

SN32F800 Series

USER'S MANUAL

SN32F805

SONiX 32-Bit Cortex-M0 Micro-Controller

SONiX reserves the right to make change without further notice to any products herein to improve reliability, function or design. SONiX does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. SONiX products are not designed, intended, or authorized for use as components in systems intended, for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SONiX product could create a situation where personal injury or death may occur. Should Buyer purchase or use SONiX products for any such unintended or unauthorized application. Buyer shall indemnify and hold SONiX and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that SONiX was negligent regarding the design or manufacture of the part.

AMENDENT HISTORY

Version	Date	Description
1.0	2021/07/09	First released.
1.1	2022/03/22	1. Modify IHRC accuracy in 3.2.1.1 Internal High-speed RC Oscillator (IHRC) and 20.2 ELECTRICAL CHARACTERISTIC . 2. Modify typing errors of description in 3.4.1 AHB Clock Enable register (SYS1_AHBCLKEN) . 3. Modify pin circuit diagrams in 1.7 PIN CIRCUIT DIAGRAMS . 4. Modify typing errors in 3.1.4 EXTERNAL RESET .
1.2	2022/05/25	1. Modify Value of Temp Sensor Voltage at 25°C in 20.2 ELECTRICAL CHARACTERISTIC . 2. Modify Note in 7.8 TEMPERATURE SENSOR . 3. Remove description of auto-baud function in 15.6.6 UART n Interrupt Identification register (UARTn_I) (n=0,1). 4. Modify description of MUL in 15.5 BAUD RATE CALCULATION and 15.6.11 UART n Fractional Divider register (UARTn_FD) (n=0,1). 5. Modify typing errors in 9.2.1 COMPARATOR ENABLE .
1.3	2022/10/13	1. Modify description of ADC, SPI and I2C in 1.1 FEATURES . 2. Remove AIN13 of ADC in 7. 12+2 CHANNEL ANALOG TO DIGITAL CONVERTOR (ADC) .
1.4	2022/12/19	1. Modify address of L4BYTE[31:0] & H4BYTE[31:0] in 2.6 UNIQUE NUMBER . 2. Remove description of VREG150 in 1.6 PIN DESCRIPTIONS . 3. Remove description of P0.0/P0.1 in 1.8 PIN CHARACTERISTICS and 6 PERIPHERAL FUNCTION PIN ASSIGNMENT (PFPA) . 4. Modify ADC Operating Voltage in 20.2 ELECTRICAL CHARACTERISTIC . 5. Update 22. PACKAGE INFORMATION .
1.5	2023/05/08	1. Update 23.4 DATECODE SYSTEM . 2. Update description of maximum PCLK in 10.1 OVERVIEW
1.6	2023/07/17	1. Update 1.7 PIN CIRCUIT DIAGRAMS 2. Modify typing error in 3.3.1 Analog Block Control register (SYS0_ANBCTRL)
1.7	2023/11/01	1. Modify typing error in 6.4.2 PFPA for CT16B1 register (PFPA_CT16B1) . 2. Remove description of WDTCLKSEL register in 11 WATCHDOG TIMER (WDT) . 3. Update CS table in 17.8.1 CODE SECURITY (CS) . 4. Update SPI speed spec in 13.2 FEATURES .
1.8	2024/01/23	1. Add ADC Internal voltage reference, DAC output current and ESD result in 20.2 ELECTRICAL CHARACTERISTIC . 2. Update connector pin table in 21 FLASH ROM PROGRAMMING PIN . 3. Add new chapter 13.6 TIMING CHARACTERISTICS . 4. Modify SPI Data Fetch register in 13.7 SPI REGISTERS .
1.9	2024/06/14	1. Update ADC program flow in 7.5.3 ADC PROGRAM . 2. Remove description of CIS bit in 10.8.6 CT16Bn Count Control register . 3. Modify description in 13.6 TIMING CHARACTERISTICS .
2.0	2024/11/15	1. Update COMPARATOR Vcm in 20.2 ELECTRICAL CHARACTERISTIC . 2. Update description of P2.4 in 1.5 PIN ALLOCATION TABLE . 3. Modify typing error in 6.4.6 PFPA for CT16B2 register (PFPA_CT16B2) . 4. Modify typing error in 1.4 PIN ASSIGNMENT . 5. Modify description of CT16B in 10.2 FEATURES . 6. Modify typing error in 1.6 PIN DESCRIPTIONS .
2.1	2025/05/29	1. Modify notice in 4.3.2 DEEP-SLEEP MODE .
2.2	2025/06/19	1. Remove MER bit in 17.11.3 Flash Control register (FLASH_CTRL) .

Table of Content

AMENDMENT HISTORY	2
1 PRODUCT OVERVIEW	13
1.1 FEATURES	13
1.2 SYSTEM BLOCK DIAGRAM	17
1.3 CLOCK GENERATION BLOCK DIAGRAM	18
1.4 PIN ASSIGNMENT	19
1.5 PIN ALLOCATION TABLE	20
1.6 PIN DESCRIPTIONS	21
1.7 PIN CIRCUIT DIAGRAMS	26
1.8 PIN CHARACTERISTICS	29
2 CENTRAL PROCESSOR UNIT (CPU)	30
2.1 MEMORY MAP	30
2.2 SYSTEM TICK TIMER	31
2.2.1 OPERATION	31
2.2.2 SYSTICK USAGE HINTS AND TIPS	32
2.2.3 SYSTICK REGISTERS	32
2.2.3.1 System Tick Timer Control and Status register (SYSTICK_CTRL)	32
2.2.3.2 System Tick Timer Reload value register (SYSTICK_LOAD)	32
2.2.3.3 System Tick Timer Current Value register (SYSTICK_VAL)	33
2.2.3.4 System Tick Timer Calibration Value register (SYSTICK_CALIB)	33
2.3 NESTED VECTORED INTERRUPT CONTROLLER (NVIC)	34
2.3.1 INTERRUPT AND EXCEPTION VECTORS	34
2.3.2 NVIC REGISTERS	35
2.3.2.1 IRQ0~31 Interrupt Set-Enable Register (NVIC_ISER)	35
2.3.2.2 IRQ0~31 Interrupt Clear-Enable Register (NVIC_ICER)	35
2.3.2.3 IRQ0~31 Interrupt Set-Pending Register (NVIC_ISPR)	36
2.3.2.4 IRQ0~31 Interrupt Clear-Pending Register (NVIC_ICPR)	36
2.3.2.5 IRQ0~31 Interrupt Priority Register (NVIC_IPRn) (n=0~7)	36
2.4 APPLICATION INTERRUPT AND RESET CONTROL (AIRC)	37
2.5 CODE OPTION TABLE	38
2.6 UNIQUE NUMBER	38
2.7 CORE REGISTER OVERVIEW	39
3 SYSTEM CONTROL	40
3.1 RESET	40
3.1.1 POWER-ON RESET (POR)	40

3.1.2	WATCHDOG RESET (WDT RESET).....	41
3.1.3	BROWN-OUT RESET.....	41
3.1.3.1	BROWN OUT DESCRIPTION	41
3.1.3.2	THE SYSTEM OPERATING VOLTAGE DECSRIPTION	42
3.1.3.3	BROWN-OUT RESET IMPROVEMENT	42
3.1.4	EXTERNAL RESET	43
3.1.4.1	SIMPLY RC RESET CIRCUIT	44
3.1.4.2	DIODE & RC RESET CIRCUIT	44
3.1.4.3	ZENER DIODE RESET CIRCUIT.....	45
3.1.4.4	VOLTAGE BIAS RESET CIRCUIT	45
3.1.4.5	EXTERNAL RESET IC	46
3.1.5	SOFTWARE RESET	46
3.2	SYSTEM CLOCK	47
3.2.1	INTERNAL RC CLOCK SOURCE	47
3.2.1.1	Internal High-speed RC Oscillator (IHRC)	47
3.2.1.2	Internal Low-speed RC Oscillator (ILRC)	47
3.2.2	EXTERNAL CLOCK SOURCE	47
3.2.2.1	External High-speed (EHS) Clock	47
3.2.3	External Low-speed (ELS) Clock	48
3.2.3.1	Bypass Mode	49
3.2.4	PLL	50
3.2.4.1	PLL Frequency selection	50
3.2.4.2	PLL Recommended Frequency settings	51
3.2.5	SYSTEM CLOCK (SYSCLK) SELECTION.....	51
3.2.6	CLOCK-OUT CAPABILITY.....	51
3.3	SYSTEM CONTROL REGISTERS 0	52
3.3.1	Analog Block Control register (SYS0_ANBCTRL).....	52
3.3.2	PLL control register (SYS0_PLLCTRL).....	52
3.3.2.1	RECOMMEND FREQUENCY SETTING.....	53
3.3.3	Clock Source Status register (SYS0_CSST).....	53
3.3.4	System Clock Configuration register (SYS0_CLKCFG)	53
3.3.5	AHB Clock Prescale register (SYS0_AHBCP).....	54
3.3.6	System Reset Status register (SYS0_RSTST)	54
3.3.7	LVD Control register (SYS0_LVDCTRL).....	55
3.3.8	External RESET Pin Control register (SYS0_EXRSTCTRL)	55
3.3.9	SWD Pin Control register (SYS0_SWDCRTL).....	56
3.3.10	Interrupt Vector Table Mapping register (SYS0_IVTM).....	56
3.3.11	Anti-EFT Ability Control register (SYS0_ANTIEFT).....	56
3.3.12	IHRC Frequency Adjustment register (SYS0_IHRCADJ).....	56
3.4	SYSTEM CONTROL REGISTERS 1	58

3.4.1	AHB Clock Enable register (SYS1_AHBCLKEN)	58
3.4.2	APB Clock Prescale register 0 (SYS1_APB0CP0).....	59
3.4.3	APB Clock Prescale register 1 (SYS1_APB1CP1).....	59
3.4.4	Peripheral Reset register (SYS1_PRST).....	60
4	SYSTEM OPERATION MODE	62
4.1	OVERVIEW	62
4.2	NORMAL MODE	62
4.3	LOW-POWER MODES	62
4.3.1	SLEEP MODE	62
4.3.2	DEEP-SLEEP MODE.....	63
4.4	WAKEUP	64
4.4.1	OVERVIEW	64
4.4.2	WAKEUP TIME.....	64
4.5	STATE MACHINE OF PMU	65
4.6	OPERATION MODE COMPARSION TABLE	66
4.7	PMU REGISTERS	67
4.7.1	Power Control register (PMU_CTRL).....	67
5	GENERAL PURPOSE I/O PORT (GPIO)	68
5.1	OVERVIEW	68
5.2	GPIO MODE	68
5.3	GPIO REGISTERS.....	69
5.3.1	GPIO Port n Data register (GPIO _n _DATA) (n=0,1,2,3).....	69
5.3.2	GPIO Port n Mode register (GPIO _n _MODE) (n=0,1,2,3).....	69
5.3.3	GPIO Port n Configuration register (GPIO _n _CFG) (n=0,1,2,3).....	69
5.3.4	GPIO Port n Interrupt Sense register (GPIO _n _IS) (n=0,1,2,3).....	72
5.3.5	GPIO Port n Interrupt Both-edge Sense register (GPIO _n _IBS) (n=0,1,2,3).....	72
5.3.6	GPIO Port n Interrupt Event register (GPIO _n _IEV) (n=0,1,2,3).....	72
5.3.7	GPIO Port n Interrupt Enable register (GPIO _n _IE) (n=0,1,2,3).....	72
5.3.8	GPIO Port n Raw Interrupt Status register (GPIO _n _RIS) (n=0,1,2,3)	72
5.3.9	GPIO Port n Interrupt Clear register (GPIO _n _IC) (n=0,1,2,3).....	73
5.3.10	GPIO Port n Bits Set Operation register (GPIO _n _BSET) (n=0,1,2,3).....	73
5.3.11	GPIO Port n Bits Clear Operation register (GPIO _n _BCLR) (n=0,1,2,3).....	73
5.3.12	GPIO Port n Open-Drain Control register (GPIO _n _ODCTRL) (n=0,1,2,3)	73
6	PERIPHERAL FUNCTION PIN ASSIGNMENT (PFPA)	75
6.1	OVERVIEW	75
6.2	FEATURES	75
6.3	PIN ASSIGNMENT LIST	75
6.4	PFPA REGISTERS.....	76

6.4.1	PFPA for CT16B0 register (PFPA_CT16B0)	76
6.4.2	PFPA for CT16B1 register (PFPA_CT16B1)	76
6.4.3	PFPA for UART register (PFPA_UART)	76
6.4.4	PFPA for I2C register (PFPA_I2C)	77
6.4.5	PFPA for SPI register (PFPA_SPI)	77
6.4.6	PFPA for CT16B2 register (PFPA_CT16B2)	78
6.4.7	PFPA for CT16B5 register (PFPA_CT16B5)	78
7	12+2 CHANNEL ANALOG TO DIGITAL CONVERTOR (ADC)	80
7.1	OVERVIEW	80
7.2	FEATURES	81
7.3	ADC CONVERTING TIME	81
7.4	ADC OFFSET CALIBRATION	82
7.5	ADC CONTROL NOTICE	82
7.5.1	ADC SIGNAL	82
7.5.2	ADC OVERRUN	82
7.5.3	ADC PROGRAM	82
7.5.4	ADC PIN CONFIGURATION	83
7.6	ADC CONVERSION MODES	83
7.6.1	Single Mode	84
7.6.2	Continuous Mode	84
7.7	ADC WATCHDOG WINDOW	86
7.8	TEMPERATURE SENSOR	87
7.9	ADC CIRCUIT	88
7.10	ADC REGISTERS	89
7.10.1	ADC Management register (ADC_ADM)	89
7.10.2	ADC Data register (ADC_ADB)	90
7.10.3	ADC Interrupt Enable register (ADC_IE)	91
7.10.4	ADC Raw Interrupt Status register (ADC_RIS)	91
7.10.5	ADC Raw Interrupt Status register (ADC_IC)	92
7.10.6	ADC Convert Control register (ADC_CONVCTRL)	92
7.10.7	ADC Window Watchdog register (ADC_AWW)	93
7.10.8	ADC Window Watchdog Threshold register (ADC_AWWTH)	93
7.10.9	ADC Management register 1 (ADC_ADM1)	93
7.10.10	ADC Management register 2 (ADC_ADM2)	94
8	12-BIT DIGITAL TO ANALOG CONVERTOR (DAC)	95
8.1	OVERVIEW	95
8.2	FEATURES	95
8.3	TIMING DIAGRAM	96

8.4	DAC OUTPUT VOLTAGE.....	96
8.5	DAC REGISTERS.....	97
8.5.1	DAC Management register (DAC_ADM).....	97
8.5.2	DAC Internal Reference Voltage Source register (DAC_VIREF).....	97
8.5.3	DAC Clock Source register (DAC_CLK).....	98
8.5.4	DAC Trigger Control register (DAC_TGCTRL).....	99
8.5.5	DAC Data Control register 0 (DAC_DATACTRL0).....	100
8.5.6	DAC Data Control register 1 (DAC_DATACTRL1).....	100
8.5.7	DAC Interrupt Enable register (DAC_IE).....	100
8.5.8	DAC Raw Interrupt Status register (DAC_RIS).....	100
8.5.9	DAC Interrupt Clear register (DAC_IC).....	101
9	RAIL TO RAIL ANALOG COMPARATOR	102
9.1	OVERVIEW	102
9.2	NORMAL COMPARATOR MODE.....	104
9.2.1	COMPARATOR ENABLE	104
9.2.2	CMnOUT, CMnG AND CMnIF.....	104
9.2.3	COMPARATOR OUTPUT DEBOUNCE TIME CONTROL.....	105
9.3	COMPARATOR APPLICATION NOTICE	106
9.4	COMPARATOR REGISTERS	107
9.4.1	CMP Control register (CMP_CTRL).....	107
9.4.2	CMP Internal Reference Voltage Source register (CMP_VIREF).....	108
9.4.3	CMP Output Status register (CMP_OS).....	108
9.4.4	CMP Interrupt Enable register (CMP_IE).....	108
9.4.5	CMP Raw Interrupt Status register (CMP_RIS).....	109
9.4.6	CMP Interrupt Clear register (CMP_IC).....	109
9.4.7	CMP Output Debounce register (CMP_DB).....	109
10	16-BIT TIMER WITH CAPTURE FUNCTION	110
10.1	OVERVIEW	110
10.2	FEATURES	110
10.3	PIN DESCRIPTION	111
10.4	BLOCK DIAGRAM.....	111
10.5	TIMER OPERATION	112
10.5.1	Edge-aligned Up-counting Mode	112
10.5.2	Edge-aligned Down-counting Mode.....	113
10.5.3	Center-aligned Counting Mode	113
10.6	PWM.....	114
10.6.1	PWM Mode 1	114
10.6.2	PWM Mode 2	115

10.7	INVERSE PWM OUTPUT WITH DEAD-BAND PERIOD	116
10.8	CT16Bn REGISTERS	118
10.8.1	CT16Bn Timer Control register (CT16Bn_TMRCTRL) (n=0,1,2)	118
10.8.2	CT16Bn Timer Control register (CT16Bn_TMRCTRL) (n=5)	118
10.8.3	CT16Bn Timer Counter register (CT16Bn_TC) (n=0,1,2,5)	119
10.8.4	CT16Bn Prescale register (CT16Bn_PRE) (n=0,1,2,5).....	119
10.8.5	CT16Bn Prescale Counter register (CT16Bn_PC) (n=0,1,2,5).....	119
10.8.6	CT16Bn Count Control register (CT16Bn_CNTCTRL) (n=0,1,2,5).....	120
10.8.7	CT16Bn Match Control register (CT16Bn_MCTRL) (n=0,1,2,5)	120
10.8.8	CT16Bn Match register 0~3 (CT16Bn_MR0~3) (n=0,1,2,5).....	121
10.8.9	CT16Bn Capture Control register (CT16Bn_CAPCTRL) (n=0,1,2,5)	121
10.8.10	CT16Bn Capture 0 register (CT16Bn_CAP0) (n=0,1,2,5)	122
10.8.11	CT16Bn External Match register (CT16Bn_EM) (n=0,1,2,5)	122
10.8.12	CT16Bn PWM Control register (CT16Bn_PWMCTRL) (n=0,1,5).....	123
10.8.13	CT16Bn PWM Control register (CT16Bn_PWMCTRL) (n=2).....	124
10.8.14	CT16Bn Timer Raw Interrupt Status register (CT16Bn_RIS) (n=0,1,2,5)	126
10.8.15	CT16Bn Timer Interrupt Clear register (CT16Bn_IC) (n=0,1,2,5).....	126
10.8.16	CT16Bn Timer Match register 9 (CT16Bn_MR9) (n=0,1,2,5).....	126
10.8.17	CT16Bn PWMmN Dead-band Period register (CT16Bn_PWMmNDB) (n=2).....	127
11	WATCHDOG TIMER (WDT).....	128
11.1	OVERVIEW	128
11.2	BLOCK DIAGRAM.....	129
11.3	WDT REGISTERS	130
11.3.1	Watchdog Configuration register (WDT_CFG).....	130
11.3.2	Watchdog Timer Constant register (WDT_TC).....	130
11.3.3	Watchdog Feed register (WDT_FEED)	131
12	REAL-TIME CLOCK (RTC)	132
12.1	OVERVIEW	132
12.2	FEATURES	132
12.3	FUNCTIONAL DESCRIPTION	132
12.3.1	INTRODUCTION	132
12.3.2	RTC FLAG ASSERTION	132
12.3.3	RTC OPERATION	132
12.4	BLOCK DIAGRAM.....	133
12.5	RTC REGISTERS	134
12.5.1	RTC Control register (RTC_CTRL)	134
12.5.2	RTC Clock Source Select register (RTC_CLKS).....	134
12.5.3	RTC Interrupt Enable register (RTC_IE).....	134

12.5.4	RTC Raw Interrupt Status register (RTC_RIS)	134
12.5.5	RTC Interrupt Clear register (RTC_IC).....	135
12.5.6	RTC Second Counter Reload Value register (RTC_SECCNTV).....	135
12.5.7	RTC Second Count register (RTC_SECCNT)	135
13	SPI.....	136
13.1	OVERVIEW	136
13.2	FEATURES	136
13.3	PIN DESCRIPTION	136
13.4	INTERFACE DESCRIPTION	137
13.4.1	SPI	137
13.4.2	COMMUNICATION FLOW.....	138
13.4.2.1	SINGLE-FRAME	138
13.4.2.2	MULTI-FRAME	138
13.5	AUTO-SEL.....	138
13.6	TIMING CHARACTERISTICS.....	139
13.6.1	MASTER MODE.....	139
13.6.2	SLAVE MODE	139
13.7	SPI REGISTERS	140
13.7.1	SPI n Control register 0 (SPIn_CTRL0) (n=0)	140
13.7.2	SPI n Control register 1 (SPIn_CTRL1) (n=0)	141
13.7.3	SPI n Clock Divider register (SPIn_CLKDIV) (n=0)	141
13.7.4	SPI n Status register (SPIn_STAT) (n=0).....	141
13.7.5	SPI n Interrupt Enable register (SPIn_IE) (n=0).....	142
13.7.6	SPI n Raw Interrupt Status register (SPIn_RIS) (n=0)	142
13.7.7	SPI n Interrupt Clear register (SPIn_IC) (n=0).....	143
13.7.8	SPI n Data register (SPIn_DATA) (n=0)	143
13.7.9	SPI n Data Fetch Delay register (SPIn_DFDLY) (n=0).....	143
14	I2C.....	144
14.1	OVERVIEW	144
14.2	FEATURES	144
14.3	PIN DESCRIPTION	144
14.4	I2C PROPOCOL.....	145
14.4.1	7-BIT ADDRESSING MODES	145
14.4.1.1	MASTER TRANSMITTER MODE	145
14.4.1.2	MASTER RECEIVER MODE.....	146
14.4.1.3	SLAVE TRANSMITTER MODE.....	146
14.4.1.4	SLAVE RECEIVER MODE	146
14.4.2	10-BIT ADDRESSING MODES	146

1.1.1.1	MASTER TRANSMITTER MODE	146
1.1.1.2	MASTER RECEIVER MODE.....	146
14.5	ARBITRATION	147
14.6	CLOCK STRETCHING	147
14.7	GENERAL CALL ADDRESS	147
14.8	TIMING CHARACTERISTICS.....	148
14.8.1	MASTER TRANSMITTER MODE	148
14.8.2	SLAVE TRANSMITTER MODE	148
14.9	I2C REGISTERS	149
14.9.1	I2C n Control register (I2Cn_CTRL) (n=0,1).....	149
14.9.2	I2C n Status register (I2Cn_STAT) (n=0,1).....	150
14.9.3	I2C n TX Data register (I2Cn_TXDATA) (n=0,1)	151
14.9.4	I2C n RX Data register (I2Cn_RXDATA) (n=0,1).....	151
14.9.5	I2C n Slave Address 0 register (I2Cn_SLVADDR0) (n=0,1).....	151
14.9.6	I2C n Slave Address 1~3 register (I2Cn_SLVADDR1~3) (n=0,1)	151
14.9.7	I2C n SCL High Time register (I2Cn_SCLHT) (n=0,1).....	152
14.9.8	I2C n SCL Low Time register (I2Cn_SCLLT) (n=0,1).....	152
14.9.9	I2C n Timeout Control register (I2Cn_TOCTRL) (n=0,1)	152
15	UNIVERSAL ASYNCHRONOUS RECEIVER AND TRANSMITTER (UART)	153
15.1	OVERVIEW	153
15.2	FEATURES	153
15.3	PIN DESCRIPTION	153
15.4	BLOCK DIAGRAM.....	154
15.5	BAUD RATE CALCULATION	155
15.6	UART REGISTERS	157
15.6.1	UART n Receiver Buffer register (UARTn_RB) (n=0,1).....	157
15.6.2	UART n Transmitter Holding register (UARTn_TH) (n=0,1).....	157
15.6.3	UART n Divisor Latch LSB registers (UARTn_DLL) (n =0,1)	157
15.6.4	UART n Divisor Latch MSB register (UARTn_DLM) (n=0,1)	158
15.6.5	UART n Interrupt Enable register (UARTn_IE) (n=0,1)	158
15.6.6	UART n Interrupt Identification register (UARTn_II) (n=0,1)	158
15.6.7	UART n FIFO Control register (UARTn_FIFOCTRL) (n=0,1).....	159
15.6.8	UART n Line Control register (UARTn_LC) (n=0,1)	159
15.6.9	UART n Line Status register (UARTn_LS) (n=0,1).....	160
15.6.10	UART n Scratch Pad register (UARTn_SP) (n=0,1).....	161
15.6.11	UART n Fractional Divider register (UARTn_FD) (n=0,1)	161
15.6.12	UART n Control register (UARTn_CTRL) (n=0,1).....	162
15.6.13	UART n Half-duplex Enable register (UARTn_HDEN) (n=0,1)	162
16	CYCLIC REDUNDANCY CHECK (CRC).....	164

16.1	OVERVIEW	164
16.2	FEATURES	164
16.3	CRC REGISTERS	165
16.3.1	CRC Control register (CRC_CTRL).....	165
16.3.2	CRC Data register (CRC_DATA).....	165
17	FLASH.....	166
17.1	OVERVIEW	166
17.2	EMBEDDED FLASH MEMORY	166
17.3	FEATURES	166
17.4	ORGANIZATION.....	167
17.5	READ	167
17.6	PROGRAM/ERASE.....	167
17.7	EMBEDDED BOOT LOADER	167
17.8	FLASH MEMORY CONTROLLER (FMC).....	168
17.8.1	CODE SECURITY (CS).....	168
17.8.2	PROGRAM FLASH MEMORY.....	169
17.8.3	ERASE	169
17.8.3.1	PAGE ERASE	169
17.8.3.2	MASS ERASE.....	169
17.9	READ PROTECTION.....	169
17.10	HW CHECKSUM.....	169
17.11	FMC REGISTERS.....	170
17.11.1	Flash Low Power Control register (FLASH_LPCTRL).....	170
17.11.2	Flash Status register (FLASH_STATUS)	170
17.11.3	Flash Control register (FLASH_CTRL).....	170
17.11.4	Flash Data register (FLASH_DATA).....	171
17.11.5	Flash Address register (FLASH_ADDR)	171
17.11.6	Flash Checksum register (FLASH_CHKSUM)	171
17.11.7	Flash Write Protect register (FLASH_WP).....	171
18	SERIAL-WIRE DEBUG (SWD).....	172
18.1	OVERVIEW	172
18.2	FEATURES	172
18.3	PIN DESCRIPTION	172
18.4	DEBUG NOTE.....	172
18.4.1	LIMITATIONS	172
18.4.2	DEBUG RECOVERY.....	172
18.4.3	INTERNAL PULL-UP/DOWN RESISTORS on SWD PINS.....	172
19	DEVELOPMENT TOOL	174

19.1	SN-LINK-V3	175
19.2	SN32F800 STARTER-KIT	176
20	ELECTRICAL CHARACTERISTIC	177
20.1	ABSOLUTE MAXIMUM RATING	177
20.2	ELECTRICAL CHARACTERISTIC	177
20.3	CHARACTERISTIC GRAPHS	180
21	FLASH ROM PROGRAMMING PIN	182
22	PACKAGE INFORMATION	183
22.1	QFN 32 PIN 4x4	183
22.2	LQFP 32 PIN	184
23	MARKING DEFINITION	185
23.1	MARKING INDETIFICATION SYSTEM	185
23.2	INTRODUCTION	186
23.3	MARKING EXAMPLE	186
23.4	DATECODE SYSTEM	187

1 PRODUCT OVERVIEW

1.1 FEATURES

- ◆ **Memory configuration**
32KB on-chip Flash programming memory.
8KB SRAM.
4KB Boot ROM
- ◆ **Operation Frequency up to 60MHz**
- ◆ **Interrupt sources**
ARM® Cortex®-M0 built-in Nested Vectored Interrupt Controller (NVIC).
- ◆ **I/O pin configuration**
Up to 30 General Purpose I/O (GPIO) pins with configurable pull-up/pull-down resistors.
GPIO pins can be used as edge and level sensitive interrupt sources.
All IO with typical 20mA driving/sinking current.
- ◆ **Programmable Watchdog Timer (WDT)**
Programmable watchdog frequency with watchdog clock source and divider.
- ◆ **System tick timer**
The 24-bit SysTick timer clock source is fixed to the system clock, and is intended to generate a fixed 10-ms interrupt.
- ◆ **Real-Time Clock (RTC)**
- ◆ **LVD with separate thresholds**
Reset: 1.7/2.6/2.7/2.8/2.9/3.0V for VDD
Interrupt: 2.6/2.7/2.8/2.9/3.0V for VDD
- ◆ **Fcpu (Instruction cycle)**
 $F_{CPU} = F_{HCLK} = F_{SYSCLK}/1, F_{SYSCLK}/2, F_{SYSCLK}/4, \dots, F_{SYSCLK}/128.$
- ◆ **Operating modes**
Normal
Sleep
Deep-sleep
- ◆ **Cyclic Redundancy Check (CRC)**
CRC-16
CRC-16-CCITT
CRC-32
- ◆ **PFPA**
- ◆ **Serial Wire Debug (SWD)**
- ◆ **In-Circuit Programming (ICP) supported**
- ◆ **Timer**
3 16-bit timer support up-counting, down-counting, and center-aligned mode.
1 16-bit timers support up-counting mode.
16 sets PWM
4 sets inverse PWM with programmable dead-band
PCLK up to 120MHz
- ◆ **Working voltage 1.8V ~ 5.5V**
- ◆ **12+2 CH 12-bit SAR ADC**
12 external ADC input
1 internal battery measurement
5-level internal reference voltage source (VDD, 4.5V, 3V, 2.5V, 2V)
16-frame ADB FIFO
Single/Continuous mode
Single/Multi-channel auto conversion
- ◆ **12-Bit DACs**
4 12-bit DACs with 4 independent output channels
Internal reference voltage source
- ◆ **2 Rail to Rail Comparators**
Internal reference voltage source 3V/2V/1.5V
Programmable 16-level internal ref. voltage divider
3 external negative inputs
3 external positive inputs
- ◆ **Interface**
-Two I2C controllers supporting I2C-bus specification with multiple address recognition.
-Two UART controllers with fractional baud rate generation.
-One SPI controller.
- ◆ **System clocks**
-External high clock: Crystal type 4MHz~25MHz
-External low clock: Crystal type 32.768 KHz
-Internal high clock: RC type 48MHz
-Internal low clock: RC type 32 KHz
-PLL allows CPU operation up to the maximum CPU rate without the need for a high-frequency crystal.
-Clock output function which can reflect the internal high/low RC oscillator, HCLK, PLL output, and external low clock.
- ◆ **Package (Chip form support)**
QFN 32
LQFP32

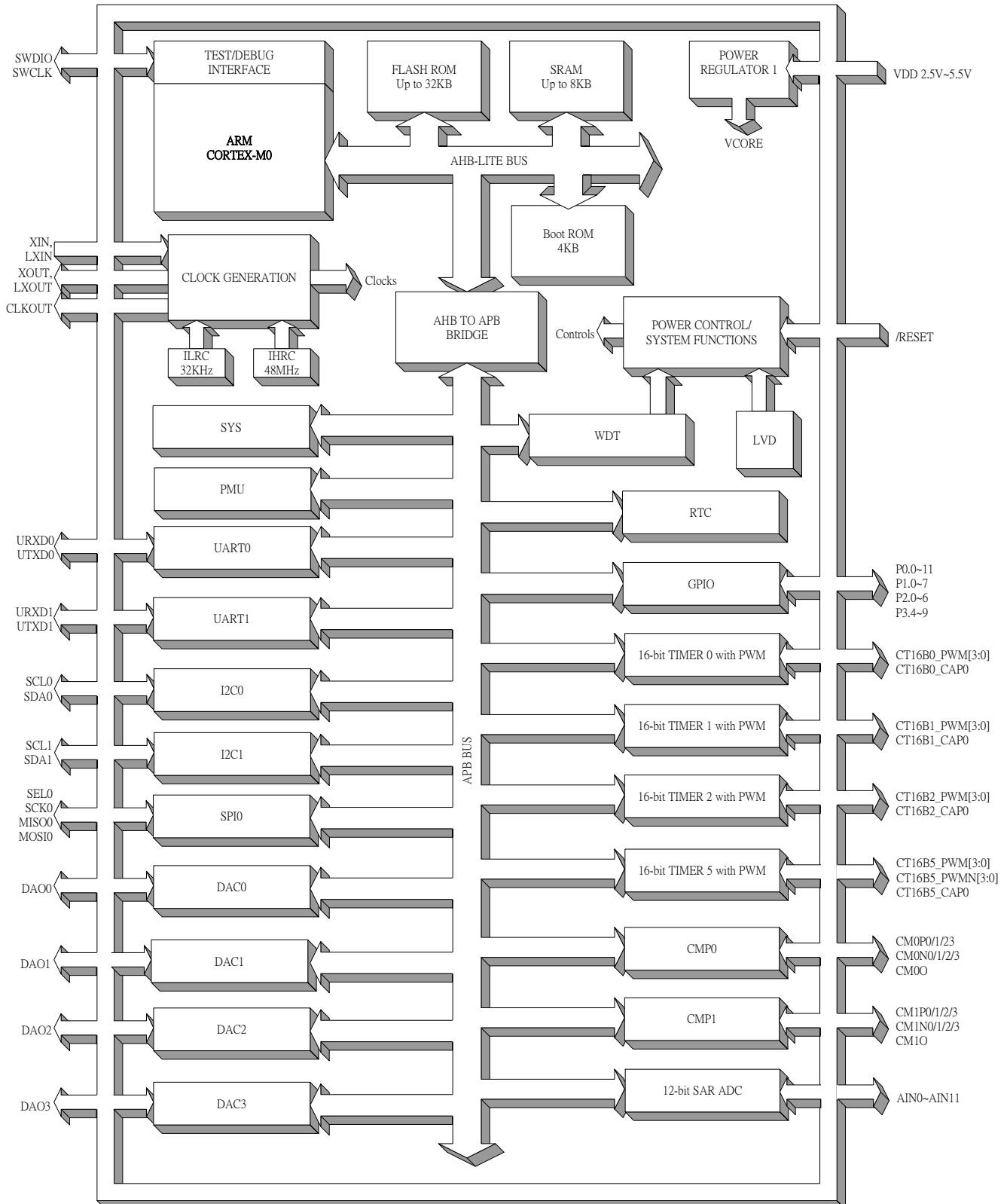
Features Selection Table

Chip	ROM (KB)	RAM (KB)	Fcpu. (Max MHz)	TIMER	UART	I2C	SPI	PWM	12-bit ADC	12-bit DAC	CMP	GPIO with Wakeup	PKG
SN32F805J/F	32	8	60	24-bit x 1 16-bit x 4	2	2	1	16+4	12+2	4	2	30	QFN32/ LQFP32

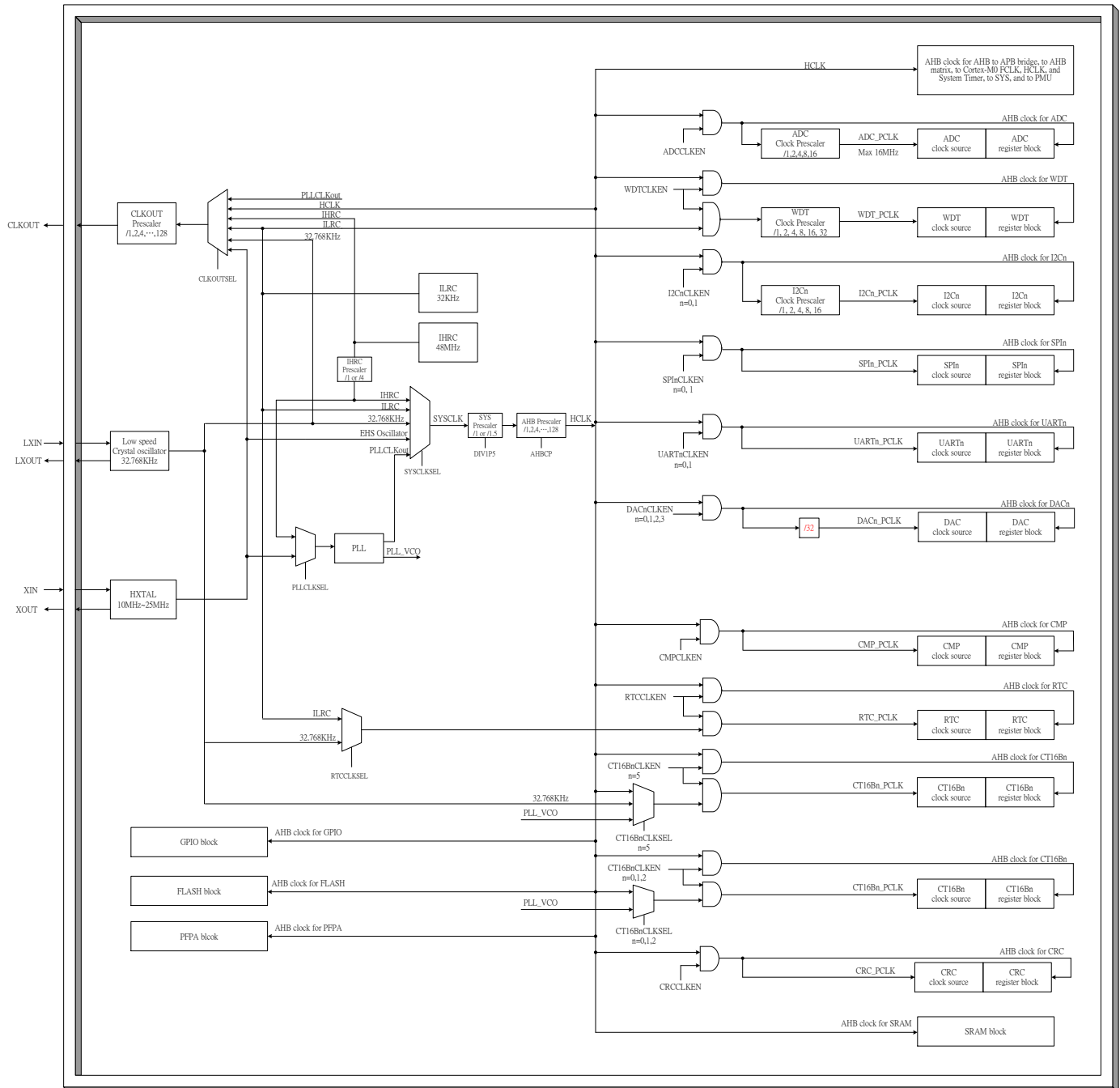
	SN32F700	SN32F700B	SN32F800
Cortex	M0	M0	M0
Max Pin out	48	48	32
Max Freq	50MHz	48MHz	60MHz
Working Volts	1.8~3.6V	1.8~5.5V	1.8~5.5V
User ROM	32KB	32KB	32KB
SRAM	8KB	8KB	8KB
Boot ROM	4KB	1.5KB	4KB
I/O	43	44	30
16-bit Timer	2	3 (support LXTAL clock source)	4 (support LXTAL clock source)
32-bit Timer	2	0	0
PWM	13	9+4	16+4
UART	2 (16-byte receive and transmit FIFOs)	2 (No FIFOs, support UART only)	2 (No FIFOs, support UART only)
SPI	2	1	1
I2C	2	1 (Remove Monitor mode)	2 (Remove Monitor mode)
I2S	1	X	X
12-bit ADC	10CH Max 250KHz sampling rate	10+1 CH Max 250KHz sampling rate 1. With 4-level internal reference voltage source (VDD, 4.5V, 3V, 2V)	12+2 CH Max 1MHz sampling rate 1. With Temp sensor inside, 5-level internal reference voltage source (VDD, 4.5V, 3V, 2.5V, 2V) 2. 16-Frame FIFO 3. Support Continuous mode and Multi-channel conversion
12-bit DAC	X	X	4
CMP	X	2	2
RTC	V (Support Second/Alarm/Overflow interrupt)	V (Support Second interrupt only)	V (Support Second interrupt only)
IHRC	12MHz	12MHz	48MHz (HCLK=IHRC/4 by default)
ILRC	16KHz	16KHz	32KHz
EHS Xtal	10MHz~25MHz	10MHz~16MHz	4MHz~25MHz
PLL	$F_{VCO} = F_{CLKIN} / F * M = 156\sim 320\text{MHz}$ $F_{CLKOUT} = F_{VCO} / P$ $2 < M \leq 31$ $P = 6, 8, 10, 12, 14$	$F_{VCO} = F_{CLKIN} * M = 72\sim 120\text{MHz}$ $F_{CLKOUT} = F_{VCO} / P$ $M = 4, 6, 8, 10, 12$ $P = 2 \text{ or } 4$	$F_{VCO} = F_{CLKIN} / F * M = 96\sim 120\text{MHz}$ $F_{CLKOUT} = F_{VCO} / P$ $2 \leq M \leq 31$ $P = 2 \text{ or } 4$
WDT	V	V	V
FLASH ROM	Erase Page (512 bytes) 25ms	No need to Erase page Program 1 page (64 bytes) 5ms	Erase Page (512 bytes) 5ms
Clock tree	$F_{cpu} = F_{SYSCLK}/1, F_{SYSCLK}/2, \dots, F_{SYSCLK}/512$	1. $F_{cpu} = F_{SYSCLK}/1, F_{SYSCLK}/2, \dots, F_{SYSCLK}/128$ 2. Remove SYS1_PRST register 3. Remove SYS1_CP0 & CP1 registers	$F_{cpu} = F_{SYSCLK}/1, F_{SYSCLK}/2, \dots, F_{SYSCLK}/128$
PMU	1. Only P0.0~P0.11 can wakeup MCU in Deep sleep mode (enter WAKEUP_ISR)	1. All GPIO can wakeup MCU in Deep sleep mode (enter P0/1/2/3_ISR)	1. All GPIO can wakeup MCU in Deep sleep mode (enter P0/1/2/3_ISR)

	2. Only GPIOCLKEN bit to control HCLK for all GPIO 3. 16-byte Backup registers	2. Remove 16-byte Backup registers. 3. Remove Deep Power-down mode.	2. Remove 16-byte Backup registers. 3. Remove Deep Power-down mode.
--	---	--	--

1.2 SYSTEM BLOCK DIAGRAM

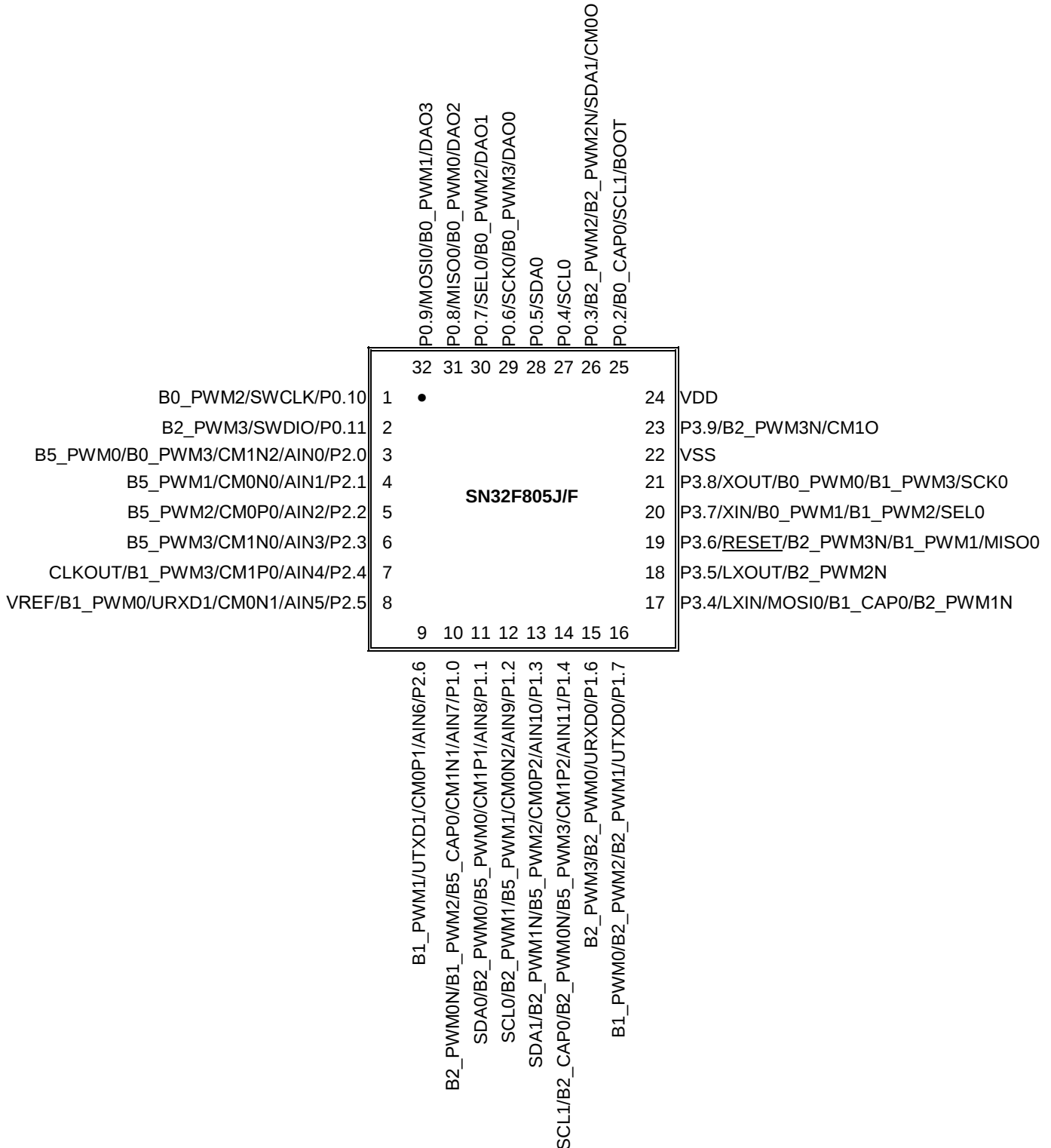


1.3 CLOCK GENERATION BLOCK DIAGRAM



1.4 PIN ASSIGNMENT

SN32F805J/F (QFN 32 pins 4x4/LQFP 32 pins)



- * **Note:**
1. The pins which are not pin-out shall be set correctly to decrease power consumption in low-power modes. Strongly recommended to set these pins as input pull-up.
 2. SONiX provide Boot loader to check the status of P0.2 (BOOT pin) during boot procedure. If BOOT pin is Low during Boot procedure, MCU will execute code in Boot loader instead of User code. We strongly recommended NOT using BOOT pin as output pin to drive the LED, otherwise, the BOOT pin status may be low during boot procedure.

1.5 PIN ALLOCATION TABLE

Port	UART	SPI	I2C	CT16	ADC	CMP	Other
P0.2			SCL1	B0_CAP0			BOOT
P0.3			SDA1	B2_PWM2 B2_PWM2N		CM00	
P0.4			SCL0				
P0.5			SDA0				
P0.6		SCK0		B0_PWM3			DAO0
P0.7		SEL0		B0_PWM2			DAO1
P0.8		MISO0		B0_PWM0			DAO2
P0.9		MOSI0		B0_PWM1			DAO3
P0.10							SWCLK
P0.11				B2_PWM3			SWDIO
P1.0				B2_PWM0N B1_PWM2 B5_CAP0	AIN7	CM1N1	
P1.1			SDA0	B2_PWM0 B5_PWM0	AIN8	CM1P1	
P1.2			SCL0	B2_PWM1/ B5_PWM1	AIN9	CM0N2	
P1.3			SDA1	B2_PWM1N B5_PWM2	AIN10	CM0P2	
P1.4			SCL1	B2_PWM0N B5_PWM3 B2_CAP0	AIN11	CM1P2	
P1.6	URXD0			B2_PWM3 B2_PWM0			
P1.7	UTXD0			B1_PWM0 B2_PWM2 B2_PWM1			
P2.0				B5_PWM0 B0_PWM3	AIN0	CM1N2	
P2.1				B5_PWM1	AIN1	CM0N0	
P2.2				B5_PWM2	AIN2	CM0P0	
P2.3				B5_PWM3	AIN3	CM1N0	
P2.4				B1_PWM3	AIN4	CM1P0	CLKOUT
P2.5	URXD1			B1_PWM0	AIN5	CM0N1	VREF
P2.6	UTXD1			B1_PWM1	AIN6	CM0P1	
P3.4		MOSI0		B2_PWM1N B1_CAP0			LXIN
P3.5				B2_PWM2N			LXOUT
P3.6		MISO0		B2_PWM3N B1_PWM1			<u>RESET</u>

P3.7		SEL0		B0_PWM1 B1_PWM2			XIN
P3.8		SCK0		B0_PWM0 B1_PWM3			XOUT
P3.9				B2_PWM3N		CM10	

1.6 PIN DESCRIPTIONS

PIN NAME	TYPE	DESCRIPTION
VDD, VSS	P	Power supply input pins for digital circuit.
P0.2/CT16B0_CAP0/ SCL1/BOOT	I/O	P0.2 — General purpose digital input/output pin.
		SCL1 — I2C1 clock pin.
	I	CT16B0_CAP0 — CT16B0 Capture input 0.
		BOOT — Internal pull-up in Boot loader, tie LOW to keep in Boot loader or left HIGH to exit Boot loader and execute User program at boot time.
P0.3/SDA1/CM00/ CT16B2_PWM2/ CTB2_PWM2N	I/O	P0.3 — General purpose digital input/output pin.
		SDA1 — I2C1 data pin.
	O	CT16B2_PWM2 — CT16B2 PWM output 2.
		CT16B2_PWM2N — CT16B2 Negative PWM output 2.
		CM00 — Comparator 0 output pin.
P0.4/SCL0	I/O	P0.4 — General purpose digital input/output pin.
		SCL0 — I2C0 clock pin.
P0.5/SDA0	I/O	P0.5 — General purpose digital input/output pin.
		SDA0 — I2C0 data pin
P0.6/SCK0/DAO0/ CT16B0_PWM3	I/O	P0.6 — General purpose digital input/output pin.
		SCK0 — SPI0 Serial clock pin.
	O	CT16B0_PWM3 — CT16B0 PWM output 3.
		DAO0 — Output pin of DAC0.
P0.7/SEL0/DAO1/ CT16B0_PWM2	I/O	P0.7 — General purpose digital input/output pin.
		SEL0 — SPI0 Select pin.
	O	CT16B0_PWM2 — CT16B0 PWM output 2.
		DAO1 — Output pin of DAC1.

P0.8/MISO0/DAO2/ CT16B0_PWM0	I/O	P0.8 — General purpose digital input/output pin.
		MISO0 — SPI0 Master In Slave Out pin.
	O	CT16B0_PWM0 — CT16B0 PWM output 0.
		DAO2 — Output pin of DAC2.
P0.9/MOSI0/DAO3/ CT16B0_PWM1	I/O	P0.9 — General purpose digital input/output pin.
		MOSI0 — SPI0 Master Out Slave In pin.
	O	CT16B0_PWM1 — CT16B0 PWM output 1.
		DAO3 — Output pin of DAC3.
P0.10/SWCLK/ CT16B0_PWM2	I/O	P0.10 — General purpose digital input/output pin.
	I	SWCLK — Serial Wire Clock pin.
	O	CT16B0_PWM2 — CT16B0 PWM output 2.
P0.11/SWDIO/ CT16B2_PWM3	I/O	P0.11 — General purpose digital input/output pin.
		SWDIO — Serial Wire Debug input/output pin.
	O	CT16B2_PWM3 — CT16B2 PWM output 3.
P1.0/AIN7/CM1N1/ CT16B2_PWM0N/ CT16B1_PWM2/ CT16B5_CAP0	I/O	P1.0 — General purpose digital input/output pin.
	I	AIN7 — ADC channel input pin 7.
		CM1N1 — Comparator 1 Negative input pin 1.
		CT16B5_CAP0 — CT16B5 Capture input 0.
	O	CT16B2_PWM0N — CT16B2 Negative PWM output 0.
		CT16B1_PWM2 — CT16B1 PWM output 2.
P1.1/AIN8/CM1P1/ SDA0/ CT16B2_PWM0/ CT16B5_PWM0	I/O	P1.1 — General purpose digital input/output pin.
		SDA0 — I2C0 data pin.
	I	AIN8 — ADC channel input pin 8.
		CM1P1 — Comparator 1 Positive input pin 1.
	O	CT16B2_PWM0 — CT16B2 PWM output 0.
		CT16B5_PWM0 — CT16B5 PWM output 0.
P1.2/AIN9/CM0N2 SCL0/ CT16B2_PWM1/	I/O	P1.2 — General purpose digital input/output pin.
		SCL0 — I2C0 clock pin.
	I	AIN9 — ADC channel input pin 9.

CT16B5_PWM1		CM0N2 — Comparator 0 Negative input pin 2.
	O	CT16B2_PWM1 — CT16B2 PWM output 1.
		CT16B5_PWM1 — CT16B5 PWM output 1.
P1.3/AIN10/CM0P2/ SDA1/ CT16B0_PWM1N/ CT16B5_PWM2	I/O	P1.3 — General purpose digital input/output pin.
		SDA1 — I2C1 data pin.
	I	AIN10 — ADC channel input pin 10.
		CM0P2 — Comparator 0 Positive input pin 2.
	O	CT16B2_PWM1N — CT16B2 Negative PWM output 1.
		CT16B5_PWM2 — CT16B5 PWM output 2.
P1.4/AIN11/CM1P2/ SCL1/ CT16B2_PWM0N/ CT16B5_PWM3/ CT16B2_CAP0	I/O	P1.4 — General purpose digital input/output pin.
		SCL1 — I2C1clock pin.
	I	AIN11 — ADC channel input pin 11.
		CM1P2 — Comparator 1 Positive input pin 2.
		CT16B2_CAP0 — CT16B2 Capture input 0.
	O	CT16B2_PWM0N — CT16B2 Negative PWM output 0.
		CT16B5_PWM3 — CT16B5 PWM output 3.
P1.6/URXD0/ CT16B2_PWM0/ CT16B2_PWM3	I/O	P1.6 — General purpose digital input/output pin.
	I	URXD0 — UART0 Receiver input pin.
	O	CT16B2_PWM0 — CT16B2 PWM output 0.
		CT16B2_PWM3 — CT16B2 PWM output 3.
P1.7/UTXD0/ CT16B1_PWM0/ CT16B2_PWM1/ CT16B2_PWM2	I/O	P1.7 — General purpose digital input/output pin.
	O	UTXD0 — UART0 Transmitter output pin.
		CT16B1_PWM0 — CT16B1 PWM output 0.
		CT16B2_PWM1 — CT16B2 PWM output 1.
		CT16B2_PWM2 — CT16B2 PWM output 2.
P2.0/AIN0/CM1N2/ CT16B5_PWM0/ CT16B0_PWM3	I/O	P2.0 — General purpose digital input/output pin.
	I	AIN0 — ADC channel input pin 0.
		CM1N2 — Comparator 1 Negative input pin 2.
	O	CT16B0_PWM3 — CT16B0 PWM output 3.

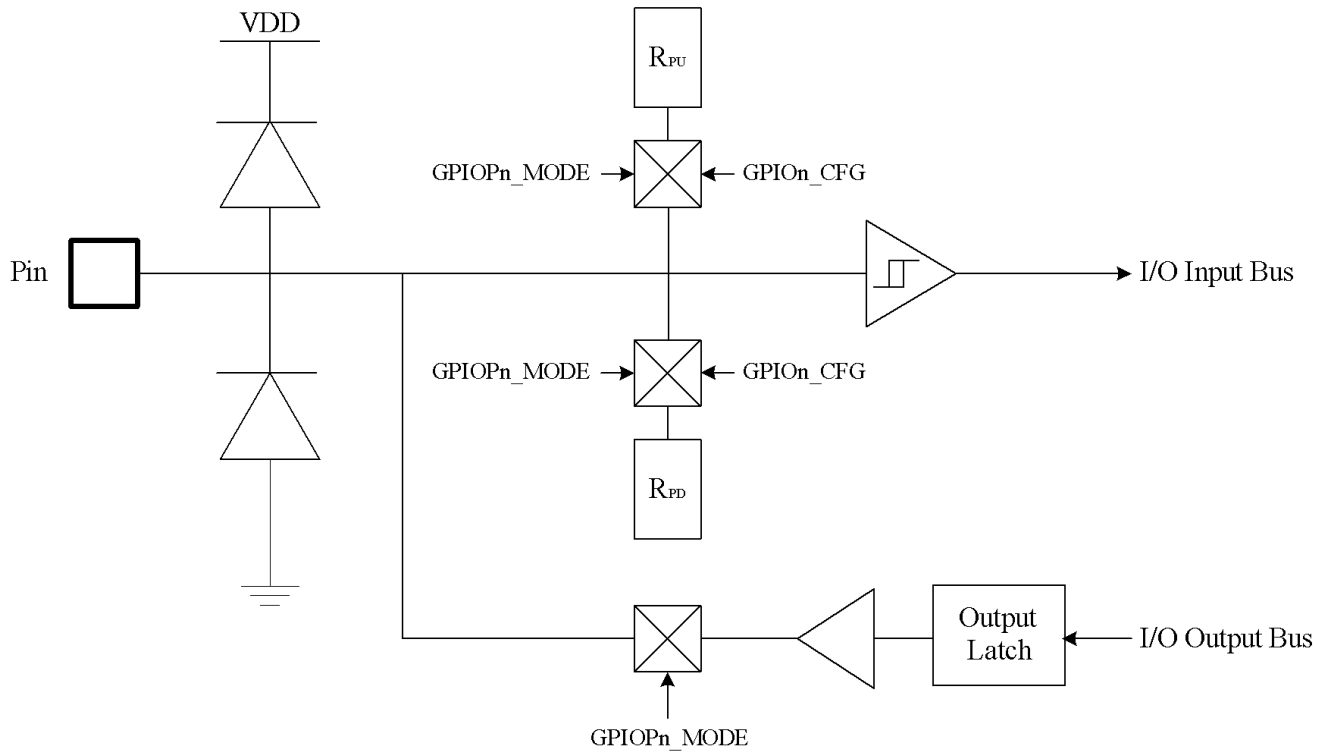
		CT16B5_PWM0 — CT16B5 PWM output 0.
P2.1/AIN1/CM0N0/ CT16B5_PWM1	I/O	P2.1 — General purpose digital input/output pin.
	I	AIN1 — ADC channel input pin 1.
		CM0N0 — Comparator 0 Negative input pin 0.
	O	CT16B5_PWM1 — CT16B5 PWM output 1.
P2.2/AIN2/CM0P0/ CT16B5_PWM2	I/O	P2.2 — General purpose digital input/output pin.
	I	AIN2 — ADC channel input pin 2.
		CM0P0 — Comparator 0 Positive input pin 0.
	O	CT16B5_PWM2 — CT16B5 PWM output 2.
P2.3/AIN3/CM1N0/ CT16B5_PWM3	I/O	P2.3 — General purpose digital input/output pin.
	I	AIN3 — ADC channel input pin 3.
		CM1N0 — Comparator 1 Negative input pin 0.
	O	CT16B5_PWM3 — CT16B5 PWM output 3.
P2.4/AIN4/CM1P0/ CLKOUT/ CT16B1_PWM3	I/O	P2.4 — General purpose digital input/output pin.
	I	AIN4 — ADC channel input pin 4.
		CM1P0 — Comparator 1 Positive input pin 0.
	O	CT16B1_PWM3 — CT16B5 PWM output 3.
P2.5/AIN5/CM0N1/ URXD1/VREF/ CT16B1_PWM0	I/O	P2.5 — General purpose digital input/output pin.
	I	AIN5 — ADC channel input pin 5.
		URXD1 — UART1 Receiver input.
		CM0N1 — Comparator 0 Negative input pin 1.
	O	VREF — VREF pin for ADC/DAC/CMP
		CT16B1_PWM0 — CT16B1 PWM output 0.
P2.6/AIN6/CM0P1/ UTXD1/ CT16B1_PWM1	I/O	P2.6 — General purpose digital input/output pin.
	I	AIN6 — ADC channel input pin 6.
		CM0P1 — Comparator 0 Positive input pin 1.
	O	UTXD1 — UART1 Transmitter input.
		CT16B1_PWM1 — CT16B1 PWM output 1.
P3.4/LXIN/MOSIO/	I/O	P3.4 — General purpose digital input/output pin.

CT16B2_PWM1N/ CT16B1_CAP0	I	LXIN — External low-speed X'tal input pin. CT16B1_CAP0 — CT16B1 Capture input 0.
	I/O	MOSI0 — SPI0 Master Out Slave In pin.
	O	CT16B2_PWM1N — CT16B2 Negative PWM output 1.
P3.5/LXOUT/ CT16B2_PWM2N	I/O	P3.5 — General purpose digital input/output pin.
	O	LXOUT — External low-speed X'tal output pin. CT16B2_PWM2N — CT16B2 Negative PWM output 2.
P3.6/ <u>RESET</u> /MISO0/ CT16B1_PWM1/ CT16B2_PWM3N	I/O	P3.6 — General purpose digital input/output pin. MISO0 — SPI0 Master In Slave Out pin.
	I	RESET — External Reset input. Schmitt trigger structure, active “Low”, normally stay “High”.
	O	CT16B1_PWM1 — CT16B1 PWM output 1. CT16B2_PWM3N — CT16B2 Negative PWM output 3.
P3.7/XIN/SEL0/ CT16B0_PWM1/ CT16B1_PWM2	I/O	P3.7 — General purpose digital input/output pin. SEL0 — SPI0 Select pin.
	I	XIN — External high-speed X'tal input pin.
	O	CT16B0_PWM1 — CT16B0 PWM output 1. CT16B1_PWM2 — CT16B1 PWM output 2.
P3.8/XOUT/SCK0/ CT16B0_PWM0/ CT16B1_PWM3	I/O	P3.8 — General purpose digital input/output pin. SCK0 — SPI0 Serial clock pin.
	O	XOUT — External high-speed X'tal output pin. CT16B0_PWM0 — CT16B0 PWM output 0. CT16B1_PWM3 — CT16B1 PWM output 3.
P3.9/CM1O/ CT16B2_PWM3N	I/O	P3.9 — General purpose digital input/output pin.
	O	CM1O — Comparator 1 output pin. CT16B2_PWM3N — CT16B2 Negative PWM output 3.

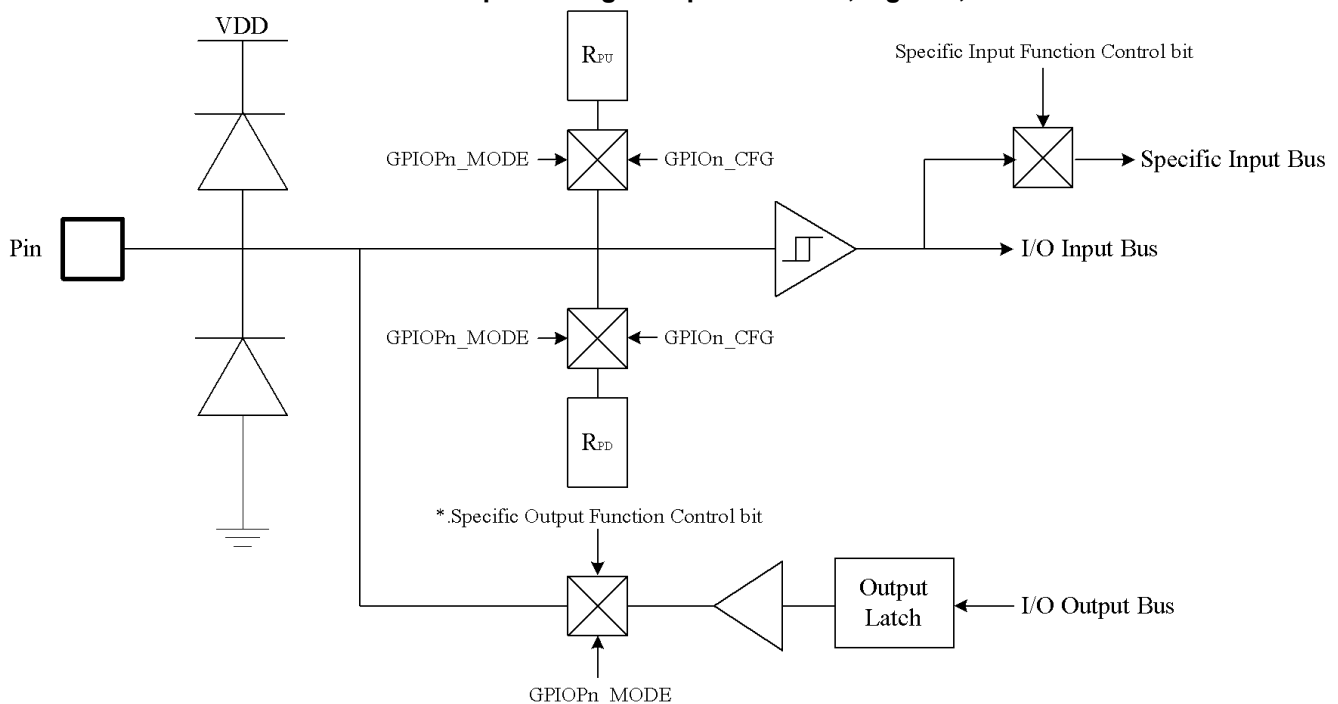
Please refer to [Peripheral Function Pin Assignment \(PFPA\)](#) chapters for setting of each GPIOs.

1.7 PIN CIRCUIT DIAGRAMS

- Normal Bi-direction I/O Pin.

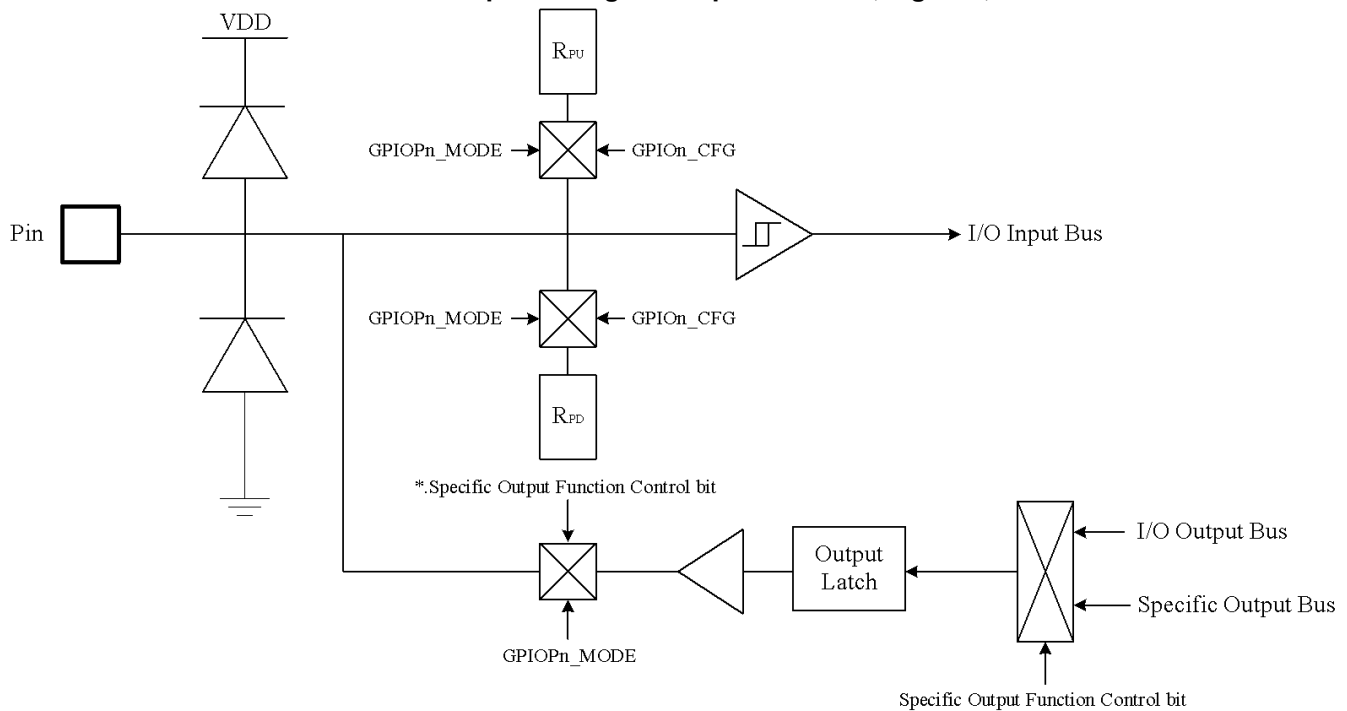


- Bi-direction I/O Pin Shared with Specific Digital Input Function, e.g. SPI, I2C...



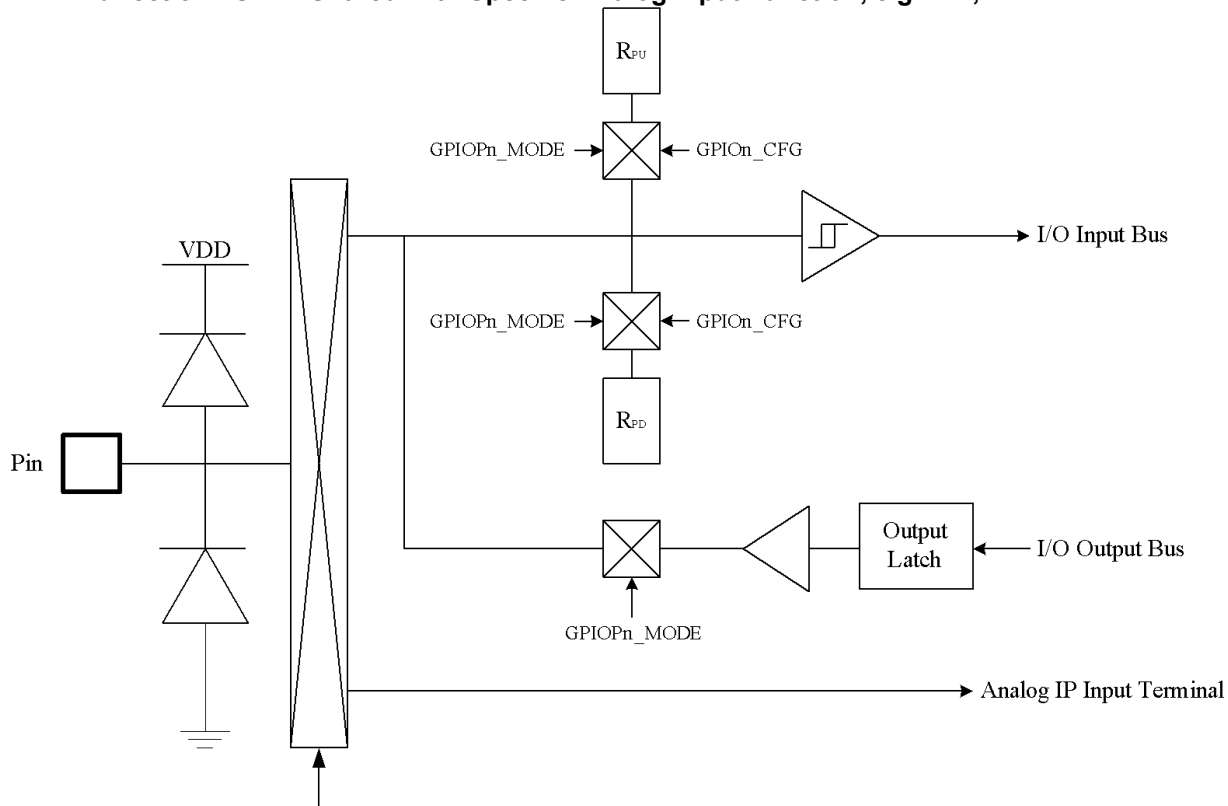
*. Some specific functions switch I/O direction directly, not through GPIOFn_MODE register.

● **Bi-direction I/O Pin Shared with Specific Digital Output Function, e.g. SPI, I2C...**



*. Some specific functions switch IO direction directly, not through GPIOn_MODE register.

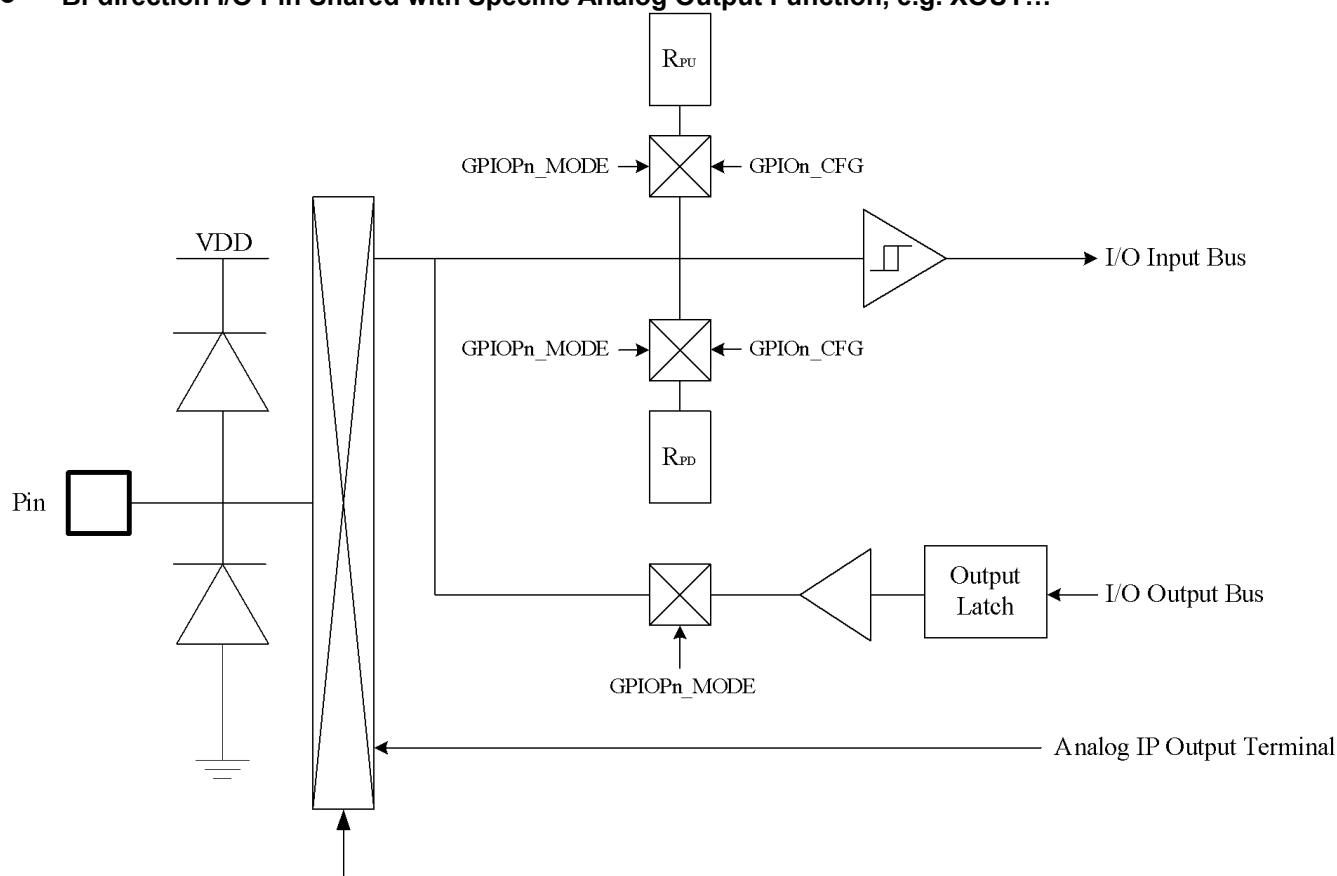
● **Bi-direction I/O Pin Shared with Specific Analog Input Function, e.g. XIN, ADC...**



*. Specific Analog Function Control bit

*. Some specific functions switch I/O direction directly, not through GPIOn_MODE register.

- **Bi-direction I/O Pin Shared with Specific Analog Output Function, e.g. XOUT...**



*. Specific Analog Function Control bit

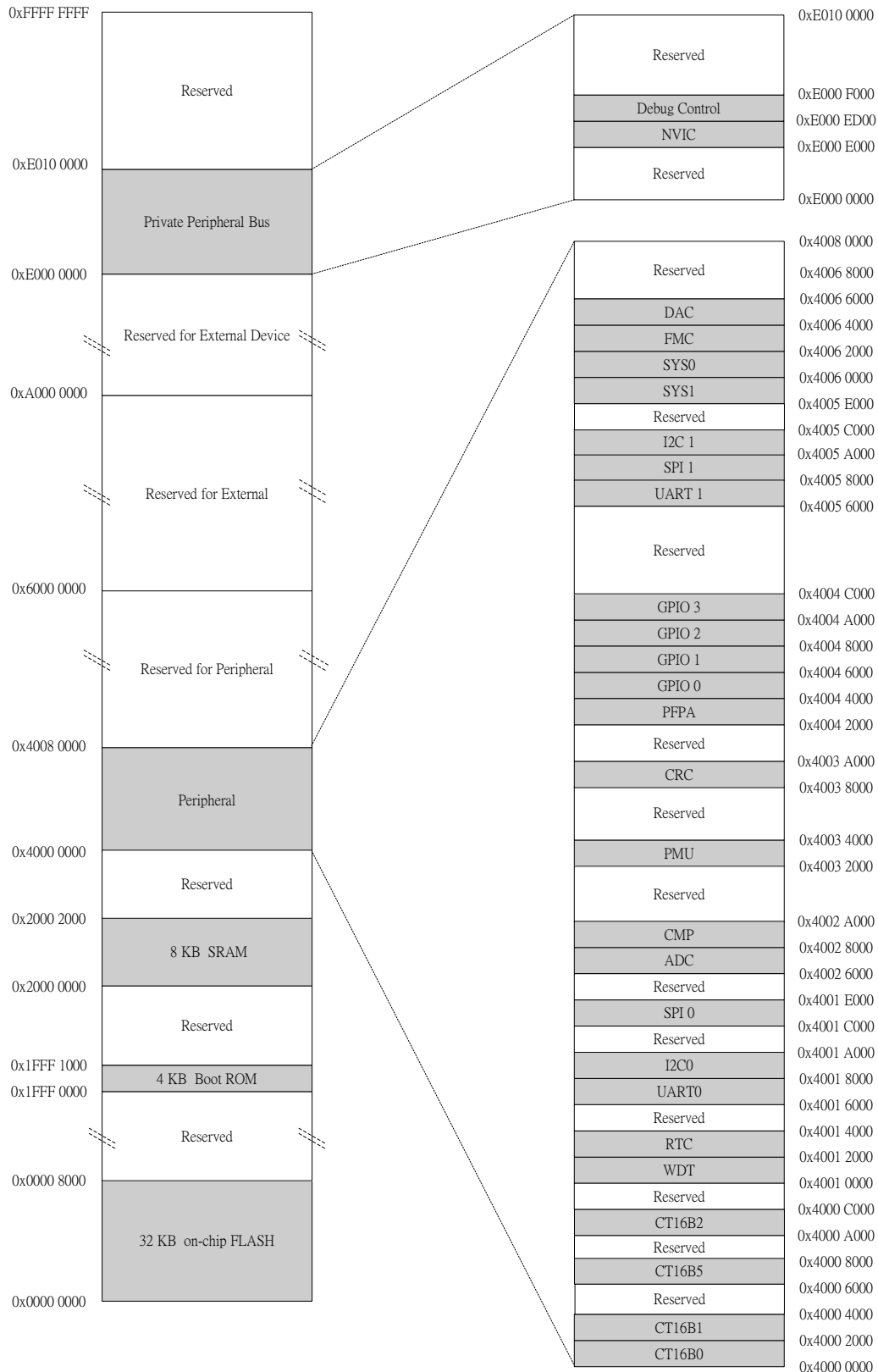
*. Some specific functions switch I /O direction directly , not through GPIOn_MODE register.

1.8 PIN CHARACTERISTICS

Port	Dir	ADC	CMP Input	DAC Output	Open-Drain	Pull-up Resistor (75KΩ@3.3V)	Pull-down Resistor (75KΩ@3.3V)	Drive Current 20mA @VDD-0.5V	Sink Current 20mA @VSS+0.5V
P0.2	I/O				V	V	V	V	V
P0.3	I/O				V	V	V	V	V
P0.4	I/O				V	V	V	V	V
P0.5	I/O				V	V	V	V	V
P0.6	I/O			V		V	V	V	V
P0.7	I/O			V		V	V	V	V
P0.8	I/O			V		V	V	V	V
P0.9	I/O			V		V	V	V	V
P0.10	I/O					V	V	V	V
P0.11	I/O					V	V	V	V
P1.0	I/O	V	V			V	V	V	V
P1.1	I/O	V	V		V	V	V	V	V
P1.2	I/O	V	V		V	V	V	V	V
P1.3	I/O	V	V		V	V	V	V	V
P1.4	I/O	V	V		V	V	V	V	V
P1.6	I/O				V	V	V	V	V
P1.7	I/O				V	V	V	V	V
P2.0	I/O	V	V			V	V	V	V
P2.1	I/O	V	V			V	V	V	V
P2.2	I/O	V	V			V	V	V	V
P2.3	I/O	V	V			V	V	V	V
P2.4	I/O	V	V			V	V	V	V
P2.5	I/O	V	V		V	V	V	V	V
P2.6	I/O	V	V		V	V	V	V	V
P3.4	I/O					V	V	V	V
P3.5	I/O					V	V	V	V
P3.6	I/O					V	V	V	V
P3.7	I/O					V	V	V	V
P3.8	I/O					V	V	V	V
P3.9	I/O					V	V	V	V

2 CENTRAL PROCESSOR UNIT (CPU)

2.1 MEMORY MAP



2.2 SYSTEM TICK TIMER

The SysTick timer is an integral part of the Cortex-M0. The SysTick timer is intended to generate a fixed 10-ms interrupt for use by an operating system or other system management software.

Since the SysTick timer is a part of the Cortex-M0, it facilitates porting of software by providing a standard timer that is available on Cortex-M0 based devices.

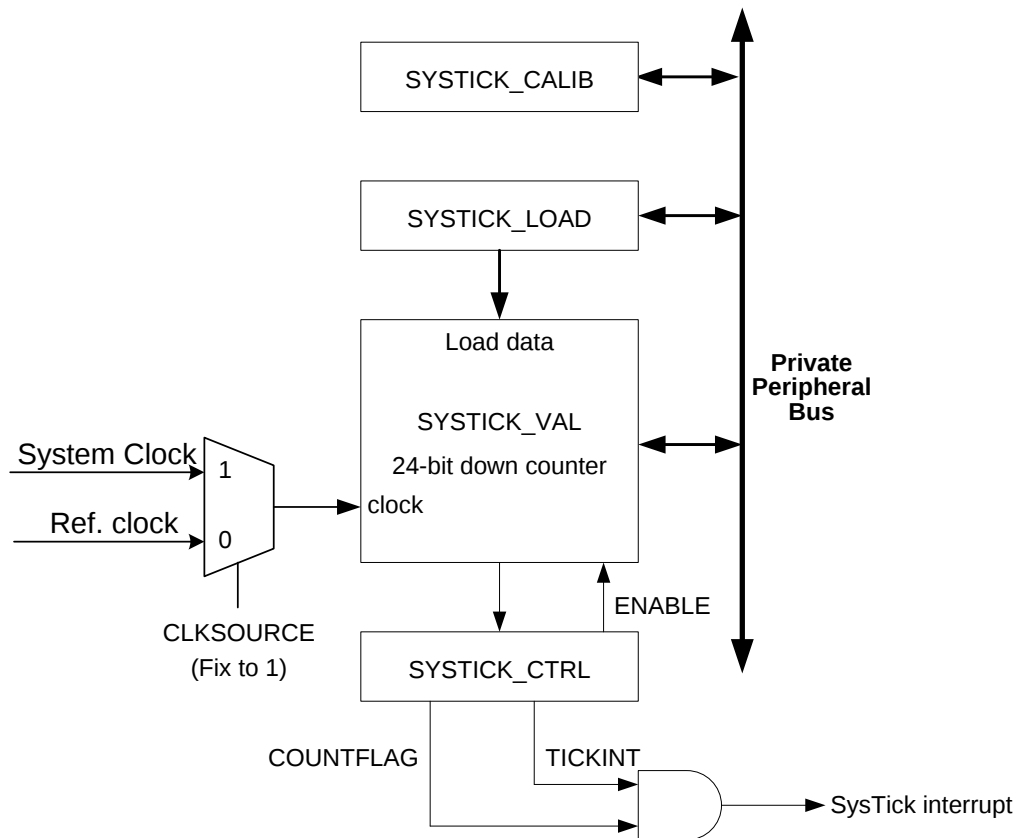
Refer to the *Cortex-M0 User Guide* for details.

2.2.1 OPERATION

The SysTick timer is a 24-bit timer that counts down to zero and generates an interrupt.

The intent is to provide a fixed 10-ms time interval between interrupts. The system tick timer is enabled through the SysTick control register. The system tick timer clock is fixed to the frequency of the system clock.

The block diagram of the SysTick timer:



When SysTick timer is enabled, the timer counts down from the current value (SYSTICK_VAL) to zero, reloads to the value in the SysTick Reload Value Register (SYSTICK_LOAD) on the next clock edge, then decrements on subsequent clocks. When the counter transitions to zero, the COUNTFLAG status bit is set to 1. The COUNTFLAG bit clears on reads.

*** Note: When the processor is halted for debugging the counter does not decrease.**

2.2.2 SYSTICK USAGE HINTS AND TIPS

The interrupt controller clock updates the SysTick counter. Some implementations stop this clock signal for low power mode. If this happens, the SysTick counter stops.

Ensure SW uses word accesses to access the SysTick registers.

The SysTick counter reload and current value are not initialized by HW. This means the correct initialization sequence for the SysTick counter is:

1. Program the reload value in SYSTICK_LOAD register.
2. Clear the current value by writing any value to SYSTICK_VAL register.
3. Program the Control and Status (SYSTICK_CTRL) register.

2.2.3 SYSTICK REGISTERS

2.2.3.1 System Tick Timer Control and Status register (SYSTICK_CTRL)

Address: 0xE000 E010 (Refer to Cortex-M0 Spec)

Bit	Name	Description	Attribute	Reset
31:17	Reserved		R	0
16	COUNTFLAG	This flag is set when the System Tick counter counts down to 0, and is cleared by reading this register.	R/W	0
15:3	Reserved		R	0
2	CLKSOURCE	Selects the SysTick timer clock source. 0: reference clock. 1: system clock. (Fixed)	R	1
1	TICKINT	System Tick interrupt enable. 0: Disable the System Tick interrupt 1: Enable the System Tick interrupt, the interrupt is generated when the System Tick counter counts down to 0.	R/W	0
0	ENABLE	System Tick counter enable. 0: Disable 1: Enable	R/W	0

2.2.3.2 System Tick Timer Reload value register (SYSTICK_LOAD)

Address: 0xE000 E014 (Refer to Cortex-M0 Spec)

The RELOAD register is set to the value that will be loaded into the SysTick timer whenever it counts down to zero. This register is set by software as part of timer initialization. The SYSTICK_CALIB register may be read and used as the value for RELOAD if the CPU or external clock is running at the frequency intended for use with the SYSTICK_CALIB value.

The following example illustrates selecting the SysTick timer reload value to obtain a 10 ms time interval with the system clock set to 50 MHz.

The SysTick clock = system clock = 50 MHz

RELOAD = (system tick clock frequency × 10 ms) - 1 = (50 MHz × 10 ms) - 1 = 0x0007A11F.

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23:0	RELOAD	Value to load into the SYSTICK_VAL when the counter is enabled and when it reaches 0.	R/W	0x5F7F9B

2.2.3.3 System Tick Timer Current Value register (SYSTICK_VAL)

Address: 0xE000 E018 (Refer to Cortex-M0 Spec)

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23:0	CURRENT	Reading this register returns the current value of the System Tick counter. Writing any value clears the System Tick counter and the COUNTFLAG bit in SYSTICK_CTRL.	R/W	0x7E7F35

2.2.3.4 System Tick Timer Calibration Value register (SYSTICK_CALIB)

Address: 0xE000 E01C (Refer to Cortex-M0 Spec)

Bit	Name	Description	Attribute	Reset
31	NOREF	Indicates the reference clock to M0 is provided or not. 1: No reference clock provided.	R	1
30	SKEW	Indicates whether the TENMS value is exact, an inexact TENMS value can affect the suitability of SysTick as a software real time clock. 0: TENMS value is exact 1: TENMS value is inexact, or not given.	R	0
29:24	Reserved		R	0
23:0	TENMS	Reload value for 10ms timing, subject to system clock skew errors. If the value reads as zero, the calibration value is not known.	R/W	0xA71FF

2.3 NESTED VECTORED INTERRUPT CONTROLLER (NVIC)

All interrupts including the core exceptions are managed by the NVIC. NVIC has the following Features:

- The NVIC supports 32 vectored interrupts.
- 4 programmable interrupt priority levels with hardware priority level masking.
- Low-latency exception and interrupt handling.
- Efficient processing of late arriving interrupts.
- Implementation of System Control Registers
- Software interrupt generation.

2.3.1 INTERRUPT AND EXCEPTION VECTORS

Execution No.	Priority	Function	Description	Address Offset
0	-	-	Reserved	0x0000 0000
1	-3	Reset	Reset	0x0000 0004
2	-2	NMI_Handler	Non maskable interrupt.	0x0000 0008
3	-1	HardFault_Handler	All class of fault	0x0000 000C
4~10	Reserved	Reserved	Reserved	-
11	Settable	SVCCall		0x0000 002C
12~13	Reserved	Reserved	Reserved	-
14	Settable	PendSV		0x0000 0038
15	Settable	SysTick		0x0000 003C
16	Settable	IRQ0/NDTIRQ	NDT	0x0000 0040
17	Settable	IRQ1/CMP0IRQ	CMP0	0x0000 0044
18	Settable	IRQ2/CMP1IRQ	CMP1	0x0000 0048
19	Settable	IRQ3/DACIRQ	DAC	0x0000 004C
20	Settable	IRQ4/		0x0000 0050
21	Settable	IRQ5/	CMP3	0x0000 0054
22	Settable	IRQ6/	SPI0	0x0000 0058
23	Settable	IRQ7/	SPI1	0x0000 005C
24	Settable	IRQ8/	UART2	0x0000 0060
25	Settable	IRQ9/	UART3	0x0000 0064
26	Settable	IRQ10/	I2C0	0x0000 0068
27	Settable	IRQ11/	I2C1	0x0000 006C
28	Settable	IRQ12/CT16B2IRQ	CT16B2	0x0000 0070
29	Settable	IRQ13/SPI0IRQ	SPI0	0x0000 0074
30	Settable	IRQ14/		0x0000 0078

31	Settable	IRQ15/I2C0IRQ	I2C0	0x0000 007C
32	Settable	IRQ16/CT16B0IRQ	CT16B0	0x0000 0080
33	Settable	IRQ17/CT16B1IRQ	CT16B1	0x0000 0084
34	Settable	IRQ18/		0x0000 0088
35	Settable	IRQ19/CT16B5IRQ	CT16B5	0x0000 008C
36	Settable	IRQ20/		0x0000 0090
37	Settable	IRQ21/UART0IRQ	UART0	0x0000 0094
38	Settable	IRQ22/UART1IRQ	UART1	0x0000 0098
39	Settable	IRQ23/I2C1IRQ	I2C1	0x0000 009C
40	Settable	IRQ24/ADCIRQ	ADC	0x0000 00A0
41	Settable	IRQ25/WDTIRQ	WDT	0x0000 00A4
42	Settable	IRQ26/LVDIRQ	LVD	0x0000 00A8
43	Settable	IRQ27/RTCIRQ	RTC	0x0000 00AC
44	Settable	IRQ28/P3IRQ	GPIO3	0x0000 00B0
45	Settable	IRQ29/P2IRQ	GPIO2	0x0000 00B4
46	Settable	IRQ30/P1IRQ	GPIO1	0x0000 00B8
47	Settable	IRQ31/P0IRQ	GPIO0	0x0000 00BC

2.3.2 NVIC REGISTERS

2.3.2.1 IRQ0~31 Interrupt Set-Enable Register (NVIC_ISER)

Address: 0xE000 E100 (Refer to Cortex-M0 Spec.)

The ISER enables interrupts, and shows the interrupts that are enabled.

Bit	Name	Description	Attribute	Reset
31:0	SETENA[31:0]	Interrupt set-enable bits. Write→ 0: No effect 1: Enable interrupt. Read→ 0: Interrupt disabled 1: Interrupt enabled.	R/W	0

2.3.2.2 IRQ0~31 Interrupt Clear-Enable Register (NVIC_ICER)

Address: 0xE000 E180 (Refer to Cortex-M0 Spec.)

The ICER disables interrupts, and shows the interrupts that are enabled.

Bit	Name	Description	Attribute	Reset
31:0	CLRENA[31:0]	Interrupt clear-enable bits. Write→ 0: No effect 1: Disable interrupt. Read→ 0: Interrupt disabled 1: Interrupt enabled.	R/W	0

2.3.2.3 IRQ0~31 Interrupt Set-Pending Register (NVIC_ISPR)

Address: 0xE000 E200 (Refer to Cortex-M0 Spec.)

The ISPR forces interrupts into the pending state, and shows the interrupts that are pending.

- * **Note: Writing 1 to the ISPR bit corresponding to**
1. **an interrupt that is pending has no effect**
 2. **a disabled interrupt sets the state of that interrupt to pending.**

Bit	Name	Description	Attribute	Reset
31:0	SETPEND[31:0]	Interrupt set-pending bits. Write→ 0: No effect 1: Change interrupt state to pending Read→ 0: Interrupt is not pending 1: Interrupt is pending	R/W	0

2.3.2.4 IRQ0~31 Interrupt Clear-Pending Register (NVIC_ICPR)

Address: 0xE000 E280 (Refer to Cortex-M0 Spec.)

The ICPR removes the pending state from interrupts, and shows the interrupts that are pending.

- * **Note: Writing 1 to an ICPR bit does not affect the active state of the corresponding interrupt.**

Bit	Name	Description	Attribute	Reset
31:0	CLRPEND[31:0]	Interrupt clear-pending bits. Write→ 0: No effect 1: Removes pending state of an interrupt Read→ 0: Interrupt is not pending 1: Interrupt is pending	R/W	0

2.3.2.5 IRQ0~31 Interrupt Priority Register (NVIC_IPRn) (n=0~7)

Address: 0xE000 E400 + 0x4 * n (Refer to Cortex-M0 Spec.)

The interrupt priority registers provide an 8-bit priority field for each interrupt, and each register holds four priority fields. This means the number of registers is implementation-defined, and corresponds to the number of implemented interrupts.

Bit	Name	Description	Attribute	Reset
31:24	PRI_(4*n+3)	Each priority field holds a priority value, 0-192. The lower the value, the greater the priority of the corresponding interrupt. The processor implements only bits[31:30] of each field, bits [29:24] read as zero and ignore writes. This means writing 255 to a priority register saves value 192 to the register.	R/W	0
23:16	PRI_(4*n+2)	Each priority field holds a priority value, 0-192. The lower the value, the greater the priority of the corresponding interrupt. The processor implements only bits[23:22] of each field, bits [21:16] read as zero and ignore writes. This means writing 255 to a priority register saves value 192 to the register.	R/W	0
15:8	PRI_(4*n+1)	Each priority field holds a priority value, 0-192. The lower the value, the greater the priority of the corresponding interrupt. The processor implements only bits[15:14] of each field, bits [13:8] read as zero and ignore writes. This means writing 255 to a priority register saves value 192 to the register.	R/W	0
7:0	PRI_4*n	Each priority field holds a priority value, 0-192. The lower the value, the greater the priority of the corresponding interrupt. The processor implements	R/W	0

	only bits[7:6] of each field, bits [5:0] read as zero and ignore writes. This means writing 255 to a priority register saves value 192 to the register.		
--	---	--	--

2.4 APPLICATION INTERRUPT AND RESET CONTROL (AIRC)

Address: 0xE000 ED0C (Refer to Cortex-M0 Spec)

The entire MCU, including the core, can be reset by SW by setting the SYSRESREQ bit in the AIRC register in Cortex-M0 spec.

*** Note: To write to this register, user must write 0x05FA to the VECTKEY field at the same time, otherwise the processor ignores the write.**

Bit	Name	Description	Attribute	Reset
31:16	VECTKEY	Register key. Read as unknown. Write 0x05FA to VECTKEY, otherwise the write is ignored.	R/W	0
15	ENDIANESS	Data endianness implemented 0: Little-endian 1: Big-endian	R	0
14:3	Reserved		R	0
2	SYSRESETREQ	System reset request. This bit read as 0. 0: No effect 1: Requests a system level reset.	W	0
1	VECTCLRACTIVE	Reserved for debug use. This bit read as 0. When writing to the register you must write 0 to this bit, otherwise behavior is Unpredictable.	W	0
0	Reserved		R	0

2.5 CODE OPTION TABLE

Address: 0x1FFF 2000

Bit	Name	Description	Attribute	Reset
31:16	Code Security[15:0]	Code Security 0xFFFF: CS0 0x5A5A: CS1 0xA5A5: CS2	R/W	0xFFFF
15:4	Reserved		R/W	0xFFFC
3	Reserved		R/W	1
2	BOOTPINEN	Boot Pin enable 0: Disable 1: Enable	R/W	1
1	EXTRSTHWDIS	External reset HW disable bit 0: Enable (HW enables, FW can NOT control with SYS0_EXRSTCTRL) 1: Disable (FW can control with SYS0_EXRSTCTRL)	R/W	1
0	BLEN	Boot loader enable 0: Disable 1: Enable	R/W	1

2.6 UNIQUE NUMBER

The unique number is a 8-byte unique device serial number of each IC. In other words, the unique number is different and discontinuous for each IC.

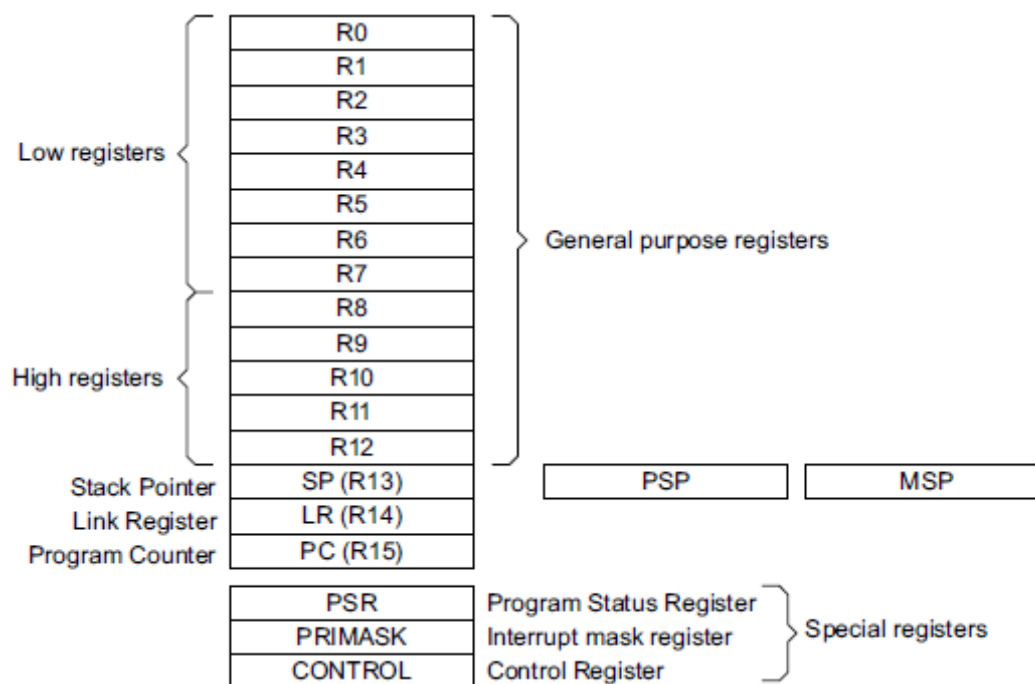
Address: 0x1FFF 221C

Bit	Name	Description	Attribute	Reset
31:0	L4BYTE[31:0]	Lower 4 bytes of Unique number	R	By Die

Address: 0x1FFF 2220

Bit	Name	Description	Attribute	Reset
31:0	H4BYTE[31:0]	High 4 bytes of Unique number	R	By Die

2.7 CORE REGISTER OVERVIEW



Register	Description (Refer to Cortex-M0 Spec)
R0~R12	General-purpose registers for data operations.
SP (R13)	The Stack Pointer (SP). In Thread mode, the CONTROL register indicates the stack pointer to use, Main Stack Pointer (MSP) or Process Stack Pointer (PSP) On reset, the processor loads the MSP with the value from address 0x00000000.
LR (R14)	The Link Register (LR). It stores the return information for subroutines, function calls, and exceptions.
PC (R15)	The Program Counter (PC). It contains the current program address. On reset, the processor loads the PC with the value of the reset vector, at address 0x00000004.
PSR	The Program Status Register (PSR) combines: • Application Program Status Register (APSR) • Interrupt Program Status Register (IPSR) • Execution Program Status Register (EPSR). These registers are mutually exclusive bit fields in the 32-bit PSR. The diagram shows the 32-bit PSR structure with bit positions 31 down to 0. The APSR section (bits 31-27) includes flags N, Z, C, and V. The IPSR section (bits 26-2) includes an exception number. The EPSR section (bits 1-0) includes a T bit. Various other bits are reserved.
PRIMASK	The PRIMASK register prevents activation of all exceptions with configurable priority.
CONTROL	The CONTROL register controls the stack used when the processor is in Thread mode.

3 SYSTEM CONTROL

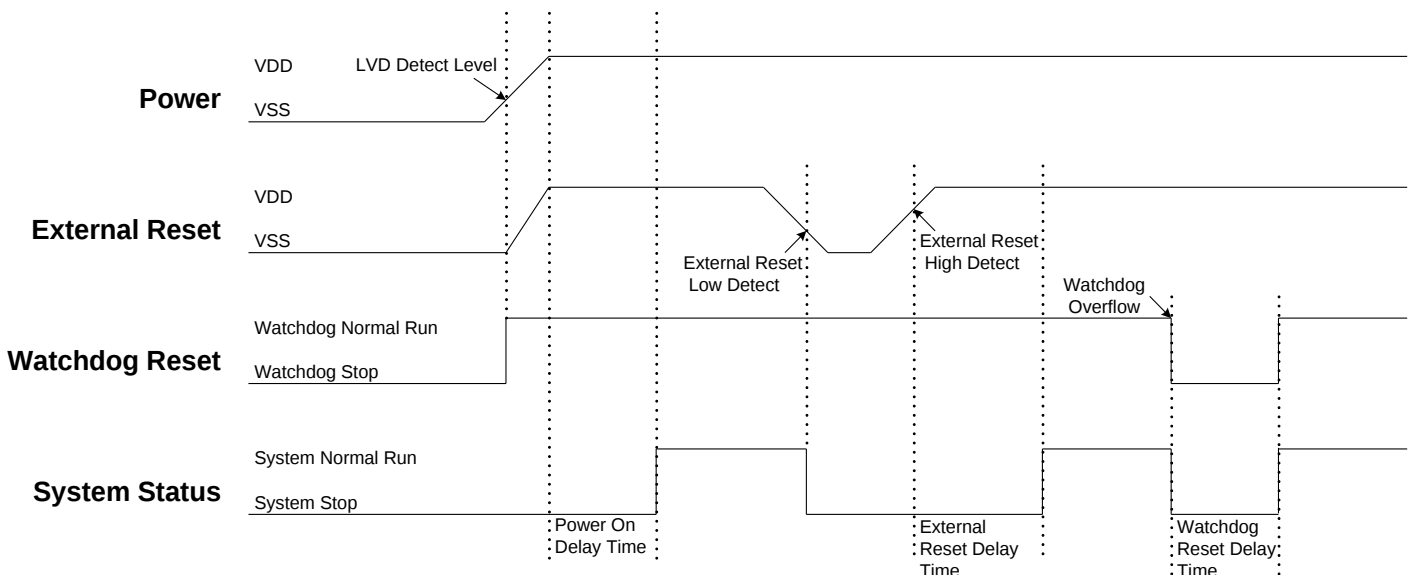
3.1 RESET

A system reset is generated when one of the following events occurs:

1. A low level on the RST pin (external reset).
2. Power-on reset (POR reset)
3. LVD reset
4. Watchdog Timer reset (WDT reset)
5. Software reset (SW reset)

The reset source can be identified by checking the reset flags in [System Reset Status register \(SYS0_RSTST\)](#). These sources act on the RST pin and it is always kept low during the delay phase. The RESET service routine vector is fixed at address 0x00000004 in the memory map. For more details, refer to [Interrupt and Exception Vectors](#).

Finishing any reset sequence needs some time. The system provides complete procedures to make the power on reset successful. For different oscillator types, the reset time is different. That causes the VDD rise rate and start-up time of different oscillator is not fixed. RC type oscillator's start-up time is very short, but the crystal type is longer. Under client terminal application, users have to take care of the power on reset time for the master terminal requirement. The reset timing diagram is as following.



3.1.1 POWER-ON RESET (POR)

The power on reset depends on LVD operation for most power-up situations. The power supplying to system is a rising curve and needs some time to achieve the normal voltage. Power on reset sequence is as following:

- **Power-up:** System detects the power voltage up and waits for power stable.
- **External reset (only external reset pin enable):** System checks external reset pin status. If external reset pin is not high level, the system keeps reset status and waits external reset pin released.
- **System initialization:** All system registers is set as initial conditions and system is ready.
- **Oscillator warm up:** Oscillator operation is successfully and supply to system clock.
- **Program executing:** Power on sequence is finished and program executes from Boot loader.

3.1.2 WATCHDOG RESET (WDT RESET)

Watchdog reset is a system protection. In normal condition, system works well and clears watchdog timer by program. Under error condition, system is in unknown situation and watchdog can't be clear by program before watchdog timer overflow. Watchdog timer overflow occurs and the system is reset. After watchdog reset, the system restarts and returns normal mode. Watchdog reset sequence is as following.

- **Watchdog timer status:** System checks watchdog timer overflow status. If watchdog timer overflow occurs, the system is reset.
- **System initialization:** All system registers is set as initial conditions and system is ready.
- **Oscillator warm up:** Oscillator operation is successfully and supply to system clock.
- **Program executing:** Power on sequence is finished and program executes from 0x0.

Watchdog timer application note is as following.

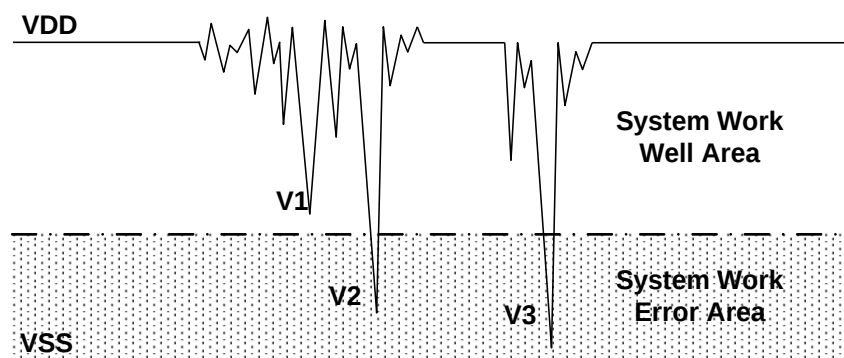
- Before clearing watchdog timer, check I/O status and check RAM contents can improve system error.
- Don't clear watchdog timer in interrupt vector and interrupt service routine. That can improve main routine fail.
- Clearing watchdog timer program is only at one part of the program. This way is the best structure to enhance the watchdog timer function.

* **Note:** Please refer to the "WATCHDOG TIMER" about watchdog timer detail information.

3.1.3 BROWN-OUT RESET

3.1.3.1 BROWN OUT DESCRIPTION

The brown-out reset is a power dropping condition. The power drops from normal voltage to low voltage by external factors (e.g. EFT interference or external loading changed). The brown out reset would make the system not work well or executing program error.



Brown-Out Reset Diagram

The power dropping might through the voltage range that's the system dead-band. The dead-band means the power range can't offer the system minimum operation power requirement. The above diagram is a typical brown out reset diagram. There is a serious noise under the VDD, and VDD voltage drops very deep. There is a dotted line to separate the system working area. The above area is the system work well area. The below area is the system work error area called dead-band. V1 doesn't touch the below area and not affect the system operation. But the V2 and V3 is under the below area and may induce the system error occurrence. Let system under dead-band includes some conditions.

DC application:

The power source of DC application is usually using battery. When low battery condition and MCU drive any loading, the power drops and keeps in dead-band. Under the situation, the power won't drop deeper and not touch the system reset voltage. That makes the system under dead-band.

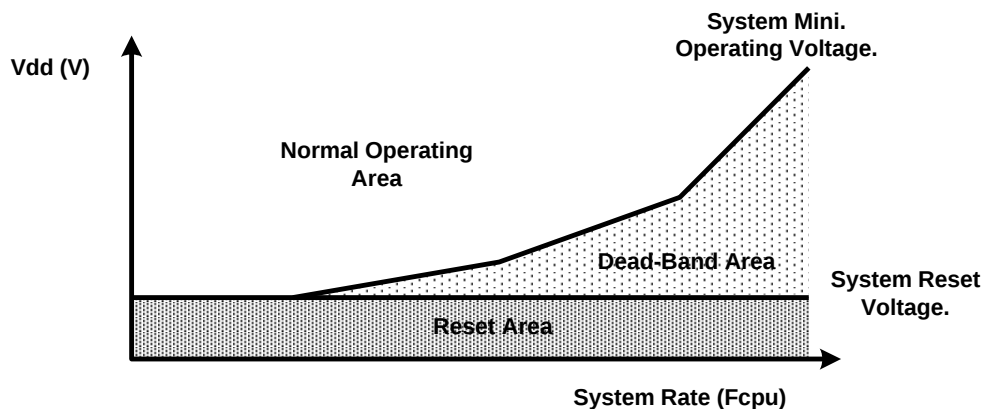
AC application:

In AC power application, the DC power is regulated from AC power source. This kind of power usually couples with AC noise that makes the DC power dirty. Or the external loading is very heavy, e.g. driving motor. The loading operating induces noise and overlaps with the DC power. VDD drops by the noise, and the system works under unstable power situation.

The power on duration and power down duration are longer in AC application. The system power on sequence protects the power on successful, but the power down situation is like DC low battery condition. When turn off the AC power, the VDD drops slowly and through the dead-band for a while.

3.1.3.2 THE SYSTEM OPERATING VOLTAGE DECSRIPTION

To improve the brown out reset needs to know the system minimum operating voltage which is depend on the system executing rate and power level. Different system executing rates have different system minimum operating voltage. The electrical characteristic section shows the system voltage to executing rate relationship.



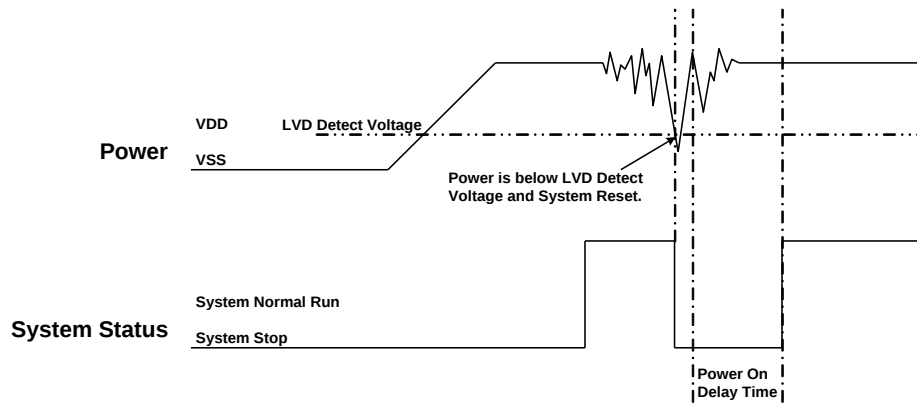
Normally the system operation voltage area is higher than the system reset voltage to VDD, and the reset voltage is decided by LVD detect level. The system minimum operating voltage rises when the system executing rate upper even higher than system reset voltage. The dead-band definition is the system minimum operating voltage above the system reset voltage.

3.1.3.3 BROWN-OUT RESET IMPROVEMENT

How to improve the brown reset condition? There are some methods to improve brown out reset as following.

- LVD reset
- Watchdog reset
- Reduce the system executing rate
- External reset circuit. (Zener diode reset circuit, Voltage bias reset circuit, External reset IC)

*** Note:** The “Zener diode reset circuit”, “Voltage bias reset circuit” and “External reset IC” can completely improve the brown out reset, DC low battery and AC slow power down conditions.

LVD reset:

The LVD (low voltage detector) is built-in SONiX 32-bit MCU to be brown out reset protection. When the VDD drops and is below LVD detect voltage, the LVD asserts an interrupt signal to the NVIC. This signal can be enabled for interrupt in the Interrupt Enable Register in the NVIC in order to cause a CPU interrupt; if not, SW can monitor the signal by reading a dedicated status register. An additional threshold level can be selected to cause a forced reset of the chip. The LVD detect level is different by each MCU. The LVD voltage level is a point of voltage and not easy to cover all dead-band range. Using LVD to improve brown out reset is dependent on application requirement and environment. If the power variation is very deep, violent and trigger the LVD, the LVD can be the protection. If the power variation can touch the LVD detect level and make system work error, the LVD can't be the protection and need to other reset methods. More detail LVD information is in the electrical characteristic section.

Watchdog reset:

The watchdog timer is a protection to make sure the system executes well. Normally the watchdog timer would be clear at one point of program. Don't clear the watchdog timer in several addresses. The system executes normally and the watchdog won't reset system. When the system is under dead-band and the execution error, the watchdog timer can't be clear by program. The watchdog is continuously counting until overflow occurrence. The overflow signal of watchdog timer triggers the system to reset and return to normal mode after reset sequence. This method also can improve brown out reset condition and make sure the system to return normal mode.

If the system reset by watchdog and the power is still in dead-band, the system reset sequence won't be successful and the system stays in reset status until the power return to normal range.

Reduce the system executing rate:

If the system rate is fast and the dead-band exists, to reduce the system executing rate can improve the dead-band. The lower system rate is with lower minimum operating voltage. Select the power voltage that's no dead-band issue and find out the mapping system rate. Adjust the system rate to the value and the system exits the dead-band issue. This way needs to modify whole program timing to fit the application requirement.

External reset circuit:

The external reset methods also can improve brown out reset and is the complete solution. There are three external reset circuits to improve brown out reset including "Zener diode reset circuit", "Voltage bias reset circuit" and "External reset IC". These three reset structures use external reset signal and control to make sure the MCU be reset under power dropping and under dead-band. The external reset information is described in the next section.

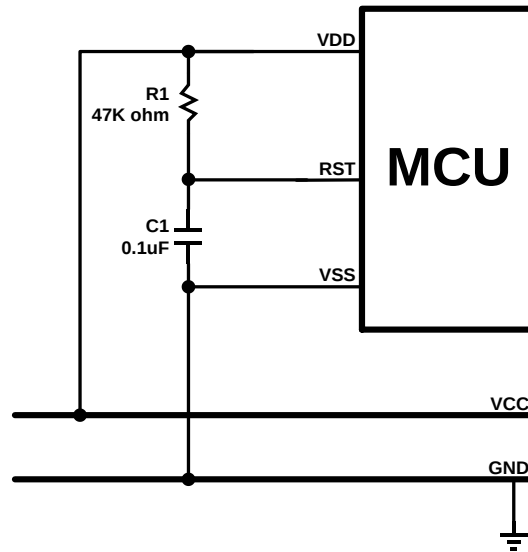
3.1.4 EXTERNAL RESET

External reset function is controlled by [External RESET pin control \(SYS0_EXRSTCTRL\)](#) register. Default value is 1, which means external reset function is disabled. External reset pin is Schmitt Trigger structure and low level active. The system is running when reset pin is high level voltage input. The reset pin receives the low voltage and the system is reset. The external reset operation activates in power on and normal running mode. During system power-up, the external reset pin must be high level input, or the system keeps in reset status. External reset sequence is as following.

- **External reset (only external reset pin enable):** System checks external reset pin status. If external reset pin is not high level, the system keeps reset status and waits external reset pin released.
- **System initialization:** All system registers is set as initial conditions and system is ready.
- **Oscillator warm up:** Oscillator operation is successfully and supply to system clock.
- **Program executing:** Power on sequence is finished and program executes from Boot loader.

The external reset can reset the system during power on duration, and good external reset circuit can protect the system to avoid working at unusual power condition, e.g. brown out reset in AC power application.

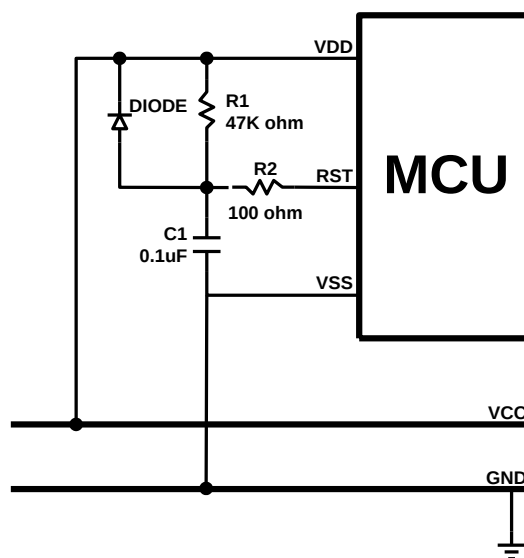
3.1.4.1 SIMPLY RC RESET CIRCUIT



This is the basic reset circuit, and only includes R1 and C1. The RC circuit operation makes a slow rising signal into reset pin as power up. The reset signal is slower than VDD power up timing, and system occurs a power on signal from the timing difference.

*** Note: The reset circuit is no any protection against unusual power or brown out reset.**

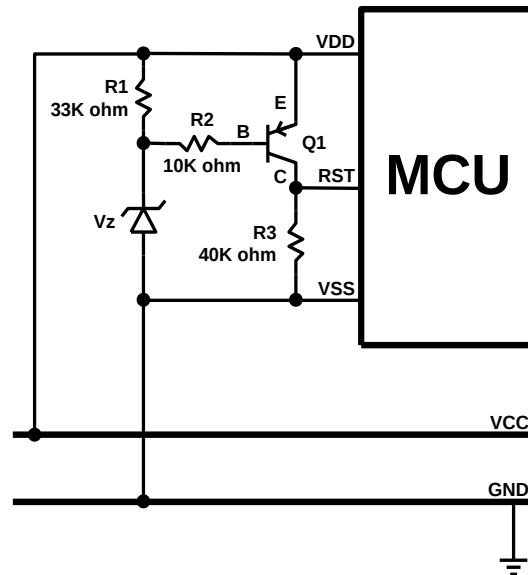
3.1.4.2 DIODE & RC RESET CIRCUIT



This is the better reset circuit. The R1 and C1 circuit operation is like the simply reset circuit to make a power on signal. The reset circuit has a simply protection against unusual power. The diode offers a power positive path to conduct higher power to VDD. It is can make reset pin voltage level to synchronize with VDD voltage. The structure can improve slight brown out reset condition.

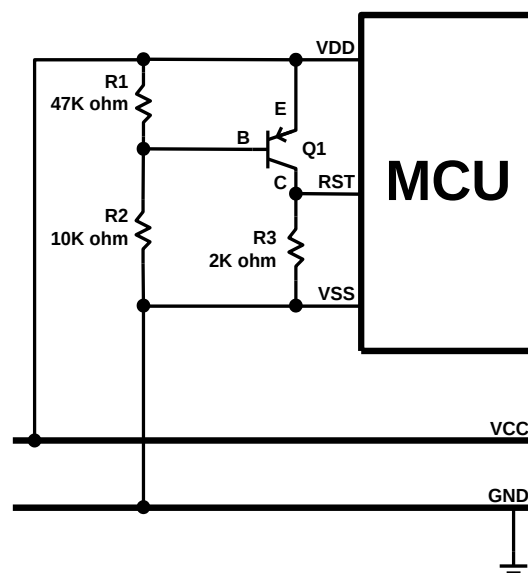
* **Note:** The R2 100 ohm resistor of “Simply reset circuit” and “Diode & RC reset circuit” is necessary to limit any current flowing into reset pin from external capacitor C in the event of reset pin breakdown due to Electrostatic Discharge (ESD) or Electrical Over-stress (EOS).

3.1.4.3 ZENER DIODE RESET CIRCUIT



The Zener diode reset circuit is a simple low voltage detector and can **improve brown out reset condition completely**. Use Zener voltage to be the active level. When VDD voltage level is above “ $V_z + 0.7V$ ”, the C terminal of the PNP transistor outputs high voltage and MCU operates normally. When VDD is below “ $V_z + 0.7V$ ”, the C terminal of the PNP transistor outputs low voltage and MCU is in reset mode. Decide the reset detect voltage by Zener specification. Select the right Zener voltage to conform the application.

3.1.4.4 VOLTAGE BIAS RESET CIRCUIT



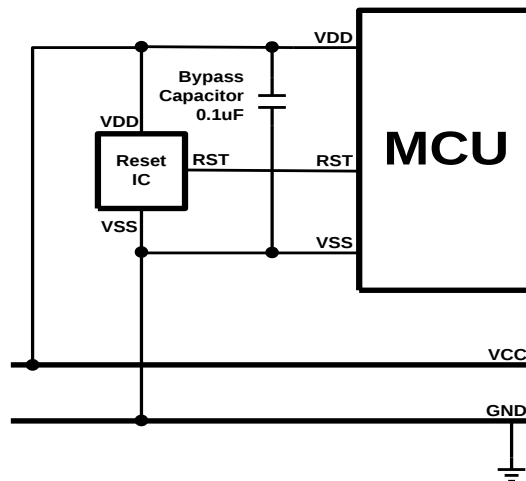
The voltage bias reset circuit is a low cost voltage detector and can **improve brown out reset condition completely**. The operating voltage is not accurate as Zener diode reset circuit. Use R1, R2 bias voltage to be the active level. When

VDD voltage level is above or equal to $0.7V \times (R1 + R2) / R1$, the C terminal of the PNP transistor outputs high voltage and MCU operates normally. When VDD is below $0.7V \times (R1 + R2) / R1$, the C terminal of the PNP transistor outputs low voltage and MCU is in reset mode.

Decide the reset detect voltage by R1, R2 resistances. Select the right R1, R2 value to conform the application. In the circuit diagram condition, the MCU's reset pin level varies with VDD voltage variation, and the differential voltage is 0.7V. If the VDD drops and the voltage lower than reset pin detect level, the system would be reset. If want to make the reset active earlier, set the $R2 > R1$ and the cap between VDD and C terminal voltage is larger than 0.7V. The external reset circuit is with a stable current through R1 and R2. For power consumption issue application, e.g. DC power system, the current must be considered to whole system power consumption.

* **Note:** Under unstable power condition as brown out reset, “Zener diode reset circuit” and “Voltage bias reset circuit” can protect system no any error occurrence as power dropping. When power drops below the reset detect voltage, the system reset would be triggered, and then system executes reset sequence. That makes sure the system work well under unstable power situation.

3.1.4.5 EXTERNAL RESET IC



The external reset circuit also uses external reset IC to enhance MCU reset performance. This is a high cost and good effect solution. By different application and system requirement to select suitable reset IC. The reset circuit can improve all power variation.

3.1.5 SOFTWARE RESET

The entire MCU, including the core, can be reset by software by setting the SYSRESREQ bit in the [AIRC \(Application Interrupt and Reset Control\)](#) register in Cortex-M0 spec.

The software-initiated system reset sequence is as follows:

1. A software reset is initiated by setting the SYSRESREQ bit.
2. An internal reset is asserted.
3. The internal reset is deasserted and the MCU loads from memory the initial stack pointer, the initial program counter, and the first instruction designated by the program counter, and then begins execution.

3.2 SYSTEM CLOCK

Different clock sources can be used to drive the system clock (SYSCLK):

- 48 MHz internal high speed RC (IHRC)
- 32 KHz internal low speed RC (ILRC)
- PLL clock
- High speed external (EHS) crystal clock
- Low speed external (ELS) 32.768 KHz crystal

Each clock source can be switched on or off independently when it is not used, to optimize power consumption.

The micro-controller is a dual clock system. There are high-speed clock and low-speed clock. The high-speed clock is generated from the external oscillator & on-chip PLL circuit. The low-speed clock is generated from on-chip low-speed RC oscillator circuit (ILRC 32KHz).

3.2.1 INTERNAL RC CLOCK SOURCE

3.2.1.1 Internal High-speed RC Oscillator (IHRC)

The internal high-speed oscillator is 48MHz RC type. The accuracy is $\pm 2.5\%$ under commercial condition.

The IHRC can be switched on and off using the IHRcen bit in [Analog Block Control register \(SYS0_ANBCTRL\)](#).

3.2.1.2 Internal Low-speed RC Oscillator (ILRC)

The system low clock source is the internal low-speed oscillator built in the micro-controller. The low-speed oscillator uses RC type oscillator circuit. The frequency is affected by the voltage and temperature of the system. In common condition, the frequency of the RC oscillator is about 32 KHz.

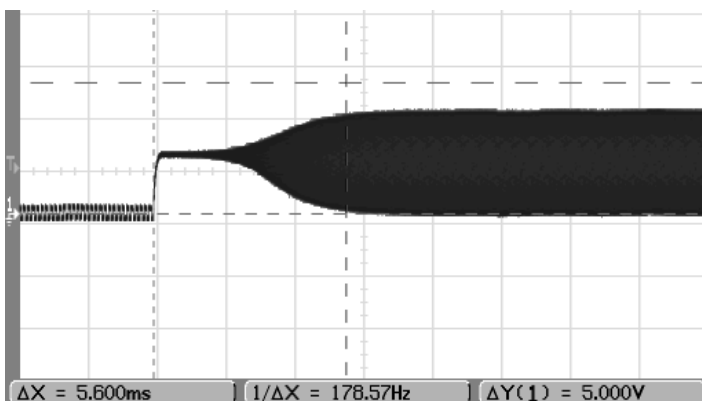
* **Note:** The ILRC can ONLY be switched on and off by HW.

3.2.2 EXTERNAL CLOCK SOURCE

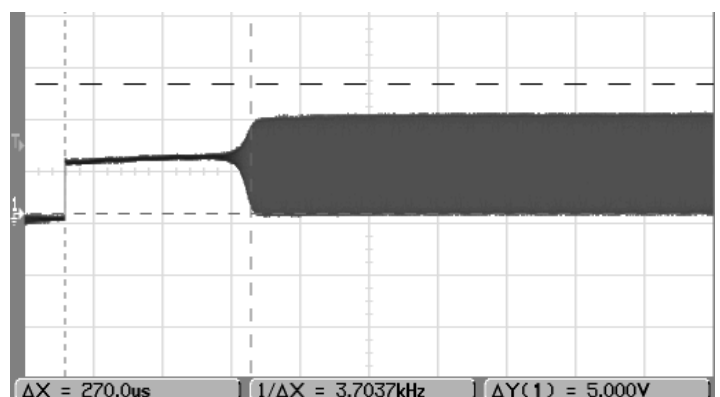
3.2.2.1 External High-speed (EHS) Clock

External high clock includes Crystal/Ceramic modules. The startup time of Crystal is longer. The oscillator start-up time decides reset time length.

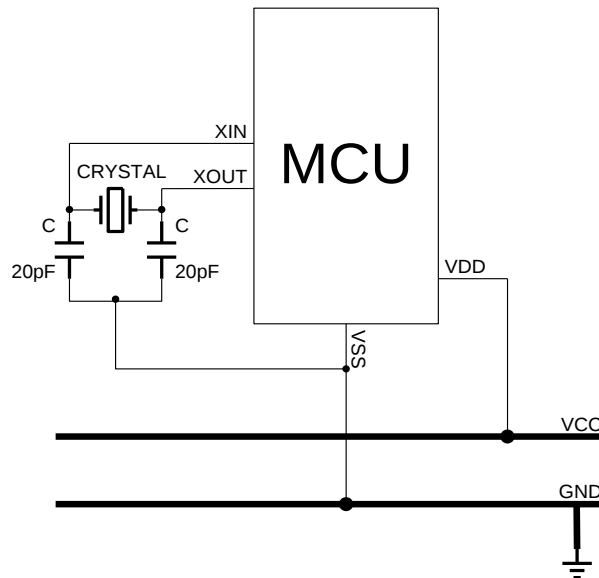
4MHz Crystal



4MHz Ceramic



Crystal/Ceramic devices are driven by XIN, XOUT pins. For high/normal/low frequency, the driving currents are different.



★ **Note:** Connect the Crystal/Ceramic and C as near as possible to the XIN/XOUT/VSS pins of MCU.

- **Structure:** 4MHz~25MHz EHS external crystal/ceramic resonator
- **Main Purpose:** System high clock source, RTC clock source, and PLL clock source.
- **Warm-up Time:** 2048*F_{EHS}
- **XIN/XOUT Shared Pin Selection:**

Oscillator Mode	XIN pin	XOUT pin
IHRC	GPIO	GPIO
EHS X'TAL	Crystal/Ceramic	Crystal/Ceramic

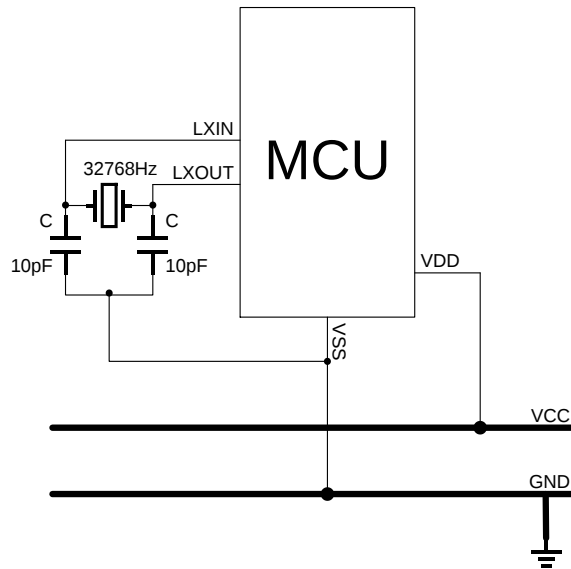
The resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and startup stabilization time. The loading capacitance values must be adjusted according to the selected oscillator.

The EHS crystal is switched on and off using the EHSEN bit in [Analog Block Control register \(SYS0_ANBCTRL\)](#).

3.2.3 External Low-speed (ELS) Clock

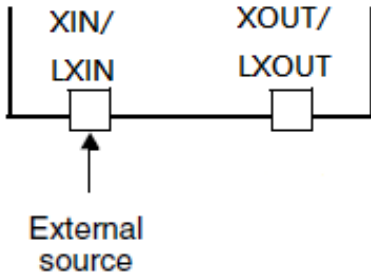
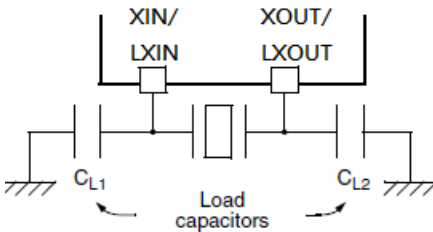
The low-speed oscillator can use 32768Hz crystal oscillator circuit.

Crystal devices are driven by LXIN, LXOUT pins. The 32768 crystal and 10pF capacitor must be as near as possible to MCU. The ELS crystal is switched on and off using the ELSSEN bit in [Analog Block Control register \(SYS0_ANBCTRL\)](#).



* **Note:** Connect the Crystal/Ceramic and C as near as possible to the LXIN/LXOUT/VSS pins of MCU. The capacitor between LXIN/LXOUT and VSS must be 10pF.

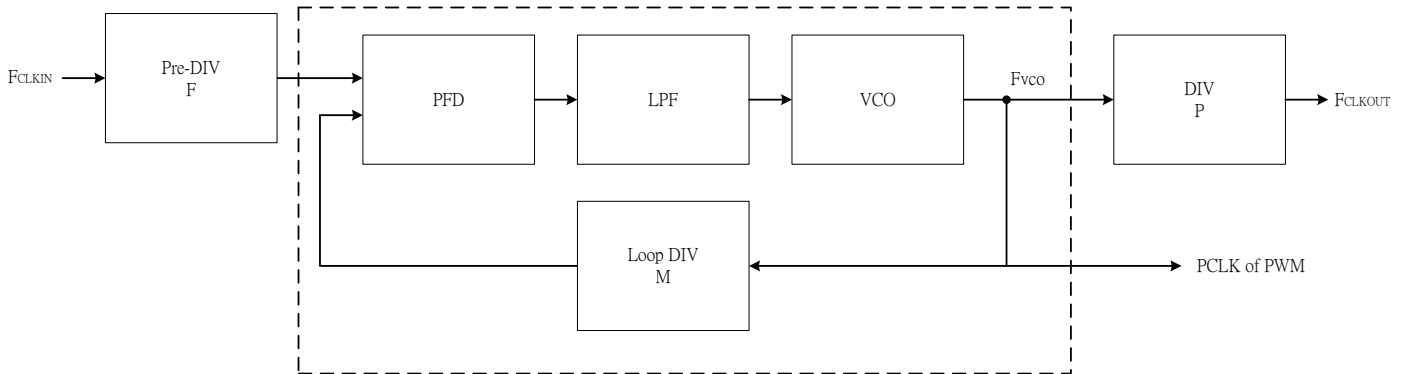
3.2.3.1 Bypass Mode

Clock Source	H/W Configuration	Description
External clock source (Bypass)		In Bypass mode, the external clock signal (square, sinus or triangle) with ~50% duty cycle must be provided to drive the XIN/LXIN pin while the XOUT/LXOUT pin should be the inverse of the input clock signal. EHS X'tal can have a frequency of up to 25 MHz. Select this mode by setting EHSEN bit in Analog Block Control register (SYS0_ANBCTRL). ELS X'TAL must have a frequency of 32.768 KHz. You select this mode by setting ELSEN bit in Analog Block Control register (SYS0_ANBCTRL).
External X'TAL (EHS/ELS X'TAL)		The 4 to 25 MHz EHS X'TAL has the advantage of producing a very accurate rate on the main clock ELS X'TAL must have a frequency of 32.768 KHz.

3.2.4 PLL

SONiX 32-bit MCU uses the PLL to create the clocks for the core and peripherals. The input frequency range is 4MHz to 25MHz. The input clock is divided down and fed to the Phase-Frequency Detector (PFD). This block compares the phase and frequency of its inputs, and generates a control signal when phase and/or frequency do not match. The loop filter filters these control signals and drives the voltage controlled oscillator (VCO), which generates the main clock and optionally two additional phases. The VCO frequency range is 96MHz to 120MHz. These clocks are divided by P by the programmable post divider to create the output clock(s). The VCO output clock is then divided by M by the programmable feedback divider to generate the feedback clock. The output signal of the phase-frequency detector is also monitored by the lock detector, to signal when the PLL has locked on to the input clock.

The PLL settling time is 100 μ s.



3.2.4.1 PLL Frequency selection

The PLL frequency equations:

$$F_{VCO} = F_{CLKIN} / F * M$$

$$F_{CLKOUT} = F_{VCO} / P$$

The PLL frequency is determined by the following parameters:

- F_{CLKIN} : Frequency from the PLLCLKSEL multiplexer.
- F_{VCO} : Frequency of the Voltage Controlled Oscillator (VCO); 96 to 120 MHz.
- F_{CLKOUT} : Frequency of PLL output.
- P: System PLL post divider ratio, controlled by PSEL bits in [PLL control register \(SYS0_PLLCTRL\)](#).
- F: System PLL front divider ratio, controlled by FSEL bits in [PLL control register \(SYS0_PLLCTRL\)](#).
- M: System PLL feedback divider ratio, controlled by MSEL bits in [PLL control register \(SYS0_PLLCTRL\)](#).

To select the appropriate values for M, P, and F, it is recommended to follow these constraints:

1. $4\text{MHz} \leq F_{CLKIN} \leq 25\text{MHz}$
2. $4\text{MHz} \leq (F_{CLKIN} / F) \leq 16\text{MHz}$
3. $96\text{MHz} \leq F_{VCO} \leq 120\text{MHz}$
4. $1 \leq F \leq 3$
5. $2 \leq M \leq 31$
6. $P = 2 \text{ or } 4$
7. $F_{CLKOUT} = 30\text{MHz}, 60\text{MHz}, 50\text{MHz}, 24\text{MHz}, 48\text{MHz}$ with jitter $< \pm 500 \text{ ps}$

3.2.4.2 PLL Recommended Frequency settings

F_{CLKIN} (MHz)	$F[1:0]$ 1~3	F_{CLKIN} / F 4~16 MHz	$M[4:0]$ 2~31	$F_{VCO}(MHz) = F_{CLKIN} / F * M$ 96 ~ 120 MHz	PSEL	P	$F_{CLKOUT}(MHz) = F_{VCO} / P$
4	1	4	30	120	0	2	60
4	1	4	30	120	1	4	30
10	1	10	12	120	0	2	60
10	1	10	10	100	0	2	50
10	1	10	12	120	1	4	30
12	1	12	8	96	0	2	48
12	1	12	10	120	0	2	60
16	1	16	6	96	0	2	48
25	2	12.5	8	100	0	2	50

3.2.5 SYSTEM CLOCK (SYSCLK) SELECTION

After a system reset, the IHRC is selected as system clock. When a clock source is used directly or through the PLL as system clock, it is not possible to stop it.

A switch from one clock source to another occurs only if the target clock source is ready (clock stable after startup delay or PLL locked). If a clock source which is not yet ready is selected, the switch will occur when the clock source is ready.

Ready bits in [SYS0_CSST](#) register indicate which clock(s) is (are) ready and SYSCLKST bits in [SYS0_CLKCFG](#) register indicate which clock is currently used as system clock.

3.2.6 CLOCK-OUT CAPABILITY

The MCU clock output (CLKOUT) capability allows the clock to be output onto the external CLKOUT pin. The configuration registers of the corresponding GPIO port must be programmed in alternate function mode.

One of 6 clock signals can be selected as clock output:

1. HCLK
2. IHRC
3. ILRC
4. PLL clock output
5. ELS X'TAL
6. EHS X'TAL

The selection is controlled by the CLKOUTSEL bits in [SYS1_AHBCLKEN](#) register.

3.3 SYSTEM CONTROL REGISTERS 0

Base Address: 0x4006 0000

3.3.1 Analog Block Control register (SYS0_ANBCTRL)

Address Offset: 0x00

Reset value: 0x0000 0001

* **Note:** *EHSEN / ELSEN / IHRCEN bit can NOT be cleared if the EHS X'tal / ELS X'tal / IHRC is selected as system clock or is selected to become the system clock.*

Bit	Name	Description	Attribute	Reset
31:17	Reserved		R	0
16	IHRCDIV	IHRC clock prescaler 0: IHRC48M / 4 1: IHRC48M / 1	R/W	0
15:6	Reserved		R	0
5	EHSFREQ	Frequency range (driving ability) of EHS X'TAL 0: <=8MHz 1: >8MHz	R/W	0
4	EHSEN	External high-speed clock enable 0: Disable EHS X'TAL. 1: Enable EHS X'TAL.	R/W	0
3	Reserved		R	0
2	ELSEN	External low-speed oscillator enable 0: Disable External 32.768 KHz oscillator 1: Enable External 32.768 KHz oscillator	R/W	0
1	Reserved		R	0
0	IHRCEN	Internal high-speed clock enable 0: Disable internal 48 MHz RC oscillator. 1: Enable internal 48 MHz RC oscillator.	R/W	1

3.3.2 PLL control register (SYS0_PLLCTRL)

Address Offset: 0x04

* **Note:** *PLLEN bit can NOT be cleared if the PLL is selected as system clock or is selected to become the system clock.*

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15	PLLEN	PLL enable 0: Disable 1: Enable	R/W	0
14:13	Reserved		R	0
12	PLLCLKSEL	System PLL clock source 0: IHRC 1: EHS X'TAL 4MHz~25MHz	R/W	0
11:10	Reserved		R	0

9:8	FSEL[1:0]	Front divider value F: 1~3	R/W	01b
7:6	Reserved		R	0
5	PSEL	Post divider value. 0: P=2 1: P=4	R/W	0
4:0	MSEL[4:0]	Loop divider value. M: 2~31	R/W	01000b

3.3.2.1 RECOMMEND FREQUENCY SETTING

$$F_{VCO} = F_{CLKIN} / F * M$$

$$F_{CLKOUT} = F_{VCO} / P$$

F_{CLKIN} (MHz)	F[1:0] 1~3	F_{CLKIN} / F 4~16 MHz	M[4:0] 2~31	F_{VCO}(MHz)= F_{CLKIN} / F * M 96 ~ 120 MHz	PSEL	P	F_{CLKOUT}(MHz)= F_{VCO} / P
4	1	4	30	120	0	2	60
4	1	4	30	120	1	4	30
10	1	10	12	120	0	2	60
10	1	10	10	100	0	2	50
10	1	10	12	120	1	4	30
12	1	12	8	96	0	2	48
12	1	12	10	120	0	2	60
16	1	16	6	96	0	2	48
25	2	12.5	8	100	0	2	50

3.3.3 Clock Source Status register (SYS0_CSST)

Address Offset: 0x08

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	PLLRDY	PLL clock ready flag 0: PLL unlocked 1: PLL locked	R	0
5	Reserved		R	0
4	EHSRDY	External high-speed clock ready flag 0: EHS oscillator not ready 1: EHS oscillator ready	R	0
3	Reserved		R	0
2	ELSRDY	External low-speed clock ready flag 0: ELS oscillator not ready 1: ELS oscillator ready	R	0
1	Reserved		R	0
0	IHRCDY	IHRC ready flag 0: IHRC not ready 1: IHRC ready	R	1

3.3.4 System Clock Configuration register (SYS0_CLKCFG)

Address Offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6:4	SYSCCLKST[2:0]	System clock switch status Set and cleared by HW to indicate which clock source is used as system clock. 000: IHRC is used as system clock 001: ILRC is used as system clock 010: EHS X'TAL is used as system clock 011: ELS X'TAL is used as system clock 100: PLL is used as system clock Other: Reserved	R	0
3	Reserved		R	0
2:0	SYSCCLKSEL[2:0]	System clock switch Set and cleared by SW. 000: IHRC 001: ILRC 010: EHS X'TAL 011: ELS X'TAL 100: PLL output Other: Reserved	R/W	0

3.3.5 AHB Clock Prescale register (SYS0_AHBCP)

Address Offset: 0x10

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	DIV1P5	SYSCCLK prescaler 0: SYSCCLK = SYSCCLK clock source / 1 1: SYSCCLK = SYSCCLK clock source / 1.5	R/W	0
2:0	AHBPRES[3:0]	AHB clock source prescale value 000: SYSCCLK / 1 001: SYSCCLK / 2 010: SYSCCLK / 4 011: SYSCCLK / 8 100: SYSCCLK / 16 101: SYSCCLK / 32 110: SYSCCLK / 64 111: SYSCCLK / 128	R/W	0

3.3.6 System Reset Status register (SYS0_RSTST)

Address Offset: 0x14

This register contains the reset source except DPDWAKEUP reset, since the MODE bits in [PMU_CTRL](#) register had presented this case.

Bit	Name	Description	Attribute	Reset
31:5	Reserved		R	0
4	PORRSTF	POR reset flag Set by HW when a POR reset occurs. 0: Read→No POR reset occurred Write→Clear this bit 1: POR reset occurred.	R/W	1
3	EXTRSTF	External reset flag Set by HW when a reset from the <u>RESET</u> pin occurs. 0: Read→No reset from RESET pin occurred Write→Clear this bit 1: Reset from RESET pin occurred.	R/W	0

2	LVDRSTF	LVD reset flag Set by HW when a LVD reset occurs. 0: Read→No LVD reset occurred Write→Clear this bit 1: LVD reset occurred.	R/W	0
1	WDTRSTF	WDT reset flag Set by HW when a WDT reset occurs. 0: Read→No watchdog reset occurred Write→Clear this bit 1: Watchdog reset occurred.	R/W	0
0	SWRSTF	Software reset flag Set by HW when a software reset occurs. 0: Read→No software reset occurred Write→Clear this bit 1: Software reset occurred.	R/W	1

3.3.7 LVD Control register (SYS0_LVDCTRL)

Address Offset: 0x18

The LVD control register selects six separate threshold values for generating a LVD interrupt to the NVIC or LVD reset.

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15	LV DEN	LVD enable 1: Disable 0: Enable	R/W	0
14	LVDRSTEN	LVD Reset enable 1: Disable 0: Enable	R/W	0
13:7	Reserved		R	0
6:4	LVDINTLVL[2:0]	LVD interrupt level 000: The interrupt assertion threshold voltage is 2.80V 001: The interrupt assertion threshold voltage is 2.60V 010: The interrupt assertion threshold voltage is 2.70V 011: Reserved for booting 100: The interrupt assertion threshold voltage is 2.90V 101: The interrupt assertion threshold voltage is 3.00V Other: Reserved	R/W	011b
3	Reserved		R	0
2:0	LVDRSTLVL[2:0]	LVD reset level 000: The reset assertion threshold voltage is 2.80V 001: The reset assertion threshold voltage is 2.60V 010: The reset assertion threshold voltage is 2.70V 011: Reserved for booting 100: The reset assertion threshold voltage is 2.90V 101: The reset assertion threshold voltage is 3.00V Other: Reserved	R/W	011b

3.3.8 External RESET Pin Control register (SYS0_EXRSTCTRL)

Address Offset: 0x1C

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	RESETDIS	External RESET pin disable bit. 0: Enable external <u>RESET</u> pin. (P3.6 acts as <u>RESET</u> pin) 1: Disable. (P3.6 acts as GPIO pin)	R/W	1

3.3.9 SWD Pin Control register (SYS0_SWDCTRL)

Address Offset: 0x20

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	SWDDIS	SWD pin disable bit. 0: Enable SWD pin. (P0.11 acts as SWDIO pin, P0.10 acts as SWCLK pin) 1: Disable. (P0.11 and P0.10 act as GPIO pins)	R/W	0

3.3.10 Interrupt Vector Table Mapping register (SYS0_IVTM)

Address Offset: 0x24

This register decides whether the ARM interrupt vector table is mapping to Boot ROM or User ROM

Bit	Name	Description	Attribute	Reset
31:16	IVTMKEY[15:0]	IVTM register key. Read as 0. Behaviour of writing to this register is ignored unless writing 0xA5A5 to IVTMKEY at the same time.	W	0
15:2	Reserved		R	0
1:0	IVTM[1:0]	Interrupt table mapping selection 00: Map to Boot ROM 01: Map to User ROM Other: Reserved	R/W	By BLEN in code option

3.3.11 Anti-EFT Ability Control register (SYS0_ANTIEFT)

Address Offset: 0x30

This register decides the HW anti-EFT ability.

Bit	Name	Description	Attribute	Reset
31:3	Reserved		R	0
2:0	AEFT[2:0]	HW anti-EFT ability. 000: No 010: Low 011: Medium 100: Strong	R/W	000

3.3.12 IHRC Frequency Adjustment register (SYS0_IHRCADJ)

Address Offset: 0x34

Total 32 counts available for IHRC frequency adjusting and each step varies between +1% from -1%.

Bit	Name	Description	Attribute	Reset
31:16	SYSKEY[15:0]	System register key Read as 0. Behaviour of writing to this register is ignored unless writing 0xA5A5 to SYSKEY at the same time.	W	0
15:9	Reserved		R	0
8:4	ADJ[4:0]	IHRC frequency adjusting bits.	R/W	0
3:2	Reserved		R	0

1	DIR	IHRC frequency adjusting direction bit 0: Positive 1: Negative	R/W	0
0	ADJEN	IHRC frequency adjustment enable bit 0: Disable 1: Enable	R/W	0

3.4 SYSTEM CONTROL REGISTERS 1

Base Address: 0x4005 E000

3.4.1 AHB Clock Enable register (SYS1_AHBCLKEN)

Address Offset: 0x00

The SYS_AHBCLKEN register enables the AHB clock to individual system and peripheral blocks.

*** Note:**

1. When the clock is disabled, the peripheral register values may not be readable by SW and the value returned is always 0x0.
2. HW will replace GPIO with CLKOUT function directly if CLKOUTSEL is Not 0.
3. User shall disable the AHB clock for individual peripheral to decrease power consumption by demand.

Bit	Name	Description	Attribute	Reset
31	DACCLKEN	Enable clock for DAC. 0: Disable 1: Enable	R/W	0
30:28	CLKOUTSEL[2:0]	Clock output source 000: Disable 001: ILRC clock 010: ELS clock 011: Reserved 100: HCLK 101: IHRC clock 110: EHS clock 111: PLL clock output	R/W	0
27	CRCCLKEN	Enable clock for CRC. 0: Disable 1: Enable	R/W	0
26:25	Reserved		R	0
24	WDTCLKEN	Enable clock for WDT. 0: Reserved 1: Enable	R	1
23	RTCCLKEN	Enable clock for RTC. 0: Disable 1: Enable	R/W	0
22	Reserved		R	0
21	I2C0CLKEN	Enable clock for I2C0. 0: Disable 1: Enable	R/W	0
20	I2C1CLKEN	Enable clock for I2C1. 0: Disable 1: Enable	R/W	0
19:18	Reserved		R	0
17	UART1CLKEN	Enable clock for UART1. 0: Disable 1: Enable	R/W	0
16	UART0CLKEN	Enable clock for UART0. 0: Disable 1: Enable	R/W	0
15	Reserved		R	0

14	CMPCLKEN	Enable clock for CMP. 0: Disable 1: Enable	R/W	0
13	Reserved		R	0
12	SPI0CLKEN	Enable clock for SPI0. 0: Disable 1: Enable	R/W	0
11	ADCCLKEN	Enable clock for ADC. 0: Disable 1: Enable	R/W	0
10	CT16B2CLKEN	Enable clock for CT16B2. 0: Disable 1: Enable	R/W	0
9	CT16B5CLKEN	Enable clock for CT16B5. 0: Disable 1: Enable	R/W	0
8	Reserved		R	0
7	CT16B1CLKEN	Enable clock for CT16B1. 0: Disable 1: Enable	R/W	0
6	CT16B0CLKEN	Enable clock for CT16B0. 0: Disable 1: Enable	R/W	0
5:0	Reserved		R	0

3.4.2 APB Clock Prescale register 0 (SYS1_APB0P0)

Address Offset: 0x04

* Note: Must reset the corresponding peripheral with SYS1_PRST register after changing the prescale value.				
--	--	--	--	--

Bit	Name	Description	Attribute	Reset
31:19	Reserved		R	0
18:16	ADCPRE[2:0]	ADC clock source prescaler 000: ADC_PCLK = HCLK / 1 001: ADC_PCLK = HCLK / 2 010: ADC_PCLK = HCLK / 4 011: ADC_PCLK = HCLK / 8 100: ADC_PCLK = HCLK / 16 Other: Reserved	R/W	0
15:0	Reserved		R	0

3.4.3 APB Clock Prescale register 1 (SYS1_APB0P1)

Address Offset: 0x08

* Note: Must reset the corresponding peripheral with SYS1_PRST register after changing the prescale value.				
--	--	--	--	--

Bit	Name	Description	Attribute	Reset
31	Reserved		R	0
30:28	CLKOUTPRE[2:0]	Clock-out source prescaler 000: Clock-out source / 1 001: Clock-out source / 2 010: Clock-out source / 4 011: Clock-out source / 8 100: Clock-out source / 16 101: Clock-out source / 32 110: Clock-out source / 64 111: Clock-out source / 128	R/W	0
27	Reserved		R	0
26:24	I2C1PRE[2:0]	I2C1 clock source prescaler 000: HCLK / 1 001: HCLK / 2 010: HCLK / 4 011: HCLK / 8 100: HCLK / 16 Other: Reserved	R/W	0
23	Reserved		R	0
22:20	WDTPRE[2:0]	WDT clock source prescaler 000: WDT_PCLK = WDT clock source / 1 001: WDT_PCLK = WDT clock source / 2 010: WDT_PCLK = WDT clock source / 4 011: WDT_PCLK = WDT clock source / 8 100: WDT_PCLK = WDT clock source / 16 101: WDT_PCLK = WDT clock source / 32 Other: Reserved	R/W	0
19:11	Reserved		R	0
10:8	I2C0PRE[2:0]	I2C0 clock source prescaler 000: HCLK / 1 001: HCLK / 2 010: HCLK / 4 011: HCLK / 8 100: HCLK / 16 Other: Reserved	R/W	0
7:0	Reserved		R	0

3.4.4 Peripheral Reset register (SYS1_PRST)

Address Offset: 0x10

All bits are cleared by HW automatically after setting as “1”.

Bit	Name	Description	Attribute	Reset
31:30	Reserved		R	0
29	DACRST	DAC reset 0: No effect 1: Reset DAC	R/W	0
28:27	Reserved		R	0
26	CRCRST	CRC reset 0: No effect 1: Reset CRC	R/W	0
25	Reserved		R	0
24	WDTRST	WDT reset 0: No effect	R/W	0

		1: Reset WDT		
23	RTCRST	RTC reset 0: No effect 1: Reset RTC	R/W	0
22	Reserved		R	0
21	I2C0RST	I2C0 reset 0: No effect 1: Reset I2C0	R/W	0
20	I2C1RST	I2C1 reset 0: No effect 1: Reset I2C1	R/W	0
19:18	Reserved		R	0
17	UART1RST	UART1 reset 0: No effect 1: Reset UART1	R/W	0
16	UART0RST	UART0 reset 0: No effect 1: Reset UART0	R/W	0
15	Reserved		R	0
14	CMPRST	CMP reset 0: No effect 1: Reset CMP	R/W	0
13	Reserved		R	0
12	SPI0RST	SPI0 reset 0: No effect 1: Reset SPI0	R/W	0
11	ADCRST	ADC reset 0: No effect 1: Reset ADC	R/W	0
10	Reserved		R	0
9	CT16B5RST	CT16B5 reset 0: No effect 1: Reset CT16B5	R/W	0
8	CT16B2RST	CT16B2 reset 0: No effect 1: Reset CT16B2	R/W	0
7	CT16B1RST	CT16B1 reset 0: No effect 1: Reset CT16B1	R/W	0
6	CT16B0RST	CT16B0 reset 0: No effect 1: Reset CT16B0	R/W	0
5:4	Reserved		R	0
3	GPIOP3RST	GPIO port 3 reset 0: No effect 1: Reset GPIO port 3	R/W	0
2	GPIOP2RST	GPIO port 2 reset 0: No effect 1: Reset GPIO port 2	R/W	0
1	GPIOP1RST	GPIO port 1 reset 0: No effect 1: Reset GPIO port 1	R/W	0
0	GPIOP0RST	GPIO port 0 reset 0: No effect 1: Reset GPIO port 0	R/W	0

4 SYSTEM OPERATION MODE

4.1 OVERVIEW

The chip builds in 3 operating modes for difference clock rate and power saving reason. These modes control oscillators, op-code operation and analog peripheral devices' operation.

- Normal mode
- Sleep mode
- Deep sleep mode

4.2 NORMAL MODE

In Normal mode, the ARM Cortex-M0 core, memories, and peripherals are clocked by the system clock. The [SYS1_AHBCLKEN](#) register controls which peripherals are running.

Selected peripherals have individual peripheral clocks with their own clock dividers in addition to the system clock. The peripheral clocks can be disabled respectively.

The power to various analog blocks (IHRC, EHS X'TAL, ELS X'TAL, PLL, Flash, LVD, ADC) can be controlled at any time individually through the enable bit of all blocks.

4.3 LOW-POWER MODES

There are 2 special modes of processor power reduction: Sleep mode, and Deep-sleep mode. The [PMU_CTRL](#) register controls which mode is desired.

The CPU clock rate may also be controlled as needed by changing clock sources, re-configuring PLL values, and/or altering the system clock divider value. This allows a trade-off of power versus processing speed based on application requirements.

Run-time power control allows disable the clocks to individual on-chip peripherals, allowing fine tuning of power consumption by eliminating all dynamic power use in any peripherals that are not required for the application. Selected peripherals have their own clock divider for power control.

*** Note:**

1. *The debug mode is not supported in Deep-sleep mode.*
2. *The pins which are not pin-out shall be set correctly to decrease power consumption in low-power modes. Strongly recommended to set these pins as input pull-up.*

4.3.1 SLEEP MODE

In Sleep mode, the system clock to the ARM Cortex-M0 core is stopped and execution of instructions is suspended.

Peripheral functions, if selected to be clocked in [SYS1_AHBCLKEN](#) register, continue operation during Sleep mode and may generate interrupts to cause the processor to resume execution. Sleep mode eliminates dynamic power used by the processor itself, memory systems and related controllers, and internal buses.

The power state of the analog blocks (IHRC, EHS X'TAL, ELS X'TAL, PLL, Flash, LVD, ADC) is determined by the enable bit of all blocks.

The processor state and registers, peripheral registers, and internal SRAM values are maintained and the logic levels of the pins remain static.

The pins of port 1 and port 2 configured as ADC input channel must be set as input mode, inactive (no pull-down/pull-up resistor enabled, Schmitt trigger disabled, Data register keep low) with [GPIO1_MODE](#), [GPIO1_CFG](#), [GPIO2_MODE](#) and [GPIO2_CFG](#) register by program to avoid current leakage.

Wake up the chip from Sleep mode by an interrupt occurs.

The RESET pin has keep functionality in Sleep mode.

The Sleep mode is entered by using the following steps:

1. Write 4 to [PMU_CTRL](#) register.
2. Execute ARM Cortex-M0 WFI instruction.

4.3.2 DEEP-SLEEP MODE

In Deep-sleep mode, the system clock to the ARM Cortex-M0 core is stopped, and execution of instructions is suspended.

The clock to the peripheral functions are stopped because the power state of oscillators are powered down, the clock source are stopped, except RTC low speed clock source (ELS X'tal, ILRC) if used.

- **Note: User SHALL decide to power down low speed clock source (ELS X'TAL, ILRC oscillator) or not if RTC is enabled.**

The processor state and registers, peripheral registers, and internal SRAM values are maintained and the logic levels of the pins remain static.

All GPIO pins are served as wakeup pins. The user must program the GPIO registers for each pin to set the appropriate edge polarity for the corresponding wakeup event, only edge sensitive is supported to wakeup MCU. The system will exit Deep-sleep mode when GPIO indicates a GPIO interrupt to the ARM core. Furthermore, the interrupts corresponding to each input must be enabled in the NVIC.

- **Note: Strongly recommended to set the GPIO wake-up condition to both edge sensing.**

The RESET pin has keep functionality in Deep-sleep mode.

The Deep-sleep mode is entered by using the following steps:

1. Write 2 to [PMU_CTRL](#) register.
2. Execute ARM WFI instruction.

The advantage of the Deep-sleep mode is that can power down clock generating blocks such as oscillators and PLL, thereby gaining far greater dynamic power savings over Sleep mode. In addition, the Flash can be powered down in Deep-sleep mode resulting in savings in static leakage power, however at the expense of longer wake-up times for the Flash memory.

4.4 WAKEUP

4.4.1 OVERVIEW

Under low power mode, program doesn't execute. The wakeup trigger can wake the system up to normal mode. The wakeup function builds in interrupt operation and trigger system executing interrupt service routine as system wakeup occurrence.

- * The wakeup trigger sources of the Sleep mode are all interrupts and the RESET pin.
- * The wakeup trigger sources of the Deep-sleep mode are the GPIO interrupt, RTC interrupt, CT16B5 interrupt, and the RESET pin.

4.4.2 WAKEUP TIME

When the system is in Sleep mode, the high clock is enabled or disabled by F/W. If the high clock stops and MCU is waken up from Sleep mode, MCU waits for 2048 external high-speed oscillator clocks and 32 internal high-speed oscillator clocks as the wakeup time to stable the oscillator circuit. After the wakeup time, the system goes into the normal mode.

* **Note: Wakeup from Sleep mode spends NO wakeup time if the clock doesn't stop.**

When the system is in Deep-sleep mode, the high clock will stop. When MCU is waken up from Deep-sleep mode, MCU waits for 10us and IHRC/EHS warm up time ($32T_{IHRC}/2048T_{EHS}$). After the wakeup time, the system goes into the normal mode.

The value of the external high clock oscillator wakeup time from Deep sleep mode is as the following.

The total Wakeup time of EHS X'tal = 10us + $1/F_{EHS} * 2048$ (sec) + high clock start-up time

- Example: $F_{EHS}=20\text{MHz}$, the wakeup time is as the following.

$$\begin{aligned} \text{The total Wakeup time} &= 10\mu\text{s} + 1/F_{EHS} * 2048 + \text{oscillator start-up time} \\ &= 112.4 \mu\text{s} + \text{oscillator start-up time} \quad (F_{EHS} = 20\text{MHz}) \end{aligned}$$

The value of the IHRC wakeup time from Deep sleep mode is as the following.

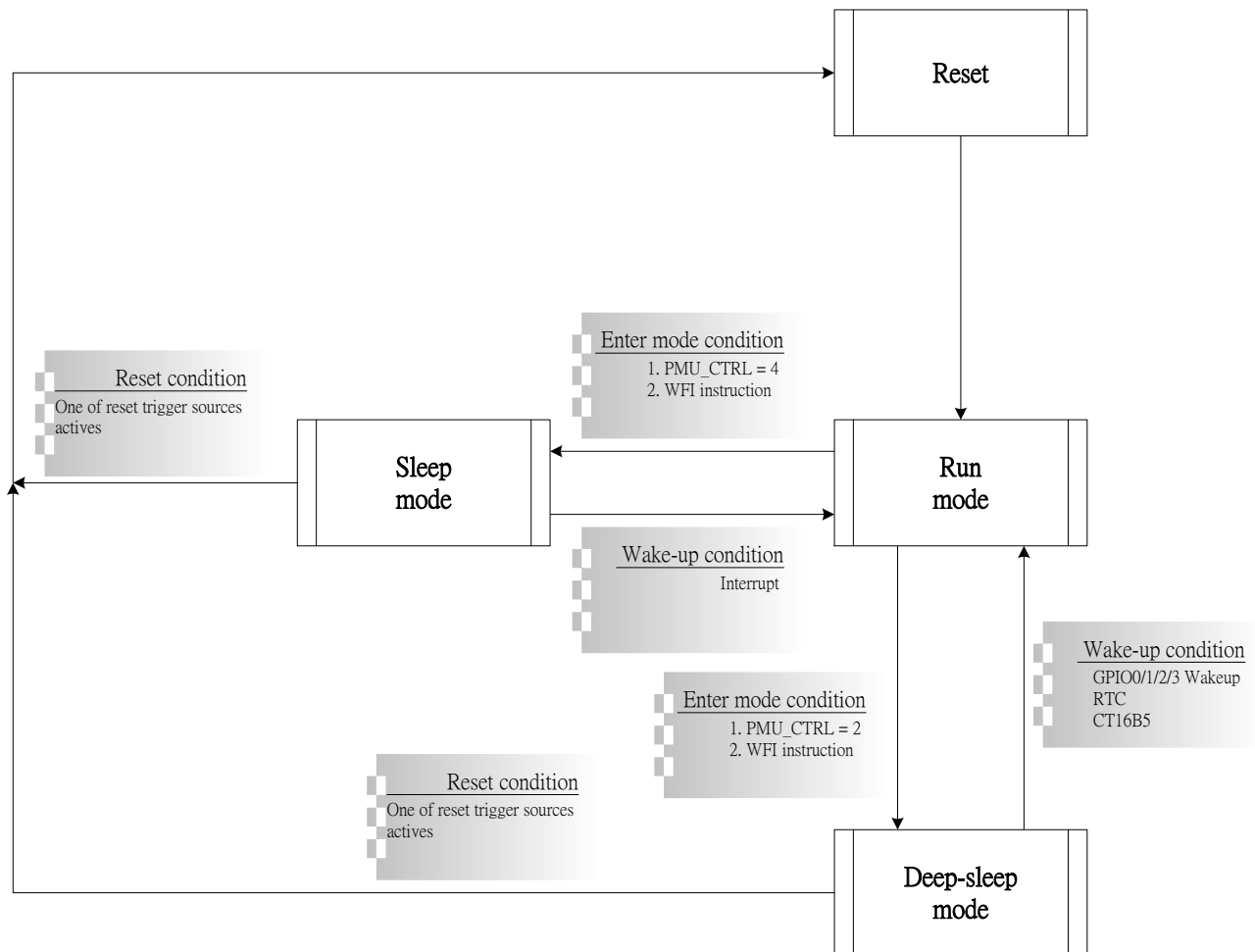
The total Wakeup time of IHRC = 10us + $1/F_{IHRC} * 32$ (sec)

- Example: $F_{IHRC}=12\text{MHz}$, the wakeup time is as the following.

$$\text{The total Wakeup time} = 10\mu\text{s} + 1/F_{IHRC} * 32 = 12.67 \mu\text{s} \quad (F_{IHRC} = 12\text{MHz})$$

* **Note: The high clock start-up time is depended on the VDD and oscillator type of high clock.**

4.5 STATE MACHINE OF PMU



4.6 OPERATION MODE COMPARSION TABLE

Operation Mode	Normal Mode	Low-Power Mode	
		Sleep Mode	Deep-Sleep Mode
IHRC	By IHRCEN		Disable
ILRC	ON		Enable if used as clock source of RTC
EHS X'TAL	By EHSEN		Disable
ELS X'TAL	By ELSEN		Enable if used as clock source of RTC/CT16B5
PLL	By PLEN		Disable
Cortex-M0	Running	Stop	Stop
Flash ROM	Enable	Disable	Disable
RAM	Enable	Maintain	Maintain
ADC	By ADENB		Disable
DAC	By DACEN		Disable
LVD	By LVDEN		Disable
RTC	By RTCEN		By RTCEN
CMP	By CMnEN & CMnMODE		Disable
OPA	By OPnEN		Disable
Peripherals	By Enable bit of each peripherals		Disable HCLK
IO status	-	Maintained	Maintained
Wakeup Source	N/A	All interrupts, RESET pin	GPIO0/1/2/3 interrupt, CT16B5 interrupt, RTC interrupt, RESET pin

4.7 PMU REGISTERS

Base Address: 0x4003 2000

4.7.1 Power Control register (PMU_CTRL)

Address Offset: 0x40

The power control register selects whether one of the ARM Cortex-M0 controlled power-down modes (Sleep mode or Deep-sleep mode) is entered and provides the flags for Sleep or Deep-sleep modes respectively.

*** Note: The pins which are not pin-out shall be set correctly to decrease power consumption in low-power modes. Strongly recommended to set these pins as input pull-up.**

Bit	Name	Description	Attribute	Reset
31:3	Reserved		R	0
2:0	MODE[2:0]	Low power mode selection 010: WFI instruction will make MCU enter Deep-sleep mode. 100: WFI instruction will make MCU enter Sleep mode. Other: Normal mode	R/W	0

5 GENERAL PURPOSE I/O PORT (GPIO)

5.1 OVERVIEW

Digital ports can be configured input/output by SW

- Each individual port pin can serve as external interrupt input pin.
- Interrupts can be configured on single falling or rising edges and on both edges.
- The I/O configuration registers control the electrical characteristics of the pads.
- Internal pull-up/pull-down resistor.
- Most of the I/O pins are mixed with analog pins and special function pins.

5.2 GPIO MODE

All GPIO pins are inputs and floating by default. The MODE bits in the [GPIO_n_CFG](#) (n=0,1,2,3) register allow the selection of on-chip pull-up or pull-down resistors for each pin.

5.3 GPIO REGISTERS

Base Address: 0x4004 4000 (GPIO 0)
 0x4004 6000 (GPIO 1)
 0x4004 8000 (GPIO 2)
 0x4004 A000 (GPIO 3)

5.3.1 GPIO Port n Data register (GPIOn_DATA) (n=0,1,2,3)

Address offset: 0x00

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	DATA[11:0]	Input data (read) or output data (write) for Pn.0 to Pn.11	R/W	0

5.3.2 GPIO Port n Mode register (GPIOn_MODE) (n=0,1,2,3)

Address offset: 0x04

*** Note: HW will switch I/O Mode directly when Specific function (Peripheral, ADC) is enabled, not through GPIOn_MODE register.**

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	MODE[11:0]	Selects pin x as input or output (x = 0 to 11) 0: Pn.x is configured as input 1: Pn.x is configured as output.	R/W	0

5.3.3 GPIO Port n Configuration register (GPIOn_CFG) (n=0,1,2,3)

Address offset: 0x08

Reset value: 0x00AAAAAA

*** Note: HW will switch I/O Mode directly when Specific function (Peripheral, ADC) is enabled, not through GPIOn_MODE register.**

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23:22	CFG11[1:0]	Configuration of Pn.11 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
21:20	CFG10[1:0]	Configuration of Pn.10 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled,	R/W	10b

		Data register keep low)		
19:18	CFG9[1:0]	Configuration of Pn.9 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
17:16	CFG8[1:0]	Configuration of Pn.8 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
15:14	CFG7[1:0]	Configuration of Pn.7 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
13:12	CFG6[1:0]	Configuration of Pn.6 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
11:10	CFG5[1:0]	Configuration of Pn.5 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
9:8	CFG4[1:0]	Configuration of Pn.4 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
7:6	CFG3[1:0]	Configuration of Pn.3 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
5:4	CFG2[1:0]	Configuration of Pn.2 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
3:2	CFG1[1:0]	Configuration of Pn.1 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled). 11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
1:0	CFG0[1:0]	Configuration of Pn.0 00: Pull-up resistor enabled. 01: Pull-down resistor enabled. 10: Inactive (no pull-down/up resistor enabled, Schmitt trigger enabled).	R/W	10b

		11: Inactive (no pull-down/up resistor enabled, Schmitt trigger disabled, Data register keep low)		
--	--	---	--	--

5.3.4 GPIO Port n Interrupt Sense register (GPIO_n_IS) (n=0,1,2,3)

Address offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	IS[11:0]	Selects interrupt on pin x as level or edge sensitive (x = 0 to 11). 0: Interrupt on Pn.x is configured as edge sensitive. 1: Interrupt on Pn.x is configured as event sensitive.	R/W	0

5.3.5 GPIO Port n Interrupt Both-edge Sense register (GPIO_n_IBS) (n=0,1,2,3)

Address offset: 0x10

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	IBS[11:0]	Selects interrupt on Pn.x to be triggered on both edges (x = 0 to 11). 0: Interrupt on Pn.x is controlled through register GPIO _n _IEV. 1: Both edges on Pn.x trigger an interrupt.	R/W	0

5.3.6 GPIO Port n Interrupt Event register (GPIO_n_IEV) (n=0,1,2,3)

Address offset: 0x14

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	IEV[11:0]	Selects interrupt on pin x to be triggered rising or falling edges (x = 0 to 11). 0: Depending on setting in register GPIO _n _IS, Rising edges or HIGH level on Pn.x trigger an interrupt. 1: Depending on setting in register GPIO _n _IS, Falling edges or LOW level on Pn.x trigger an interrupt.	R/W	0

5.3.7 GPIO Port n Interrupt Enable register (GPIO_n_IE) (n=0,1,2,3)

Address offset: 0x18

Bits set to HIGH in the GPIO_n_IE register allow the corresponding pins to trigger their individual interrupts. Clearing a bit disables interrupt triggering on that pin.

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	IE[11:0]	Selects interrupt on pin x to be enabled (x = 0 to 11). 0: Disable Interrupt on Pn.x 1: Enable Interrupt on Pn.x	R/W	0

5.3.8 GPIO Port n Raw Interrupt Status register (GPIO_n_RIS) (n=0,1,2,3)

Address offset: 0x1C

This register indicates the status for GPIO control raw interrupts. A GPIO interrupt is sent to the interrupt controller if the corresponding bit in GPIO_n_IE register is set.

Bit	Name	Description	Attribute	Reset
-----	------	-------------	-----------	-------

31:12	Reserved		R	0
11:0	IF[11:0]	GPIO raw interrupt flag (x = 0 to 11). 0: No interrupt on Pn.x 1: Interrupt requirements met on Pn.x.	R	0

5.3.9 GPIO Port n Interrupt Clear register (GPION_IC) (n=0,1,2,3)

Address offset: 0x20

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	IC[11:0]	Selects interrupt flag on pin x to be cleared (x = 0 to 11). 0: No effect 1: Clear interrupt flag on Pn.x	W	0

5.3.10 GPIO Port n Bits Set Operation register (GPION_BSET) (n=0,1,2,3)

Address offset: 0x24

In order for SW to set GPIO bits without affecting any other pins in a single write operation, the GPIO bit is set if the corresponding bit in the GPION_BSET register is set.

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	BSET[11:0]	Bit Set enable (x = 0 to 11) 0: No effect on Pn.x 1: Set Pn.x to "1"	W	0

5.3.11 GPIO Port n Bits Clear Operation register (GPION_BCLR) (n=0,1,2,3)

Address offset: 0x28

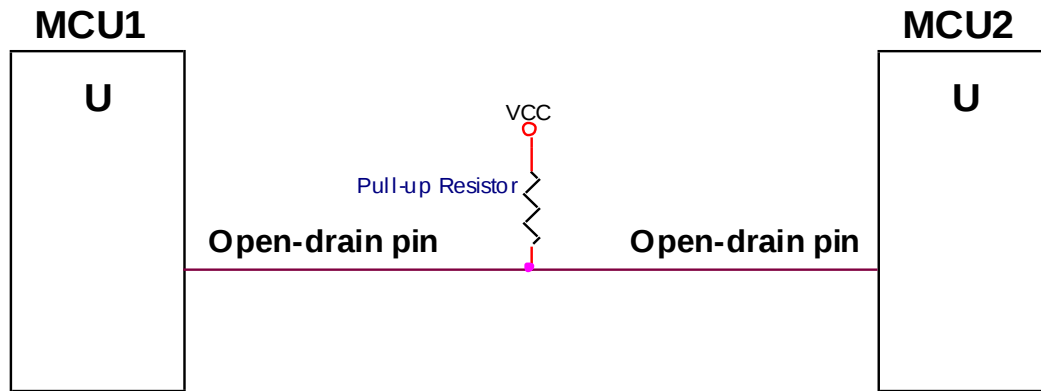
In order for SW to clear GPIO bits without affecting any other pins in a single write operation, the GPIO bit is cleared if the corresponding bit in this register is set.

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	BCLR[11:0]	Bit clear enable (x = 0 to 11) 0: No effect on Pn.x 1: Clear Pn.x.	W	0

5.3.12 GPIO Port n Open-Drain Control register (GPION_ODCTRL) (n=0,1,2,3)

Address offset: 0x2C

Several I/Os have built-in open-drain function and must be set as output mode when enable open-drain function. Open-drain external circuit is as following.



The external pull-up resistor is necessary. The digital output function of I/O only supports sink current capability, so the open-drain output high is driven by pull-up resistor, and output low is sunken by MCU's pin.

*** Note: VCC shall be less than or equal to VDD of MCU1 and MCU2.**

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11	Pn11OC	Pn.11 open-drain control bit (n=0) 0: Disable 1: Enable	RW	0
10	Pn10OC	Pn.10 open-drain control bit (n=0) 0: Disable 1: Enable	RW	0
9	Pn9OC	Pn.9 open-drain control bit (n=0,3) 0: Disable 1: Enable	RW	0
8	Pn8OC	Pn.8 open-drain control bit (n=0,3) 0: Disable 1: Enable	RW	0
7	Pn7OC	Pn.7 open-drain control bit (n=0,1,3) 0: Disable 1: Enable	RW	0
6	Pn6OC	Pn.6 open-drain control bit (n=0,1,2,3) 0: Disable 1: Enable	RW	0
5	Pn5OC	Pn.5 open-drain control bit (n=0,1,2,3) 0: Disable 1: Enable	RW	0
4	Pn4OC	Pn.4 open-drain control bit (n=0,1,2,3) 0: Disable 1: Enable	RW	0
3	Pn3OC	Pn.3 open-drain control bit (n=0,1,2) 0: Disable 1: Enable	RW	0
2	Pn2OC	Pn.2 open-drain control bit (n=0,1,2) 0: Disable 1: Enable	RW	0
1	Pn1OC	Pn.1 open-drain control bit (n=0,1,2). 0: Disable 1: Enable	RW	0
0	PnOC	Pn.0 open-drain control bit (n=0,1,2) 0: Disable 1: Enable	RW	0

6 PERIPHERAL FUNCTION PIN ASSIGNMENT (PFPA)

6.1 OVERVIEW

PFPA registers are used to provide flexible assignment of digital peripheral functions to desired external pins of different packages.

6.2 FEATURES

- Flexible assignment of digital peripheral functions to desired pins.
- Supported functions are UART, I2C, SPI, and PWM.

6.3 PIN ASSIGNMENT LIST

Peripheral	Pin Name	PA0	PA1		Peripheral	Pin Name	PA0	PA1
UART1	URXD1	P2.5			CT16B1	PWM0	P2.5	P1.7
UART1	UTXD1	P2.6			CT16B1	PWM1	P2.6	P3.6
SPI0	SCK0	P0.6	P3.8		CT16B1	PWM2	P1.0	P3.7
SPI0	SEL0	P0.7	P3.7		CT16B1	PWM3	P2.4	P3.8
SPI0	MISO0	P0.8	P3.6		CT16B2	PWM0	P1.6	P1.1
SPI0	MOSI0	P0.9	P3.4		CT16B2	PWM0N	P1.4	P1.0
I2C0	SCL0	P0.4	P1.2		CT16B2	PWM1	P1.7	P1.2
I2C0	SDA0	P0.5	P1.1		CT16B2	PWM1N	P3.4	P1.3
I2C1	SCL1	P0.2	P1.4		CT16B2	PWM2	P0.3	P1.7
I2C1	SDA1	P0.3	P1.3		CT16B2	PWM2N	P3.5	P0.3
CT16B0	PWM0	P0.8	P3.8		CT16B2	PWM3	P0.11	P1.6
CT16B0	PWM1	P0.9	P3.7		CT16B2	PWM3N	P3.6	P3.9
CT16B0	PWM2	P0.10	P0.7		CT16B5	PWM0	P1.1	P2.0
CT16B0	PWM3	P2.0	P0.6		CT16B5	PWM1	P1.2	P2.1
					CT16B5	PWM2	P1.3	P2.2
					CT16B5	PWM3	P1.4	P2.3

6.4 PFPA REGISTERS

Base Address: 0x4004 2000

6.4.1 PFPA for CT16B0 register (PFPA_CT16B0)

Address offset: 0x00

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	PWM3	Pin to be assigned as CT16B0_PWM3. 0: P2.0 1: P0.6	R/W	0
5	Reserved		R	0
4	PWM2	Pin to be assigned as CT16B0_PWM2. 0: P0.10 1: P0.7	R/W	0
3	Reserved		R	0
2	PWM1	Pin to be assigned as CT16B0_PWM1. 0: P0.9 1: P3.7	R/W	0
1	Reserved		R	0
0	PWM0	Pin to be assigned as CT16B0_PWM0. 0: P0.8 1: P3.8	R/W	0

6.4.2 PFPA for CT16B1 register (PFPA_CT16B1)

Address offset: 0x04

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	PWM3	Pin to be assigned as CT16B1_PWM3 0: P2.4 1: P3.8	R/W	0
5	Reserved		R	0
4	PWM2	Pin to be assigned as CT16B1_PWM2 0: P1.0 1: P3.7	R/W	0
3	Reserved		R	0
2	PWM1	Pin to be assigned as CT16B1_PWM1 0: P2.6 1: P3.6	R/W	0
1	Reserved		R	0
0	PWM0	Pin to be assigned as CT16B1_PWM0 0: P2.5 1: P1.7	R/W	0

6.4.3 PFPA for UART register (PFPA_UART)

Address offset: 0x08

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	URXD1	Pin to be assigned as URXD1 0: P2.5	RW	0
5	Reserved		R	0
4	UTXD1	Pin to be assigned as UTXD1 0: P2.6	RW	0
3:0	Reserved		R	0

6.4.4 PFPA for I2C register (PFPA_I2C)

Address offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	SCL1	Pin to be assigned as SCL1. 0: P0.2 1: P1.4	RW	0
5	Reserved		R	0
4	SDA1	Pin to be assigned as SDA1. 0: P0.3 1: P1.3	RW	0
3	Reserved		R	0
2	SCL0	Pin to be assigned as SCL0. 0: P0.4 1: P1.2	RW	0
1	Reserved		R	0
0	SDA0	Pin to be assigned as SDA0. 0: P0.5 1: P1.1	RW	0

6.4.5 PFPA for SPI register (PFPA_SPI)

Address offset: 0x10

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	SEL0	Pin to be assigned as SEL0. 0: P0.7 1: P3.7	RW	0
5	Reserved		R	0
4	SCK0	Pin to be assigned as SCK0. 0: P0.6 1: P3.8	RW	0
3	Reserved		R	0
2	MOSI0	Pin to be assigned as MOSI0. 0: P0.9 1: P3.4	RW	0
1	Reserved		R	0

0	MISO0	Pin to be assigned as MISO0. 0: P0.8 1: P3.6	RW	0
----------	--------------	--	----	---

6.4.6 PFPA for CT16B2 register (PFPA_CT16B2)

Address offset: 0x18

Bit	Name	Description	Attribute	Reset
31:15	Reserved		R	0
14	PWM3N	Pin to be assigned as CT16B2_PWM3N 0: P3.6 1: P3.9	RW	0
13	Reserved		R	0
12	PWM3	Pin to be assigned as CT16B2_PWM3 0: P0.11 1: P1.6	RW	0
11	Reserved		R	0
10	PWM2N	Pin to be assigned as CT16B2_PWM2N 0: P3.5 1: P0.3	RW	0
9	Reserved		R	0
8	PWM2	Pin to be assigned as CT16B2_PWM2 0: P0.3 1: P1.7	RW	0
7	Reserved		R	0
6	PWM1N	Pin to be assigned as CT16B2_PWM1N 0: P3.4 1: P1.3	RW	0
5	Reserved		R	0
4	PWM1	Pin to be assigned as CT16B2_PWM1 0: P1.7 1: P1.2	RW	0
3	Reserved		R	0
2	PWM0N	Pin to be assigned as CT16B2_PWM0N 0: P1.4 1: P1.0	RW	0
1	Reserved		R	0
0	PWM0	Pin to be assigned as CT16B2_PWM0 0: P1.6 1: P1.1	RW	0

6.4.7 PFPA for CT16B5 register (PFPA_CT16B5)

Address offset: 0x24

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	PWM3	Pin to be assigned as CT16B5_PWM3. 0: P1.4 1: P2.3	RW	0
5	Reserved		R	0

4	PWM2	Pin to be assigned as CT16B5_PWM2. 0: P1.3 1: P2.2	R/W	0
3	Reserved		R	0
2	PWM1	Pin to be assigned as CT16B5_PWM1. 0: P1.2 1: P2.1	R/W	0
1	Reserved		R	0
0	PWM0	Pin to be assigned as CT16B5_PWM0. 0: P1.1 1: P2.0	R/W	0

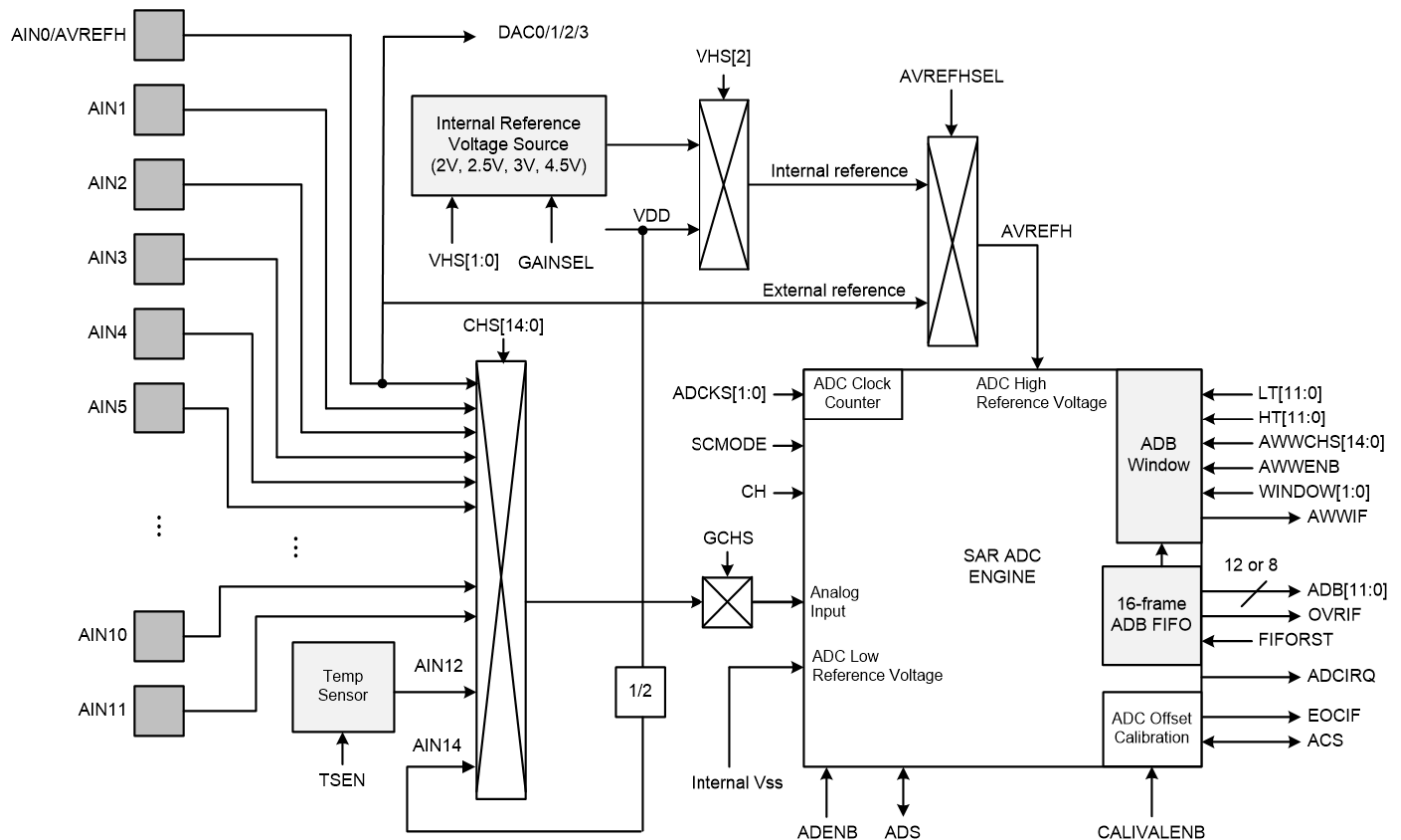
7 12+2 CHANNEL ANALOG TO DIGITAL CONVERTOR (ADC)

7.1 OVERVIEW

This analog to digital converter (ADC) has 12 external input sources, one internal channel to internal Temperature sensor and one internal channel to 1/2 VDD, with up to 4096-step resolution to transfer analog signal into 12-bits digital data. The ADC resolution can be selected 8-bit or 12-bit through ADLEN bit in ADR register. The ADC converting rate can be selected by ADCKS[1:0] bits. These two parameters decide the ADC converting time. The ADC is capable of maximum 1MSPS converting rate with 16MHz clock. The ADC also has built-in calibration circuit. The calibration cycle must be run at least once before starting A/D conversion. During calibration cycle, ADS bit is ignored.

The sequence of ADC operation is to select input source CHS[14:0] bits, CH bit, and SCMODE bit at first, then set GCHS and ADS bit to "1" to start conversion. When the conversion is complete, the ADC circuit will set EOC bit to "1" and final value output in ADB register. When ADC is enabled (ADENB=1) and global channel is enabled (GCHS=1), the ADC shared pins transfers to ADC purpose and disable GPIO function and disable pull-up/pull-down resistor by HW automatically. When ADC is disabled or global channel is disabled, the ADC pins returns to GPIO last status including pull-up/pull-down resistor. Use CHS[14:0] to select AIN pin and GCHS enables global ADC channel, the analog signal inputs to ADC engine.

The AIN12 is internal Temperature sensor. The AIN14 is 1/2 VDD channel, there is no any input pin from outside. AIN14 can be a good battery detector for battery system. To select appropriate internal AVREFH level and compare value, a high performance and cheaper low battery detector is built in the system.



* **Note:**

1. **ADC_PCLK shall be less than 16MHz.**
2. **The analog input level must be between the AVREFH and AVREFL.**
3. **The AVREFH level must be between the AVDD and AVREFL + 2.0V.**
4. **ADC programming notice**
 - **Set ADC input pins as input mode and inactive (no pull-down/ pull-up resistor enabled, Schmitt trigger disabled, Data register keep low) with GPIO1_MODE, GPIO1_CFG, GPIO2_MODE and GPIO2_CFG register by program**
 - **Disable ADC (set ADENB = "0") before enter low-power (Sleep/Deep-sleep) mode to save power consumption.**
 - **Delay 100us after enable ADC (set ADENB = "1") to wait ADC circuit ready for conversion.**
 - **The calibration cycle must be run at least once before starting A/D conversion. During calibration cycle, ADS bit is ignored.**

7.2 FEATURES

1. Sampling rate up to 1Msps
2. Internal voltage reference
3. 16-frame FIFO for ADB
4. Self-calibration.
5. Programmable sampling time
6. Support auto-conversion mode
7. ADB watchdog
8. Integrated factory-calibrated temperature sensor

7.3 ADC CONVERTING TIME

The ADC converting time is from ADS=1 (Start to ADC convert) to EOC=1 (End of ADC convert). The converting time duration is depend on ADC resolution and ADC clock rate.

ADC clock source is controlled by ADCKS[2:0] bits. The ADC converting time affects ADC performance. If input high rate analog signal, it is necessary to select a high ADC converting rate. If the ADC converting time is slower than analog signal variation rate, the ADC result would be error. So to select a correct ADC clock rate and ADC resolution to decide a right ADC converting rate is very important.

12-bit ADC conversion time = 1/ADC MLCK * 16 sec

ADCKS [2:0]	MCLK ADC Clock	ADC_PCLK = 12 MHz		ADC_PCLK = Max 16 MHz	
		ADC Conversion Time (us)	ADC Conversion Rate (KHz)	ADC Conversion Time (us)	ADC Conversion Rate (KHz)
000	ADC_PCLK	1.33	750	1	1000
001	ADC_PCLK/2	2.67	375	2	500
010	ADC_PCLK/4	5.33	187.5	4	250
011	ADC_PCLK/8	10.67	93.75	8	125
100	ADC_PCLK/16	21.33	46.88	16	62.5
101	ADC_PCLK/32	42.67	23.44	32	31.25

7.4 ADC OFFSET CALIBRATION

The calibration must be run at least one time before starting A/D conversion. The internal calibration register will keep the value even when ADC is disabled. It removes the offset error which may vary from chip to chip due to process variation. The calibration is initiated by software by setting ACS to 1. ACS bit remains 1 during all the calibration sequence. It is then cleared by hardware as soon the calibration completes. The internal analog calibration is kept even the ADC is disabled. When the ADC operating conditions change, it is recommended to re-run a calibration cycle.

7.5 ADC CONTROL NOTICE

7.5.1 ADC SIGNAL

The ADC high reference voltage includes internal VDD/4.5V/3V/2.5V/2V or external reference voltage source from P2.0/AVREFH pin controlled by AVREFHSEL bit. The ADC low reference voltage is ground.

ADC reference voltage range limitation is “(ADC high reference voltage – low reference voltage) $\geq$ 2V”. ADC low reference voltage is Vss = 0V. So **ADC high reference voltage range is 2V~Vdd**. The range is ADC external high reference voltage range.

- **ADC Internal Low Reference Voltage = 0V**
- **ADC Internal High Reference Voltage = VDD/4.5V/3V/2.5V/2V** (AVREFHSEL=0)
- **ADC External High Reference Voltage = 2V~VDD** (AVREFHSEL =1)

ADC sampled input signal voltage must be from ADC low reference voltage to ADC high reference. If the ADC input signal voltage is over the range, the ADC converting result is error (full scale or zero).

- **ADC Low Reference Voltage (VSS) $\leq$ ADC Sampled Input Voltage $\leq$ ADC High Reference Voltage**

7.5.2 ADC OVERRUN

The ADC has a 16-frame data buffer. The [ADC_ADB](#) register is the top layer of the ADC data FIFO, and contains the oldest A/D converter result.

The overrun flag (OVRIF) indicates a data overrun event, when the converted data was not read in time before the data from a new conversion is available. The OVRIF bit is set in the ADC_RIS register if the ADB FIFO remains unread data at the time when a new conversion completes.

The ADC interrupt can be generated if the OVRIE bit is set in ADC_IE register. When an overrun condition occurs, the ADC keeps operating and can continue to convert unless the software decides to stop and reset the sequence by setting the ADSTOP bit.

Configure whether the data is preserved or overwritten when an overrun event occurs by programming the OVRMOD bit in [ADC_ADM](#) register.

- **OVRMOD=0:** An overrun event preserves the data register from being overwritten, the old data is maintained, the new conversion is discarded.
- **OVRMOD=1:** The data register is overwritten with the last conversion result and the previous unread data is lost.

7.5.3 ADC PROGRAM

The first step of ADC execution is to setup ADC configuration. The ADC program setup sequence and notices are as following.

- **Step 1:** If the ADC high reference voltage is from external voltage source, set AVREFHSEL to 1. The ADC external high reference voltage inputs from P2.0 pin. It is necessary to set P2.0 as input mode without pull-up resistor.
- **Step 2:** Set ADENB to 1 to enable ADC. **When ADENB is enabled, the system must be delay 100us to be the ADC warm-up time by program, and then set ADS to do ADC converting. The 100us delay time is necessary after ADENB setting (not ADS setting), or the ADC converting result would be error.** Normally, the ADENB is set one time when the system under normal run condition, and do the delay time only one time.

- **Step 3:** Start ADC offset calibration by setting ACS = 1 and wait for ACS bit to be cleared. It means that the calibration is completed. Set CALIVALENB = 1 to enable offset calibration. The ADC must be calibrated once after MCU power on or the ADC converting result would be error.
- **Step 4:** Select the ADC input pins by CHS[14:0], set CH and SCMODE bits, and set ADC input pins as input mode and inactive (no pull-down/ pull-up resistor enabled, Schmitt trigger disabled, Data register keep low) with GPIO1_MODE, GPIO1_CFG, GPIO2_MODE and GPIO2_CFG register by program.
- **Step 5:** Start to execute ADC conversion by setting ADS = 1.
- **Step 6:** Wait the end of ADC converting through checking EOC = 1 or ADCIF = 1. If ADC interrupt function is enabled, the program executes ADC interrupt service when ADC interrupt occurrence. **ADS is cleared when the end of ADC converting automatically. EOC bit indicates ADC processing status immediately and is cleared when ADS = 1. Users needn't to clear it by program.**

7.5.4 ADC PIN CONFIGURATION

ADC input pins are shared with P2.0~P2.6 and P1.0~P1.4 digital I/O pins. ADC channel selection is through CHS[14:0] bits in [ADC_CONVCTRL](#) register.

Connect an analog signal to COMS digital input pin, especially, the analog signal level is about 1/2 VDD will cause extra current leakage. In the power down mode, the above leakage current will be a big problem. Unfortunately, if users connect more than one analog input signal to AINx will encounter above current leakage situation.

The P2.0/AIN0 can be ADC external high reference voltage input pin when AVREFHSEL =1. In the condition, P2.0 GPIO mode must be set as input mode and inactive (no pull-down/up resistor enabled, Schmitt trigger disabled) with [GPIO2_MODE](#) and [GPIO2_CFG](#) register by program.

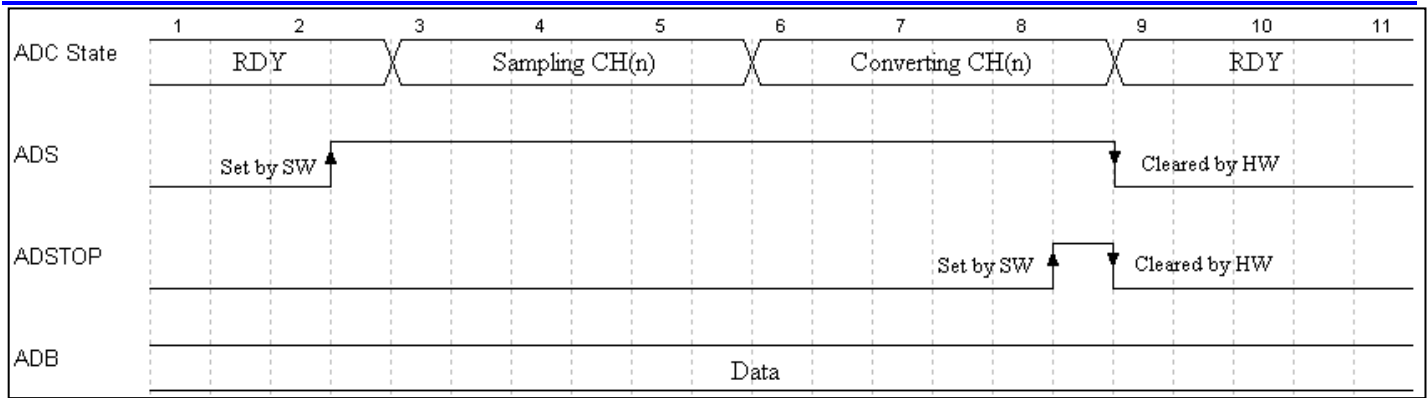
Only one pin of Port 2 can be configured as ADC input in the same time. The pins of Port 1 and Port 2 configured as ADC input channel must be set as input mode, inactive (no pull-down/pull-up resistor enabled, Schmitt trigger disabled, Data register keep low) with [GPIO1_MODE](#), [GPIO1_CFG](#), [GPIO2_MODE](#) and [GPIO2_CFG](#) register by program to avoid current leakage.

* **Note: The GPIO mode of ADC input channels used must be set as input mode and inactive (no pull-down/pull-up resistor enabled, Schmitt trigger disabled, Data register keep low) with GPIO1_MODE, GPIO1_CFG, GPIO2_MODE and GPIO2_CFG register by program.**

7.6 ADC CONVERSION MODES

The ADC can either converts a single channel or a sequence of multiple channels, determined by CH bit in ADC_CONVCTRL register. The ADC will convert only one channel selected by CHS[14:0] bits in ADC_CONVCTRL register if CH bit is 0, or convert a sequence of multiple channels if CH bit is 1. The multiple channels are selected by CHS[14:0] bits in ADC_CONVCTRL register, and each analog input channel has a dedicated selection bit.

Besides, the ADC supports either Single mode or Continuous mode, determined by SCMODE bit in ADC_CONVCTRL register. The ADSTOP bit can be set to 1 to stop the continuous mode.



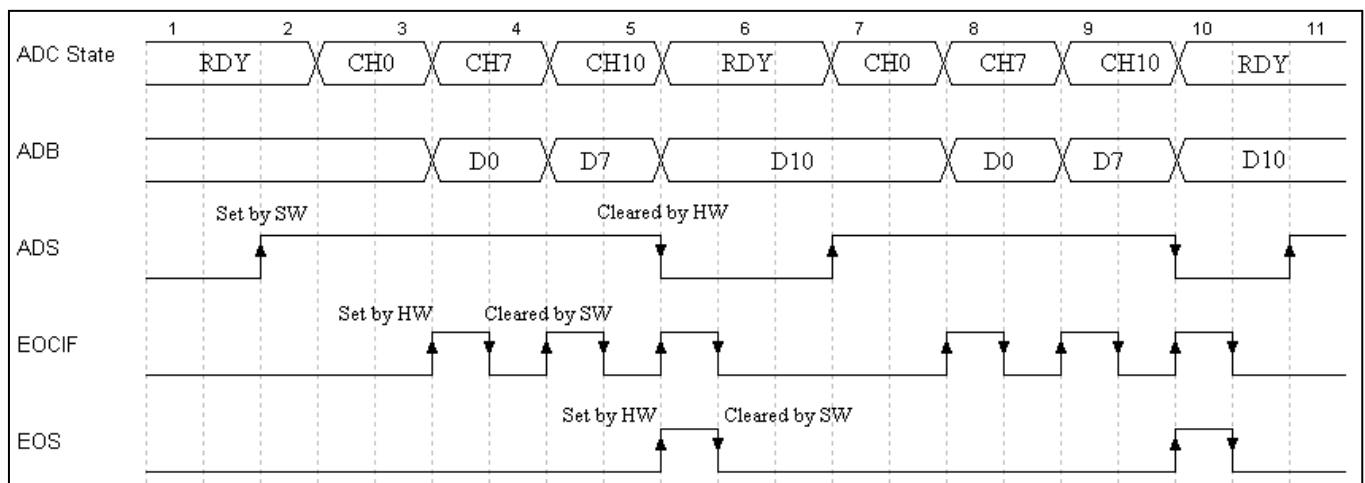
7.6.1 Single Mode

Clear SCMODE bit and set ADS bit to 1 to start the ADC conversion in Single mode. The selected channels will be converted once per trigger.

When each conversion is complete, the converted data are stored in the 16-frame ADB FIFO. The EOCIF[14:0] bit of the converted channels will be set individually to indicate the end of the conversion, and the ADC interrupt is generated if the related EOCIE[14:0] bits are set.

If a sequence of multiple channels is selected, and the conversion of all channels of the sequence is complete, the EOSIF bit will be set to indicate the end of sequence, and the ADC interrupt is generated if the EOSIE bit is set.

Then the ADC stops until a new external trigger event occurs or the ADSTART bit is set again.

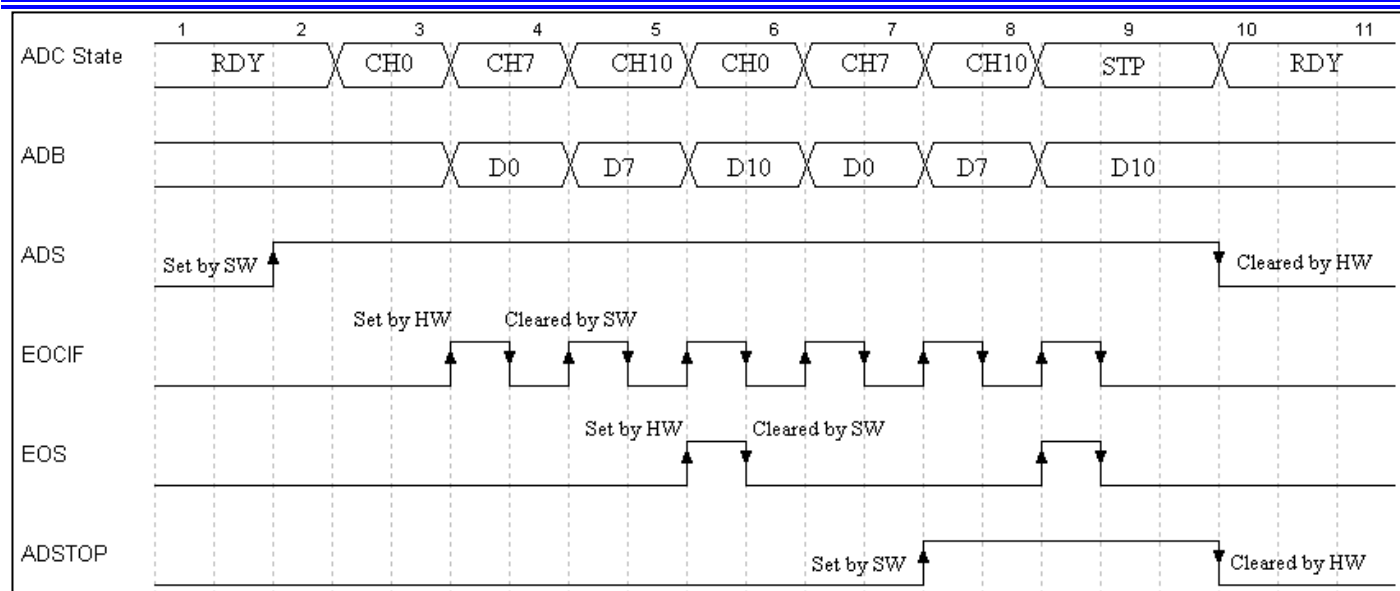


7.6.2 Continuous Mode

Set both SCMODE bit and ADS bit to 1 to start the ADC conversion in Continuous mode. The ADC will convert either one channel or a sequence of multiple channels once and then automatically re-starts and continuously performs the same sequence of conversions until the ADSTOP bit is set to 1 to stop the continuous mode.

When each conversion is complete, the converted data are stored in the 16-frame ADB FIFO. The EOCIF[14:0] bit of the converted channels will be set individually to indicate the end of the conversion, and the ADC interrupt is generated if the related EOCIE[14:0] bits are set.

If a sequence of multiple channels is selected, and the conversion of all channels of the sequence is complete, the EOSIF bit will be set to indicate the end of sequence, and the ADC interrupt is generated if the EOSIE bit is set.



7.7 ADC WATCHDOG WINDOW

The ADC window watchdog (AWW) is enabled by setting AWWEN bit to 1, it will monitor whether the measured data is inside or outside of the AWW high threshold value and low threshold value in ADC_AWW register. The AWWIF bit is set to 1 to notify the system when a desired condition is matched, and the ADC interrupt is generated if the AWWIE bit is set.

- AWDWINDOW[1:0] = 00b: ADB[11:0] > HT[11:0]

ADC_AWWTH	ADB[11:0]	AWWIF
	0x3FF	AWWIF=1
	...	
	0x081	
HT[11:0]=0x080	0x080	AWWIF=0
	0x07F	
	...	
LT[11:0]=0x000	0x000	

- AWDWINDOW[1:0] = 10b : ADB[11:0] < LT[11:0]

ADC_AWWTH	ADB[11:0]	AWWIF
HT[11:0]=0x3FF	0x3FF	AWWIF=0
	...	
	0x041	
LT[11:0]=0x040	0x040	
	0x03F	AWWIF=1
	...	
	0x000	

- AWDWINDOW[1:0] = 01b: LT[11:0] < ADB[11:0] < HT[11:0]

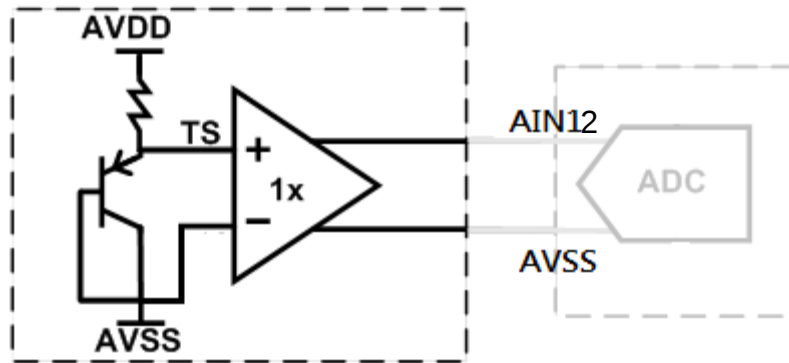
ADC_AWWTH	ADB[11:0]	AWWIF
	0x3FF	AWWIF=0
	...	
	0x081	
HT[11:0]=0x80	0x080	
	0x07F	AWWIF=1
	...	
	0x041	
LT[11:0]=0x040	0x040	AWWIF=0
	0x03F	
	...	
	0x000	

7.8 TEMPERATURE SENSOR

In most applications, the characteristic of the temperature sensor might change in different temperature. To get the temperature information, a temperature sensor (TS) is built-in for temperature measurement, and is internally connected to AIN12 input channel which is used to convert the sensor output voltage into a digital value.

If TSEN = 1, the temperature sensor is enabled. When not in use, this sensor can be put in power down mode if TSEN = 0.

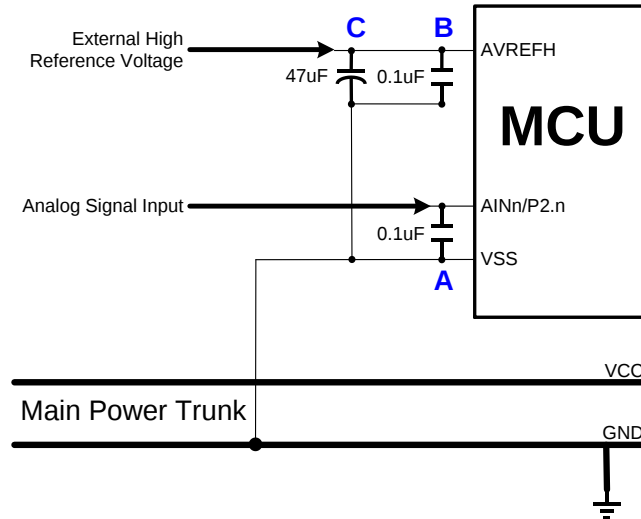
- * **Note:**
1. Typical temperature parameter is 3.2mV/°C.
 2. The Temp sensor is converted with differential mode, 1 LSB = Vref / 2048



$$V(TS) = AVREFH * (ADB \text{ of } AIN12 - 2048) / 2048$$

Temperature	AVREFH	ADC output (12-Bit)	V(TS)
0°C	2.5V	2756	0.864V
25°C	2.5V	2819	0.941V
105°C	2.5V	3026	1.194V

7.9 ADC CIRCUIT



The analog signal is inputted to ADC input pin "AINn/P2.n". The ADC input signal must be through a 0.1uF capacitor "A". The 0.1uF capacitor is set between ADC input pin and VSS pin, and must be on the side of the ADC input pin as possible. Don't connect the capacitor's ground pin to ground plain directly, and must be through VSS pin. The capacitor can reduce the power noise effective coupled with the analog signal.

If the ADC high reference voltage is from external voltage source, the external high reference is connected to AVREFH pin (P2.0). The external high reference source must be through a 47uF "C" capacitor first, and then 0.1uF capacitor "B". These capacitors are set between AVREFH pin and VSS pin, and must be on the side of the AVREFH pin as possible. Don't connect the capacitor's ground pin to ground plain directly, and must be through VSS pin.

7.10 ADC REGISTERS

Base Address: 0x4002 6000

7.10.1 ADC Management register (ADC_ADM)

Address Offset: 0x00

*** Note:**

1. When ADC is enabled (ADENB=1) and global channel is enabled (GCHS=1), the ADC shared pins transfers to ADC purpose and disable GPIO function and disable pull-up/pull-down resistor by HW automatically, the P2.n/AINn's digital I/O function including pull-up is isolated.
2. When ADC is disabled (ADENB=0) or global channel is disabled (GCHS=0), the ADC pins returns to last GPIO status.
3. If P2.0 is used as external reference voltage input pin, users should set P2.0 as input mode without pull-up.
4. GAINSEL bit shall be set when ADC sample rate is more than 250KHz, or higher resolution is needed when using internal reference voltage, P2.5 becomes VREF pin, and MUST connect 0.1uF capacitor to VSS.
5. If ADC internal voltage reference, CMP internal voltage reference, or DAC internal voltage reference is used, ADC_ADM2 SHALL be set to 0x3.

Bit	Name	Description	Attribute	Reset
31:22	Reserved		R	0
21	GAINSEL	VIREF gain select bit 0: Low-gain 1: High-gain (Shall be selected when ADC sample rate is more than 250KHz, or higher resolution is needed when using internal reference voltage, P2.5 becomes VREF pin, and MUST connect 0.1uF capacitor to VSS)	R/W	0
20	TSEN	Temp sensor enable bit 0: Disable 1: Enable	R/W	0
19	Reserved		R	0
18	FIFORST	ADB FIFO Reset bit 0: No effect 1: Reset ADB FIFO (Data in FIFO is cleared). This bit will be cleared by HW automatically.	W	0
17	OVRMODE	Overflow mode select bit 0: Preserve (The old data is preserved when the overrun occurs) 1: Overwrite (The ADC data FIFO is overwritten with the latest data)	R/W	0
16	ADSTOP	ADC stop control bit. 0: No effect 1: Stop ADC converting. ADSTOP is cleared when the end of ADC converting is stopped automatically.	R/W	0
15:13	VHS[2:0]	Internal reference voltage level selection. 000: Internal 2.0V as ADC internal reference high voltage 001: Internal 3.0V as ADC internal reference high voltage 010: Internal 4.5V as ADC internal reference high voltage 011: Internal 2.5V as ADC internal reference high voltage 100~110: Reserved 111: VDD as ADC internal reference high voltage	R/W	000b
12	AVREFHSEL	ADC high reference voltage source select bit 0: Internal reference voltage. (P2.0 is GPIO or AIN0 pin) 1: Enable external reference voltage from P2.0	R/W	0
11	ADENB	ADC Enable bit. In power saving mode, disable ADC to reduce power consumption.	R/W	0

		0: Disable 1: Enable		
10:8	ADCKS[2:0]	ADC Clock (MCLK) source divider 000: ADC_PCLK / 1 001: ADC_PCLK / 2 010: ADC_PCLK / 4 011: ADC_PCLK / 8 100: ADC_PCLK / 16 101: ADC_PCLK / 32 Other: Reversed	R/W	0
7	ADLEN	ADC resolution control bit. 0: 8-bit ADC 1: 12-bit ADC	R/W	1
6	ADS	ADC start control bit. 0: ADC converting stops. 1: Start to execute ADC converting. ADS is cleared when the end of ADC converting automatically.	R/W	0
5	EOC	ADC status bit. Indicates ADC processing status immediately and is cleared when ADS = 1. 0: ADC is progressing (converting or calibrating). 1: End of converting and reset ADS bit.	R/W	0
4	GCHS	ADC global channel select bit. 0: Disable AIN channel 1: Enable AIN channel	R/W	0
3:0	Reserved		R	0

7.10.2 ADC Data register (ADC_ADB)

Address Offset: 0x04

The ADC has a 16-frame data buffer. This register is the top layer of the ADC data FIFO, and contains the oldest A/D converter result.

* Note: 1. The initial value of ADC buffer (ADB) after reset is unknown. 2. The Temp sensor is converted with differential mode, 1 LSB = $V_{ref} / 2048$				
--	--	--	--	--

Bit	Name	Description	Attribute	Reset
31:13	Reserved		R	0
12	FIRST	First ADB of Multiple channels 0: Not first ADB of Multiple channels 1: First ADB of Multiple channels	R	0
11:0	ADB[11:0]	ADB11~ADB4 bits for 8-bit ADC ADB11~ADB0 bits for 12-bit ADC	R	0

The AINx (exclude AIN12) input voltage vs. ADB's output data

AIN n	ADB11	ADB10	ADB9	ADB8	ADB7	ADB6	ADB5	ADB4	ADB3	ADB2	ADB1	ADB0
0/4096*VREFH	0	0	0	0	0	0	0	0	0	0	0	0
1/4096*VREFH	0	0	0	0	0	0	0	0	0	0	0	1
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
4094/4096*VREFH	1	1	1	1	1	1	1	1	1	1	1	0
4095/4096*VREFH	1	1	1	1	1	1	1	1	1	1	1	1

The AIN12 input voltage vs. ADB's output data

AIN n	ADB11	ADB10	ADB9	ADB8	ADB7	ADB6	ADB5	ADB4	ADB3	ADB2	ADB1	ADB0
-------	-------	-------	------	------	------	------	------	------	------	------	------	------

-2048/2048*VREFH	0	0	0	0	0	0	0	0	0	0	0	0
-2047/2048*VREFH	0	0	0	0	0	0	0	0	0	0	0	1
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
0	1	0	0	0	0	0	0	0	0	0	0	0
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
2046/2048*VREFH	1	1	1	1	1	1	1	1	1	1	1	0
2047/2048*VREFH	1	1	1	1	1	1	1	1	1	1	1	1

7.10.3 ADC Interrupt Enable register (ADC_IE)

Address offset: 0x0C

This register allows control over which A/D channels generate an interrupt when a conversion is complete. For example, it may be desirable to use some A/D channels to monitor sensors by continuously performing conversions on them. The most recent results are read by the application program whenever they are needed. In this case, an interrupt is not desirable at the end of each conversion for some A/D channels.

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23	EOCALIE	End of calibration interrupt enable bit 0: Disable 1: Enable	R/W	0
22	OVRIE	Overrun raw interrupt enable bit 0: Disable 1: Enable	R/W	0
21	AWWIE	ADC Window WDT interrupt enable bit 0: Disable 1: Enable	R/W	0
20	EOSIE	End of sequence interrupt enable bit 0: Disable 1: Enable	R/W	0
19:15	Reserved		R	0
14:0	EOCIE[14:0]	End of conversion interrupt enable bit. (EOCIE[13] is reserved.) 0: Disable 1: Enable	R/W	0

7.10.4 ADC Raw Interrupt Status register (ADC_RIS)

Address offset: 0x10

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23	EOCALIF	End of calibration raw interrupt flag 0: No interrupt requirements met 1: Interrupt requirements met	R	0
22	OVRIF	Overrun raw interrupt flag 0: No interrupt requirements met 1: Interrupt requirements met	R	0
21	AWWIF	ADC window WDT raw interrupt flag 0: No interrupt requirements met 1: Interrupt requirements (ADB matches AWW condition) met	R	0
20	EOSIF	End of sequence raw interrupt flag 0: No sequence of conversion is completed 1: Interrupt requirements (The multiple channels selected by CHS[14:0] bits are all converted) met	R	0
19:15	Reserved		R	0

14:0	EOCIF[14:0]	End of conversion interrupt flag. (EOCIF[13] is reserved.) 0: No conversion is completed on AINx 1: Interrupt requirements (AINx finishes conversion) met on AINx	R	0
------	-------------	---	---	---

7.10.5 ADC Raw Interrupt Status register (ADC_IC)

Address offset: 0x18

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23	EOCALIC	End of calibration interrupt flag clear bit 0: No effect 1: Clear EOCALIF bit	W	0
22	OVRIC	Overrun raw interrupt flag clear bit 0: No effect. 1: Clear OVRIF bit	W	0
21	AWWIC	ADC window WDT raw interrupt flag clear bit 0: No effect. 1: Clear AWWIF bit	W	0
20	EOSIC	End of sequence raw interrupt flag clear bit 0: No effect. 1: Clear EOSIF bit	W	0
19:15	Reserved		R	0
14:0	EOCIC[14:0]	End of conversion raw interrupt flag clear bit. (EOCIC[13] is reserved.) 0: No effect. 1: Clear related EOCIF[x] bit	W	0

7.10.6 ADC Convert Control register (ADC_CONVCTRL)

Address offset: 0x1C

*** Note:**

1. ADC can't be stopped in Continuous mode when only 1 channel is enabled and CH bit is 1 (multiple channel).
2. Recommended to stop ADC when EOSIF is set if CH bit is 1 (multiple channel).

Bit	Name	Description	Attribute	Reset
31:22	Reserved		R	0
21	SCMODE	Single/Continuous mode select bit 0: Single mode 1: Continuous mode	R/W	0
20	CH	Converting channel number 0: Single channel 1: Multiple channels	R/W	0
19:15	Reserved		R	0
14:0	CHS[14:0]	AINx (AIN0~AIN11, AIN12=Temperature sensor, AIN13=reserved, AIN14=AVDD/2) selection bits 0: Not selected to convert 1: AINx selected to convert	R/W	0

7.10.7 ADC Window Watchdog register (ADC_AWW)

Address offset: 0x20

Bit	Name	Description	Attribute	Reset
31:23	Reserved		R	0
22:21	WINDOW[1:0]	AWW window select bits 00: ADB > AWDHT 01: AWDLT < ADB < AWDHT 10: ADB < AWDLT 11: Reserved	R/W	0
20	AWWENB	ADC Window Watchdog enable bit 0: Disable 1: Enable	R/W	0
19:15	Reserved		R	0
14:0	AWWCHS[14:0]	AWW channel selection bits. (AWWCHS [13] is reserved.) 0: Not selected 1: AINx selected to be monitored	R/W	0

7.10.8 ADC Window Watchdog Threshold register (ADC_AWWTH)

Address offset: 0x24

Bit	Name	Description	Attribute	Reset
31:28	Reserved		R	0
27:16	HT[11:0]	AWW window High threshold value	R/W	0
15:12	Reserved		R	0
11:0	LT[11:0]	AWW window Low threshold value	R/W	0

7.10.9 ADC Management register 1 (ADC_ADM1)

Address offset: 0x30

*** Note:**

1. The calibration must be run at least one time before starting A/D conversion.
2. CALIVALENB SHALL be set 1.

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	ACS	ADC Calibration start bit 0: ADC calibration stops. 1: ADC calibration starts, and is cleared in the end of calibration automatically.	R/W	0
2	CALIVALENB	ADC calibration value enable bit 0: ADC conversion without calibration value 1: ADC conversion with calibration value	R/W	0
1:0	Reserved		R	0

7.10.10 ADC Management register 2 (ADC_ADM2)

Address offset: 0x34

*** Note:** If ADC internal voltage reference, CMP internal voltage reference, or DAC internal voltage reference is used, IT[1:0] bits of ADC_ADM2 SHALL be set to 11b.

Bi	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1:0	IT[1:0]	Set "11b" if ADC internal voltage reference, CMP internal reference, or DAC internal voltage reference is used.	R/W	0

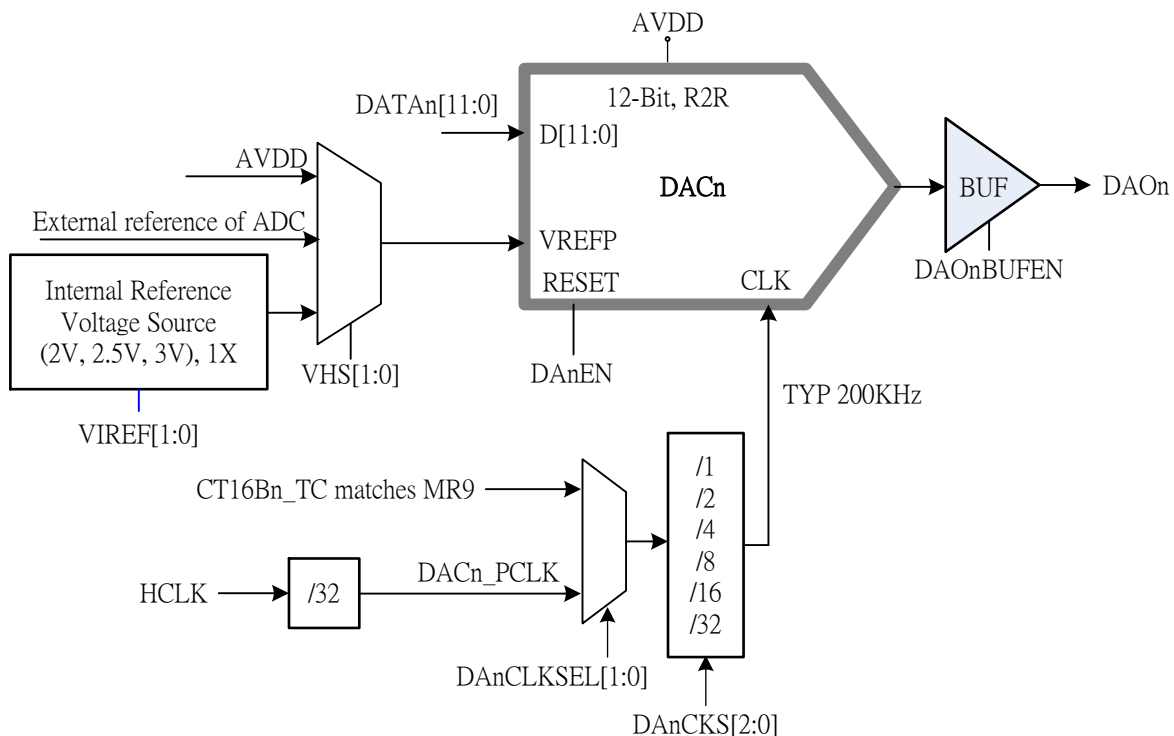
8 12-BIT DIGITAL TO ANALOG CONVERTOR (DAC)

8.1 OVERVIEW

The micro-controller builds in 4 Rail-to-Rail structure 12-bit DAC. Each DAC channel can be powered on by setting the corresponding DACnEN bit in DAC_ADM register. Each DAC integrates one output buffer that can be used to reduce the output impedance and to enhance analog output driving current without having to add an external operational amplifier.

The VREFP comes from AVDD, External reference voltage of ADC, or internal voltage reference 2V/2.5V/3.0V selected by VIREF[1:0] bits. 4 DAC shares the same voltage reference.

The DAC can be triggered by CT16_B0/1/2/5 timer or Software. The DAC outputs are updated on the rising edge of the clock.



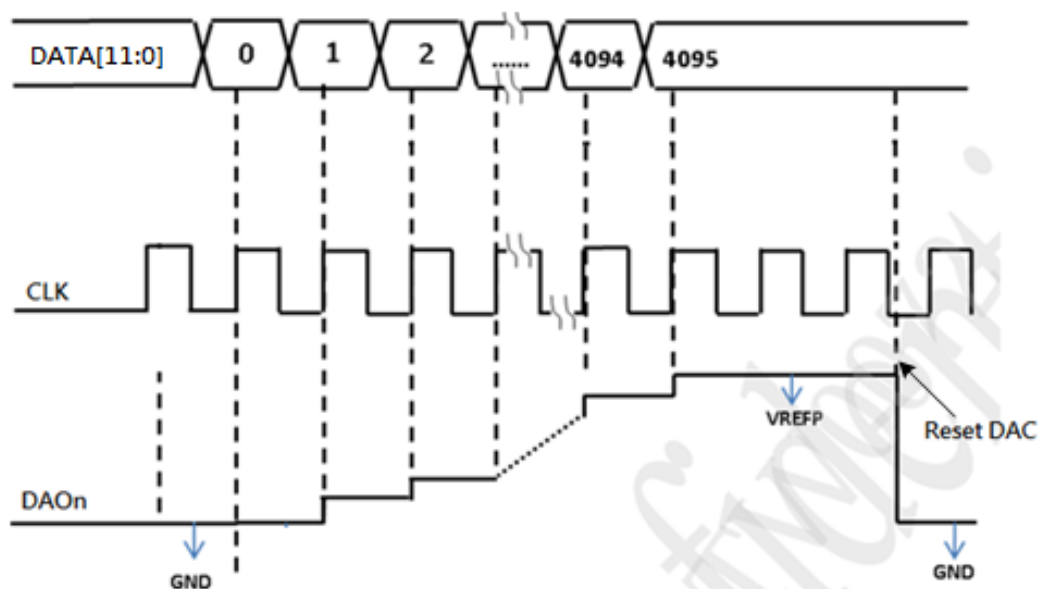
8.2 FEATURES

1. Programmable internal buffer.
2. Internal voltage reference.
3. External trigger sources.

8.3 TIMING DIAGRAM

* Note: The DACn_CLK shall be less or equal to 200K Hz.

HCLK (MHz)	DACn_PCLK (KHz)	DACnCKS[2:0]		DACn_CLK (KHz)
12	375	001b	DACn_PCLK / 2	187.5
48	1500	011b	DACn_PCLK / 8	187.5
60	1875	100b	DACn_PCLK / 16	117



8.4 DAC OUTPUT VOLTAGE

Digital inputs are converted to output voltages on a linear conversion between 0 and VREF. The analog output voltages on each DAC channel pin are determined by the following equation:

$$DAOn = VREF \times \frac{DAC_DATAn}{4096}$$

8.5 DAC Registers

Base Address: 0x4006 4000 (DAC)

8.5.1 DAC Management register (DAC_ADM)

Address Offset: 0x00

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11	DAO3BUFEN	DAC3 Output buffer enable bit 0: Disable 1: Enable	R/W	0
10	DAO2BUFEN	DAC2 Output buffer enable bit 0: Disable 1: Enable	R/W	0
9	DAO1BUFEN	DAC1 Output buffer enable bit 0: Disable 1: Enable	R/W	0
8	DAO0BUFEN	DAC0 Output buffer enable bit 0: Disable 1: Enable	R/W	0
7:4	Reserved		R	0
3	DAC3EN	DAC3 enable bit 0: Disable 1: Enable	R/W	0
2	DAC2EN	DAC2 enable bit 0: Disable 1: Enable	R/W	0
1	DAC1EN	DAC1 enable bit 0: Disable 1: Enable	R/W	0
0	DAC0EN	DAC0 enable bit 0: Disable 1: Enable	R/W	0000b

8.5.2 DAC Internal Reference Voltage Source register (DAC_VIREF)

Address Offset: 0x04

*** Note:**

1. External reference voltage from ADC shall more or equal to 1.5V.
2. If ADC internal voltage reference, CMP internal voltage reference, or DAC internal voltage reference is used, ADC_ADM2 SHALL be set to 0x3.

Bit	Name	Description	Attribute	Reset
31:18	Reserved		R	0
17:16	VHS[1:0]	DAC internal reference high voltage select bit 00: AVDD 01: VIREF 10: External reference voltage from ADC 11: Reserved	R/W	0

15:2	Reserved		RW	0
1:0	VIREF	DAC internal reference voltage (V _{IREF}) source 00: Disable 01: Internal 2.0V 10: Internal 2.5V 11: Internal 3.0V	RW	0

8.5.3 DAC Clock Source register (DAC_CLK)

Address Offset: 0x08

*** Note:** The DACn_CLK shall be less or equal to 200K Hz.

Bit	Name	Description	Attribute	Reset
31:15	Reserved		R	0
14:12	DA3CKS[2:0]	DAC3 Clock source divider 000: Clock = DAC3 clock source / 1 001: Clock = DAC3 clock source / 2 010: Clock = DAC3 clock source / 4 011: Clock = DAC3 clock source / 8 100: Clock = DAC3 clock source / 16 101: Clock = DAC3 clock source / 32 Other: Reversed	RW	0
11	Reserved		R	0
10:8	DA2CKS[2:0]	DAC2 Clock source divider 000: Clock = DAC2 clock source / 1 001: Clock = DAC2 clock source / 2 010: Clock = DAC2 clock source / 4 011: Clock = DAC2 clock source / 8 100: Clock = DAC2 clock source / 16 101: Clock = DAC2 clock source / 32 Other: Reversed	RW	0
7	Reserved		R	0
6:4	DA1CKS[2:0]	DAC1 Clock source divider 000: Clock = DAC1 clock source / 1 001: Clock = DAC1 clock source / 2 010: Clock = DAC1 clock source / 4 011: Clock = DAC1 clock source / 8 100: Clock = DAC1 clock source / 16 101: Clock = DAC1 clock source / 32 Other: Reversed	RW	0
3	Reserved		R	0
2:0	DA0CKS[2:0]	DAC0 Clock source divider 000: Clock = DAC0 clock source / 1 001: Clock = DAC0 clock source / 2 010: Clock = DAC0 clock source / 4 011: Clock = DAC0 clock source / 8 100: Clock = DAC0 clock source / 16 101: Clock = DAC0 clock source / 32 Other: Reversed	RW	0

8.5.4 DAC Trigger Control register (DAC_TGCTRL)

Address Offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:20	Reserved		R	0
19	DA3SWTGR	DAC3 Software trigger bit 0: SW trigger stops 1: SW trigger starts, and is cleared by HW.	RW	0
18	DA2SWTGR	DAC2 Software trigger bit 0: SW trigger stops 1: SW trigger starts, and is cleared by HW.	RW	0
17	DA1SWTGR	DAC1 Software trigger bit 0: SW trigger stops 1: SW trigger starts, and is cleared by HW.	RW	0
16	DA0SWTGR	DAC0 Software trigger bit 0: SW trigger stops 1: SW trigger starts, and is cleared by HW.	RW	0
15	Reserved		R	0
14:12	DA3TGSEL[2:0]	DAC3 Trigger source select bits 000: PCLK (=HCLK / 32) 001: CT16B0_TC matches MR9 010: CT16B1_TC matches MR9 011: CT16B2_TC matches MR9 100: Reserved 101: Reserved 110: CT16B5_TC matches MR9 111: Software trigger	RW	0
11	Reserved		R	0
10:8	DA2TGSEL[2:0]	DAC2 Trigger source select bits 000: PCLK (=HCLK / 32) 001: CT16B0_TC matches MR9 010: CT16B1_TC matches MR9 011: CT16B2_TC matches MR9 100: Reserved 101: Reserved 110: CT16B5_TC matches MR9 111: Software trigger	RW	0
7	Reserved		R	0
6:4	DA1TGSEL[2:0]	DAC1 Trigger source select bits 000: PCLK (=HCLK / 32) 001: CT16B0_TC matches MR9 010: CT16B1_TC matches MR9 011: CT16B2_TC matches MR9 100: Reserved 101: Reserved 110: CT16B5_TC matches MR9 111: Software trigger	RW	0
3	Reserved		R	0
2:0	DA0TGSEL[2:0]	DAC0 Trigger source select bits 000: PCLK (=HCLK / 32) 001: CT16B0_TC matches MR9 010: CT16B1_TC matches MR9 011: CT16B2_TC matches MR9 100: Reserved 101: Reserved 110: CT16B5_TC matches MR9 111: Software trigger	RW	0

8.5.5 DAC Data Control register 0 (DAC_DATACTRL0)

Address Offset: 0x10

Bit	Name	Description	Attribute	Reset
31:28	Reserved		R	0
27:16	DATA1[11:0]	12-bit DAC1 setup data bits.	W/R	0
15:12	Reserved		R	0
11:0	DATA0[11:0]	12-bit DAC0 setup data bits.	W/R	0

8.5.6 DAC Data Control register 1 (DAC_DATACTRL1)

Address Offset: 0x14

Bit	Name	Description	Attribute	Reset
31:28	Reserved		R	0
27:16	DATA3[11:0]	12-bit DAC3 setup data bits.	W/R	0
15:12	Reserved		R	0
11:0	DATA2[11:0]	12-bit DAC2 setup data bits.	W/R	0

8.5.7 DAC Interrupt Enable register (DAC_IE)

Address Offset: 0x20

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	DAC3IE	DAC3 interrupt enable bit 0: Disable 1: Enable	R/W	0
2	DAC2IE	DAC2 interrupt enable bit 0: Disable 1: Enable	R/W	0
1	DAC1IE	DAC1 interrupt enable bit 0: Disable 1: Enable	R/W	0
0	DAC0IE	DAC0 interrupt enable bit 0: Disable 1: Enable	R/W	0

8.5.8 DAC Raw Interrupt Status register (DAC_RIS)

Address Offset: 0x24

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0

3	DAC3IF	Interrupt flag for match DAC3. 0: No interrupt on DAC3 1: Interrupt requirements (DAC3 is triggered and outputs) met on DAC3.	R	0
2	DAC2IF	Interrupt flag for match DAC2. 0: No interrupt on DAC2 1: Interrupt requirements (DAC2 is triggered and outputs) met on DAC2.	R	0
1	DAC1IF	Interrupt flag for match DAC1. 0: No interrupt on DAC1 1: Interrupt requirements (DAC1 is triggered and outputs) met on DAC1.	R	0
0	DAC0IF	Interrupt flag for match DAC0. 0: No interrupt on DAC0 1: Interrupt requirements (DAC0 is triggered and outputs) met on DAC0.	R	0

8.5.9 DAC Interrupt Clear register (DAC_IC)

Address Offset: 0x28

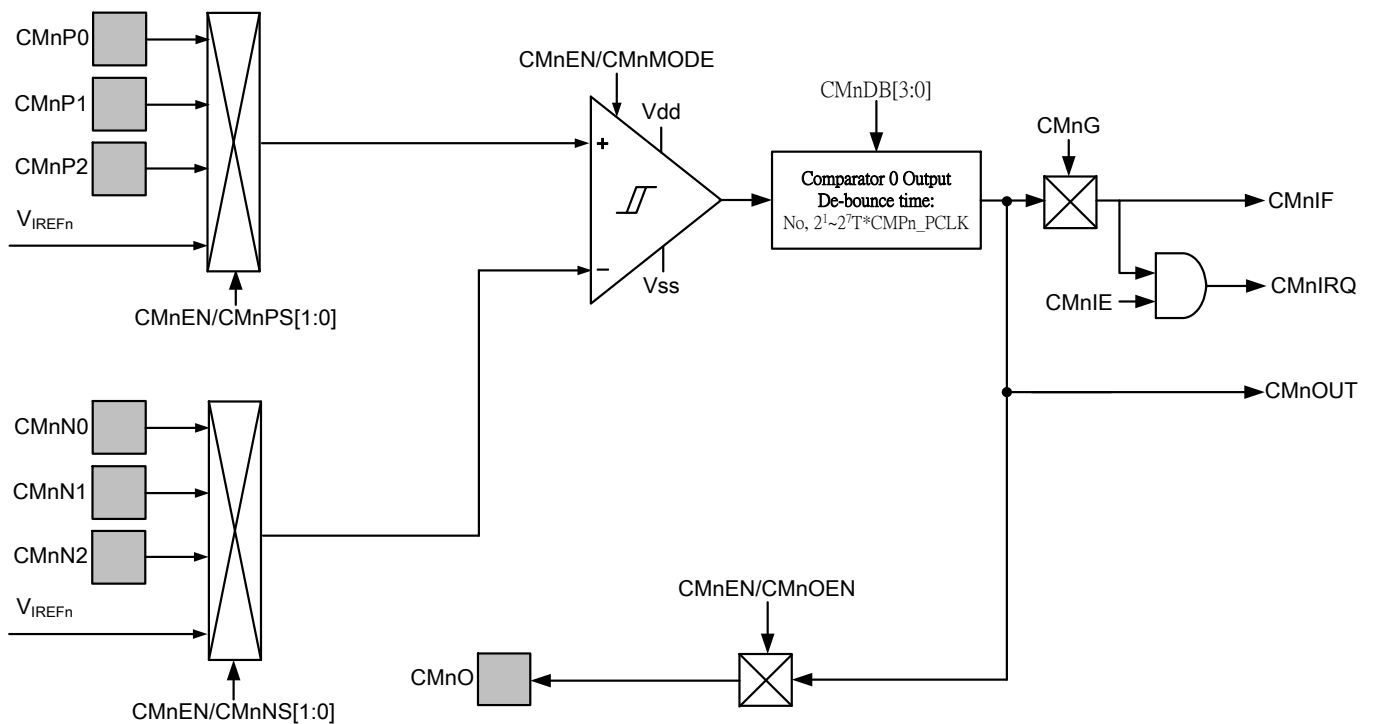
Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	DAC3IC	0: No effect 1: Clear DAC3IF bit	W	0
2	DAC2IC	0: No effect 1: Clear DAC2IF bit	W	0
1	DAC1IC	0: No effect 1: Clear DAC1IF bit	W	0
0	DAC0IC	0: No effect 1: Clear DAC0IF bit	W	0

9 RAIL TO RAIL ANALOG COMPARATOR

9.1 OVERVIEW

The micro-controller builds in 2 sets comparators. The comparators are Rail-to-Rail structure. That means the input/output voltage is real from Vdd~Vss. When the positive input voltage is greater than the negative input voltage, the comparator output is high. When the positive input voltage is smaller than the negative input voltage, the comparator output is low.

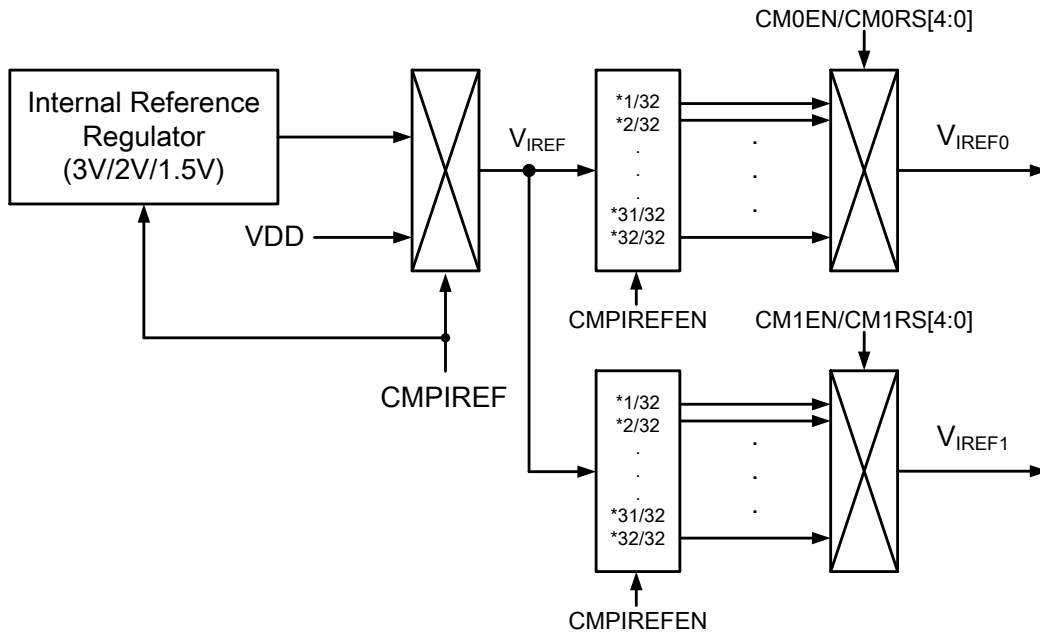
The CMnOUT and CMnIF bits indicate the comparator result. The CMnOUT shows the comparator result immediately, but the CMnIF only indicates the event of the comparator result. The event condition is controlled by CMnG bit and includes rising edge (CMnOUT changes from low to high), falling edge (CMnOUT changes from high to low). The CMnIF = 1 condition makes the comparator interrupt service executed when CMnIE (comparator interrupt enable control bit) is set.



The comparator builds in internal reference to replace comparator external negative input voltage source and controlled by CMPIREF[1:0] and CMPIREFEN bits. The internal reference voltage is 3V/2V/1.5V from internal LDO, or VDD. When CMnNS[1:0] = 00b~10b, the comparator negative input is from external voltage source through CMnN0, CMnN1, or CMnN2 pin. When CMnNS[1:0] = 11b, the comparator negative voltage source is from internal reference and CMnN0/1/2 pins are GPIO function.

The comparator positive input terminal is controlled by CMnPS[1:0]. When CMnPS[1:0] = 01b~11b, the comparator positive input is from CMnP0, CMnP1, or CMnP2 pin. When CMnPS[1:0] = 00b, CMnP0/1/2 pins are GPIO function.

Besides, the comparator output signal is through a de-bounce circuit to filter comparator transient status. The de-bounce time is controlled by CMnDB[2:0] bits.



The main purposes of comparator are as following.

- **Normal comparator function:** General comparator mode compares the two tensions of positive input terminal and negative input terminal.
- **Interrupt function:** When comparator output edge direction equals to edge selection, the CMnIRQ activates and the system points program counter to interrupt vector to do interrupt sequence when CMnIE (comparator interrupt enable control bit) is set.

9.2 NORMAL COMPARATOR MODE

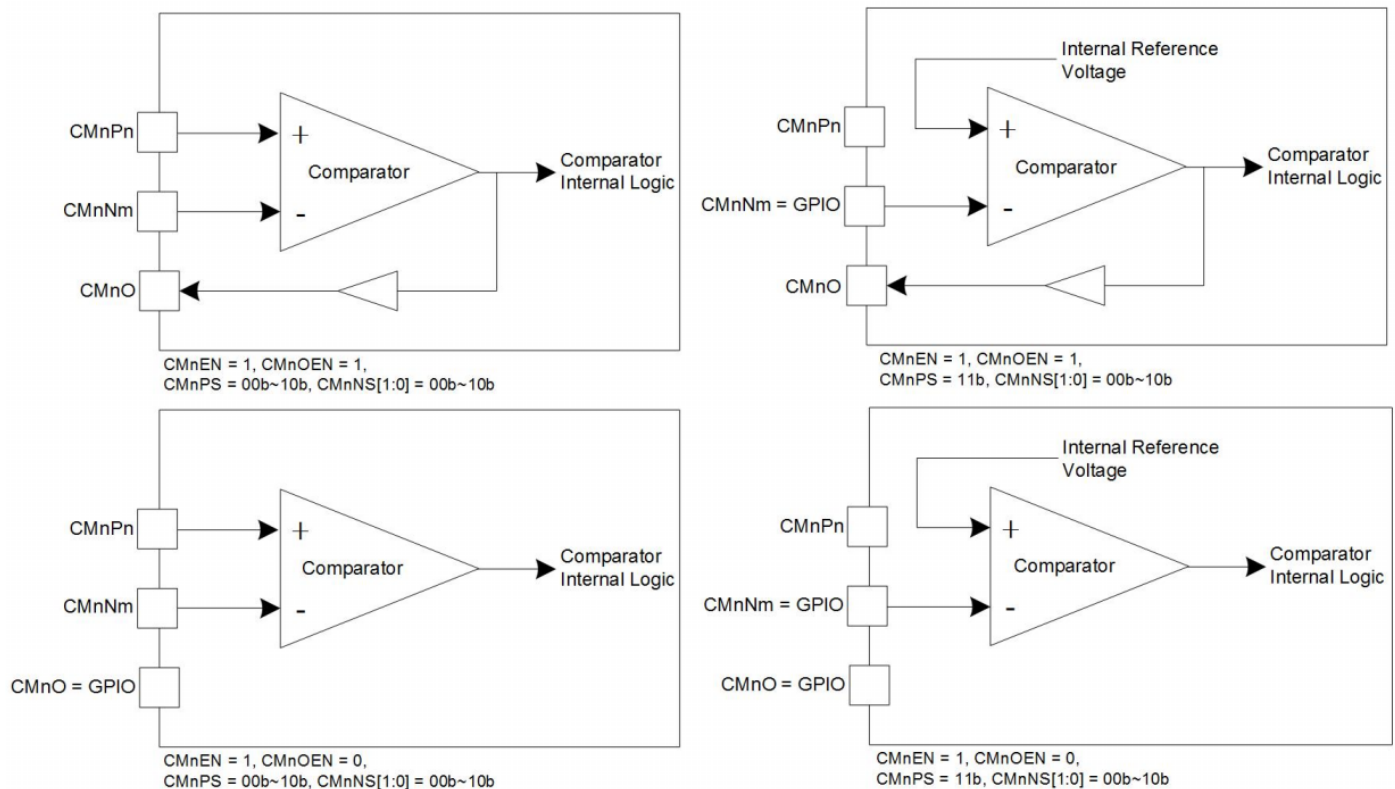
9.2.1 COMPARATOR ENABLE

The comparator pins are shared with GPIO controlled by CMnEN bit.

When CM0EN=1 and CM0NS[1:0] = 00b~10b, the comparator negative voltage source is from external voltage source through CM0N0, CM0N1, or CM0N2 pin. When CM0NS[1:0] = 11b, the comparator negative voltage source is from internal reference and CM0N0/1/2 pins are GPIO function. When CM0PS[1:0] = 01b~11b, the comparator positive input is from CM0P0, CM0P1, or CM0P2 pin. When CM0PS[1:0] = 00b, CM0P0/1/2 pins are GPIO function.

When CM1EN=1 and CM1NS[1:0] = 00b~10b, the comparator negative voltage source is from external voltage source through CM1N0, CM1N1, or CM1N2 pin. When CM1NS[1:0] = 11b, the comparator negative voltage source is from internal reference and CM1N0/1/2 pins are GPIO function. When CM1PS[1:0] = 01b~11b, the comparator positive input is from CM1P0, CM1P1, or CM1P2 pin. When CM1PS[1:0] = 00b, CM1P0/1/2 pins are GPIO function.

CMnOEN controls Comparator output connected to GPIO or not. When CMnOEN=1, Comparator output terminal is connected to GPIO pins and isolate GPIO function. When CMnOEN=0, comparator output status can be read through CMnOUT flag and CMnO pin is GPIO mode.

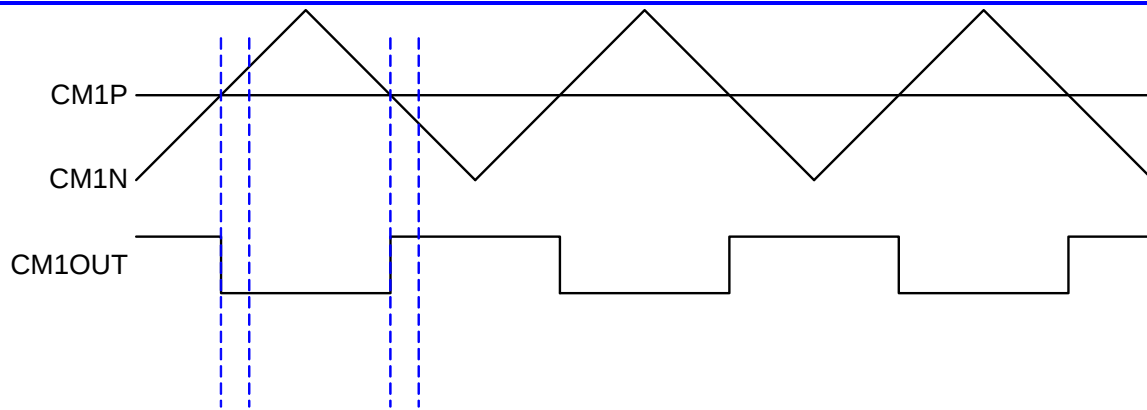


9.2.2 CMnOUT, CMnG AND CMnIF

The CMnOUT and CMnIF bits indicate the comparator result. The CMnOUT shows the comparator result immediately, but the CMnIF only indicates the event condition of the comparator result.

Comparator n compares positive terminal's voltage and negative terminal's voltage, and then output result to output pin. When $V_+ > V_-$, comparator outputs high status. When $V_+ < V_-$, comparator outputs low status.

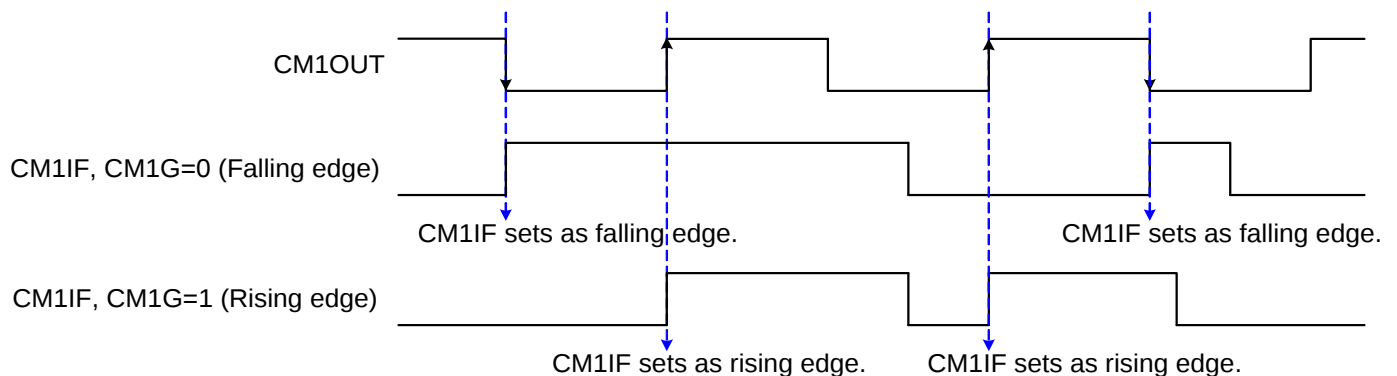
Take Comparator 1 as example,



The event condition is controlled by register and includes rising edge (CMnOUT changes from low to high), or falling edge (CMnOUT changes from high to low), controlled by CMnG bit. When CMnG= 0, the comparator n interrupt trigger direction is falling edge. When CMnG= 1, the comparator n interrupt trigger direction is rising edge. When comparator output edge event occurs and equal CMnG condition, CMnIF flag is issued. If CMnIE= 1, program counter points to interrupt vector to execute interrupt service routine.

*** Note: CMnOUT is comparator raw output without latch. It varies depend on the comparator process result. But the CMnIF is latch comparator output result. It must be cleared by program.**

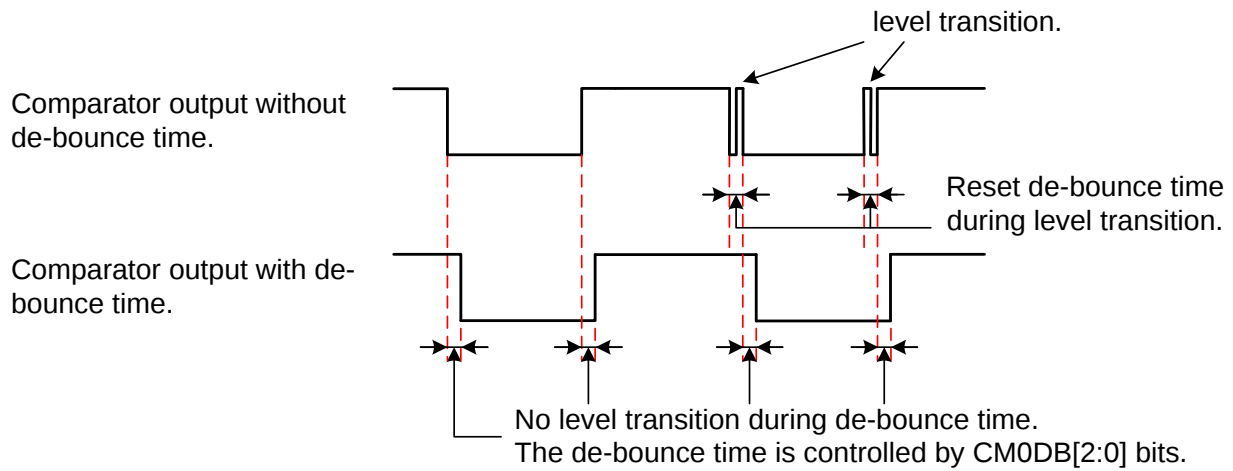
Take Comparator 1 as example,



***. CM1IF is cleared by program.**

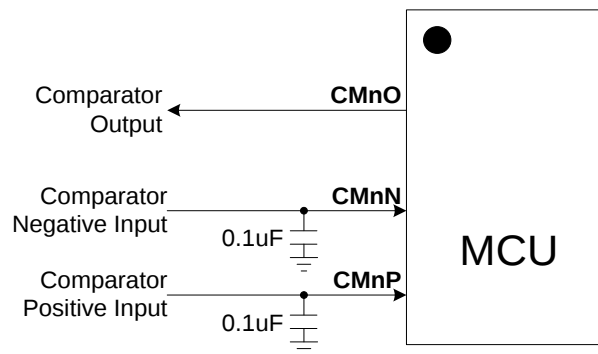
9.2.3 COMPARATOR OUTPUT DEBOUNCE TIME CONTROL

The critical condition is comparator positive voltage equal to comparator negative voltage, and the voltage range is decided comparator offset parameter of input common mode. In the voltage range, the comparator output signal is unstable and keeps oscillating until the differential voltage exits the range. In the condition, the comparator flag (CM0IF) latches the first exchanging and issue the status, but the status is a transient, not a stable condition. So the comparator builds in a filter to de-bounce the transient condition. The comparator output signal is through a de-bounce circuit to filter comparator transient status. The de-bounce time is controlled by CM0DB[2:0] bits that means the comparator minimum response time is 2*CMP_PCLK, 4*CMP_PCLK, 8*CMP_PCLK, ..., 128*CMP_PCLK, or no de-bounce. The de-bounce time depends on the signal slew rate and selected by program.



9.3 COMPARATOR APPLICATION NOTICE

The comparator is to compares the positive voltage and negative voltage to output result. The positive and negative sources are analog signal. In hardware application circuit, the comparator input pins must be connected a 0.1uF capacitor to reduce power noise and make the input signal more stable. The application circuit is as following.



9.4 COMPARATOR Registers

Base Address: 0x4002 8000 (CMP)

9.4.1 CMP Control register (CMP_CTRL)

Address Offset: 0x00

Bit	Name	Description	Attribute	Reset
31:29	Reserved		R	0
28	CM1G	CMP1 interrupt trigger direction control bit. 0: Falling edge trigger (CMP1 output status is from high to low as $V_+ < V_-$) 1: Rising edge trigger (CMP1 output status is from low to high as $V_+ > V_-$)	R/W	0
27	Reserved		R	0
27:26	CM1OEN	CMP1 output pin control bit. 0: Disable (CM1O is GPIO mode) 1: Enable (P3.9 is CM1O and isolate GPIO function)	R/W	0
25:21	CM1RS[4:0]	CMP1 internal reference voltage (V_{REF1}) selection bits. 0000: $V_{REF1} = V_{REF}$ 00001~11111: $V_{REF1} = V_{REF} * CM1RS[4:0] / 32$	R/W	00000b
20:19	CM1NS[1:0]	CMP1 negative input pin selection bit 00: CM1N0 is comparator negative input pin, and isolate GPIO function 01: CM1N1 is comparator negative input pin, and isolate GPIO function 10: CM1N2 is comparator negative input pin, and isolate GPIO function 11: V_{REF1} . CM1N0/CM1N1/CM1N2 pins are GPIO mode.	R/W	00b
18:17	CM1PS[1:0]	CMP1 positive input pin selection bits 00: CM1P0 is comparator positive input pin, and isolate GPIO function 01: CM1P1 is comparator positive input pin, and isolate GPIO function 10: CM1P2 is comparator positive input pin, and isolate GPIO function 11: V_{REF1} . CM1P0/CM1P1/CM1P2 pins are GPIO mode.	R/W	0
16	CM1EN	CMP1 enable bit. 0: Disable (CM1P0/1/2, CM1N0/1/2, CM1O are GPIO mode) 1: Enable	R/W	0
15:13	Reserved		R	0
12	CM0G	CMP0 interrupt trigger direction control bit. 0: Falling edge trigger (CMP0 output status is from high to low as $V_+ < V_-$) 1: Rising edge trigger (CMP0 output status is from low to high as $V_+ > V_-$)	R/W	0
11	Reserved		R	0
11:10	CM0OEN	CMP0 output pin control bits 0: Disable (CM0O is GPIO mode) 1: Enable (P0.3 is CM0O and isolate GPIO function)	R/W	0
9:5	CM0RS[4:0]	CMP0 internal reference voltage (V_{REF0}) selection bits. 00000: $V_{REF0} = V_{REF}$ 00001~11111: $V_{REF0} = V_{REF} * CM0RS[4:0] / 32$	R/W	00000b
4:3	CM0NS[1:0]	CMP0 negative input selection bits 00: CM0N0 is comparator negative input pin, and isolate GPIO function 01: CM0N1 is comparator negative input pin, and isolate GPIO function 10: CM0N2 is comparator negative input pin, and isolate GPIO function 11: V_{REF0} . CM0N0/CM0N1/CM0N2 pins are GPIO mode.	R/W	00b
2:1	CM0PS[1:0]	CMP0 positive input pin selection bits 00: CM0P0 is comparator positive input pin, and isolate GPIO function 01: CM0P1 is comparator positive input pin, and isolate GPIO function 10: CM0P2 is comparator positive input pin, and isolate GPIO function 11: V_{REF0} . CM0P0/CM0P1/CM0P2 pins are GPIO mode.	R/W	00b
0	CM0EN	CMP0 enable bit. 0: Disable (CM0P0/1/2, CM0N0/1/2, CM0O are GPIO mode) 1: Enable	R/W	0

9.4.2 CMP Internal Reference Voltage Source register (CMP_VIREF)

Address Offset: 0x08

*** Note:** If ADC internal voltage reference, CMP internal voltage reference, or DAC internal voltage reference is used, ADC_ADM2 SHALL be set to 0x3.

Bit	Name	Description	Attribute	Reset
31:3	Reserved		R	0
2:1	CMPIREF[1:0]	Comparator internal reference voltage (V_{IREF}) source 00: VDD 01: internal 1.5V 10: internal 2V 11: Internal 3V	RW	00b
0	CMPIREFEN	Comparator internal reference voltage (V_{IREF}) enable 0: Disable 1: Enable	RW	0

9.4.3 CMP Output Status register (CMP_OS)

Address Offset: 0x0C

*** Note:** CMnOUT is comparator raw output without latch. It varies depend on the comparator process result. But the CMnIF is latch comparator output result. It must be cleared by program.

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1	CM1OUT	CMP1 output flag bit. 0: $V_{1+} < V_{1-}$ 1: $V_{1+} > V_{1-}$	R	0
0	CM0OUT	CMP0 output flag bit. 0: $V_{0+} < V_{0-}$ 1: $V_{0+} > V_{0-}$	R	0

9.4.4 CMP Interrupt Enable register (CMP_IE)

Address Offset: 0x10

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
1	CM1IE	CMP1 interrupt enable control bit. 0: Disable 1: Enable	RW	0
0	CM0IE	CMP0 interrupt enable control bit. 0: Disable 1: Enable	RW	0

9.4.5 CMP Raw Interrupt Status register (CMP_RIS)

Address offset: 0x14

This register indicates the status for comparator raw interrupts. A CMP0/1 interrupt is sent to the interrupt controller if the corresponding CMnIE bit is set.

*** Note: CMnOUT is comparator raw output without latch. It varies depend on the comparator process result. But the CMnIF is latch comparator output result. It must be cleared by program.**

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1	CM1IF	CMP1 raw interrupt flag 0: No interrupt on CMP1 1: Interrupt requirements met on CMP1	R	0
0	CM0IF	CMP0 raw interrupt flag 0: No interrupt on CMP0 1: Interrupt requirements met on CMP0	R	0

9.4.6 CMP Interrupt Clear register (CMP_IC)

Address offset: 0x18

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1	CM1IC	Selects interrupt flag on CMP1 to be cleared 0: No effect 1: Clear interrupt flag on CMP1	W	0
0	CM0IC	Selects interrupt flag on CMP0 to be cleared 0: No effect 1: Clear interrupt flag on CMP0	W	0

9.4.7 CMP Output Debounce register (CMP_DB)

Address offset: 0x1C

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6:4	CM1DB[2:0]	Count for CMP1 output debounce time 000: No CMP1 output debounce time 001~111: CMP1 output debounce time = $2^{CM1DB[2:0]} * CMP1_PCLK$ cycle	W/R	000b
3	Reserved		R	0
2:0	CM0DB[2:0]	Count for CMP0 output debounce time 000: No CMP0 output debounce time 001~111: CMP0 output debounce time = $2^{CM0DB[2:0]} * CMP0_PCLK$ cycle	W/R	000b

10 16-BIT TIMER WITH CAPTURE FUNCTION

10.1 OVERVIEW

The microcontroller builds in four 16-bit timers (CT16B0/1/2/5). Each Counter/timer is designed to count cycles of the peripheral clock (PCLK) or an externally supplied clock and can optionally generate interrupts or perform other actions at specified timer values based on four match registers. Each counter/timer also includes one capture input to trap the timer value when an input signal transitions, optionally generating an interrupt.

In PWM mode, up to 12 match and a global match registers can be used to provide a single-edge controlled PWM output on the match output pins.

	CT16B0	CT16B1	CT16B2	CT16B5
Clock Source	HCLK PLL_VCO			HCLK PLL_VCO ELS
Counter Mode	Up/Down/ Center-aligned counting	Up/Down/ Center-aligned counting	Up/Down/ Center-aligned counting	Up/Down/ Center-aligned counting
PWM	PWM0/1/2/3	PWM0/1/2/3	PWM0/1/2/3+ PWM0N/1N/2N/3N	PWM0/1/2/3
PCLK	max 120MHz	max 120MHz	max 120MHz	max 120MHz

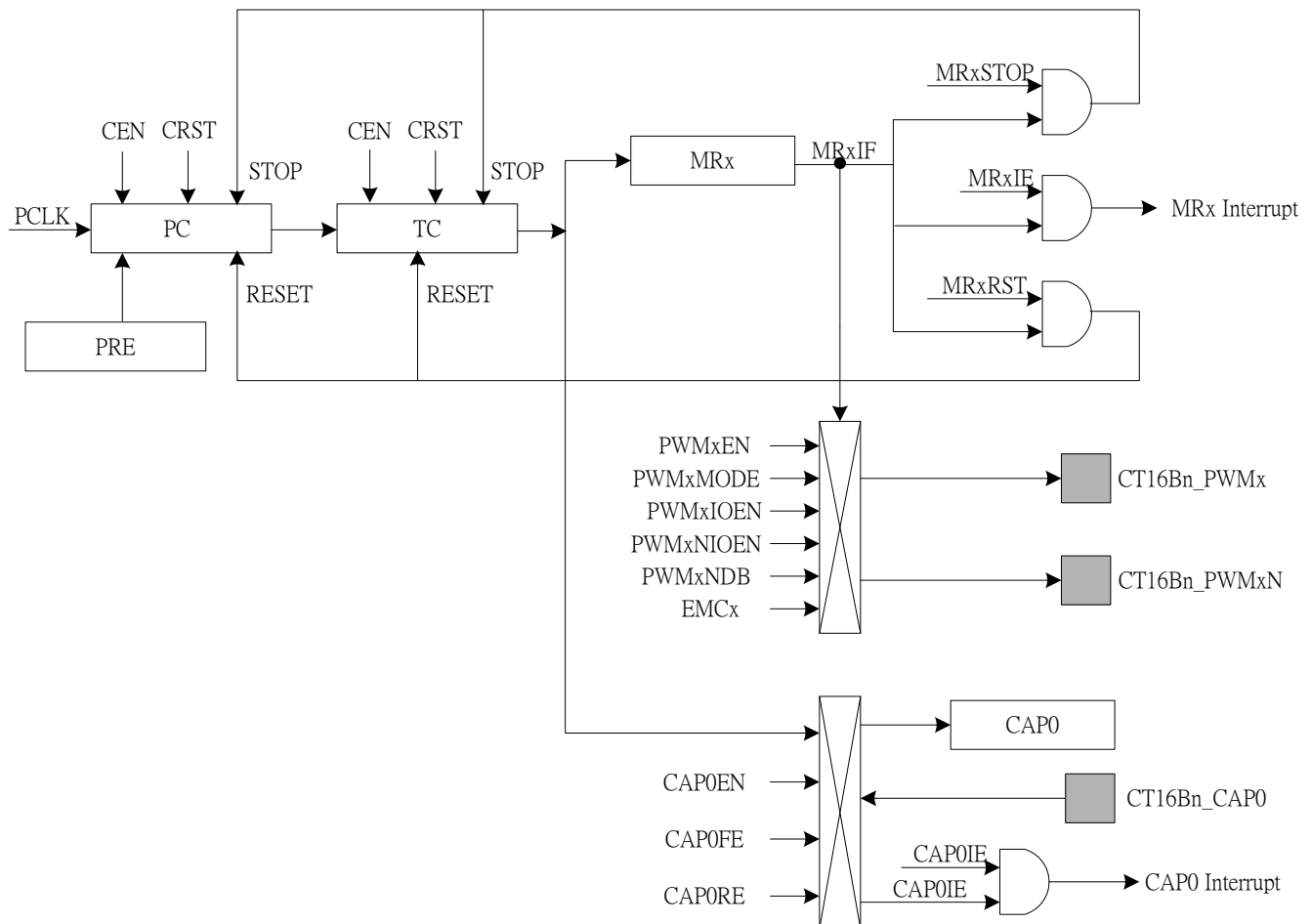
10.2 FEATURES

- 16-bit counter/timers with a programmable 8-bit prescaler.
- Counter or timer operation
- 16-bit capture channels that can take a snapshot of the timer value when an input signal transitions. A capture event may also optionally generate an interrupt.
- The timer and prescaler may be configured to be cleared on a designated capture event. This feature permits easy pulse-width measurement by clearing the timer on the leading edge of an input pulse and capturing the timer value on the trailing edge.
- 16-bit match registers that allow:
 - Continuous operation with optional interrupt generation on match.
 - Stop timer on match with optional interrupt generation.
 - Reset timer on match with optional interrupt generation.
- Up to 4 (CT16B0, CT16B1, CT16B2, CT16B5) PWM outputs corresponding to match registers with the following capabilities:
 - Set LOW on match.
 - Set HIGH on match.
 - Toggle on match.
 - Do nothing on match.
- For CT16B2, up to 4 complete inverse waveform of the PWM signals, and builds in programmable dead-band function.

10.3 PIN DESCRIPTION

Pin Name	Type	Description	GPIO Configuration
CT16Bn_CAP0	I	Capture channel input 0	Depends on GPIO _n _CFG
CT16Bn_PWMx	O	Output channel x of Match/PWM output.	
CT16Bn_PWMxN	O	Inverse Output channel of Match/PWMx output.	

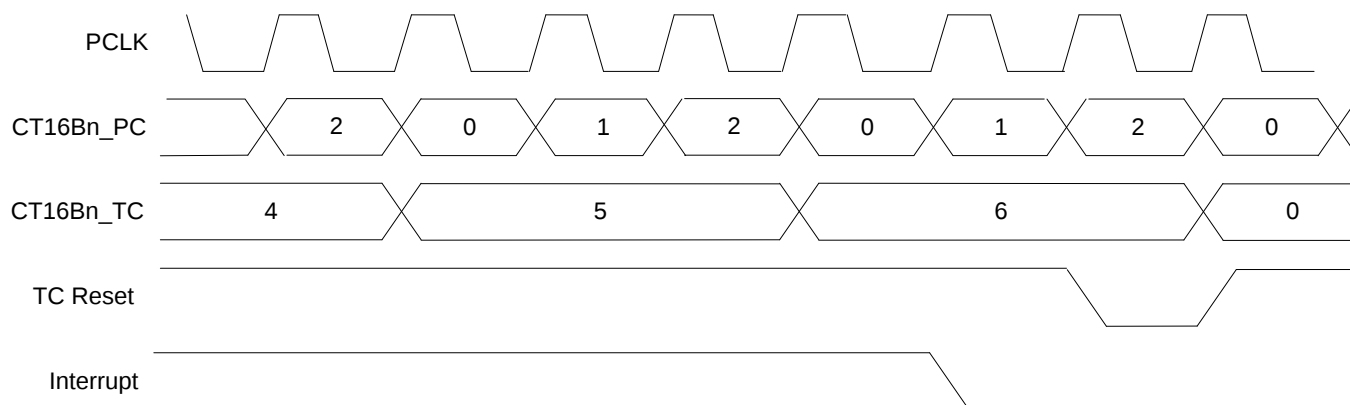
10.4 BLOCK DIAGRAM



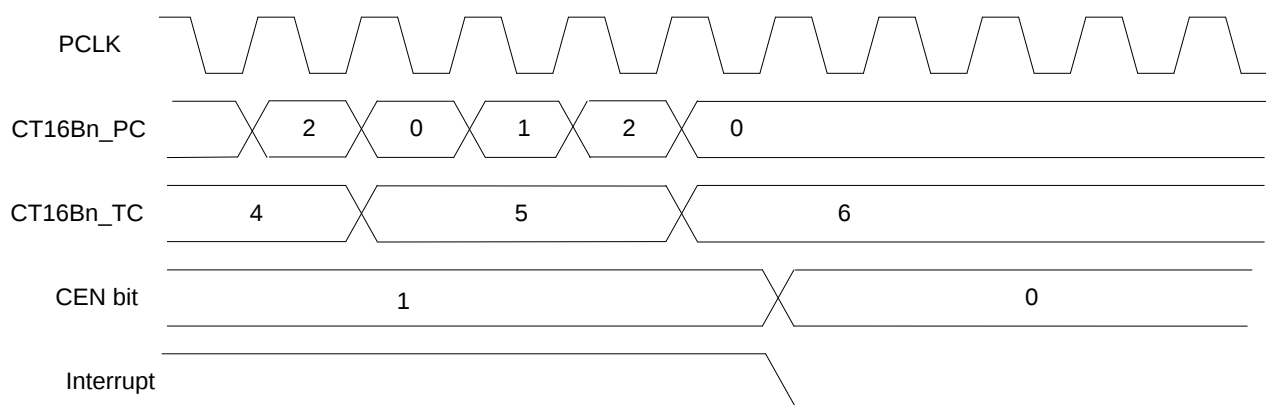
10.5 TIMER OPERATION

10.5.1 Edge-aligned Up-counting Mode

The following figure shows a timer configured to reset the count (TC) and generate an interrupt on match in Edge-aligned up-counting mode. The [CT16Bn_PRE](#) register is set to 2, and the [CT16Bn_MRx](#) register is set to 6. At the end of the timer cycle where the match occurs, the timer count is reset. This gives a full length cycle to the match value. The interrupt indicating that a match occurred is generated in the next clock after the timer reached the match value.



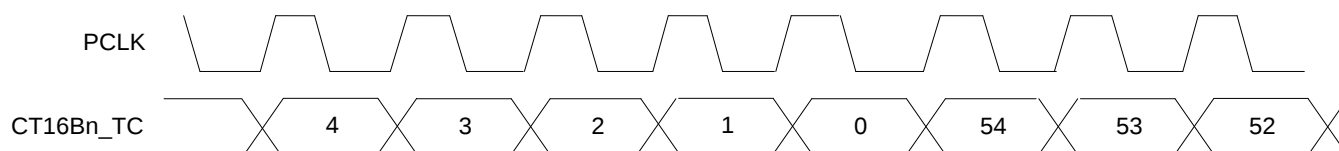
The following figure shows a timer configured to stop and generate an interrupt on match in Edge-aligned up-counting mode. The [CT16Bn_PRE](#) register is set to 2, and the [CT16Bn_MRx](#) register is set to 6. In the next clock after the timer reaches the match value, the CEN bit in [CT16Bn_TMRCTRL](#) register is cleared, and the interrupt indicating that a match occurred is generated.



10.5.2 Edge-aligned Down-counting Mode

The timer count TC[15:0] will be reset to the value of CT16Bn_MR9 after resetting counter or TC reaches 0. Besides, TC is blocked while the value of CT16Bn_MR9 is zero.

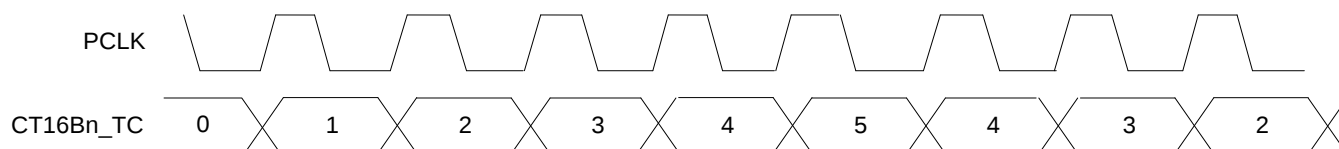
The following figure shows a timer configured to reset the count in Edge-aligned down-counting mode. The [CT16Bn_PRE](#) register is set to 0, and the [CT16Bn_MR9](#) register is set to 54. After TC reaches 0, the timer count is reset and loaded from the value of CT16Bn_MR9.



10.5.3 Center-aligned Counting Mode

In Center-aligned counting mode, TC counts up from 0 to the value of CT16Bn_MR9, and then counts down to 0 alternatively. Besides, TC is blocked while the value of CT16Bn_MR9 is zero.

The following figure shows a timer in Center-aligned counting mode. The [CT16Bn_PRE](#) register is set to 0, and the [CT16Bn_MR9](#) register is set to 5.



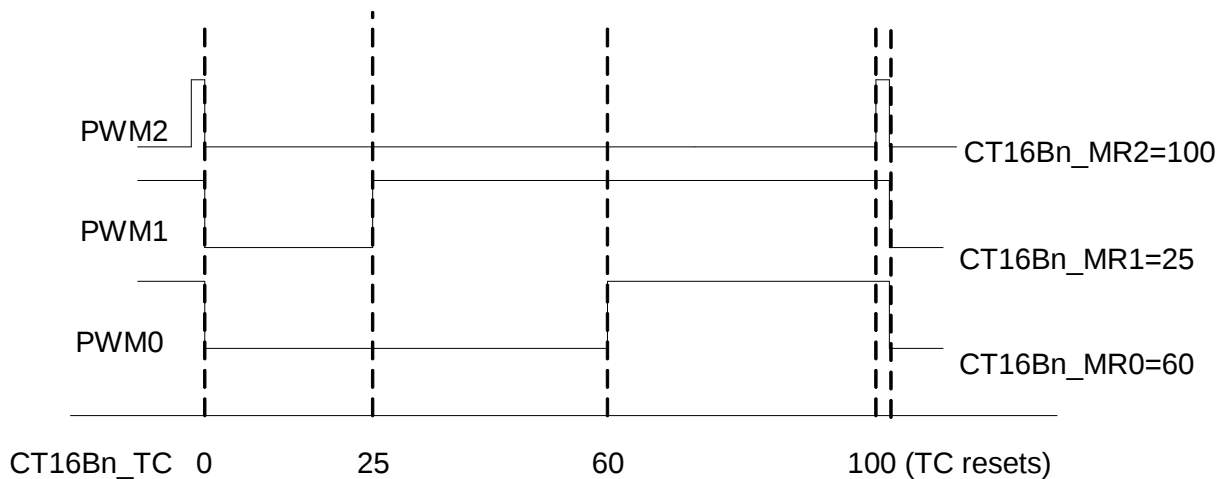
10.6 PWM

10.6.1 PWM Mode 1

- PWMn is 0 when $TC < MRn$ during Up-counting period
- PWMn is 0 when $TC \leq MRn$ during Down-counting period

Take Edge-aligned up-counting Mode as example,

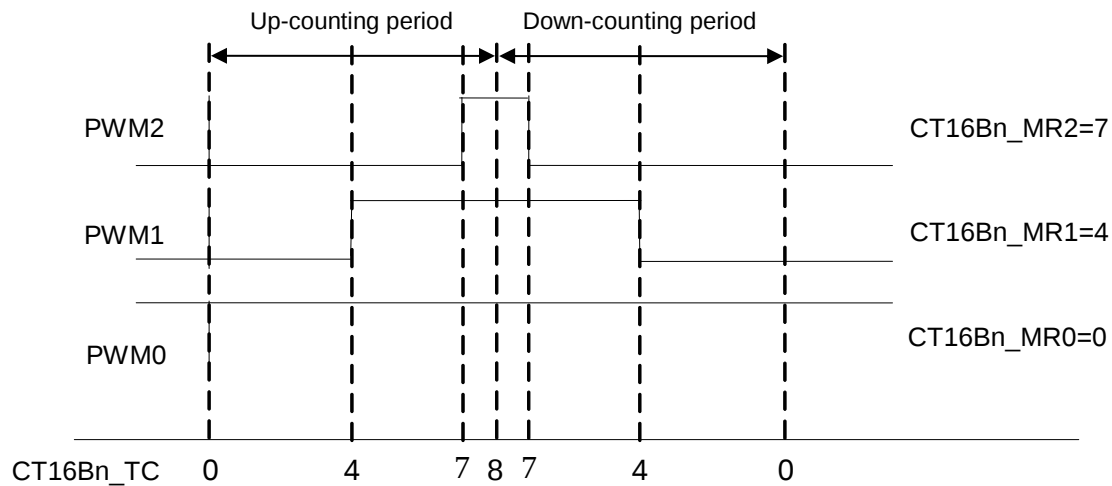
1. All single edge controlled PWM outputs go LOW at the beginning of each PWM cycle (timer is set to zero) unless their match value in CT16Bn_MR0~3 registers is equal to zero.
2. Each PWM output will go HIGH when its match value is reached. If no match occurs, the PWM output remains continuously LOW.
3. If a match value larger than the PWM cycle length is written to the CT16Bn_MR0~3 registers, and the PWM signal is HIGH already, then the PWM signal will be cleared on the next start of the next PWM cycle.
4. If a match register contains the same value as the timer reset value (the PWM cycle length), then the PWM output will be reset to LOW on the next clock tick. Therefore, the PWM output will always consist of a one clock tick wide positive pulse with a period determined by the PWM cycle length.
5. If a match register is set to zero, then the PWM output will go HIGH the first time the timer goes back to zero and will stay HIGH continuously.



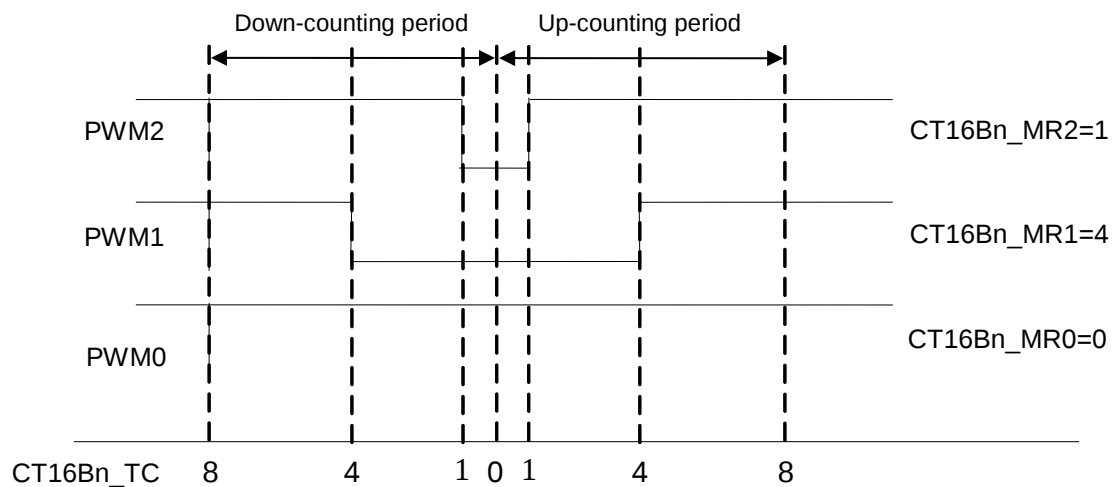
*** Note:** When the match outputs are selected to perform as PWM outputs, the timer reset (MRnRST) and timer stop (MRnSTOP) bits in [CT16Bn_MCTRL](#) register must be set to zero except for the match register setting the PWM cycle length. For this register, set the MRnR bit to one to enable the timer reset when the timer value matches the value of the corresponding match register.

The following figure shows the PWM mode 1 wave form in Center-aligned counting mode.

Case1: The [CT16Bn_PRE](#) register is set to 0, the [CT16Bn_MR9](#) register is set to 8, the [CT16Bn_MR2](#) register is set to 7, the [CT16Bn_MR1](#) register is set to 4, and the [CT16Bn_MR0](#) register is set to 0.



Case 2: The [CT16Bn_PRE](#) register is set to 0, the [CT16Bn_MR9](#) register is set to 8, the [CT16Bn_MR2](#) register is set to 1, the [CT16Bn_MR1](#) register is set to 4, and the [CT16Bn_MR0](#) register is set to 0.



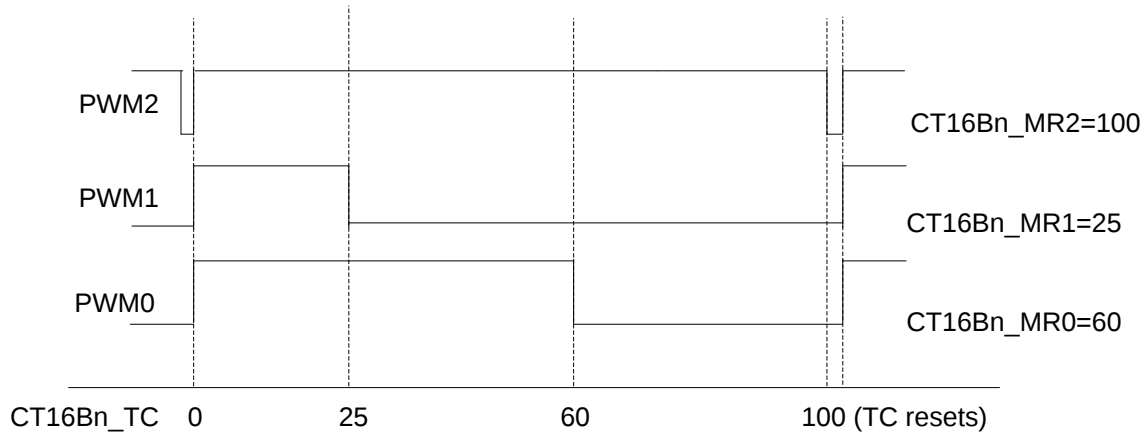
10.6.2 PWM Mode 2

- PWMn is 1 when TC<MRn during Up-counting period
- PWMn is 1 when TC≤MRn during Down-counting period
- Not support in Center-aligned counting mode

Take Edge-aligned up-counting Mode as example,

1. All single edge controlled PWM outputs go HIGH at the beginning of each PWM cycle (timer is set to zero) unless their match value in CT16Bn_MR0~3 registers is equal to zero.
2. Each PWM output will go LOW when its match value is reached. If no match occurs, the PWM output remains continuously HIGH.
3. If a match value larger than the PWM cycle length is written to the CT16Bn_MR0~3 registers, and the PWM signal is LOW already, then the PWM signal will go HIGH on the next start of the next PWM cycle.
4. If a match register contains the same value as the timer reset value (the PWM cycle length), then the PWM output will be reset to HIGH on the next clock tick. Therefore, the PWM output will always consist of a one clock tick wide low pulse with a period determined by the PWM cycle length.

5. If a match register is set to zero, then the PWM output will go LOW the first time the timer goes back to zero and will stay LOW continuously.



*** Note:** When the match outputs are selected to perform as PWM outputs, the timer reset (MRnRST) and timer stop (MRnSTOP) bits in [CT16Bn_MCTRL](#) register must be set to zero except for the match register setting the PWM cycle length. For this register, set the MRnR bit to one to enable the timer reset when the timer value matches the value of the corresponding match register.

10.7 INVERSE PWM OUTPUT WITH DEAD-BAND PERIOD

The CT16B2_PWMm builds in inverse output function controlled by PWMmNIOEN[1:0] bits in [CT16Bn_PWMCTRL](#) register.

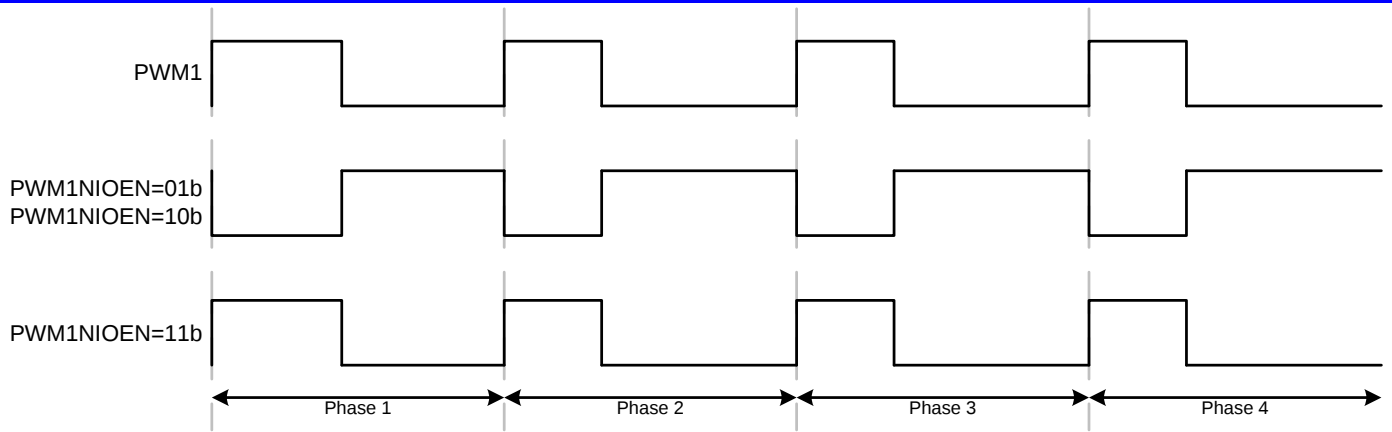
- When PWMmNIOEN[1:0] = 00b, PWMmN pin is GPIO mode.
- When PWMmNIOEN[1:0] = 01b, PWMmN pin changes to inverse PWM output pin which outputs the inverse PWM signal of PWMm with dead-band period, but same High signal during dead-band period.
- When PWMmNIOEN[1:0] = 10b, PWMmN pin changes to inverse PWM output pin which outputs the inverse PWMm signal with dead-band period, but same Low signal during dead-band period.
- When PWMmNIOEN[1:0] = 11b, PWMmN pin changes to non-inverse PWM output pin which outputs the same PWMm signal with dead-band period.

The dead-band period is symmetrical at left-right terminal of PWM high pulse width, and the PWM dead-band period is controlled by [CT16Bn_PWMmNDB](#) register. This register is only usable when PWMmNIOEN[1:0] ≠ 00b, and the PWM dead-band function is disabled when the value of this register is 0.

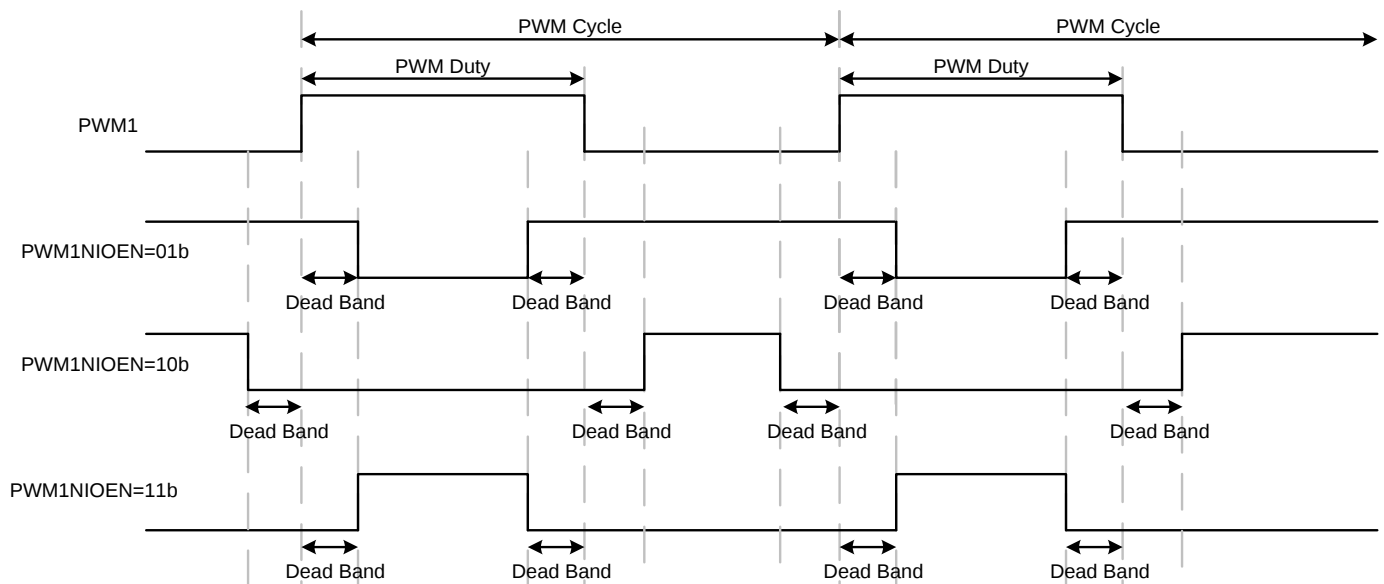
If DB = 1, the dead-band period is set as $1 \times \text{CT16Bn_PCLK} \times (\text{PR} + 1)$, $1 \times \text{CT16Bn_PCLK} \times (\text{PR} + 1)$ dead-band is in the left side of PWM high pulse, and the other side also includes one dead-band duration, so the total dead-band period is $2 \times \text{CT16Bn_PCLK} \times (\text{PR} + 1)$.

To take care the PWM high pulse width with dead-bane function is necessary. Recommend the dead-band period less than PWM high pulse width, or the PWM high pulse width disappears.

The PWMmN waveform without dead-band:



The PWMmN waveform with dead-band:



- * **Note:** If the dead-band period is longer than PWM duty, the PWM1N is no output.
- * **Note:** When the dead-band function is enabled in Center-aligned mode, and MR9RST=1, CT16Bn_PWMxN will always output "0"
- * **Note:** When the dead-band function is enabled
 - System will reset TC refer to MR9RST ONLY in Up-counting mode
 - In Down counting mode, TC[15:0] will be reloaded from CT16Bn_MR9 after resetting counter
 - System will reset TC refer to MR9RST ONLY in Center-aligned mode

10.8 CT16Bn REGISTERS

Base Address: 0x4000 0000 (CT16B0)
 0x4000 2000 (CT16B1)
 0x4000 A000 (CT16B2)
 0x4000 6000 (CT16B5)

10.8.1 CT16Bn Timer Control register (CT16Bn_TMRCTRL) (n=0,1,2)

Address Offset: 0x00

*** Note: CEN bit shall be set at last!**

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6:4	CM[2:0]	Counting mode selection 000: Edge-aligned Up-counting mode 001: Edge-aligned Down-counting mode 010: Center-aligned mode 1. The match interrupt flag is set during the down-counting period 100: Center-aligned mode 2. The match interrupt flag is set during the up-counting period 110: Center-aligned mode 3. The match interrupt flag is set during both up-counting and down-counting period Other: Reserved	R/W	000b
3	Reserved		R	0
2	CLKSEL	CT16Bn PCLK source. 0: HCLK 1: PLL_VCO	R/W	0
1	CRST	Counter Reset. 0: Disable counter reset. 1: Timer Counter and the Prescale Counter are synchronously reset on the next positive edge of PCLK. This is cleared by HW when the counter reset operation finishes.	R/W	0
0	CEN	Counter Enable 0: Disable Counter. 1: Enable Timer Counter and Prescale Counter for counting.	R/W	0

10.8.2 CT16Bn Timer Control register (CT16Bn_TMRCTRL) (n=5)

Address Offset: 0x00

*** Note: CEN bit shall be set at last!**

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6:4	CM[2:0]	Counting mode selection 000: Edge-aligned Up-counting mode 001: Edge-aligned Down-counting mode 010: Center-aligned mode 1. The match interrupt flag is set during the down-counting period 100: Center-aligned mode 2. The match interrupt flag is set during the up-counting period 110: Center-aligned mode 3. The match interrupt flag is set during both	R/W	000b

		up-counting and down-counting period Other: Reserved		
3:2	CLKSEL	CT16Bn PCLK source. 00: HCLK 01: PLL_VCO 10: ELS X'tal 11: Reserved	R/W	0
1	CRST	Counter Reset. 0: Disable counter reset. 1: Timer Counter and the Prescale Counter are synchronously reset on the next positive edge of PCLK. This is cleared by HW when the counter reset operation finishes.	R/W	0
0	CEN	Counter Enable 0: Disable Counter. 1: Enable Timer Counter and Prescale Counter for counting.	R/W	0

10.8.3 CT16Bn Timer Counter register (CT16Bn_TC) (n=0,1,2,5)

Address Offset: 0x04

In Edge-aligned up-counting mode (CM[2:0]=000b), unless it is reset before reaching its upper limit, the TC will count up to the value 0x0000FFFF and then wrap back to the value 0x00000000. This event does not cause an interrupt, but a Match register can be used to detect an overflow if needed.

In Edge-aligned down-counting mode (CM[2:0]=001b), the TC[15:0] should be reset to the value of CT16Bn_MR9 after resetting counter (SW set CRST to 1).

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	TC[15:0]	Timer Counter	R/W	0

10.8.4 CT16Bn Prescale register (CT16Bn_PRE) (n=0,1,2,5)

Address Offset: 0x08

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	PR[7:0]	Prescale max value.	R/W	0

10.8.5 CT16Bn Prescale Counter register (CT16Bn_PC) (n=0,1,2,5)

Address Offset: 0x0C

The 8-bit Prescale Counter controls division of PCLK by some constant value before it is applied to the Timer Counter. This allows control of the relationship between the resolution of the timer and the maximum time before the timer overflows. The Prescale Counter is incremented on every PCLK. When it reaches the value stored in the Prescale Register, the Timer Counter is incremented, and the Prescale Counter is reset on the next PCLK. This causes the TC to increment on every PCLK when PR = 0, every 2 PCLKs when PR = 1, etc.

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	PC[7:0]	Prescale Counter	R/W	0

10.8.6 CT16Bn Count Control register (CT16Bn_CNTCTRL) (n=0,1,2,5)

Address Offset: 0x10

This register is used to select between Timer and Counter mode, and in Counter mode to select the pin and edges for counting.

When Counter Mode is chosen as a mode of operation, the CAP input is sampled on every rising edge of the PCLK clock. After comparing two consecutive samples of this CAP input, one of the following four events is recognized: rising edge, falling edge, either of edges or no changes in the level of the selected CAP input. Only if the identified event occurs, and the event corresponds to the one selected by CTM bits in this register, will the Timer Counter register be incremented.

Effective processing of the externally supplied clock to the counter has some limitations. Since two successive rising edges of the PCLK clock are used to identify only one edge on the CAP selected input, the frequency of the CAP input cannot exceed one half of the PCLK clock. Consequently, the duration of the HIGH/LOW levels on the same CAP input in this case cannot be shorter than $1/(2 \times \text{PCLK})$.

*** Note: If Counter mode is selected in the CNTCTRL register, bit 2~0 of Capture Control (CAPCTRL) register must be programmed as 0x0.**

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1:0	CTM[1:0]	Counter/Timer Mode. This field selects which rising PCLK edges can increment Timer's Prescale Counter (PC), or clear PC and increment Timer Counter (TC). 00: Timer Mode: every rising PCLK edge 01: Counter Mode: TC is incremented on rising edges on the CAP0 input. 10: Counter Mode: TC is incremented on falling edges on the CAP0 input. 11: Counter Mode: TC is incremented on both edges on the CAP0 input.	R/W	0

10.8.7 CT16Bn Match Control register (CT16Bn_MCTRL) (n=0,1,2,5)

Address Offset: 0x14

*** Note: When the dead-band function is enabled in Center-aligned mode, and MR9RST=1, CT16Bn_PWMxN will always output "0"**

*** Note: When the dead-band function is enabled**

- System will reset TC refer to MR9RST ONLY in Up-counting mode
- In Down counting mode, TC[15:0] will be reloaded from CT16Bn_MR9 after resetting counter
- System will reset TC refer to MR9RST ONLY in Center-aligned mode

Bit	Name	Description	Attribute	Reset
31	MR9STOP	Stop MR9: TC and PC will stop and CEN bit will be cleared if MR9 matches TC. 0: Disable 1: Enable	R/W	0
30	MR9RST	Enable reset TC when MR9 matches TC. 0: Disable 1: Enable	R/W	0
29	MR9IE	Enable generating an interrupt based on CM[2:0] when MR9 matches the value in the TC. 0: Disable 1: Enable	R/W	0
28:12	Reserved		R	0
11	MR3STOP	Stop MR3: TC and PC will stop and CEN bit will be cleared if MR3 matches TC.	R/W	0

		0: Disable 1: Enable		
10	MR3RST	Enable reset TC when MR3 matches TC. 0: Disable 1: Enable	R/W	0
9	MR3IE	Enable generating an interrupt based on CM[2:0] when MR3 matches the value in the TC. 0: Disable 1: Enable	R/W	0
8	MR2STOP	Stop MR2: TC and PC will stop and CEN bit will be cleared if MR2 matches TC. 0: Disable 1: Enable	R/W	0
7	MR2RST	Enable reset TC when MR2 matches TC. 0: Disable 1: Enable	R/W	0
6	MR2IE	Enable generating an interrupt based on CM[2:0] when MR2 matches the value in the TC. 0: Disable 1: Enable	R/W	0
5	MR1STOP	Stop MR1: TC and PC will stop and CEN bit will be cleared if MR1 matches TC. 0: Disable 1: Enable	R/W	0
4	MR1RST	Enable reset TC when MR1 matches TC. 0: Disable 1: Enable	R/W	0
3	MR1IE	Enable generating an interrupt based on CM[2:0] when MR1 matches the value in the TC. 0: Disable 1: Enable	R/W	0
2	MR0STOP	Stop MR0: TC and PC will stop and CEN bit will be cleared if MR0 matches TC. 0: Disable 1: Enable	R/W	0
1	MR0RST	Enable reset TC when MR0 matches TC. 0: Disable 1: Enable	R/W	0
0	MR0IE	Enable generating an interrupt based on CM[2:0] when MR0 matches the value in the TC. 0: Disable 1: Enable	R/W	0

10.8.8 CT16Bn Match register 0~3 (CT16Bn_MR0~3) (n=0,1,2,5)

Address Offset: 0x18, 0x1C, 0x20, 0x24

The Match register values are continuously compared to the Timer Counter (TC) value. When the two values are equal, actions can be triggered automatically. The action possibilities are to generate an interrupt, reset the Timer Counter, or stop the timer. Actions are controlled by the settings in the CT16Bn_MCTRL register.

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	MR[15:0]	Timer counter match value	R/W	0

10.8.9 CT16Bn Capture Control register (CT16Bn_CAPCTRL) (n=0,1,2,5)

Address Offset: 0x28

The Capture Control register is used to control whether the Capture register is loaded with the value in the Counter/timer when the capture event occurs, and whether an interrupt is generated by the capture event. Setting both the rising and falling bits at the same time is a valid configuration, resulting in a capture event for both edges.

*** Note: HW will switch I/O Configuration directly when CAP0EN=1.**

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	CAP0EN	Capture 0 function enable bit 0: Disable 1: Enable.	R/W	0
2	CAP0IE	Interrupt on CT16Bn_CAP0 event: a CAP0 load due to a CT16Bn_CAP0 event will generate an interrupt. 0: Disable 1: Enable	R/W	0
1	CAP0FE	Capture on CT16Bn_CAP0 falling edge: a sequence of 1 then 0 on CT16Bn_CAP0 will cause CAP0 to be loaded with the contents of TC. 0: Disable 1: Enable	R/W	0
0	CAP0RE	Capture on CT16Bn_CAP0 rising edge: a sequence of 0 then 1 on CT16Bn_CAP0 will cause CAP0 to be loaded with the contents of TC. 0: Disable 1: Enable	R/W	0

10.8.10 CT16Bn Capture 0 register (CT16Bn_CAP0) (n=0,1,2,5)

Address Offset: 0x2C

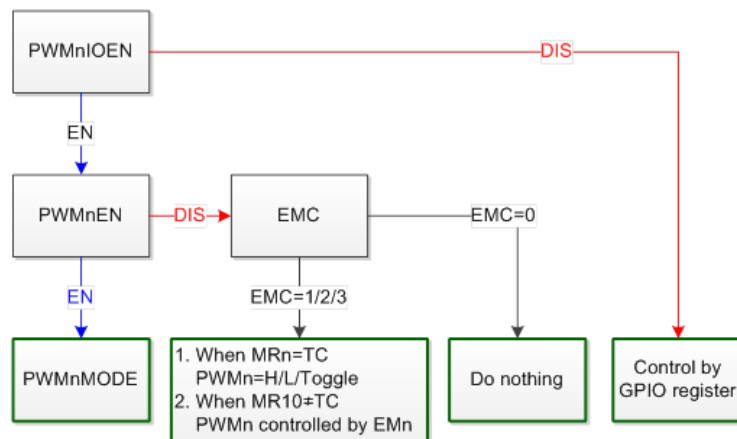
Each Capture register is associated with a device pin and may be loaded with the counter/timer value when a specified event occurs on that pin. The settings in the Capture Control register determine whether the capture function is enabled, and whether a capture event happens on the rising edge of the associated pin, the falling edge, or on both edges.

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	CAP0[15:0]	Timer counter capture value	R	0

10.8.11 CT16Bn External Match register (CT16Bn_EM) (n=0,1,2,5)

Address Offset: 0x30

The External Match register provides both control and status of CT16B0_PWM[2:0]. If the match outputs are configured as PWM output, the function of the external match registers is determined by the [PWM rules](#).



Bit	Name	Description	Attribute	Reset
-----	------	-------------	-----------	-------

31:12	Reserved		R	0
11:10	EMC3[1:0]	Determines the functionality of CT16Bn_PWM3 00: Do Nothing. 01: CT16Bn_PWM3 pin is LOW 10: CT16Bn_PWM3 pin is HIGH 11: Toggle CT16Bn_PWM3 pin.	R/W	0
9:8	EMC2[1:0]	Determines the functionality of CT16Bn_PWM2. 00: Do Nothing. 01: CT16Bn_PWM2 pin is LOW 10: CT16Bn_PWM2 pin is HIGH 11: Toggle CT16Bn_PWM2 pin.	R/W	0
7:6	EMC1[1:0]	Determines the functionality of CT16Bn_PWM1. 00: Do Nothing. 01: CT16Bn_PWM1 pin is LOW 10: CT16Bn_PWM1 pin is HIGH. 11: Toggle CT16Bn_PWM1.	R/W	0
5:4	EMC0[1:0]	Determines the functionality of CT16Bn_PWM0. 00: Do Nothing. 01: CT16Bn_PWM0 pin is LOW 10: CT16Bn_PWM0 pin is HIGH 11: Toggle CT16Bn_PWM0.	R/W	0
3	EM3	When EMC3≠00b and MR3≠TC, this bit will drive the state of CT16Bn_PWM3 output.	R/W	0
2	EM2	When EMC2≠00b and MR2≠TC, this bit will drive the state of CT16Bn_PWM2 output.	R/W	0
1	EM1	When EMC1≠00b and MR1≠TC, this bit will drive the state of CT16Bn_PWM1 output.	R/W	0
0	EM0	When EMC0≠00b and MR0≠TC, this bit will drive the state of CT16Bn_PWM0 output.	R/W	0

10.8.12 CT16Bn PWM Control register (CT16Bn_PWMCTRL) (n=0,1,5)

Address Offset: 0x34

The PWM Control register is used to configure the match outputs as PWM outputs. Each match output can be independently set to perform either as PWM output or as match output whose function is controlled by [CT16Bn_EM](#) register.

For CT16B0, CT16B1, and CT16B5, a maximum of 4 single edge controlled PWM outputs can be selected via the CT16Bn_PWMCTRL[3:0] bits. One additional match register determines the PWM cycle length. When a match occurs in any of the other match registers, the PWM output is set to HIGH. The timer is reset by the match register that is configured to set the PWM cycle length. When the timer is reset to zero, all currently HIGH match outputs configured as PWM outputs are cleared.

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23	PWM3IOEN	CT16Bn_PWM3/GPIO selection bit 0: CT16Bn_PWM3 pin act as GPIO 1: CT16Bn_PWM3 pin act as match output, and output signal depends on PWM2EN bit.	R/W	0
22	PWM2IOEN	CT16Bn_PWM2/GPIO selection bit 0: CT16Bn_PWM2 pin act as GPIO 1: CT16Bn_PWM2 pin act as match output, and output signal depends on PWM2EN bit.	R/W	0
21	PWM1IOEN	CT16Bn_PWM1/GPIO selection bit 0: CT16Bn_PWM1 pin act as GPIO 1: CT16Bn_PWM1 pin act as match output, and output signal depends on PWM1EN bit.	R/W	0
20	PWM0IOEN	CT16Bn_PWM0/GPIO selection bit 0: CT16Bn_PWM0 pin act as GPIO	R/W	0

		1: CT16Bn_PWM0 pin act as match output, and output signal depends on PWM0EN bit.		
19:12	Reserved		R	0
11:10	PWM3MODE[1:0]	PWM3 output mode 00: PWM mode 1 01: PWM mode 2 10: Forced to 0. 11: Forced to 1.	R/W	0
9:8	PWM2MODE[1:0]	PWM2 output mode 00: PWM mode 1 01: PWM mode 2 10: Forced to 0. 11: Forced to 1.	R/W	0
7:6	PWM1MODE[1:0]	PWM1 output mode 00: PWM mode 1 01: PWM mode 2 10: Forced to 0. 11: Forced to 1.	R/W	0
5:4	PWM0MODE[1:0]	PWM0 output mode 00: PWM mode 1 01: PWM mode 2 10: Forced to 0. 11: Forced to 1.	R/W	0
3	PWM3EN	PWM3 enable 0: CT16Bn_PWM3 is controlled by EM3 1: PWM mode is enabled for CT16Bn_PWM3.	R/W	0
2	PWM2EN	PWM2 enable 0: CT16Bn_PWM2 is controlled by EM2. 1: PWM mode is enabled for CT16Bn_PWM2.	R/W	0
1	PWM1EN	PWM1 enable 0: CT16Bn_PWM1 is controlled by EM1. 1: PWM mode is enabled for CT16Bn_PWM1.	R/W	0
0	PWM0EN	PWM0 enable 0: CT16Bn_PWM0 is controlled by EM0. 1: PWM mode is enabled for CT16Bn_PWM0.	R/W	0

10.8.13 CT16Bn PWM Control register (CT16Bn_PWMCTRL) (n=2)

Address Offset: 0x34

The PWM Control register is used to configure the match outputs as PWM outputs. Each match output can be independently set to perform either as PWM output or as match output whose function is controlled by [CT16Bn_EM](#) register.

For CT16B2, a maximum of 4 single edge controlled PWM outputs can be selected via the CT16Bn_PWMCTRL[3:0] bits. One additional match register determines the PWM cycle length. When a match occurs in any of the other match registers, the PWM output is set to HIGH. The timer is reset by the match register that is configured to set the PWM cycle length. When the timer is reset to zero, all currently HIGH match outputs configured as PWM outputs are cleared.

Bit	Name	Description	Attribute	Reset
31:30	PWM3NIOEN[1:0]	CT16Bn_PWM3N/GPIO selection bit 00: CT16Bn_PWM3N pin act as GPIO 01: CT16Bn_PWM3N pin outputs the inverse signal with dead-band of CT16Bn_PWM3, but same High signal during dead-band period. 10: CT16Bn_PWM3N pin outputs the inverse signal with dead-band of CT16Bn_PWM3, but same Low signal during dead-band period. 11: CT16Bn_PWM3N pin outputs the same signal with dead-band of CT16Bn_PWM3.	R/W	0
29:28	PWM2NIOEN[1:0]	CT16Bn_PWM2N/GPIO selection bit 00: CT16Bn_PWM2N pin act as GPIO 01: CT16Bn_PWM2N pin outputs the inverse signal with dead-band of CT16Bn_PWM2, but same High signal during dead-band period. 10: CT16Bn_PWM2N pin outputs the inverse signal with dead-band of CT16Bn_PWM2, but same Low signal during dead-band period. 11: CT16Bn_PWM2N pin outputs the same signal with dead-band of	R/W	0

		CT16Bn_PWM2.		
27:26	PWM1NIOEN[1:0]	CT16Bn_PWM1N/GPIO selection bit 00: CT16Bn_PWM1N pin act as GPIO 01: CT16Bn_PWM1N pin outputs the inverse signal with dead-band of CT16Bn_PWM1, but same High signal during dead-band period. 10: CT16Bn_PWM1N pin outputs the inverse signal with dead-band of CT16Bn_PWM1, but same Low signal during dead-band period. 11: CT16Bn_PWM1N pin outputs the same signal with dead-band of CT16Bn_PWM1.	R/W	0
25:24	PWM0NIOEN[1:0]	CT16Bn_PWM0N/GPIO selection bit 00: CT16Bn_PWM0N pin act as GPIO 01: CT16Bn_PWM0N pin outputs the inverse signal with dead-band of CT16Bn_PWM0, but same High signal during dead-band period. 10: CT16Bn_PWM0N pin outputs the inverse signal with dead-band of CT16Bn_PWM0, but same Low signal during dead-band period. 11: CT16Bn_PWM0N pin outputs the same signal with dead-band of CT16Bn_PWM0.	R/W	0
23	PWM3IOEN	CT16Bn_PWM3/GPIO selection bit 0: CT16Bn_PWM3 pin act as GPIO 1: CT16Bn_PWM3 pin act as match output, and output signal depends on PWM3EN bit.	R/W	0
22	PWM2IOEN	CT16Bn_PWM2/GPIO selection bit 0: CT16Bn_PWM2 pin act as GPIO 1: CT16Bn_PWM2 pin act as match output, and output signal depends on PWM2EN bit.	R/W	0
21	PWM1IOEN	CT16Bn_PWM1/GPIO selection bit 0: CT16Bn_PWM1 pin act as GPIO 1: CT16Bn_PWM1 pin act as match output, and output signal depends on PWM1EN bit.	R/W	0
20	PWM0IOEN	CT16Bn_PWM0/GPIO selection bit 0: CT16Bn_PWM0 pin act as GPIO 1: CT16Bn_PWM0 pin act as match output, and output signal depends on PWM0EN bit.	R/W	0
19:12	Reserved		R	0
11:10	PWM3MODE[1:0]	PWM3 output mode 00: PWM mode 1 PWM3 is 0 when TC<MR3 during Up-counting period PWM3 is 0 when TC≤MR3 during Down-counting period 01: PWM mode 2 PWM3 is 1 when TC<MR3 during Up-counting period PWM3 is 1 when TC≤MR3 during Down-counting period 10: PWM3 is forced to 0. 11: PWM3 is forced to 1.	R/W	0
9:8	PWM2MODE[1:0]	PWM2 output mode 00: PWM mode 1 01: PWM mode 2 10: Forced to 0. 11: Forced to 1.	R/W	0
7:6	PWM1MODE[1:0]	PWM1 output mode 00: PWM mode 1 01: PWM mode 2 10: Forced to 0. 11: Forced to 1.	R/W	0
5:4	PWM0MODE[1:0]	PWM0 output mode 00: PWM mode 1 01: PWM mode 2 10: Forced to 0. 11: Forced to 1.	R/W	0
3	PWM3EN	PWM3 enable 0: CT16Bn_PWM2 is controlled by EM3. 1: PWM mode is enabled for CT16Bn_PWM3	R/W	0
2	PWM2EN	PWM2 enable 0: CT16Bn_PWM2 is controlled by EM2.	R/W	0

		1: PWM mode is enabled for CT16Bn_PWM2.		
1	PWM1EN	PWM1 enable 0: CT16Bn_PWM1 is controlled by EM1. 1: PWM mode is enabled for CT16Bn_PWM1.	R/W	0
0	PWM0EN	PWM0 enable 0: CT16Bn_PWM0 is controlled by EM0. 1: PWM mode is enabled for CT16Bn_PWM0.	R/W	0

10.8.14 CT16Bn Timer Raw Interrupt Status register (CT16Bn_RIS) (n=0,1,2,5)

Address Offset: 0x38

This register indicates the raw status for Timer/PWM interrupts. A Timer/PWM interrupt is sent to the interrupt controller if the corresponding bit in the CT16Bn_IE register is set.

Bit	Name	Description	Attribute	Reset
31:6	Reserved		R	0
5	MR9IF	Interrupt flag for match channel 9. 0: No interrupt on match channel 9 1: Interrupt requirements met on match channel 9.	R	0
4	CAP0IF	Interrupt flag for capture channel 0. 0: No interrupt on CAP0 1: Interrupt requirements met on CAP0.	R	0
3	MR3IF	Interrupt flag for match channel 3. 0: No interrupt on match channel 3 1: Interrupt requirements met on match channel 3.	R	0
2	MR2IF	Interrupt flag for match channel 2. 0: No interrupt on match channel 2 1: Interrupt requirements met on match channel 2.	R	0
1	MR1IF	Interrupt flag for match channel 1. 0: No interrupt on match channel 1 1: Interrupt requirements met on match channel 1.	R	0
0	MR0IF	Interrupt flag for match channel 0. 0: No interrupt on match channel 0 1: Interrupt requirements met on match channel 0.	R	0

10.8.15 CT16Bn Timer Interrupt Clear register (CT16Bn_IC) (n=0,1,2,5)

Address Offset: 0x3C

Bit	Name	Description	Attribute	Reset
31:6	Reserved		R	0
5	MR9IC	0: No effect 1: Clear MR9IF bit	W	0
4	CAP0IC	0: No effect 1: Clear CAP0IF bit	W	0
3	MR3IC	0: No effect 1: Clear MR3IF bit	W	0
2	MR2IC	0: No effect 1: Clear MR2IF bit	W	0
1	MR1IC	0: No effect 1: Clear MR1IF bit	W	0
0	MR0IC	0: No effect 1: Clear MR0IF bit	W	0

10.8.16 CT16Bn Timer Match register 9 (CT16Bn_MR9) (n=0,1,2,5)

Address Offset: 0x40

The Match register value is continuously compared to the Timer Counter (TC) value. When the two values are equal,

actions can be triggered automatically. The action possibilities are to reset the Timer Counter, or stop the timer. Actions are controlled by the settings in the CT16Bn_MCTRL[31:30].

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	MR[15:0]	Timer counter match value	R/W	0

10.8.17 CT16Bn PWMmN Dead-band Period register (CT16Bn_PWMmNDB) (n=2)

Address Offset: 0x44 (m=0), 0x48 (m=1), 0x4C (m=2), 0x50 (m=3)

The PWMmNDB register is used to configure the dead-band period of the PWMmN outputs, and is only usable when PWMmNIOEN[1:0] ≠ 00b

Bit	Name	Description	Attribute	Reset
31:10	Reserved		R	0
9:0	DB[9:0]	Count of PWMmN output dead-band period time PWMmN output dead-band period time = DB * CT16Bn_PCLK * (PR+1) cycle	R/W	0

11 WATCHDOG TIMER (WDT)

11.1 OVERVIEW

The purpose of the Watchdog is to reset the MCU within a reasonable amount of time if it enters an erroneous state. When enabled, the Watchdog will generate a system reset or interrupt if the user program fails to "feed" (or reload) the Watchdog within a predetermined amount of time.

The Watchdog consists of a divide by 128 fixed pre-scaler and a 8-bit counter. The clock is fed to the timer via a pre-scaler. The timer decrements when clocked. The minimum value from which the counter decrements is 0x01. Hence the minimum Watchdog interval is $(T_{WDT_PCLK} \times 128 \times 1)$ and the maximum Watchdog interval is $(T_{WDT_PCLK} \times 128 \times 256)$.

The Watchdog should be used in the following manner:

1. Set the prescale value for the watchdog clock with WDTPRE bits in [APB Clock Prescale register 1 \(SYS1_APBPCP1\)](#) register.
2. Set the Watchdog timer constant reload value in [WDT_TC](#) register.
3. Enable the Watchdog and setup the Watchdog timer operating mode in [WDT_CFG](#) register.
4. The Watchdog should be fed again by writing 0x55AA to [WDT_FEED](#) register before the Watchdog counter underflows to prevent reset or interrupt.

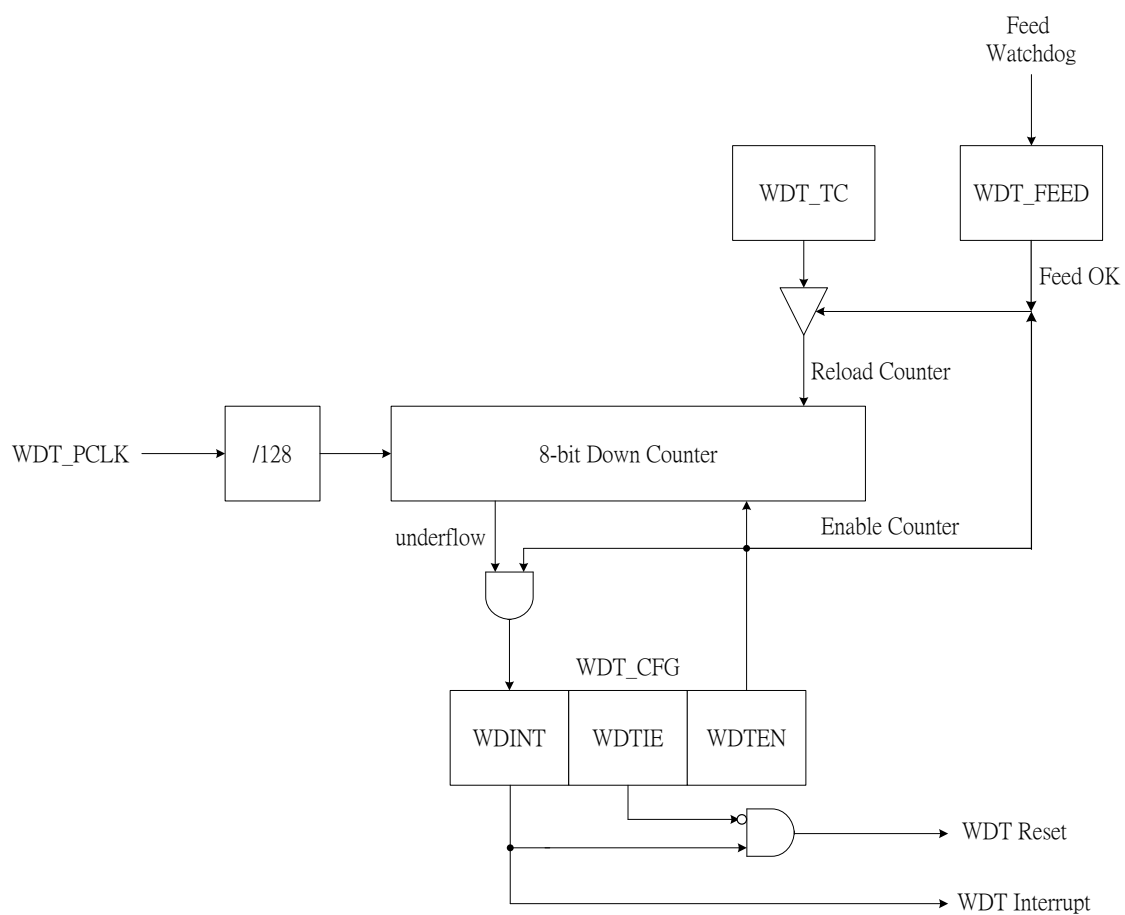
When the watchdog is started by setting the WDTEN in [WDT_CFG](#) register, the time constant value is loaded in the watchdog counter and the counter starts counting down. When the Watchdog is in the reset mode and the counter underflows, the CPU will be reset, loading the stack pointer and program counter from the vector table as in the case of external reset. Whenever the value 0x55AA is written in [WDT_FEED](#) register, the WDT_TC value is reloaded in the watchdog counter and the watchdog reset or interrupt is prevented.

The watchdog timer block uses two clocks: HCLK and WDT_PCLK. HCLK is used for the AHB accesses to the watchdog registers and is derived from the system clock. The WDT_PCLK is used for the watchdog timer counting.

The clock to the watchdog register block can be disabled in [AHB Clock Enable register \(SYS1_AHBCLKEN\)](#) register for power savings.

Watchdog reset or interrupt will occur any time the watchdog is running and has an operating clock source.

11.2 BLOCK DIAGRAM



11.3 WDT REGISTERS

Base Address: 0x4001 0000

11.3.1 Watchdog Configuration register (WDT_CFG)

Address Offset: 0x00

The WDT_CFG register controls the operation of the Watchdog through the combination of WDTEN and WDTIE bits. This register indicates the raw status for Watchdog Timer interrupts. A WDT interrupt is sent to the interrupt controller if both the WDINT bit and the WDTIE bit are set.

Bit	Name	Description	Attribute	Reset
31:16	WDKEY	Watchdog register key. Read as 0. When writing to the register you must write 0x5AFA to WDKEY, otherwise behavior of writing to the register is ignored.	W	0
15:3	Reserved		R	0
2	WDINT	Watchdog interrupt flag 0: Read→Watchdog does not cause an interrupt. Writ→ Clear this flag. SW shall feed Watchdog before clearing. 1: Watchdog timeout and causes an interrupt (Only when WDTIE =1).	R/W	0
1	WDTIE	Watchdog interrupt enable 0: Watchdog timeout will cause a chip reset. (Watchdog reset mode) Watchdog counter underflow will reset the MCU, and will clear the WDINT flag. 1: Watchdog timeout will cause an interrupt. (Watchdog interrupt mode)	R/W	0
0	WDTEN	Watchdog enable 0: Disable 1: Enable. When enable the watchdog, the WDT_TC value is loaded in the watchdog counter.	R/W	0

11.3.2 Watchdog Timer Constant register (WDT_TC)

Address Offset: 0x08

The WDT_TC register determines the time-out value. Every time a feed sequence occurs the WDT_TC content is reloaded in to the Watchdog timer. It's an 8-bit counter. Thus the time-out interval is $T_{WDT_PCLK} \times 128 \times 1 \sim T_{WDT_PCLK} \times 128 \times 256$.

Watchdog overflow time = $(31.25\mu s \times 1) \times 128 \times 1 \sim (31.25\mu s \times 32) \times 128 \times 256$
= 4ms ~ 32768ms

Bit	Name	Description	Attribute	Reset
31:16	WDKEY	Watchdog register key. Read as 0. When writing to the register you must write 0x5AFA to WDKEY, otherwise behavior of writing to the register is ignored.	W	0
15:8	Reserved		R	0
7:0	TC[7:0]	Watchdog timer constant reload value = TC[7:0]+1 0000 0000 : Timer constant = 1 0000 0001 : Timer constant = 2 1111 1110 : Timer constant = 255 1111 1111 : Timer constant = 256	R/W	0xFF

11.3.3 Watchdog Feed register (WDT_FEED)

Address Offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:16	WDKEY	Watchdog register key. Read as 0. When writing to the register you must write 0x5AFA to WDKEY, otherwise behavior of writing to the register is ignored.	W	0
15:0	FV[15:0]	Feed value (Read as 0x0) 0x55AA: The watchdog is fed, and the WDT_TC value is reloaded in the watchdog counter.	W	0

12 REAL-TIME CLOCK (RTC)

12.1 OVERVIEW

The RTC is an independent timer. The RTC provides a set of continuously running counters which can be used to provide a clock-calendar function with suitable software.

The counter values can be written to set the current time/date of the system.

12.2 FEATURES

- Programmable prescale value: division factor up to 2^{20}
- 32-bit programmable counter for long-term measurement
- The RTC clock source could be any of the following:
 - ELS X'TAL
 - ILRC
- Reset sources of the RTC Core (Prescale value, Counter and Divider):
 - “Cold” boot
- One dedicated enabled interrupt
 - Seconds interrupt: generating a periodic interrupt signal with a programmable period length (up to 1 second).

12.3 FUNCTIONAL DESCRIPTION

12.3.1 INTRODUCTION

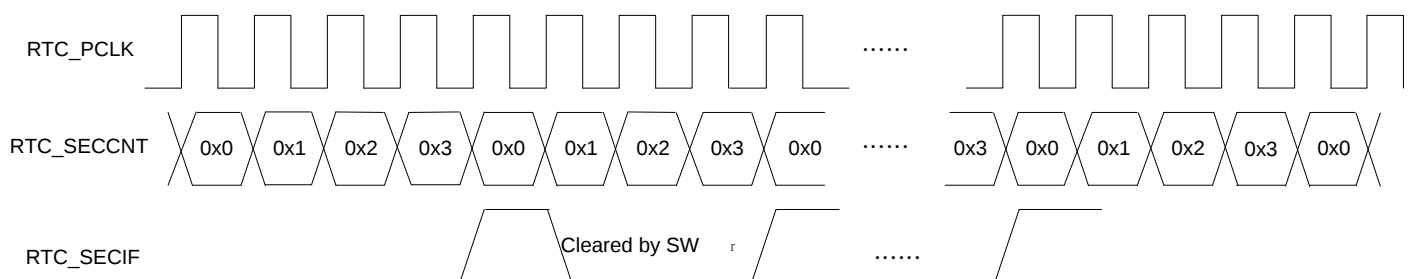
RTC core includes a 20-bit preload value (RTC SECCNTV). Every clock period, the RTC generates an interrupt (Second Interrupt) if it is enabled in [RTC_IE](#) register.

12.3.2 RTC FLAG ASSERTION

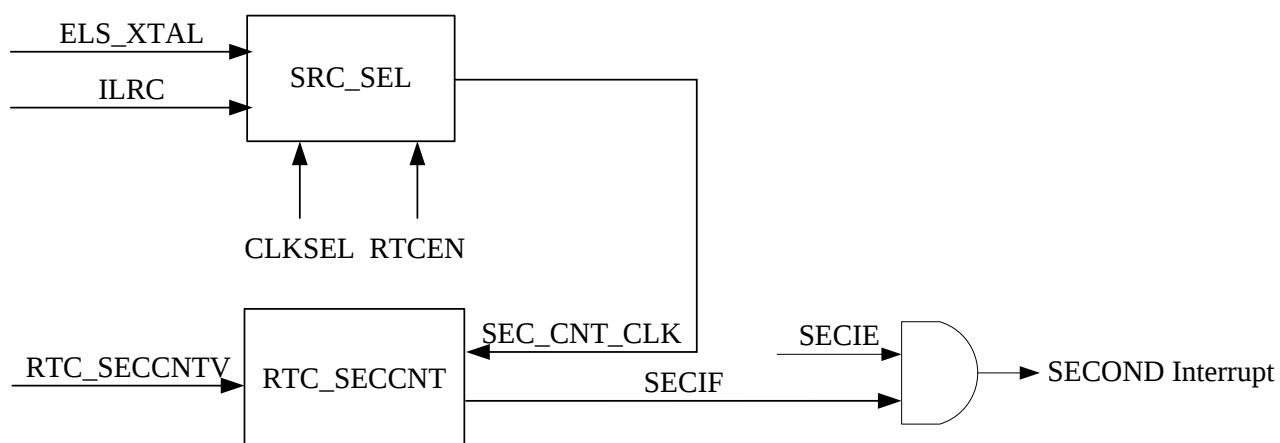
The RTC Second interrupt flag (SECIF) is asserted on each RTC Core clock cycle before the update of the RTC Counter.

12.3.3 RTC OPERATION

The following figure shows the RTC waveform when it is configured with RTC_SECCNTV=3.



12.4 BLOCK DIAGRAM



12.5 RTC REGISTERS

Base Address: 0x4001 2000

12.5.1 RTC Control register (RTC_CTRL)

Address offset: 0x00

* **Note: RTCEN bit shall be set at last!**

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	RTCEN	RTC enable bit 0: Disable 1: Enable. Reset SEC_CNT.	R/W	0

12.5.2 RTC Clock Source Select register (RTC_CLKS)

Address offset: 0x04

* **Note: SW shall disable RTC (RTCEN=0) when changing the value of this register.**

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	CLKSEL	RTC clock source selection. HW will reset SEC_CNT when changing the value. 0: ILRC 1: ELS X'TAL	R/W	0

12.5.3 RTC Interrupt Enable register (RTC_IE)

Address offset: 0x08

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	SECIE	Second interrupt enable 0: Disable 1: Enable	R/W	0

12.5.4 RTC Raw Interrupt Status register (RTC_RIS)

Address offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	SECIF	Second interrupt flag This bit is set by HW when SEC_CNT=SEC_CNTV. An interrupt is generated if SECIE=1. 0: Second flag condition not met. 1: Second flag condition met.	R	0

12.5.5 RTC Interrupt Clear register (RTC_IC)

Address offset: 0x10

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	SECIC	0: No effect 1: Clear SECIF bit	W	0

12.5.6 RTC Second Counter Reload Value register (RTC_SECNTV)

Address offset: 0x14

Reset value: 0x8000

Bit	Name	Description	Attribute	Reset
31:20	Reserved		R	0
19:0	SECNTV[19:0]	RTC second counter reload value. Update this register will reset RTC_SECNT register. The zero value is not recommended, and will be replaced with default value (0x8000) by HW.	R/W	0x8000

12.5.7 RTC Second Count register (RTC_SECNT)

Address offset: 0x18

The RTC core has one 32-bit programmable counter, and this register keeps the current counting value of this counter.

Bit	Name	Description	Attribute	Reset
31:0	SECNT[31:0]	RTC second counter The current value of the RTC counter.	R	0

13_{SPI}

13.1 OVERVIEW

The SPI is a Synchronous Serial Port controller capable of operation on a SPI, and 4-wire SSI bus. It can interact with multiple masters and slaves on the bus. Only a single master and a single slave can communicate on the bus during a given data transfer. Data transfers are in principle full duplex, with frames of 4 to 16 bits of data flowing from the master to the slave and from the slave to the master. In practice it is often the case that only one of these data flows carries meaningful data.

13.2 FEATURES

- Compatible with Motorola SPI bus.
- Synchronous Serial Communication.
- Supports master or slave operation.
- 8-frame FIFO for both transmitter and receiver.
- 4-bit to 16-bit frame.
- Maximum SPI speed of 30 Mbps (master) or 16 Mbps (slave)
- Data transfer format is from MSB or LSB controlled by register.
- The start phase of data sampling location selection is 1st-phase or 2nd-phase controlled register.

13.3 PIN DESCRIPTION

Pin Name	Type	Description	GPIO Configuration
SCKn	O	SPI Serial clock (Master)	
	I	SPI Serial clock (Slave)	Depends on GPIO _n _CFG
SELn	O	SPI Slave Select/SSI Frame Sync (Master)	
	I	SPI Slave Select (Slave)	Depends on GPIO _n _CFG
MISO _n	I	Master In Slave Out (Master)	Depends on GPIO _n _CFG
	O	Master In Slave Out (Slave)	
MOSI _n	O	Master Out Slave In (Master)	
	I	Master Out Slave In (Slave)	Depends on GPIO _n _CFG

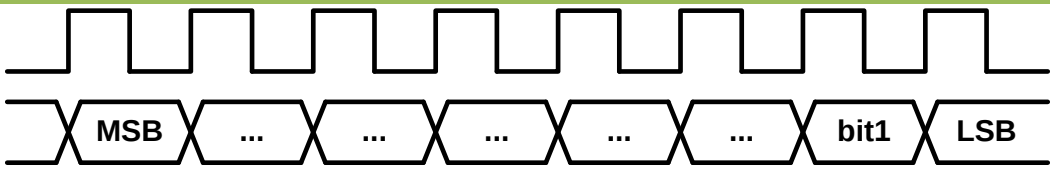
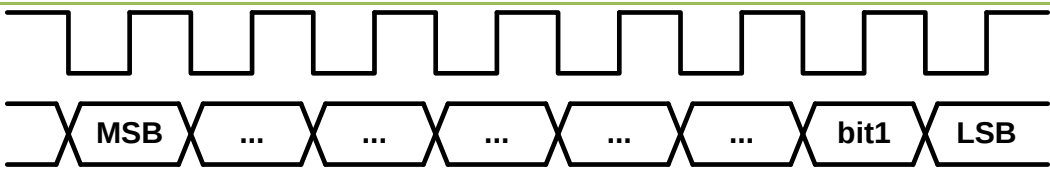
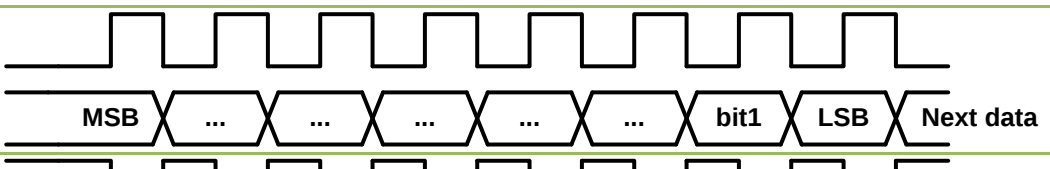
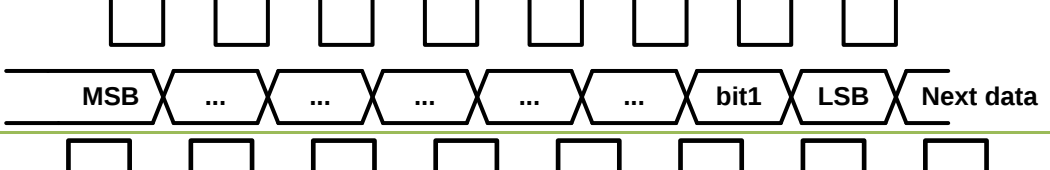
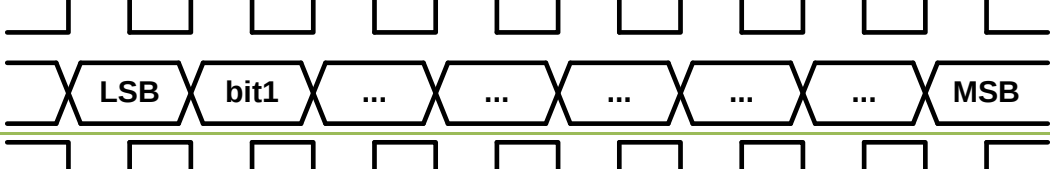
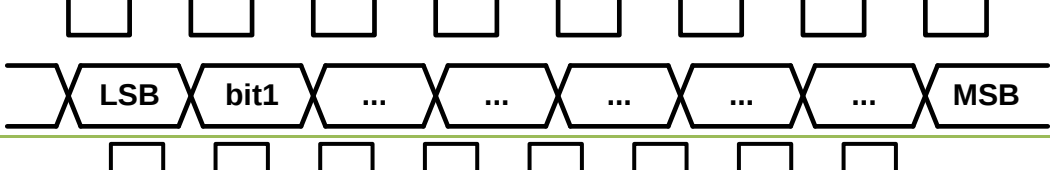
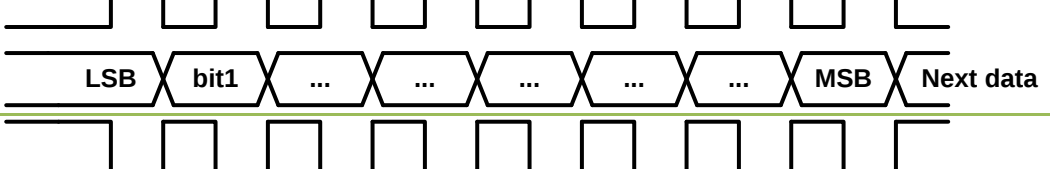
13.4 INTERFACE DESCRIPTION

13.4.1 SPI

The SPI interface is a 4-wire interface where the SEL signal behaves as a slave select. The main feature of the SPI format is that the inactive state and phase of the SCK signal are programmable through the CPOL and CPHA bits in [SPIn_CTRL1](#) register.

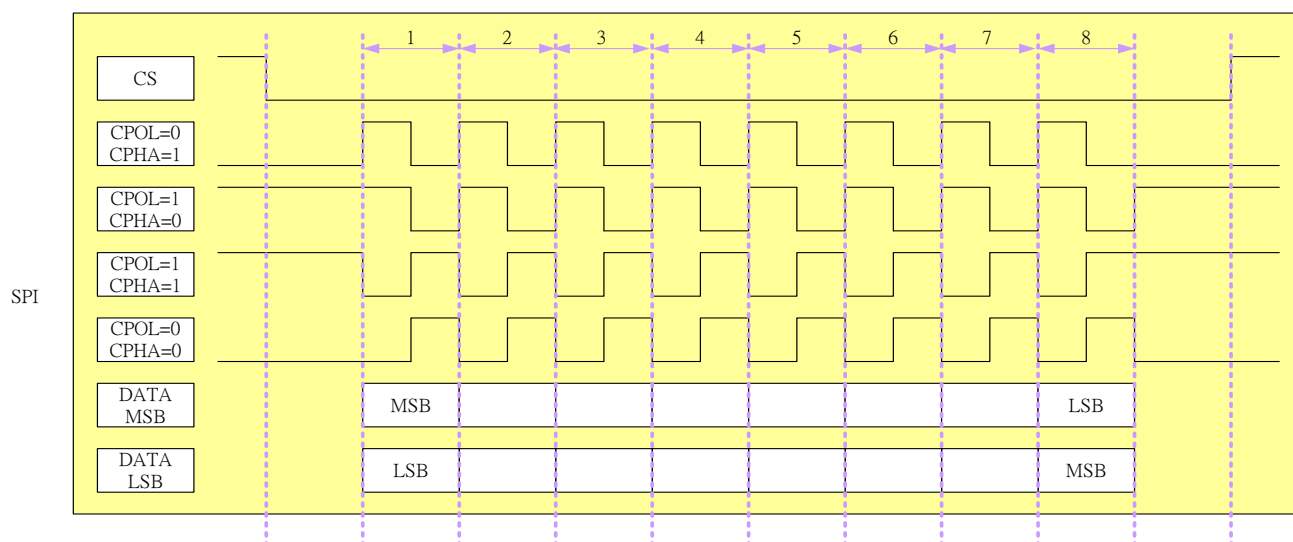
When the “CPOL” clock polarity control bit is LOW, it produces a steady state low value on the SCK pin. If the CPOL clock polarity control bit is HIGH, a steady state high value is placed on the CLK pin when data is not being transferred. The “CPHA” clock phase bit controls the phase of the clock on which data is sampled. When CPHA=1, the SCK first edge is for data transition, and receive and transmit data is at SCK 2nd edge. When CPHA=0, the 1st bit is fixed already, and the SCK first edge is to receive and transmit data.

The SPI data transfer timing as following figure:

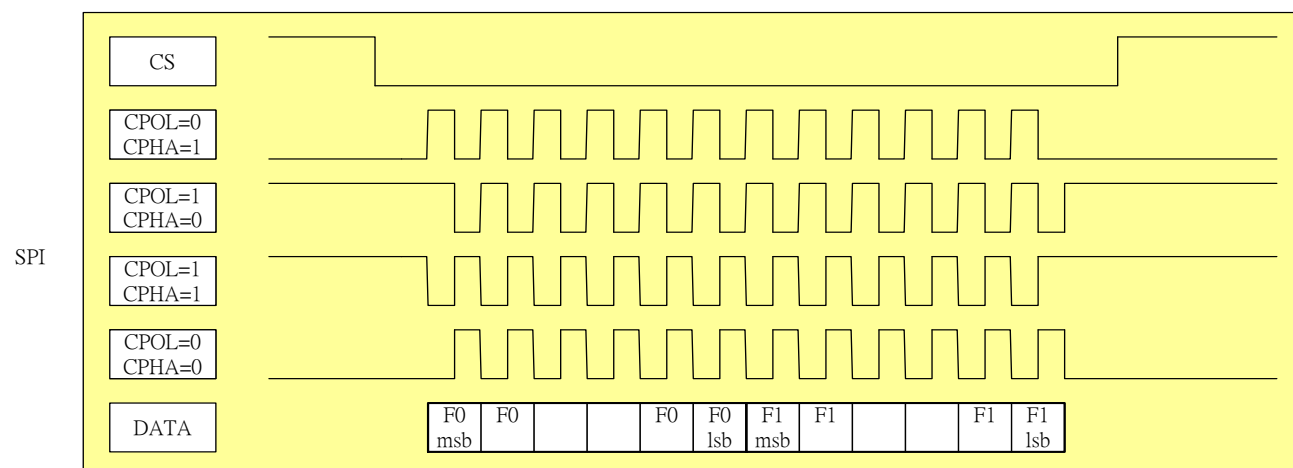
MSLB	CPOL	CPHA	SCK Idle Status	Diagrams
0	0	1	Low	
0	1	1	High	
0	0	0	Low	
0	1	0	High	
1	0	1	Low	
1	1	1	High	
1	0	0	Low	
1	1	0	High	

13.4.2 COMMUNICATION FLOW

13.4.2.1 SINGLE-FRAME



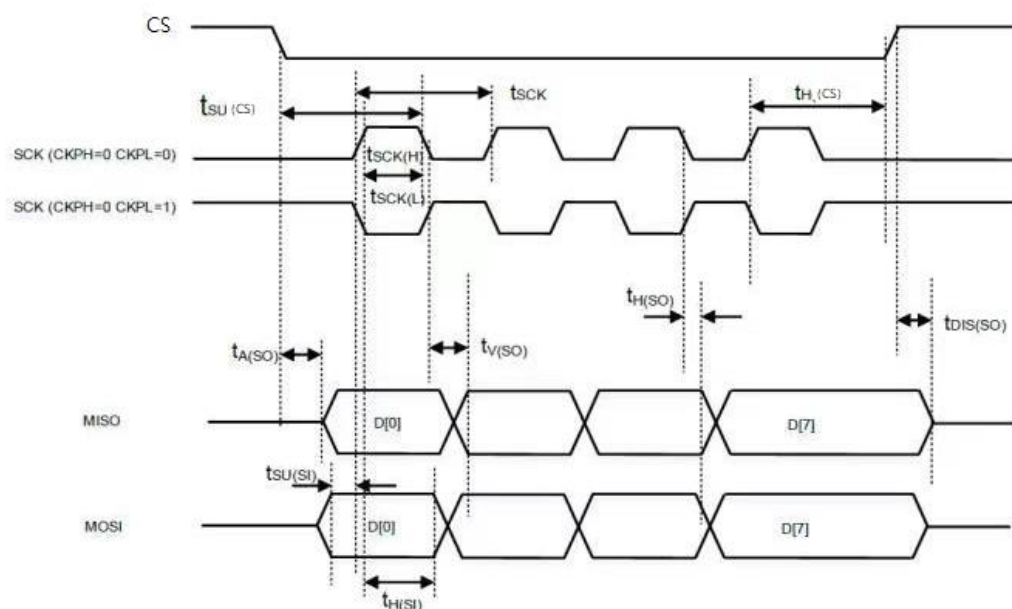
13.4.2.2 MULTI-FRAME



13.5 AUTO-SEL

The Auto-SEL function is disabled (SELDIS = 1) by default, HW does NOT control SELn pin at all, and SELn pin is GPIO. If Auto-SEL function is enabled (SELDIS = 0), SPI HW controls the SELn activity, and SELn is assigned by [PFPA SPI](#) register.

13.6 TIMING CHARACTERISTICS



13.6.1 MASTER MODE

Symbol	Parameter	Min	Typ	Max	Unit
t_{sck}	SCK out period	33			ns
$t_{sck}(H)$	SCK out high pulse	$t_{sck} * 0.4$			ns
$t_{sck}(L)$	SCK out low pulse	$t_{sck} * 0.4$			ns
$t_{su}(CS)$	CS out setup time	$t_{sck} * 2$			ns
$t_h(CS)$	CS out hold time	$t_{sck} * 1.5$			ns
$t_{su}(MISO)$	Data in setup time	$t_{sck} * 0.4$			ns
$t_h(MISO)$	Data in hold time	$t_{sck} * 0.4$			ns
$t_{su}(MOSI)$	Data out setup time	$t_{sck} * 0.4$			ns
$t_h(MOSI)$	Data out hold time	$t_{sck} * 0.4$			ns

13.6.2 SLAVE MODE

Symbol	Parameter	Min	Typ	Max	Unit
t_{sck}	SCK in period	33			ns
$t_{sck}(H)$	SCK in high pulse	$t_{sck} * 0.4$			ns
$t_{sck}(L)$	SCK in low pulse	$t_{sck} * 0.4$			ns
$t_{su}(CS)$	CS in setup time	$t_{sck} * 1.5$			ns
$t_h(CS)$	CS in hold time	t_{sck}			ns
$t_{su}(MISO)$	Data out setup time	$t_{sck} * 0.4$			ns
$t_h(MISO)$	Data out hold time	$t_{sck} * 0.4$			ns
$t_{su}(MOSI)$	Data in setup time	$t_{sck} * 0.4$			ns
$t_h(MOSI)$	Data in hold time	$t_{sck} * 0.4$			ns

13.7 SPI REGISTERS

Base Address: 0x4001 C000 (SPI0)

13.7.1 SPI n Control register 0 (SPIn_CTRL0) (n=0)

Address Offset: 0x00

*** Note:**

1. Must reset SPI FSM with FRESET[1:0] after changing any configuration of SPI when SPIEN = 1.
2. HW will switch I/O configurations refer to FORMAT bit directly when SPIEN = 1.

Bit	Name	Description	Attribute	Reset
31:19	Reserved		R	0
18	SELDIS	Auto-SEL disable bit. 0: Enable Auto-SEL flow control. 1: Disable Auto-SEL flow control.	R/W	1
17:15	RXFIFOTH[2:0]	RX FIFO Threshold level 000: RX FIFO threshold level = 0 001: RX FIFO threshold level = 1 111: RX FIFO threshold level = 7	R/W	000b
14:12	TXFIFOTH[2:0]	TX FIFO Threshold level 000: TX FIFO threshold level = 0 001: TX FIFO threshold level = 1 111: TX FIFO threshold level = 7	R/W	000b
11:8	DL[3:0]	Data length = DL[3:0] + 1 0000~0001: Reversed 0010: data length = 3 1110: data length = 15 1111: data length = 16	R/W	1111b
7:6	FRESET[1:0]	SPI FSM and FIFO Reset bit 00: No effect 01: Reserved 10: Reserved 11: Reset finite state machine and FIFO. (BUF_BUSY = 0, data in shift BUF is cleared, TX_EMPTY = 1, TX_FULL = 0, RX_EMPTY = 1, RX_FULL = 0, and data in FIFO is cleared). This bit will be cleared by HW automatically.	W	0
5	Reserved		R	0
4	FORMAT	Interface format. 0: SPI 1: Reserved	R/W	0
3	MS	Master/Slave selection bit 0: Act as Master. 1: Act as Slave.	R/W	0
2	SDODIS	Slave data output disable bit (ONLY used in slave mode) 0: Enable slave data output. 1: Disable slave data output. (MISO=0)	R/W	0
1	LOOPBACK	Loop back mode enable 0: Disable 1: Data input from data output	R/W	0
0	SPIEN	SPI enable bit 0: Disable 1: Enable.	R/W	0

13.7.2 SPI n Control register 1 (SPIn_CTRL1) (n=0)

Address Offset: 0x04

Bit	Name	Description	Attribute	Reset
31:3	Reserved		R	0
2	CPHA	Clock phase for edge sampling. 0: Data changes at clock falling edge, latches at clock rising edge when CPOL = 0; Data changes at clock rising edge, latches at clock falling edge when CPOL = 1. 1: Data changes at clock rising edge, latches at clock falling edge when CPOL = 0; Data changes at clock falling edge, latches at clock rising edge when CPOL = 1.	R/W	0
1	CPOL	Clock polarity selection bit 0: SCK idles at Low level. 1: SCK idles at High level.	R/W	0
0	MLSB	MSB/LSB selection bit 0: MSB transmit first. 1: LSB transmit first.	R/W	0

13.7.3 SPI n Clock Divider register (SPIn_CLKDIV) (n=0)

Address Offset: 0x08

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	DIV[7:0]	SPIn clock divider 0: SCK = SPIn_PCLK / 2 1: SCK = SPIn_PCLK / 4 2: SCK = SPIn_PCLK / 6 X: SCK = SPIn_PCLK / (2X+2)	R/W	0

13.7.4 SPI n Status register (SPIn_STAT) (n=0)

Address Offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	RXFIFOTHF	RX FIFO threshold flag 0: Data in RX FIFO ≤ RXFIFOTH 1: Data in RX FIFO > RXFIFOTH	R	0
5	TXFIFOTHF	TX FIFO threshold flag 0: Data in TX FIFO > TXFIFOTH 1: Data in TX FIFO ≤ TXFIFOTH	R	1
4	BUSY	Busy flag. 0: SPI controller is idle. 1: SPI controller is transferring.	R	0
3	RX_FULL	RX FIFO full flag. 0: RX FIFO is NOT full. 1: RX FIFO is full.	R	0
2	RX_EMPTY	RX FIFO empty flag 0: RX FIFO is NOT empty. 1: RX FIFO is empty.	R	1
1	TX_FULL	TX FIFO full flag. 0: TX FIFO is NOT full. 1: TX FIFO is full.	R	0
0	TX_EMPTY	TX FIFO empty flag 0: TX FIFO is NOT empty. In Master mode, the transmitter will begin to transmit automatically.	R	1

1: TX FIFO is empty.

13.7.5 SPI n Interrupt Enable register (SPIn_IE) (n=0)

Address Offset: 0x10

This register controls whether each of the four possible interrupt conditions in the SPI controller is enabled.

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	TXFIFOTHIE	TX FIFO threshold interrupt enable 0: Disable 1: Enable	R/W	0
2	RXFIFOTHIE	RX FIFO threshold interrupt enable 0: Disable 1: Enable	R/W	0
1	RXTOIE	RX time-out interrupt enable 0: Disable 1: Enable	R/W	0
0	RXOVFIE	RX Overflow interrupt enable 0: Disable 1: Enable	R/W	0

13.7.6 SPI n Raw Interrupt Status register (SPIn_RIS) (n=0)

Address Offset: 0x14

This register contains the status for each interrupt condition, regardless of whether or not the interrupt is enabled in SPIn_IE register.

This register indicates the status for SPI control raw interrupts. An SPI interrupt is sent to the interrupt controller if the corresponding bit in the SPIn_IE register is set.

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	TXFIFOTHIF	TX FIFO threshold interrupt flag 0: No TX FIFO threshold interrupt 1: TX FIFO threshold triggered.	R	0
2	RXFIFOTHIF	RX FIFO threshold interrupt flag 0: No RX FIFO threshold interrupt 1: RX FIFO threshold triggered.	R	0
1	RXTOIF	RX time-out interrupt flag RXTO occurs when the RX FIFO is not empty, and has not been read for a time-out period (32*SPIn_PCLK). The time-out period is the same for master and slave modes. 0: RXTO doesn't occur. 1: RXTO occurs.	R	0
0	RXOVFIF	RX Overflow interrupt flag RXOVF occurs when the RX FIFO is full and another frame is completely received. The ARM spec implies that the preceding frame data is overwritten by the new frame data when this occurs. 0: RXOVF doesn't occur. 1: RXOVF occurs.	R	0

13.7.7 SPI n Interrupt Clear register (SPIn_IC) (n=0)

Address Offset: 0x18

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	TXFIFOTHIC	0: No effect 1: Clear TXFIFOTHIF bit	W	0
2	RXFIFOTHIC	0: No effect 1: Clear RXFIFOTHIF bit	W	0
1	RXTOIC	0: No effect 1: Clear RXTOIF bit.	W	0
0	RXOVFIC	0: No effect 1: Clear RXOVFIF bit.	W	0

13.7.8 SPI n Data register (SPIn_DATA) (n=0)

Address Offset: 0x1C

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	DATA[15:0]	<u>Write</u> SW can write data to be sent in a future frame to this register when TX_FULL = 0 in SPIn_STAT register (TX FIFO is not full). If the TX FIFO was previously empty and the SPI controller is not busy on the bus, transmission of the data will begin immediately. Otherwise the data written to this register will be sent as soon as all previous data has been sent (and received). <u>Read</u> SW can read data from this register when RX_EMPTY=0 in SPIn_STAT register (Rx FIFO is not empty). When SW reads this register, the SPI controller returns data from the least recent frame in the RX FIFO. If the data length is less than 16 bit, the data is right-justified in this field with higher order bits filled with 0s.	R/W	0

13.7.9 SPI n Data Fetch Delay register (SPIn_DFDLY) (n=0)

Address Offset: 0x20

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	DFETCH_EN	SPI data fetch control bit 0: Disable 1: Enable when SCKn frequency > 6MHz	R/W	0

14_{I2C}

14.1 OVERVIEW

The I2C bus is bidirectional for inter-IC control using only two wires: Serial Clock Line (SCL) and Serial Data line (SDA). Each device is recognized by a unique address and can operate as either a receiver-only device (e.g., an LCD driver) or a transmitter with the capability to both receive and send information (such as memory). Transmitters and/or receivers can operate in either master or slave mode, depending on whether the chip has to initiate a data transfer or is only addressed. The I2C is a multi-master bus and can be controlled by more than one bus master connected to it. It is also SMBus 2.0 compatible.

The I2C interface is byte oriented and has four operating modes:

- Master transmitter mode
- Master receiver mode
- Slave transmitter mode
- Slave receiver mode

14.2 FEATURES

The I2C interface complies with the entire I2C specification, supporting the ability to turn power off to the ARM Cortex-M0 without interfering with other devices on the same I2C-bus.

- Standard I2C-compliant bus interfaces may be configured as Master or Slave.
- I2C Master features:
 - Clock generation
 - Start and Stop generation
- I2C Slave features:
 - Programmable I2C Address detection
 - Optional recognition of up to four distinct slave addresses
 - Stop bit detection
- Supports different communication speeds:
 - Standard Speed (up to 100KHz)
 - Fast Speed (up to 400 KHz)
- Arbitration is handled between simultaneously transmitting masters without corruption of serial data on the bus.
- Programmable clock allows adjustment of I2C transfer rates.
- Data transfer is bidirectional between masters and slaves.
- Serial clock synchronization allows devices with different bit rates to communicate via one serial bus.
- Serial clock synchronization is used as a handshake mechanism to suspend and resume serial transfer.
- I2C-bus can be used for test and diagnostic purposes.
- Generation and detection of 7-bit/10-bit addressing and General Call.

14.3 PIN DESCRIPTION

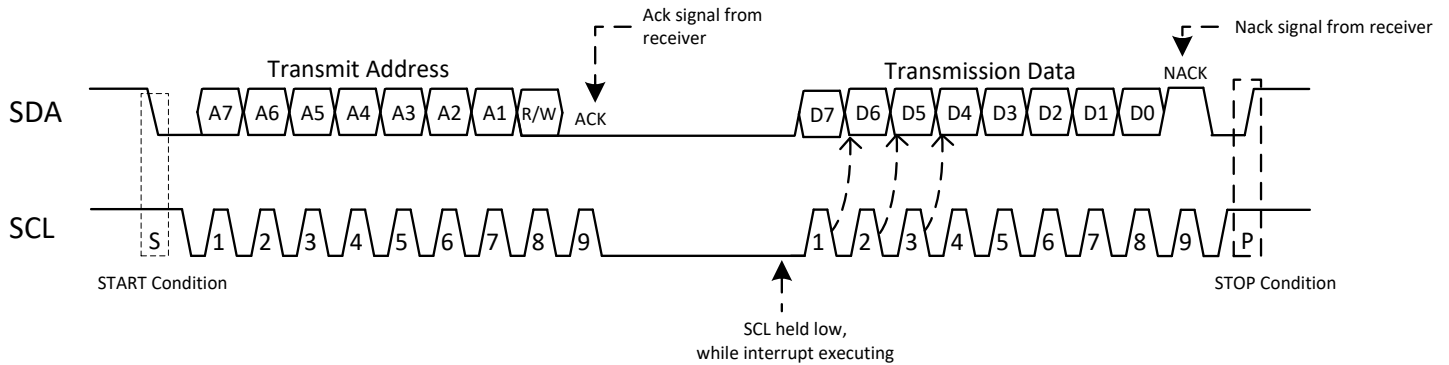
Pin Name	Type	Description	GPIO Configuration
SCLn	I/O	I2C Serial clock	Output with Open-drain Input depends on GPIO _n _CFG
SDAn	I/O	I2C Serial data	Output with Open-drain Input depends on GPIO _n _CFG

14.4 I2C PROTOCOL

I2C transmission structure includes a START(S) condition, 8-bit address byte, one or more data byte and a STOP (P) condition. START condition is generated by master to initial any transmission.

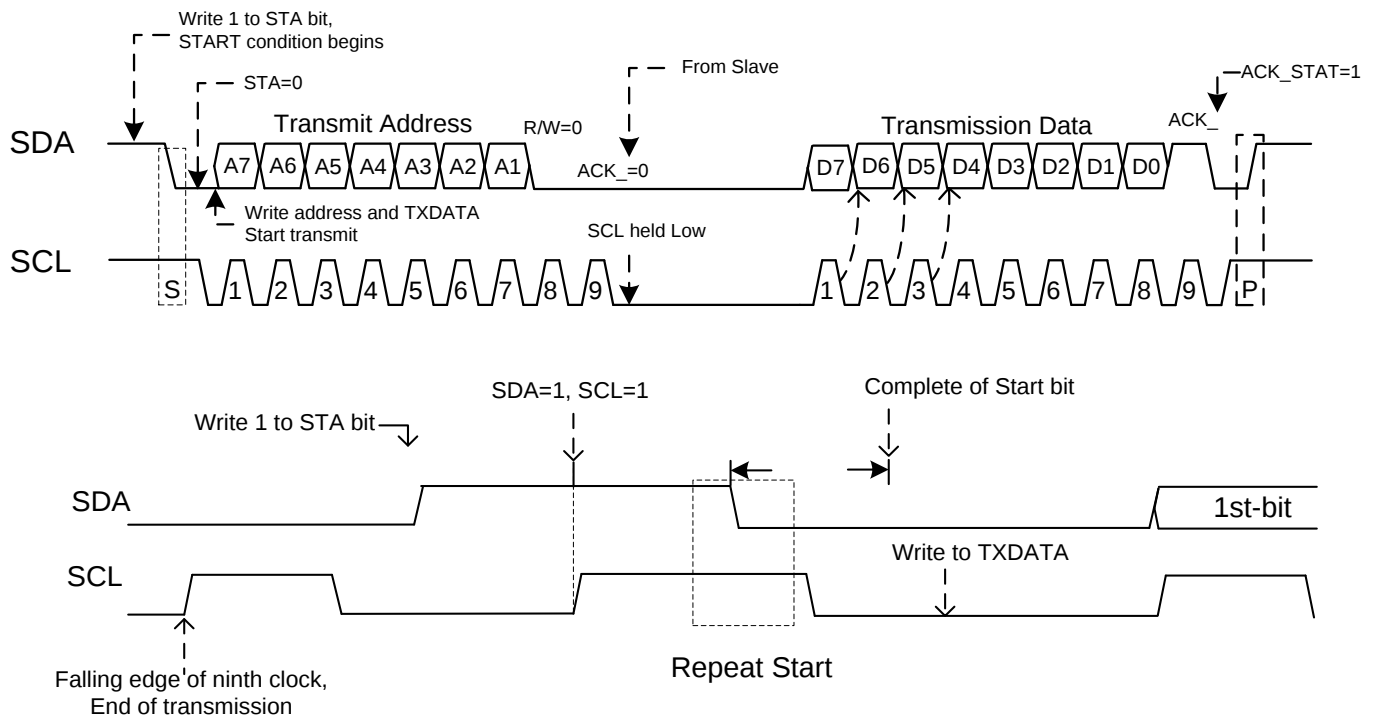
Data is transmitted with the Most Significant Bit (MSB) first. In address byte, the higher 7-bit is address bit and the lowest bit is data direction (R/W) bit. When R/W=0, it assigns a “WRITE” operation. When R/W=1, it assigns a “READ” operation.

After each byte is received, the receiver (a master or a slave) must send an acknowledge (ACK) bit. If transmitter can't receive an ACK, it will recognize a not acknowledge (NACK). In WRITE operation, the master will transmit data to the slave and then waits for ACK from slave. In READ operation, the slave will transmit data to the master and then waits for ACK from master. In the end, the master will generate a STOP condition to finish transmission.

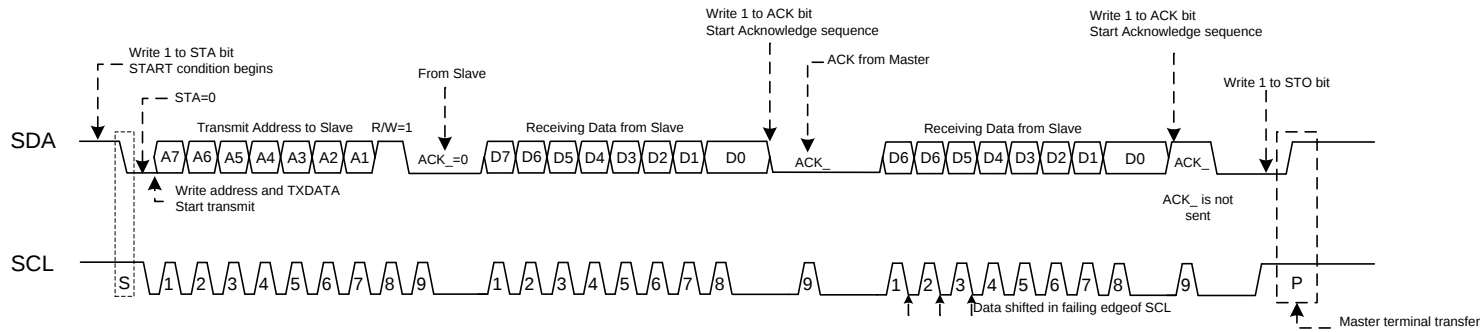


14.4.1 7-BIT ADDRESSING MODES

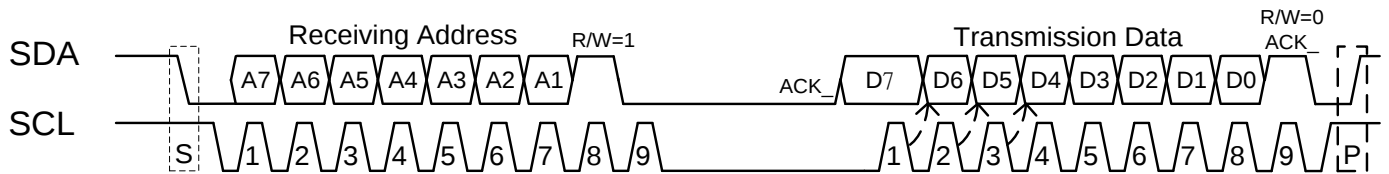
14.4.1.1 MASTER TRANSMITTER MODE



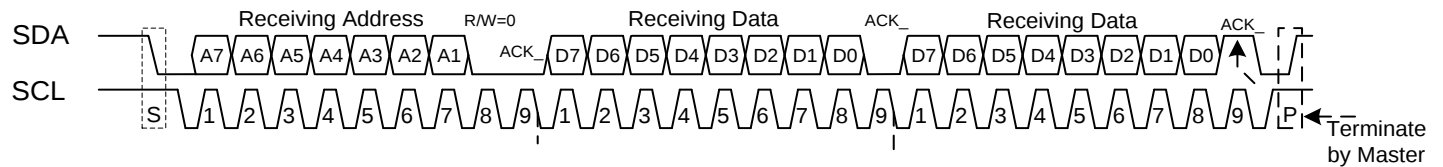
14.4.1.2 MASTER RECEIVER MODE



14.4.1.3 SLAVE TRANSMITTER MODE

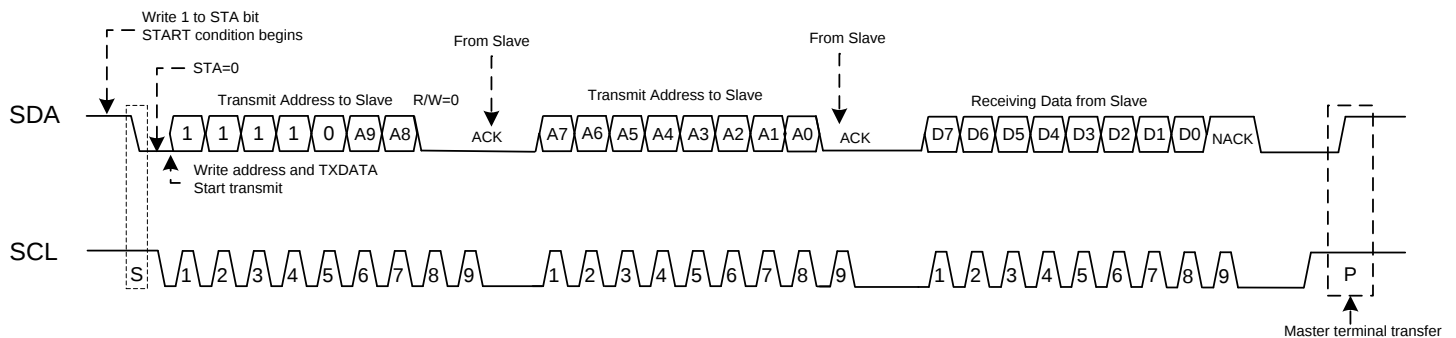


14.4.1.4 SLAVE RECEIVER MODE

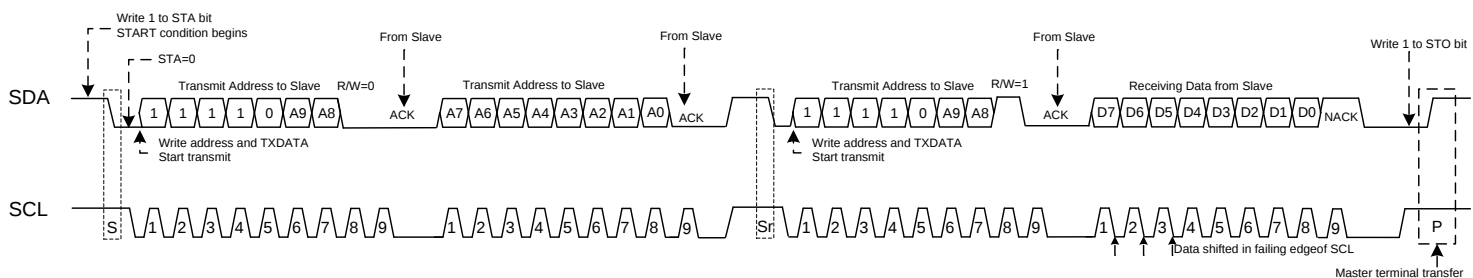


14.4.2 10-BIT ADDRESSING MODES

1.1.1.1 MASTER TRANSMITTER MODE



1.1.1.2 MASTER RECEIVER MODE



14.5 ARBITRATION

In multi-master condition, more than one master may transmit on bus in the same time. It must be decided which master has the control of bus and complete its transmission. Clock synchronization and arbitration are used to configure multi-master transmission.

Clock synchronization is executed by synchronizing the SCL signal with another devices. When two masters want to transmit data in the same time, the clock synchronization will start by the High to Low transition on the SCL. If master 1 pulls the SCL line LOW first, it holds the SCL in LOW status until the SCL line is released to HIGH status. However, if another master still pulls the SCL line LOW, the SCL Low to High transition of master 1 may not change SCL status (SCL line is still LOW). The SCL will transit from LOW to HIGH when the all masters release the SCL line. In the duration, the master1 will wait for SCL transition from LOW to HIGH, and then continue its transmission.

After clock synchronization, the clock of all devices is synchronized with the SCL clock. Arbitration is used to decide which master can complete its transmission by SDA signal. Two masters may send out a START condition and transmit data on bus in the same time, and may be influenced by each other. Arbitration will force one master to lose the control on bus. Data transmission will keep until two masters output different data signal. If one master transmits HIGH status and another master transmits LOW status, the SDA will be pulled low. The master which pulls the SDA line High will detect the different with SDA and loses the control on bus. The master which pulls the SDA line LOW status wins the bus control and continues its transmission. There is no data miss during arbitration.

Arbitration may also be lost in the master receiver mode. Loss of arbitration in this mode can only occur while the I2C block is returning a “not acknowledge” to the bus. Arbitration is lost when another device on the bus pulls this signal low. Since this can occur only at the end of a serial byte, the I2C block generates no further clock pulses.

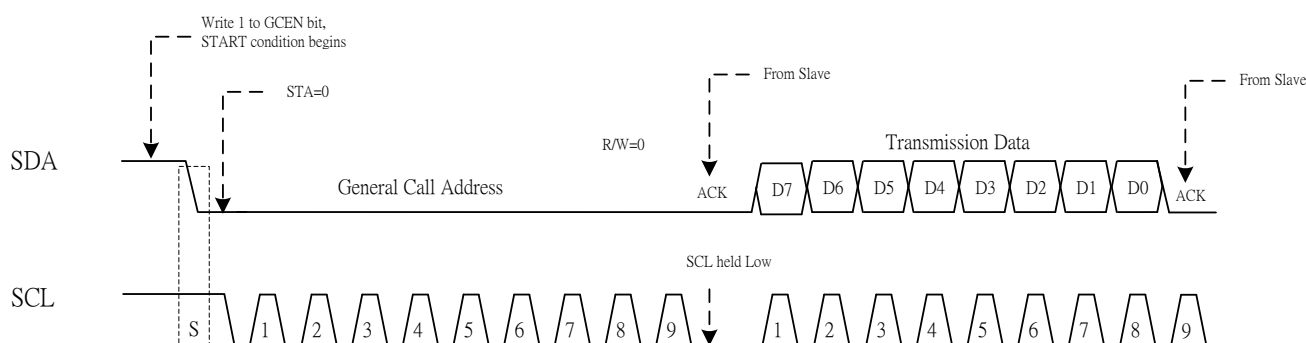
14.6 CLOCK STRETCHING

Clock stretching pauses a transaction by pulling the SCL line LOW. The transaction cannot continue until the line is released HIGH again. Clock stretching is optional.

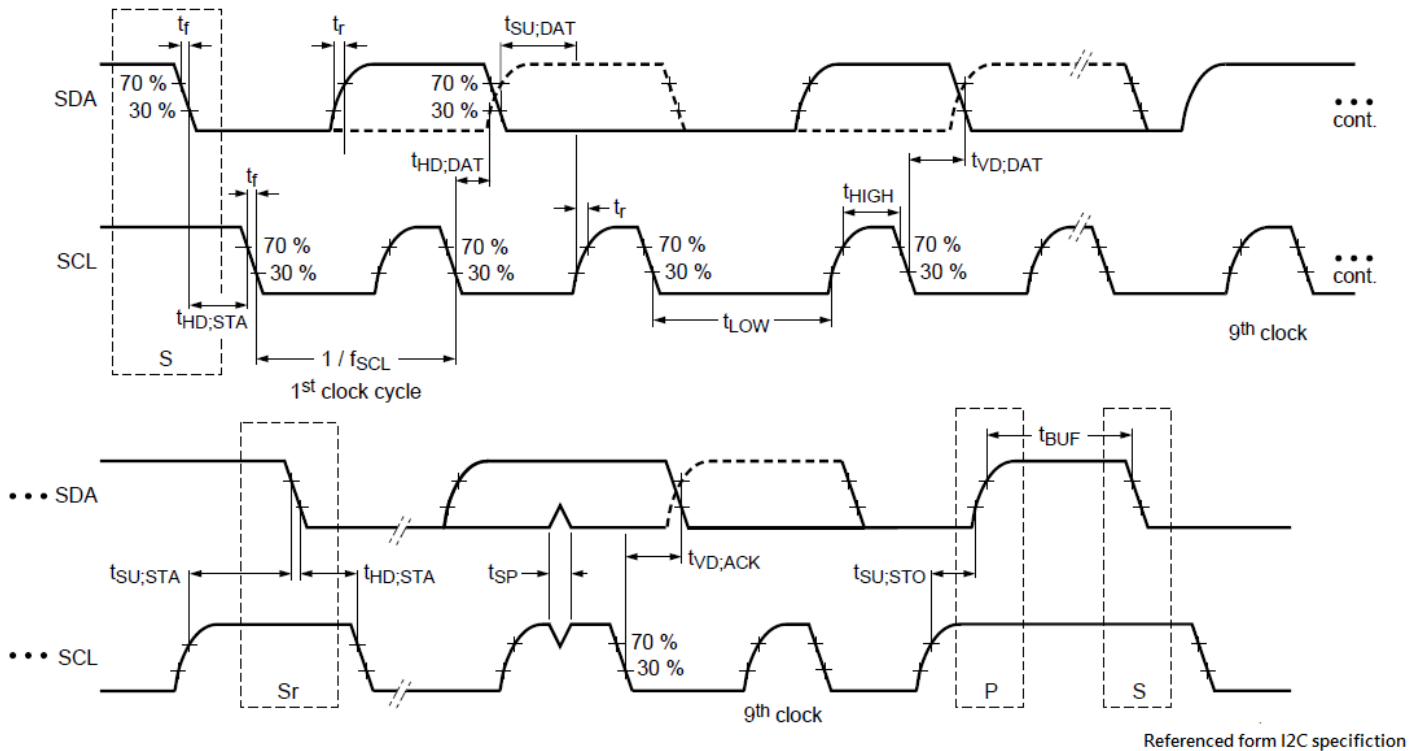
On the byte level, a device may be able to receive bytes of data at a fast rate, but needs more time to store a received byte or prepare another byte to be transmitted. Slaves can then pull the SCL line LOW after reception and acknowledgment of a byte to force the master into a wait state until the slave is ready for the next byte transfer in a type of handshake procedure.

14.7 GENERAL CALL ADDRESS

The general call address is a special address which is reserved as all “0” of 7-bit address and is for addressing every device connected to the I2C-bus at the same time. However, if a device does not need any of the data supplied within the general call structure, it can ignore this address by not issuing an acknowledgment (ACK). If a device does require data from a general call address, it acknowledges this address and behaves as a slave-receiver. The master does not actually know how many devices acknowledged if one or more devices respond. The second and following bytes are acknowledged by every slave-receiver capable of handling this data. A slave who cannot process one of these bytes must ignore it by not-acknowledging. If one or more slaves acknowledge, the not-acknowledge will not be seen by the master. The meaning of the general call address is specified in the second byte.



14.8 TIMING CHARACTERISTICS



14.8.1 MASTER TRANSMITTER MODE

- t_{HIGH} : $(SCLHT + 1) * I2C_PCLK$ cycle
- t_{LOW} : $(SCLLT + 1) * I2C_PCLK$ cycle
- $t_{HD;STA}$: $(SCLLT + 4) * I2C_PCLK$ cycle
- $t_{HD;DAT}$: $2 * I2C_PCLK$ cycle ~ $3 * I2C_PCLK$ cycle
- $t_{SU;STA}$: $(SCLL + 3) * I2C_PCLK$ cycle
- $t_{SU;STO}$: $(SCLLT + 2) * I2C_PCLK$ cycle ~ $(SCLLT + 3) * I2C_PCLK$ cycle

14.8.2 SLAVE TRANSMITTER MODE

- t_{HIGH} : controlled by Master
- t_{LOW} : controlled by Master
- $t_{HD;DAT}$: $2 * I2C_PCLK$ cycle ~ $3 * I2C_PCLK$ cycle
- $t_{SU;DAT}$: $t_{LOW} - 2 * I2C_PCLK$ cycle ~ $t_{LOW} - 3 * I2C_PCLK$ cycle

14.9 I2C REGISTERS

Base Address: 0x4001 8000 (I2C0)
0x4005 A000 (I2C1)

14.9.1 I2C n Control register (I2Cn_CTRL) (n=0,1)

Address Offset: 0x00

Setting of the bits in this register controls operation of the I2C interface.

When STA =1 and the I2C interface is not already in master mode, it enters master mode, checks the bus and generates a START condition if the bus is free. If the bus is not free, it waits for a STOP condition (which will free the bus) and generates a START condition after a delay of a half clock period of the internal clock generator. If the I2C interface is already in master mode and data has been transmitted or received, it transmits a Repeated START condition. STA may be set at any time, including when the I2C interface is in an addressed slave mode.

When STO = 1 in master mode, a STOP condition is transmitted on the I2C bus. When the bus detects the STOP condition, STO is cleared automatically. In slave mode, setting STO bit can recover from an error condition. In this case, no STOP condition is transmitted to the bus. The HW behaves as if a STOP condition has been received and it switches to “not addressed” slave receiver mode.

If STA and STO are both set, then a STOP condition is transmitted on the I2C bus if the interface is in master mode, and transmits a START condition thereafter. If the I2C interface is in slave mode, an internal STOP condition is generated, but is not transmitted on the bus.

*** Note:**

1. ***I2CEN shall be set at last.***
2. ***HW will assign SCL0 and SDA0 pins as output pins with open-drain function instead of GPIO automatically.***
3. ***ACK and NACK bits can't both be “1” when receiving data.***
4. ***User has to write 1 to ACK or NACK bit in Master mode to continue next RX process.***

Bit	Name	Description	Attribute	Reset
31:9	Reserved		R	0
8	I2CEN	I2C Interface enable bit 0: Disable. The STO bit is forced to “0”. 1: Enable. I2CEN shall not be used to temporarily release the I2C bus since the bus status is lost when I2CEN resets. The ACK flag should be used instead.	RW	0
7:6	Reserved		R	0
5	STA	START bit. 0: No START condition or Repeated START condition will be generated. 1: Cause the I2C interface to enter master mode and transmit a START or a Repeated START condition. Automatically cleared by HW.	RW	0
4	STO	STOP flag 0: Stop condition idle. 1: Cause the I2C interface to transmit a STOP condition in master mode, or recover from an error condition in slave mode. Automatically cleared by HW.	RW	0
3	Reserved		R	0
2	ACK	Assert ACK flag. 0: Master mode→ No function Slave mode→Return a NACK after receiving address or data. 1: An ACK will be returned during the acknowledge clock pulse on SCLn when ➤ The address in the Slave Address register has been received. ➤ The General Call address has been received while the General	RW	0

		Call enable bit (GCEN) in SLVADDR0 register is set. ➤ A data byte has been received while the I2C is in the master receiver mode. ➤ A data byte has been received while the I2C is in the addressed slave receiver mode. HW will clear after issuing ACK automatically.		
1	NACK	Assert NACK flag. 0: No function 1: An NACK will be returned during the acknowledge clock pulse on SCLn when ➤ A data byte has been received while the I2C is in the master receiver mode. HW will clear after issuing NACK automatically.	RW	0
0	Reserved		R	0

14.9.2 I2C n Status register (I2Cn_STAT) (n=0,1)

Address Offset: 0x04

Check this register when I2C interrupt occurs, and all status will be cleared automatically by writing I2Cn_CTRL or I2Cn_TXDATA register.

Following events will trigger I2C interrupt if I2C interrupt is enabled in NVIC interrupt controller.

- START/Repeat START condition
- STOP condition
- Timeout
- Data byte transmitted or received
- ACK Transmit or received
- NACK Transmit or received

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15	I2CIF	I2C Interrupt flag. 0: I2C status doesn't change. 1: Read→I2C status changes. Write→Clear this flag.	RW	0
14:10	Reserved		R	0
9	TIMEOUT	Time-out status 0: No Timeout 1: Timeout	R	0
8	LOST_ARB	Lost arbitration 0: Not lost arbitration 1: Lost arbitration	R	0
7	SLV_TX_HIT	0: No matched slave address. 1: Slave address hit, and is called for TX in slave mode.	R	0
6	SLV_RX_HIT	0: No matched slave address. 1: Slave address hit, and is called for RX in slave mode.	R	0
5	MST	Master/Slave status 0: I2C is in Slave state. 1: I2C is in Master state.	R	0
4	START_DN	Start done status 0: No START bit. 1: MASTER mode→ a START bit was issued. SLAVE mode→a START bit was received.	R	0
3	STOP_DN	Stop done status 0: No STOP bit. 1: MASTER mode→a STOP condition was issued. SLAVE mode→a STOP condition was received.	R	0
2	NACK_STAT	NACK done status 0 : Not received a NACK 1 : Received a NACK	R	0

1	ACK_STAT	ACK done status 0 : Not received an ACK 1 : Received an ACK	R	0
0	RX_DN	RX done status 0: No RX with ACK/NACK transfer. 1: 8-bit RX with ACK/NACK transfer is done.	R	0

14.9.3 I2C n TX Data register (I2Cn_TXDATA) (n=0,1)

Address Offset: 0x08

This register contains the data to be transmitted.

In Master TX mode, CPU writes this register will trigger a TX function. In Slave TX mode, CPU has to write this register before next TX procedure.

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	DATA[7:0]	Data to be transmitted.	R/W	0x00

14.9.4 I2C n RX Data register (I2Cn_RXDATA) (n=0,1)

Address Offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	DATA[7:0]	Contains the data received. Read this register when RX_DN = 1.	R	0x00

14.9.5 I2C n Slave Address 0 register (I2Cn_SLVADDR0) (n=0,1)

Address Offset: 0x10

Only used in slave mode. In master mode, this register has no effect.

If this register contains 0x00, the I2C will not acknowledge any address on the bus. Register ADR0 to ADR3 will be cleared to this disabled state on reset.

Bit	Name	Description	Attribute	Reset
31	ADD_MODE	Slave address mode. 0 : 7-bit address mode 1: 10-bit address mode	RW	0
30	GCEN	General call address enable bit. 0: Disable 1: Enable general call address (0x0)	RW	0
29:10	Reserved		R	0
9:0	ADDR[9:0]	The I2C slave address. ADD[9:0] is valid when ADD_MODE = 1 ADD[7:1] is valid when ADD_MODE = 0	R/W	0

14.9.6 I2C n Slave Address 1~3 register (I2Cn_SLVADDR1~3) (n=0,1)

Address Offset: 0x14, 0x18, 0x1C

Bit	Name	Description	Attribute	Reset
31:10	Reserved		R	0
9:0	ADDR[9:0]	The I2C slave address. ADD[9:0] is valid when ADD_MODE = 1 ADD[7:1] is valid when ADD_MODE = 0	R/W	0

14.9.7 I2C n SCL High Time register (I2Cn_SCLHT) (n=0,1)

Address Offset: 0x20

* Note: I2C Bit Frequency = $I2Cn_PCLK / (I2Cn_SCLHT + I2Cn_SCLLT)$

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	SCLH[7:0]	Count for SCL High Period time , shall more than 5 SCL High Period Time = (SCLH+1) * I2C0_PCLK cycle	R/W	0x04

14.9.8 I2C n SCL Low Time register (I2Cn_SCLLT) (n=0,1)

Address Offset: 0x24

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	SCLL[7:0]	Count for SCL Low Period time, shall more than 5 SCL Low Period Time = (SCLL+1) * I2C0_PCLK cycle	R/W	0x04

14.9.9 I2C n Timeout Control register (I2Cn_TOCTRL) (n=0,1)

Address Offset: 0x2C

Timeout happens when Master/Slave SCL remained LOW for:

$TO * 32 * I2C0_PCLK$ cycle.

When I2C timeout occurs, the I2C transfer will return to "IDLE" state, and issue a TO interrupt to inform user. That means SCL/SDA will be released by HW after timeout. User can issue a STOP after timeout interrupt occurred in Master mode.

Time-out status will be cleared automatically by writing I2Cn_CTRL or I2Cn_TXDATA register.

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	TO[15:0]	Count for checking Timeout. 0: Disable Timeout checking N: Timeout period time = $N * 32 * I2Cn_PCLK$ cycle	R/W	0x0

15 UNIVERSAL ASYNCHRONOUS RECEIVER AND TRANSMITTER (UART)

15.1 OVERVIEW

The UART offers a flexible means of full-duplex data exchange with external equipment requiring an industry standard NRZ asynchronous serial data format. The serial interface is applied to low speed data transfer and communicate with low speed peripheral devices.

The UART offers a very wide range of baud rates using a fractional baud rate generator.

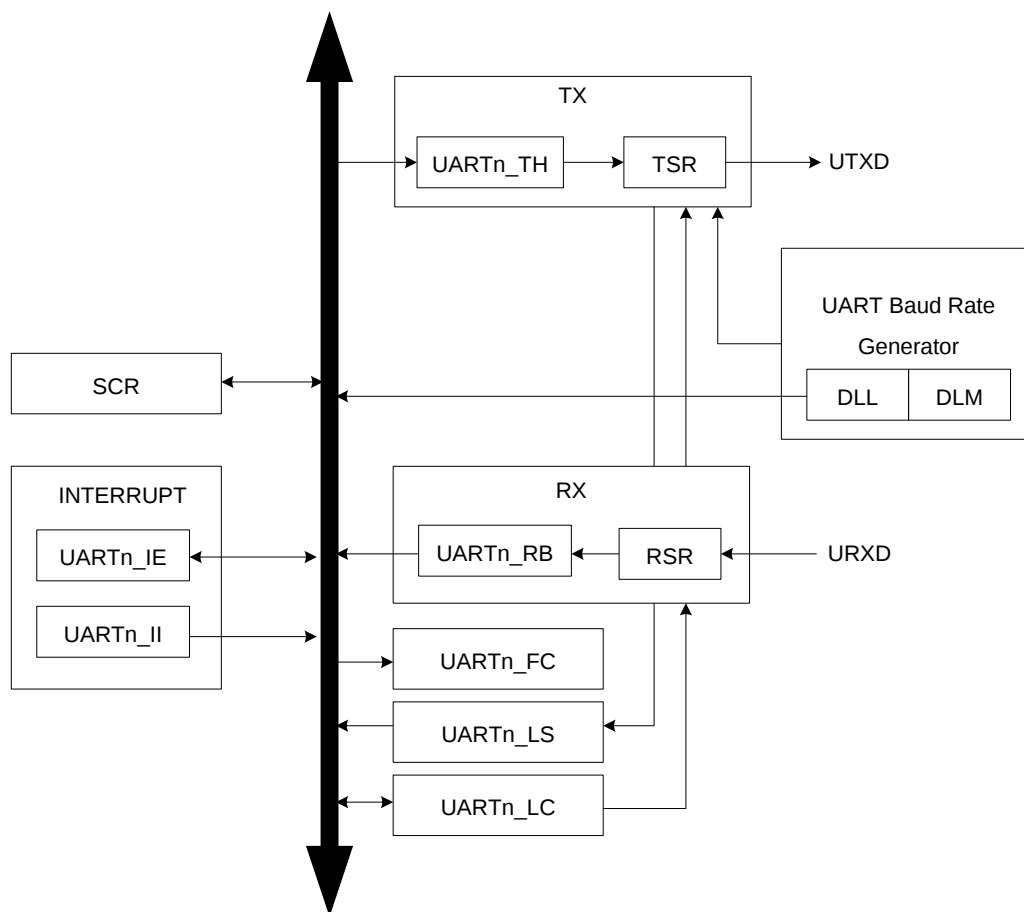
15.2 FEATURES

- Full-duplex, 2-wire asynchronous data transfer.
- Single-wire half-duplex communication
- Register locations conform to 16550 industry standard.
- Receiver FIFO trigger points at 1 byte.
- Built-in baud rate generator.
- Software or hardware flow control.

15.3 PIN DESCRIPTION

Pin Name	Type	Description	GPIO Configuration
UTXDn	O	Serial Transmit data.	
URXDn	I	Serial Receive data.	Depends on GPIO _n _CFG

15.4 BLOCK DIAGRAM



15.5 BAUD RATE CALCULATION

The UART baud rate is calculated as:

$$\text{UART}_{\text{BAUDRATE}} = \frac{\text{UARTn_PCLK}}{\text{Oversampling} \times (256 \times \text{DLM} + \text{DLL}) \times (1 + \text{DIVADDVAL} / \text{MUL})}$$

Where UARTn_PCLK is the peripheral clock, [UARTn_DLM](#) and [UARTn_DLL](#) are the standard UART baud rate divider registers, and DIVADDVAL and MULVAL are UART fractional baud rate generator specific parameters in [UARTn_FD](#) register.

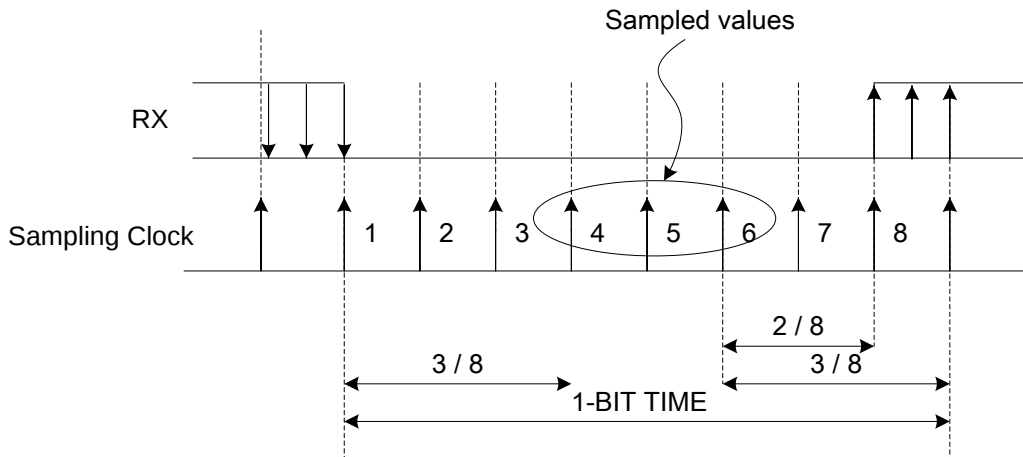
The value of MUL and DIVADDVAL should comply with the following conditions:

1. $1 \leq \text{MUL} \leq 15$
2. $0 \leq \text{DIVADDVAL} \leq 14$
3. $\text{DIVADDVAL} < \text{MUL}$
4. $\text{MUL} - \text{DIVADDVAL} \neq 2$
5. Oversampling is 8 or 16

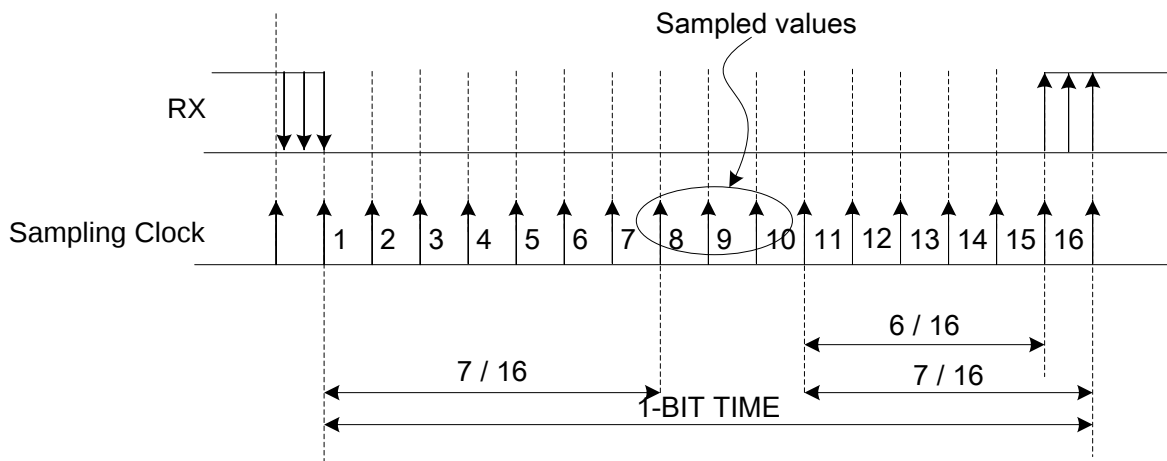
The value of the [UARTn_FD](#) register should not be modified while transmitting/receiving data or data may be lost or corrupted.

The oversampling method can be selected by programming the OVER8 bit in [UARTn_FD](#) register and can be either 16 or 8 times the baud rate clock.

- OVER8=1: Oversampling by 8 to achieve higher speed (up to UARTn_PCLK/8). In this case the maximum receiver tolerance to clock deviation is reduced.



- OVER8=0: Oversampling by 16 to increase the tolerance of the receiver to clock deviations. In this case, the maximum speed is limited to maximum UARTn_PCLK/16



If the [UARTn_FD](#) register value does not comply with these two requests, then the fractional divider output is undefined. If DIVADDVAL is zero then the fractional divider is disabled, and the clock will not be divided.

UART can operate with or without using the Fractional Divider. The desired baud rate can be achieved using several different Fractional Divider settings. The following algorithm illustrates one way of finding a set of DLM, DLL, MUL, and DIVADDVAL values. Such set of parameters yields a baud rate with a relative error of less than 1.1% from the desired one.

The following example illustrates selecting the DIVADDVAL, MUL, DLM, and DLL to generate BR = 115200 when UARTn_PCLK = 12 MHz, and Oversampling = 16.

$$\text{UART}_{\text{BAUDRATE}} = \frac{\text{UARTn_PCLK}}{\text{Oversampling} \times (256 \times \text{DLM} + \text{DLL}) \times (1 + \text{DIVADDVAL} / \text{MUL})}$$

$$115200 = \frac{12000000}{16 \times (256 \times \text{DLM} + \text{DLL}) \times (1 + \text{DIVADDVAL} / \text{MUL})}$$

$$(256 \times \text{DLM} + \text{DLL}) \times (1 + \text{DIVADDVAL} / \text{MUL}) = 6.51$$

Since the value of MULVAL and DIVADDVAL should comply with the following conditions:

1. $1 \leq \text{MUL} \leq 15$
2. $0 \leq \text{DIVADDVAL} \leq 14$
3. $\text{DIVADDVAL} < \text{MUL}$
4. $\text{MULVAL} - \text{DIVADDVAL} \neq 2$

Thus, the suggested UART settings would be: DLM = 0, DLL = 4, DIVADDVAL = 5, and MUL = 8 (fill in 7 in the MULVAL bits). The baud rate generated is 115384, and has a relative error of 0.16% from the originally specified 115200.

15.6 UART REGISTERS

Base Address: 0x4001 6000 (UART0)
0x4005 6000 (UART1)

15.6.1 UART n Receiver Buffer register (UARTn_RB) (n=0,1)

Address Offset: 0x00

This register is the byte of the UART RX FIFO, and contains the character received and can be read via the bus interface. The LSB (bit 0) contains the first-received data bit. If the character received is less than 8 bits, the unused MSBs are padded with zeros.

The Divisor Latch Access Bit (DLAB) in the [UARTn_LC](#) register must be zero in order to access this register. Since PE, FE and BI bits correspond to the byte on the top of the UART RX FIFO (i.e. the one that will be read in the next read from this register), the right approach for fetching the valid pair of received byte and its status bits is first to read the content of the [UARTn_LS](#) register, and then to read a byte from this register.

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	RB[7:0]	Contains the received byte in the UART RX FIFO.	R	0

15.6.2 UART n Transmitter Holding register (UARTn_TH) (n=0,1)

Address Offset: 0x00

This register is the byte of the UART TX FIFO. The byte is the character in the TX FIFO and can be written via the bus interface. The LSB represents the first bit to transmit.

The Divisor Latch Access Bit (DLAB) in [UARTn_LC](#) register must be zero in order to access this register.

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	TH[7:0]	The byte will be sent when it is the byte in TX FIFO and the transmitter is available.	W	0

15.6.3 UART n Divisor Latch LSB registers (UARTn_DLL) (n =0,1)

Address Offset: 0x00

The UART Divisor Latch is part of the UART Baud Rate Generator and holds the value used (optionally with the Fractional Divider) to divide the UARTn_PCLK clock in order to produce the baud rate clock, which must be the multiple of the desired baud rate that is specified by the Oversampling Register (typically 16X).

The UARTn_DLL and UARTn_DLM registers together form a 16-bit divisor, and DLAB bit in [UARTn_LC](#) register must be one in order to access these registers.

DLL contains the lower 8 bits of the divisor and DLM contains the higher 8 bits. A zero value is treated like 0x0001.

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	DLL[7:0]	The UART Divisor Latch LSB Register, along with the DLM register, determines the baud rate of the UART.	RW	0

15.6.4 UART n Divisor Latch MSB register (UARTn_DLM) (n=0,1)

Address Offset: 0x04

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	DLM[7:0]	The UART Divisor Latch MSB Register, along with the DLL register, determines the baud rate of the UART.	R/W	0

15.6.5 UART n Interrupt Enable register (UARTn_IE) (n=0,1)

Address Offset: 0x04

The DLAB bit in the [UARTn_LC](#) register must be zero in order to access this register.

Bit	Name	Description	Attribute	Reset
31:5	Reserved		R	0
4	TEMTIE	TEMT interrupt enable bit. The status of this interrupt can be read from TEMT bit in UARTn_LS register. 0: Disable 1: Enable	R/W	0
3	Reserved		R	0
2	RLSIE	Receive Line Status (RLS) interrupt enable bit. The status of this interrupt can be read from UARTn_LS [4:1]. 0: Disable 1: Enable	R/W	0
1	THREIE	THRE interrupt enable bit. The status of this interrupt can be read from THRE bit in UARTn_LS register. 0: Disable 1: Enable	R/W	0
0	RDAIE	RDA interrupt enable bit. Enables the Receive Data Available interrupt. It also controls the Character Receive Time-out interrupt. 0: Disable 1: Enable	R/W	0

15.6.6 UART n Interrupt Identification register (UARTn_II) (n=0,1)

Address Offset: 0x08

This register provides a status code that denotes the priority and source of a pending interrupt.

The interrupts are frozen during a UARTn_II register access. If an interrupt occurs during a UARTn_II register access, the interrupt is recorded for the next UARTn_II register access.

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	100b
3:1	INTID[2:0]	Interrupt identification which identifies an interrupt corresponding to the UARTn RX FIFO. 0x3: 1 - Receive Line Status (RLS). 0x2: 2a - Receive Data Available (RDA). 0x1: 3a - THRE Interrupt. 0x7: 3b – TEMT Interrupt Other: Reserved	R	0
0	INTSTATUS	Interrupt status. The pending interrupt can be determined by evaluating	R	1

		UARTn_I[3:1]. 0: At least one interrupt is pending. 1: No interrupt is pending.		
--	--	---	--	--

Given the status of UARTn_I[3:0], an interrupt handler routine can determine the cause of the interrupt and how to clear the active interrupt. The UARTn_I register must be read in order to clear the interrupt prior to exiting the Interrupt service routine.

Interrupt	UARTn_I [3:0]	Priority	Interrupt Source	Interrupt Reset
RLS	0110	Highest	Overrun error (OE), Parity error (PE), Framing error (FE) or Break interrupt (BI)	Read UARTn_LS register
RDA	0100	2 nd	RX data in FIFO reached trigger level (FCR0=1)	Read UARTn_RB register or UART FIFO drops below trigger level
THRE	0010	3 rd	THRE	Read UARTn_I register (if source of interrupt) or Write THR register
TEMT	1110	3 rd	TEMT	Read UARTn_I register (if source of interrupt) or Write THR register

15.6.7 UART n FIFO Control register (UARTn_FIFOCtrl) (n=0,1)

Address Offset: 0x08

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:6	RXTL[1:0]	RX Trigger Level. These two bits determine how many receiver UART FIFO characters must be written before an interrupt is activated. 00: Trigger level 0 (1 character) Other: Reserved	W	0
5:1	Reserved		R	0
0	FIFOEN	FIFO enable 0: No effect 1: Enable for both UART Rx and TX FIFOs and UARTn_FIFOCtrl[7:1] access. This bit must be set for proper UART operation.	W	1

15.6.8 UART n Line Control register (UARTn_LC) (n=0,1)

Address Offset: 0x0C

This register determines the format of the data character that is to be transmitted or received.

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7	DLAB	Divisor Latch Access bit 0: Disable access to Divisor Latches. 1: Enable access to Divisor Latches.	R/W	0
6	BC	Break Control bit 0: Disable break transmission. 1: Enable break transmission. Output pin UART TXD is forced to logic 0.	R/W	0

5:4	PS[1:0]	Parity Select bits 00: Odd parity. Number of 1s in the transmitted character and the attached parity bit will be odd. 01: Even Parity. Number of 1s in the transmitted character and the attached parity bit will be even. 10: Forced 1 stick parity. 11: Forced 0 stick parity.	R/W	0
3	PE	Parity Enable bit 0: Disable parity generation and checking. 1: Enable parity generation and checking.	R/W	0
2	SBS	Stop Bit Select bit 0: 1 stop bit. 1: 2 stop bits (1.5 if WLS bits=00)	R/W	0
1:0	WLS[1:0]	Word Length Select bits 00: 5-bit character length. 01: 6-bit character length. 10: 7-bit character length. 11: 8-bit character length.	R/W	0

15.6.9 UART n Line Status register (UARTn_LS) (n=0,1)

Address Offset: 0x14

*** Note:**

1. The break interrupt (BI) is associated with the character in the UARTn_RB FIFO.
2. The framing error (FE) is associated with the character in the UARTn_RB FIFO.
3. The parity error (PE) is associated with the character in UARTn_RB FIFO.

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7	RXFE	Error in RX FIFO flag. RXFE =1 when a character with a RX error such as framing error, parity error, or break interrupt, is loaded into the UARTn_RB register. This bit is cleared when the UARTn_LS register is read and there are no subsequent errors in the UART FIFO. 0: UARTn_RB register contains no UART RX errors or FIFOEN=0 1: UARTn_RB register contains at least one UART RX error.	R	0
6	TEMT	Transmitter Empty flag TEMT=1 when both THR and TSR are empty; TEMT is cleared when either the TSR or the THR contain valid data. 0: THR and/or TSR contains valid data. 1: THR and TSR are empty.	R	1
5	THRE	Transmitter Holding Register Empty flag THRE indicates that the UART is ready to accept a new character for transmission. In addition, this bit causes the UART to issue THRE interrupt to if THREIE=1. THRE=1 when a character is transferred from the THR into the TSR. The bit is reset to logic 0 concurrently with the loading of the Transmitter Holding Register by the CPU. 0: THR contains valid data. 1: THR (TX FIFO) is empty.	R	1
4	BI	Break Interrupt flag. When RXD1 is held in the spacing state (all zeros) for one full character transmission (start, data, parity, stop), a break interrupt occurs. Once the break condition has been detected, the receiver goes idle until RXD1 goes to marking state (all ones). A UARTn_LS register read clears BI bit. The time of break detection is dependent on FIFOEN bit in UARTn_FIFCTRL register. 0: Break interrupt status is inactive. 1: Break interrupt status is active.	R	0

3	FE	Framing Error flag. When the stop bit of a received character is a logic 0, a framing error occurs. A UARTn_LS register read clears FE bit. The time of the framing error detection is dependent on FIFOEN bit in UARTn_FIFOCtrl register. Upon detection of a framing error, the RX will attempt to re-synchronize to the data and assume that the bad stop bit is actually an early start bit. However, it cannot be assumed that the next received byte will be correct even if there is no Framing Error. 0: Framing error status is inactive. 1: Framing error status is active.	R	0
2	PE	Parity Error flag. When the parity bit of a received character is in the wrong state, a parity error occurs. A UARTn_LS register read clears PE bit. Time of parity error detection is dependent on FIFOEN bit in UARTn_FIFOCtrl register. 0: Parity error status is inactive. 1: Parity error status is active.	R	0
1	OE	Overrun Error flag. The overrun error condition is set as soon as it occurs. A UARTn_LS register read clears OE bit. OE=1 when UART RSR has a new character assembled and the UARTn_RB FIFO is full. In this case, the UARTn_RB FIFO will not be overwritten and the character in the UARTn_RS register will be lost. 0: Overrun error status is inactive. 1: Overrun error status is active.	R	0
0	RDR	Receiver Data Ready flag RDR=1 when the UARTn_RB FIFO holds an unread character and is cleared when the UARTn_RB FIFO is empty. 0: UARTn_RB FIFO is empty. 1: UARTn_RB FIFO contains valid data.	R	0

15.6.10 UART n Scratch Pad register (UARTn_SP) (n=0,1)

Address Offset: 0x1C

This register has no effect on the UART operation. This register can be written and/or read at user's discretion. There is no provision in the interrupt interface that would indicate to the host that a read or write of this register has occurred.

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:0	PAD[7:0]	A readable, writable byte.	R/W	0

15.6.11 UART n Fractional Divider register (UARTn_FD) (n=0,1)

Address Offset: 0x28

This register controls the clock prescaler for the baud rate generation and can be read and written at the user's discretion. This prescaler takes the APB clock and generates an output clock according to the specified fractional requirements.

In most applications, the UART samples received data 16 times in each nominal bit time, and sends bits that are 16 input clocks wide. OVER8 bit allows software to control the ratio between the input clock and bit clock. This is required for smart card mode, and provides an alternative to fractional division for other modes.

*** Note:**

1. If the fractional divider is active (DIVADDVAL>0) and UARTn_DLM=0, the value of the UARTn_DLL register must ≥ 3 .
2. $MUL - DIVADDVAL \neq 2$, so $MULVAL[3:0] - DIVADDVAL[3:0] \neq 1$.

Bit	Name	Description	Attribute	Reset
31:9	Reserved		R	0
8	OVER8	Oversampling value 0: Oversampling by 16 1: Oversampling by 8	R/W	0
7:4	MULVAL[3:0]	Baud rate pre-scaler multiplier value MUL = MULVAL[3:0] +1 0000: Baud rate pre-scaler multiplier value is 1 for HW 0001: Baud rate pre-scaler multiplier value is 2 for HW ... 1111: Baud rate pre-scaler multiplier value is 16 for HW.	R/W	0
3:0	DIVADDVAL[3:0]	Baud rate generation pre-scaler divisor value. If this field is 0, fractional baud rate generator will not impact the UART baud rate	R/W	0

15.6.12 UART n Control register (UARTn_CTRL) (n=0,1)

Address Offset: 0x30

In addition to HW flow control (Auto-CTS and Auto-RTS mechanisms), this register enables implementation of SW flow control.

When TXEN = 1, the UART transmitter will keep sending data as long as they are available. As soon as TXEN bit becomes 0, UART transmission will stop.

It is strongly suggested to let the UART HW implemented auto flow control features take care of limit the scope of TXEN to SW flow control.

*** Note: It is advised that TXEN and RXEN are set in the same instruction if needed in order to minimize the setup and the hold time of the receiver.**

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7	TXEN	0: Disable TX function. The transfer of characters from the UARTn_TH register or TX FIFO into the transmit shift register is blocked. While a character is being sent, the transmission of that character is completed, but no further characters are sent until this bit is set again. 1: Data written to the UARTn_TH register is output on the TXD pin as soon as any preceding data has been sent.	R/W	1
6	RXEN	0: Disable RX related function 1: Enable RX	R/W	1
5:4	Reserved		R	0
3:1	MODE[2:0]	UARTn Mode 000: UART mode. HW will switch GPIO to UTXDn and URXDn. Other: Reserved	R/W	0
0	UARTEN	UART enable 0: Disable . All UART shared pins act as GPIO. 1: Enable. HW switches GPIO to UART pin according to MODE bits automatically.	R/W	0

15.6.13 UART n Half-duplex Enable register (UARTn_HDEN) (n=0,1)

Address Offset: 0x34

After reset the UART will be in full-duplex mode, meaning that both TX and RX work independently. After setting the HDEN bit, the UART will be in half-duplex mode. In this mode, the UART ensures that the receiver is locked when idle, or will enter a locked state after having received a complete ongoing character reception. Line conflicts must be handled in SW.

The behavior of the UART is unpredictable when data is presented for reception while data is being transmitted. For this

reason, the value of the HDEN register should not be modified while sending or receiving data, or data may be lost or corrupted.

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	HDEN	Half-duplex mode enable bit 0: Disable 1: Enable	R/W	0

16 CYCLIC REDUNDANCY CHECK (CRC)

16.1 OVERVIEW

The CRC (cyclic redundancy check) calculation unit is used to get a CRC code from 16- or 32-bit data word and a generator polynomial. Among other applications, CRC-based techniques are used to verify data transmission or storage integrity. The CRC calculation circuit helps compute a signature of the software during runtime, to be compared with a reference signature generated at link time and stored at a given memory location.

16.2 FEATURES

1. Support

CRC-32 polynomial: $X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$

CRC-16 polynomial: $X^{16}+X^{15}+X^2+1$

CRC-16-CCITT polynomial: $X^{16}+X^{12}+X^5+1$

	CRC-16-CCITT $X^{16}+X^{12}+X^5+1$	CRC-16 $X^{16}+X^{15}+X^2+1$	CRC-32(-IEEE802.3) $X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$
Poly	0x1021	0x8005	0x04C11DB7
Seed(Init)	0xFFFF	0x0000	0xFFFFFFFF
XOROut	0x0000	0x0000	0xFFFFFFFF
RefIn	No	Yes	Yes
RefOut	No	Yes	Yes

2. Handles 16-, 32-bit data size

3. Single input/output 32-bit data register

4. Input buffer to avoid bus stall during calculation

5. CRC computation done in 4T IHRC clock cycles for the 32-bit data size

6. Polynomial representations of cyclic redundancy checks

16.3 CRC REGISTERS

Base Address: 0x4003 8000

16.3.1 CRC Control register (CRC_CTRL)

Address offset: 0x00

Bit	Name	Description	Attribute	Reset
31:6	Reserved	Reserved	R/W	0
4	BUSY	CRC calculation busy flag. 0: CRC calculation Idle/Finished. 1: CRC calculation is in process.	R	0
3	URCRCEN	Enable bit of the CRC calculation for the User ROM, except the last page. 1: Start the CRC operation for the User ROM, except the last page. This bit is set only by SW and reset by HW. 0: Stop/Finish operation.	R/W	0
2	RESET	Reset bit 0: No effect. 1: Reset the CRC calculation circuit. (Reset the initial seed value and BUSY bit to 0). Clear this bit when the reset operation had finished by HW.	R/W	0
1:0	CRC[1:0]	CRC Polynomial 00: CRC-16-CCITT 01: CRC-16 10: CRC-32 11: Reserved	R/W	0

16.3.2 CRC Data register (CRC_DATA)

Address offset: 0x04

* **Note: Support 8-bit (Byte) Write ONLY!**

Bit	Name	Description	Attribute	Reset
31:0	DATA[31:0]	Data to be input or read. Write: Input 8-bit data to the CRC calculator and start to calculation process. Read: Output the previous CRC calculation result depends on the CRC Polynomial.	R/W	0

17 FLASH

17.1 OVERVIEW

SONiX 32-bit MCU integrated device feature in-system programmable (ISP) FLASH memory for convenient, upgradeable code storage. The FLASH memory may be programmed via the SONiX 32-bit MCU programming interface or by application code for maximum flexibility. SONiX 32-bit MCU provides security options at the disposal of the designer to prevent unauthorized access to information stored in FLASH memory.

- The MCU is stalled during Flash program and erase operations, although peripherals (Timers, WDT, I/O, PWM, etc.) remain active.
- Watchdog timer should be cleared if enabled before the Flash write or erase operation.
- The erase operation sets all the bits in the Flash page to logic 1.
- HW will hold system clock and automatically move out data from RAM and do programming, after programming finished, HW will release system clock and let MCU execute the next instruction.

17.2 EMBEDDED FLASH MEMORY

The Flash memory is organized as 32-bit wide memory cells that can be used for storing both code and data constants, and is located at a specific base address in the memory map of chip.

The high-performance Flash memory module in chip has the following key features:

- Memory organization: the Flash memory is organized as a User ROM, Boot ROM.

User ROM	8K × 32 bits divided into 64 pages of 512 Bytes
Boot ROM	1K × 32 bits divided into 8 pages of 512 Bytes

The Flash interface implements instruction access and data access based on the AHB protocol. It implements the logic necessary to carry out Flash memory operations (Program/Erase). Program/Erase operations can be performed over the whole product voltage range.

17.3 FEATURES

- Read interface (32-bit)
- Flash Program / Erase operation
- Code Option includes Code Security (CS)

Write operations to the main memory block and the code options are managed by an embedded Flash Memory Controller (FMC). The high voltage needed for Program/Erase operations is internally generated. The main Flash memory can be read/write protected against different levels of Code Security (CS).

During a write operation to the Flash memory, any attempt to read the Flash memory will stall the bus. The read operation will proceed correctly once the write operation has completed. This means that code or data fetches cannot be made while a write/erase operation is ongoing.

For write and erase operations on the Flash memory, the IHRC will be turn ON by FMC. The Flash memory can be programmed and erased using ICP and ISP.

17.4 ORGANIZATION

Block	Name	Base Address	Size (Byte)
User ROM	Page 0	0x00000000 ~ 0x000001FF	512
	Page 1	0x00000200 ~ 0x000003FF	512
	.	.	.
	Page 63	0x00007E00 ~ 0x00007FFF	512
Boot Loader	Page 0	0x1FFF0000 ~ 0x1FFF01FF	512
	Page 1	0x1FFF0200 ~ 0x1FFF03FF	512
	.	.	.
	Page 7	0x1FFF0E00 ~ 0x1FFF0FFF	512

17.5 READ

The embedded Flash module can be addressed directly, as a common memory space. Any data read operation accesses the content of the Flash module through dedicated read senses and provides the requested data.

The read interface consists of a read controller on one side to access the Flash memory, and an AHB interface on the other side to interface with the CPU. The main task of the read interface is to generate the control signals to read from the Flash memory as required by the CPU.

17.6 PROGRAM/ERASE

The Flash memory erase operation can be performed at page level.

To ensure that there is no over-programming, the Flash programming and erase controller blocks are clocked by IHRC.

17.7 EMBEDDED BOOT LOADER

The embedded boot loader is used to reprogram the Flash memory using the UART0 serial interface. This program is located in the Boot ROM and is programmed by SONiX during production.

17.8 FLASH MEMORY CONTROLLER (FMC)

The FMC handles the program and erase operations of the Flash memory.

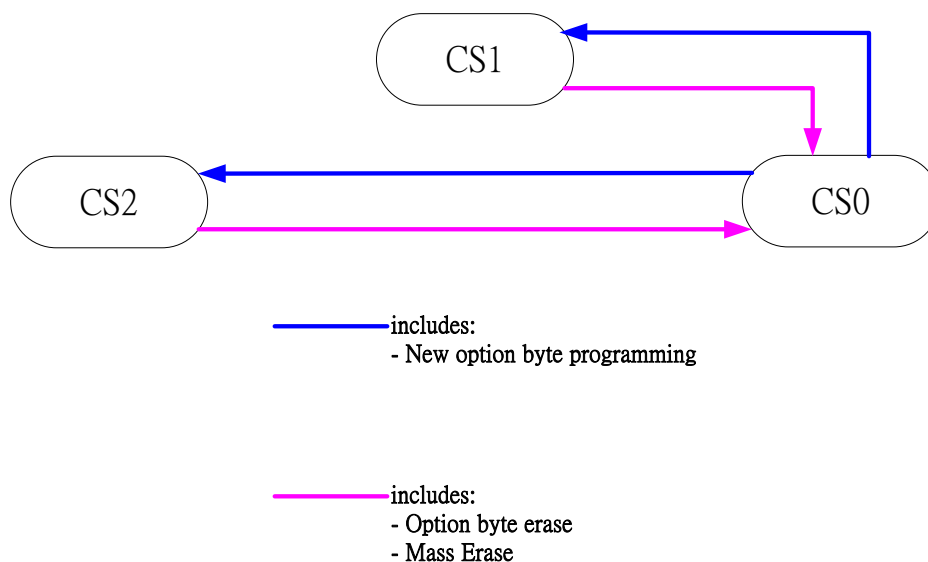
17.8.1 CODE SECURITY (CS)

Code Security is a mechanism that allows the user to enable different levels of security in the system so that access to the on-chip Flash and use of the ISP can be restricted.

* **Note: Any Code Security change becomes effective only after the MCU has been Reboot.**

User ROM		CS0	CS1	CS2	Description
WRITER	Read	O	X	X	
	Erase	O	O(*)	O(*)	(*) WRITER will change the CS level to CS0.
	Program	O	O	O	
FW (EEPROM emulation)	Read	O	O	O	
	Erase	O	O	O	
	Program	O	O	O	
SWD	Read	O	X	X	
	Erase	O	X	X	
	Program	O	X	X	

* **Note: User may try to change security level from CS2 to CS0, or from CS1 to CS0. HW shall:**
1. Mass erase the User ROM first. User shall NOT execute this operation in debug mode, since the SWD communication may fail during the mass erase procedure.
2. Update security level.



17.8.2 PROGRAM FLASH MEMORY

The Flash memory can be programmed 32 bits (4 bytes) at a time. CPU can program the main Flash memory by performing standard page write operations. The PG bit in the FLASH_CTRL register must be set. When the data is filled in the FLASH_DATA register, FMC preliminarily increases the data address, and checks the address to be programmed. If the following errors happen, the program operation is skipped and a warning is issued by the PGERR bit in FLASH_STATUS register.

- Start to Erase/Program and find that the address is over page boundary
- Start to Erase/Program and find that the address is illegal (>ROM size)
- Fill in Data and the address is already over Page Boundary
- The address to be programmed contains a value different from 0xFFFFFFFF before programming.

The main Flash memory programming sequence in standard mode is as follows:

1. Set the PG bit in the FLASH_CTRL register.
2. Fill in the target address in the FLASH_ADDR register.
3. Wait for the BUSY bit to be reset.
4. Perform the continuous data write until all of the data had been filled in the FLASH_DATA register.
5. Wait for the BUSY bit to be reset.
6. Set the START bit to start programming.
7. Wait for the BUSY bit to be reset.
8. (Optional) Read the programmed value and verify.

17.8.3 ERASE

The Flash memory can be erased page by page.

17.8.3.1 PAGE ERASE

A page of the Flash memory can be erased using the Page Erase feature of the FMC. To erase a page, the procedure below should be followed:

1. Set the PER bit in the FLASH_CTRL register
2. Program the FLASH_ADDR register to select a page to be erased
3. Set the START bit in the FLASH_CTRL register
4. Wait for the BUSY bit to be reset
5. (Optional) Read the erased page and verify

17.8.3.2 MASS ERASE

When the Flash memory read protection is changed from protected to unprotected, a Mass Erase of the User ROM is performed by HW before reprogramming the read protection option.

17.9 READ PROTECTION

The read protection is activated by setting the Code Security bytes in Code option.

When the Flash memory read protection is changed from protected to unprotected, a Mass Erase of the User ROM is performed by HW before reprogramming the read protection option.

17.10 HW CHECKSUM

HW checksum is the checksum of User ROM. If the read protection is enabled, the users can still readout the HW checksum through Writer or ISP AP.

17.11 FMC REGISTERS

Base Address: 0x4006 2000

17.11.1 Flash Low Power Control register (FLASH_LPCTRL)

Address offset: 0x00

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:16	FMCKEY	FMC verify key. Read as 0. When writing to the register you must write 0x5AFA to FMCKEY, otherwise behavior of writing to the register is ignored.	W	0
15:6	Reserved		R	0
5:0	LPMODE[5:0]	Flash Low Power mode selection bit 000000b: HCLK ≤ 30MHz 011101b: HCLK > 30MHz Other: Reserved, and may cause unexpected error to force MCU enter Hardfault handler.	RW	000b

17.11.2 Flash Status register (FLASH_STATUS)

Address offset: 0x04

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:3	Reserved		R	0
2	ERR	Error flag 0: Read→No error. Write→Clear this flag. 1: Set by HW when - Start to Erase/Program and find that the address is over page boundary - Start to Erase/Program and find that the address is illegal. (>ROM size) - Fill in Data and the address is already over Page Boundary - The address to be programmed contains a value different from 0xFFFFFFFF before programming.	RW	0
1	Reserved		R	0
0	BUSY	Busy flag 0: Flash operation is not busy. 1: Flash operation is in progress. This is set on the beginning of a Flash operation and reset when the operation finishes or when an error occurs by HW.	R	0

17.11.3 Flash Control register (FLASH_CTRL)

Address offset: 0x08

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7	CHK	Checksum calculation chosen This bit is set only by SW and reset when the BUSY bit resets.	RW	0
6	START	Start Erase/Programming operation 1: Triggers an Erase/Programming operation. This bit is set only by SW and resets when the BUSY bit resets. 0: Stop/Finish operation.	RW	0

5:2	Reserved		R	0
1	PER	Page Erase mode chosen. This bit is set only by SW and reset when the BUSY bit resets.	R/W	0
0	PG	Flash Programming mode chosen. This bit is set only by SW and reset when the BUSY bit resets.	R/W	0

17.11.4 Flash Data register (FLASH_DATA)

Address offset: 0x0C

For Page Program operations, this should be updated by SW to indicate the data to be programmed.

Bit	Name	Description	Attribute	Reset
31:0	DATA[31:0]	Data to be programmed.	R/W	0

17.11.5 Flash Address register (FLASH_ADDR)

Address offset: 0x10

The Flash address to be erased or programmed should be updated by SW, and the PG bit or PER bit shall be set before filling in the Flash address.

* <i>Note: Write access to this register is blocked when the BUSY bit in the FLASH_STATUS register is set.</i>				
--	--	--	--	--

Bit	Name	Description	Attribute	Reset
31:0	FAR[31:0]	Flash Address Choose the Flash start address to erase when Page Erase is selected, or to program when Page Program is selected.	R/W	0

17.11.6 Flash Checksum register (FLASH_CHKSUM)

Address offset: 0x14

Bit	Name	Description	Attribute	Reset
31:16	BRCHKSUM[15:0]	Checksum of Boot ROM.	R	0
15:0	URCHKSUM[15:0]	Checksum of User ROM.	R	0

17.11.7 Flash Write Protect register (FLASH_WP)

Address offset: 0x14

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	WP[15:0]	Flash Write protect key. 0x5AFA: The User ROM, except the last page, is Write-protected (Not able to be erased and programmed by FW). Other: The whole User ROM can be accessed.	R/W	0

18 SERIAL-WIRE DEBUG (SWD)

18.1 OVERVIEW

SWD functions are integrated into the ARM Cortex-M0. The ARM Cortex-M0 is configured to support up to four breakpoints and two watch points.

18.2 FEATURES

- Supports ARM Serial Wire Debug (SWD) mode.
- Direct debug access to all memories, registers, and peripherals.
- No target resources are required for the debugging session.
- Up to four breakpoints.
- Up to two data watch points that can also be used as triggers.

18.3 PIN DESCRIPTION

Pin Name	Type	Description	GPIO Configuration
SWCLK	I	Serial Wire Clock pin in SWD mode.	
SWDIO	I/O	Serial Wire Data Input/Output pin in SWD mode.	

18.4 DEBUG NOTE

18.4.1 LIMITATIONS

Debug mode changes the way in which reduced power modes work internal to the ARM Cortex-M0 CPU, and this ripples through the entire system. These differences mean that power measurements should not be made while debugging, the results will be higher than during normal operation in an application.

During a debugging session, the SysTick Timer is automatically stopped whenever the CPU is stopped. Other peripherals are not affected.

18.4.2 DEBUG RECOVERY

User code may disable SWD function in order to use P0.10 and P0.11 as GPIO, and may not debug by SWD function to debug or download FW any more.

SONiX provide Boot loader to check the status of P0.2 (BOOT pin) during boot procedure. If P0.2 is Low during Boot procedure, MCU will execute code in Boot loader instead of User code, so SWD function is not disabled.

Exit Boot loader, user code can still configure P0.2 as other functions such as GPIO.

*** Note: We strongly recommended NOT using BOOT pin as output pin to drive the LED, otherwise, the BOOT pin status may be low during boot procedure.**

18.4.3 INTERNAL PULL-UP/DOWN RESISTORS on SWD PINS

To avoid any uncontrolled IO levels, the device embeds internal pull-up and pull-down resistor on the SWD input pins:

- SWDIO/JTMS: Internal pull-up
- SWCLK/JTCK: Internal pull-down

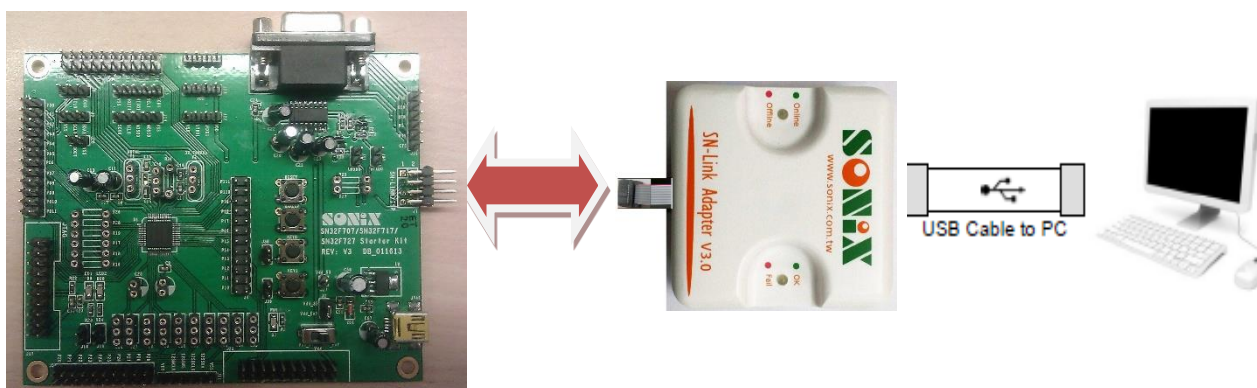
Once a SWD function is disabled by SW, the GPIO controller takes control again.

19 DEVELOPMENT TOOL

SONiX provides an Embedded ICE emulator system to offer 32-bit series MCU firmware development.

SONiX 32-bit series Embedded ICE Emulator System includes:

- SONiX 32-bit MCU Starter-Kit.
- SN-LINK-V3
- USB cable to provide communications between the SN-LINK-V3 and PC.
- IDE Tools (KEIL RVMDK)



SONiX 32-bit MCU Starter-Kit.

SN-LINK-V3

IDE Tools

SONiX 32-bit series Embedded ICE Emulator Feature:

- Target's Operating Voltage: 1.8V~5.5V.
- Up to 4 hardware break points.
- System clock rate up to 50MHz.
- Oscillator supports IHRC, ILRC, EHS/ELS X'tal.

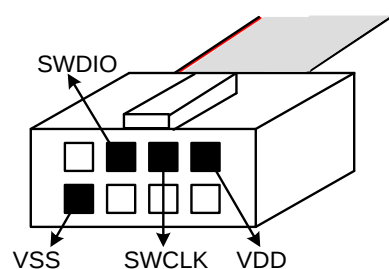
SONiX 32-bit series Embedded ICE Emulator Limitation:

- SWCLK and SWDIO pins are shared with GPIO pins. In embedded ICE mode, the shared GPIO function can't work.

19.1 SN-LINK-V3

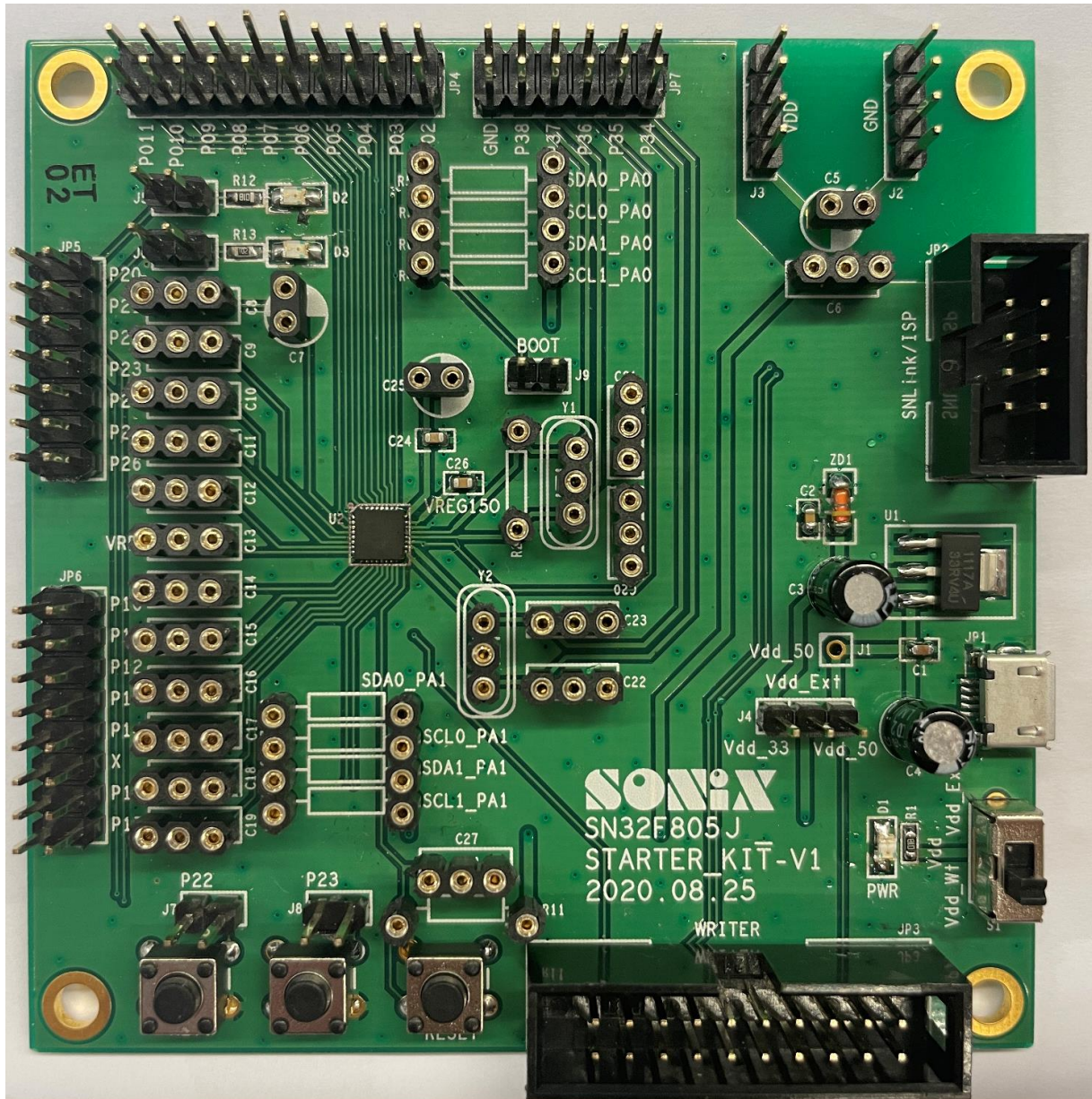
SN-LINK-V3 is a high speed emulator for SONiX 32-bit MCU. It debugs and programs based on SWD protocol. In addition to debugger functions, the SN-LINK-V3 also may be used as a programmer to load firmware from PC to MCU for engineering production, even mass production.

SN-LINK-V3 communicates with SONiX 32-bit MCU through SWD interface. The pin definition of the Modular cable is as following:



19.2 SN32F800 STARTER-KIT

SONiX 32-bit MCU Starter-kit is an easy-development platform. It includes real chip and I/O connectors to input signal or drive extra device of user's application. It is a simple platform to develop application as target board not ready. The starter-kit can be replaced by target board because of integrated SWD debugger circuitry.



- JP1: Micro USB connector.
- S1: VDD power source is 3.3V/5.0V from board, Writer, or external power.
- U2: SN32F805 real chip.
- D1: Power LED.
- C8~C19: 12-ch ADC capacitors.
- RESET button: External reset trigger source.
- Y1: External high-speed X'tal
- Y2: External low-speed 32.768KHz X'tal
- JP2: SN-LINK connector
- J9: Short to force MCU stay in Boot loader.

20 ELECTRICAL CHARACTERISTIC

20.1 ABSOLUTE MAXIMUM RATING

Supply voltage (Vdd).....	- 0.3V ~ 5.5V
Input in voltage (Vin).....	Vss – 0.2V ~ Vdd + 0.2V
Operating ambient temperature (Topr).....	-40°C ~ + 105°C
Storage ambient temperature (Tstor)	-40°C ~ + 125°C

20.2 ELECTRICAL CHARACTERISTIC

All of voltages refer to Vss, Typical Vdd = 3.3V, Fosc = 12MHz, ambient temperature is 25℃ unless otherwise note.							
PARAMETER	SYM.	DESCRIPTION		MIN.	TYP.	MAX.	UNIT
Operating Voltage	Vdd	Supply voltage for core and external rail		1.8	3.3	5.5	V
VDD rise rate	V _{POR}	VDD rise rate to ensure internal power-on reset		0.05	-	-	V/ms
Power Consumption							
Supply Current	Idd1	Normal mode	System clock = 12MHz [1][2][3]	-	1.5	-	mA
			System clock = 24MHz [1][3][4]	-	3	-	mA
			System clock = 36MHz [1][3][4]	-	3.5	-	mA
			System clock = 48MHz [1][3][4]	-	4	-	mA
			System clock = 60MHz [1][3][4]	-	5	-	mA
	Idd2	Sleep Mode	System clock = 12MHz [1][2][3][5]	-	500	-	uA
	Idd3		System clock = 32KHz [1][3][5]	-	100	-	uA
	Idd4	Deep-sleep Mode	Vdd=3.3V [1][3][5]	-	1	5	uA
Port Pins, RESET pin							
High-level input voltage	V _{IH}			0.7Vdd	-	Vdd	V
Low-level input voltage	V _{IL}			Vss	-	0.3Vdd	V
Input voltage	V _i			0	-	Vdd	V
Output voltage	V _o			0	-	Vdd	V
I/O port pull-up resistor	R _{PU}	Vin = Vss , Vdd = 5.0V Vin = Vss , Vdd = 3.3V		40 50	50 75	80 100	KΩ
I/O port pull-down resistor	R _{PD}	Vin = 5.0V Vin = 3.3V		25 20	35 25	50 35	KΩ
I/O High-level output source current	I _{OH}	Standard port and RESET pins	V _{OP} = Vdd– 0.5V	16	24	32	mA
I/O Low-level output sink current	I _{OL}	Standard port and RESET pins	V _{OP} = Vss + 0.5V	20	36	50	mA
ADC							
ADC Operating Voltage	V _{ADC}			2.0	-	5.5	V
AIN0 ~ AIN11 input voltage	V _{ani}			0	-	Avrefh	V
ADC Resolution	RES			10		12	Bit
ADC reference Voltage	V _{ref}			2.0	-	-	V
*ADC enable time	T _{ast}	Ready to start convert after set ADENB = “1”		100	-	-	us
*ADC current consumption	I _{ADC}	Vdd=3.3V, ADS=0		-	500	-	uA
ADC Clock Frequency	F _{ADCLK}			-	-	16	MHz
ADC Conversion Cycle Time	F _{ADCYL}			16	-	-	1/F _{ADCLK}
ADC Sampling Rate	F _{ADSMP}			-	-	1000	KHz

Differential Nonlinearity	DNL	Vdd= AVREFH=5.5V	-1	-	+1	LSB	
Integral Nonlinearity	INL	Vdd= AVREFH=5.5V	-2	+/-1	+2	LSB	
No Missing Code	NMC	Vdd= AVREFH=5.5V	10	-	12	Bit	
ADC offset Voltage	V _{ADCOffset}		-5	-	+5	mV	
ADC Internal voltage reference	V _{ADCIREF}	Internal VDD reference voltage, Vdd=5.5V		VDD		V	
		Internal 4.5V reference voltage, Vdd=5.5V	4.41	4.5	4.59	V	
		Internal 3V reference voltage, Vdd=5.5V	2.94	3	3.06	V	
		Internal 2.5V reference voltage, Vdd=5.5V	2.45	2.5	2.55	V	
		Internal 2V reference voltage, Vdd=5.5V	1.96	2	2.04	V	
Temp Sensor Slope	Slope			3.2*		mV/°C	
Temp Sensor Linearity	T _L		-3*	+/-1.5*	+3*	°C	
Temp Sensor Voltage at 25°C	V ₂₅		-	0.94*	-	V	
DAC							
DAC Operating Voltage	V _{DAC}		2.5	-	5.5	V	
DAC Resolution	RES			12		Bit	
Differential Nonlinearity	DNL	Vdd=5.5V , AVREFH=2.4V	-1	-	+1	LSB	
Integral Nonlinearity	INL	Vdd=5.5V , AVREFH=2.4V	-4	+/-1	+4	LSB	
DAC Clock Frequency	F _{DACLK}				200	KHz	
Output Current	I _{OUT}		-2		+2	mA	
Output Loading	C _{LOAD}				100	pF	
DAC Internal voltage reference	V _{DACIREF}	Internal 3.0V reference voltage, Vdd=5.5V	2.94	3	3.06	V	
		Internal 2.5V reference voltage, Vdd=5.5V	2.45	2.5	2.55	V	
		Internal 2.0V reference voltage, Vdd=5.5V	1.96	2	2.04	V	
COMPARATOR							
CMP Operating Voltage	V _{CMP}		1.8		5.5	V	
Supply Current	I _{COMP}	Vdd=5.5V		75	100	uA	
Input Offset Voltage	V _{OS}	V _{cm} =Vdd/2	-2		+2	mV	
Response Time	T _{rs}	Positive input voltage = 1/2*Vdd. Negative input voltage transitions from Vss to Vdd.		50*	100	ns	
Output Slew Rate Time	T _{sr}	Vo=rising Vss~Vdd or falling Vdd~Vss. Vdd=5V.	-	-	10	ns	
Common Mode Input Voltage Range	V _{cmr}	Vdd=5.5V	V _{ss} +0.5		V _{dd} -0.5	V	
CMP Internal voltage reference	V _{CMPIREF}	Internal 3V reference voltage, Vdd=5.5V	2.94	3	3.06	V	
FLASH							
Supply Voltage	V _{dd1}		1.35	1.50	1.65	V	
Endurance time	T _{EN}	Erase + Program	20K	*100K	-	Cycle	
Page Erase current	I _{PER}		-	1	2	mA	
Program current	I _{PG}		-	2.5	3.5	mA	
Page erase time	T _{PE}	1-Page (512 bytes)	4	-	5	ms	
Mass erase time	T _{MER}		20	-	40	ms	
1-Word Programming time	T _{PG}	1-Word (32 bits)		50	60	us	
MISC							
Low Voltage Detector	LVD	Interrupt/Reset	Level 0	1.67	1.70	1.73	V
			Level 1	2.57	2.60	2.63	V
			Level 2	2.67	2.70	2.73	V
			Level 3	2.77	2.80	2.83	V
			Level 4	2.87	2.90	2.93	V
			Level 5	2.97	3.00	3.03	V
IHRC Freq.	F _{IHRC}	T=25°C, Vdd=1.8V~ 5.5V	46.80	48	49.20	MHz	
		T=-40°C ~105°C, Vdd=1.8V~5.5V	46.80	48	49.20	MHz	
ESD_HBM	V _{ESD_HBM}	ESD human body mode	4000	-	-	V	
ESD_CDM	V _{ESD_CDM}	ESD charged device model	750	-	-	V	

* These parameters are for design reference, not tested.

[1] IDD measurements were performed with all pins configured as GPIO outputs driven LOW and pull-up resistors disabled, code while(1); executed, and VDD=3.3V

[2] IHRC and ILRC are enabled, external X'tal are disabled, and PLL is disabled.

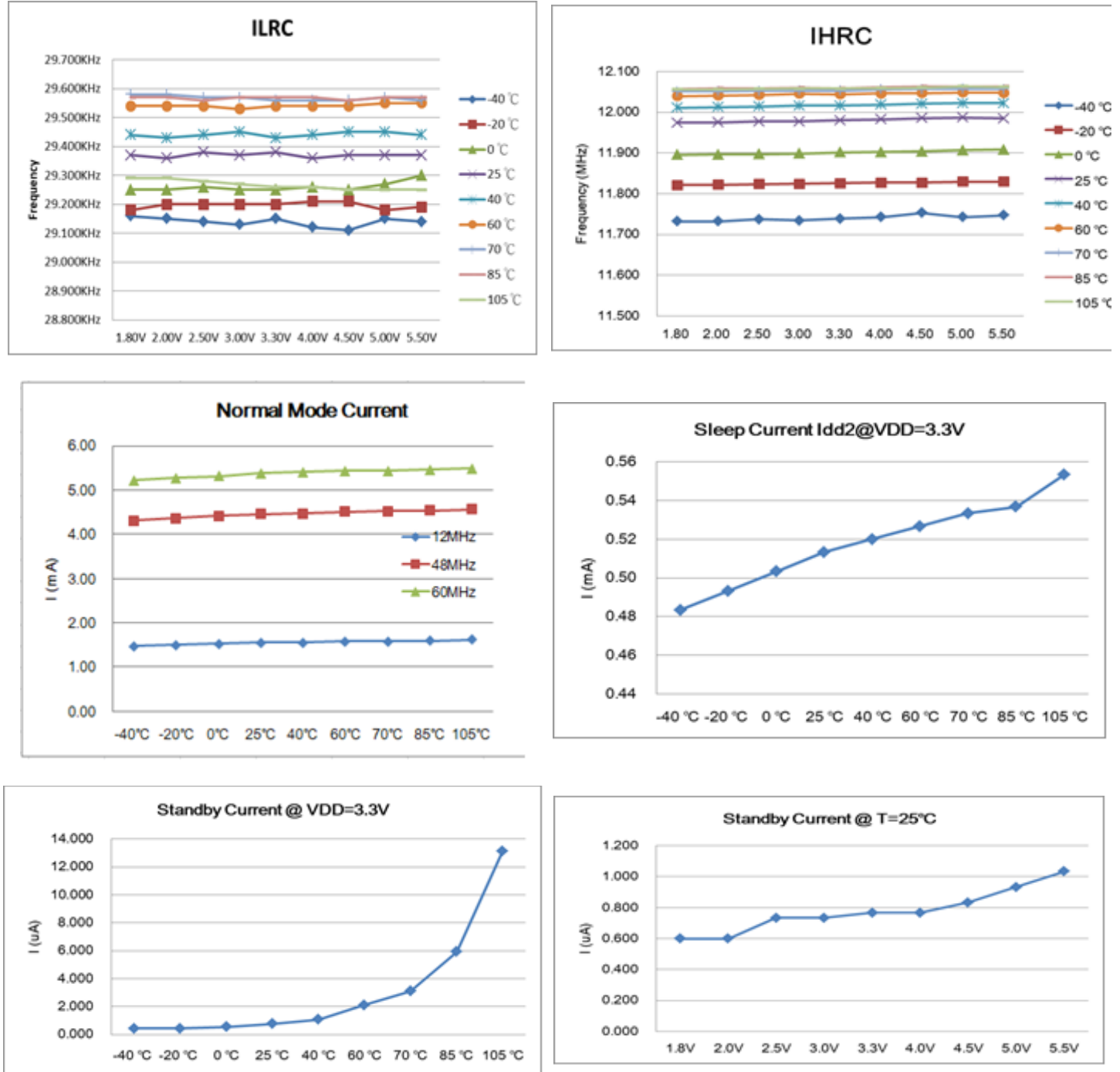
[3] LVD and all peripherals are disabled.

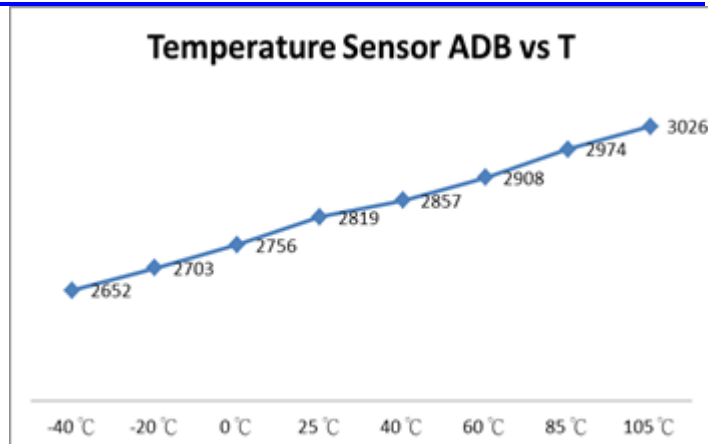
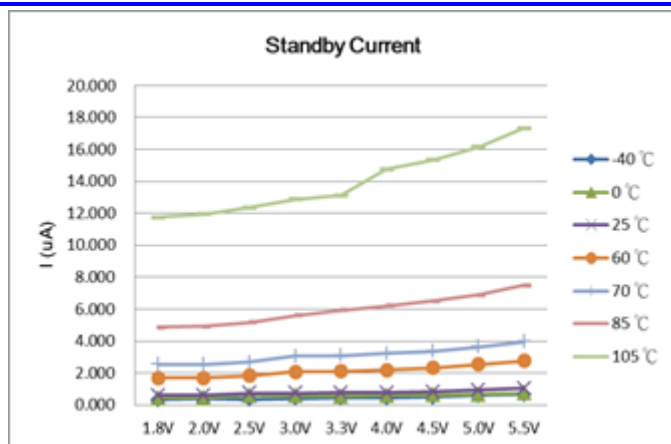
[4] IHRC is disabled, external high X'tal is enabled, and PLL is enabled.

[5] All oscillators and analog blocks are turned off.

20.3 CHARACTERISTIC GRAPHS

The Graphs in this section are for design guidance, not tested or guaranteed. In some graphs, the data presented are outside specified operating range. This is for information only and devices are guaranteed to operate properly only within the specified range.





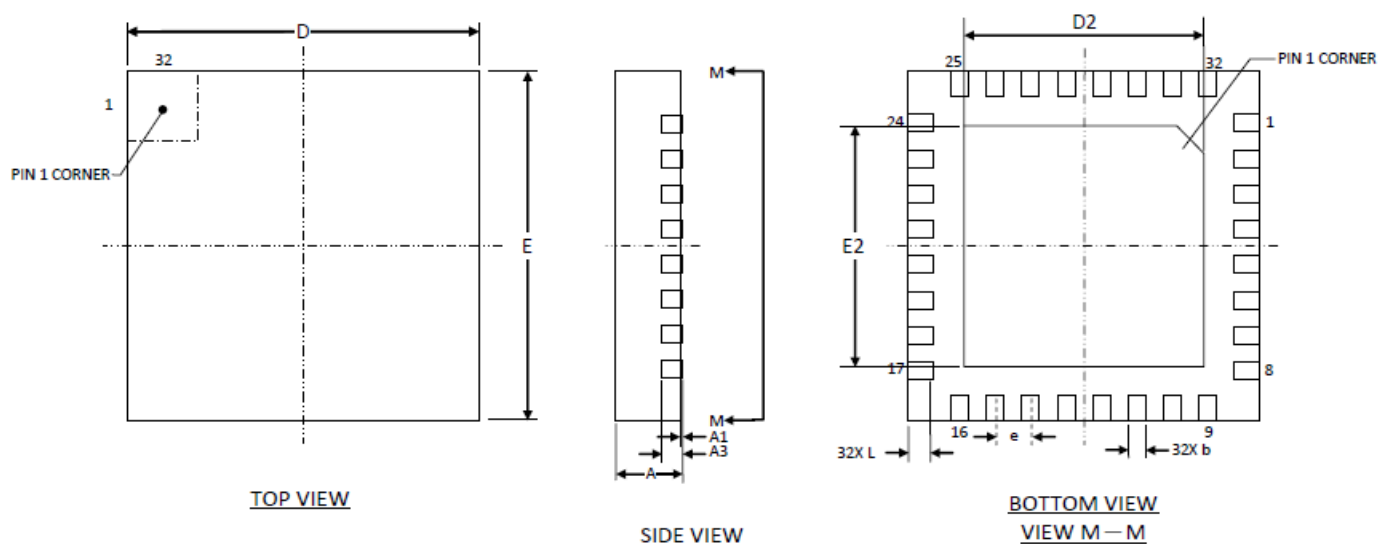
*Supply Current V.S. Operating Temperature (Operating Conditions: All pins configured as GPIO outputs driven Low and pull-up resistors disabled and VDD = 3.3V)

21 FLASH ROM PROGRAMMING PIN

Programming Information of SN32F800 Series											
Chip Name		SN32F805									
Writer Connector JP5											
Number	Name	Number	Pin	Number	Pin	Number	Pin	Number	Pin	Number	Pin
1	VDD	24	VDD								
2	GND	22	VSS								
3	CLK	10	P1.0								
4	CE										
5	PGM	1	P0.10								
6	OE	2	P0.11								
7	D1										
8	D0										
9	D3										
10	D2										
11	D5										
12	D4										
13	D7										
14	D6										
15	VDD										
16	-										
17	HLS										
18	RST										
19	-										
20	ALSB/PDB	13	P1.3								

22 PACKAGE INFORMATION

22.1 QFN 32 PIN 4x4

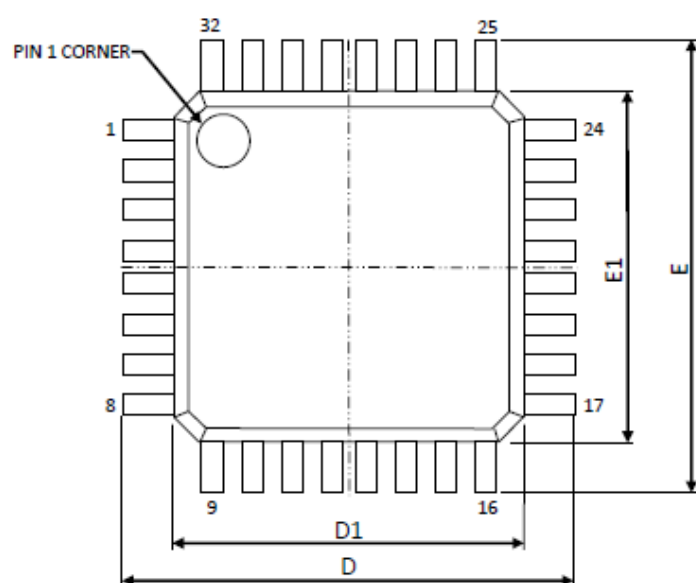


SYMBOLS	Dimension in mm			Dimension in inch		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	0.70	0.80	0.90	0.028	0.031	0.035
A1	0.00	0.02	0.05	0.000	0.000	0.002
A3	0.20 REF			0.008 REF		
b	0.15	0.20	0.25	0.006	0.008	0.010
D	4.00 BSC			0.157 BSC		
E	4.00 BSC			0.157 BSC		
e	0.40 BSC			0.016 BSC		
D2	2.00	2.45	2.9	0.080	0.096	0.114
E2	2.00	2.45	2.9	0.080	0.096	0.114
L	0.25	0.35	0.45	0.010	0.013	0.017

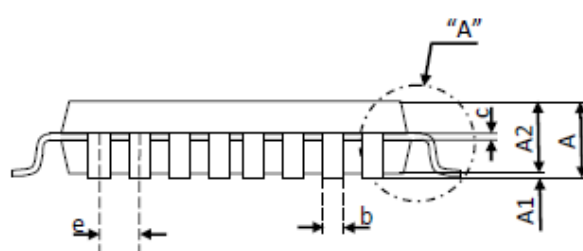
Notes :

1. CONTROLLING DIMENSION : MILLIMETER (mm)

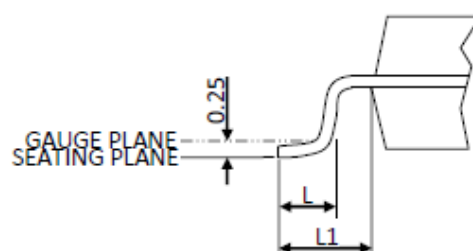
22.2 LQFP 32 PIN



TOP VIEW



SIDE VIEW



DETAIL "A"

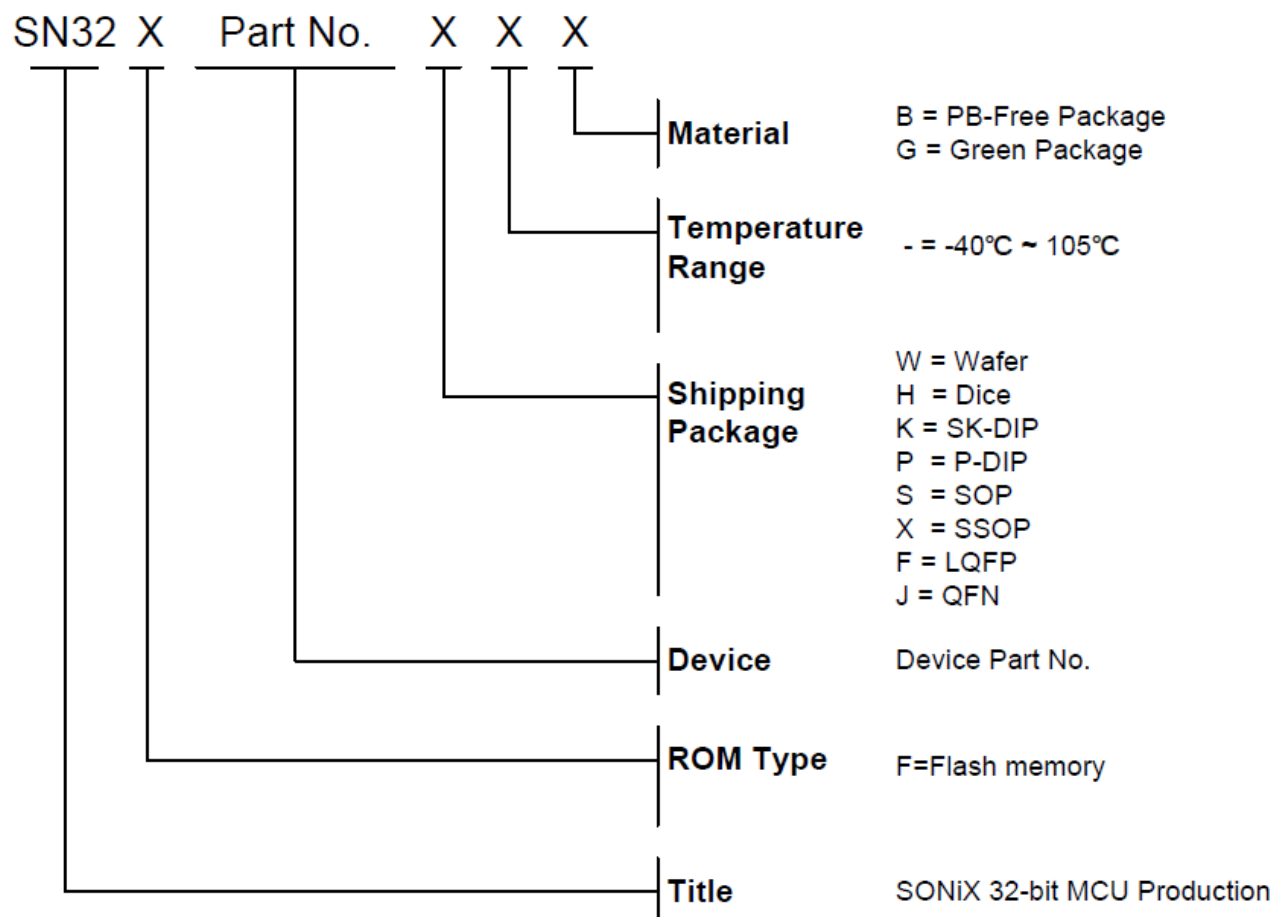
SYMBOLS	Dimension in mm			Dimension in inch		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	--	--	1.60	--	--	0.063
A1	0.05	--	0.25	0.002	--	0.01
A2	1.35	1.40	1.45	0.053	0.055	0.057
b	0.30	--	0.45	0.012	--	0.018
c	0.09	--	0.20	0.004	--	0.008
D	9.00 BSC			0.354 BSC		
D1	7.00 BSC			0.276 BSC		
E	9.00 BSC			0.354 BSC		
E1	7.00 BSC			0.276 BSC		
e	0.80 BSC			0.031 BSC		
L	0.40	0.60	0.80	0.016	0.024	0.031
L1	1.00 REF			0.039 REF		

Notes :

1. CONTROLLING DIMENSION : MILLIMETER (mm)
2. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION.
3. DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION.

23 MARKING DEFINITION

23.1 MARKING IDENTIFICATION SYSTEM



23.2 INTRODUCTION

There are many different types in SONiX 32-bit MCU production line.

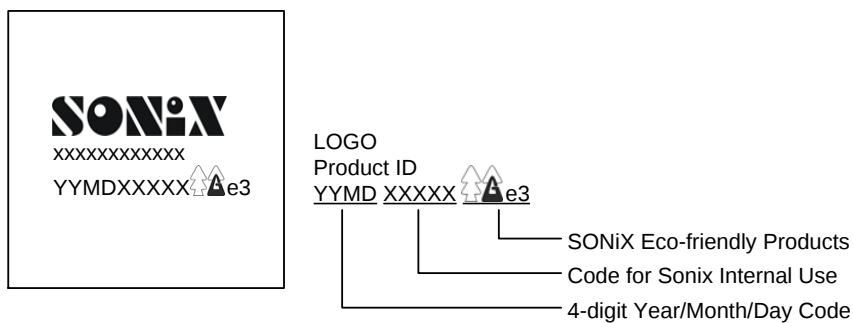
This note lists the marking definitions of all 32-bit MCU for order or obtaining information.

23.3 MARKING EXAMPLE

Name	ROM Type	Device	Package	Temperature	Material
SN32F805JG	Flash memory	800	QFN	-40°C ~105°C	Green Package
SN32F805FG	Flash memory	800	LQFP	-40°C ~105°C	Green Package
SN32F800W	Flash memory	800	Wafer	-40°C ~105°C	-
SN32F800H	Flash memory	800	Dice	-40°C ~105°C	-

23.4 DATECODE SYSTEM

The figure below is an example of the marking. Contents such as the product ID or symbol may vary according to different packages.



SONIX reserves the right to make change without further notice to any products herein to improve reliability, function or design. SONIX does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. SONIX products are not designed, intended, or authorized for use as components in systems intended, for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SONIX product could create a situation where personal injury or death may occur. Should Buyer purchase or use SONIX products for any such unintended or unauthorized application. Buyer shall indemnify and hold SONIX and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that SONIX was negligent regarding the design or manufacture of the part.

Main Office:

Address: 10F-1, NO. 36, Taiyuan Street., Chupei City, Hsinchu, Taiwan R.O.C.
Tel: 886-3-5600 888
Fax: 886-3-5600 889

Taipei Office:

Address: 15F-2, NO. 171, Song Ted Road, Taipei, Taiwan R.O.C.
Tel: 886-2-2759 1980
Fax: 886-2-2759 8180

Hong Kong Office:

Unit No.705,Level 7 Tower 1,Grand Central Plaza 138 Shatin Rural Committee Road, Shatin, New Territories, Hong Kong.
Tel: 852-2723-8086
Fax: 852-2723-9179

Technical Support by Email:

Sn8fae@sonix.com.tw