bytes, data bytes, or acknowledge bits during a serial transfer. A bus error can also occur when external interference affects the internal I2C module signal. The *si* is set when a bus error occurs. To recover from a bus error, the *sto* flag must be set and *si* must be cleared. This puts the I2C module into the un-addressable slave mode (defined state) and clears the *sto* flag (other bits in I2C_CR are unaffected). The SDA and SCL lines are released (no STOP condition is sent).

Table 632. Other Miscellaneous States Table

State code	I2C Bus and Hardware State	Application Software Respond				Next Action of I2C Hardware	
		Read/Write I2C_DATA	Write I2C_CR				
			sta	sto	si	aa	
F8H	no relevant information available. <i>si</i> =0.	No I2C_DATA related action	No I2C_DATA related action			Wait or perform the current transmission	
00H	A bus error occurs on the master or selected slave due to an illegal START or STOP condition. 0x00 state also occurs when external interference causes I2C to enter undefined state	No I2C_DATA related action	0	1	0	X	The internal hardware is affected only in the master or addressable slave mode. In general, the bus is released and the I2C module switches to the un-addressable slave mode and <i>sto</i> is reset.

18.5 I2C Operation Mode

18.5.1 Initialization Procedure

Example for initializing the I2C interface as a slave/master.

1. Load its own slave address into I2C_ADDR and enable general call identification (if needed);
2. Enable the I2C interrupt.
3. Write 0x44 to register I2C_CR to set the ens and aa bits and enable the slave function.

For the master function, write 0x40 to register I2C_CR.

18.5.2 Port Configuration Procedure

Example for mapping I2C interface signal SCL, SDA to P26 and P25 pins.

1. Configure P26 and P25 as open-drain output mode. Configure P2OD[6] and P2OD[5] as 1.
2. Configure P26 and P25 output value register. Configure P2OUT[6], P2OUT[5] as 1.
3. Configure P26 and P25 input and output configuration registers. Configure P2DIR[6] and P2DIR[5] as 0.
4. Configure P26 and P25 function configuration register. Configure P26_sel and P25_sel as 6.
5. Configure P26 and P25 pull-up enabling configuration register. Configure P2PU[6] and P2PU[5] as 1.

18.5.3 Start Master Transmission Function

The master transmission operation can be performed by setting up buffers, pointers, and data counts and then sending the START condition.

1. Initialize the master data counter.
2. Setup the slave address to which the data will be sent, and add a *write* bit.
3. Write 0x20 to I2C_CR to set the *sta* bit.
4. Setup the data to be sent in the master transmission buffer.
5. Initialize the master data counter to match the length of the message being sent.
6. Exit.

18.5.4 Start Master Receiving Function

The master receiving operation can be performed by setting up buffers, pointers, and data counts and then sending the START condition.

1. Initialize the master data counter.
2. Setup the slave address to which the data will be sent, and add a *read* bit.
3. Write 0x20 to I2C_CR to set the *sta* bit.
4. Setup data to be transmitted in the master receiveing buffer.
5. Initialize the master data counter to match the length of the message being sent;
6. Exit.

18.5.5 I2C Interrupt Procedure

Determine the state of I2C and the corresponding state program.

1. Read the state of I2C from I2C_STAT.
2. Use the state value to jump to one of the 26 possible state programs.

18.5.6 State with No Specified Mode

1. State: 0x00 bus error

Enter the un-addressable slave mode and release the bus.

- a) Write 0x14 to I2C_CR to set the sto and aa bits.
- b) Write 0xF7 to I2C_CR to clear the si flag.
- c) Exit.

2. Master state

State 08 and 10 are for the master transmitting mode and the master receiving mode. The R/W bit determines whether the next state is the master transmitting mode or master receiving mode.

3. State: 0x08

The START condition has been sent. The slave address plus R/W bit will be transmitted and the ACK bit will be received.

- a) Write the slave address and R/W bit to I2C_DATA.
- b) Write 0x04 to I2C_CR to set the aa bit.
- c) Write 0xF7 to I2C_CR to clear the si flag.
- d) Setup the master transmitting mode data buffer.
- e) Setup the master receiving mode data buffer.
- f) Initialize the master data counter.
- g) Exit.

4. State: 0x10

A repeated START condition has been sent. The slave address plus R/W bit will be transmitted and the ACK bit will be received.

- a) Write the slave address and R/W bit to I2C_DATA.
- b) Write 0x04 to I2C_CR to set the aa bit.
- c) Write 0xF7 to I2C_CR to clear the si flag.
- d) Setup the master transmitting mode data buffer.
- e) Setup the master receiving mode data buffer.
- f) Initialize the master data counter.
- g) Exit.

18.5.7 Master Transmitter State

1. State: 0x18

The previous state of 8 or 10 indicates that the slave address and the write operation bit have been transmitted and a response has been received. The first data byte will be sent and the ACK bit will be received.

- a) Load the first data byte of the master transmitting buffer into I2C_DATA.

- b) Write 0x04 to I2C_CR to set the *aa* bit.
- c) Write 0xF7 to I2C_CR to clear the *si* flag.
- d) Add one to the transmitting buffer pointer.
- e) Exit.

2. State: 0x28

The data has been sent and an ACK has been received. A STOP condition is sent if the transmitted data is the last data byte, otherwise the next data byte is sent.

- a) Subtract one to the data counter. If it is not the last data byte, skip to step e).
- b) Write 0x14 to I2C_CR to set the *sta* and *aa* bits.
- c) Write 0xF7 to I2C_CR to clear the *si* flag.
- d) Exit.
- e) Load the next data byte of the master transmitting buffer into I2C_DATA.
- f) Write 0x04 to I2C_CR to set the *aa* bit.
- g) Write 0xF7 to I2C_CR to clear the *si* flag.
- h) Add one to the master transmitting buffer pointer.
- i) Exit.

3. State: 0x30

Data has been sent and a NON_ACK has been received. A STOP condition will be sent soon.

- a) Write 0x14 to I2C_CR to set the *sta* and *aa* bits.
- b) Write 0xF7 to I2C_CR to clear the *si* flag.
- c) Exit.

4. State: 0x38

The arbitration has been lost in the process of sending the slave address and the *write* operation bit or data. The bus has been released and entered the un-addressable slave mode. A new START condition is sent when the bus is idle again.

- a) Write 0x24 to I2C_CR to set the *sta* and *aa* bits.
- b) Write 0xF7 to I2C_CR to clear the *si* flag.
- c) Exit.

18.5.8 Master Receiving States

1. State: 0x40

The previous state is 08 or 10 indicating that the slave address and the *read* operation bit have been transmitted and an ACK has been received. It will receive the data and return an ACK.

- a) Write 0x04 to I2C_CR to set the *aa* bit.

b) Write 0xF7 to I2C_CR to clear the *si* flag.

c) Exit.

2. State: 0x48

The slave address and *read* operation bit have been transmitted and a NON_ACK has been received. A STOP condition will be sent.

a) Write 0x14 to I2C_CR to set the *sfo* and *aa* bit.

b) Write 0xF7 to I2C_CR to clear the *si* flag.

c) Exit.

3. State: 0x50

Data has been received and an ACK has been returned. The data will be read from I2C_DATA. It will receive other data. If this is the last data byte, a NON_ACK is returned, otherwise an ACK is returned.

a) Read the data byte in I2C_DATA and store it in the master receiving buffer.

b) Subtract one to the master data counter. If it is not the last data byte, jump to step e).

c) Write 0xF3 to I2C_CR to clear the *si* flag and the *aa* bit.

d) Exit.

e) Write 0x04 to I2C_CR to set the *aa* bit.

f) Write 0xF7 to I2C_CR to clear the *si* flag.

g) Add one to the master receiving buffer .

h) Exit.

4. State: 0x58

Data has been received and a NON_ACK has been returned. The data will be read from I2C_DATA and a STOP condition will be sent.

a) Read the data byte in I2C_DATA and store it in the master receiving buffer;

b) I2C_CR writes 0x14 to set the *sfo* and *aa* bits.

c) Write 0xF7 to I2C_CR to clear the *si* flag.

d) Exit.

18.5.9 Slave Receiver State

1. State: 0x60

It has received its own slave address and *write* operation bit and has returned an ACK. It will receive data bytes and return an ACK.

a) Write 0x04 to I2C_CR to set the *aa* bit.

b) Write 0xF7 to I2C_CR to clear the *si* flag.

c) Setup the data buffer for the slave receiving mode.

d) Initialize the slave data counter.

e) Exit.

2. State: 0x68

Arbitration has been lost while transmitting the slave address and R/W bit when used as a bus master. It has received its own slave address and *write* operation bit and have returned an ACK. It will receive the data and return an ACK. When the bus is idle again, set *sta* to restart the master mode.

- a) Write 0x24 to I2C_CR to set the *sta* and *aa* bits.
- b) Write 0xF7 to I2C_CR to clear the *si* flag.
- c) Setup the data buffer for the slave receiving mode.
- d) Initialize the slave data counter;
- e) Exit.

3. State: 0x70

A general call has been received and an ACK has been returned. It will receive a data byte and return an ACK.

- a) Write 0x04 to I2C_CR to set the *aa* bit.
- b) Write 0xF7 to I2C_CR to clear the *si* flag.
- c) Setup the data buffer for the slave receiving mode.
- d) Initialize the slave data counter.
- e) Exit.

4. State: 0x78

The arbitration has been lost while transmitting the slave address and R/W bit when used as a bus master. A general call has been received and an ACK has been returned. It will receive the data and return an ACK. When the bus is idle again, set *sta* to restart the master mode.

- a) Write 0x24 to I2C_CR to set the *sta* and *aa* bits.
- b) Write 0xF7 to I2C_CR to clear the *si* flag.
- c) Setup the data buffer for the slave receiving mode.
- d) Initialize the slave data counter.
- e) Exit.

5. State: 0x80

Its own slave address has been addressed before. Data has been received and an ACK is returned. Further data will be read.

- a) Read the data byte of I2C_DATA and store it in the slave receiving buffer.
- b) Add one to the slave data counter. If it is not the last data byte, jump to step e).
- c) Write 0xF3 to I2C_CR to clear the *si* flag and the *aa* bit.
- d) Exit.
- e) Write 0x04 to I2C_CR to set the *aa* bit.
- f) Write 0xF7 to I2C_CR to clear the *si* flag.
- g) Add one to the slave receiving buffer pointer.
- h) Exit.

6. State: 0x88

Its own slave address has been addressed before. Data has been received and a NON_ACK is returned. It will not store the received data and enter the un-addressable slave mode.

- a) Write 0x04 to I2C_CR to set the aa bit.
- b) Write 0xF7 to I2C_CR to clear the si flag.
- c) Exit.

7. State: 0x90

The general call addressing has been performed previously. Data has been received and an ACK is returned. It will store the data to be received. Only the first data byte will be received with an ACK is returned. NON_ACK will be returned after receiving other data bytes.

- a) Read the data byte of I2C_DATA and put it into the slave receiving buffer;
- b) Write 0xF3 to I2C_CR to clear the si flag and the aa bit.
- c) Exit.

8. State: 0x98

The general call addressing has been performed previously. Data has been received and NON_ACK is returned. It will not store the received data and will enter the un-addressable slave mode.

- a) Write 0x04 to I2C_CR to set the aa bit.
- b) Write 0xF7 to I2C_CR to clear the si flag.
- c) Exit.

9. State: 0xA0

A STOP condition or a repeated START condition have been received, but is still addressed as a slave. It will not store the received data and enter the un-addressable slave mode.

- a) Write 0x04 to I2C_CR to set the aa bit.
- b) Write 0xF7 to I2C_CR to clear the si flag.
- c) Exit.

18.5.10 Slave Transmitter State

1. State: 0xA8

It has received its own slave address and *read* operation bit and returned an ACK. The data will be sent and the ACK bit will be received.

- a) Load the first data byte of the slave transmitting buffer into I2C_DATA.
- b) Write 0x04 to I2C_CR to set the aa bit.
- c) Write 0xF7 to I2C_CR to clear the si flag.
- d) Setup the data buffer for the slave transmitting mode.
- e) Add one to the slave transmitting buffer pointer.
- f) Exit.

2. State: 0xB0

The arbitration has been lost while transmitting the slave address and R/W bit when used as a bus master. It has received its own slave address and *read* operation bit and returned an ACK. The data will be sent and the ACK bit will be received. When the bus is idle again, set *sta* to restart the master mode.

- a) Load the first data byte of the slave transmitting buffer into I2C_DATA.
- b) Write 0x24 to I2C_CR to set the *sta* and *aa* bits.
- c) Write 0xF7 to I2C_CR to clear the *si* flag.
- d) Setup the data buffer for the slave transmitting mode.
- e) Add one to the slave transmitting buffer pointer.
- f) Exit.

3. State: 0xB8

The data has been sent and an ACK has been received. The further data will be sent and the ACK bit will be received.

- a) load the data byte of the slave transmitting buffer into I2C_DATA.
- b) Write 0x04 to I2C_CR to set the *aa* bit.
- c) Write 0xF7 to I2C_CR to clear the *si* flag.
- d) Add one to the slave transmitting buffer pointer.
- e) Exit.

4. State: 0xC0

Data has been sent and a NON_ACK has been received. Enter the un-addressable slave mode.

- a) Write 0x04 to I2C_CR to set the *aa* bit.
- b) Write 0xF7 to I2C_CR to clear the *si* flag.
- c) Exit.

5. State: 0xC8

The last data byte has been sent and an ACK has been received. Enter the un-addressable slave mode.

- a) Write 0x04 to I2C_CR to set the *aa* bit.
- b) Write 0xF7 to I2C_CR to clear the *si* flag.
- c) Exit.

18.6 I2C Register Description

Base address: 0x40000400

Table 633. Register List

Offset	Register Name	Access	Description
0x00	I2C_TMRUN	RW	I2C baud rate counter enabling register
0x04	I2C_TM	RW	I2C baud rate counter configuration register
0x08	I2C_CR	RW	I2C configuration register

0x0c	I2C_DATA	RW	I2C data register
0x10	I2C_ADDR	RW	I2C address register
0x14	I2C_STAT	RO	I2C state register

18.6.1 I2C Baud Rate Counter Enabling Register (I2C_TMRUN)

Offset address: 0x00

Reset value: 0x0000 0000

Table 634. I2C Baud Rate Counter Enabling Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 635. I2C Baud Rate Counter Enabling Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															
-															

Table 636. I2C Baud Rate Counter Enabling Register (3)

Bit	Flag	Description
31:1	Reserved	
0	tme	Baud rate counter enabling. 0: disable, 1: enable.

18.6.2 I2C Baud Rate Counter Configuration Register (I2C_TM)

Offset address: 0x04

Reset value: 0x0000 0000

Table 637. I2C Baud Rate Counter Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 638. I2C Baud Rate Counter Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								tm							
-								R/W							

Table 639. I2C Baud Rate Counter Configuration Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	tm	Tm: baud rate counter configuration. $F_o = F_i / 8 * (N+1)$, N = tm N > 0

18.6.3 I2C Configuration Register (I2C_CR)

Offset address: 0x08

Reset value: 0x0000 0000

Table 640. I2C Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 641. I2C Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								ens	sta	sto	si	aa	Res.	h1m	
-								R/W	R/W	R/W	R/W	R/W	R/W	-	

Table 642. I2C Configuration Register (3)

Bit	Flag	Description
31:7	Reserved	
6	ens	I2C module enabling. 0: disable, 1: enable.
5	sta	START flag enabling. 0: disable, 1: enable.
4	sto	STOP flag enabling. 0: disable, 1: enable.
3	si	I2C interrupt flag
2	aa	Acknowledgment flag enabling. 0: disable, 1: enable.
1	Reserved	
0	h1m	I2C high speed 1 Mbps mode enabling. 0: disable, 1: enable.

18.6.4 I2C Data Register (I2C_DATA)

Offset address: 0x0c

Reset value: 0x00000000

Table 643. I2C Data Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 644. I2C Data Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								i2cdat							
-								R/W							

Table 645. I2C Data Register (3)

Bit	Flag	Description
31:8	Reserved	

7:0	i2cdata	I2C data register. In I2C transmitting mode, write the transmitting data to this register. In I2C receiving mode, read the received data from this register.
-----	---------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------

18.6.5 I2C Address Register (I2C_ADDR)

Offset address: 0x10

Reset value: 0x00000000

Table 646. I2C Address Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 647. I2C Address Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved								i2cadr								GC
-								R/W								R/W

Table 648. I2C Address Register (3)

Bit	Flag	Description
31:8	Reserved	
7:1	i2cadr	I2C slave mode address
0	GC	Broadcast address acknowledgment enabling. 0: disable, 1: enable.

18.6.6 I2C State Register (I2C_STAT)

Offset address: 0x14

Reset value: 0x00000000

Table 649. I2C State Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 650. I2C State Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								i2csta							
-								R/W							

Table 651. I2C State Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	i2csta	I2C state register

19 SPI

19.1 Introduction to SPI

The SPI (Serial Peripheral Interface) bus is a synchronous serial peripheral interface that allows the MCU to exchange information with various peripheral devices in a serial direction. The SPI interface uses 4 lines, the serial clock line (SCLK), the master output / slave input line (MOSI), the master input / slave output line (MISO), and the active low slave-selection line (SSN).

19.2 SPI Major Features

The SPI controller supports the following features:

- Can be configured as a master or slave by programming.
- Full duplex communication capability.
- 7 configurable baud rates.
- 4-wire transmission mode.
- The maximum baud rate of the master mode as 1/2 system clock.
- The maximum baud rate of the slave mode as 1/4 system clock.
- Configurable serial clock polarity and phase.
- Support for interrupt mode.
- 8-bit data transmission with higher bits transmitted the first.

19.3 SPI Function Description

19.3.1 SPI Master Mode

All data transfer on the bus is initiated by SPI master device by setting the master / slave SPI0_CR.4 control bit 1 to setup SPI as master mode. When the SPI is in master mode, enable SPI (set SPI0_CR.6 to 1) and meanwhile write a byte to the SPI data register SPI0_DATA, thus the data transmission begins. SPI master immediately moves out data serially on the MOSI line while providing the serial clock SCK. The spif interrupt flag is set to 1 after the transmission. If the interrupt is allowed, an interrupt request will be generated. In full duplex applications, while master SPI devices transmitting data to slaves, the addressed slave SPI devices can transmit data to the masters on MISO line simultaneously. Therefore, the spif flag is both a transmission completion flag and a ready-for-receiving-data flag. The processor gets the received data by reading the SPI0_DATA register.

19.3.1.1 Operation Flow

1. Port configuration. Configure the port controller to map SCK, MISO and MOSI signals to the correct pins.
2. The value of the chip selection signal - SPI_CS is determined by the value of the register SPI0_SSNO in master mode.
3. SPI baud rate configuration. Set SPI0_CR.7, SPI0_CR.1 and SPI0_CR.0.
4. Serial clock configuration. Set SPI clock polarity cpol (SPI0_CR.3) and clock phase cpha (SPI0_CR.2). See the SPI0_CR register for details.

5. Master mode configuration with mstr(SPI0_CR.4) = 1.
6. Enable SPI with spen(SPI0_CR.6) = 1
7. Slave selection. Configure as SPI0_SS.N.0 = 0.
8. Start transmitting data. The data to be sent to the slave is written to the SPI data register SPI0_DATA.
9. Wait for the transmitting/receiving data completion. Ready to transmit/receive the next data

The below item is for interrupt service routine.

10. 1. Read the SPI0_DATA register, meaning receiving the data sent from the slave device.

Notes:

1. The spif interrupt is cleared to 0 after step 10 completes.
2. Repeat step 8 and 9 if for continuing data sending/receiving.
3. In multiprocessor communication, the SSN pin can be replaced with GPIO.

19.3.1.2 Timing

Using the clock polarity bit *cpol* and phase bit *cpha* in the SPI control register SPI0_CR, it selects one from the 4 combinations as the serial clock. *Cpol* (SPI0_CR.3) represents if the SCK is high or low when SPI is idle. *Cpha* (SPI0_CR.2) represents the 2 clock phase options (edges used by data sampling). The master and slave must be configured to use the same clock phase and polarity. The baud rate setting is valid for the master only and ignored for the slave. The master mode data/clock timing is as follows.

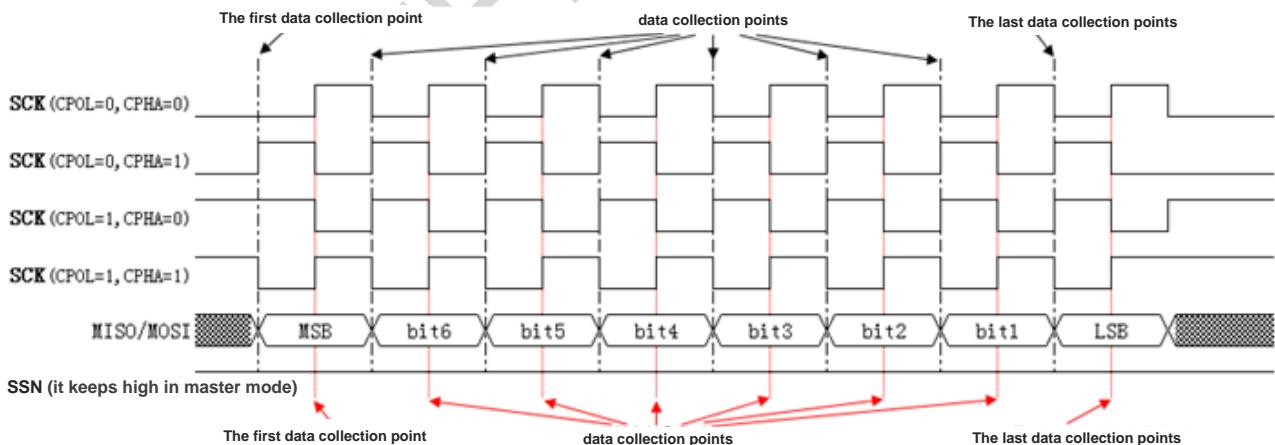


Figure 103. Master Mode Data/Clock Timing

19.3.2 SPI Slave Mode

SPI acts as an SPI slave when it is not configured as master when being enabled. SPI is placed in slave mode by setting the master / slave SPI0_CR.4 control bit to 0. When the SPI is in slave mode, the master controls the serial clock (SCK) and moves in data from the MOSI. The counter in the SPI logic counts the SCK edges. When the 8-bit data shift completes, the *spif* flag is set to 1. The received data is obtained by reading SPI0_DATA. The slave device cannot start the data transmission function. The data to be sent to the master device is pre-installed by writing SPI0_DATA. Then controlled by master SCK, the data is moved to

the MISO line bit by bit and sent to the master device.

19.3.2.1 Operation Flow

1. Port configuration. Configure the port controller to map SCK, MISO and MOSI signals to the correct pins.
2. Select 1 GPIO as the chip selection signal in the slave mode. Refer to the chapter of *Port Control Register* for details.
3. SPI baud rate configuration. Set SPI0_CR.7, SPI0_CR.1 and SPI0_CR.0.
4. Serial clock configuration. Set SPI clock polarity *cpol* (SPI0_CR.3) and clock phase *cpha* (SPI0_CR.2). See the SPI0_CR register for details.
5. Slave mode configuration with *mstr*(SPI0_CR.4) = 0.
6. Enable SPI with setting *spen*(SPI0_CR.6) as 1
7. The data to be sent to the master is written to the SPI data register SPI0_DATA.
8. Wait for the transmitting/receiving data completion. Ready to transmit/receive the next data

The below item is for interrupt service routine.

10. Read the SPI0_DATA register, meaning receiving the data sent from the slave device.

Notes:

1. The spif interrupt is cleared to 0 after step 9 completes.
2. When the slave clock phase *cpha* (SPI0_CR.2) is configured to 0, the master can only transfer one byte of data to the slave each time the SPI_CS signal is pulled low. If each time the master pulls the SPI_CS signal low and continuously transfers data to the slave, the slave must increase one more operation of *writing* 0 to the SPI0_DATA in the interrupt service routine.
3. Repeat steps 7 and 8 to continue transmitting/receiving data.

19.3.2.2 Timing

The slave data/clock timing is as follows.

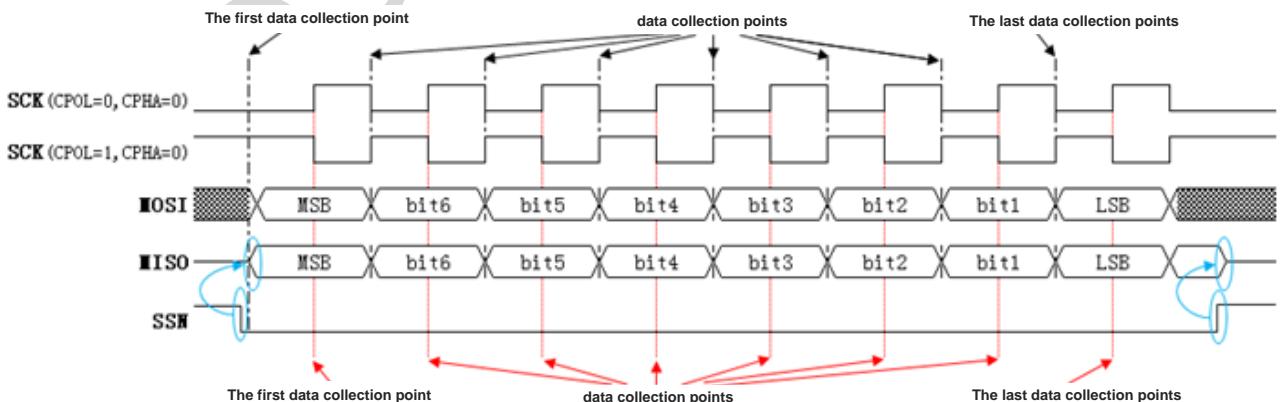


Figure 104. Slave Mode Data/Clock Timing Diagram (cpha = 0)

19.4 SPI Interrupt

If the SPI interrupt is enabled, an interrupt will be generated when the SPI transmission completion interrupt flag *spif* (SPI0_STAT.7) or the master mode error interrupt flag *mdf* (SPI0_STAT.4) in the SPI is set to 1.

- At the end of each byte transmission, the SPI transmission completion Interrupt flag bit, *spif* (SPI0_STAT.7), is automatically set to 1 by the hardware.
- When the SPI is configured in the master mode, the external SSN input is low. At this time, the SSN input level conflicts with the SPI operating mode. The SPI master mode error interrupt flag *mdf* (SPI0_STAT.4) is automatically set by the hardware in the case.

19.5 SPI Multi-Master/Multi-Slave Mode

This product can be configured as a master in the SPI single-master/single-slave system. The SPI0 chip selection configuration register can be configured to output high and low signals to the SPI_CS pin. As a slave, it supports to configure the GPIO_CTRL1 register discussed in Chapter 7 to select a GPIO pin as the source of SPI_SS. However, in the multi-master/multi-slave system, as shown in the following figure, it must be configured according to the corresponding process.

When the system is a single-master/multi-slave one, the SPI_CS pin can be used as the chip selection signal of slave 1, and the chip selection signals of other slaves are connected by the GPIO pin. When the system is multi-master and multi-slave, all slave chip selection signals are connected through the GPIO pin. The master must also connect to the SPI_CS signal of other masters through the GPIO pin to monitor whether the bus is occupied.

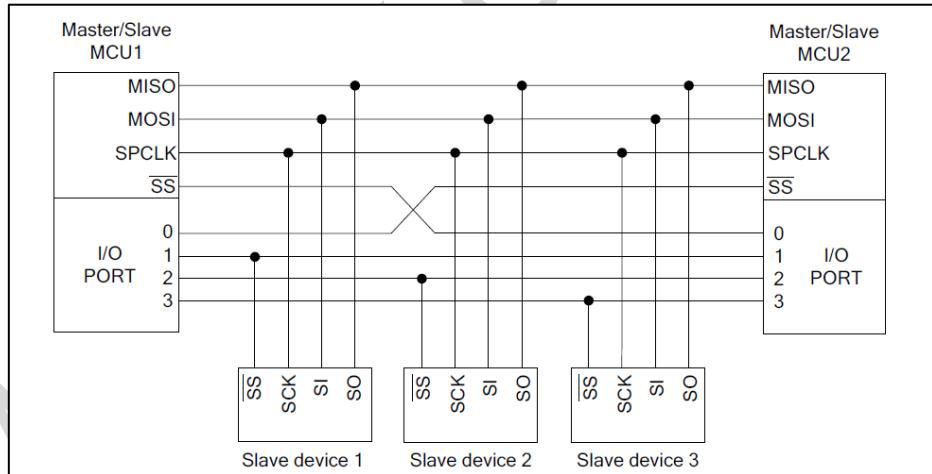


Figure 105. SPI Multi-Master/Multi-Slave Mode

19.6 SPI Register Description

Base address: 0x40000800

Table 652. Register List

Offset	Register Name	Access	Description
0x00	SPI0_CR	RW	SPI0 configuration register
0x04	SPI0_SS_N	RW	SPI0 chip selection configuration register
0x08	SPI0_STAT	RO	SPI0 state register
0x0c	SPI0_DATA	RW	SPI0 data register

19.6.1 SPI0 Configuration Register (SPI0_CR)

Offset address: 0x00

Reset value: 0x0000 0014

Table 653. SPI0 Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 654. SPI0 Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								spr2	spen	Res.	mstr	cpol	cpha	spr1	spr0
-								R/W	R/W	-	R/W	R/W	R/W	R/W	R/W

Table 655. SPI0 Configuration Register (3)

Bit	Flag	Description
31:8	Reserved	
7	spr2	Baud rate selection bit 2. Refer to spr 0.
6	spen	Module enabling register. 0: enable, 1: disable.
5	Reserved	
4	mstr	Master/slave mode selection. 0: slave, 1: master
3	cpol	Clock polarity selection register. 0: low, 1: high.
2	cpha	Clock phase selection register. 0: first edge, 1: second edge.
1	spr1	Baud rate selection bit 1. Refer to spr 0.
		Baud rate selection bit 0

Table 19-2. Master Mode Baud rate Selection

0	spr0	SPR2	SPR1	SPR0	SPI_CLK Rate
		0	0	0	fsys/2
		0	0	1	fsys /4

			0	1	0	fsys /8	
			0	1	1	fsys /16	
			1	0	0	fsys /32	
			1	0	1	fsys /64	
			1	1	0	fsys /128	
			1	1	1	0	

19.6.2 SPI0 Chip Selection Configuration Register(SPI0_SSN)

Offset address: 0x04

Reset value: 0x00000001

Table 656. SPI0 Chip Selection Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Table 657. SPI0 Chip Selection Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Table 658. SPI0 Chip Selection Configuration Register (3)

Bit	Flag	Description
31:1	Reserved	
0	ssn	Ssn output value. In master mode, the software configures the ssn value to control the SPI_CS port level.

19.6.3 SPI0 State Register (SPI0_STAT)

Offset address: 0x08

Reset value: 0x00000000

Table 659. SPI0 State Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Table 660. SPI0 State Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Table 661. SPI0 State Register (3)

Bit	Flag	Description
31:8	Reserved	
7	spif	Transmission completion interrupt flag
6	wcol	Writing conflict flag
5	sserr	Slave mode SSN error flag
4	mdf	Master mode error flag
3:0	Reserved	

19.6.4 SPI0 Data Register (SPI0_DATA)

Offset address: 0x0c

Reset value: 0x00000000

Table 662. SPI0 Data Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 663. SPI0 Data Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								spdat							
-								R/W							

Table 664. SPI0 Data Register (3)

Bit	Flag	Description
31:8	Reserved	
7:0	Spdat	Data register. In transmitting mode, write the sending data to this register In receiving mode, read the received data from this register

20 CLK_TRIM

20.1 Introduction to CLK_TRIM

The CLK_TRIM (clock trimming) module is a circuit designed for clock calibration/monitor. It selects the accurate clock source in calibration mode to calibrate the inaccurate clock source and repeats the calibration by adjusting the parameters of the inaccurate clock source until the frequency of the calibrated clock source reaching the accuracy requirement. For this product, the error that may occurs in the count value in the calibration mode is within the allowable error range. In monitoring mode, a stable clock source is selected to monitor the system operating clock. In the preset monitoring period, it monitors the failure of the system operating clock and an interrupt occurs in the failure case. In calibration mode and monitoring mode, the required clock source must be initialized and enabled. For the specific configuration process, please refer to Chapter 4 *System Controller*.

20.2 CLK_TRIM Features

CLK_TRIM supports the following features:

- Calibration mode
- Monitoring mode
- 32-bit reference clock counter with support of initial value loading
- 32-bit clock counter with support of configurable calibration overflow value
- 6 reference clock sources
- 4 types of calibrated clock sources
- Support for interrupt mode

20.3 CLK_TRIM Function Description

20.3.1 CLK_TRIM Calibration Mode

The calibration mode is mainly used to select an accurate clock source as a reference clock to calibrate an inaccurate clock source.

The software repeats calibration according to the following operation flow and adjust the parameters of the clock source to be calibrated until it meets the frequency accuracy requirements.

20.3.1.1 Operation Flow

1. Set the CLKTRIM_CR .refclk_sel register to select the reference clock.
2. Set the CLKTRIM_CR .calclk_sel register to select the clock to be calibrated.
3. Set the CLKTRIM_REFCON .rcntval register as the calibration time.
4. Set the CLKTRIM_CR .IE register to enable interrupts.
5. Set the CLKTRIM_CR .trim_start register to start calibration.
6. The reference clock counter and the clock counter to be calibrated start count.

7. When the reference clock counter is counted down from the initial value to 0, CLKTRIM_IFR.stop is set to 1 to trigger an interrupt.
8. The interrupt service routine determines that CLKTRIM_IFR.stop is 1 and read the values of the registers CLKTRIM_REFcnt and CLKTRIM_CALCNT.
9. Clear the CLKTRIM_CR .trim_start register to end the calibration.

Notes:

The counter of the clock to be calibrated may overflow before CLKTRIM_IFR.stop is set due to the calibration time being set too long in the calibration process. *CLKTRIM_IFR.calcnt_of* is set to 1 and an interrupt is triggered in the case. When the interrupt service routine checks out *CLKTRIM_IFR.calcnt_of* is set as 1, it clears the CLKTRIM_CR.trim_start register and ends the calibration. The calibration cannot be performed correctly In this case. Therefore the calibration time must be adjusted to do recalibration.

The steps are as follows:

1. Set the CLKTRIM_REFCON .rcntval register to adjust the calibration time.
2. Set the CLKTRIM_CR .trim_start register to resume calibration.

20.3.2 CLK_TRIM Monitoring Mode

This monitoring mode is mainly used to select a stable clock source as a reference clock to monitor the abnormal state of the system operating clock periodically in preset time. In this monitoring mode, only the external XTH clock or the external XTL clock can be selected as monitored clock.

20.3.2.1 Operation Flow

1. Set the CLKTRIM_CR .refclk_sel register to select the reference clock.
2. Set the CLKTRIM_CR .calclk_sel register to select the monitored clock.
3. Set the CLKTRIM_REFCON .rcntval register as the monitoring time interval.
4. Set the CLKTRIM_CALCON. ccntval register as the monitored clock counter overflow time.
5. Set the CLKTRIM_CR .mon_en register to enable monitoring.
6. Set the CLKTRIM_CR .IE register to enable interrupts.
7. Set the CLKTRIM_CR .trim_start register to start monitoring.
8. The reference clock counter and the monitored clock counter start count.
9. When the reference clock counter count reaches the monitoring interval, it determines if the monitored clock counter overflows. If overflow occurs, it indicates that the monitored clock is operating properly. Otherwise it indicates the monitored clock fails. CLKTRIM_IFR .xtal32k_fault/xtal32m_fault is set to 1 to trigger an interrupt in the case.
10. When interrupt occurs, if SYS_CTRL1.CLOCK_FAULT_EN is set to 1 (refer to Chapter 4 *Register Description*), the system clock source is automatically switched to the internal RC clock, the interrupt service routine is processed, and the interrupt flag bit CLKTRIM_IFR .xtal32k_fault/xtal32m_fault is cleared. The CLKTRIM_CR .trim_start register is cleared to end monitoring after then.

20.4 CLK_TRIM Register Description

Base address: 0x40001800

Table 665. Register List

Offset	Register Name	Access	Register Description
0x00	CLKTRIM_CR	RW	Configuration register
0x04	CLKTRIM_REFCON	RW	Reference counter initial value configuration register
0x08	CLKTRIM_REFCNT	RO	Reference counter value register
0x0c	CLKTRIM_CALCNT	RO	Calibration counter value register
0x10	CLKTRIM_IFR	RO	Interrupt flag register
0x14	CLKTRIM_ICLR	RW	Interrupt flag clearing register
0x18	CLKTRIM_CALCON	RW	Calibration counter overflow value configuration register

20.4.1 Configuration Register(CLKTRIM_CR)

Offset address: 0x00

Reset value: 0x0000 0000

Table 666. Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 667. Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				IE	mon_en	calclk_sel	refclk_sel	trim_start							
-				R/W	R/W	R/W	R/W	R/W							

Table 668. Configuration Register (3)

Bit	Flag	Description
31:8	Reserved	
7	IE	Interrupt enabling register. 0: disable, 1: enable.
6	mon_en	Monitoring mode enabling register. 0: disable, 1: enable.
5:4	calclk_sel	Register for selecting clock to be calibrated and monitored. 00: RCH 01: XTH 10: RCL 11: XTL
3:1	refclk_sel	Reference clock selection register. 000: RCH

		001: XTH 010: RCL 011: XTL 100: IRC10K 101: EXT_CLK_IN
0	trim_start	Calibration/monitor start register. 0: disable, 1: enable.

20.4.2 Reference Counter Initial Value Configuration Register (CLKTRIM_REFCON)

Offset address: 0x04

Reset value: 0x00000000

Table 669. Reference Counter Initial Value Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Rcntval[31:16]															
R/W															

Table 670. Reference Counter Initial Value Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Rcntval[15:0]															
R/W															

Table 671. Reference Counter Initial Value Configuration Register (3)

Bit	Flag	Description
31:0	rcntval	Reference counter initial value

20.4.3 Reference Counter Value Configuration Register (CLKTRIM_REFCNT)

Offset address: 0x08

Reset value: 0x00000000

Table 672. Reference Counter Value Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Refcnt[31:16]															
RO															

Table 673. Reference Counter Value Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Refcnt[15:0]															
RO															

Table 674. Reference Counter Value Configuration Register (3)

Bit	Flag	Description

31:0	refcnt	Reference counter value
------	--------	-------------------------

20.4.4 Calibration Counter Value Register (CLKTRIM_CALCNT)

Offset address: 0x0c

Reset value: 0x00000000

Table 675. Calibration Counter Value Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
calcnt[31:16]															
RO															

Table 676. Calibration Counter Value Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
calcnt[15:0]															
RO															

Table 677. Calibration Counter Value Register (3)

Bit	Flag	Description
31:0	calcnt	Calibration counter value

20.4.5 Interrupt Flag Register (CLKTRIM_IFR)

Offset address: 0x10

Reset value: 0x0000 0000

Table 678. Interrupt Flag Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 679. Interrupt Flag Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										xth_fault	xtl_fault	calcnt_of	stop		
-										RO	RO	RO	RO		

Table 680. Interrupt Flag Register (3)

Bit	Flag	Description
31:4	Reserved	
3	xth_fault	XTH failure flag. CLKTRIM_ICLR.xth_fault_clr, write 0 to clear the flag.
2	xtl_fault	XTL failure flag. CLKTRIM_ICLR.xtl_fault_clr, write 0 to clear the flag.
1	calcnt_of	Calibration counter overflow flag.

		CLKTRIM_CR.star, write 0 to clear the flag.
0	stop	Reference counter stopping flag. CLKTRIM_CR.start, write 0 to clear the flag.

20.4.6 Interrupt Flag Clearing Register (CLKTRIM_ICLR)

Offset address: 0x14

Reset value: 0xf

Table 681. Interrupt Flag Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 682. Interrupt Flag Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										xth_fault_clr	xtl_fault_clr	Reserved			
-										R/W	R/W	-			

Table 683. Interrupt Flag Clearing Register (3)

Bit	Flag	Description
31:4	Reserved	
3	xth_fault_clr	Clearing XTH failure flag, write 0 to clear the flag.
2	xtl_fault_clr	Clearing XTL failure flag, write 0 to clear the flag.
1:0	Reserved	

20.4.7 Calibration Counter Overflow Value Configuration Register (CLKTRIM_CALCON)

Offset address: 0x18

Reset value: 0xffff ffff

Table 684. Calibration Counter Overflow Value Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ccntval[31:16]															
R/W															

Table 685. Calibration Counter Overflow Value Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Ccntval[15:0]															
R/W															

Table 686. Calibration Counter Overflow Value Configuration Register (3)

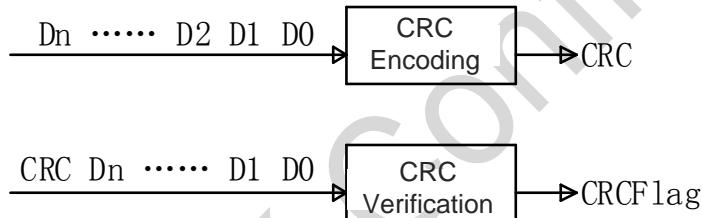
Bit	Flag	Description
31:0	ccntval	Calibration counter overflow value

21 CRC16

21.1 Overview

The cyclic redundancy check (CRC) calculation unit obtains a CRC calculation result of an arbitrary byte data based on a fixed generator polynomial. In applications, CRC technique is primarily used to verify the correctness and integrity of data transmission or data storage.

A typical application of the CRC algorithm in data transmission is shown in the below figure.

**Figure 106. Schematic Diagram of CRC Application**

21.2 Functional Description

The algorithm of this module follows the definition of ISO/IEC13239, applying a 16-bit length CRC, and the calculation polynomial is $x^{16} + x^{12} + x^5 + 1$ with the initial value as 0xFFFF.

The functions of this module includes,

- CRC coding and CRC check
- 3 bit width access modes including 8 bits, 16 bits and 32 bits.
 - Examples of input data at 8-bit width are 0x00, 0x11, 0x22, 0x33, 0x44, 0x55 and 0x660x77.
 - Examples of input data at 16-bit width are 0x1100, 0x3322, 0x5544 and 0x7766.
 - Examples of input data at 32-bit width are 0x33221100 and 0x77665544.

21.2.1 CRC Encoding Mode

With the encoding mode, it encodes the original data to calculate its CRC value. The operation flow is as follows.

Step 1, write 0xFFFF to CRC_RESULT.CRC16 to initialize the CRC calculation.

Step 2, write the original data to be encoded into the CRC_DATA register in the 8-bit/16-bit/32-bit formation.

Step 3, Read CRC_RESULT.CRC16, that is the CRC value.

21.2.2 CRC Verification Mode

With the verification mode, it verifies whether the encoded data has been tampered with. The operation flow is as follows.

Step 1, write 0xFFFF to CRC_RESULT[15:0] to initialize the CRC calculation.

Step 2, the encoded data is sequentially written into the CRC_DATA register in the 8-bit/16-bit/32-bit formation.

Notes:

when writing the CRC value to the CRC_DATA register in 8-bit formation, the lower 8 bits should be written first, followed by the higher 8 bits.

Step 3, read CRC_RESULT.FLAG to determine if the encoded data has been tampered with.

21.3 Register

Base address: 0x4002 0900

Register	Offset Address	Description
CRC_RESULT	0x04	CRC result register. The result is obtained by reading this register after the calculation is completed.
CRC_DATA	0x80-0xFF	CRC data register for inputting the data to be calculated.

21.3.1 CRC Result Register (CRC_RESULT)

Offset address: 0x04

Reset value: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															FLAG
R															RO

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RESULT															
R/W															

Bit	Flag	Description
31:17	Reserved	
16	FLAG	<p>Check result flag. 0: current check fails, 1: current check succeeds.</p> <p>Register [16] is a read-only bit with write operation ineffective. When performing CRC check, this bit should be read after all data and 16-bit CRC code input into the data register. Reading as 1 means the check is successful.</p>
15:0	RESULT	<p>This register is used to update and save the results of each CRC calculation. After the calculation, a 16-bit CRC encoding result can be obtained by reading this register.</p> <p>According to the standard, after the calculation completion, the 16-bit CRC code value is the reverse value of the calculation register value. Therefore, the reading result of this register [15:0] is the reverse value of the current [15:0] register.</p>

Notes:

1. According to the standard, after the calculation completion, the 16-bit CRC code value is the reverse value of the calculation register value. Therefore, the reading result of this register [15:0] is the reverse value of the current [15:0] register.
2. Register [16] is a read-only bit and the write operation is ineffective. When performing CRC check, this bit should be read after all data and 16-bit CRC code is input into the data register. Reading as 1 means the check is successful.

21.3.2 CRC Data Register (CRC_DATA)

Offset address: 0x80-0xFF

Reset value: 0x0000 0000

Table 687. CRC Data Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRC_DATA[31:16]															
W															

Table 688. CRC Data Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC_DATA[15:0]															
W															

Table 689. CRC Data Register (3)

Bit	Flag	Description
31:0	CRC_DATA	<p>This register is used to input the data that needs to be calculated.</p> <p>The address of this register is an address range (0x80 - 0xFF). Operation on any address within this range is regarded as operating the register. The purpose of this definition is for the software to use the STM instruction to perform continuous 32-bit data writing operation on this register so as to speed up the calculation. This register supports 8/16/32-bit input mode.</p>

Notes:

1. The address of this register is a address range (8'h80-8'hFF). Operation of any address within this range is regarded as operating the register. The purpose of this definition is for the software to use the STM instruction to perform continuous 32-bit data writting operation to this register so as to speed up the calculation. This register supports 8/16-bit input mode as well.

22 ADC Module

22.1 Module Introduction

The external analog signal needs to be converted to a digital signal for further processing by the MCU. The L005 integrates a 12-bit high-precision and high-slew-rate successive approximation analog-to-digital converter (SAR ADC) module. The module is capable of the following features.

- 12-bit conversion accuracy.
- 1 M SPS conversion speed.
- 12 conversion channels including 9 pin channels, built-in temperature sensor, built-in 1.2 V reference voltage and 1/3 power supply voltage
- 4 reference sources including power supply voltage, ExRef pin, built-in 1.5 V reference voltage and built-in 2.5 V reference voltage.
- ADC voltage input range: 0 ~ Vref.
- 3 conversion modes including single conversion, continuous conversion and accumulated conversion.
- The conversion rate of the ADC being configurable by the software
- Built-in signal amplifier converting high-impedance signals.
- Supports of on-chip peripherals automatically triggering ADC conversion, which reduces chip power consumption and improves real-time conversion effectively.

22.2 ADC Block Diagram

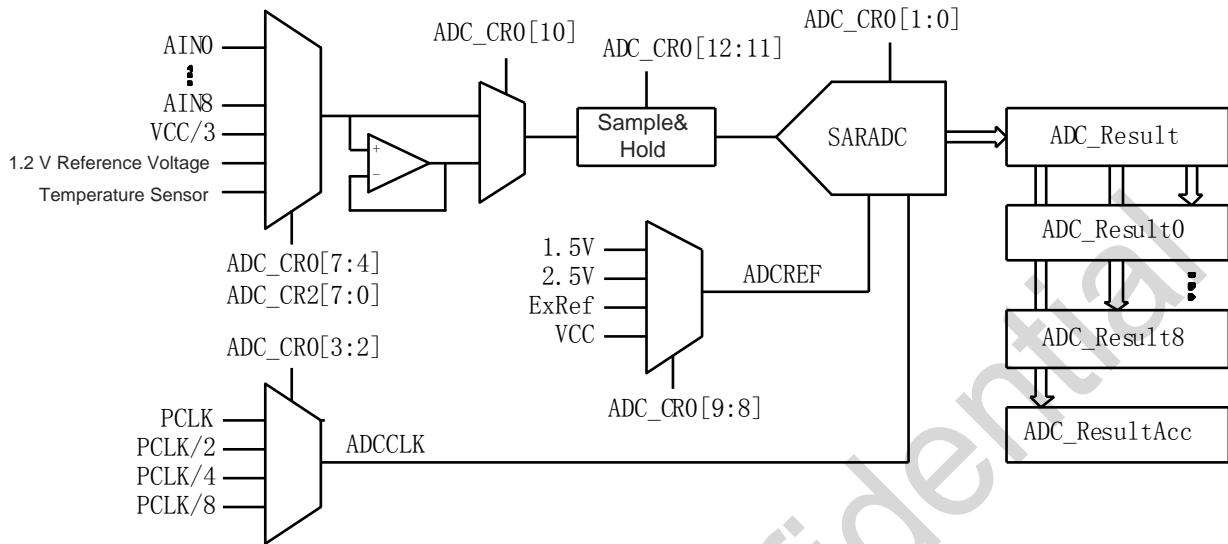


Figure 107. Schematic Block Diagram of the ADC

22.3 Conversion Timing and Speed

The conversion timing of the ADC is shown in the figure below. A complete ADC conversion consists of a sampling process and a successive comparison process. The sampling process requires 4~12 AdcClks, which are configured by ADC_CR0.SAM. The successive comparison process requires 16 AdcClks. Therefore, a total of 20 to 28 AdcClks are required for 1 ADC conversion.

The unit of ADC conversion speed is SPS representing how many ADC conversions per second. The ADC conversion speed is calculated as: *the frequency of the AdcClk / the number of AdcClks required for a single ADC conversion*.

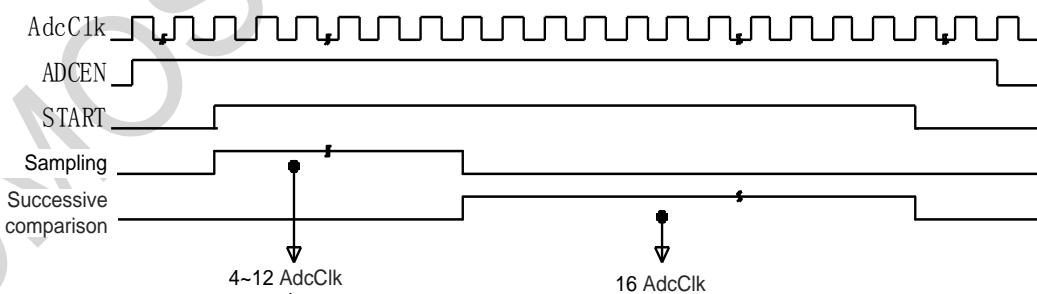


Figure 108. ADC Conversion Timing Diagram

The ADC conversion speed is related to the ADC reference voltage and VCC voltage. The maximum conversion speed is shown in the below table.

Table 690. Maximum Conversion Speed

ADC Reference Voltage	VCC Voltage	Maximum Conversion Speed	Maximum AdcClk Frequency
Internal 1.5 V	1.8 ~ 5.5 V	200 kSPS	4 MHz
Internal 2.5 V	2.8 ~ 5.5 V	200 kSPS	4 MHz
VCC / ExRef	1.8 ~ 2.4 V	200 kSPS	4 MHz
VCC / ExRef	2.4 ~ 2.7 V	500 kSPS	16 MHz
VCC / ExRef	2.7 ~ 5.5 V	1 MSPS	24 MHz

22.4 Single Conversion Mode

In single conversion mode, only 1 conversion is performed after the ADC is started, and the 12 ADC channels can be converted all together. This mode can be initiated either by setting the ADC_CR0.START bit or by setting an external trigger for ADC_CR1[9:0]. Once the ADC conversion for the selected channel is complete, the ADC_CR0.START bit is automatically cleared and the conversion result is stored in the ADC_result register.

The operation flow for starting the ADC single conversion by the START bit is as follows.

Step 1, configure the corresponding bits of P0ADS ~ P3ADS and configure the ADC channel to be converted as an analog port.

Step 2, set P3ADS.6 to 1 and configure the ADC external reference voltage pin as an analog port.

Note: this step can be skipped if the ADC reference voltage does not select an external voltage reference pin.

Step 3, set BGR_CR.BGR_EN to 1, and enable the BGR module.

Step 4, set ADC_CR0.ADCEN to 1 to enable the ADC module.

Step 5, delay 20 us, wait for the ADC and BGR module to start up.

Step 6, set ADC_CR1.CT to 0 to select the single conversion mode.

Step 7, configure ADC_CR0. SREF to select the reference voltage of the ADC.

Step 8, configure ADC_CR0. SAM and ADC_CR0. CLKSEL to set the conversion speed of the ADC.

Step 9, configure ADC_CR0. SEL to select the channel to be converted.

Step 10, set ADC_CR0.START to 1 to start ADC single conversion.

Step 11, wait for ADC_CR0.START to become 0, then read the ADC_result register to get the ADC conversion result.

Step 12, If it needs to convert other channels, repeat Step9 ~ Step11.

Step 13, set ADC_CR0.ADCEN and BGR_CR.BGR_EN to 0 to disable the ADC module and BGR module.

The operation flow for starting the ADC single conversion by external trigger is as follows.

Step 1: configure the corresponding bits of P0ADS~P3ADS and configure the ADC channel to be converted as an analog port.

Step 2: set P3ADS.6 to 1, and configure the ADC external reference voltage pin as an analog port.

Note: this step can be skipped if the ADC reference voltage does not select an external voltage reference pin.

Step 3, set BGR_CR.BGR_EN to 1 and enable the BGR module.

- Step 4, set ADC_CR0.ADCEN to 1 to enable the ADC module.
- Step 5, delay 20 us, wait for the ADC and BGR module to start up.
- Step 6, set ADC_CR1.CT to 0 to select single conversion mode.
- Step 7, set ADC_HT to 0x00.
- Step 8, set ADC_CR1.HtCmp to 1 to enable the ADC high threshold comparison function.
- Step 9, set ADC_CR0. IE to 1 to enable ADC interrupt.
- Step 10, enable the ADC interrupt in the NVIC interrupt vector table.
- Step 11, configure ADC_CR0. SREF to select the reference voltage of the ADC.
- Step 12, configure ADC_CR0. SAM and ADC_CR0. CLKSEL to set the conversion speed of the ADC.
- Step 13, configure ADC_CR0. SEL to select the channel to be converted.
- Step 14, set ADC_IFR to 0x00 to clear the ADC interrupt flag.
- Step 15, configure ADC_CR1. TRIGS1 and ADC_CR1. TRIGS0 to select the external trigger condition.
- Step 16, the ADC module generates an interrupt when the external trigger condition triggers the ADC to complete the conversion. The ADC_result register can be read by users in the ADC interrupt service routine to get the ADC conversion result.
- Step 17, If it needs to convert other channels, repeat Step13 ~ Step16.
- Step 18, set ADC_CR0.ADCEN and BGR_CR.BGR_EN to 0 to disable the ADC module and BGR module.

22.5 Continuous Conversion Mode

In continuous conversion mode, it can perform multiple conversions on multiple channels in sequence after ADC starts. The convertible ADC channels are AIN0 ~ AIN7. The total number of ADC conversions is configured in ADCCR2.ADCCNT. The channel to be converted is configured in ADC_CR2[7:0]. This mode can be initiated either by setting the ADC_CR0.START bit or setting an external trigger for ADC_CR2[9:0]. After the continuous conversion is started, the ADC module sequentially converts the channels to be converted among AIN0 ~ AIN7 until the total number of conversions are completed. After the ADC module completes the total number of conversions, the ADC_IFR.CONT_INTF bit is automatically set to 1, and the conversion result is stored in the ADC_result 0 ~ ADC_result 7 registers corresponding to the conversion channel. If the total number of conversions is greater than the number of ADC channels to be converted, only the latest conversion result is saved in the ADC_result 0 ~ ADC_result 7 register.

The figure below demonstrates the process of 10 consecutive conversions of AIN0, AIN1, and AIN5. After the START bit is set by the register, the internal state machine of the ADC sequentially converts AIN0, AIN1, and AIN5 until the count value of ADCCNT becomes zero.

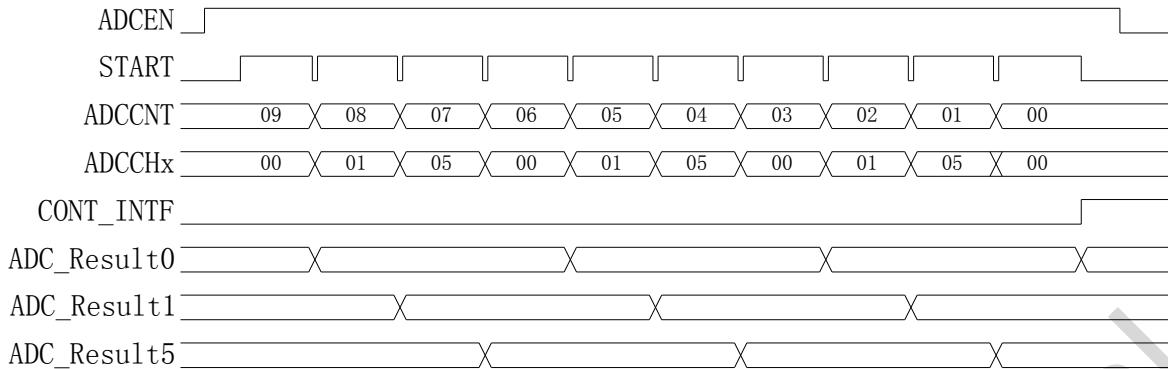


Figure 109. Continuous Conversion Process Example

The operation flow for starting the ADC continuous conversion using the START bit is as follows.

Step 1, configure the corresponding bits of P0ADS ~ P3ADS and configure the ADC channel to be converted as an analog port.

Step 2, set P3ADS.6 to 1 and configure the ADC external reference voltage pin as an analog port.

Notes: this step can be skipped if the ADC reference voltage does not select an external voltage reference pin.

Step 3, set BGR_CR.BGR_EN to 1 and enable the BGR module.

Step 4, set ADC_CR0.ADCEN to 1 to enable the ADC module.

Step 5, delay 20 us to wait for the ADC and BGR module starting-up.

Step 6, set ADC_CR1.CT to 1 to select continuous conversion mode.

Step 7, set ADC_CR1[14:12] to 0 to disable the conversion result comparison function.

Step 8, configure ADC_CR2. ADCCNT to select the total number of conversions for continuous conversion.

Step 9, configure ADC_CR0. SREF to select the reference voltage of the ADC.

Step 10, configure ADC_CR0. SAM and ADC_CR0. CLKSEL to set the conversion speed of the ADC.

Step 11, configure ADC_CR2[7:0] to select the channel to be converted.

Step 12, set ADC_ICLR. CONT_INTC is 0 to clear the ADC_IFR. CONT_INTF flag.

Step 13, set ADC_CR0. StateRst to 1, the continuous conversion state is reset then.

Step 14, set ADC_CR0.START to 1 to start ADC continuous conversion.

Step 15, wait for ADC_IFR. CONT_INTF becomes 1, then read the ADC_result 0 ~ ADC_result 7 register to get the conversion result of the corresponding channel.

Step 16, If it needs to convert other channels, repeat step 11 ~ step 15.

Step 17, set ADC_CR0.ADCEN and BGR_CR.BGR_EN to 0 to disable the ADC module and BGR module.

22.6 Continuous Conversion Accumulation Mode

In the continuous conversion accumulation mode, it can perform multiple conversions on multiple channels and accumulate the result of each conversion after ADC starts. The convertible ADC channels are AIN 0 ~ AIN 7. The total number of ADC conversions is configured by ADCCR2.ADCCNT. The channel to be converted is configured by ADC_CR2[7:0]. This mode can

be initiated either by setting the ADC_CR0.START bit or by setting an external trigger for ADC_CR2[9:0]. After the continuous conversion is started, the ADC module sequentially converts the channels to be converted among AIN 0 ~ AIN 7 until the total number of conversions are completed. After the total number of conversions are completed, the ADC_IFR.CONT_INTF bit is set automatically, and the accumulated value of the conversion result is stored in the ADC_result_acc register.

The figure below demonstrates the process of 10 consecutive conversions of AIN 0, AIN 1, and AIN 5. After the START bit is set by the register, the internal state machine of the ADC sequentially converts AIN 0, AIN 1, and AIN 5 until the count value of ADCCNT becomes zero. The ADC_result_acc register is automatically increased each time the conversion is completed. The conversion results of AIN 0, AIN 1, and AIN 5 presented in the figure are 0x010, 0x020, and 0x040, respectively.

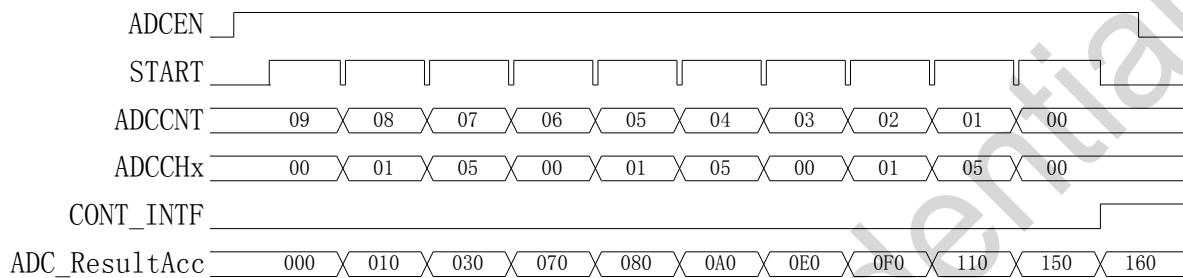


Figure 110. Example of ADC Continuous Conversion Accumulation Process

The operation flow for starting the ADC continuous conversion accumulation operation using the START bit is as follows.

Step 1, configure the corresponding bits of P0ADS ~ P3ADS and configure the ADC channel to be converted as an analog port.

Step 2, set P3ADS.6 to 1, and configure the ADC external reference voltage pin as an analog port.

Notes: this step can be skipped if the ADC reference voltage does not select an external voltage reference pin.

Step 3, set BGR_CR.BGR_EN to 1 to enable the BGR module.

Step 4, set ADC_CR0.ADCEN to 1 to enable the ADC module.

Step 5, delay 20 us to wait for the ADC and BGR module starting-up.

Step 6, set ADC_CR1.CT to 1 to select continuous conversion mode.

Step 7, set ADC_CR1[14:12] to 0 to disable the conversion result comparison function.

Step 8, set ADC_CR1.RACC_EN is 1 to enable the automatic conversion of ADC conversion.

Step 9, configure ADC_CR2.ADCCNT to select the total number of conversions for continuous conversion.

Step 10, configure ADC_CR0.SREF to select the reference voltage of the ADC.

Step 11, configure ADC_CR0.SAM and ADC_CR0.CLKSEL to set the conversion speed of the ADC.

Step 12, configure ADC_CR2[7:0] to select the channel to be converted.

Step 13, set ADC_ICLR.CONT_INTC is 0 to clear the ADC_IFR.CONT_INTF flag.

Step 14, set ADC_CR1.RACC_CLR is 0 to clear the ADC_result_acc register.

Step 15, set ADC_CR0.StateRst to 1, then the continuous conversion state is reset.

Step 16, set ADC_CR0.START to 1 to start ADC continuous conversion.

Step17, wait for ADC_IFR.CONT_INTF becomes 1, theb read the ADC_result_acc register to get the accumulated value of

the conversion result.

Step 18, If it needs to convert other channels, repeat step 12 ~ step 17.

Step 19, set ADC_CR0.ADCEN and BGR_CR.BGR_EN to 0 to disable the ADC module and BGR module.

22.7 ADC Conversion Results Comparison

When the ADC conversion completes, it compares the ADC conversion result to the threshold set by the user with supports of higher threshold comparison, lower threshold comparison and interval value comparison. This function needs to set the corresponding control bits HtCmp, LtCmp and RegCmp to 1. It performs automatic monitoring of the analog quantity until the ADC conversion result meets users' expectations then triggers interrupt request for user program having related processing.

For higher threshold comparison, ADC_IFR.HHT_INTF is set when the ADC conversion result is within the range of [ADC_LT, 4095]. ADC_IFR.HHT_INTF is cleared by writing 0 to ADC_ICLR.HHT_INTC.

For lower threshold comparison, ADC_IFR.LLT_INTF is set when the ADC conversion result is within the range of [0, ADC_LT]. Writing 0 to ADC_ICLR.LLT_INTC clears ADC_IFR.LLT_INTF.

For interval value comparison, ADC_IFR.REG_INTF is set when the ADC conversion result is within the range of [ADC_LT, ADC_HT]. Writing 0 to ADC_IFR.REG_INTC clears ADC_IFR.REG_INTF.

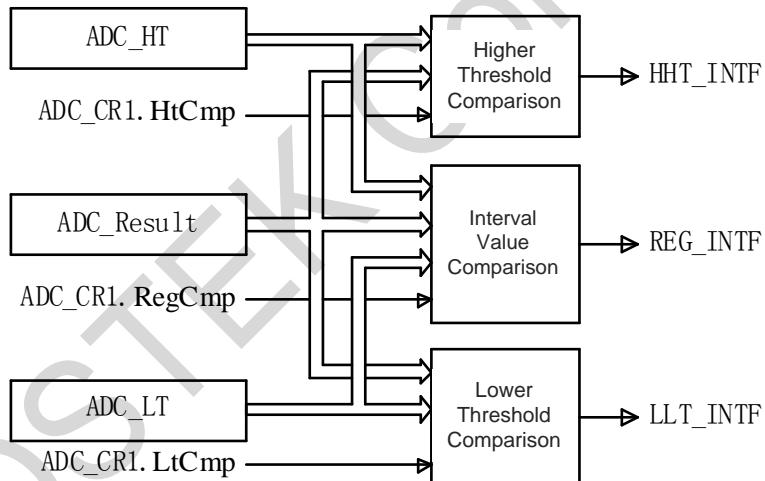


Figure 111. ADC Conversion Results Comparison

22.8 ADC Interrupt

The ADC interrupt request is listed in the below table.

Table 691. ADC Interrupt

Interrupt Source	Interrupt Flag	Interrupt Enabling
ADC continuous conversion completes	ADC_IFR.CONT_INTF	ADC_CR0.IE
ADC conversion result is within the interval value range	ADC_IFR.REG_INTF	
conversion result is within the higher threshold threshold range.	ADC_IFR.HHT_INTF	
conversion result is within the lower threshold range.	ADC_IFR.LLT_INTF	

22.9 Measuring Ambient Temperature via Temperature Sensor

The output voltage of the temperature sensor changes with the ambient temperature, so the corresponding ambient temperature can be calculated according to the output voltage of the temperature sensor. The ambient temperature can be measured when the measurement channel of the ADC module selects the output voltage of the temperature sensor.

The calculation is as follows.

$$\text{Ambient temperature} = 25 + 0.0839 \times V_{\text{ref}} \times (\text{AdcValue} - \text{Trim})$$

Among them,

1. V_{ref} is the reference voltage of the current ADC module with value as 1.5 or 2.5 V.
2. AdcValue is the measured output voltage of the temperature sensor with value range as 0 ~ 4095 V.
3. Trim is a 16-bit calibration value read from the Flash memory when calculating. The storage address is listed in the table below.

Table 692. Measuring Ambient Temperature via Temperature Sensor

ADC Reference Voltage	Calibration Value Storage Address	Calibration Value Accuracy
Internal 1.5 V	0x00100C34	± 3 °C
Internal 2.5 V	0x00100C36	± 3 °C

The calculation example is as follows:

Condition 1: $V_{\text{ref}} = 2.5 \text{ V}$, AdcValue = 0x7E5, Trim = 0x76C.

Temperature 1: $25 + 0.0839 \times 2.5 \times (0x7E5 - 0x76C) = 50^{\circ}\text{C}$.

Condition 2: $V_{\text{ref}} = 1.5 \text{ V}$, AdcValue = 0x72D, Trim = 0x76C

Temperature 2: $25 + 0.0839 \times 1.5 \times (0x72D - 0x76C) = 17^{\circ}\text{C}$.

The operating procedure for measuring ambient temperature via ADC is as follows.

Step 1, set BGR_CR.BGR_EN to 3 to enable the BGR module and temperature sensor module.

Step 2, set ADC_CR0.ADCEN to 1 to enable the ADC module.

Step 3, delay 20 us to wait for the ADC and BGR module starting-up.

Step 4, set ADC_CR1.CT to 0 to select the single conversion mode.

Step 5, configure ADC_CR0. SREF to select the reference voltage of the ADC as internal 1.5 V or internal 2.5 V.

Step 6, configure ADC_CR0. SAM and ADC_CR0. CLKSEL to set the conversion speed of the ADC.

Step 7, set ADC_CR0. SEL as 0x0A and select the channel to be converted as the output of the temperature sensor.

Step 8, set ADC_CR0. BUFEN to 1 to enable the input signal amplifier.

Step 9, set ADC_CR0.START to 1 to start ADC single conversion.

Step 10, wait for ADC_CR0.START to become 0, then read the ADC_result register to get the ADC conversion result.

Step 11, set ADC_CR0.ADCEN and BGR_CR.BGR_EN to 0 to disable the ADC module and BGR module.

Step 12, read the temperature sensor calibration value and calculate the current ambient temperature according to the formula.

22.10 ADC Module Register

Base address: 0x40002400

Table 693. ADC Register

Register	Offset Address	Description
ADC_CR0	0x004	ADC configuration register 0
ADC_CR1	0x008	ADC configuration register 1
ADC_CR2	0x00C	ADC configuration register 2
ADC_result0	0x030	ADC channel 0 conversion result
ADC_result1	0x034	ADC channel 1 conversion result
ADC_result2	0x038	ADC channel 2 conversion result
ADC_result3	0x03C	ADC channel 3 conversion result
ADC_result4	0x040	ADC channel 4 conversion result
ADC_result5	0x044	ADC channel 5 conversion result
ADC_result6	0x048	ADC channel 6 conversion result
ADC_result7	0x04C	ADC channel 7 conversion result
ADC_result8	0x050	ADC channel 8 conversion result
ADC_result_acc	0x054	ADC conversion result accumulated value
ADC_HT	0x058	ADC comparison higher threshold
ADC_LT	0x05C	ADC comparison lower threshold
ADC_IFR	0x060	ADC interrupt flag register
ADC_ICLR	0x064	ADC interrupt clearing register
ADC_result	0x068	ADC conversion result

22.10.1 ADC Configuration Register 0 (ADC_CR0)

Offset address 0x004

Reset value 0x000013F0

Table 694. ADC Configuration Register 0 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 695. ADC Configuration Register 0 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
StateRst	IE	Res.	SAM	BUFEN	SREF	SEL		CLKSEL	START		ADCEN				
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 696. ADC Configuration Register 0 (3)

Bit	Flag	Description
31:16	Reserved	Reserved
15	StateRst	ADC continuous conversion state control. 1: reset ADC continuous conversion state 0: invalid
14	IE	ADC interrupt control. 1: enable 0: disable
13	Reserved	Reserved
12:11	SAM	ADC sampling period selection 00: 4 sampling periods 01: 6 sampling periods 10:8 sampling period 11:12 sampling period
10	BUFEN	ADC input signal amplifier control. 0: the amplifier is disabled and the external input signal is connected to ADC directly. 1: the amplifier is enabled and the external input signal is connected to ADC after it's amplified by the amplifier. It is used for high-impedance signal.
9:8	SREF	ADC reference voltage selection. 00: internal 1.5 V 01: internal 2.5 V 10: external reference voltage ExRef (P3.6) 11: power supply voltage

Bit	Flag	Description
7:4	SEL	ADC conversion channel selection (single conversion mode). 0000: select channel 0 input P 2.4 0001: select channel 1 input P 2.6 0010: select channel 2 input P 3.2 0011: select channel 3 input P 3.3 0100: select channel 4 input P 3.4 0101: select channel 5 input P 3.5 0110: select channel 6 input P 3.6 0111: select channel 7 input P 0.1 1000: select channel 8 input P 0.2 1001: 1/3 VCC Notes: ADC_CR0.BUFEN must be 1 1010: built-in temperature sensor output voltage (Note: ADC_CR0.BUFEN must be 1.) 1011: internal reference 1.2 V output voltage (Note: ADC_CR0.BUFEN must be 1.)
3:2	CLKSEL	ADC clock selection 00: PCLK clock 01: PCLK clock divided by 2 10: PCLK clock divided by 4 11: PCLK clock divided by 8
1	START	ADC conversion control. 1: Start ADC conversion 0: Stop ADC conversion
0	ADCEN	ADC enabling control 1: enable ADC 0: disable ADC

22.10.2 ADC Configuration Register 1 (ADC_CR1)

Offset address: 0x0008

Reset value: 0x00007000

Table 697. ADC Configuration Register 1 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 698. ADC Configuration Register 1 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RACC_CLR	RegCmp	HtCmp	LtCmp	HtCmp	CT	TRIGS1				TRIGS0					
R/W	R/W	R/W	R/W	R/W	R/W	R/W				R/W					

Table 699. ADC Configuration Register 1 (3)

Bit	Flag	Description
31:16	Reserved	Reserved
15	RACC_CLR	<p>ADC conversion result accumulated value register clearing. 1: ineffective 0: ADC conversion result accumulated value register (ADC_result_acc) is cleared.</p> <p>Notes: this bit is read as 0. Please pay special attention to the value of this bit to prevent malfunction when operating this register.</p>
14	RegCmp	<p>ADC interval value comparison function enabling. 1: enable 0: disable</p>
13	HtCmp	<p>ADC higher threshold comparison function enabling. 1: enable 0: disable</p>
12	LtCmp	<p>ADC lower threshold comparison function enabling. 1: enable 0: disable</p>
11	RACC_EN	<p>ADC conversion result automatic accumulation function enabling. 1: enable 0: disable</p>
10	CT	<p>ADC conversion mode selection. 1: continuous conversion mode 0: single conversion mode</p>

Bit	Flag	Description
9:5	TRIGS1	<p>ADC conversion automatic triggering selection 2.</p> <p>00000: disable the automatic triggering of ADC conversion.</p> <p>00001: timer 0 interrupt, trigger ADC conversion automatically.</p> <p>00010: timer 1 interrupt, trigger ADC conversion automatically.</p> <p>00011: timer 2 interrupt, trigger ADC conversion automatically.</p> <p>00100: LPTimer interrupt, trigger ADC conversion automatically.</p> <p>00101: timer 4 interrupt, trigger ADC conversion automatically.</p> <p>00110: timer 5 interrupt, trigger ADC conversion automatically.</p> <p>00111: timer 6 interrupt, trigger ADC conversion automatically.</p> <p>01000: UART0 interrupt, trigger ADC conversion automatically.</p> <p>01001: UART1 interrupt, trigger ADC conversion automatically.</p> <p>01010: LPUART interrupt, trigger ADC conversion automatically.</p> <p>01011: VC0 interrupt, trigger ADC conversion automatically.</p> <p>01100: VC1 interrupt, trigger ADC conversion automatically.</p> <p>01101: RTC interrupt, trigger ADC conversion automatically.</p> <p>01110: PCA interrupt, trigger ADC conversion automatically.</p> <p>01111: SPI interrupt, trigger ADC conversion automatically.</p> <p>10000: P01 interrupt, trigger ADC conversion automatically.</p> <p>10001: P02 interrupt, trigger ADC conversion automatically.</p> <p>10010: P03 interrupt, trigger ADC conversion automatically.</p> <p>10011: P14 interrupt, trigger ADC conversion automatically.</p> <p>10100: P15 interrupt, trigger ADC conversion automatically.</p> <p>10101: P23 interrupt, trigger ADC conversion automatically.</p> <p>10110: P24 interrupt, trigger ADC conversion automatically.</p> <p>10111: P25 interrupt, trigger ADC conversion automatically.</p> <p>11000: P26 interrupt, trigger ADC conversion automatically.</p> <p>11001: P27 interrupt, trigger ADC conversion automatically.</p> <p>11010: P31 interrupt, trigger ADC conversion automatically.</p> <p>11011: P32 interrupt, trigger ADC conversion automatically.</p> <p>11100: P33 interrupt, trigger ADC conversion automatically.</p> <p>11101: P34 interrupt, trigger ADC conversion automatically.</p> <p>11110: P35 interrupt, trigger ADC conversion automatically.</p> <p>11111: P36 interrupt, trigger ADC conversion automatically.</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. The interrupt of TIM4/5/6 triggers ADC automatic conversion. It is required to enable the corresponding interrupt of TIM4/5/6. Moreover, it needs to configure the spread frequency of the Advanced Timer and the interrupt trigger selection register TIMX_CR to select the interrupt source that can trigger the ADC. 2. It uses the rising edge of each interrupt flag to trigger the ADC. If repeated triggering is needed, it needs to clear the interrupt flag. Don't enable the NVIC interrupt If it does not need to enter interrupt service routine.

Bit	Flag	Description
4:0	TRIGS0	<p>ADC conversion automatic triggering selection 2.</p> <p>00000: enable trigger ADC conversion automatically.</p> <p>00001: select Timer 0 interrupt, trigger ADC conversion automatically.</p> <p>00010: select Timer 1 interrupt, trigger ADC conversion automatically.</p> <p>00011: select Timer 2 interrupt, trigger ADC conversion automatically.</p> <p>00100: select LPTimer interrupt, trigger ADC conversion automatically.</p> <p>00101: select Timer 4 interrupt, trigger ADC conversion automatically.</p> <p>00110: select Timer 5 interrupt, trigger ADC conversion automatically.</p> <p>00111: select Timer 6 interrupt, trigger ADC conversion automatically.</p> <p>01000: select UART0 interrupt, trigger ADC conversion automatically.</p> <p>01001: select UART1 interrupt, trigger ADC conversion automatically.</p> <p>01010: select LPUART interrupt, trigger ADC conversion automatically.</p> <p>01011: select VCO interrupt, trigger ADC conversion automatically.</p> <p>01100: select VC1 interrupt, trigger ADC conversion automatically.</p> <p>01101: select RTC interrupt, trigger ADC conversion automatically.</p> <p>01110: select PCA interrupt, trigger ADC conversion automatically.</p> <p>01111: select SPI interrupt, trigger ADC conversion automatically.</p> <p>10000: select P01 interrupt, trigger ADC conversion automatically.</p> <p>10001: select P02 interrupt, trigger ADC conversion automatically.</p> <p>10010: select P03 interrupt, trigger ADC conversion automatically.</p> <p>10011: select P14 interrupt, trigger ADC conversion automatically.</p> <p>10100: select P15 interrupt, trigger ADC conversion automatically.</p> <p>10101: select P23 interrupt, trigger ADC conversion automatically.</p> <p>10110: select P24 interrupt, trigger ADC conversion automatically.</p> <p>10111: select P25 interrupt, trigger ADC conversion automatically.</p> <p>11000: select P26 interrupt, trigger ADC conversion automatically.</p> <p>11001: select P27 interrupt, trigger ADC conversion automatically.</p> <p>11010: select P31 interrupt, trigger ADC conversion automatically.</p> <p>11011: select P32 interrupt, trigger ADC conversion automatically.</p> <p>11100: select P33 interrupt, trigger ADC conversion automatically.</p> <p>11101: select P34 interrupt, trigger ADC conversion automatically.</p> <p>11110: select P35 interrupt, trigger ADC conversion automatically.</p> <p>11111: select P36 interrupt, trigger ADC conversion automatically.</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. The interrupt of TIM4/5/6 triggers ADC automatic conversion. It is required to enable the corresponding interrupt of TIM4/5/6. Moreover, it needs to configure the spread frequency of the Advanced Timer and the interrupt trigger selection register TIMX_CR to select the interrupt source that can trigger the ADC. 2. It uses the rising edge of each interrupt flag to trigger the ADC. If repeated triggering is needed, it needs to clear the interrupt flag. Don't enable the NVIC interrupt if it does not need to enter interrupt service routine.

22.10.3 ADC Configuration Register 2 (ADC_CR2)

Offset address: 0x000C

Reset value: 0x00000000

Table 700. ADC Configuration Register 2 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 701. ADC Configuration Register 2 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADCCNT								CH7EN	CH6EN	CH5EN	CH4EN	CH3EN	CH2EN	CH1EN	CH0EN
R/W								R/W							

Table 702. ADC Configuration Register 2 (3)

Bit	Flag	Description
31:16	Reserved	Reserved
15:8	ADCCNT	ADC continuous conversion number configuration. 0: 1 conversion 1: 2 conversions 255: 256 conversions
7	CH7EN	ADC continuous conversion enabling on channel 7. 1: enable 0: disable
6	CH6EN	ADC continuous conversion enabling on channel 6. 1: enable 0: disable
5	CH5EN	ADC continuous conversion enabling on channel 5. 1: enable 0: disable
4	CH4EN	ADC continuous conversion enabling on channel 4. 1: enable 0: disable
3	CH3EN	ADC continuous conversion enabling on channel 3. 1: enable 0: disable
2	CH2EN	ADC continuous conversion enabling on channel 2. 1: enable 0: disable
1	CH1EN	ADC continuous conversion enabling on channel 1. 1: enable 0: disable
0	CH0EN	ADC continuous conversion enabling on channel 0. 1: enable

		0: disable
--	--	------------

22.10.4 ADC Conversion Result for Channel 0 (ADC_result0)

Offset address: 0x030

Reset value: 0x00000000

Table 703. ADC Conversion Result for Channel 0 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 704. ADC Conversion Result for Channel 0 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result0											
-				R											

Table 705. ADC Conversion Result for Channel 0 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	result0	ADC conversion Result for channel 0

22.10.5 ADC Conversion Result for Channel 1 (ADC_result1)

Offset address: 0x034

Reset value: 0x00000000

Table 706. ADC Conversion Result for Channel 1 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 707. ADC Conversion Result for Channel 1 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result 1											
-				R											

Table 708. ADC Conversion Result for Channel 1 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	Result 1	ADC conversion result for channel 1.

22.10.6 ADC Conversion Result for Channel 2 (ADC_result2)

Offset address: 0x038

Reset value: 0x00000000

Table 709. ADC Conversion Result for Channel 2 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 710. ADC Conversion Result for Channel 2 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result 2											
-				R											

Table 711. ADC Conversion Result for Channel 2 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	Result 2	ADC conversion result for channel 2.

22.10.7 ADC Conversion Result for Channel 3 (ADC_result3)

Offset address: 0x03C

Reset value: 0x00000000

Table 712. ADC Conversion Result for Channel 3 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 713. ADC Conversion Result for Channel 3 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result 3											
-				R											

Table 714. ADC Conversion Result for Channel 3 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	Result 3	ADC conversion result for channel 3.

22.10.8 ADC Conversion Result for Channel 4 (ADC_result4)

Offset address: 0x040

Reset value: 0x00000000

Table 715. ADC Conversion Result for Channel 4 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 716. ADC Conversion Result for Channel 4 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result 4											
-				R											

Table 717. ADC Conversion Result for Channel 4 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	result4	ADC conversion result for channel 4.

22.10.9 ADC Conversion Result for Channel 5 (ADC_result5)

Offset address: 0x044

Reset value: 0x00000000

Table 718. ADC Conversion Result for Channel 5 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 719. ADC Conversion Result for Channel 5 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result 5											
-				R											

Table 720. ADC Conversion Result for Channel 5 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	Result 5	ADC conversion result for channel 5.

22.10.10 ADC Conversion Result for Channel 6 (ADC_result6)

Offset address: 0x048

Reset value: 0x00000000

Table 721. ADC Conversion Result for Channel 6 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 722. ADC Conversion Result for Channel 6 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result 6											

-	R
---	---

Table 723. ADC Conversion Result for Channel 6 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	Result 6	ADC conversion result for channel 6.

22.10.11 ADC Conversion Result for Channel 7 (ADC_result7)

Offset address: 0x04C

Reset value: 0x00000000

Table 724. ADC Conversion Result for Channel 7 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 725. ADC Conversion Result for Channel 7 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result 7											
-				R											

Table 726. ADC Conversion Result for Channel 7 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	Result 7	ADC conversion result for channel 7.

22.10.12 ADC Conversion Result for Channel 8 (ADC_result8)

Offset address: 0x050

Reset value: 0x00000000

Table 727. ADC Conversion Result for Channel 8 (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 728. ADC Conversion Result for Channel 8 (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result 8											
-				R											

Table 729. ADC Conversion Result for Channel 8 (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	Result 8	ADC conversion result for channel 8.

22.10.13 ADC Conversion Result Accumulated Value (ADC_result_acc)

Offset address: 0x054

Reset value: 0x00000000

Table 730. ADC Conversion Result Accumulated Value (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												Result_acc[19:16]			
-												R			

Table 731. ADC Conversion Result Accumulated Value (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Result_acc[15:0]												R			
-												R			

Table 732. ADC Conversion Result Accumulated Value (3)

Bit	Flag	Description
31:20	Reserved	Reserved
19:0	Result_acc	Accumulated value of the ADC conversion result.

22.10.14 ADC Comparison Higher Threshold (ADC_HT)

Offset address: 0x058

Reset value: 0x00000FFF

Table 733. ADC Comparison Higher Threshold (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												R			

Table 734. ADC Comparison Higher Threshold (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				HT											
-				R/W											

Table 735. ADC Comparison Higher Threshold (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	HT	The higher threshold for ADC conversion result comparison.

22.10.15 ADC Comparison Lower Threshold (ADC_LT)

Offset address: 0x05C

Reset value: 0x00000000

Table 736. ADC Comparison Lower Threshold (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 737. ADC Comparison Lower Threshold (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				LT											
-				R/W											

Table 738. ADC Comparison Lower Threshold (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	LT	The lower threshold for ADC conversion result comparison.

22.10.16 Interrupt Flag Register (ADC_IFR)

Offset address: 0x060

Reset value: 0x00000000

Table 739. Interrupt Flag Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 740. Interrupt Flag Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								CONT_INTF	REG_INTF		HHT_INTF		LLT_INTF		
-								R	R		R		R		

Table 741. Interrupt Flag Register (3)

Bit	Flag	Description
31:4	Reserved	Reserved
3	CONT_INTF	ADC continuous conversion completion flag. 1: complete 0: not complete
2	REG_INTF	ADC conversion result interval value comparison flag. 1: ADC conversion result is within the range - [ADC_LT, ADC_HT) 0: ADC conversion result is out of the range - [ADC_LT, ADC_HT)
1	HHT_INTF	ADC conversion result higher threshold comparison flag. 1: ADC conversion result is within the range - [ADC_HT, 4095] 0: ADC conversion result is out of the range - [ADC_HT, 4095]

Bit	Flag	Description
0	LLT_INTF	ADC conversion result lower threshold comparison flag. 1: ADC conversion result is within the range - [0, ADC_LT) 0: ADC conversion result is out of the range - [0, ADC_LT)

22.10.17 Interrupt Flag Clearing Register (ADC_ICLR)

Offset address: 0x064

Reset value: 0x00000004

Table 742. Interrupt Flag Clearing Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 743. Interrupt Flag Clearing Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								CONT_INTC	REG_INTC	HHT_INTC	LLT_INTC				
-								W	W	W	W				

Table 744. Interrupt Flag Clearing Register (3)

Bit	Flag	Description
31:4	Reserved	Reserved
3	CONT_INTC	Writing 0 to clear the continuous conversion completion flag. Writing 1 is ineffective.
2	REG_INTC	Writing 0 to clear the ADC conversion interval value comparison flag. Writing 1 is ineffective.
1	HHT_INTC	Writing 0 to clear the ADC conversion higher threshold comparison flag. Writing 1 is ineffective.
0	LLT_INTC	Writing 0 to clear the ADC conversion lower threshold comparison flag. Writing 1 is ineffective.

22.10.18 ADC Result (ADC_result)

Offset address 0x068

Reset value 0x00000000

Table 745. ADC Result (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 746. ADC Result (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Result											
-				R											

Table 747. ADC Result (3)

Bit	Flag	Description
31:12	Reserved	Reserved
11:0	Result	ADC conversion result.

23 VC Module

23.1 Introduction to Analog Voltage Comparator (VC)

The analog voltage comparator (VC) is used to compare the magnitude of the two input analog voltages and output high/low levels based on the comparison result. When the + input voltage is higher than the - input voltage, the voltage comparator output is high, otherwise when the + input voltage is lower than the - input voltage, the voltage comparator output is low. The analog voltage comparator integrated in this product provides the following features.

- Support voltage comparison function.
- Support internal 64-step VCC partial voltage (the divided source voltage should be greater than 1.8 V)
- Support using the 8 external input ports and the reference voltage of the on-chip BGR output as the voltage comparator's input.
- Support 3 software configurable interrupt triggering modes including high level trigger, rising edge trigger and falling edge trigger.
- Support using the output of the voltage comparator as the input of the Base Timer and LPTimer gate ports.
- Support using the output of the voltage comparator as the brake input or capture input of the Advanced Timer.
- Support operating in ultra-low power mode and waking up the chip from ultra-low power mode via the interrupt output of the voltage comparator.
- Support filtering time configured by software to enhance the chip's anti-interference ability.

23.2 Voltage Comparator Block Diagram

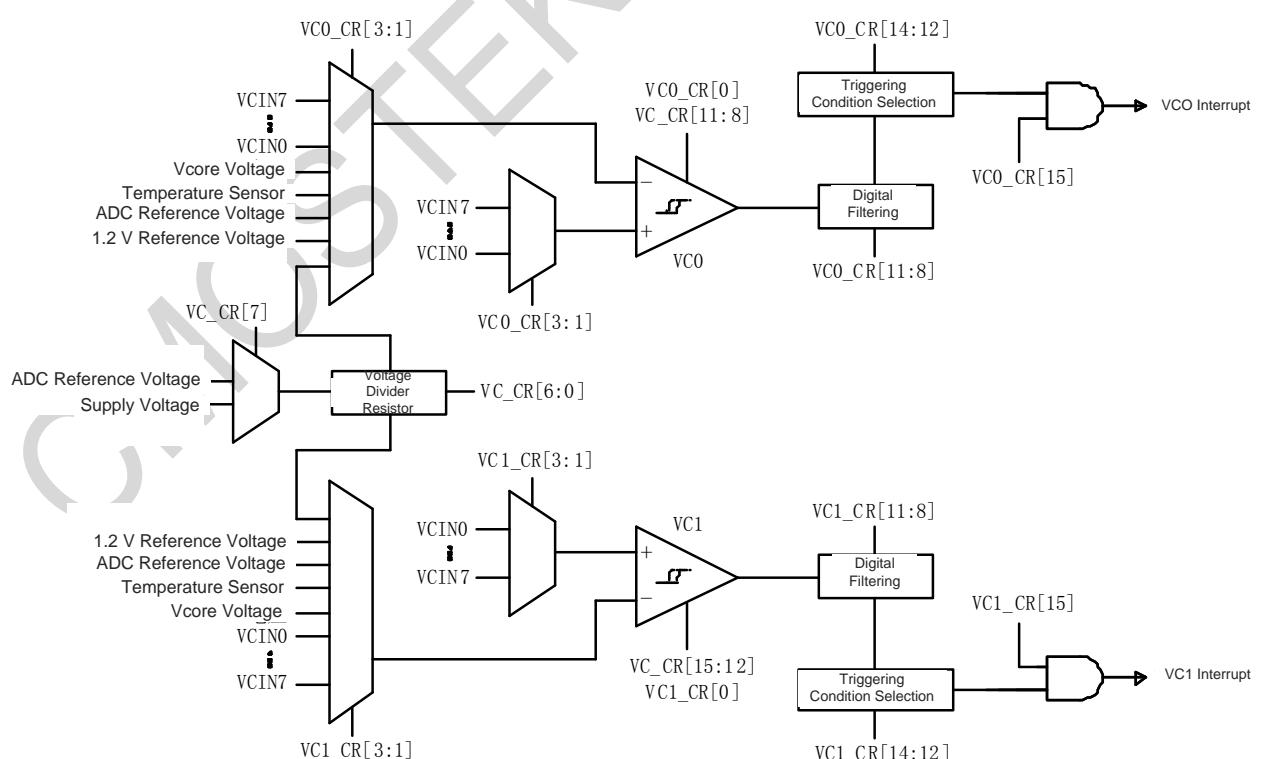


Figure 112. VC Block Diagram

23.3 Setup/Response Time

For a voltage comparator, the time from the VC enabling or VC input voltage changing to the correct output depends on the BIAS_SEL control bit in the VC Control Register (VC_CR). The higher the current, the faster the VC response. The typical VC response time is adjustable from 50 ns to 20 us with 4 options.

If using a reference voltage (1.5 V or 2.5 V) from the internal BGR as the comparison reference voltage, the internal BGR startup time is approximately 20 us. The voltage comparator can output normally until the internal BGR becomes stable.

23.4 Filtering Time

Besides the setup/response time of the voltage comparator, users can set a longer filtering time to filter out system noise, such as the high current noise when a motor stops.

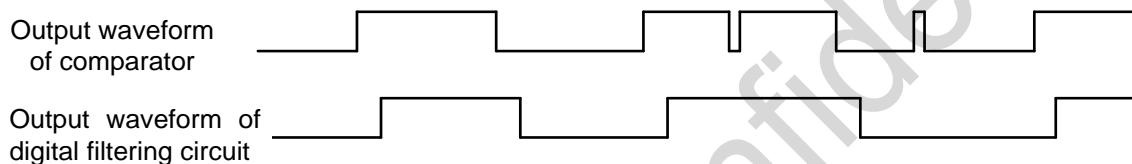


Figure 113. VC Filtering Response Time

23.5 Hysteresis Function

The voltage comparator can select the hysteresis function and the hysteresis function is shown in the below figure if the function is enabled.

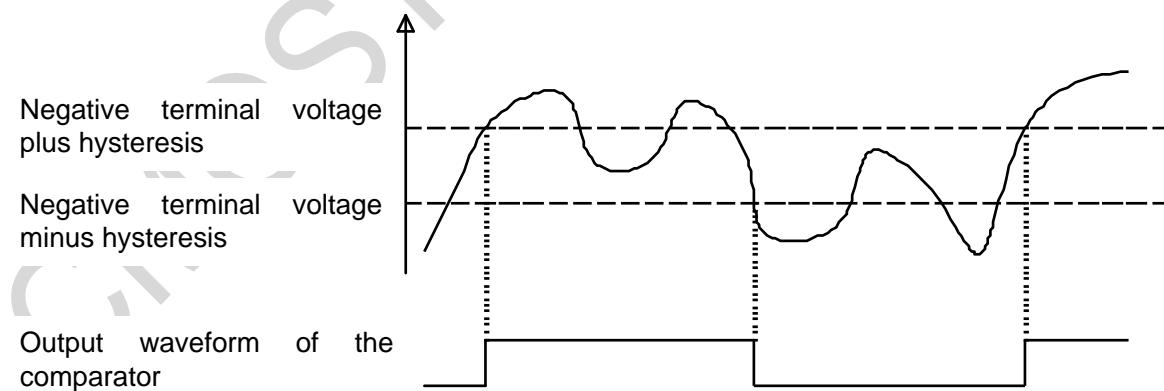


Figure 114. VC Hysteresis Function

23.6 VC Register

Base address: 0x40002400

Table 748. VC Register

Register	Offset address	Description
VC_CR	0x010	VC0/1 configuration register 0
VC0_CR	0x014	VC0 configuration register
VC1_CR	0x018	VC1 configuration register
VC0_OUT_CFG	0x01C	VC0 output configuration register
VC1_OUT_CFG	0x020	VC1 output configuration register
VC_INT	0x024	VC interrupt register

23.6.1 VC Configuration Register (VC_CR)

Offset address: 0x010

Reset value: 0x000000020

Table 749. VC Configuration Register

Bit	Flag	Description
31:16	Reserved	Reserved
15:14	VC1_HYS_SEL	<p>VC1 hysteresis selection.</p> <p>00: no hysteresis function</p> <p>01: hysteresis voltage is about 10 mV</p> <p>10: hysteresis voltage is about 20 mV</p> <p>11: hysteresis voltage is about 30 mV</p>
13:12	VC1_BIAS_SEL	<p>VC1 power consumption selection (the greater the power consumption, the shorter the response time).</p> <p>00: 300 nA</p> <p>01: 1.2 uA</p> <p>10: 10 uA (need to start BGR with startup time of 30us approximately)</p> <p>11: 20 uA (need to start BGR with startup time of 30us approximately)</p>

Bit	Flag	Description
11:9	VC0_HYS_SEL	<p>VC0 hysteresis selection.</p> <p>00: no hysteresis function</p> <p>01: hysteresis voltage is about 10 mV</p> <p>10: hysteresis voltage is about 20 mV</p> <p>11: hysteresis voltage is about 30 mV</p>
9:8	VC0_BIAS_SEL	<p>VC0 power consumption selection (the greater the power consumption, the shorter the response time).</p> <p>00: 300 nA</p> <p>01: 1.2 uA</p> <p>10: 10 uA (need to start BGR with startup time of 30us approximately)</p> <p>11: 20 uA (need to start BGR with startup time of 30us approximately)</p>
7	VC_REF2P5_SEL	<p>VC_DIV reference voltage V_{ref} selection.</p> <p>0: VCC</p> <p>1: reference voltage selected by ADC_CR0.SREF</p>
6	VC_DIV_EN	<p>6-bit DAC enabling.</p> <p>1: enable</p>
5:0	VC_DIV	<p>6-bit DAC configuration.</p> <p>000000: 1/64 V_{ref}</p> <p>000001: 2/64 V_{ref}</p> <p>000010: 3/64 V_{ref}</p> <p>000011: 4/64 V_{ref}</p> <p>...</p> <p>111110: 63/64 V_{ref}</p> <p>111111: V_{ref}</p>

23.6.2 VC0 Configuration Register (VC0_CR)

Offset address: 0x014

Reset value: 0x00000000

Table 750. VC0 Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 751. VC0 Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	level	rising	falling	debounce_time			FLTEN	N_SEL			P_SEL			EN	
R/W	R/W	R/W	R/W	R/W			R/W	R/W			R/W			R/W	

Table 752. VC0 Configuration Register (3)

Bit	Flag	Description
31:16	Reserved	Reserved
15	IE	VC interrupt enabling. 1: enable, 0: disable.
14	level	VC output signal high level triggers INT flag
13	rising	VC output signal rising edge triggers INT flag
12	Falling	VC output signal falling edge triggers INT flag
11:9	debounce_time	VC output filtering time configuration. 111: filtering time is about 28.8 ms 110: filtering time is about 7.2 ms 101: filtering time is about 1.8 ms 100: filtering time is about 450 us 011: filtering time is about 112 us 010: filtering time is about 28 us 001: filtering time is about 14 us 000: filtering time is about 7 us Notes: the configuration of filtering time is effective only when FLTEN = 1.
8	FLTEN	1: enable VC filtering 0: non VC filtering function
7:4	N_SEL	Voltage comparator '-' input selection. 0000: select channel 0 input P2.3 0001: select channel 1 input P2.5 0010: select channel 2 input P3.2 0011: select channel 3 input P3.3 0100: select channel 4 input P3.4 0101: select channel 5 input P3.5 0110: select channel 6 input P3.6 0111: select channel 7 input P0.1 1000: voltage divider resistor output voltage 1001: built-in temperature sensor output voltage 1010: internal reference 1.2 V output voltage 1011: REF of ADC 1100: LDO output voltage

Bit	Flag	Description
3:1	P_SEL	Voltage comparator '+' input selection. 000: select channel 0 input P2.3 001: select channel 1 input P2.5 010: select channel 2 input P3.2 011: select channel 3 input P3.3 100: select channel 4 input P3.4 101: select channel 5 input P3.5 110: select channel 6 input P3.6 111: select channel 7 input P0.1
0	EN	Voltage comparator enabling. 1: enable 0: disable

23.6.3 VC1 Configuration Register (VC1_CR)

Offset address: 0x018

Reset value: 0x00000000

Table 753. VC1 Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 754. VC1 Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	Level	Rising	Falling	Debounce_time		FLTEN	N_SEL			P_SEL		EN			
R/W	R/W	R/W	R/W	R/W		R/W	R/W			R/W		R/W			

Table 755. VC1 Configuration Register (3)

Bit	Flag	Description
31:16	Reserved	Reserved
15	IE	VC interrupt enabling. 1: enable 0: disable
14	level	VC output signal triggering interrupt selection. 1: enable high level triggering INT flag 0: disable high level triggering INT flag
13	rising	
12	falling	VC output signal triggering interrupt selection. 1: enable falling edge triggering INT flag 0: disable falling edge triggering INT flag

Bit	Flag	Description
11:9	debounce_time	<p>VC output filtering time configuration.</p> <p>111: filtering time is about 28.8 ms 110: filtering time is about 7.2 ms 101: filtering time is about 1.8 ms 100: filtering time is about 450 us 011: filtering time is about 112 us 010: filtering time is about 28 us 001: filtering time is about 14 us 000: filtering time is about 7 us</p> <p>Notes: the configuration of filtering time is effective only when FLTEN = 1.</p>
8	FLTEN	<p>1: enable VC filtering 0: non VC filtering function</p>
7:4	N_SEL	<p>Voltage comparator '-' input selection.</p> <p>0000: select channel 0 input P2.3 0001: select channel 1 input P2.5 0010: select channel 2 input P3.2 0011: select channel 3 input P3.3 0100: select channel 4 input P3.4 0101: select channel 5 input P3.5 0110: select channel 6 input P3.6 0111: select channel 7 input P0.1 1000: voltage divider resistor output voltage 1001: built-in temperature sensor output voltage 1010: internal reference 1.2 V output voltage 1011: REF of ADC 1100: LDO output voltage</p>
3:1	P_SEL	<p>Voltage comparator '+' input selection.</p> <p>000: select channel 0 input P2.3 001: select channel 1 input P2.5 010: select channel 2 input P3.2 011: select channel 3 input P3.3 100: select channel 4 input P3.4 101: select channel 5 input P3.5 110: select channel 6 input P3.6 111: select channel 7 input P0.1</p>
0	EN	<p>Voltage comparator enabling.</p> <p>1: enable 0: disable</p>

23.6.4 VC0 Output Configuration Register (VC0_OUT_CFG)

Offset address: 0x01C

Reset value: 0x00000000

Table 756. VC0 Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 757. VC0 Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Brake	TIM6	INV_Timer6	TIM5	INV_Timer5	TIM4	INV_Timer4	PCAEI	PCACAP0	INV_PCA	TM3ECLK	TIM3G	TIM2G	TIM1G	TIM0G	INV_Timer
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 758. VC0 Output Configuration Register (3)

Bit	Flag	Description
31:16	Reserved	Reserved
15	Brake	VC0 as the brake control of advanced timer. 1: enable, 0: disable.
14	TIM6	The enabling of outputting VC0 filtering result to TIM6 capture input 1: enable, 0: disable.
13	INV_TIM6	The enabling of outputting the reverse VC0 filtering result to TIM6. 1: reverse is enabled, 0: reverse is disable with the input is in the same direction as the VC output.
12	TIM5	The enabling of outputting VC0 filtering result to TIM5 capture input. 1: enable, 0: disable.
11	INV_TIM5	The enabling of outputting the reverse VC0 filtering result to TIM5. 1: reverse is enabled, 0: reverse is disable and the input is in the same direction as the VC output.
10	TIM4	The enabling of outputting VC0 filtering result to TIM4 capture input. 1: enable, 0: disable.
9	INV_TIM4	The enabling of outputting the reverse VC0 filtering result to TIM4. 1: reverse is enabled, 0: reverse is disable and the input is in the same direction as the VC output.
8	PCAEI	The enabling of outputting VC0 filtering result to PCA external clock. 1: enable, 0: disable.
7	PCACAP0	The enabling of outputting VC0 filtering result to PCA capture 0. 1: enable, 0: disable.
6	INV_PCA	The enabling of outputting the reverse VC0 filtering result to PCA. 1: reverse is enabled, 0: reverse is disable, and the input is in the same direction with the VC output.
5	TIM3ECLK	The enabling of outputting VC0 filtering result to LPTIMER external clock. 1: enable, 0: disable.
4	TIM3G	The enabling of outputting VC0 filtering result to LPTIMER3 GATE. 1: enable, 0: disable.
3	TIM2G	The enabling of outputting VC0 filtering result to TIM2 GATE. 1: enable, 0: disable.
2	TIM1G	The enabling of outputting VC0 filtering result to TIM1 GATE. 1: enable, 0: disable.
1	TIM0G	The enabling of outputting VC0 filtering result to TIM0 GATE. 1: enable, 0: disable.

Bit	Flag	Description
0	INV_Timer	The enabling of outputting the reverse VC0 filtering result to TIM0/1/2 and LPTimer. 1: reverse is enabled, 0: reverse is disable, and the input is in the same direction with the VC output.

23.6.5 VC1 Configuration Register (VC1_OUT_CFG)

Offset address: 0x020

Reset value: 0x00000000

Table 759. VC1 Output Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 760. VC1 Output Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Brake	TIM6	INV_Timer6	TIM5	INV_Timer5	TIM4	INV_Timer4	PCAEI	PCACAO	INV_PCA	TM3ECLK	TIM3G	TIM2G	TIM1G	TIM0G	INV_Timer
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 761. VC1 Output Configuration Register (3)

Bit	Flag	Description
31:16	Reserved	Reserved
15	brake	VC1 as the brake control of Advanced Timer. 1: enable, 0: disable.
14	TIM6	The enabling of outputting VC1 filtering result to TIM6 capture input 1: enable, 0: disable.
13	INV_TIM6	The enabling of outputting the reverse VC1 filtering result to TIM6. 1: reverse is enabled, 0: reverse is disable and the input is in the same direction with the VC output.
12	TIM5	The enabling of outputting VC1 filtering result to TIM5 capture input. 1: enable, 0: disable.
11	INV_TIM5	The enabling of outputting the reverse VC1 filtering result to TIM5. 1: reverse is enabled, 0: reverse is disable and the input is in the same direction with the VC output.
10	TIM4	The enabling of outputting VC1 filtering result to TIM4 capture input. 1: enable, 0: disable.
9	INV_TIM4	The enabling of outputting the reverse VC1 filtering result to TIM4. 1: reverse is enabled, 0: reverse is disable and the input is in the same direction with the VC output.
8	PCAEI	The enabling of outputting VC1 filtering result to PCA external clock. 1: enable, 0: disable.

7	PCACAP1	The enabling of outputting VC1 filtering result to PCA capture 0. 1: enable, 0: disable.
6	INV_PCA	The enabling of outputting the reverse VC1 filtering result to PCA. 1: reverse is enabled, 0: reverse is disable and the input is in the same direction with the VC output.
5	TIM3ECLK	The enabling of outputting VC1 filtering result to LPTIMER external clock. 1: enable, 0: disable.
4	TIM3G	The enabling of outputting VC1 filtering result to LPTIMER3 GATE. 1: enable, 0: disable.
3	TIM2G	The enabling of outputting VC1 filtering result to TIM2 GATE. 1: enable, 0: disable.
2	TIM1G	The enabling of outputting VC1 filtering result to TIM1 GATE. 1: enable, 0: disable.
1	TIM0G	The enabling of outputting VC1 filtering result to TIM0 GATE. 1: enable, 0: disable.
0	INV_Timer	The enabling of outputting the reverse VC1 filtering result to TIM0/1/2 and LPTimer. 1: reverse is enabled, 0: reverse is disable and the input is in the same direction with the VC output.

23.6.6 VC Interrupt Register (VC_IFR)

Offset address: 0x024

Reset value: 0x00000000

Table 762. VC Interrupt Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 763. VC Interrupt Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										VC1_Filter	VC0_Filter	VC1_INTF	VC0_INTF		
-										R	R	R/W	R/W		

Table 764. VC Interrupt Register (3)

Bit	Flag	Description
31:4	Reserved	Reserved
3	VC1_Filter	The state after VC1 filtering
2	VC0_Filter	The state after VC0 filtering
1	VC1_INTF	VC1 interrupt flag. 1: Interrupt occurs, 0: Interrupt does not occur. Writing 0 clears the interrupt flag and writing 1 is ineffective.
0	VC0_INTF	VC0 interrupt flag. 1: Interrupt occurs, 0: Interrupt does not occur. Writing 0 clears the interrupt flag and writing 1 is ineffective.

24 LVD Module

24.1 Introduction to LVD

LVD can be used to monitor the voltage of the VCC and chip pins. When the comparison between the monitored voltage and the LVD threshold meets the triggering condition, the LVD generates an interrupt or reset signal, and users can perform some urgent tasks based on the signals.

LVD supports the following features.

- 4 monitoring sources including VCC, P03, P23 and P25.
- 16-step threshold voltage, flexible for various applications.
- 8 trigger conditions which are the combination of high level, rising edge and falling edge.
- 2 types of triggering results, reset and interrupt.
- 8th-order filtering configuration to prevent false triggering.
- Hysteresis function with strong anti-interference capability.

24.2 LVD Block Diagram

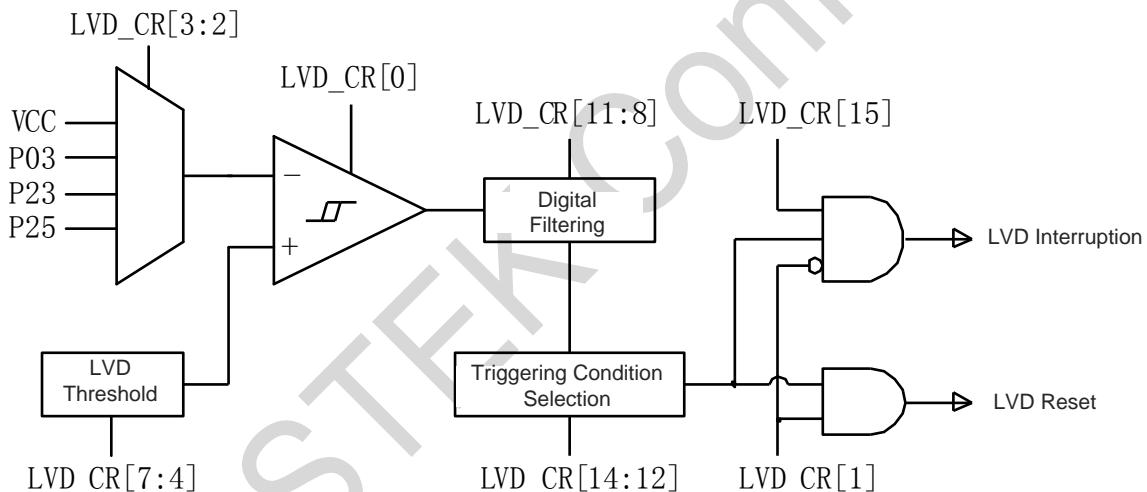


Figure 115. LVD Block Diagram

24.3 Digital Filtering

If the operating environment of the chip is harsh, it may contain noise signals in the output of the hysteresis comparator. When the digital filtering module is enabled, the noise signal with a pulse width shorter than LVD_CR.Debounce_time in the output waveform of the hysteresis comparator can be filtered out. If the digital filtering module is disabled, the input and output signals of the digital filtering module are the same. The filtering function and the digital filtering module enabling is shown in the below figure.

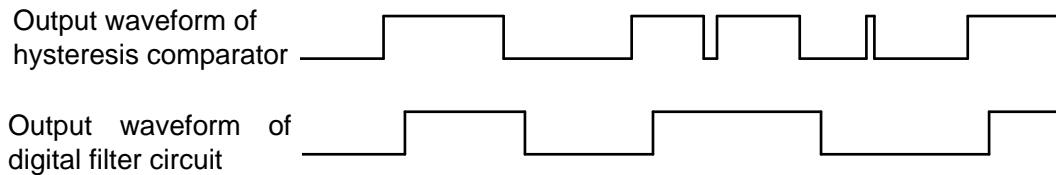


Figure 116. LVD Filtering Output

24.4 Hysteresis Function

With hysteresis function support in the built-in voltage comparator of the LVD, it will not output reverse signal until the input signal is 20 mV above or below the threshold voltage. The hysteresis function enhances the chip's immunity to interference, as shown in the following figure.

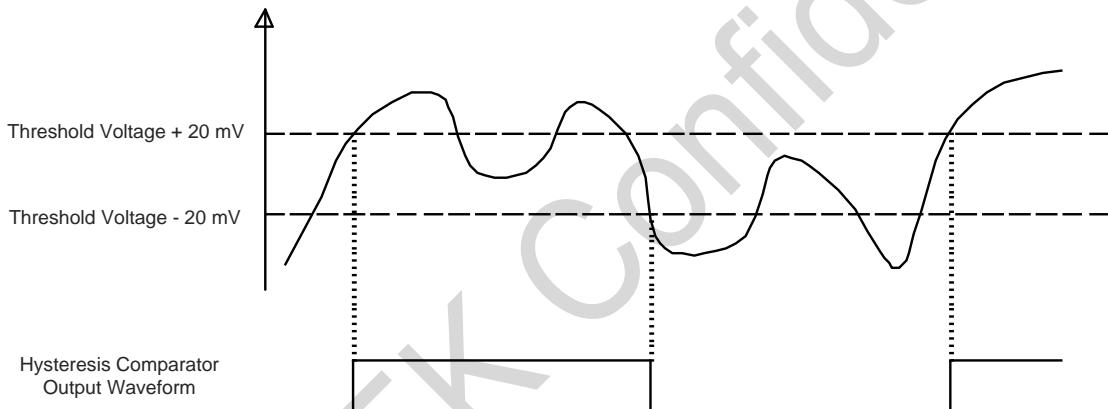


Figure 117. LVD Hysteresis Response

24.5 Configuration Examples

24.5.1 LVD Configured as Low Voltage Reset

In this mode, the MCU is reset when the monitored voltage is lower than the threshold voltage.

The configuration flow is as follows.

- Step 1, configure LVD_CR.Source_sel to select the voltage source to be monitored.
- Step 2, configure LVD_CR.VTDS and select the threshold voltage of LVD.
- Step 3, configure LVD_CR.Debounce_time and select the LVD filter time.
- Step 4, configure LVD_CR.FLTEN to enable LVD filtering.
- Step 5, set LVD_CR.HTEN to 1, select high level to trigger LVD action.
- Step 6, set LVD_CR.ACT to 1, and select LVD action as reset.

Step 7, set LVD_CR.LVDEN to 1, and enable LVD.

24.5.2 LVD Configured as Voltage Change Interrupt

In this mode, an interrupt is generated when the monitored voltage is above or below the threshold voltage.

The configuration flow is as follows.

Step 1, configure LVD_CR.Source_sel to select the voltage source to be monitored.

Step 2, configure LVD_CR.VTDS and select the threshold voltage of LVD.

Step 3, configure LVD_CR.Debounce_time and select the LVD filtering time.

Step 4, configure LVD_CR.FLTEN to enable LVD filtering.

Step 5, set LVD_CR.RTEN and LVD_CR.FTEN to 1 and select the level change to trigger the LVD action.

Step 6, set LVD_CR.ACT to 0 and select LVD action as interrupt.

Step 7, set LVD_CR.IE to 1 and enable LVD interrupt.

Step 8, enable the LVD interrupt in the NVIC interrupt vector table.

Step 9, set LVD_CR.LVDEN to 1 and enable LVD.

Step 10, write 0x00 to LVD_IFR in the interrupt service routine to clear the interrupt flag.

24.6 LVD Register

Base address: 0x40002400

Table 765. LVD Register

Register	Offset Address	Description
LVD_CR	0x028	LVD configuration register
LVD_IFR	0x02C	LVD interrupt flag register

24.6.1 LVD Configuration Register (LVD_CR)

Offset address 0x028

Reset value 0x00000100

Table 766. LVD Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 767. LVD Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	HTEN	RTEN	FTEN	Debounce_time			FLTEN	VTDS			Source_sel	ACT	LVDEN		
R/W	R/W	R/W	R/W	R/W			R/W	R/W			R/W	R/W	R/W		

Table 768. LVD Configuration Register (3)

Bit	Flag	Description
31:16	Reserved	Reserved
15	IE	LVD interrupt enabling. 1: enable 0: disable
14	HTEN	Be enabled by high level (the monitored voltage is below the threshold voltage). 1: enable 0: disable
13	RTEN	Be enabled by rising edge (the monitored voltage transition from above the threshold voltage to below the threshold voltage) 1: enable 0: disable
12	FTEN	Be enabled by falling edge (the monitored voltage transition from below the threshold voltage to above the threshold voltage). 1: enable 0: disable
11:9	Debounce_time	Digital filtering time configuration. 111: about 29 ms 110: about 7.3 ms 101: about 1.8 ms 100: about 480 us 011: about 130 us 010: about 50 us 001: about 40 us 000: about 30 us Notes: the configuration of filtering time is effective only when FTEN = 1.
8	FLTEN	Digital filtering function configuration. 1: enable digital filtering 0: disable digital filtering

Bit	Flag	Description
7:4	VTDS	LVD monitored voltage selection. 1111: 3.3 V 1110: 3.2 V 1101: 3.1 V 1100: 3.0 V 1011: 2.9 V 1010: 2.8 V 1001: 2.7 V 1000: 2.6 V 0111: 2.5 V 0110: 2.4 V 0101: 2.3 V 0100: 2.2 V 0011: 2.1 V 0010: 2.0 V 0001: 1.9 V 0000: 1.8 V
3:2	Source_sel	LVD monitored source selection. 11: P2.5 port input voltage 10: P2.3 port input voltage 01: P0.3 port input voltage 00: VCC supply voltage
1	ACT	LVD triggered action selection. 1: system reset 0: NVIC interrupt
0	LVDEN	LVD enabling control. 1: enable 0: disable

24.6.2 LVD Interrupt Register (LVD_IFR)

Offset address: 0x02C

Reset value: 0x00000000

Table 769. LVD Interrupt Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 770. LVD Interrupt Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													INTF		
-													R/W		

Table 771. LVD Interrupt Register (3)

Bit	Flag	Description
31:1	Reserved	Reserved
0	INTF	LVD interrupt flag. 1: LVD interrupt occurs 0: LVD interrupt does not occur Write 0 to clear the interrupt flag. Writing 1 is ineffective.

25 Other Analog Register

Base address: 0x40002400

Table 772. Other Analog Register

Register	Offset address	Description
BGR_option	0x078	BGR control register

25.1 BGR Configuration Register (BGR_CR)

Offset address: 0x000

Reset value: 0x00000000

Table 773. BGR Configuration Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 774. BGR Configuration Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														TS_EN	BGR_EN
-														R/W	R/W

Table 775. BGR Configuration Register (3)

Bit	Flag	Description
31:2	Reserved	Reserved
1	TS_EN	Internal temperature sensor enabling. 1: enable 0: Disable Notes: the TS requires a approximately 20 us settling time for becoming stable after startup.
0	BGR_EN	Master BGR enabling. 1: enable 0: Disable Notes: it takes about 20 us for BGR to output a high precision reference voltage after it starts.

26 SWD Debug Interface

The JTAG/SWD of general-purpose MCU is a non-encrypted traditional JTAG/SWD debug interface with severe potential security risks. To ensure the security of client programs and the system, this product integrates a security authorization circuit on the SWD port. When the product is shipped from the factory, the SWD debug interface is configured by default. If a non-0XFF value is written in the security byte of the host software, the SWD debug port is disconnected automatically upon reset or the next power-on. The SWD debug interface cannot be set open by writing 0XFF in the security byte. The SWD debug interface can be set open only by erasing the chip completely.

Notes:

1. When users do not set the security byte, P27 and P31, configured with pull-up input, are automatically configured as the SWD debug ports. Users can also configure the debug interface as GPIO by configuring the SYSCTRL1.SWD_USED_IO register.
2. When users set the security byte to a value other than 0XFF, the P27 and P31 ports are automatically disconnected from the SWD debug port, meaning that the SWD debug function is not available. Users can configure the SYSCTRL1.SWD_USED_IO register to use the debug interface P31 and P27 as GPIOs.

26.1 SWD Authorization Key Operating Process

1. For a blank chip, the value of the security byte is 0XFF, thus the SWD is open by default.
2. Users develop software using Keil. After development completion, software can be downloaded directly through Keil or the writer.
3. After software program being download, users download authorization keys through the host computer.
4. After the MCU reset, the key takes effect immediately.
5. If users need SWD debug again, it needs to erase the chip to re-open the SWD channel.

26.2 Additional Function for SWD Debug

As ARM Cortex-M0+ CPU is embedded in this product, the debug function of this product is consistent with that of the Cortex-M0+. Moreover, the product offers some additional debug function to enhance user experience.

26.3 Debug Mode Module Operating State Control (DEBUG_ACTIVE)

Reset value: 0x000007FF (the register is valid only in SWD debug mode)

Reset value: 0x000007FF (the register is valid only in SWD debug mode)

Offset address: 0x038

Table 776. Debug Mode Module Operating State Control Register (1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

Table 777. Debug Mode Module Operating State Control Register (2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					TICK	RTC	WDT	PCA	TIM6	TIM5	TIM4	LPTIM	TIM2	TIM1	TIM0
-					R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 778. Debug Mode Module Operating State Control Register (3)

Bit	Flag	Description
31:12	Reserved	Reserved
10	TICK	SYSTICK_pause, pause SYSTICK count function in the SWD debug interface. 1: pause count function 0: perform count normally
9	RTC	RTC_pause, pause RTC count function in the SWD debug interface. 1: pause count function 0: perform count normally
8	WDT	WDT_pause, pause WDT count function in the WDT debug interface. 1: pause count function 0: perform count normally
7	PCA	PCA_pause, pause PCA count function in the PCA debug interface. 1: pause count function 0: perform count normally
6	TIM6	Pause timer 6 count function in the debug interface. 1: pause count function 0: perform count normally
5	TIM5	Pause timer 5 count function in the debug interface. 1: pause count function 0: perform count normally
4	TIM4	Pause timer 4 count function in the debug interface. 1: pause count function 0: perform count normally
3	LPTIM	Pause LPTimer count function in the debug interface. 1: pause count function 0: perform count normally
2	TIM2	Pause timer 2 count function in the debug interface. 1: pause count function 0: perform count normally
1	TIM1	Pause timer 1 count function in the debug interface. 1: pause count function 0: perform count normally

0	TIM0	Pause timer 0 count function in the debug interface. 1: pause count function 0: perform count normally
---	------	--------------------------------------------------------------------------------------------------------------

CMOSTEK Confidential

27 Appendix A: SysTick Timer

27.1 Introduction to SysTick Timer

To support multitasking, the OS needs to perform context switching periodically, so that it needs interrupt program execution by such hardware resources as timers. When the timer interrupt is generated, the processor will perform OS task scheduling in the exception processing, and perform OS maintenance work as well. The Cortex-M0 processor has a simple timer called SysTick , which is used to generate periodic interrupt requests.

SysTick is a 24-bit timer which counts down. When the timer count is reduced to 0, a programmable value is reloaded and a SysTick exception (exception number 15) is generated. This exception event causes the execution of the SysTick exception handler, which is part of the OS processing.

For systems that do not require an OS, the SysTick timer can be used for other purposes as well, such as timing, counting, or providing an interrupt source for tasks that require periodic execution. With flexible control function of the SysTick exception, If the SysTick exception is disabled, the SysTick timer can still be used for polling function, such as checking the current count value or polling the overflow flag.

27.2 SysTick Setup

As the reloaded value and current value of the SysTick timer are not defined during reset procedure, to prevent abnormal results it needs to follow the below configuration flow.

- Step 1, configure SysTick->CTRL. ENABLE as 0 to disable the SysTick.
- Step 2, configure SysTick->CTRL. CLKSOURCE to select the clock source of SysTick.
- Step 3, configure SysTick->LOAD to select the SysTick overflow period.
- Step 4, write any value to SysTick->VAL to clear SysTick->VAL and SysTick->CTRL. COUNTFLAG.
- Step 5, configure SysTick->CTRL. TICKINT as 1 to enable SysTick interrupt.
- Step 6, configure SysTick->CTRL. ENABLE as 1 to enable the SysTick.
- Step 7, read SysTick->CTRL in the interrupt service routine to clear the overflow flag.

Note: the SysTick overflow cycle is SysTick->LOAD + 1. The configuration example is as follows.

Table 779. Configuration Example

Timer Source	SysTick->LOAD	Overflow Cycle
RCH 4M	3999	1 ms
XTL 32.768K	327	10.01 ms

27.3 SysTick Register

Table 780. SysTick Register

Address	Name	CMSIS Symbol	Full Name
0xE000E010	SYS_CSR	SysTick->CTRL	SysTick control and state register
0xE000E014	SYS_RVR	SysTick->LOAD	SysTick reloaded value register
0xE000E018	SYS_CVR	SysTick->VAL	SysTick current value register

27.3.1 SysTick Control and State Register (CTRL)

Table 781. SysTick Control and State Register

Bit	Symbol	Function Description	Type	Reset Value
31:17	Reserved	-	-	-
16	COUNTFLAG	Systick timer overflow flag. 1: Systick timer underflow occurs 0: Systick timer overflow does not occur The COUNTFLAG flag can be cleared by reading this register.	RO	0
15:3	Reserved	-	-	-
2	CLKSOURCE	SysTick clock source selection. 1: HCLK 0: XTL	R/W	0
1	TICKINT	SysTick interrupt enabling. 1: enable 0: disable	R/W	0
0	ENABLE	SysTick timer enabling. 1: enable 0: disable	R/W	0

27.3.2 SysTick Reload Register (LOAD)

Table 782. SysTick Reload Register

Bit	Symbol	Function Description	Type	Reset value
31:24	Reserved	-	-	-
23:0	RELOAD	SysTick timer reloaded value	RW	Not defined

27.3.3 SysTick Current Value Register (VAL)

Table 783. SysTick Current Value Register

Bit	Symbol	Function Description	Type	Reset value
31:24	Reserved	-	-	-
23:0	CURRENT	Reading this register to get the current value of the SysTick timer. Writing any value to this register clears both this register and the COUNTFLAG flag.	RW	Not defined

28 Revise History

Table 784. Revise History Records

Version No.	Chapter	Description	Date
0.8	All	Initial version	2018-04-25

29 Contacts

CMOSTEK Microelectronics Co., Ltd. Shenzhen Branch

Address: 2/F Building 3, Pingshan Private Enterprise S.T. Park, Xili, Nanshan District, Shenzhen, Guangdong, China

Tel: +86-755-83231427

Post Code: 518057

Sales: sales@cmostek.com

Supports: support@cmostek.com

Website: www.cmostek.com

Copyright. CMOSTEK Microelectronics Co., Ltd. All rights are reserved.

The information furnished by CMOSTEK is believed to be accurate and reliable. However, no responsibility is assumed for inaccuracies and specifications within this document are subject to change without notice. The material contained herein is the exclusive property of CMOSTEK and shall not be distributed, reproduced, or disclosed in whole or in part without prior written permission of CMOSTEK. CMOSTEK products are not authorized for use as critical components in life support devices or systems without express written approval of CMOSTEK. The CMOSTEK logo is a registered trademark of CMOSTEK Microelectronics Co., Ltd. All other names are the property of their respective owners.