SN32F240C Series

USER'S MANUAL

SN32F248C/247C/246C/2451C

SONiX 32-Bit Cortex-M0 Micro-Controller

AMENDENT HISTORY

Version	Date	Description
0.1	2023/02/14	First version released.
0.2	2023/05/15	1. Fix the type error. 2. Modify the 1.1 FEATURES. 3. Added the 1.5 PIN ALLOCATION TABLE & 1.8 PIN CHARACTERISTICS. 4. Modify the 2.5 CODE OPTION TABLE. 5. Added the 3.2.2 PLL. 6. Added the Chapter 10 SPI1 description. 7. Added the Chapter 12 UART description. 8. Added the Chapter 13 CYCLIC REDUNDANCY CHECK. 9. Modify the Chapter 18 ELECTRICAL CHARACTERISTIC. 10. Modify the 21.4 DATECODE SYSTEM.
0.3	2023/06/03	1. Modify the 1.1 FEATURES. 2. Modify the 3.3.10 SYS0_IVTM description. 3. Modify the Chapter 18 ELECTRICAL CHARACTERISTIC. 4. Modify the 20 PACKAGE INFORMATION.
0.4	2023/07/12	1. Fix the type error. 2. Modify the 1.1 FEATURES. *3. Remove AIN17 & AIN18 in 7.1 OVERVIEW Diagram.* 4. Modify the Chapter 18 ELECTRICAL CHARACTERISTIC.
1.0	2023/12/07	1. Fix the 2.6 UNIQUE NUMBER register address.
1.1	2024/02/21	1. Modify the QFN33 to QFN32. 2. Modify the 1.3 CLOCK GENERATION BLOCK DIAGRAM. 3. Modify the 7.2 ADC CONVERSION TIME. 4. Add electrical characteristics of VDDIO1 and related pins. 5. Modify the 20 PACKAGE INFORMATION.
1.2	2024/05/29	1. Update 1.7 PIN CIRCUIT DIAGRAMS. 2. Add new chapter 10.6 TIMING CHARACTERISTICS. 3. Add ESD result in 18.2 ELECTRICAL CHARACTERISTIC. 4. Fix the 20 PACKAGE INFORMATION.

Table of Content

1 PRODUCT OVERVIEW

1.1 FEATURES

- **Memory configuration**
  128KB on-chip Flash programming memory.
  32KB SRAM.
  4KB Boot ROM.
  USB FIFO RAM: 512 bytes.
  SPI SRAM: 1KB.

- **Operation Frequency up to 48MHz**

- **Interrupt sources**
  ARM Cortex-M0 built-in Nested Vectored Interrupt Controller (NVIC).

- **I/O pin configuration**
  Up to 57 General Purpose I/O (GPIO) pins with configurable pull-up resistors.
  GPIO pins can be used as edge and level sensitive interrupt sources.
  Up to 20 High-current (420 mA) output sink pins P0.8~P0.15, P1.0~P1.11
  The other I/O pins with 20mA driving/sinking current.

- **Programmable WatchDog Timer (WDT)**
  Programmable watchdog frequency with watchdog clock source and divider.

- **System tick timer**
  24-bit timer.
  The system tick timer clock is fixed to the frequency of the system clock.
  The SysTick timer is intended to generate a fixed 10-ms interrupt.

- **LVD with separate thresholds**
  Reset: 1.35V for V_{CORE} 1.5V.
  Reset: 2.2V/2.7V/3.6V for VDD.
  Interrupt: 2.2V/2.7V/3.6V for VDD.

- **Full Speed USB 2.0**
  3.3v regulator output for D+ internal 1.5k pull-up resistor.
  Supports one Full speed USB device address.
  Supports PS/2 mode.
  One control Endpoint and 7 configurable INT/BULK Endpoints.
  EP0 supports 64-byte FIFO depth.
  Programmable EP1~EP7 FIFO depth.
  Total 8 endpoints share 512-byte USB RAM.

- **PFPA**

- **Working voltage 2.5V ~ 5.5V**

- **Timer**
  One 16-bit general purpose timer CT16B0 with CAP0.
  One 16-bit general purpose timer CT16B1 with 24-ch PWM.

- **Interfaces**
  - Two I2C controller supporting I2C-bus specification.
  - Two SPI controller supporting SPI protocol, add 1KB SRAM FIFO to lighting ARGB LEDs.
  - Three UART controller with fractional baud rate generation.

- **16+1-ch 12-bit SAR ADC with 4-level Int. Ref. Voltage**
  -16-ch external ADC input.
  -1-ch battery measurement.
  -4-level internal reference voltage source.(VDD, 4.5V, 3V, 2V)

- **System clocks**
  Internal high clock: RC type 12MHz.
  Internal low clock: RC type 32KHz.
  -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.

- **Serial Wire Debug (SWD)**

- **Operating modes**
  Normal, Sleep, and Deep-sleep.

- **Fcpu (Instruction cycle)**
  F_{CPU} = F_{HCLK} = F_{SYSCLK}/1, F_{SYSCLK}/2, F_{SYSCLK} /4, …, F_{SYSCLK} /128.

- **In-System-Programming (ISP) supported**

- **3.3V Regulator output**
  Power for USB D+ internal pull-up resistor.

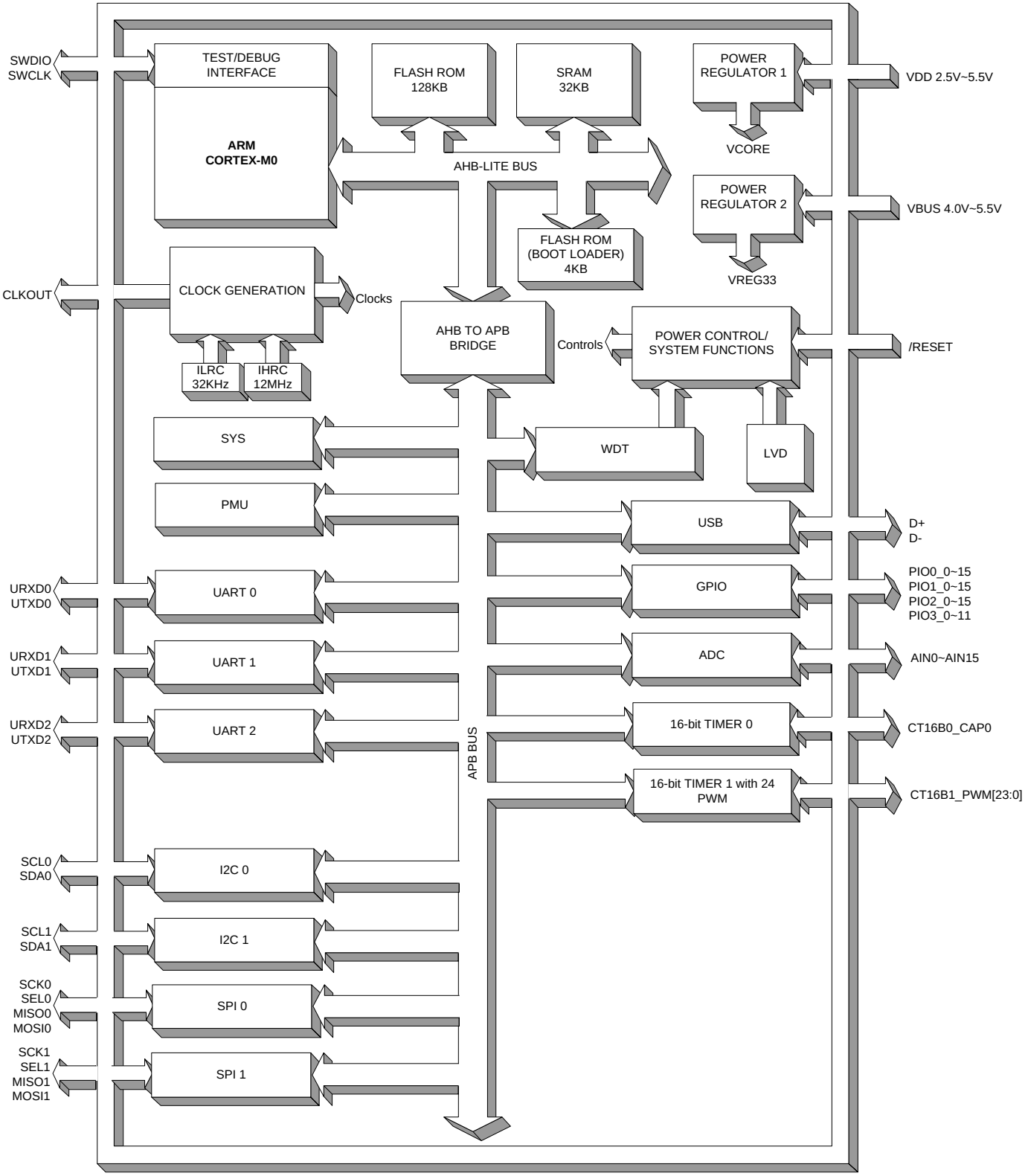
- **Cyclic Redundancy Check (CRC)**
  CRC-16
  CRC-16-CCITT
  CRC-32

- **Package (Chip form support)**
  LQFP64/48 pin
  QFN46/32 pin

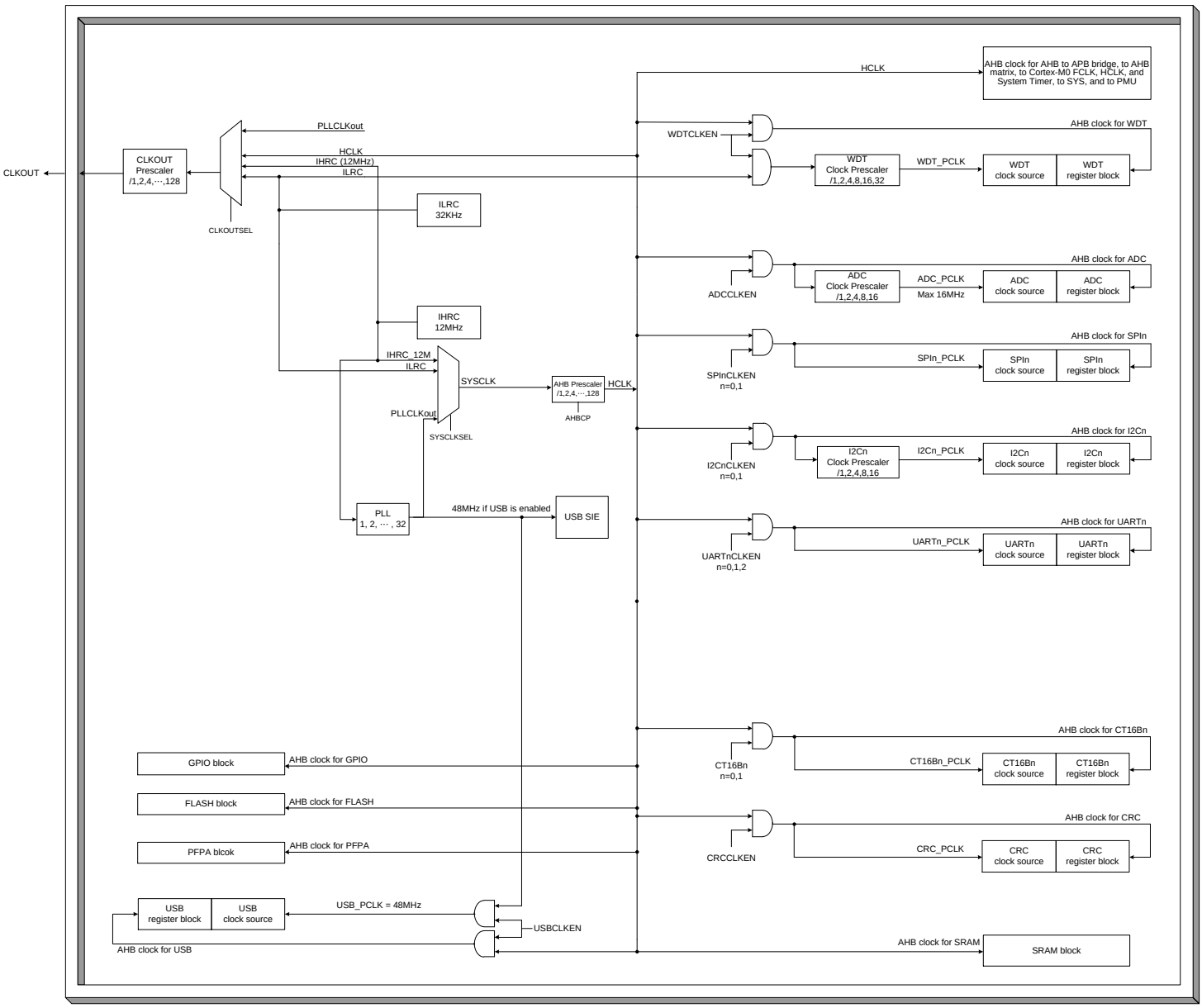
☞ Features Selection Table

Chip	ROM	RAM	F_{CPU} (Max MHz)	TIMER	UART	I2C	SPI	PWM	12-bit ADC	GPIO with Wakeup	Package
SN32F248CF	128KB	32KB	48 MHz	16-bitx2	3	2	2	24-CH	16+1	57	LQFP64
SN32F247CF	128KB	32KB	48 MHz	16-bitx2	3	2	2	24-CH	13+1	41	LQFP48
SN32F246CJ	128KB	32KB	48 MHz	16-bitx2	3	2	2	23-CH	11+1	39	QFN46
SN32F2451CJ	128KB	32KB	48 MHz	16-bitx2	1	2	2	17-CH	6+1	24	QFN32

1.2 SYSTEM BLOCK DIAGRAM

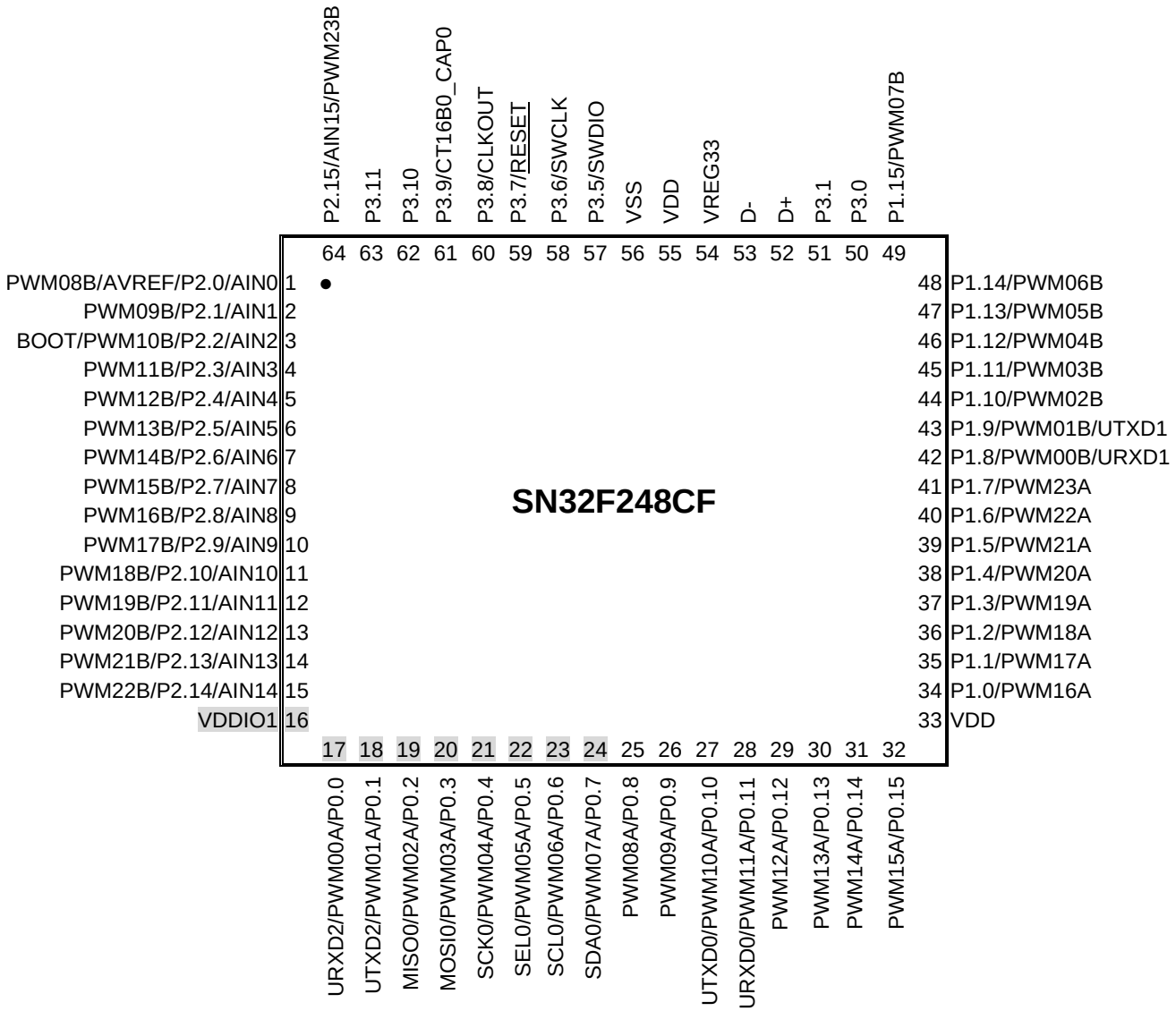


1.3 CLOCK GENERATION BLOCK DIAGRAM



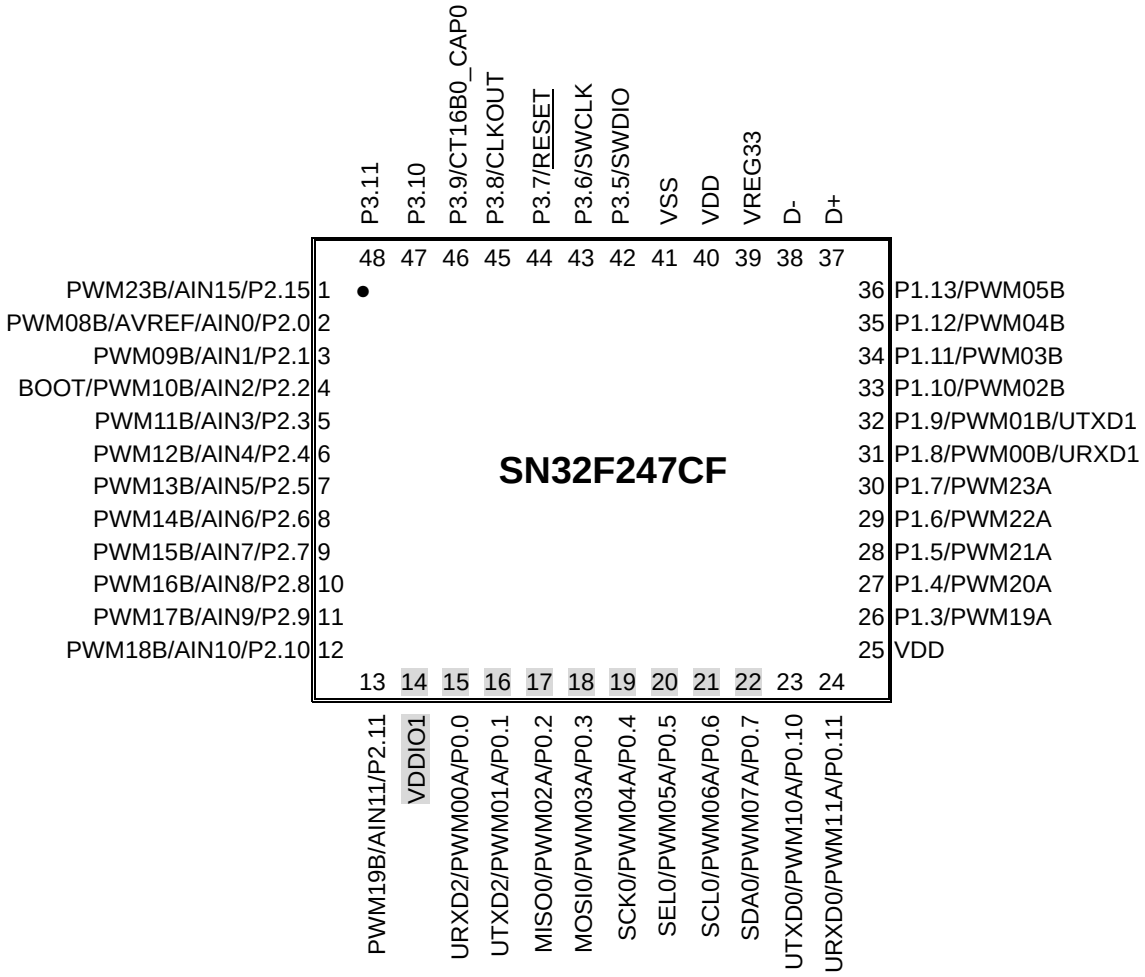
1.4 PIN ASSIGNMENT

SN32F248CF (LQFP 64 pins)



✸ ***Note: SONiX provide Boot loader to check the status of P2.2 (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.***

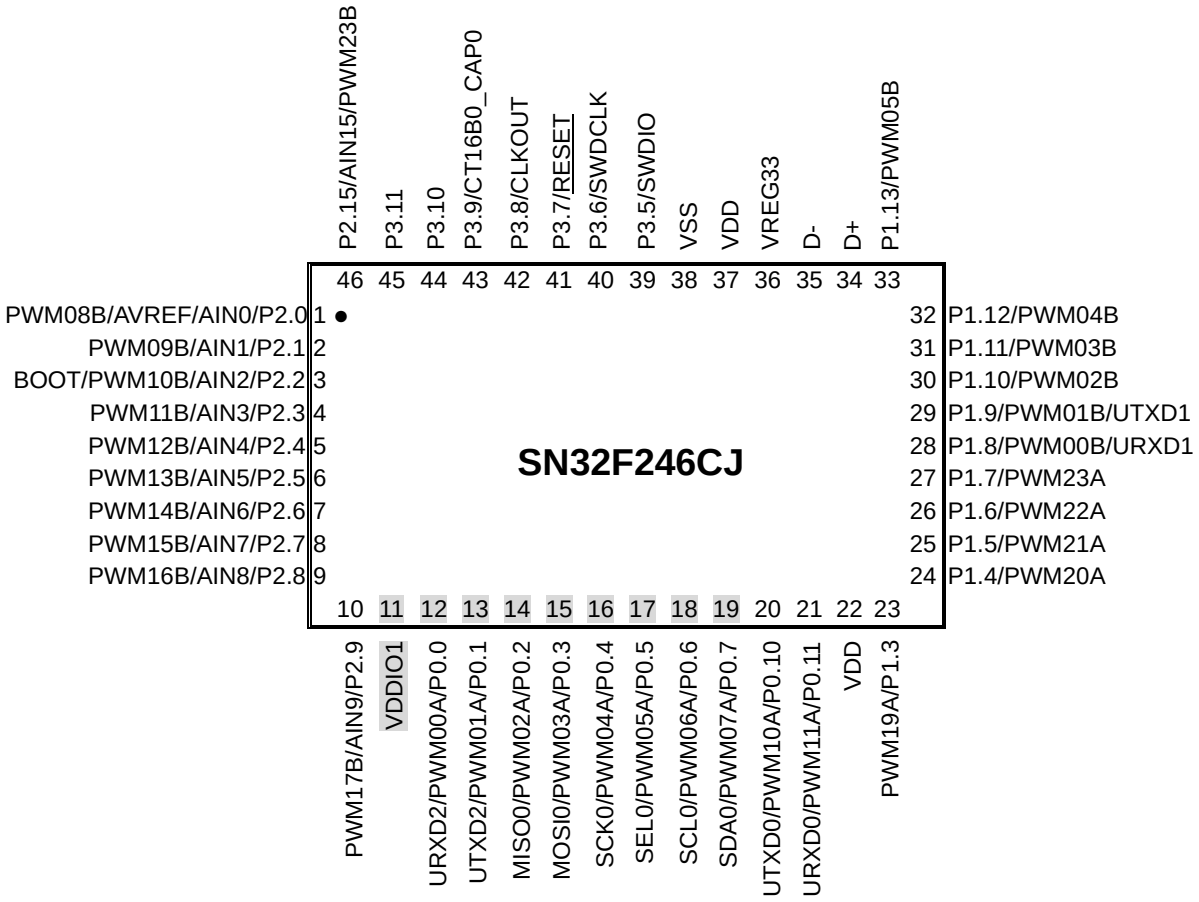
SN32F247CF (LQFP 48 pins)



✸ ***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 P2.2 (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.

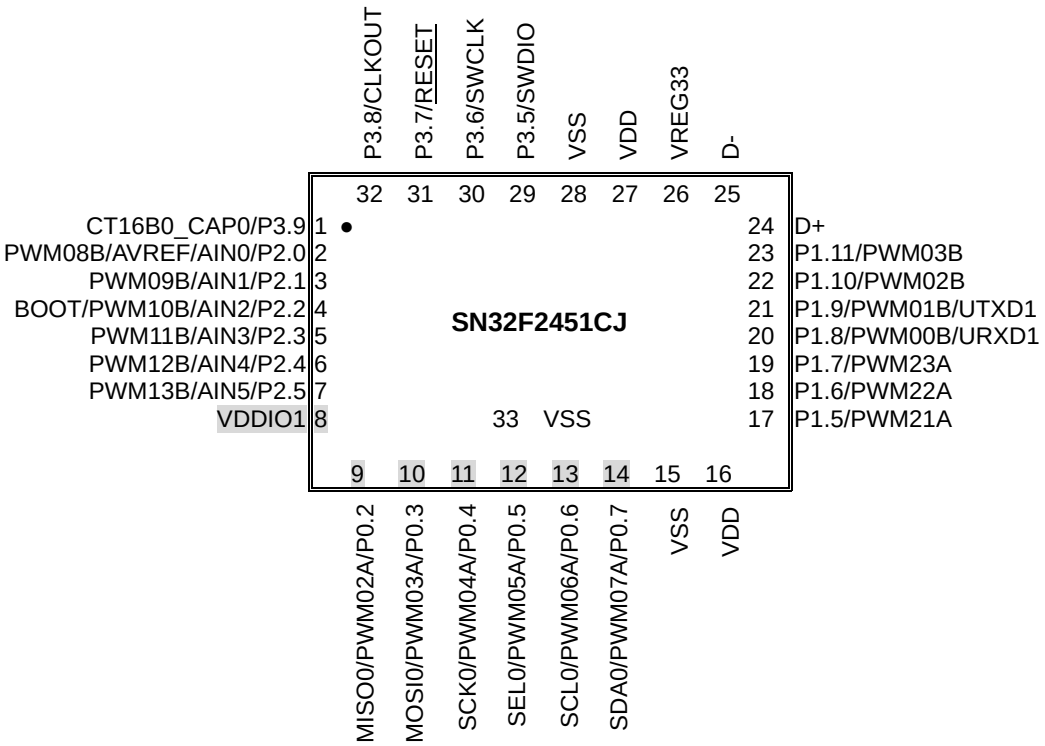
SN32F246CJ (QFN 46pins)



✷ ***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 P2.2 (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.

SN32F2451CJ (QFN 32pins 4x4)



✸ ***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 P2.2 (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

I/O	UART	SPI	I2C	CT16	ADC	Other
P0.0	URXD2			CT16B1_PWM00A		
P0.1	UTXD2	MOSI1		CT16B1_PWM01A		
P0.2		MISO0		CT16B1_PWM02A		
P0.3		MOSI0		CT16B1_PWM03A		
P0.4		SCK0		CT16B1_PWM04A		
P0.5		SEL0		CT16B1_PWM05A		
P0.6			SCL0	CT16B1_PWM06A		
P0.7			SDA0	CT16B1_PWM07A		
P0.8		SCK1		CT16B1_PWM08A		
P0.9		SEL1		CT16B1_PWM09A		
P0.10	UTXD0	MISO1		CT16B1_PWM10A		
P0.11	URXD0	MOSI1		CT16B1_PWM11A		
P0.12			SCL1	CT16B1_PWM12A		
P0.13			SDA1	CT16B1_PWM13A		
P0.14			SDA0	CT16B1_PWM14A		
P0.15			SCL0	CT16B1_PWM15A		
P1.0				CT16B1_PWM16A		
P1.1				CT16B1_PWM17A		
P1.2		SEL0		CT16B1_PWM18A		
P1.3		MOSI0 MOSI1		CT16B1_PWM19A		
P1.4		SCK0		CT16B1_PWM20A		
P1.5		MISO0		CT16B1_PWM21A		
P1.6		SEL0	SDA1	CT16B1_PWM22A		
P1.7		SCK0	SCL1	CT16B1_PWM23A		
P1.8	URXD1	MISO0	SCL1	CT16B1_PWM00B		
P1.9	UTXD1	MOSI0 MOSI1	SDA1	CT16B1_PWM01B		
P1.10		MOSI1		CT16B1_PWM02B		
P1.11		MISO1		CT16B1_PWM03B		
P1.12		SCK1		CT16B1_PWM04B		

I/O	UART	SPI	I2C	CT16	ADC	Other
P1.13		SEL1		CT16B1_PWM05B		
P1.14			SCL0	CT16B1_PWM06B		
P1.15			SDA0	CT16B1_PWM07B		
P2.0		SEL1		CT16B1_PWM08B	AIN0/ AVREF	
P2.1				CT16B1_PWM09B	AIN1	
P2.2				CT16B1_PWM10B	AIN2	BOOT
P2.3		MOSI1		CT16B1_PWM11B	AIN3	
P2.4		MISO1		CT16B1_PWM12B	AIN4	
P2.5		SCK1		CT16B1_PWM13B	AIN5	
P2.6				CT16B1_PWM14B	AIN6	
P2.7			SDA0	CT16B1_PWM15B	AIN7	
P2.8			SCL0	CT16B1_PWM16B	AIN8	
P2.9		MOSI0 MOSI1		CT16B1_PWM17B	AIN9	
P2.10		MISO0		CT16B1_PWM18B	AIN10	
P2.11		SCK0		CT16B1_PWM19B	AIN11	
P2.12		SEL0		CT16B1_PWM20B	AIN12	
P2.13			SCL1	CT16B1_PWM21B	AIN13	
P2.14			SDA1	CT16B1_PWM22B	AIN14	
P2.15		SEL1		CT16B1_PWM23B	AIN15	
P3.0						
P3.1						
P3.5						SWDIO
P3.6						SWCLK
P3.7						RESET
P3.8						CLKOUT
P3.9		MOSI1		CT16B0_CAP0		
P3.10		SCK1				
P3.11		MISO1				

1.6 PIN DESCRIPTIONS

PIN NAME	TYPE	DESCRIPTION
VDD, VSS	P	Power supply input pins for digital circuit.
VDDIO1	P	I/O driver power input pin for P0.0~P0.7.
VREG33	O	3.3V voltage output from USB 3.3V regulator.
D+/ PSCLK	I/O	**D+ —** USB differential signal line.
	I/O	**PSCLK —** PS/2's clock pin with internal 5K pull-up resistor.
D-/ PSDATA	I/O	**D- —**USB differential signal line.
	I/O	**PSDATA —**PS/2's data pin with internal 5K pull-up resistor.
P0.0/ CT16B1_PWM00A/ URXD2	I/O	**P0.0 —** General purpose digital input/output pin.
	O	**CT16B1_PWM00A —** PWM output 00 for CT16B1.
	I	**URXD2 —** Receiver input for UART2.
P0.1/ CT16B1_PWM01A/ UTXD2	I/O	**P0.1 —** General purpose digital input/output pin.
	O	**CT16B1_PWM01A —** PWM output 01 for CT16B1.
	O	**UTXD2 —** Transmitter output for UART2.
P0.2/ CT16B1_PWM02A/ MISO0	I/O	**P0.2 —** General purpose digital input/output pin.
	O	**CT16B1_PWM02A —** PWM output 02 for CT16B1.
	I/O	**MISO0 —** Master In Slave Out for SPI0.
P0.3/ CT16B1_PWM03A/ MOSI0	I/O	**P0.3 —** General purpose digital input/output pin.
	O	**CT16B1_PWM03A —** PWM output 03 for CT16B1.
	I/O	**MOSI0 —** Master Out Slave In for SPI0.
P0.4/ CT16B1_PWM04A/ SCK0	I/O	**P0.4 —** General purpose digital input/output pin.
	O	**CT16B1_PWM04A —** PWM output 04 for CT16B1.
	I/O	**SCK0 —** Serial clock for SPI0.
P0.5/ CT16B1_PWM05A/ SEL0	I/O	**P0.5 —** General purpose digital input/output pin.
	O	**CT16B1_PWM05A —** PWM output 05 for CT16B1.
	I	**SEL0 —** Slave Select for SPI0.
P0.6/ CT16B1_PWM06A/ SCL0	I/O	**P0.6 —** General purpose digital input/output pin.
	O	**CT16B1_PWM06A —** PWM output 06 for CT16B1.
	I/O	**SCL0 —** I2C0 clock input/output.
P0.7/ CT16B1_PWM07A/ SDA0	I/O	**P0.7 —** General purpose digital input/output pin.
	O	**CT16B1_PWM07A —** PWM output 07 for CT16B1.
	I/O	**SDA0 —** I2C0 data input/output.
P0.8/ CT16B1_PWM08A	I/O	**P0.8 —** General purpose digital input/output pin.
	O	**CT16B1_PWM08A —** PWM output 08 for CT16B1.
P0.9/ CT16B1_PWM09A	I/O	**P0.9 —** General purpose digital input/output pin.
	O	**CT16B1_PWM09A —** PWM output 09 for CT16B1.
P0.10/	I/O	**P0.10 —** General purpose digital input/output pin.

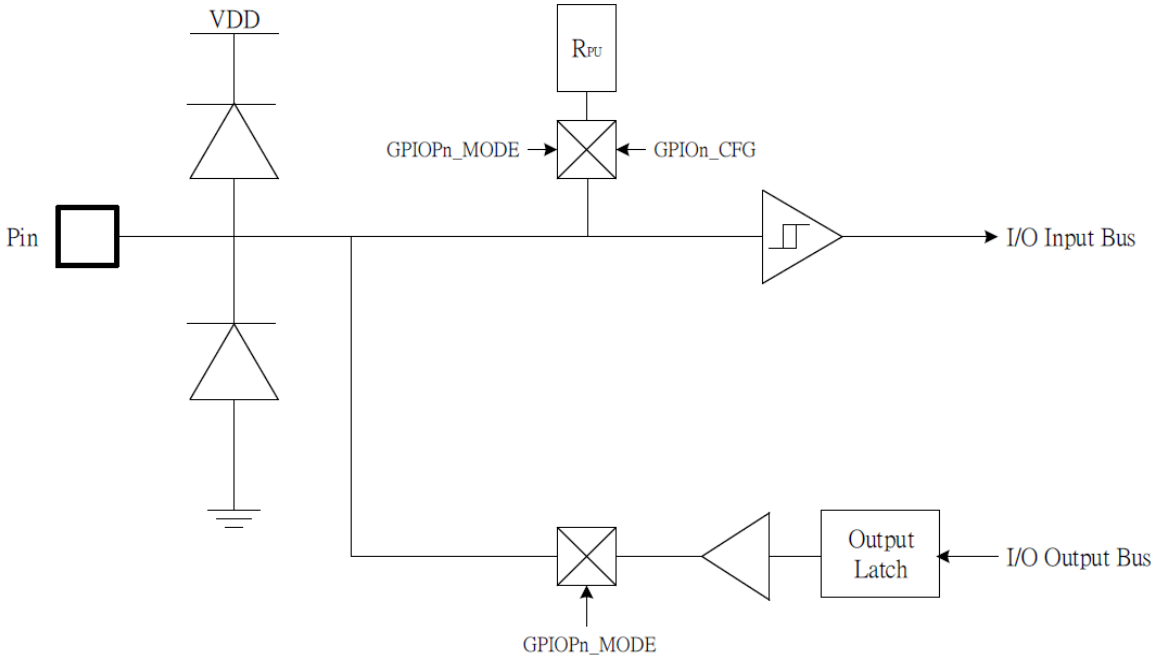
CT16B1_PWM10A/ UTXD0	O	**CT16B1_PWM10A —** PWM output 10 for CT16B1.
	O	**UTXD0 —** Transmitter output for UART0.
P0.11/ CT16B1_PWM11A/ URXD0	I/O	**P0.11 —** General purpose digital input/output pin.
	O	**CT16B1_PWM11A —** PWM output 11 for CT16B1.
	I	**URXD0 —** Receiver input for UART0.
P0.12/ CT16B1_PWM12A	I/O	**P0.12 —** General purpose digital input/output pin.
	O	**CT16B1_PWM12A —** PWM output 12 for CT16B1.
P0.13/ CT16B1_PWM13A	I/O	**P0.13 —** General purpose digital input/output pin.
	O	**CT16B1_PWM13A —** PWM output 13 for CT16B1.
P0.14/ CT16B1_PWM14A	I/O	**P0.14 —** General purpose digital input/output pin.
	O	**CT16B1_PWM14A —** PWM output 14 for CT16B1.
P0.15/ CT16B1_PWM15A	I/O	**P0.15 —** General purpose digital input/output pin.
	O	**CT16B1_PWM15A —** PWM output 15 for CT16B1.
P1.0/ CT16B1_PWM16A	I/O	**P1.0 —** General purpose digital input/output pin.
	O	**CT16B1_PWM16A —** PWM output 16 for CT16B1.
P1.1/ CT16B1_PWM17A	I/O	**P1.1 —** General purpose digital input/output pin.
	O	**CT16B1_PWM17A —** PWM output 17 for CT16B1.
P1.2/ CT16B1_PWM18A	I/O	**P1.2 —** General purpose digital input/output pin.
	O	**CT16B1_PWM18A —** PWM output 18 for CT16B1.
P1.3/ CT16B1_PWM19A	I/O	**P1.3 —** General purpose digital input/output pin.
	O	**CT16B1_PWM19A —** PWM output 19 for CT16B1.
P1.4/ CT16B1_PWM20A	I/O	**P1.4 —** General purpose digital input/output pin.
	O	**CT16B1_PWM20A —** PWM output 20 for CT16B1.
P1.5/ CT16B1_PWM21A	I/O	**P1.5 —** General purpose digital input/output pin.
	O	**CT16B1_PWM21A —** PWM output 21 for CT16B1.
P1.6/ CT16B1_PWM22A	I/O	**P1.6 —** General purpose digital input/output pin.
	O	**CT16B1_PWM22A —** PWM output 22 for CT16B1.
P1.7/ CT16B1_PWM23A	I/O	**P1.7 —** General purpose digital input/output pin.
	O	**CT16B1_PWM23A —** PWM output 23 for CT16B1.
P1.8/ CT16B1_PWM00B/ URXD1	I/O	**P1.8 —** General purpose digital input/output pin.
	O	**CT16B1_PWM00B —** PWM output 00 for CT16B1.
	I	**URXD1 —** Receiver input for UART1.
P1.9/ CT16B1_PWM01B/ UTXD1	I/O	**P1.9 —** General purpose digital input/output pin.
	O	**CT16B1_PWM01B —** PWM output 01 for CT16B1.
	O	**UTXD1 —** Transmitter output for UART1.
P1.10/ CT16B1_PWM02B	I/O	**P1.10 —** General purpose digital input/output pin.
	O	**CT16B1_PWM02B —** PWM output 02 for CT16B1.
P1.11/ CT16B1_PWM03B	I/O	**P1.11 —** General purpose digital input/output pin.
	O	**CT16B1_PWM03B —** PWM output 03 for CT16B1.
P1.12/	I/O	**P1.12 —** General purpose digital input/output pin.

CT16B1_PWM04B	O	**CT16B1_PWM04B —** PWM output 04 for CT16B1.
P1.13/ CT16B1_PWM05B	I/O	**P1.13 —** General purpose digital input/output pin.
	O	**CT16B1_PWM05B —** PWM output 05 for CT16B1.
P1.14/ CT16B1_PWM06B	I/O	**P1.14 —** General purpose digital input/output pin.
	O	**CT16B1_PWM06B —** PWM output 06 for CT16B1.
P1.15/ CT16B1_PWM07B	I/O	**P1.15 —** General purpose digital input/output pin.
	O	**CT16B1_PWM07B —** PWM output 07 for CT16B1.
P2.0/ CT16B1_PWM08B/ AIN0	I/O	**P2.0 —** General purpose digital input/output pin.
	O	**CT16B1_PWM08B —** PWM output 08 for CT16B1.
	I	**AIN0 —** ADC channel input 0.
P2.1/ CT16B1_PWM09B/ AIN1	I/O	**P2.1 —** General purpose digital input/output pin.
	O	**CT16B1_PWM09B —** PWM output 09 for CT16B1.
	I	**AIN1 —** ADC channel input 1.
P2.2/ CT16B1_PWM10B/ AIN2/ BOOT	I/O	**P2.2 —** General purpose digital input/output pin.
	O	**CT16B1_PWM10B —** PWM output 10 for CT16B1.
	I	**AIN2 —** ADC channel input 2.
	I	**BOOT —** Boot loader check pin. 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.
P2.3/ CT16B1_PWM11B/ AIN3	I/O	**P2.3 —** General purpose digital input/output pin.
	O	**CT16B1_PWM11B —** PWM output 11 for CT16B1.
	I	**AIN3 —** ADC channel input 3.
P2.4/ CT16B1_PWM12B/ AIN4	I/O	**P2.4 —** General purpose digital input/output pin.
	O	**CT16B1_PWM12B —** PWM output 12 for CT16B1.
	I	**AIN4 —** ADC channel input 4.
P2.5/ CT16B1_PWM13B/ AIN5	I/O	**P2.5 —** General purpose digital input/output pin.
	O	**CT16B1_PWM13B —** PWM output 13 for CT16B1.
	I	**AIN5 —** ADC channel input 5.
P2.6/ CT16B1_PWM14B/ AIN6	I/O	**P2.6 —** General purpose digital input/output pin.
	O	**CT16B1_PWM14B —** PWM output 14 for CT16B1.
	I	**AIN6 —** ADC channel input 6.
P2.7/ CT16B1_PWM15B/ AIN7	I/O	**P2.7 —** General purpose digital input/output pin.
	O	**CT16B1_PWM15B —** PWM output 15 for CT16B1.
	I	**AIN7 —** ADC channel input 7.
P2.8/ CT16B1_PWM16B/ AIN8	I/O	**P2.8 —** General purpose digital input/output pin.
	O	**CT16B1_PWM16B —** PWM output 16 for CT16B1.
	I	**AIN8 —** ADC channel input 8.
P2.9/ CT16B1_PWM17B/ AIN9	I/O	**P2.9 —** General purpose digital input/output pin.
	O	**CT16B1_PWM17B —** PWM output 17 for CT16B1.
	I	**AIN9 —** ADC channel input 9.
P2.10/ CT16B1_PWM18B/	I/O	**P2.10 —** General purpose digital input/output pin.
	O	**CT16B1_PWM18B —** PWM output 18 for CT16B1.

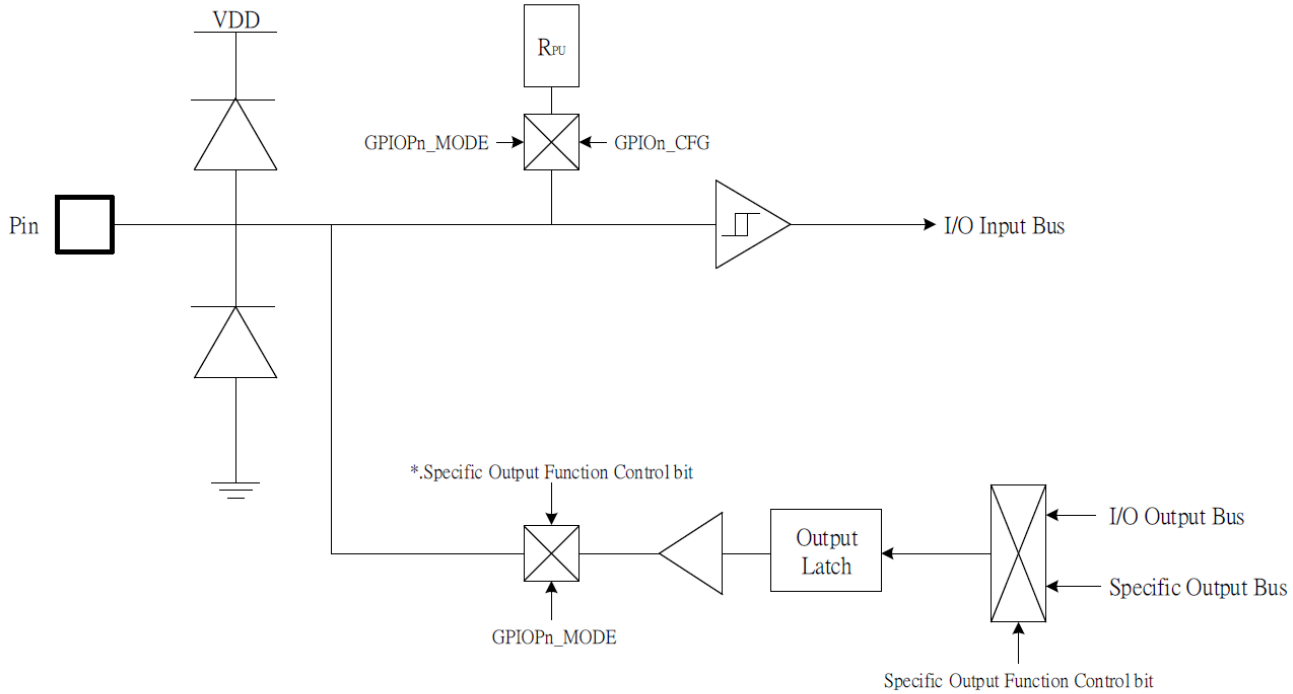
AIN10	I	**AIN10 —** ADC channel input 10.
P2.11/ CT16B1_PWM19B/ AIN11	I/O	**P2.11 —** General purpose digital input/output pin.
	O	**CT16B1_PWM19B —** PWM output 19 for CT16B1.
	I	**AIN11 —** ADC channel input 11.
P2.12/ CT16B1_PWM20B/ AIN12	I/O	**P2.12 —** General purpose digital input/output pin.
	O	**CT16B1_PWM20B —** PWM output 20 for CT16B1.
	I	**AIN12 —** ADC channel input 12.
P2.13/ CT16B1_PWM21B/ AIN13	I/O	**P2.13 —** General purpose digital input/output pin.
	O	**CT16B1_PWM21B —** PWM output 21 for CT16B1.
	I	**AIN13 —** ADC channel input 13.
P2.14/ CT16B1_PWM22B/ AIN14	I/O	**P2.14 —** General purpose digital input/output pin.
	O	**CT16B1_PWM22B —** PWM output 22 for CT16B1.
	I	**AIN14 —** ADC channel input 14.
P2.15/ CT16B1_PWM23B/ AIN15	I/O	**P2.15 —** General purpose digital input/output pin.
	O	**CT16B1_PWM23B —** PWM output 23 for CT16B1.
	I	**AIN15 —** ADC channel input 15.
P3.0	I/O	**P3.0 —** General purpose digital input/output pin.
P3.1	I/O	**P3.1 —** General purpose digital input/output pin.
P3.5/ SWDIO	I/O	**P3.5 —** General purpose digital input/output pin.
	I/O	**SWDIO —** Serial wire debug input/output.
P3.6/ SWCLK	I/O	**P3.6 —** General purpose digital input/output pin.
	I/O	**SWCLK —** Serial wire clock.
P3.7/ RESET	I/O	**P3.7 —** General purpose digital input/output pin.
	I	**RESET —** external Reset input.
P3.8/ CLKOUT	I/O	**P3.8 —** General purpose digital input/output pin.
	O	**CLKOUT —** Clockout pin.
P3.9/ CT16B0_CAP0	I/O	**P3.9 —** General purpose digital input/output pin.
	I	**CT16B0_CAP0 —** Capture input 0 for CT16B0.
P3.10	I/O	**P3.10 —** General purpose digital input/output pin.
P3.11	I/O	**P3.11 —** General purpose digital input/output pin.

1.7 PIN CIRCUIT DIAGRAMS

- **Normal Bi-direction I/O Pin.**

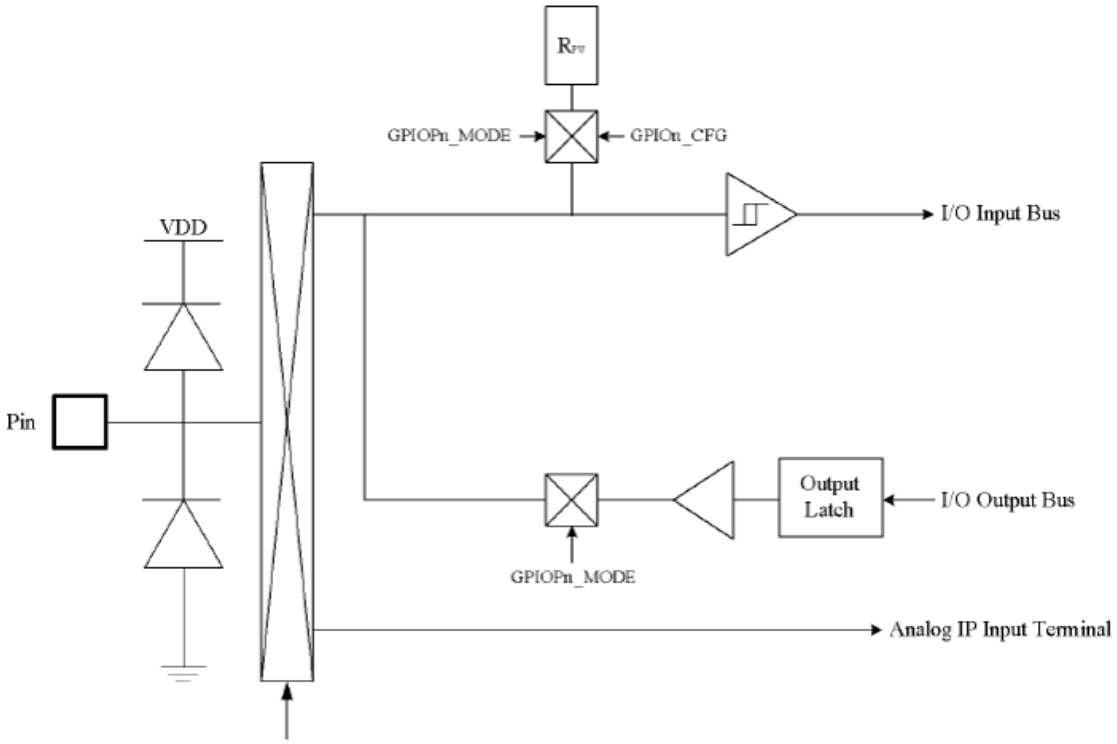


- **Bi-direction I/O Pin Shared with Specific Digital Output Function, e.g. SPI, I2C…**



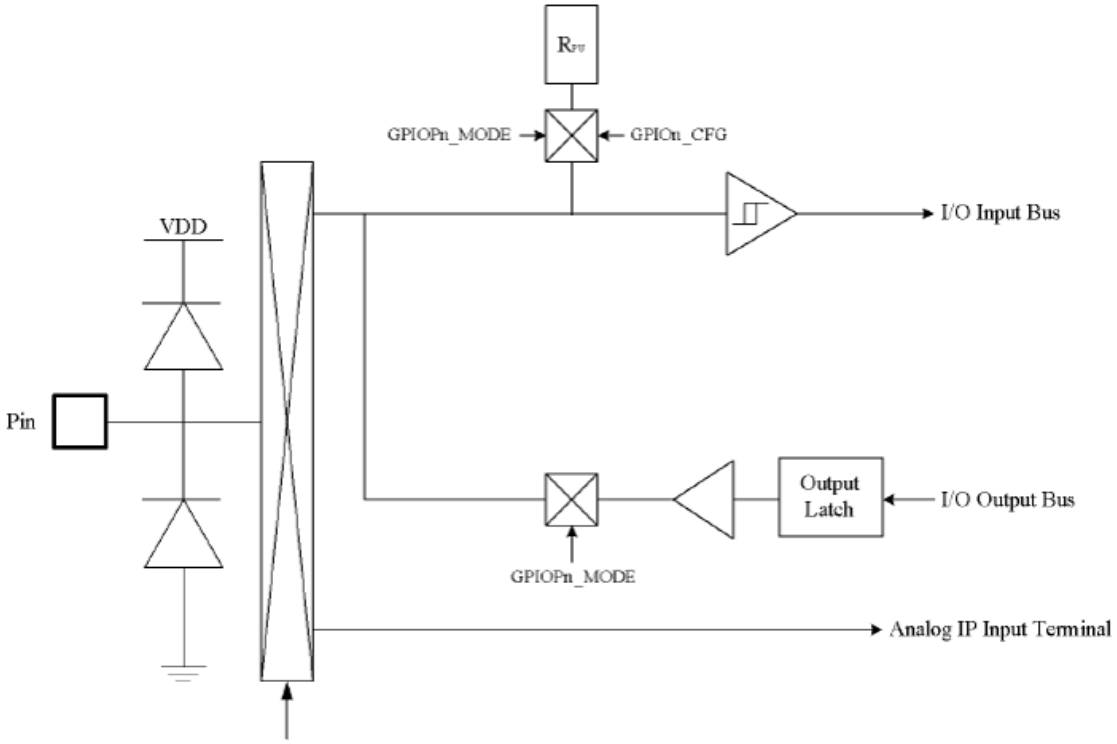
*. Some specific functions switch I/O direction directly, not through GPIOn_MODE register.

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



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

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



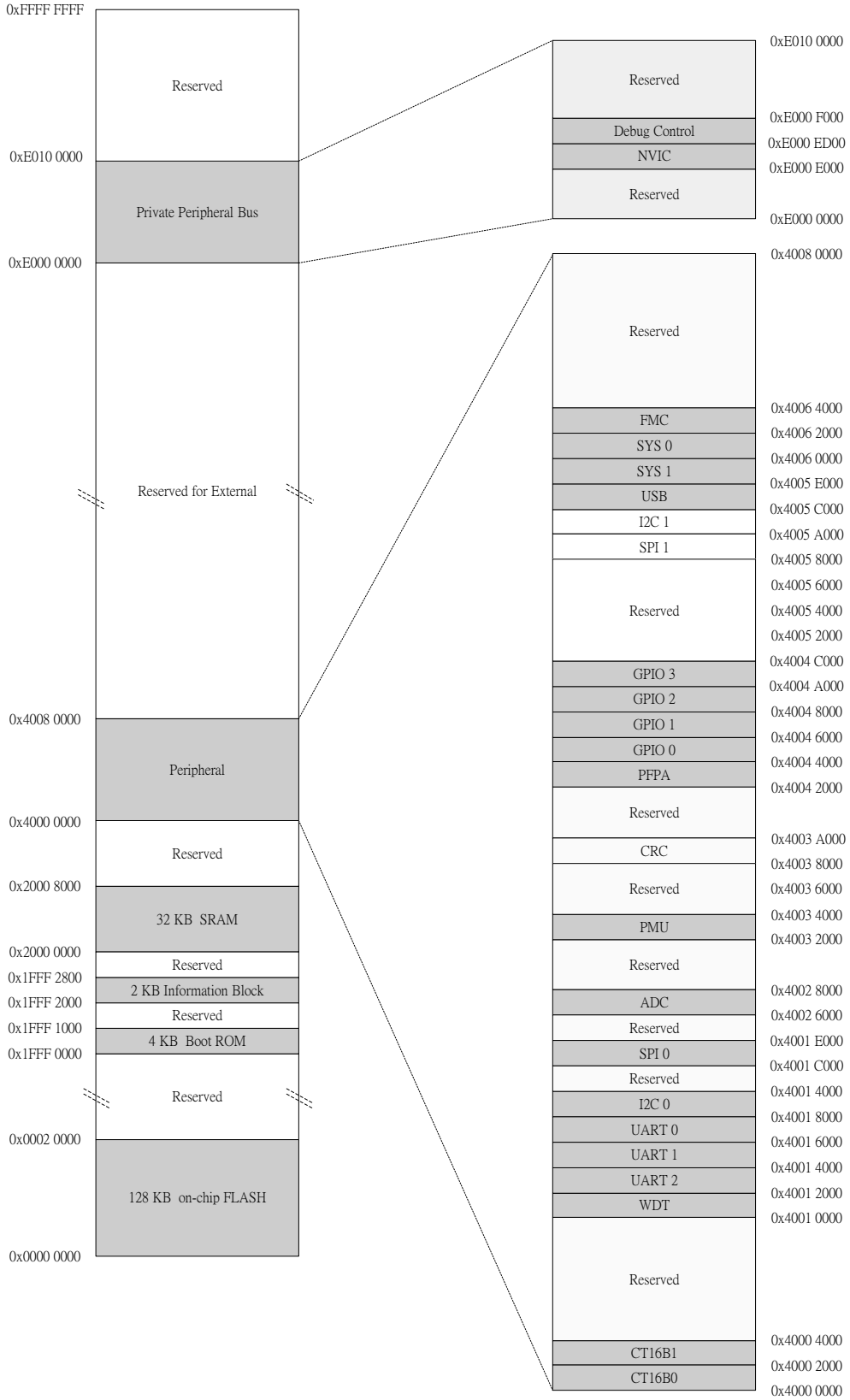
***. Some specific functions switch I/O direction directly, not through GPIOn_MODE register.**

1.8 PIN CHARACTERISTICS

Port	Direction	ADC	Open-Drain	Pull-up Resistor (75KΩ@3.3V)	Pull-down Resistor (75KΩ@3.3V)	Sink Current 420mA @VSS+1.5V
P0.0	I/O			V		
P0.1	I/O			V		
P0.2	I/O			V		
P0.3	I/O			V		
P0.4	I/O			V		
P0.5	I/O			V		
P0.6	I/O		V	V		
P0.7	I/O		V	V		
P0.8	I/O			V		V
P0.9	I/O			V		V
P0.10	I/O			V		V
P0.11	I/O			V		V
P0.12	I/O		V	V		V
P0.13	I/O		V	V		V
P0.14	I/O		V	V		V
P0.15	I/O		V	V		V
P1.0	I/O			V		V
P1.1	I/O			V		V
P1.2	I/O			V		V
P1.3	I/O			V		V
P1.4	I/O			V		V
P1.5	I/O			V		V
P1.6	I/O		V	V		V
P1.7	I/O		V	V		V
P1.8	I/O		V	V		V
P1.9	I/O		V	V		V
P1.10	I/O			V		V
P1.11	I/O			V		V
P1.12	I/O			V		
P1.13	I/O			V		
P1.14	I/O		V	V		
P1.15	I/O		V	V		
P2.0	I/O	V		V		
P2.1	I/O	V		V		
P2.2	I/O	V		V		
P2.3	I/O	V		V		
P2.4	I/O	V		V		
P2.5	I/O	V		V		
P2.6	I/O	V		V		
P2.7	I/O	V	V	V		
P2.8	I/O	V	V	V		
P2.9	I/O	V		V		
P2.10	I/O	V		V		
P2.11	I/O	V		V		
P2.12	I/O	V		V		
P2.13	I/O	V	V	V		
P2.14	I/O	V	V	V		
P2.15	I/O	V		V		
P3.0	I/O			V		
P3.1	I/O			V		
P3.5	I/O			V		
P3.6	I/O			V	V	
P3.7	I/O			V		
P3.8	I/O			V		
P3.9	I/O			V		
P3.10	I/O			V		
P3.11	I/O			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.

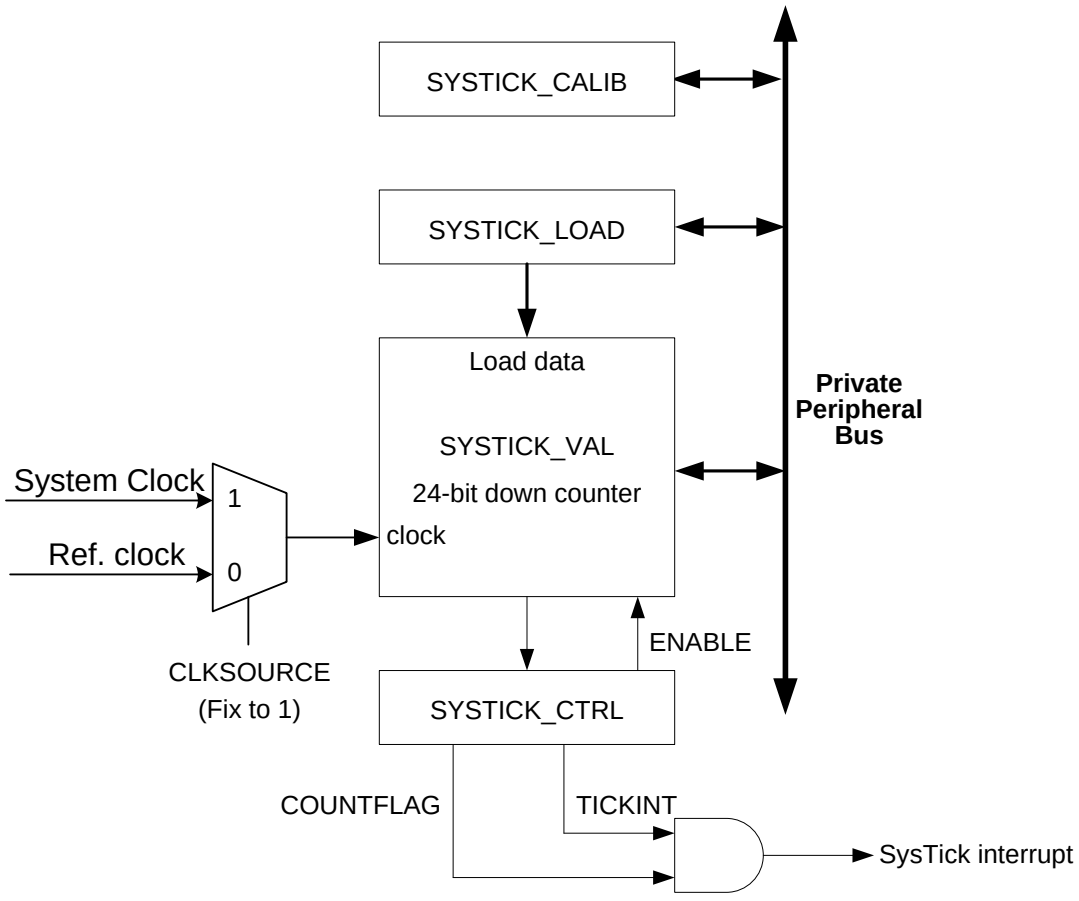
Refer to the *Cortex-M0 User Guide* for details.

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 48 MHz.

RELOAD = (system tick clock frequency × 10 ms) −1 = (48 MHz × 10 ms) −1
= 0x000752FF.

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	0

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	0

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	0x7A11F

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	SVCCalll		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/USBIRQ	USB	0x0000 0044
18	Settable	IRQ2/		0x0000 0048
19	Settable	IRQ3/		0x0000 004C
20	Settable	IRQ4/		0x0000 0050
21	Settable	IRQ5/		0x0000 0054
22	Settable	IRQ6/SPI0IRQ	SPI0	0x0000 0058
23	Settable	IRQ7/SPI1IRQ	SPI1	0x0000 005C
24	Settable	IRQ8/		0x0000 0060
25	Settable	IRQ9/		0x0000 0064
26	Settable	IRQ10/I2C0IRQ	I2C0	0x0000 0068
27	Settable	IRQ11/I2C1IRQ	I2C1	0x0000 006C
28	Settable	IRQ12/UART0IRQ	UART0	0x0000 0070
29	Settable	IRQ13/UART1IRQ	UART1	0x0000 0074
30	Settable	IRQ14/UART2IRQ	UART2	0x0000 0078

31	Settable	IRQ15/CT16B0IRQ	CT16B0	0x0000 007C
32	Settable	IRQ16/CT16B1IRQ	CT16B1	0x0000 0080
33	Settable	IRQ17/		0x0000 0084
34	Settable	IRQ18/		0x0000 0088
35	Settable	IRQ19/		0x0000 008C
36	Settable	IRQ20/		0x0000 0090
37	Settable	IRQ21/		0x0000 0094
38	Settable	IRQ22/		0x0000 0098
39	Settable	IRQ23/		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/		0x0000 00AC
44	Settable	IRQ28/P3IRQ	GPIO interrupt status of port 3	0x0000 00B0
45	Settable	IRQ29/P2IRQ	GPIO interrupt status of port 2	0x0000 00B4
46	Settable	IRQ30/P1IRQ	GPIO interrupt status of port 1	0x0000 00B8
47	Settable	IRQ31/P0IRQ	GPIO interrupt status of port 0	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.	R/W	0

		Read→ 0: Interrupt disabled 1: Interrupt enabled.		

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 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.	R/W	0

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 Others: CS2	R/W	FFFF
15:3	Reserved		R	All 1
2	BOOTPINEN	Boot Pin enable 0: Disable 1: Enable (default)	R/W	1
1	EXTRSTHWDIS	External reset HW disable bit 0: Enable (HW enables, FW can NOT control with SYS0_EXRSTCTRL) 1: Disable (default) (FW can control with SYS0_EXRSTCTRL)	R/W	1
0	BLEN	Boot loader enable 0: Disable 1: Enable (default)	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. Users can use the unique number to pair in RF application, or use as USB string serial number.

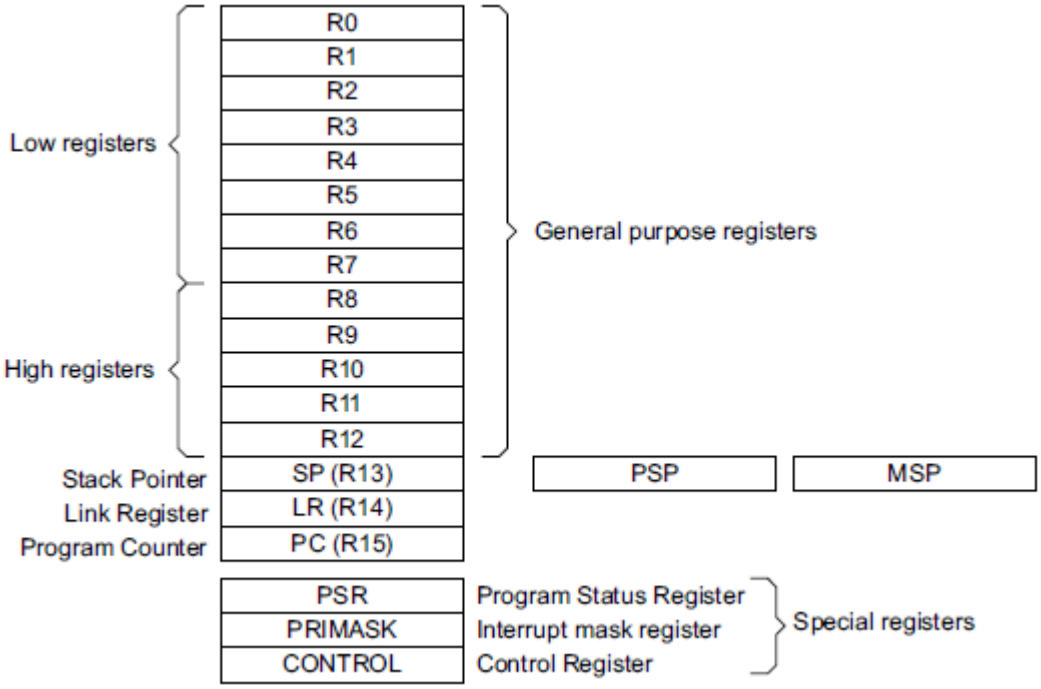
Address: 0x1FFF 2508

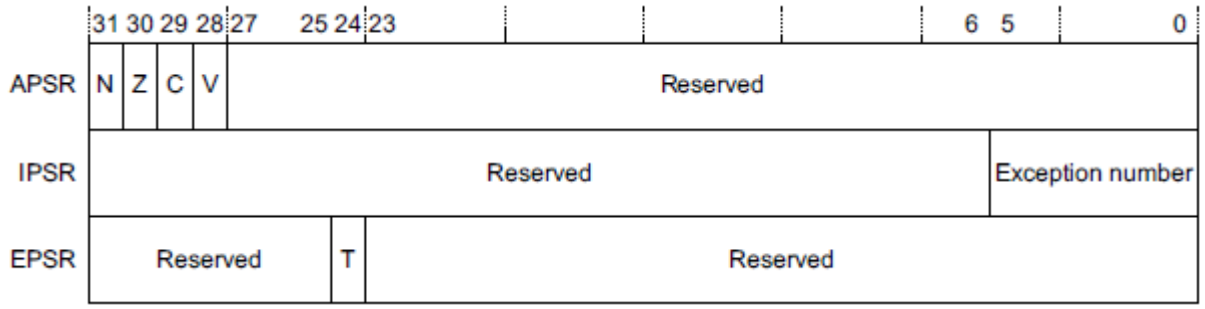
Bit	Name	Description	Attribute	Reset
31:0	L4BYTE[31:0]	Lower 4 bytes of Unique number	R	By Die

Address: 0x1FFF 2510

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. 
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.

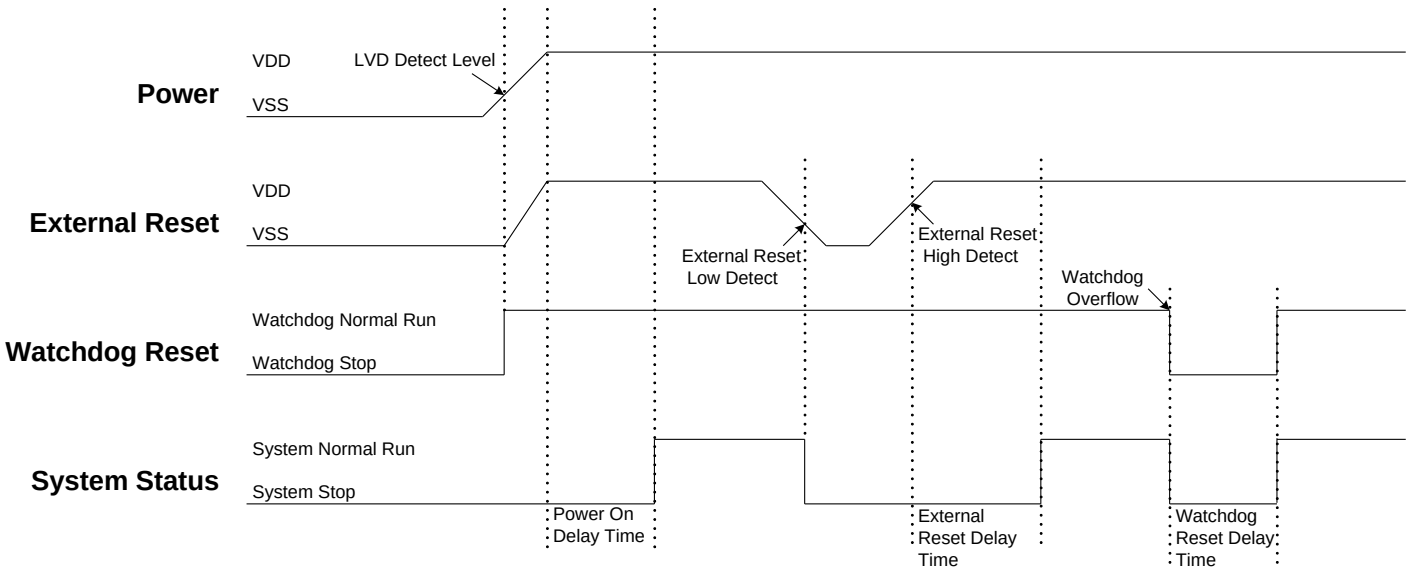
3 SYSTEM CONTROL

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 Boot loader if BLEN bit =1, or from 0x0 if BLEN bit =0.

Watchdog timer application note is as following.

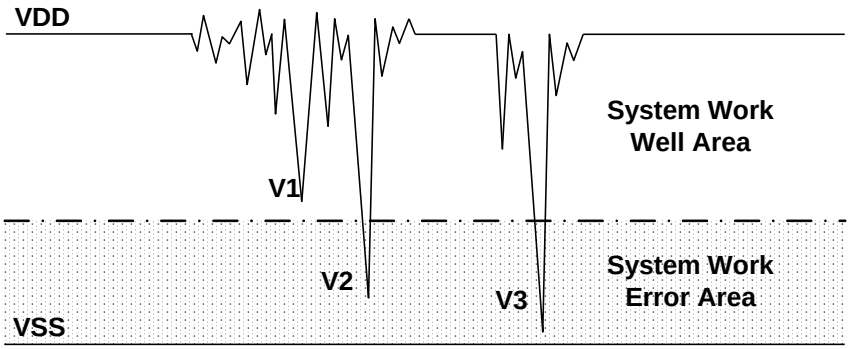
- Before clearing watchdog timer, check I/O status and check RAM contents can improve system error.
- Don't clear watchdog timer in interrupt vector and interrupt service routine. That can improve main routine fail.
- Clearing watchdog timer program is only at one part of the program. This way is the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

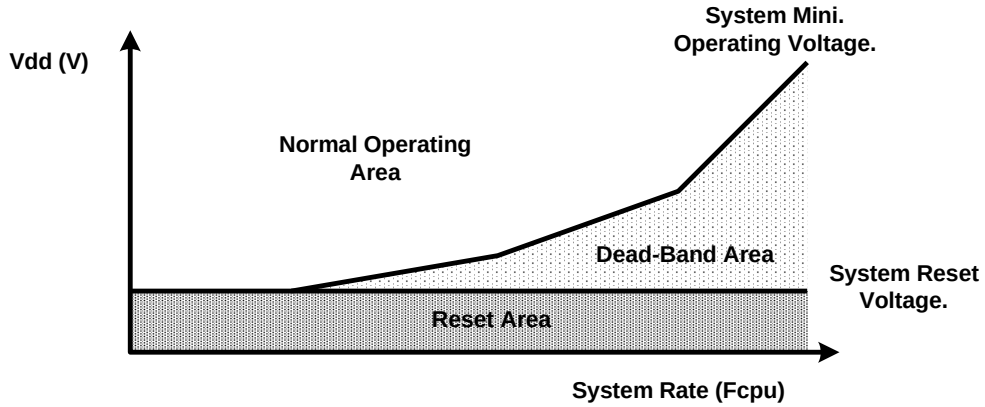
DC application:
The power source of DC application is usually using battery. When low battery condition and MCU drive any loading, the power drops and keeps in dead-band. Under the situation, the power won't drop deeper and not touch the system reset voltage. That makes the system under dead-band.

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.

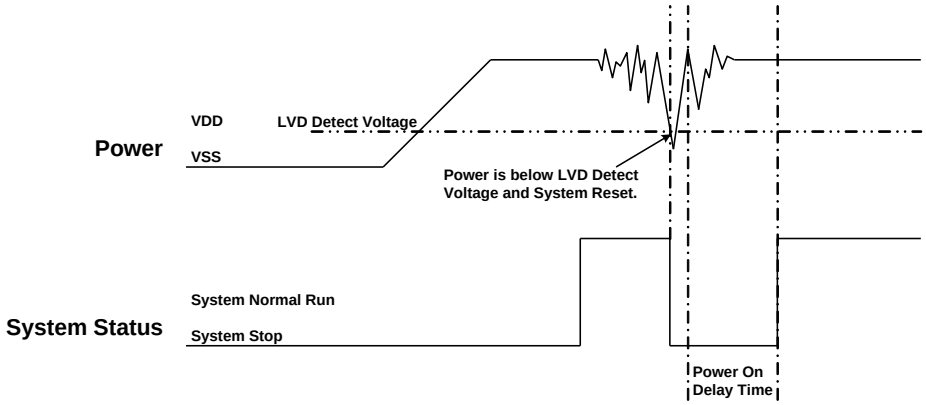
3.1.3.3 BROWN-OUT RESET IMPROVEMENT

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

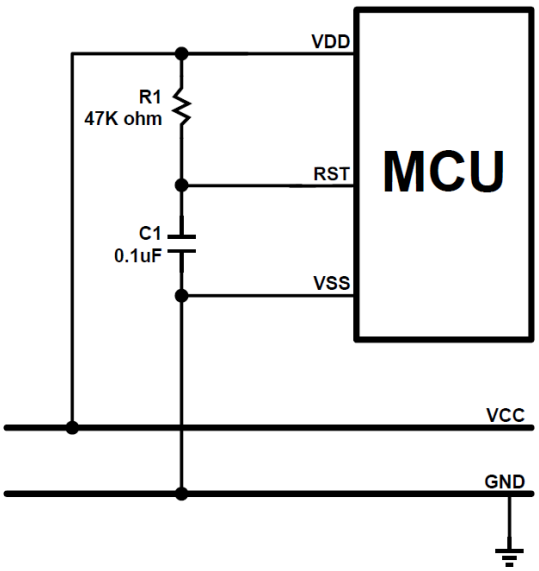
External reset function is controlled by External RESET pin control (SYS0_EXRSTCTRL) register. Default value is 1, which means external reset function is enabled. External reset pin is Schmitt Trigger structure and low level active. The system is running when reset pin is high level voltage input. The reset pin receives the low voltage and the system is reset. The external reset operation actives 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.

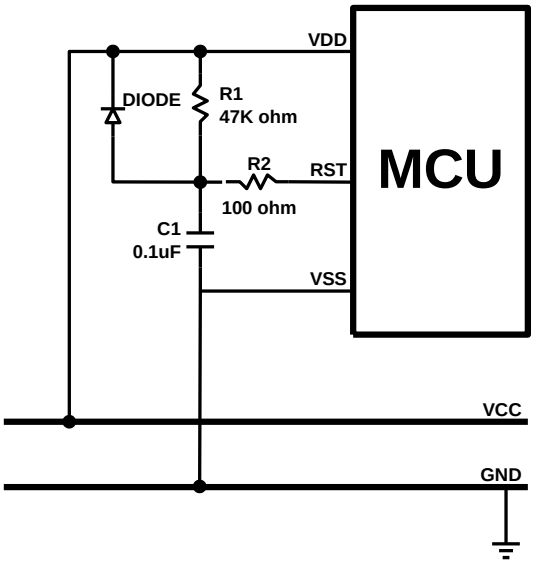
3.1.4.1 SIMPLY RC RESET CIRCUIT



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

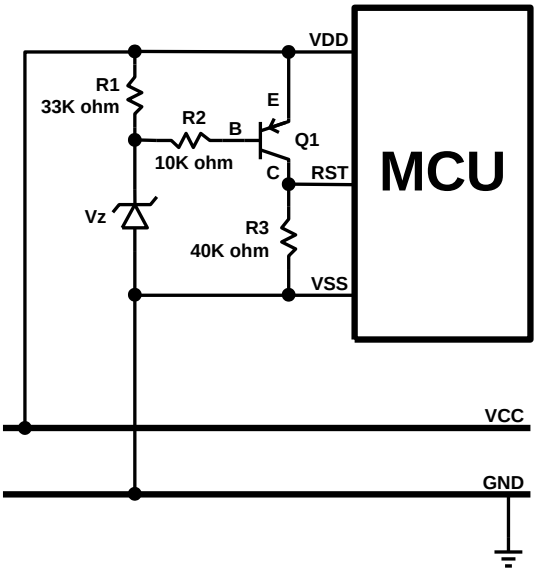


This is the better reset circuit. The R1 and C1 circuit operation is like the simply reset circuit to make a power on signal.

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.

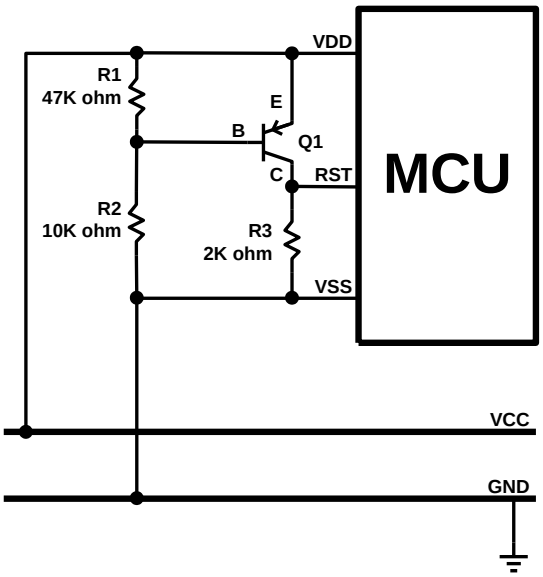
✸ ***Note: The R2 100 ohm resistor of "Simply reset circuit" and "Diode & RC reset circuit" is necessary to limit any current flowing into reset pin from external capacitor C in the event of reset pin breakdown due to Electrostatic Discharge (ESD) or Electrical Over-stress (EOS).***

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 "Vz + 0.7V", the C terminal of the PNP transistor outputs high voltage and MCU operates normally. When VDD is below "Vz + 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

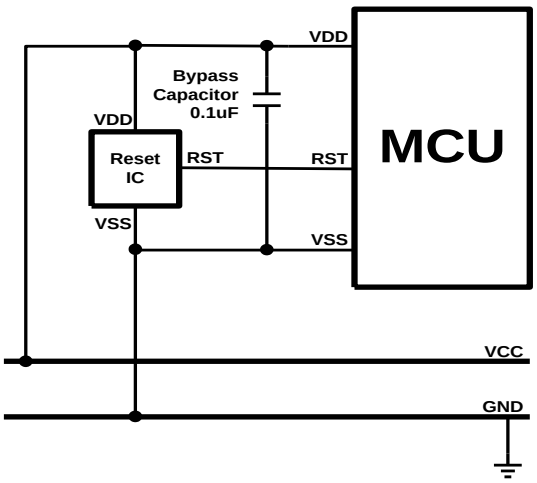


The voltage bias reset circuit is a low cost voltage detector and can **improve brown out reset condition completely**.

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 x (R1 + R2) / R1”, the C terminal of the PNP transistor outputs high voltage and MCU operates normally. When VDD is below “0.7V x (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 protects 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):
- 12 MHz internal high speed RC (IHRC)
- 32 KHz internal low speed RC (ILRC)
- PLL clock

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 12MHz RC type. The accuracy is ±0.2% 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)

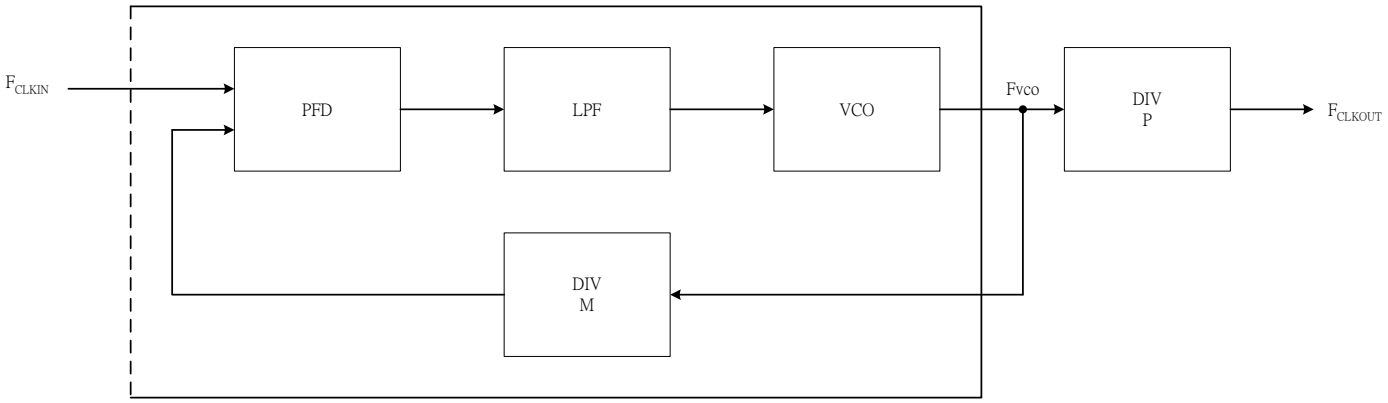
The system low clock source is the internal low-speed oscillator built in the micro-controller. The low-speed oscillator uses RC type oscillator circuit. The frequency is affected by the voltage and temperature of the system. In common condition, the frequency of the RC oscillator is about 32 KHz.

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

3.2.2 PLL

SONiX 32-bit MCU uses the PLL to create the clocks for the core and peripherals. The input frequency range is 10MHz 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. 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 µs.



3.2.2.1 PLL Frequency selection

The PLL frequency equations:

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

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

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); 96MHz.
- F_{CLKOUT}: Frequency of PLL output.
- P: System PLL post divider ratio, controlled by PSEL 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 and P, it is recommended to follow these constraints:

1. 10MHz ≤ F_{CLKIN} ≤ 25MHz
2. 96MHz
3. M = 4, 6, 8, 10, or 12
4. P = 2, or 4 (duty 50% +/- 2.5%)
5. F_{CLKOUT} = 30MHz, 50MHz, 60MHz, 24MHz, 36MHz, 48MHz, 72MHz, 64MHz with jitter < ±500 ps

3.2.3 SYSTEM CLOCK (SYSCLK) SELECTION

After a system reset, the IHRC is selected as system clock. When a clock source is used directly 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). 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.

3.2.4 CLOCK-OUT CAPABITITY

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 3 clock signals can be selected as clock output:
1. HCLK
2. IHRC
3. ILRC
4. PLL clock output

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: IHRCEN bit can NOT be cleared if the IHRC is selected as system clock or is selected to become the system clock.***

Bit	Name	Description	Attribute	Reset
31:1	Reserved		R	0
0	IHRCEN	Internal high-speed clock enable. Note: This bit can NOT be cleared if the IHRC is selected as system clock or is selected to become the system clock. 0: Disable internal 12 MHz RC oscillator. 1: Enable internal 12 MHz RC oscillator.	R/W	1

3.3.2 PLL control register (SYS0_PLLCTRL)

Address Offset: 0x04

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15	PLLEN	PLL enable Note: This bit can NOT be cleared if the PLL clock is selected as system clock or is selected to become the system clock. 0: Disable 1: Enable	R/W	0
14:6	Reserved		R	0
5	PSEL	Post divider value. 0: P = 2 1: P = 4	R/W	0
4:3	Reserved		R	0
2:0	MSEL[2:0]	Feedback divider value. 000: M = 4 001: M = 6 010: M = 8 011: M = 10 100: M = 12 Other: Reserved	R/W	0

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
5:1	Reserved		R	0
0	IHRCRDY	IHRC ready flag. After the IHRCEN bit is cleared, IHRCRDY is cleared by HW after 6 IHRC clock cycles. 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	SYSCLKST	000: IHRC is used as system clock 001: ILRC is used as system clock 100: PLL is used as system clock Other: Reserved	R	0
3	Reserved		R	0
2:0	SYSCLKSEL	System clock switch Set and cleared by SW. 000: IHRC (12MHz) 001: ILRC 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
31:3	Reserved		R	0
2:0	AHBPRE[2:0]	AHB clock source prescale value 000: SYSCLK / 1 001: SYSCLK / 2 010: SYSCLK / 4 011: SYSCLK / 8 100: SYSCLK / 16 101: SYSCLK / 32 110: SYSCLK / 64 111: SYSCLK / 128 Other: Reserved	R/W	000b

3.3.6 System Reset Status register (SYS0_RSTST)

Address Offset: 0x14

This register contains the reset source.

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 RESET 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 four 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	LVDEN	LVD enable. 0: Disable. 1: Enable.	R/W	0
14	LVDRSTEN	LVD Reset enable. 0: Disable → Flag. 1: Enable → Reset.	R/W	0
13:6	Reserved		R	0
5:4	LVDINTLVL[1:0]	LVD interrupt level. 00: 2.20V. 01: 2.70V. 10: 3.60V. Other: Reserved.	R/W	00b
3:2	Reserved		R	0
1:0	LVDRSTLVL[1:0]	LVD reset level. 00: 2.20V. 01: 2.70V. 10: 3.60V. Other: Reserved.	R/W	00b

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 RESET pin. (P3.7 acts as RESET pin) 1: Disable. (P3.7 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. (P3.5 acts as SWDIO pin, P3.6 acts as SWCLK pin) 1: Disable. (P3.5 and P3.6 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 User ROM or SRAM.

Bit	Name	Description	Attribute	Reset
31:16	IVTMKEY[15:0]	IVTM register key. Read as 0. Behavior of writing to this register is ignored unless writing 0xA5A5 to IVTMKEY at the same time.	W	0
15:3	Reserved		R	0
2:0	IVTM[2:0]	Interrupt table mapping selection. 000: Map to Boot ROM. 001: Map to User ROM 1. 010: Map to SRAM. 011: Map to User ROM 2. Other: Reserved.	R/W	By BLEN in code option

3.3.11 Noise Detect Control register (SYS0_NDTCTRL)

Address Offset: 0x28

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1	NDT5V_IE	NDT0: Disable for VDD 5V interrupt enable bit. 0: Disable. 1: Enable. (The noise on IC VDD 5V domain detected by NDT5V IP will trigger. NDT interrupt IRQ0)	R/W	0
0	Reserved		R	0

3.3.12 Noise Detect Status register (SYS0_NDTSTS)

Address Offset: 0x2C

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1	NDT5V_DET	Power noise status of NDT5V IP. 0: No power noise is detected. 1: Power noise is detected by NDT5V IP. *Cleared by write 1 to SYS0_NDTSTS[1]	R/W	0
0	Reserved		R	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.***

Bit	Name	Description	Attribute	Reset
31	Reserved		R	0
30:28	CLKOUTSEL[2:0]	Clock output source 000: Disable 001: ILRC clock 100: HCLK 101: IHRC clock 111: PLL clock output Others: Reserved.	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: Disable 1: Enable	R/W	1
23:22	Reserved		R	0
21	I2C0LKEN	Enable clock for I2C0. 0: Disable 1: Enable	R/W	0
20	I2C1LKEN	Enable clock for I2C1. 0: Disable 1: Enable	R/W	0
19	Reserved		R	0
18	UART2CLKEN	Enable clock for UART2. 0: Disable 1: Enable	R/W	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
15:14	Reserved		R	0
13	SPI1CLKEN	Enable clock for SPI1. 0: Disable 1: Enable	R/W	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
10:8	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	Reserved		R	0
4	USBCLKEN	Enable clock for USB. 0: Disable 1: Enable	R/W	0
3:0	Reserved		R	0

3.4.2 APB Clock Prescale register 0 (SYS1_APBCP0)

Address Offset: 0x04

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_APBCP1)

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. Other: Reserved	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

Bit	Name	Description	Attribute	Reset
31:28	Reserved		R	0
27	USBRST	USB reset 0: No effect 1: Reset USB	R/W	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 1: Reset WDT	R/W	0
23:22	Reserved		R	0
21	I2C0RST	I2C0 reset 0: No effect 1: Reset I2C0	R/W	0
20	I2C1RST	I2C1 reset 0: No effect	R/W	0

		1: Reset I2C1		
19	Reserved		R	0
18	UART2RST	UART2 reset 0: No effect 1: Reset UART2	R/W	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
15:14	Reserved		R	0
13	SPI1RST	SPI1 reset 0: No effect 1: Reset SPI1	R/W	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
10:8	Reserved		R	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 three operating mode 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, Flash, and LVD) can be controlled at any time individually through the enable bit of all blocks.

4.3 LOW-POWER MODES

There are two 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, 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, Flash, and LVD) 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.

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.

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 GPIOn interrupt to the ARM core. Furthermore, the interrupts corresponding to each input must be enabled in the NVIC. Wake up time is 56us at the VDD = 3.3V.

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, 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 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 256 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 10us wakeup time. After the wakeup time, the system goes into the normal mode.

The value of the IHRC wakeup time is as the following.

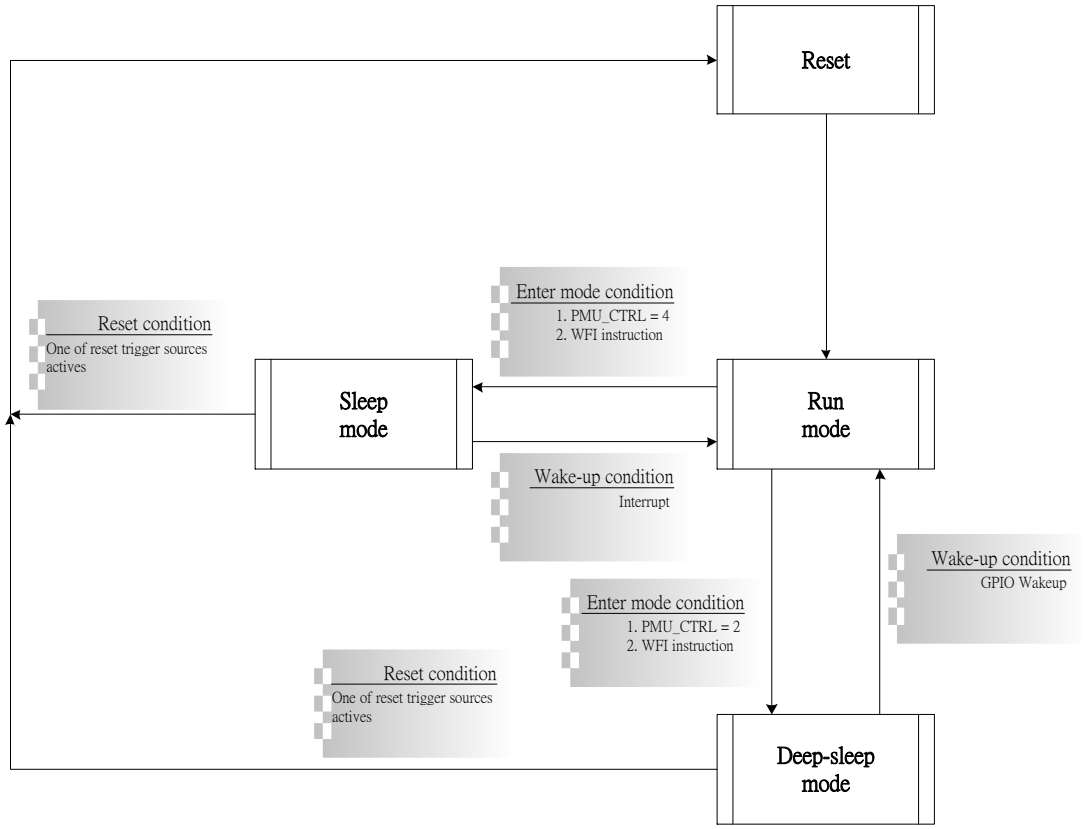
The total Wakeup time of IHRC = 10us + 1/F_{IHRC} * 32 (sec)

➢ Example: F_{IHRC}=12MHz, the wakeup time is as the following.

The total Wakeup time = 10us + 1/F_{IHRC} * 32 = 12.67us (F_{IHRC} = 12MHz)

✸ ***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
HCLK	IHRC, ILRC	IHRC	ILRC	-
IHRC	HW Enable	By IHRCEN	FW Disable	HW Disable
ILRC	HW Enable	HW Enable	HW Enable	HW Disable
Cortex-M0	Running	Stop	Stop	Stop
Flash ROM	Enable/Standby	Standby	Standby	Enable/Standby
Data RAM	Enable/Standby	Standby	Standby	Standby
3-level LVD	By LVDEN	By LVDEN		By LVDEN
USB	By USBEN	By USBEN		Disable
Peripherals	By Enable bit of each peripherals			Disable HCLK
IO status	Output Low	Output Low		Output Low
Wakeup Source	N/A	All interrupts, RESET pin		GPIO0/1/2/3 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: Disable	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.
- Individual interrupt levels can be programmed. Internal pull-up resistor.
- All GPIO pins are inputs and floating by default

5.2 GPIO MODE

The MODE bits in the GPIOn_CFG (n=0,1,2,3) register allow the selection of on-chip pull-up resistors, for each pin or select the inactive mode or inactive with Schmitt trigger disabled mode.

The possible on-chip resistor configurations are pull-up enabled, no pull-up with Schmitt trigger enabled (default), or no pull-up with Schmitt trigger disabled.

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
31:16	Reserved		R	0
15:0	DATA[15:0]	Input data (read) or output data (write) for Pn.0 to Pn.15	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) is enabled, not through GPIOn_MODE register.***

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	MODE[15:0]	Selects pin x as input or output (x = 0 to 15) 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: n=0 → 0xAAAA AAAA
n=1 → 0xAAAA AAAA
n=2 → 0xAAAA AAAA
n=3 → 0x00AA A80A

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

Bit	Name	Description	Attribute	Reset
31:30	CFG15[1:0]	Configuration of Pn.15 00: Pull-up resistor enabled. 01: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
29:28	CFG14[1:0]	Configuration of Pn.14 00: Pull-up resistor enabled. 01: Reserved.	R/W	10b

		10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-up resistor enabled, Schmitt trigger disabled, Data register keep low)		
27:26	CFG13[1:0]	Configuration of Pn.13 00: Pull-up resistor enabled. 01: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
25:24	CFG12[1:0]	Configuration of Pn.12 00: Pull-up resistor enabled. 01: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
23:22	CFG11[1:0]	Configuration of Pn.11 00: Pull-up resistor enabled. 01: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b
19:18	CFG9[1:0]	Configuration of Pn.9 00: Pull-up resistor enabled. 01: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-up resistor enabled, Schmitt trigger disabled, Data	R/W	10b

		register keep low)		
5:4	CFG2[1:0]	Configuration of Pn.2 00: Pull-up resistor enabled. 01: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-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: Reserved. 10: Inactive. (no pull-up resistor enabled, Schmitt trigger enabled). 11: Inactive. (no pull-up resistor enabled, Schmitt trigger disabled, Data register keep low)	R/W	10b

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

Address offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	IS[15:0]	Selects interrupt on pin x as level or edge sensitive. (x = 0 to 15) 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 (GPIOn_IBS) (n=0,1,2,3)

Address offset: 0x10

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	IBS[15:0]	Selects interrupt on Pn.x to be triggered on both edges. (x = 0 to 15) 0: Interrupt on Pn.x is controlled through register GPIOn_IEV. 1: Both edges on Pn.x trigger an interrupt.	R/W	0

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

Address offset: 0x14

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	IEV[15:0]	Selects interrupt on pin x to be triggered rising or falling edges. (x = 0 to 15) 0: Depending on setting in register GPIOn_IS, Rising edges or HIGH level on Pn.x trigger an interrupt. 1: Depending on setting in register GPIOn_IS, Falling edges or LOW level on Pn.x trigger an interrupt.	R/W	0

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

Address offset: 0x18

Bits set to HIGH in the GPIOn_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
31:16	Reserved		R	0
15:0	IE[15:0]	Selects interrupt on pin x to be enabled. (x = 0 to 15) 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 (GPIOn_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 GPIOn_IE register is set.

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	IF[15:0]	GPIO raw interrupt flag. (x = 0 to 15) 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
31:16	Reserved		R	0
15:0	IC[15:0]	Selects interrupt flag on pin x to be cleared. (x = 0 to 15) 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
31:16	Reserved		R	0
15:0	BSET[15:0]	Bit Set enable. (x = 0 to 15) 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
31:16	Reserved		R	0
15:0	BCLR[15:0]	Bit clear enable. (x = 0 to 15) 0: No effect on Pn.x. 1: Clear Pn.x.	W	0

5.3.12 GPIO Port n Driving Current Select register (GPIOn_DCS) (n=0,1,2)

Address offset: 0x2C

Bit	Name	Description	Attribute	Reset
31:30	DCS15[1:0]	Driving current of Pn.15 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
29:28	DCS14[1:0]	Driving current of Pn.14 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
27:26	DCS13[1:0]	Driving current of Pn.13 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
25:24	DCS12[1:0]	Driving current of Pn.12 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
23:22	DCS11[1:0]	Driving current of Pn.11 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
21:20	DCS10[1:0]	Driving current of Pn.10 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
19:18	DCS9[1:0]	Driving current of Pn.9 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
17:16	DCS8[1:0]	Driving current of Pn.8 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
15:14	DCS7[1:0]	Driving current of Pn.7 00: 5mA 01: 10mA	R/W	11b

		10: 15mA 11: 20mA		
13:12	DCS6[1:0]	Driving current of Pn.6 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
11:10	DCS5[1:0]	Driving current of Pn.5 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
9:8	DCS4[1:0]	Driving current of Pn.4 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
7:6	DCS3[1:0]	Driving current of Pn.3 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
5:4	DCS2[1:0]	Driving current of Pn.2 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
3:2	DCS1[1:0]	Driving current of Pn.1 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
1:0	DCS0[1:0]	Driving current of Pn.0 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b

5.3.13 GPIO Port n Driving Current Select register (GPIOn_DCS) (n=3)

Address offset: 0x2C

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23:22	DCS11[1:0]	Driving current of Pn.11 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
21:20	DCS10[1:0]	Driving current of Pn.10 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
19:18	DCS9[1:0]	Driving current of Pn.9 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
17:16	DCS8[1:0]	Driving current of Pn.8 00: 5mA 01: 10mA 10: 15mA	R/W	11b

		11: 20mA		
15:14	DCS7[1:0]	Driving current of Pn.7 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
13:12	DCS6[1:0]	Driving current of Pn.6 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
11:10	DCS5[1:0]	Driving current of Pn.5 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
9:4	Reserved		R	0
3:2	DCS1[1:0]	Driving current of Pn.1 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b
1:0	DCS0[1:0]	Driving current of Pn.0 00: 5mA 01: 10mA 10: 15mA 11: 20mA	R/W	11b

5.3.14 GPIO Port n Sinking Current Select register (GPIOn_SCS) (n=0,1)

Address offset: 0x30

Sinking current of P0.8~P0.15, P1.0~P1.11

Bit	Name	Description	Attribute	Reset
31:30	SCS15[1:0]	Sinking current of Pn.15 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
29:28	SCS14[1:0]	Sinking current of Pn.14 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
27:26	SCS13[1:0]	Sinking current of Pn.13 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
25:24	SCS12[1:0]	Sinking current of Pn.12 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
23:22	SCS11[1:0]	Sinking current of Pn.11 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
21:20	SCS10[1:0]	Sinking current of Pn.10 00: 100mA	R/W	00b

		01: 200mA 10: 320mA 11: 420mA		
19:18	SCS9[1:0]	Sinking current of Pn.9 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
17:16	SCS8[1:0]	Sinking current of Pn.8 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
15:14	SCS7[1:0]	Sinking current of Pn.7 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
13:12	SCS6[1:0]	Sinking current of Pn.6 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
11:10	SCS5[1:0]	Sinking current of Pn.5 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
9:8	SCS4[1:0]	Sinking current of Pn.4 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
7:6	SCS3[1:0]	Sinking current of Pn.3 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
5:4	SCS2[1:0]	Sinking current of Pn.2 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
3:2	SCS1[1:0]	Sinking current of Pn.1 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b
1:0	SCS0[1:0]	Sinking current of Pn.0 00: 100mA 01: 200mA 10: 320mA 11: 420mA	R/W	00b

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 on the package.

6.2 FEATURES

- Flexible assignment of digital peripheral functions to desired pins.
- Support functions are PWM.

6.3 PIN ASSIGNMENT LIST

Peripheral	Pin Name	PA	PB
CT16B1_PWM	CT16B1_PWM00	P0.0	P1.8
	CT16B1_PWM01	P0.1	P1.9
	CT16B1_PWM02	P0.2	P1.10
	CT16B1_PWM03	P0.3	P1.11
	CT16B1_PWM04	P0.4	P1.12
	CT16B1_PWM05	P0.5	P1.13
	CT16B1_PWM06	P0.6	P1.14
	CT16B1_PWM07	P0.7	P1.15
	CT16B1_PWM08	P0.8	P2.0
	CT16B1_PWM09	P0.9	P2.1
	CT16B1_PWM10	P0.10	P2.2
	CT16B1_PWM11	P0.11	P2.3
	CT16B1_PWM12	P0.12	P2.4
	CT16B1_PWM13	P0.13	P2.5
	CT16B1_PWM14	P0.14	P2.6
	CT16B1_PWM15	P0.15	P2.7
	CT16B1_PWM16	P1.0	P2.8
	CT16B1_PWM17	P1.1	P2.9
	CT16B1_PWM18	P1.2	P2.10
	CT16B1_PWM19	P1.3	P2.11
	CT16B1_PWM20	P1.4	P2.12
	CT16B1_PWM21	P1.5	P2.13
	CT16B1_PWM22	P1.6	P2.14
	CT16B1_PWM23	P1.7	P2.15

Peripheral	Pin Name	PA	PB	PC	PD
I2C0	SCL0	P0.6	P0.15	P1.14	P2.8
	SDA0	P0.7	P0.14	P1.15	P2.7
I2C1	SCL1	P1.7	P0.12	P2.13	P1.8
	SDA1	P1.6	P0.13	P2.14	P1.9
SPI0	SEL0	P0.5	P1.2	P1.6	P2.12
	SCK0	P0.4	P1.4	P1.7	P2.11
	MISO0	P0.2	P1.5	P1.8	P2.10

	MOSI0	P0.3	P1.3	P1.9	P2.9
SPI1	SEL1	P1.13	P0.9	P2.1	P2.15
	SCK1	P1.12	P0.8	P2.5	P3.10
	MISO1	P1.11	P0.10	P2.4	P3.11

Peripheral	Pin Name	PA	PB	PC	PD	PE	PF	PG	PH
SPI1	MOSI1	P1.10	P0.11	P2.3	P3.9	P0.1	P1.3	P1.9	P2.9

6.4 PFPA REGISTERS

Base Address: 0x4004 2000

6.4.1 PFPA for CT16B1 register (PFPA_CT16B1)

Address offset: 0x00

Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23	PWM23	Pin to be assigned as CT16B1_PWM23. 0: P1.7 1: P2.15	R/W	0
22	PWM22	Pin to be assigned as CT16B1_PWM22. 0: P1.6 1: P2.14	R/W	0
21	PWM21	Pin to be assigned as CT16B1_PWM21. 0: P1.5 1: P2.13	R/W	0
20	PWM20	Pin to be assigned as CT16B1_PWM20. 0: P1.4 1: P2.12	R/W	0
19	PWM19	Pin to be assigned as CT16B1_PWM19. 0: P1.3 1: P2.11	R/W	0
18	PWM18	Pin to be assigned as CT16B1_PWM18. 0: P1.2 1: P2.10	R/W	0
17	PWM17	Pin to be assigned as CT16B1_PWM17. 0: P1.1 1: P2.9	R/W	0
16	PWM16	Pin to be assigned as CT16B1_PWM16. 0: P1.0 1: P2.8	R/W	0
15	PWM15	Pin to be assigned as CT16B1_PWM15. 0: P0.15 1: P2.7	R/W	0
14	PWM14	Pin to be assigned as CT16B1_PWM14. 0: P0.14 1: P2.6	R/W	0
13	PWM13	Pin to be assigned as CT16B1_PWM13. 0: P0.13 1: P2.5	R/W	0
12	PWM12	Pin to be assigned as CT16B1_PWM12. 0: P0.12 1: P2.4	R/W	0
11	PWM11	Pin to be assigned as CT16B1_PWM11. 0: P0.11 1: P2.3	R/W	0
10	PWM10	Pin to be assigned as CT16B1_PWM10. 0: P0.10 1: P2.2	R/W	0
9	PWM09	Pin to be assigned as CT16B1_PWM09. 0: P0.9 1: P2.1	R/W	0
8	PWM08	Pin to be assigned as CT16B1_PWM08. 0: P0.8 1: P2.0	R/W	0
7	PWM07	Pin to be assigned as CT16B1_PWM07. 0: P0.7 1: P1.15	R/W	0
6	PWM06	Pin to be assigned as CT16B1_PWM06. 0: P0.6 1: P1.14	R/W	0

5	PWM05	Pin to be assigned as CT16B1_PWM05. 0: P0.5 1: P1.13	R/W	0
4	PWM04	Pin to be assigned as CT16B1_PWM04. 0: P0.4 1: P1.12	R/W	0
3	PWM03	Pin to be assigned as CT16B1_PWM03. 0: P0.3 1: P1.11	R/W	0
2	PWM02	Pin to be assigned as CT16B1_PWM02. 0: P0.2 1: P1.10	R/W	0
1	PWM01	Pin to be assigned as CT16B1_PWM01. 0: P0.1 1: P1.9	R/W	0
0	PWM00	Pin to be assigned as CT16B1_PWM00. 0: P0.0 1: P1.8	R/W	0

✷ ***Note:***
(1) The same PWMn can only select PA or PB output at a time. For example, PWM00 can only choose PA(P0.0) or PB(P1.8) output, can not be together to PA(P0.0) and PB(P1.8) output.
(2) The different PWMn can be both PA and PB output. For example, PWM00 can select PA(P0.0) output, and PWM01 can choose PB(P1.9) output.

6.4.2 PFPA for INTERFACE register (PFPA_INTERFACE)

Address offset: 0x04

Bit	Name	Description	Attribute	Reset
31:11	Reserved		R	0
10:8	MOSI1[2:0]	Pin to be assigned as SPI1 MOSI1 000: PA 001: PB 010: PC 011: PD 100: PE 101: PF 110: PG 111: PH	R/W	0
7:6	SPI1[1:0]	Pin to be assigned as SPI1 00: PA 01: PB 10: PC 11: PD	R/W	0
5:4	SPI0[1:0]	Pin to be assigned as SPI0 00: PA 01: PB 10: PC 11: PD	R/W	0
3:2	I2C1[1:0]	Pin to be assigned as I2C1 00: PA 01: PB 10: PC 11: PD	R/W	0
1:0	I2C0[1:0]	Pin to be assigned as I2C0 00: PA 01: PB 10: PC 11: PD	R/W	0

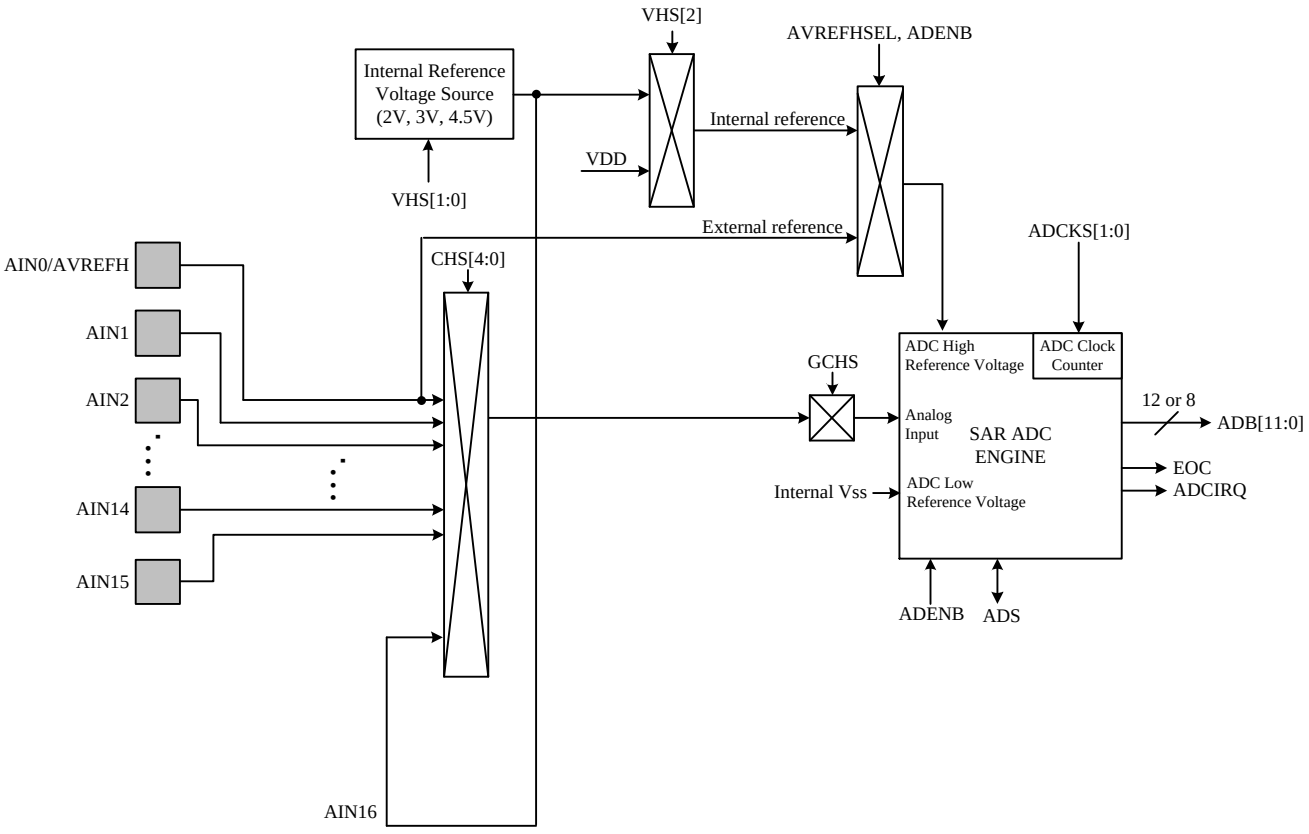
7 16+1 CHANNEL 12-BIT SAR ADC

7.1 OVERVIEW

This analog to digital converter (ADC) has 16external input sources and one internal channel to internal voltage source (2V/3V/4.5V), with up to 4096-step resolution to transfer analog signal into 12-bits digital data. The sequence of ADC operation is to select input source (AIN0 ~ AIN15) 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 resistor. Use CHS[3:0] to select AIN pin and GCHS enables global ADC channel, the analog signal inputs to ADC engine.

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 AIN16 is internal 2V or 3V or 4.5V input channel, there is no any input pin from outside. In this time ADC reference voltage must be internal VDD and External voltage, not internal 2V or 3V or 4.5V. AIN16 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: For 8-bit resolution the conversion time is 12 steps.***
  For 12-bit resolution the conversion time is 16 steps
- ***Note: The analog input level must be between the AVREFH and AVSS.***
- ***Note: The AVREFH level must be between the AVDD and AVSS + 2.0V.***
- ***Note: ADC programming notice:***
  1. Set ADC input pin I/O direction as input mode
  2. Disable pull-up resistor of ADC input pin
  3. Disable ADC (set ADENB = "0")before enter low-power (Sleep/Deep-sleep) mode to save power consumption.
  4. Delay 100us after enable ADC (set ADENB = "1") to wait ADC circuit ready for conversion.

7.2 ADC CONVERSION 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 clock /4)*16 sec

ADLEN	ADCKS [2:0]	ADC Clock	ADC_PCLK = 3 MHz		ADC_PCLK = 6 MHz		ADC_PCLK = 12 MHz	
			ADC Conversion Time (us)	ADC Conversion Rate (KHz)	ADC Conversion Time (us)	ADC Conversion Rate (KHz)	ADC Conversion Time (us)	ADC Conversion Rate (KHz)
1	000	ADC_PCLK	21.33	46.875	10.67	93.75	5.33	187.5
	001	ADC_PCLK/2	42.67	23.437	21.33	46.875	10.67	93.75
	010	ADC_PCLK/4	85.3	11.718	42.67	23.437	21.33	46.875
	011	ADC_PCLK/8	170.67	5.859	85.3	11.718	42.67	23.437
	100	ADC_PCLK/16	341.33	2.929	170.67	5.859	85.3	11.718
	101	ADC_PCLK/32	682.67	1.465	341.33	2.929	170.67	5.859

8-bit ADC conversion time = 1/(ADC clock /4)*12 sec

ADLEN	ADCKS [2:0]	ADC Clock	ADC_PCLK = 3 MHz		ADC_PCLK = 6 MHz		ADC_PCLK = 12 MHz	
			ADC Conversion Time (us)	ADC Conversion Rate (KHz)	ADC Conversion Time (us)	ADC Conversion Rate (KHz)	ADC Conversion Time (us)	ADC Conversion Rate (KHz)
0	000	ADC_PCLK	16	62.5	8	125	4	250
	001	ADC_PCLK/2	32	31.25	16	62.5	8	125
	010	ADC_PCLK/4	64	15.625	32	31.25	16	62.5
	011	ADC_PCLK/8	128	7.813	64	15.625	32	31.25
	100	ADC_PCLK/16	256	3.906	128	7.813	64	15.625
	101	ADC_PCLK/32	512	1.953	256	3.906	128	7.813

7.3 ADC CONTROL NOTICE

7.3.1 ADC Signal

The ADC high reference voltage includes internal VDD/4.5V/3V/2V and external reference voltage source from P2.0/AVREFH pin controlled by AVREFHSEL bit. If AVREFHSEL=0, ADC reference voltage is from internal voltage source; if AVREFHSEL=1, ADC reference voltage is from external voltage source (P2.0/AVREFH).
ADC reference voltage range limitation is **"(ADC high reference voltage – low reference voltage) ≧ 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/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) ≦ ADC Sampled Input Voltage ≦ ADC High Reference Voltage

7.3.2 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: Enable ADC. ADENB is ADC control bit to control. ADENB = 1 is to enable ADC. ADENB = 0 is to disable 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 2: If the ADC high reference voltage is from external voltage source, set the AVREFHSEL = 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 3: Select the ADC input pin by CHS[2:0], and enable ADC global input. **When one AIN pin is selected to be analog signal input pin, it is necessary to setup the pin as input mode and disable the pull-up resistor by program.**

Step 4: Start to execute ADC conversion by setting ADS = 1.

Step 5: Wait the end of ADC converting through checking EOC = 1 or ADCIF = 1. If ADC interrupt functions 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.***

7.3.3 ADC PIN CONFIGURATION

ADC input pins are shared with Port 2 digital I/O pins. ADC channel selection is through CHS[4:0] bits in ADC_ADM register. CHS[4:0] value points to the ADC input channel directly, CHS[4:0]=00000b selects AIN0, CHS[4:0]=00001b selects AIN1, etc.

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 Port 2 will encounter above current leakage situation.

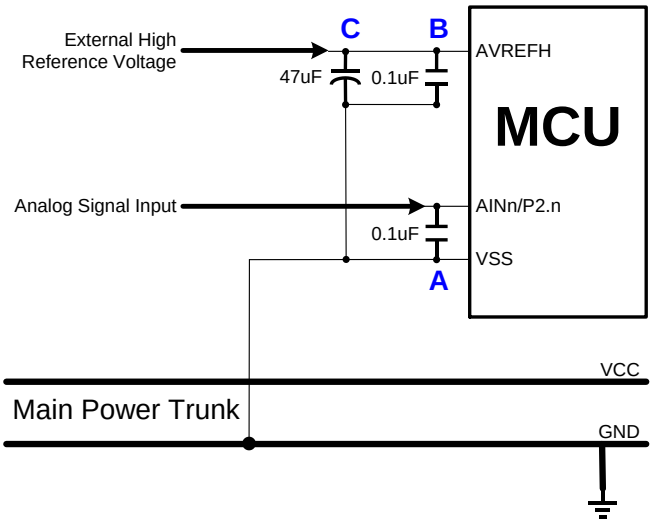
The P2.0/AIN0 can be ADC external high reference voltage input pin when AVREFHSEL =1. In the condition, P2.0 GPIO mode must be set as input mode and inactive (no pull-down/up resistor enabled, Schmitt trigger disabled) with GPIO2_MODE and GPIO2_CFG register by program.

Only one pin of Port 2 can be configured as ADC input in the same time. The pins of Port 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 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 GPIO2_MODE and GPIO2_CFG register by program.***

7.4 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.

7.5 ADC REGISTERS

Base Address: 0x4002 6000

7.5.1 ADC Management register (ADC_ADM)

Address offset: 0x00

✸ ***Note: If AIN16 channel is selected as internal 2V, 3V, or 4.5V input channel, there is no any input pin from outside. In this time ADC reference voltage must be internal VDD or External voltage, not internal 2V, 3V, or 4.5V.***

Bit	Name	Description	Attribute	Reset
31:17	Reserved		R	0
16	GCHS	ADC global channel select bit. 0: Disable AIN channel 1: Enable AIN channel	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: Reserved 100: VDD as ADC internal reference high voltage, Internal 2.0V as AIN16 101: VDD as ADC internal reference high voltage, Internal 3.0V as AIN16 110: VDD as ADC internal reference high voltage, Internal 4.5V as AIN16 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. 0: Disable 1: Enable	R/W	0
10:8	ADCKS[2:0]	ADC Clock source divider. 000: ADC_PCLK / 1 001: ADC_PCLK / 2 010: ADC_PCLK / 4 011: ADC_PCLK / 8 101: ADC_PCLK / 16 110: ADC_PCLK / 32 Other: Reversed	R/W	0
7	ADLEN	ADC resolution control bit. 0: 8-bit ADC. 1:12-bit ADC.	R/W	0
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 progressing. 1: End of converting and reset ADS bit.	R/W	0
4:0	CHS[4:0]	ADC input channels select bit. 00000: AIN0 00001: AIN1 00010: AIN2 00011: AIN3 00100: AIN4 00101: AIN5 00110: AIN6 00111: AIN7 01000: AIN8	R/W	0

		01001: AIN9 01010: AIN10 01011: AIN11 01100: AIN12 01101: AIN13 01110: AIN14 01111: AIN15 10000: AIN16 (Internal reference voltage 4.5V/3V/2V) Other: Reserved		

7.5.2 ADC Data register (ADC_ADB)

Address offset: 0x04

✸ ***Note: The initial value of ADC buffer (ADB) after reset is unknown.***

Bit	Name	Description	Attribute	Reset
31:12	Reserved		R	0
11:0	ADB[11:0]	ADB11~ADB4 bits for 8-bit ADC ADB11~ADB0 bits for 12-bit ADC	R	0

The AIN's 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

For different applications, users maybe need more than 8-bit resolution but less than 12-bit ADC converter. First, the AD resolution must be set 12-bit mode and then to execute ADC converter routine. Then delete the LSB of ADC data and get the new resolution result. The table is as following.

	ADB11	ADB10	ADB9	ADB8	ADB7	ADB6	ADB5	ADB4	ADB3	ADB2	ADB1	ADB0
8-bit	O	O	O	O	O	O	O	O	X	X	X	X
9-bit	O	O	O	O	O	O	O	O	O	X	X	X
10-bit	O	O	O	O	O	O	O	O	O	O	X	X
11-bit	O	O	O	O	O	O	O	O	O	O	O	X
12-bit.	O	O	O	O	O	O	O	O	O	O	O	O
O = Selected, X = Delete												

7.5.3 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
31:17	Reserved		R	0

16:0	IE[16:0]	These bits allow control over which A/D channels generate interrupts for conversion completion. When bit x is one, completion of a conversion on AIN x will generate an interrupt.	R/W	0

7.5.4 ADC Raw Interrupt Status register (ADC_RIS)

Address offset: 0x10

Bit	Name	Description	Attribute	Reset
31:17	Reserved		R	0
16:0	EOCIF[16:0]	ADC raw interrupt flag. (x = 0 to 16). 0: Read→No interrupt on AINx Write→Write "0" to the corresponding bit will clear the bit and reset the Interrupt if the corresponding IE bit is set. 1: Interrupt requirements (AINx finishes conversion) met on AINx.	R/W	0

8 16-BIT TIMER0 WITH CAPTURE FUNCTION

8.1 OVERVIEW

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 24 match and 1 global registers can be used to provide a single-edge controlled PWM output on the match output pins.

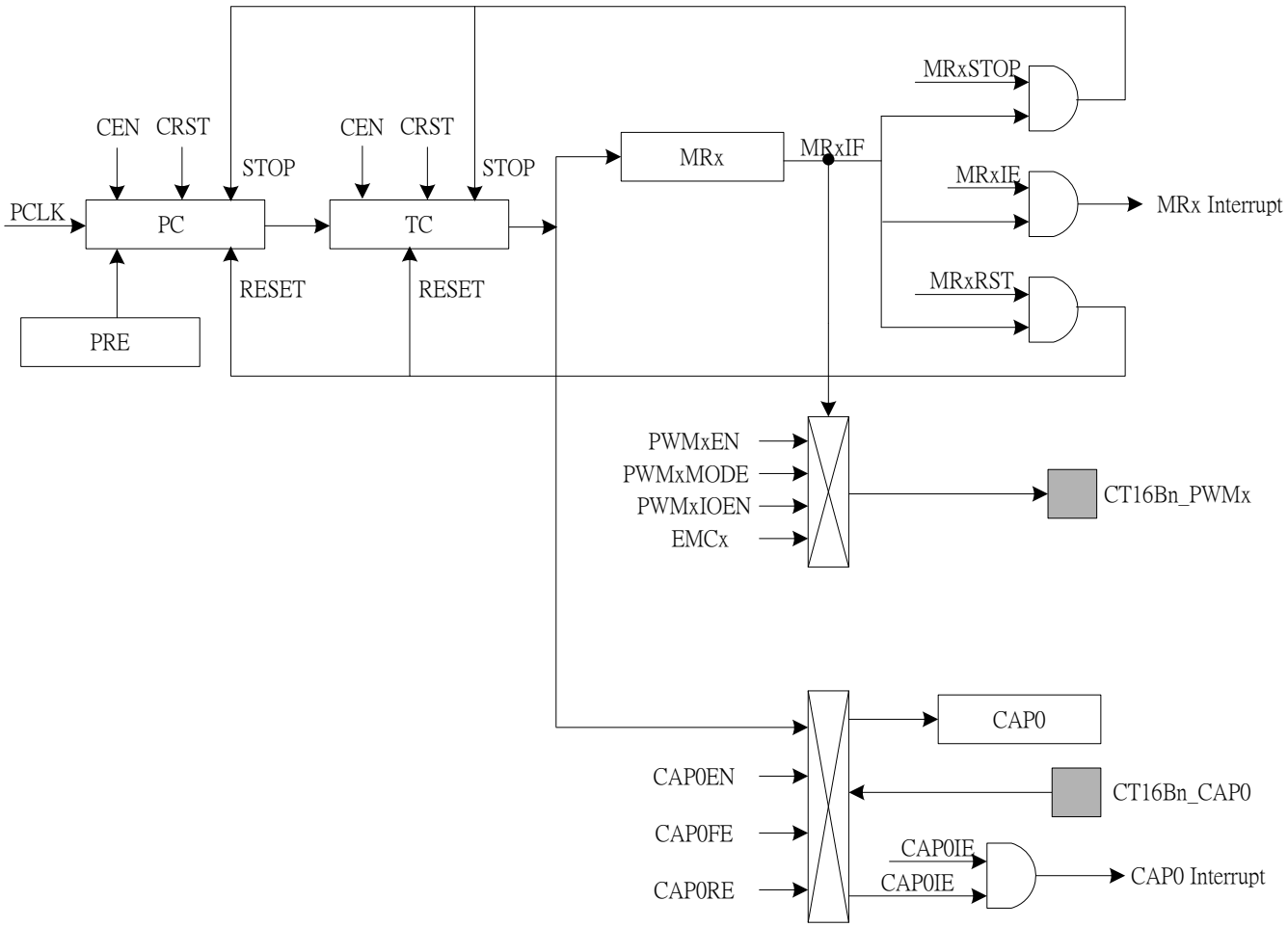
8.2 FEATURES

- Two 16-bit counter/timers with a programmable 8-bit prescaler.
- Counter or timer operation
- Two 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.
- Four 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 24(CT16B1) 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.

8.3 PIN DESCRIPTION

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

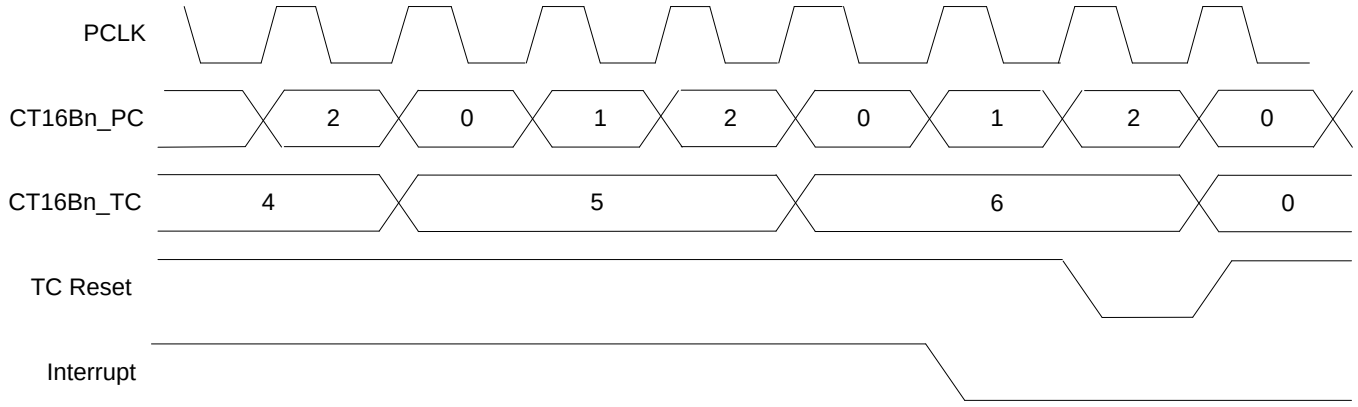
8.4 BLOCK DIAGRAM



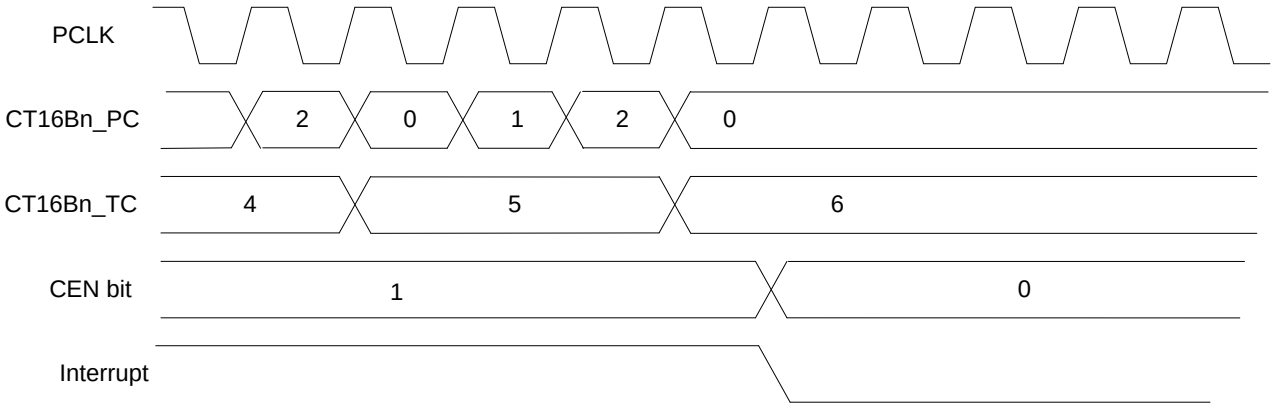
8.5 TIMER OPERATION

8.5.1 Edge-aligned Up-counting Mode

The following figure shows a timer configured to reset the count and generate an interrupt on match. 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. 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.



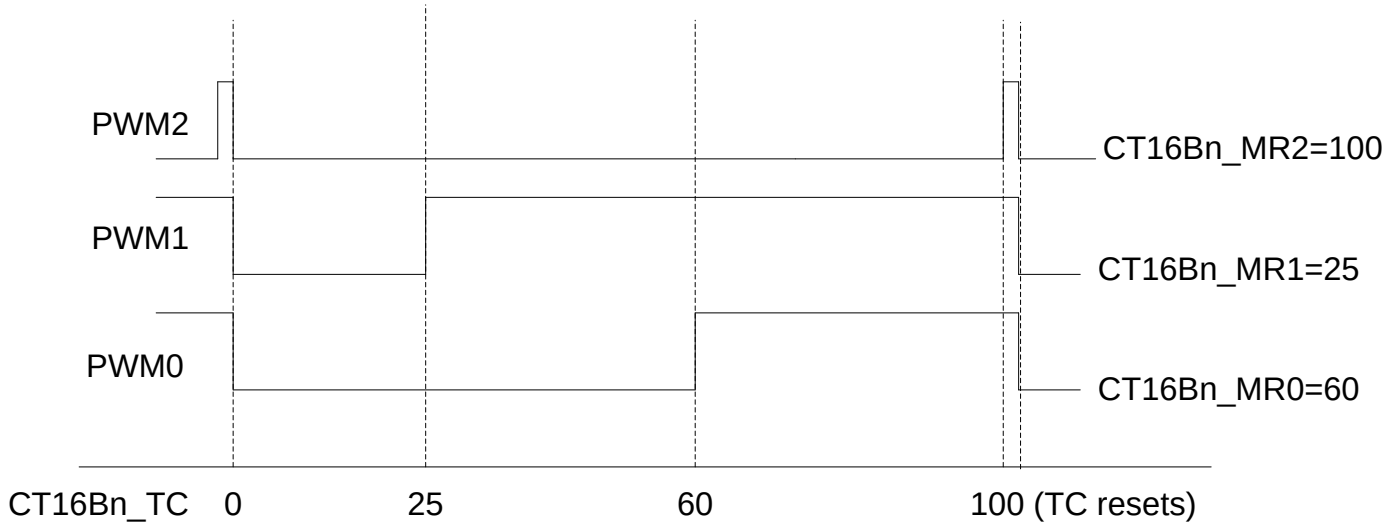
8.6 PWM

8.6.1 PWM Mode 1

- PWMn is 0 when TC<MRn during Up-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 to HIGH the first time the timer goes back to zero and will stay HIGH continuously.



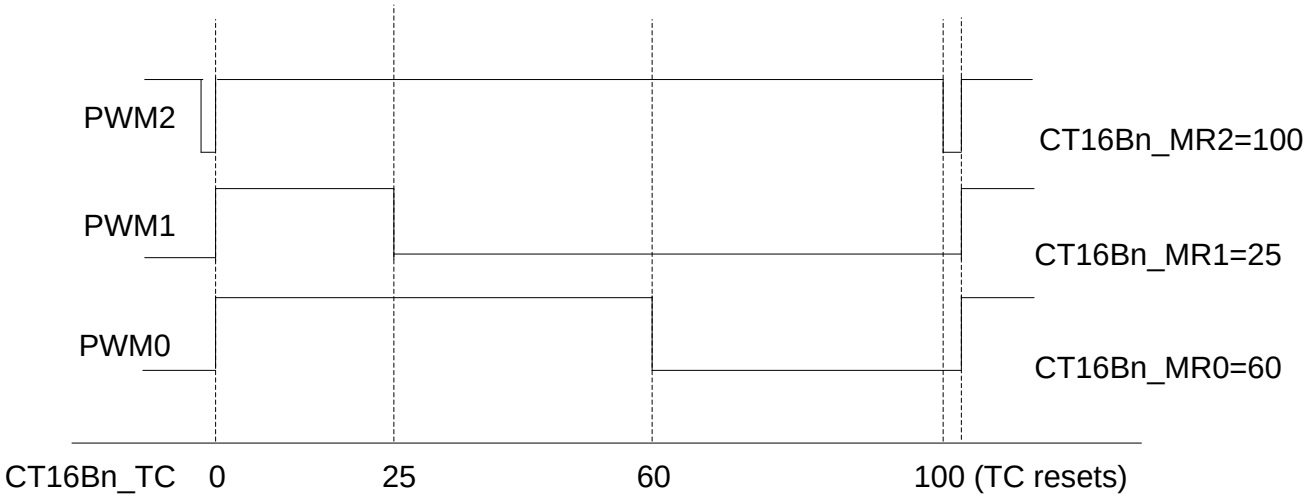
✸ ***Note: When the match outputs are selected to perform as PWM outputs, the timer reset (MRnRST) and timer stop (MRnSTOP) bits in CT16Bn_MCTRL register must be set to zero except for the match register setting the PWM cycle length. For this register, set the MRnR bit to one to enable the timer reset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

8.6.2 PWM Mode 2

➢ PWMn is 1 when TC<MRn during Up-counting period.

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.***

8.7 CT16Bn REGISTERS

Base Address: 0x4000 0000 (CT16B0)
0x4000 2000 (CT16B1)

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

Address Offset: 0x00

✸ ***Note: In order to initial TC and PC correctly, SW shall reset TC and PC by setting CRST to 1, and then enable counter by setting CEN to 1.***

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	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. CEN bit shall be set at last! * Always Edge-aligned Up-counting mode	R/W	0

8.7.2 CT16Bn Timer Counter register (CT16Bn_TC) (n=0,1)

Address Offset: 0x04

The 16-bit Timer Counter is incremented when the Prescale Counter reaches its terminal count. 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.

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

8.7.3 CT16Bn Prescale register (CT16Bn_PRE) (n=0,1)

Address Offset: 0x08

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

8.7.4 CT16Bn Prescale Counter register (CT16Bn_PC) (n=0,1)

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

8.7.5 CT16Bn Count Control register (CT16Bn_CNTCTRL) (n=0)

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 (selected by the CIS bits) 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 x PCLK).

✸ ***Note: If Counter mode is selected, bit2~0 of Capture Control (CAPCTRL) register must be programmed as 0x0.***

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3:2	CIS[1:0]	Count Input Select. In counter mode (when CTM[1:0] are not 00), these bits select which CAP0 pin is sampled for clocking. 00: CT16Bn_CAP0 Other: Reserved.	R/W	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 selected by CIS bits. 10: Counter Mode: TC is incremented on falling edges on the CAP0 input selected by CIS bits. 11: Counter Mode: TC is incremented on both edges on the CAP0 input selected by CIS bits.	R/W	0

8.7.6 CT16Bn Match Control register (CT16Bn_MCTRL) (n=0)

Address Offset: 0x14

Bit	Name	Description	Attribute	Reset
31:3	Reserved		R	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

8.7.7 CT16Bn Match Control register (CT16Bn_MCTRL) (n=1)

Address Offset: 0x14

Bit	Name	Description	Attribute	Reset
31:30	Reserved		R	0
29	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
28	MR9RST	Enable reset TC when MR9 matches TC. 0: Disable. 1: Enable.	R/W	0
27	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
26	MR8STOP	Stop MR8: TC and PC will stop and CEN bit will be cleared if MR8 matches TC. 0: Disable. 1: Enable.	R/W	0
25	MR8RST	Enable reset TC when MR8 matches TC. 0: Disable. 1: Enable.	R/W	0
24	MR8IE	Enable generating an interrupt based on CM[2:0] when MR8 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
23	MR7STOP	Stop MR7: TC and PC will stop and CEN bit will be cleared if MR7 matches TC. 0: Disable. 1: Enable.	R/W	0
22	MR7RST	Enable reset TC when MR7 matches TC. 0: Disable. 1: Enable.	R/W	0
21	MR7IE	Enable generating an interrupt based on CM[2:0] when MR7 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
20	MR6STOP	Stop MR6: TC and PC will stop and CEN bit will be cleared if MR6 matches TC. 0: Disable. 1: Enable.	R/W	0
19	MR6RST	Enable reset TC when MR6 matches TC. 0: Disable. 1: Enable.	R/W	0
18	MR6IE	Enable generating an interrupt based on CM[2:0] when MR6 matches the value in the TC.	R/W	0

		0: Disable. 1: Enable.		
17	MR5STOP	Stop MR5: TC and PC will stop and CEN bit will be cleared if MR5 matches TC. 0: Disable. 1: Enable.	R/W	0
16	MR5RST	Enable reset TC when MR5 matches TC. 0: Disable. 1: Enable.	R/W	0
15	MR5IE	Enable generating an interrupt based on CM[2:0] when MR5 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
14	MR4STOP	Stop MR4: TC and PC will stop and CEN bit will be cleared if MR4 matches TC. 0: Disable. 1: Enable.	R/W	0
13	MR4RST	Enable reset TC when MR4 matches TC. 0: Disable. 1: Enable.	R/W	0
12	MR4IE	Enable generating an interrupt based on CM[2:0] when MR4 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
11	MR3STOP	Stop MR3: TC and PC will stop and CEN bit will be cleared if MR3 matches TC. 0: Disable. 1: Enable.	R/W	0
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.	R/W	0

		1: Enable.		

8.7.8 CT16Bn Match Control register 2(CT16Bn_MCTRL2) (n=1)

Address Offset: 0x18

Bit	Name	Description	Attribute	Reset
31:30	Reserved		R	0
29	MR19STOP	Stop MR19: TC and PC will stop and CEN bit will be cleared if MR19 matches TC. 0: Disable. 1: Enable.	R/W	0
28	MR19RST	Enable reset TC when MR19 matches TC. 0: Disable. 1: Enable.	R/W	0
27	MR19IE	Enable generating an interrupt based on CM[2:0] when MR19 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
26	MR18STOP	Stop MR18: TC and PC will stop and CEN bit will be cleared if MR18 matches TC. 0: Disable. 1: Enable.	R/W	0
25	MR18RST	Enable reset TC when MR18 matches TC. 0: Disable. 1: Enable.	R/W	0
24	MR18IE	Enable generating an interrupt based on CM[2:0] when MR18 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
23	MR17STOP	Stop MR17: TC and PC will stop and CEN bit will be cleared if MR17 matches TC. 0: Disable. 1: Enable.	R/W	0
22	MR17RST	Enable reset TC when MR17 matches TC. 0: Disable. 1: Enable.	R/W	0
21	MR17IE	Enable generating an interrupt based on CM[2:0] when MR17 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
20	MR16STOP	Stop MR16: TC and PC will stop and CEN bit will be cleared if MR16 matches TC. 0: Disable. 1: Enable.	R/W	0
19	MR16RST	Enable reset TC when MR16 matches TC. 0: Disable. 1: Enable.	R/W	0
18	MR16IE	Enable generating an interrupt based on CM[2:0] when MR16 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
17	MR15STOP	Stop MR15: TC and PC will stop and CEN bit will be cleared if MR15 matches TC. 0: Disable. 1: Enable.	R/W	0
16	MR15RST	Enable reset TC when MR15 matches TC. 0: Disable. 1: Enable.	R/W	0
15	MR15IE	Enable generating an interrupt based on CM[2:0] when MR15 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0

14	MR14STOP	Stop MR14: TC and PC will stop and CEN bit will be cleared if MR14 matches TC. 0: Disable. 1: Enable.	R/W	0
13	MR14RST	Enable reset TC when MR14 matches TC. 0: Disable. 1: Enable.	R/W	0
12	MR14IE	Enable generating an interrupt based on CM[2:0] when MR14 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
11	MR13STOP	Stop MR13: TC and PC will stop and CEN bit will be cleared if MR13 matches TC. 0: Disable. 1: Enable.	R/W	0
10	MR13RST	Enable reset TC when MR13 matches TC. 0: Disable. 1: Enable.	R/W	0
9	MR13IE	Enable generating an interrupt based on CM[2:0] when MR13 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
8	MR12STOP	Stop MR12: TC and PC will stop and CEN bit will be cleared if MR12 matches TC. 0: Disable. 1: Enable.	R/W	0
7	MR12RST	Enable reset TC when MR12 matches TC. 0: Disable. 1: Enable.	R/W	0
6	MR12IE	Enable generating an interrupt based on CM[2:0] when MR12 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
5	MR11STOP	Stop MR11: TC and PC will stop and CEN bit will be cleared if MR11 matches TC. 0: Disable. 1: Enable.	R/W	0
4	MR11RST	Enable reset TC when MR11 matches TC. 0: Disable. 1: Enable.	R/W	0
3	MR11IE	Enable generating an interrupt based on CM[2:0] when MR11 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
2	MR10STOP	Stop MR10: TC and PC will stop and CEN bit will be cleared if MR10 matches TC. 0: Disable. 1: Enable	R/W	0
1	MR10RST	Enable reset TC when MR10 matches TC. 0: Disable. 1: Enable.	R/W	0
0	MR10IE	Enable generating an interrupt based on CM[2:0] when MR10 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0

8.7.9 CT16Bn Match Control register 3 (CT16Bn_MCTRL3) (n=1)

Address Offset: 0x1C

Bit	Name	Description	Attribute	Reset
31:15	Reserved		R	0

14	MR24STOP	Stop MR24: TC and PC will stop and CEN bit will be cleared if MR24 matches TC. 0: Disable 1: Enable.	R/W	0
13	MR24RST	Enable reset TC when MR24 matches TC. 0: Disable. 1: Enable.	R/W	0
12	MR24IE	Enable generating an interrupt based on CM[2:0] when MR24 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
11	MR23STOP	Stop MR23: TC and PC will stop and CEN bit will be cleared if MR23 matches TC. 0: Disable 1: Enable.	R/W	0
10	MR23RST	Enable reset TC when MR23 matches TC. 0: Disable. 1: Enable.	R/W	0
9	MR23IE	Enable generating an interrupt based on CM[2:0] when MR23 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
8	MR22STOP	Stop MR22: TC and PC will stop and CEN bit will be cleared if MR22 matches TC. 0: Disable. 1: Enable.	R/W	0
7	MR22RST	Enable reset TC when MR22 matches TC. 0: Disable. 1: Enable.	R/W	0
6	MR22IE	Enable generating an interrupt based on CM[2:0] when MR22 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
5	MR21STOP	Stop MR21: TC and PC will stop and CEN bit will be cleared if MR21 matches TC. 0: Disable. 1: Enable.	R/W	0
4	MR21RST	Enable reset TC when MR21 matches TC. 0: Disable. 1: Enable.	R/W	0
3	MR21IE	Enable generating an interrupt based on CM[2:0] when MR21 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0
2	MR20STOP	Stop MR20: TC and PC will stop and CEN bit will be cleared if MR20 matches TC. 0: Disable. 1: Enable.	R/W	0
1	MR20RST	Enable reset TC when MR20 matches TC. 0: Disable. 1: Enable.	R/W	0
0	MR20IE	Enable generating an interrupt based on CM[2:0] when MR20 matches the value in the TC. 0: Disable. 1: Enable.	R/W	0

8.7.10 CT16Bn Match register 0 (CT16Bn_MR0) (n=0)

Address Offset: 0x20

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 CT16B0_MCTRL register.

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

8.7.11 CT16Bn Match register 0~24 (CT16Bn_MR0~24) (n=1)

MR 0 ~ 24:

Address Offset: 0x20, 0x24, 0x28, 0x2C, 0x30, 0x34, 0x38, 0x3C,
0x40, 0x44, 0x48, 0x4C, 0x50, 0x54, 0x58, 0x5C,
0x60, 0x64, 0x68, 0x6C, 0x70, 0x74, 0x78, 0x7C,
0x80

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 CT16B1_MCTRL register.

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

8.7.12 CT16Bn Capture Control register (CT16Bn_CAPCTRL) (n=0)

Address Offset: 0x80

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: If Counter mode is selected in the CNTCTRL register, CAPCTRL[2:0] must be programmed as 0x0.***

Bit	Name	Description	Attribute	Reset
31:4	Reserved		R	0
3	CAP0EN	Capture 0 function enable bit 0: Disable. 1: Enable. HW switches I/O Configuration directly.	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

8.7.13 CT16Bn Capture 0 register (CT16Bn_CAP0) (n=0)

Address Offset: 0x84

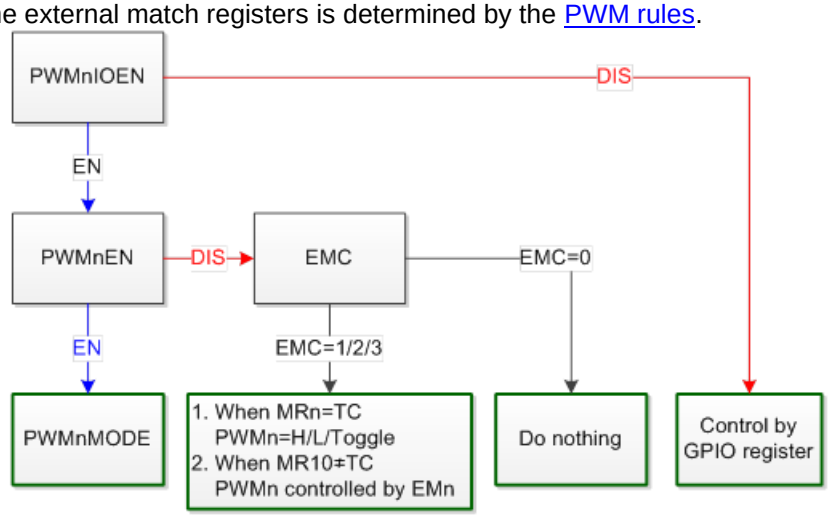
Each Capture register is associated with a device pin and may be loaded with the counter/timer value when a specified event occurs on that pin. The settings in the Capture Control register determine whether the capture function is enabled, and whether a capture event happens on the rising edge of the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

8.7.14 CT16Bn External Match register (CT16Bn_EM)(n=1)

Address Offset: 0x88

The External Match register provides status of CT16Bn_PWM [23:0]. If the match outputs are configured as PWM output, the function of the external match registers is determined by the PWM rules.



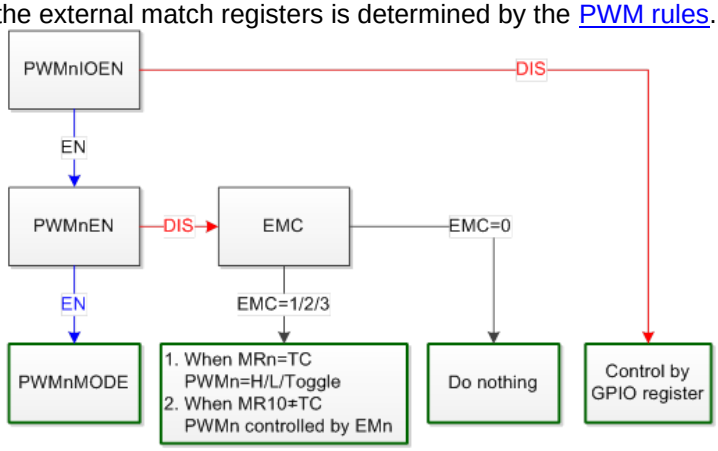
Bit	Name	Description	Attribute	Reset
31:24	Reserved		R	0
23	EM23	When EMC23≠00b and MR23≠TC, this bit will drive the state of CT16Bn_PWM23 output.	R/W	0
22	EM22	When EMC22≠00b and MR22≠TC, this bit will drive the state of CT16Bn_PWM22 output.	R/W	0
21	EM21	When EMC21≠00b and MR21≠TC, this bit will drive the state of CT16Bn_PWM21 output.	R/W	0
20	EM20	When EMC20≠00b and MR20≠TC, this bit will drive the state of CT16Bn_PWM20 output.	R/W	0
19	EM19	When EMC19≠00b and MR19≠TC, this bit will drive the state of CT16Bn_PWM19 output.	R/W	0
18	EM18	When EMC18≠00b and MR18≠TC, this bit will drive the state of CT16Bn_PWM18 output.	R/W	0
17	EM17	When EMC17≠00b and MR17≠TC, this bit will drive the state of CT16Bn_PWM17 output.	R/W	0
16	EM16	When EMC16≠00b and MR16≠TC, this bit will drive the state of	R/W	0

		CT16Bn_PWM16 output.		
15	EM15	When EMC15≠00b and MR15≠TC, this bit will drive the state of CT16Bn_PWM15 output.	R/W	0
14	EM14	When EMC14≠00b and MR14≠TC, this bit will drive the state of CT16Bn_PWM14 output.	R/W	0
13	EM13	When EMC13≠00b and MR13≠TC, this bit will drive the state of CT16Bn_PWM13 output.	R/W	0
12	EM12	When EMC12≠00b and MR12≠TC, this bit will drive the state of CT16Bn_PWM12 output.	R/W	0
11	EM11	When EMC11≠00b and MR11≠TC, this bit will drive the state of CT16Bn_PWM11 output.	R/W	0
10	EM10	When EMC10≠00b and MR10≠TC, this bit will drive the state of CT16Bn_PWM10 output.	R/W	0
9	EM9	When EMC9≠00b and MR9≠TC, this bit will drive the state of CT16Bn_PWM9 output.	R/W	0
8	EM8	When EMC8≠00b and MR8≠TC, this bit will drive the state of CT16Bn_PWM8 output.	R/W	0
7	EM7	When EMC7≠00b and MR7≠TC, this bit will drive the state of CT16Bn_PWM7 output.	R/W	0
6	EM6	When EMC6≠00b and MR6≠TC, this bit will drive the state of CT16Bn_PWM6 output.	R/W	0
5	EM5	When EMC5≠00b and MR5≠TC, this bit will drive the state of CT16Bn_PWM5 output.	R/W	0
4	EM4	When EMC4≠00b and MR4≠TC, this bit will drive the state of CT16Bn_PWM4 output.	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

8.7.15 CT16Bn External Match Control register (CT16Bn_EMC)(n=1)

Address Offset: 0x8C

The External Match Control register provides control of CT16Bn_PWM [23:0]. 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
31:30	EMC15[1:0]	Determines the functionality of CT16Bn_PWM15 when MR15=TC. 00: Do Nothing. 01: CT16Bn_PWM15 pin is LOW. 10: CT16Bn_PWM15 pin is HIGH. 11: Toggle CT16Bn_PWM15 pin.	R/W	0
29:28	EMC14[1:0]	Determines the functionality of CT16Bn_PWM14 when MR14=TC.	R/W	0

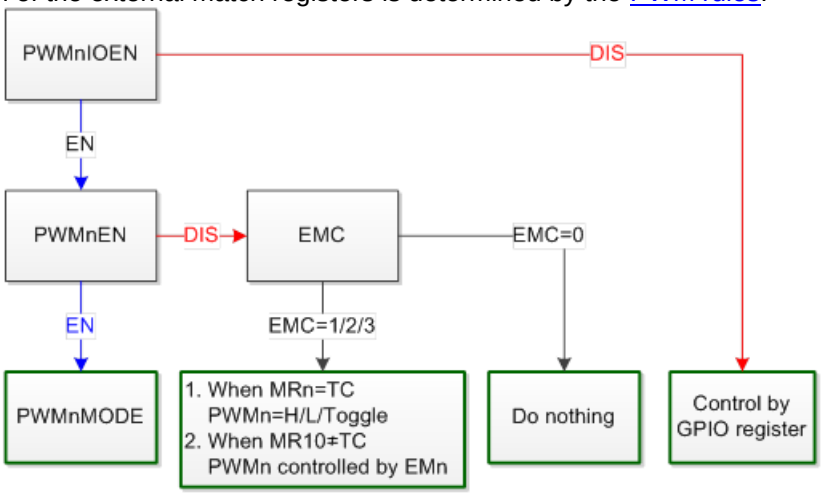
		00: Do Nothing.. 01: CT16Bn_PWM14 pin is LOW 10: CT16Bn_PWM14 pin is HIGH. 11: Toggle CT16Bn_PWM14 pin.		
27:26	EMC13[1:0]	Determines the functionality of CT16Bn_PWM13 when MR13=TC. 00: Do Nothing. 01: CT16Bn_PWM13 pin is LOW. 10: CT16Bn_PWM13 pin is HIGH. 11: Toggle CT16Bn_PWM13 pin.	R/W	0
25:24	EMC12[1:0]	Determines the functionality of CT16Bn_PWM12 when MR12=TC. 00: Do Nothing. 01: CT16Bn_PWM12 pin is LOW. 10: CT16Bn_PWM12 pin is HIGH. 11: Toggle CT16Bn_PWM12 pin.	R/W	0
23:22	EMC11[1:0]	Determines the functionality of CT16Bn_PWM11 when MR11=TC. 00: Do Nothing. 01: CT16Bn_PWM11 pin is LOW. 10: CT16Bn_PWM11 pin is HIGH. 11: Toggle CT16Bn_PWM11 pin.	R/W	0
21:20	EMC10[1:0]	Determines the functionality of CT16Bn_PWM10 when MR10=TC. 00: Do Nothing. 01: CT16Bn_PWM10 pin is LOW. 10: CT16Bn_PWM10 pin is HIGH. 11: Toggle CT16Bn_PWM10 pin.	R/W	0
19:18	EMC9[1:0]	Determines the functionality of CT16Bn_PWM9 when MR9=TC. 00: Do Nothing. 01: CT16Bn_PWM9 pin is LOW. 10: CT16Bn_PWM9 pin is HIGH. 11: Toggle CT16Bn_PWM9 pin.	R/W	0
17:16	EMC8[1:0]	Determines the functionality of CT16Bn_PWM8 when MR8=TC. 00: Do Nothing. 01: CT16Bn_PWM8 pin is LOW. 10: CT16Bn_PWM8 pin is HIGH. 11: Toggle CT16Bn_PWM8 pin.	R/W	0
15:14	EMC7[1:0]	Determines the functionality of CT16Bn_PWM7 when MR7=TC. 00: Do Nothing. 01: CT16Bn_PWM7 pin is LOW. 10: CT16Bn_PWM7 pin is HIGH. 11: Toggle CT16Bn_PWM7 pin.	R/W	0
13:12	EMC6[1:0]	Determines the functionality of CT16Bn_PWM6 when MR6=TC. 00: Do Nothing. 01: CT16Bn_PWM6 pin is LOW. 10: CT16Bn_PWM6 pin is HIGH. 11: Toggle CT16Bn_PWM6 pin.	R/W	0
11:10	EMC5[1:0]	Determines the functionality of CT16Bn_PWM5 when MR5=TC. 00: Do Nothing. 01: CT16Bn_PWM5 pin is LOW. 10: CT16Bn_PWM5 pin is HIGH. 11: Toggle CT16Bn_PWM5 pin.	R/W	0
9:8	EMC4[1:0]	Determines the functionality of CT16Bn_PWM4 when MR4=TC. 00: Do Nothing. 01: CT16Bn_PWM4 pin is LOW. 10: CT16Bn_PWM4 pin is HIGH. 11: Toggle CT16Bn_PWM4 pin.	R/W	0
7:6	EMC3[1:0]	Determines the functionality of CT16Bn_PWM3 when MR3=TC. 00: Do Nothing. 01: CT16Bn_PWM3 pin is LOW. 10: CT16Bn_PWM3 pin is HIGH. 11: Toggle CT16Bn_PWM3 pin.	R/W	0
5:4	EMC2[1:0]	Determines the functionality of CT16Bn_PWM2 when MR2=TC. 00: Do Nothing. 01: CT16Bn_PWM2 pin is LOW. 10: CT16Bn_PWM2 pin is HIGH. 11: Toggle CT16Bn_PWM2 pin.	R/W	0
3:2	EMC1[1:0]	Determines the functionality of CT16Bn_PWM1 when MR1=TC. 00: Do Nothing. 01: CT16Bn_PWM1 pin is LOW	R/W	0

		10: CT16Bn_PWM1 pin is HIGH… 11: Toggle CT16Bn_PWM1.		
1:0	EMC0[1:0]	Determines the functionality of CT16Bn_PWM0 when MR0=TC. 00: Do Nothing. 01: CT16Bn_PWM0 pin is LOW. 10: CT16Bn_PWM0 pin is HIGH. 11: Toggle CT16Bn_PWM0.	R/W	0

8.7.16 CT16Bn External Match Control register 2(CT16Bn_EMC2)(n=1)

Address Offset: 0x90

The External Match Control register 2 provides control of CT16Bn_PWM [23:0]. 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
31:16	Reserved		R	0
15:14	EMC23[1:0]	Determines the functionality of CT16Bn_PWM23 when MR23=TC. 00: Do Nothing. 01: CT16Bn_PWM23 pin is LOW. 10: CT16Bn_PWM23 pin is HIGH. 11: Toggle CT16Bn_PWM23 pin.	R/W	0
13:12	EMC22[1:0]	Determines the functionality of CT16Bn_PWM22 when MR22=TC. 00: Do Nothing. 01: CT16Bn_PWM22 pin is LOW. 10: CT16Bn_PWM22 pin is HIGH. 11: Toggle CT16Bn_PWM22 pin.	R/W	0
11:10	EMC21[1:0]	Determines the functionality of CT16Bn_PWM21 when MR21=TC. 00: Do Nothing. 01: CT16Bn_PWM21 pin is LOW. 10: CT16Bn_PWM21 pin is HIGH. 11: Toggle CT16Bn_PWM21 pin.	R/W	0
9:8	EMC20[1:0]	Determines the functionality of CT16Bn_PWM20 when MR20=TC. 00: Do Nothing. 01: CT16Bn_PWM20 pin is LOW. 10: CT16Bn_PWM20 pin is HIGH. 11: Toggle CT16Bn_PWM20 pin.	R/W	0
7:6	EMC19[1:0]	Determines the functionality of CT16Bn_PWM19 when MR19=TC. 00: Do Nothing. 01: CT16Bn_PWM19 pin is LOW. 10: CT16Bn_PWM19 pin is HIGH. 11: Toggle CT16Bn_PWM19 pin.	R/W	0
5:4	EMC18[1:0]	Determines the functionality of CT16Bn_PWM18 when MR18=TC. 00: Do Nothing. 01: CT16Bn_PWM18 pin is LOW. 10: CT16Bn_PWM18 pin is HIGH. 11: Toggle CT16Bn_PWM18 pin.	R/W	0

3:2	EMC17[1:0]	Determines the functionality of CT16Bn_PWM17 when MR17=TC. 00: Do Nothing. 01: CT16Bn_PWM17 pin is LOW. 10: CT16Bn_PWM17 pin is HIGH. 11: Toggle CT16Bn_PWM17 pin.	R/W	0
1:0	EMC16[1:0]	Determines the functionality of CT16Bn_PWM16 when MR16=TC. 00: Do Nothing. 01: CT16Bn_PWM16 pin is LOW. 10: CT16Bn_PWM16 pin is HIGH. 11: Toggle CT16Bn_PWM16 pin.	R/W	0

8.7.17 CT16Bn PWM Control register (CT16Bn_PWMCTRL) (n=1)

Address Offset: 0x94

The PWM Control register is used to configure the match outputs as PWM outputs. Each match output can be in-dependently set to perform either as PWM output or as match output whose function is controlled by CT16Bn_EM register.

For each timer, a maximum of 4 single edge controlled PWM outputs can be selected on the CT16Bn_PWMCTRL and CT16Bn_PWMCTRL2 outputs. 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	PWM15MODE[1:0]	PWM15 output. 00: PWM mode 1. PWM15 is 0 when TC<MR15 during Up-counting period. 01: PWM mode 2. PWM15 is 1 when TC<MR15 during Up-counting period. 10: PWM15 is forced to 0. 11: PWM15 is forced to 1.	R/W	0
29:28	PWM14MODE[1:0]	PWM14 output. 00: PWM mode 1. PWM14 is 0 when TC<MR14 during Up-counting period. 01: PWM mode 2. PWM14 is 1 when TC<MR14 during Up-counting period. 10: PWM14 is forced to 0. 11: PWM14 is forced to 1.	R/W	0
27:26	PWM13MODE[1:0]	PWM13 output. 00: PWM mode 1. PWM13 is 0 when TC<MR13 during Up-counting period. 01: PWM mode 2. PWM13 is 1 when TC<MR13 during Up-counting period. 10: PWM13 is forced to 0. 11: PWM13 is forced to 1.	R/W	0
25:24	PWM12MODE[1:0]	PWM12 output. 00: PWM mode 1. PWM12 is 0 when TC<MR12 during Up-counting period. 01: PWM mode 2. PWM12 is 1 when TC<MR12 during Up-counting period. 10: PWM12 is forced to 0. 11: PWM12 is forced to 1.	R/W	0
23:22	PWM11MODE[1:0]	PWM11 output. 00: PWM mode 1. PWM11 is 0 when TC<MR11 during Up-counting period. 01: PWM mode 2. PWM11 is 1 when TC<MR11 during Up-counting period. 10: PWM11 is forced to 0. 11: PWM11 is forced to 1.	R/W	0
21:20	PWM10MODE[1:0]	PWM10 output. 00: PWM mode 1.	R/W	0

		PWM10 is 0 when TC<MR10 during Up-counting period. 01: PWM mode 2. PWM10 is 1 when TC<MR10 during Up-counting period. 10: PWM10 is forced to 0. 11: PWM10 is forced to 1.		
19:18	PWM9MODE[1:0]	PWM9 output. 00: PWM mode 1. PWM9 is 0 when TC<MR9 during Up-counting period. 01: PWM mode 2. PWM9 is 1 when TC<MR9 during Up-counting period. 10: PWM9 is forced to 0. 11: PWM9 is forced to 1.	R/W	0
17:16	PWM8MODE[1:0]	PWM8 output. 00: PWM mode 1. PWM8 is 0 when TC<MR8 during Up-counting period. 01: PWM mode 2. PWM8 is 1 when TC<MR8 during Up-counting period. 10: PWM8 is forced to 0. 11: PWM8 is forced to 1.	R/W	0
15:14	PWM7MODE[1:0]	PWM7 output. 00: PWM mode 1. PWM7 is 0 when TC<MR7 during Up-counting period. 01: PWM mode 2. PWM7 is 1 when TC<MR7 during Up-counting period. 10: PWM7 is forced to 0. 11: PWM7 is forced to 1.	R/W	0
13:12	PWM6MODE[1:0]	PWM6 output. 00: PWM mode 1. PWM6 is 0 when TC<MR6 during Up-counting period. 01: PWM mode 2. PWM6 is 1 when TC<MR6 during Up-counting period. 10: PWM6 is forced to 0. 11: PWM6 is forced to 1.	R/W	0
11:10	PWM5MODE[1:0]	PWM5 output. 00: PWM mode 1. PWM5 is 0 when TC<MR5 during Up-counting period. 01: PWM mode 2. PWM5 is 1 when TC<MR5 during Up-counting period. 10: PWM5 is forced to 0. 11: PWM5 is forced to 1.	R/W	0
9:8	PWM4MODE[1:0]	PWM4output. 00: PWM mode 1. PWM4 is 0 when TC<MR4 during Up-counting period. 01: PWM mode 2. PWM4 is 1 when TC<MR4 during Up-counting period. 10: PWM4 is forced to 0. 11: PWM4 is forced to 1.	R/W	0
7:6	PWM3MODE[1:0]	PWM3 output. 00: PWM mode 1. PWM3 is 0 when TC<MR3 during Up-counting period. 01: PWM mode 2. PWM3 is 1 when TC<MR3 during Up-counting period. 10: PWM3 is forced to 0. 11: PWM3 is forced to 1.	R/W	0
5:4	PWM2MODE[1:0]	PWM2 output. 00: PWM mode 1. PWM2 is 0 when TC<MR2 during Up-counting period. 01: PWM mode 2. PWM2 is 1 when TC<MR2 during Up-counting period. 10: PWM2 is forced to 0. 11: PWM2 is forced to 1.	R/W	0
3:2	PWM1MODE[1:0]	PWM1 output. 00: PWM mode 1. PWM1 is 0 when TC<MR1 during Up-counting period. 01: PWM mode 2. PWM1 is 1 when TC<MR1 during Up-counting period. 10: PWM1 is forced to 0. 11: PWM1 is forced to 1.	R/W	0

1:0	PWM0MODE[1:0]	PWM0 output. 00: PWM mode 1. PWM0 is 0 when TC<MR0 during Up-counting period. 01: PWM mode 2. PWM0 is 1 when TC<MR0 during Up-counting period. 10: PWM0 is forced to 0. 11: PWM0 is forced to 1.	R/W	0

8.7.18 CT16Bn PWM Control register 2 (CT16Bn_PWMCTRL2) (n=1)

Address Offset: 0x98

The PWM Control register is used to configure the match outputs as PWM outputs. Each match output can be in-dependently set to perform either as PWM output or as match output whose function is controlled by CT16Bn_EM register.

For each timer, a maximum of 4 single edge controlled PWM outputs can be selected on the CT16Bn_PWMCTRL and CT16Bn_PWMCTRL2 outputs. 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
31:16	Reserved		R	0
15:14	PWM23MODE[1:0]	PWM23 output. 00: PWM mode 1. PWM23 is 0 when TC<MR23 during Up-counting period. 01: PWM mode 2. PWM23 is 1 when TC<MR23 during Up-counting period. 10: PWM23 is forced to 0. 11: PWM23 is forced to 1.	R/W	0
13:12	PWM22MODE[1:0]	PWM22 output. 00: PWM mode 1. PWM22 is 0 when TC<MR22 during Up-counting period. 01: PWM mode 2. PWM22 is 1 when TC<MR22 during Up-counting period. 10: PWM22 is forced to 0. 11: PWM22 is forced to 1.	R/W	0
11:10	PWM21MODE[1:0]	PWM21 output. 00: PWM mode 1. PWM21 is 0 when TC<MR21 during Up-counting period. 01: PWM mode 2. PWM21 is 1 when TC<MR21 during Up-counting period. 10: PWM21 is forced to 0. 11: PWM21 is forced to 1.	R/W	0
9:8	PWM20MODE[1:0]	PWM20 output. 00: PWM mode 1. PWM20 is 0 when TC<MR20 during Up-counting period. 01: PWM mode 2. PWM20 is 1 when TC<MR20 during Up-counting period. 10: PWM20 is forced to 0. 11: PWM20 is forced to 1.	R/W	0
7:6	PWM19MODE[1:0]	PWM19 output. 00: PWM mode 1. PWM19 is 0 when TC<MR19 during Up-counting period. 01: PWM mode 2. PWM19 is 1 when TC<MR19 during Up-counting period. 10: PWM19 is forced to 0. 11: PWM19 is forced to 1.	R/W	0
5:4	PWM18MODE[1:0]	PWM18 output. 00: PWM mode 1. PWM18 is 0 when TC<MR18 during Up-counting period.	R/W	0

		01: PWM mode 2. PWM18 is 1 when TC<MR18 during Up-counting period. 10: PWM18 is forced to 0. 11: PWM18 is forced to 1.		
3:2	PWM17MODE[1:0]	PWM17 output. 00: PWM mode 1. PWM17 is 0 when TC<MR17 during Up-counting period. 01: PWM mode 2. PWM17 is 1 when TC<MR17 during Up-counting period. 10: PWM17 is forced to 0. 11: PWM17 is forced to 1.	R/W	0
1:0	PWM16MODE[1:0]	PWM16 output. 00: PWM mode 1. PWM16 is 0 when TC<MR16 during Up-counting period. 01: PWM mode 2. PWM16 is 1 when TC<MR16 during Up-counting period. 10: PWM16 is forced to 0. 11: PWM16 is forced to 1.	R/W	0

8.7.19 CT16Bn PWM Enable register (CT16Bn_PWMENB) (n=1)

Address Offset: 0x9C

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 each timer, a maximum of 4 single edge controlled PWM outputs can be selected on the CT16Bn_PWMCTRL and CT16Bn_PWMCTRL2 outputs. 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	PWM23EN	PWM23 enable. 0: CT16Bn_PWM23 is controlled by EMC23. 1: PWM mode is enabled for CT16Bn_PWM23.	R/W	0
22	PWM22EN	PWM22 enable. 0: CT16Bn_PWM22 is controlled by EMC22. 1: PWM mode is enabled for CT16Bn_PWM22.	R/W	0
21	PWM21EN	PWM21 enable. 0: CT16Bn_PWM21 is controlled by EMC21. 1: PWM mode is enabled for CT16Bn_PWM21.	R/W	0
20	PWM20EN	PWM20 enable. 0: CT16Bn_PWM20 is controlled by EMC20. 1: PWM mode is enabled for CT16Bn_PWM20.	R/W	0
19	PWM19EN	PWM19 enable. 0: CT16Bn_PWM19 is controlled by EMC19. 1: PWM mode is enabled for CT16Bn_PWM19.	R/W	0
18	PWM18EN	PWM18 enable. 0: CT16Bn_PWM18 is controlled by EMC18. 1: PWM mode is enabled for CT16Bn_PWM18.	R/W	0
17	PWM17EN	PWM17 enable. 0: CT16Bn_PWM17 is controlled by EMC17. 1: PWM mode is enabled for CT16Bn_PWM17.	R/W	0
16	PWM16EN	PWM16 enable. 0: CT16Bn_PWM16 is controlled by EMC16. 1: PWM mode is enabled for CT16Bn_PWM16.	R/W	0
15	PWM15EN	PWM15 enable. 0: CT16Bn_PWM15 is controlled by EMC15. 1: PWM mode is enabled for CT16Bn_PWM15.	R/W	0
14	PWM14EN	PWM14 enable. 0: CT16Bn_PWM14 is controlled by EMC14.	R/W	0

		1: PWM mode is enabled for CT16Bn_PWM14.		
13	PWM13EN	PWM13 enable. 0: CT16Bn_PWM13 is controlled by EMC13. 1: PWM mode is enabled for CT16Bn_PWM13.	R/W	0
12	PWM12EN	PWM12 enable. 0: CT16Bn_PWM12 is controlled by EMC12. 1: PWM mode is enabled for CT16Bn_PWM12.	R/W	0
11	PWM11EN	PWM11 enable. 0: CT16Bn_PWM11 is controlled by EMC11. 1: PWM mode is enabled for CT16Bn_PWM11.	R/W	0
10	PWM10EN	PWM10 enable. 0: CT16Bn_PWM10 is controlled by EMC10. 1: PWM mode is enabled for CT16Bn_PWM10.	R/W	0
9	PWM9EN	PWM9 enable. 0: CT16Bn_PWM9 is controlled by EMC9. 1: PWM mode is enabled for CT16Bn_PWM9.	R/W	0
8	PWM8EN	PWM8 enable. 0: CT16Bn_PWM8 is controlled by EMC8. 1: PWM mode is enabled for CT16Bn_PWM8.	R/W	0
7	PWM7EN	PWM7 enable. 0: CT16Bn_PWM7 is controlled by EMC7. 1: PWM mode is enabled for CT16Bn_PWM7.	R/W	0
6	PWM6EN	PWM6 enable. 0: CT16Bn_PWM6 is controlled by EMC6. 1: PWM mode is enabled for CT16Bn_PWM6.	R/W	0
5	PWM5EN	PWM5 enable. 0: CT16Bn_PWM5 is controlled by EMC5. 1: PWM mode is enabled for CT16Bn_PWM5.	R/W	0
4	PWM4EN	PWM4 enable. 0: CT16Bn_PWM4 is controlled by EMC4. 1: PWM mode is enabled for CT16Bn_PWM4.	R/W	0
3	PWM3EN	PWM3 enable. 0: CT16Bn_PWM3 is controlled by EMC3. 1: PWM mode is enabled for CT16Bn_PWM3.	R/W	0
2	PWM2EN	PWM2 enable. 0: CT16Bn_PWM2 is controlled by EMC2. 1: PWM mode is enabled for CT16Bn_PWM2.	R/W	0
1	PWM1EN	PWM1 enable. 0: CT16Bn_PWM1 is controlled by EMC1. 1: PWM mode is enabled for CT16Bn_PWM1.	R/W	0
0	PWM0EN	PWM0 enable. 0: CT16Bn_PWM0 is controlled by EMC0. 1: PWM mode is enabled for CT16Bn_PWM0.	R/W	0

8.7.20 PWM IO Enable register (CT16Bn_PWMIOENB) (n=1)

Address Offset: 0xA0

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 each timer, a maximum of 4 single edge controlled PWM outputs can be selected on the CT16Bn_PWMCTRL and CT16Bn_PWMCTRL2 outputs. 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	PWM23IOEN	CT16Bn_PWM23/GPIO selection bit. 0: CT16Bn_PWM23 pin act as GPIO. 1: CT16Bn_PWM23 pin act as match output, and output signal depends	R/W	0

		on PWM23EN bit.		
22	PWM22IOEN	CT16Bn_PWM22/GPIO selection bit. 0: CT16Bn_PWM22 pin act as GPIO. 1: CT16Bn_PWM22 pin act as match output, and output signal depends on PWM22EN bit.	R/W	0
21	PWM21IOEN	CT16Bn_PWM21/GPIO selection bit. 0: CT16Bn_PWM21 pin act as GPIO. 1: CT16Bn_PWM21 pin act as match output, and output signal depends on PWM21EN bit.	R/W	0
20	PWM20IOEN	CT16Bn_PWM20/GPIO selection bit. 0: CT16Bn_PWM20 pin act as GPIO. 1: CT16Bn_PWM20 pin act as match output, and output signal depends on PWM20EN bit.	R/W	0
19	PWM19IOEN	CT16Bn_PWM19/GPIO selection bit. 0: CT16Bn_PWM19 pin act as GPIO. 1: CT16Bn_PWM19 pin act as match output, and output signal depends on PWM19EN bit.	R/W	0
18	PWM18IOEN	CT16Bn_PWM18/GPIO selection bit. 0: CT16Bn_PWM18 pin act as GPIO. 1: CT16Bn_PWM18 pin act as match output, and output signal depends on PWM18EN bit.	R/W	0
17	PWM17IOEN	CT16Bn_PWM17/GPIO selection bit. 0: CT16Bn_PWM17 pin act as GPIO. 1: CT16Bn_PWM17 pin act as match output, and output signal depends on PWM17EN bit.	R/W	0
16	PWM16IOEN	CT16Bn_PWM16/GPIO selection bit. 0: CT16Bn_PWM16 pin act as GPIO. 1: CT16Bn_PWM16 pin act as match output, and output signal depends on PWM16EN bit.	R/W	0
15	PWM15IOEN	CT16Bn_PWM15/GPIO selection bit. 0: CT16Bn_PWM15 pin act as GPIO. 1: CT16Bn_PWM15 pin act as match output, and output signal depends on PWM15EN bit.	R/W	0
14	PWM14IOEN	CT16Bn_PWM14/GPIO selection bit. 0: CT16Bn_PWM14 pin act as GPIO. 1: CT16Bn_PWM14 pin act as match output, and output signal depends on PWM14EN bit.	R/W	0
13	PWM13IOEN	CT16Bn_PWM13/GPIO selection bit. 0: CT16Bn_PWM13 pin act as GPIO. 1: CT16Bn_PWM13 pin act as match output, and output signal depends on PWM13EN bit.	R/W	0
12	PWM12IOEN	CT16Bn_PWM12/GPIO selection bit. 0: CT16Bn_PWM12 pin act as GPIO. 1: CT16Bn_PWM12 pin act as match output, and output signal depends on PWM12EN bit.	R/W	0
11	PWM11IOEN	CT16Bn_PWM11/GPIO selection bit. 0: CT16Bn_PWM11 pin act as GPIO. 1: CT16Bn_PWM11 pin act as match output, and output signal depends on PWM11EN bit.	R/W	0
10	PWM10IOEN	CT16Bn_PWM10/GPIO selection bit. 0: CT16Bn_PWM10 pin act as GPIO. 1: CT16Bn_PWM10 pin act as match output, and output signal depends on PWM10EN bit.	R/W	0
9	PWM9IOEN	CT16Bn_PWM9/GPIO selection bit. 0: CT16Bn_PWM9 pin act as GPIO. 1: CT16Bn_PWM9 pin act as match output, and output signal depends on PWM9EN bit.	R/W	0
8	PWM8IOEN	CT16Bn_PWM8/GPIO selection bit. 0: CT16Bn_PWM8 pin act as GPIO. 1: CT16Bn_PWM8 pin act as match output, and output signal depends on PWM8EN bit.	R/W	0
7	PWM7IOEN	CT16Bn_PWM7/GPIO selection bit. 0: CT16Bn_PWM7 pin act as GPIO. 1: CT16Bn_PWM7 pin act as match output, and output signal depends on PWM7EN bit.	R/W	0
6	PWM6IOEN	CT16Bn_PWM6/GPIO selection bit. 0: CT16Bn_PWM6 pin act as GPIO.	R/W	0

		1: CT16Bn_PWM6 pin act as match output, and output signal depends on PWM6EN bit.		
5	PWM5IOEN	CT16Bn_PWM5/GPIO selection bit. 0: CT16Bn_PWM5 pin act as GPIO. 1: CT16Bn_PWM5 pin act as match output, and output signal depends on PWM5EN bit.	R/W	0
4	PWM4IOEN	CT16Bn_PWM4/GPIO selection bit. 0: CT16Bn_PWM4 pin act as GPIO. 1: CT16Bn_PWM4 pin act as match output, and output signal depends on PWM4EN bit.	R/W	0
3	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
2	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
1	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
0	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

8.7.21 CT16Bn Timer Raw Interrupt Status register (CT16Bn_RIS) (n=0)

Address Offset: 0xA4

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
31:26	Reserved		R	0
25	CAP0IF	Interrupt flag for capture channel 0. 0: No interrupt on CAP0. 1: Interrupt requirements met on CAP0.	R	0
24:1	Reserved		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

8.7.22 CT16Bn Timer Raw Interrupt Status register (CT16Bn_RIS) (n=1)

Address Offset: 0xA4

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
31:25	Reserved		R	0
24	MR24IF	Interrupt flag for match channel 24. 0: No interrupt on match channel 24. 1: Interrupt requirements met on match channel 24.	R	0
23	MR23IF	Interrupt flag for match channel 23. 0: No interrupt on match channel 23. 1: Interrupt requirements met on match channel 23.	R	0

22	MR22IF	Interrupt flag for match channel 22. 0: No interrupt on match channel 22. 1: Interrupt requirements met on match channel 22.	R	0
21	MR21IF	Interrupt flag for match channel 21. 0: No interrupt on match channel 21. 1: Interrupt requirements met on match channel 21.	R	0
20	MR20IF	Interrupt flag for match channel 20. 0: No interrupt on match channel 20. 1: Interrupt requirements met on match channel 20.	R	0
19	MR19IF	Interrupt flag for match channel 19. 0: No interrupt on match channel 19. 1: Interrupt requirements met on match channel 19.	R	0
18	MR18IF	Interrupt flag for match channel 18. 0: No interrupt on match channel 18. 1: Interrupt requirements met on match channel 18.	R	0
17	MR17IF	Interrupt flag for match channel 17. 0: No interrupt on match channel 17 1: Interrupt requirements met on match channel 17.	R	0
16	MR16IF	Interrupt flag for match channel 16. 0: No interrupt on match channel 16. 1: Interrupt requirements met on match channel 16.	R	0
15	MR15IF	Interrupt flag for match channel 15. 0: No interrupt on match channel 15. 1: Interrupt requirements met on match channel 15.	R	0
14	MR14IF	Interrupt flag for match channel 14. 0: No interrupt on match channel 14. 1: Interrupt requirements met on match channel 14.	R	0
13	MR13IF	Interrupt flag for match channel 13. 0: No interrupt on match channel 13. 1: Interrupt requirements met on match channel 13.	R	0
12	MR12IF	Interrupt flag for match channel 12. 0: No interrupt on match channel 12. 1: Interrupt requirements met on match channel 12.	R	0
11	MR11IF	Interrupt flag for match channel 11. 0: No interrupt on match channel 11. 1: Interrupt requirements met on match channel 11.	R	0
10	MR10IF	Interrupt flag for match channel 10. 0: No interrupt on match channel 10. 1: Interrupt requirements met on match channel 10.	R	0
9	MR9IF	Interrupt flag for match channel 9. 0: No interrupt on match channel 9. 1: Interrupt requirements met on match channel 9.	R	0
8	MR8IF	Interrupt flag for match channel 8. 0: No interrupt on match channel 8. 1: Interrupt requirements met on match channel 8.	R	0
7	MR7IF	Interrupt flag for match channel 7. 0: No interrupt on match channel 7. 1: Interrupt requirements met on match channel 7.	R	0
6	MR6IF	Interrupt flag for match channel 6. 0: No interrupt on match channel 6. 1: Interrupt requirements met on match channel 6.	R	0
5	MR5IF	Interrupt flag for match channel 5. 0: No interrupt on match channel 5. 1: Interrupt requirements met on match channel 5.	R	0
4	MR4IF	Interrupt flag for match channel 4. 0: No interrupt on match channel 4. 1: Interrupt requirements met on match channel 4.	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

1	MR0IF	Interrupt flag for match channel 0. 0: No interrupt on match channel 0. 1: Interrupt requirements met on match channel 0.	R	0

8.7.23 CT16Bn Timer Interrupt Clear register (CT16Bn_IC) (n=0)

Address Offset: 0xA8

Bit	Name	Description	Attribute	Reset
31:26	Reserved		R	0
25	CAP0IC	0: No effect. 1: Clear CAP0IF bit.	W	0
24:1	Reserved		R	0
0	MR0IC	0: No effect. 1: Clear MR0IF bit.	W	0

8.7.24 CT16Bn Timer Interrupt Clear register (CT16Bn_IC) (n=1)

Address Offset: 0xA8

Bit	Name	Description	Attribute	Reset
31:25	Reserved		R	0
24	MR24IC	0: No effect. 1: Clear MR24IF bit.	W	0
23	MR23IC	0: No effect. 1: Clear MR23IF bit.	W	0
22	MR22IC	0: No effect. 1: Clear MR22IF bit.	W	0
21	MR21IC	0: No effect. 1: Clear MR21IF bit.	W	0
20	MR20IC	0: No effect. 1: Clear MR20IF bit.	W	0
19	MR19IC	0: No effect. 1: Clear MR19IF bit.	W	0
18	MR18IC	0: No effect. 1: Clear MR18IF bit.	W	0
17	MR17IC	0: No effect. 1: Clear MR17IF bit.	W	0
16	MR16IC	0: No effect. 1: Clear MR16IF bit.	W	0
15	MR15IC	0: No effect. 1: Clear MR15IF bit.	W	0
14	MR14IC	0: No effect. 1: Clear MR14IF bit.	W	0
13	MR13IC	0: No effect. 1: Clear MR13IF bit.	W	0
12	MR12IC	0: No effect. 1: Clear MR12IF bit.	W	0
11	MR11IC	0: No effect. 1: Clear MR11IF bit.	W	0
10	MR10IC	0: No effect. 1: Clear MR10IF bit.	W	0
9	MR9IC	0: No effect. 1: Clear MR9IF bit.	W	0
8	MR8IC	0: No effect. 1: Clear MR8IF bit.	W	0
7	MR7IC	0: No effect. 1: Clear MR7IF bit.	W	0
6	MR6IC	0: No effect. 1: Clear MR6IF bit.	W	0

5	MR5IC	0: No effect. 1: Clear MR5IF bit.	W	0
4	MR4IC	0: No effect. 1: Clear MR4IF 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

9 WATCHDOG TIMER (WDT)

9.1 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} × 128 × 1) and the maximum Watchdog interval is (T_{WDT_PCLK} × 128 × 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_APBCP1) 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.

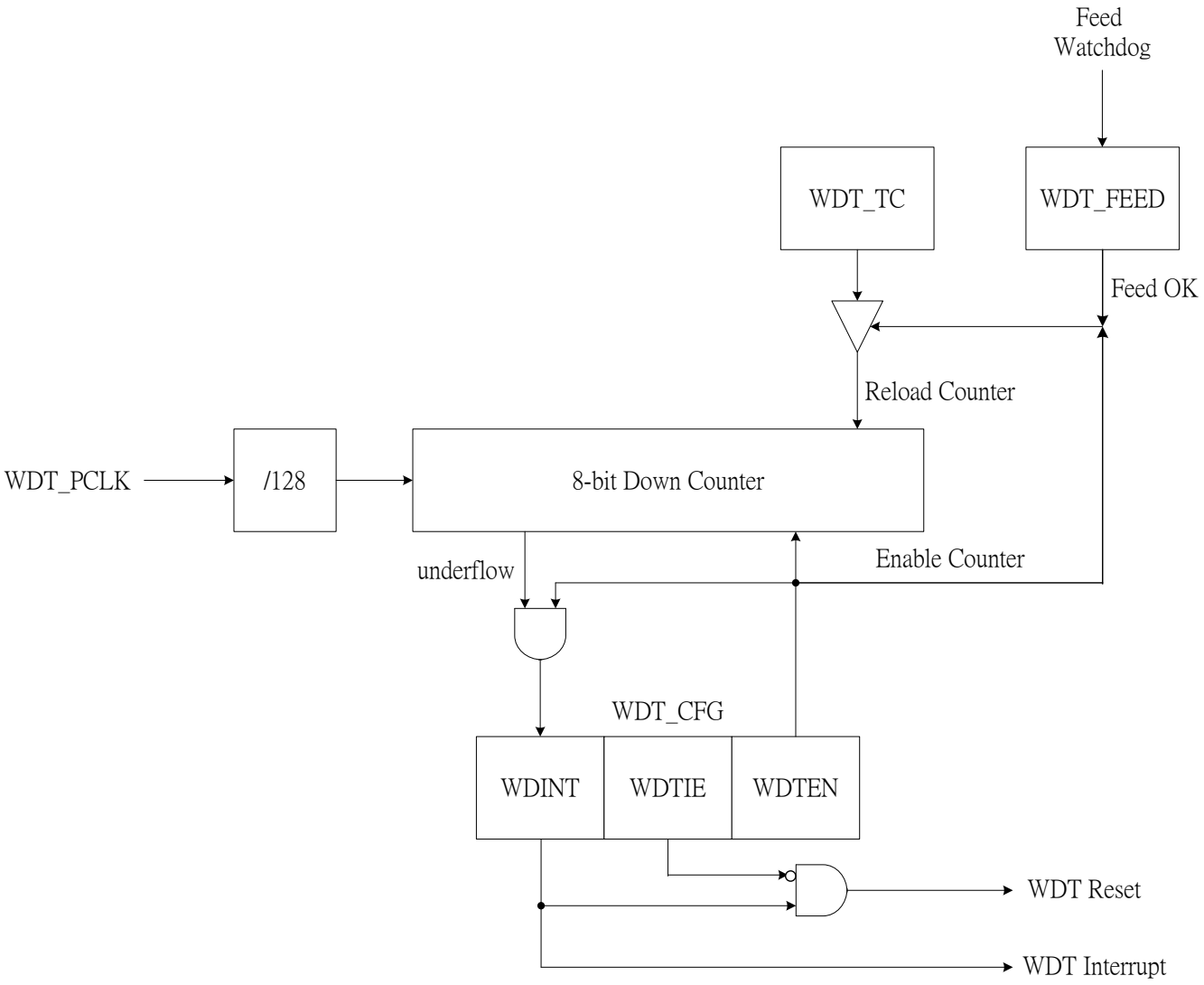
When the watchdog is started by setting the WDTEN in WDT_CFG register, the time constant value is loaded in the watchdog counter and the counter starts counting down. When the Watchdog is in the reset mode and the counter underflows, the CPU will be reset, loading the stack pointer and program counter 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 one clock: WDT_PCLK. The WDT_PCLK is used for the watchdog timer counting. Several clocks can be used as a clock source for WDT_PCLK clock: ILRC.

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.

9.2 BLOCK DIAGRAM



9.3 WDT REGISTERS

Base Address: 0x4001 0000

9.3.1 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 WDTINT 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	WDTINT	Watchdog interrupt flag. <Read> 0: Watchdog does not cause an interrupt. 1: Watchdog time-out and causes an interrupt (Only when WDTIE =1). <Write> 0: Clear this flag. **SW shall feed Watchdog before clearing.**	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 WDTINT 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

9.3.2 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} × 128 x 1 ~ T_{WDT_PCLK} × 128 x 256.

Watchdog overflow time = (31.25us x 1) x 128 x 1 ~ (31.25us x 32) x 128 x 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

- **Watchdog clock source is fixed as ILRC.**

9.3.3 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

10 SPI

10.1 OVERVIEW

The SPI controller 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.

10.2 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 24 Mbps (master) or 6 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.

10.3 PIN DESCRIPTION

Pin Name	Type	Description	GPIO Configuration
SCKn	O	SPI Serial clock (Master)	
	I	SPI Serial clock (Slave)	Depends on GPIOn_CFG
SELn	O	SPI Slave Select (Master)	
	I	SPI Slave Select (Slave)	Depends on GPIOn_CFG
MISOn	I	Master In Slave Out (Master)	Depends on GPIOn_CFG
	O	Master In Slave Out (Slave)	
MOSIn	O	Master Out Slave In (Master)	
	I	Master Out Slave In (Slave)	Depends on GPIOn_CFG

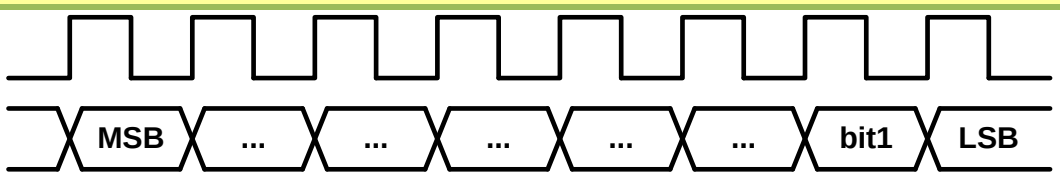
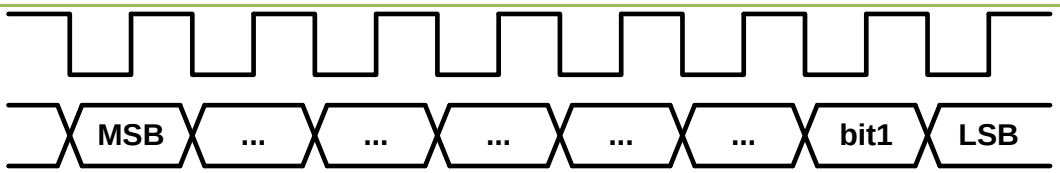
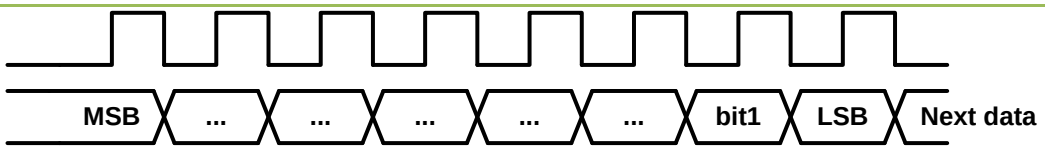
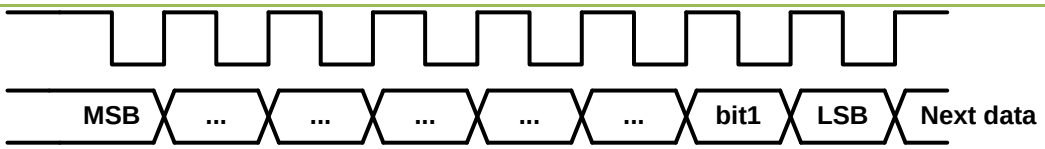
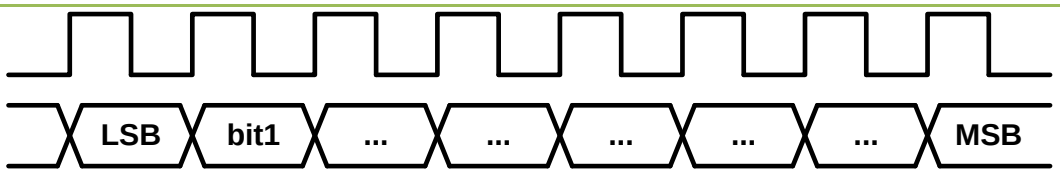
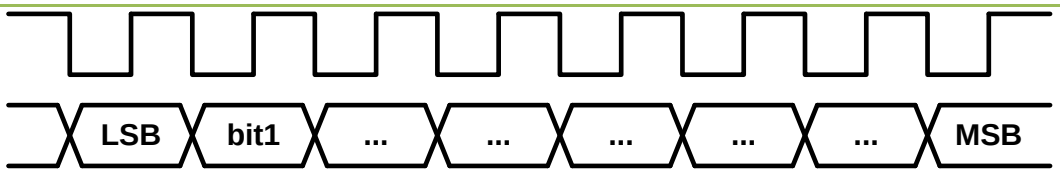
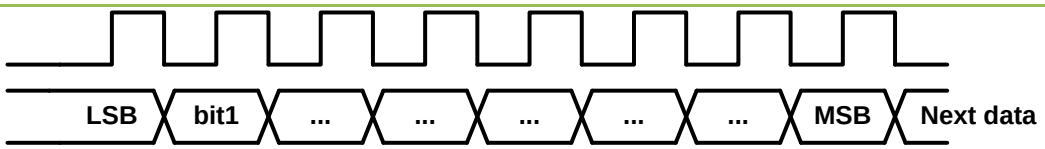
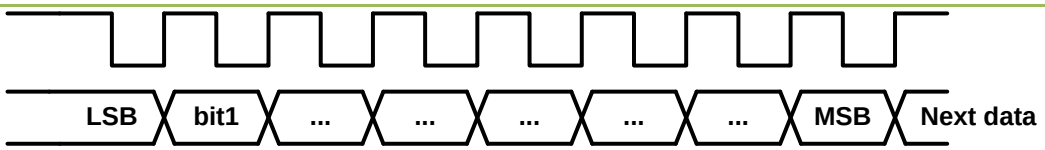
10.4 INTERFACE DESCRIPTION

10.4.1 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.

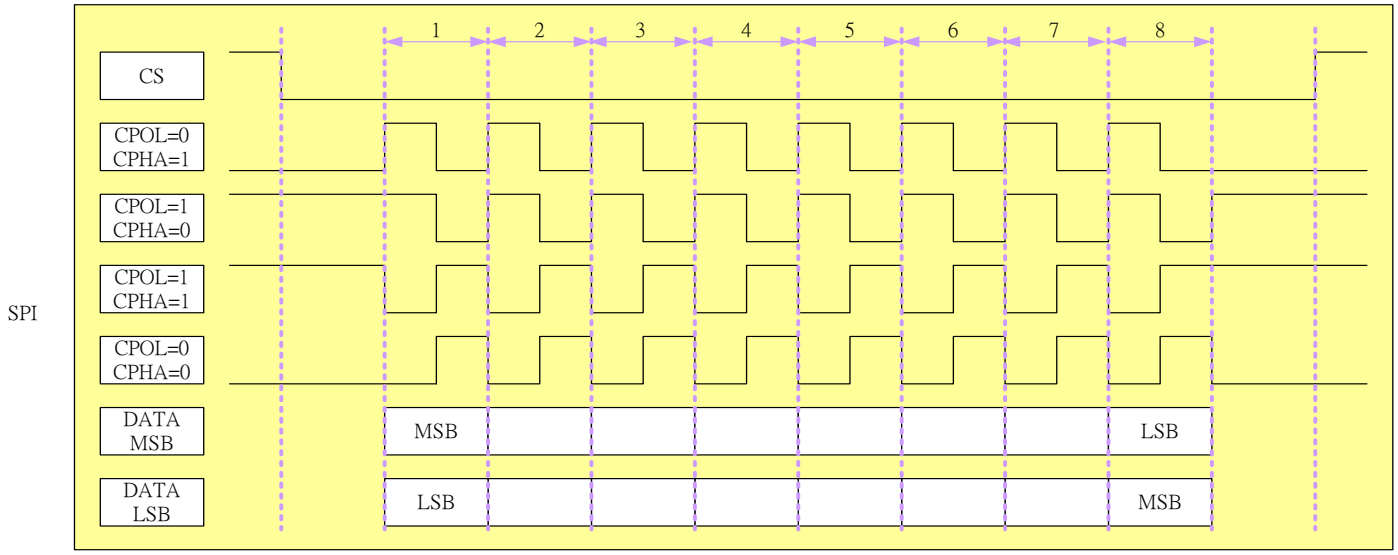
When the “CPOL” clock polarity control bit is LOW, it produces a steady state low value on the SCK pin. If the CPOL clock polarity control bit is HIGH, a steady state high value is placed on the CLK pin when data is not being transferred. The “CPHA” clock phase bit controls the phase of the clock on 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:

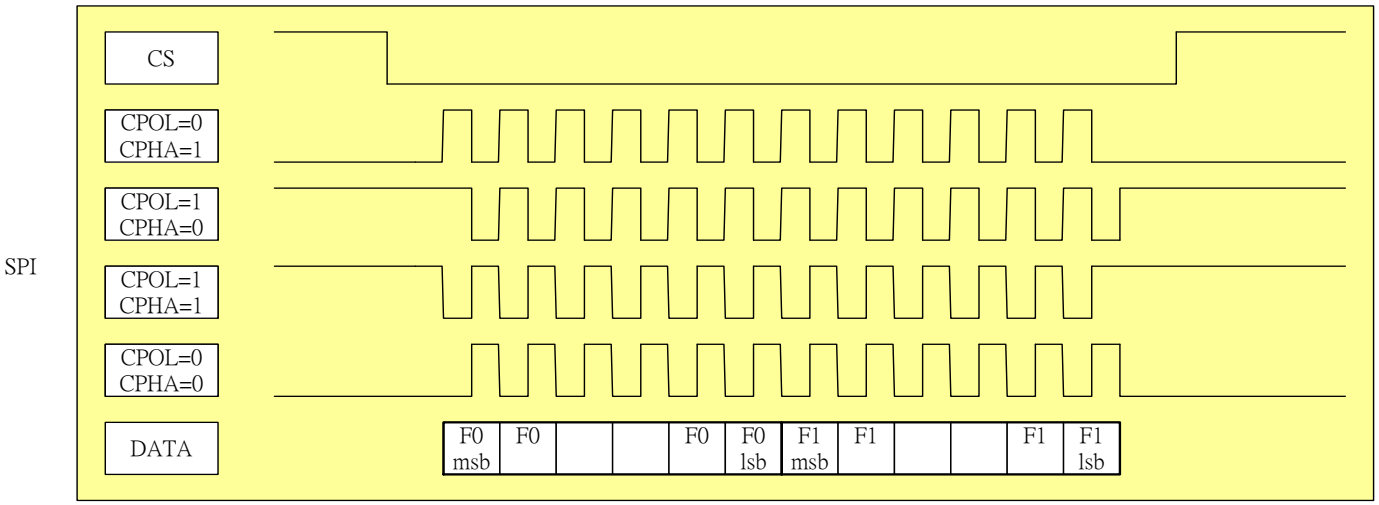
MLSB	CPOL	CPHA	SCK Idle Status	Diagrams
0	0	1	Low	
0	1	1	High	
0	0	0	Low	
0	1	0	High	
1	0	1	Low	
1	1	1	High	
1	0	0	Low	
1	1	0	High	

10.4.2 COMMUNICATION FLOW

10.4.2.1 SINGLE-FRAME



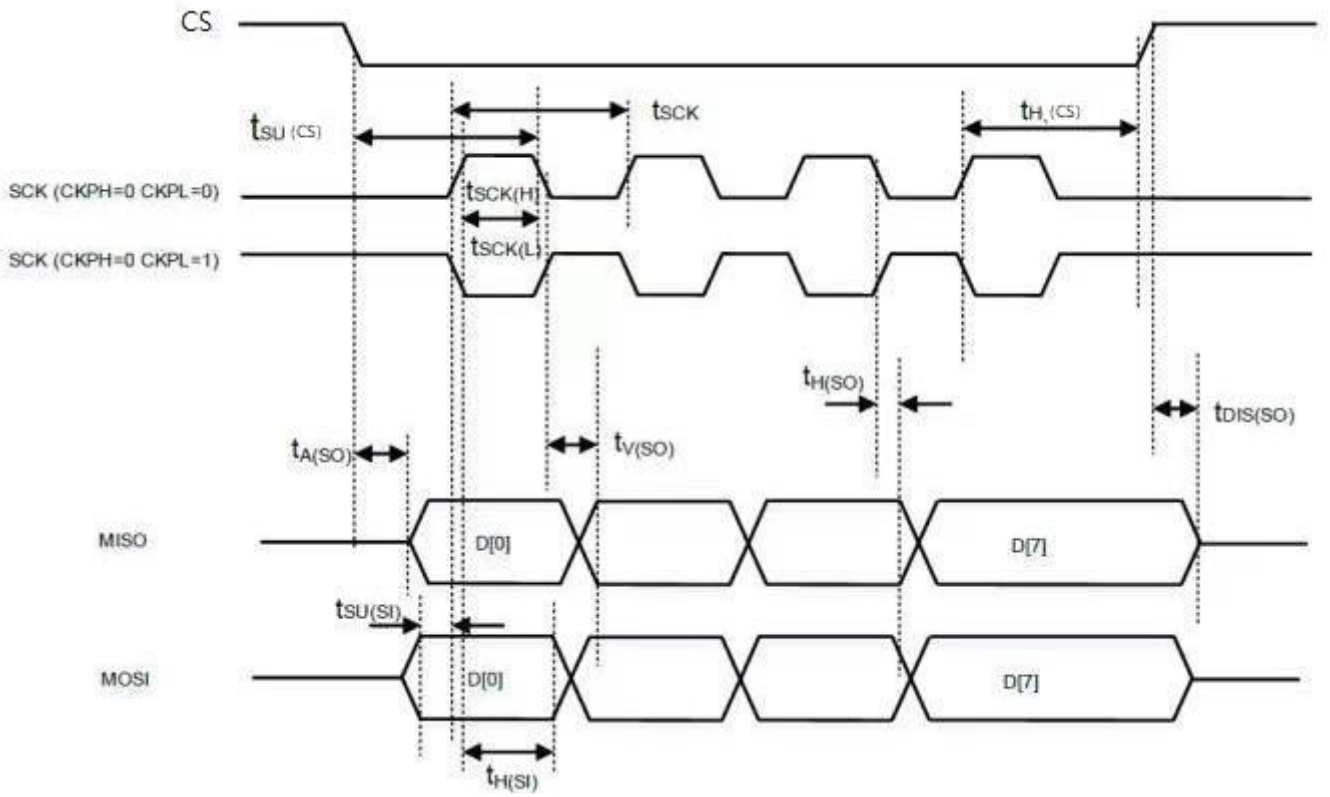
10.4.2.2 MULTI-FRAME



10.5 AUTO-SEL

The Auto-SEL function is disabled by default, and Auto-SEL data flow is controlled by HW if enabled.
If Auto-SEL function is disabled (SELDIS = 1), HW does NOT control SELn pin at all, SELn pin is GPIO.
If Auto-SEL function is enabled (SELDIS = 0), SPI HW controls the SELn activity.

10.6 TIMING CHARACTERISTICS



10.6.1 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

10.6.2 SLAVE MODE

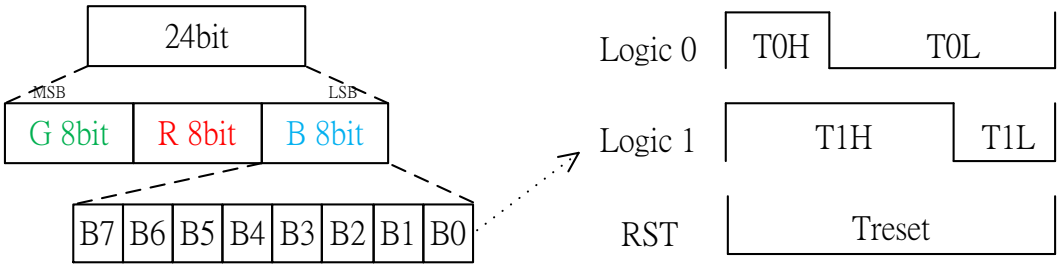
Symbol	Parameter	Min	Typ	Max	Unit
t_{sck}	SCK in period	33			ns
$t_{sck}(H)$	SCK in high pulse	t_{sck}*0.4			ns
$t_{sck}(L)$	SCK in low pulse	t_{sck}*0.4			ns
$t_{SU}(CS)$	CS in setup time	t_{sck}*1.5			ns
$t_H(CS)$	CS in hold time	t_{sck}			ns
$t_{SU}(MISO)$	Data out setup time	t_{sck}*0.4			ns
$t_H(MISO)$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

10.7 AUTO TRANSMIT(SPI1)

10.7.1.1 FUNCTION BLOCK

10.7.1.2 DATA TRANSFER TIME

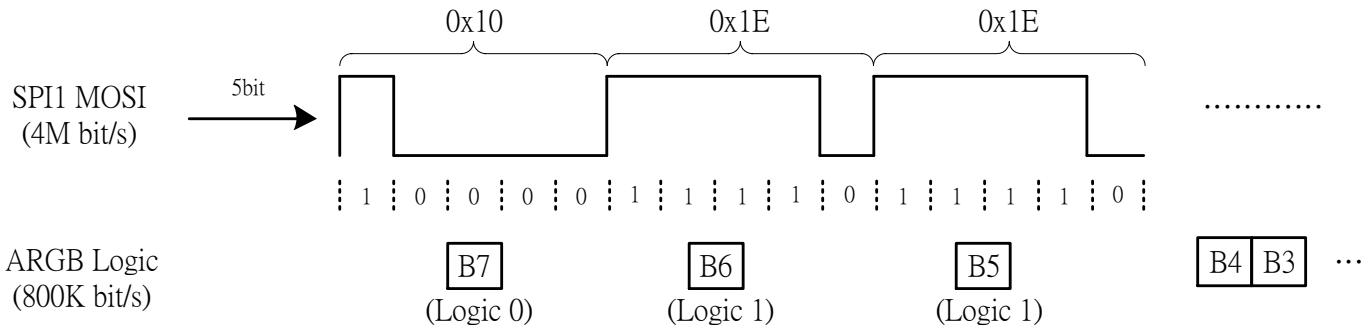
The single wire data transfer protocol supports 24-bit data for each LED RGB display data refresh. The IC receives 24-bit data and passes the remaining data to next LED. The 24-bit data consist of Green, Red and Blue data, each with 8-bit width, and are transferred with MSB first.



The transferred data are recognized based on the pulse widths received by the IC. A low bit 0 is represented by a 0.3us high pulse followed by a 0.9us low pulse. A high bit 1 is represented by a 0.9us high pulse followed by a 0.3us low pulse. A low pulse ≧ 200us is used to issue a reset command to the IC to start a new cycle of serial commands.

Name	Description	Time
Logic 0	T0H	0.3us+/-0.15us
	T0L	0.9us+/-0.15us
Logic 1	T1H	0.9us+/-0.15us
	T1L	0.3us+/-0.15us
RST	Treset	≧200us

10.7.1.3 AUTO TRANSMIT MOSI WAVEFORM



10.8 SPI REGISTERS

Base Address: 0x4001 C000 (SPI0)
0x4005 8000 (SPI1)

10.8.1 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. For SPI mode only. 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.	R/W	0

		1: Enable and HW switches I/O configurations refer to FORMAT bit directly.		

10.8.2 SPI n Control register 0 (SPIn_CTRL0) (n=1)

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
31:20	Reserved		R	0
19	ATEN	Auto Transmit enable bit 0: Disable. 1: Enable SPI1 Auto Transmit(SPI SRAM to SPI_TX Enable).	R/W	0
18	SELDIS	Auto-SEL disable bit. For SPI mode only. 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. Only set 3~8 data length in the Auto Transmit 0000~0001: Reversed. 0010: data length = 3. … 0111: data length = 8. 1000: data length = 9(Can't be used in Auto Transmit). … 1110: data length = 15(Can't be used in Auto Transmit). 1111: data length = 16(Can't be used in Auto Transmit).	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 and HW switches I/O configurations refer to FORMAT bit directly.	R/W	0

10.8.3 SPI n Control register 1 (SPIn_CTRL1) (n=0,1)

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

10.8.4 SPI n Clock Divider register (SPIn_CLKDIV) (n=0,1)

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

10.8.5 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. 1: TX FIFO is empty.	R	1

10.8.6 SPI n Status register (SPIn_STAT) (n=1)

Address Offset: 0x0C

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7	AT_STATUS	Auto Transmit STATUS H/W flag. 0: Idle status. 1: Busy status.	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. 1: TX FIFO is empty.	R	1

10.8.7 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

10.8.8 SPI n Interrupt Enable register (SPIn_IE) (n=1)

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
31:5	Reserved		R	0
4	ATIE	Auto Transmit interrupt enable. 0: Disable. 1: Enable.	R/W	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

10.8.9 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

10.8.10 SPI n Raw Interrupt Status register (SPIn_RIS) (n=1)

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
31:5	Reserved		R	0
4	ATIF	Auto Transmit interrupt flag. 0: Auto Transmit interrupt. 1: Auto Transmit triggered.	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

10.8.11 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

10.8.12 SPI n Interrupt Clear register (SPIn_IC) (n=1)

Address Offset: 0x18

Bit	Name	Description	Attribute	Reset
31:5	Reserved		R	0
4	ATIC	0: No effect. 1: Clear ATIF bit.	W	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

10.8.13 SPI n Data register (SPIn_DATA) (n=0,1)

Address Offset: 0x1C

Bit	Name	Description	Attribute	Reset
31:16	Reserved		R	0
15:0	DATA[15:0]	Write 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). Read 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

10.8.14 SPI n Data Fetch register (SPIn_DFDLY) (n=0,1)

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 is higher than 6MHz.	R/W	0

10.8.15 SPI n Read/Write Address register (SPIn_RWADDR) (n=1)

Address Offset: 0x24

Bit	Name	Description	Attribute	Reset
31:10	Reserved		R	0
9:0	RWADDR[9:0]	SPI1 SRAM address to be read or written from/to SPI1 SRAM. RWADDR[1:0] is fixed to 00b (RWADDR is a multiple of 4)	R/W	0

10.8.16 SPI n Write Data register (SPIn_RWDATA) (n=1)

Address Offset: 0x28

Bit	Name	Description	Attribute	Reset
31:0	RWDATA[31:0]	Data to be read or written from/to SPI1 SRAM.	R/W	0

10.8.17 SPI n Number of Data Transfer register (SPIn_CNT) (n=1)

Address Offset: 0x2C

This register can only be written when the Auto Transmit is disabled. Once the Auto Transmit is enabled, this register is read-only, indicating the remaining bytes to be transmitted.
Once the transfer is completed, and trigger ATIF interrupt if ATIE = 1.

Bit	Name	Description	Attribute	Reset
31:10	Reserved		R	0
9:0	CNT[9:0]	Number of times (CNT+1) to be transferred.	R/W	0

11 I2C

11.1 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.

Depending on the state of the direction bit (R/W), two types of data transfers are possible on the I2C bus:

- Data transfer from a master transmitter to a slave receiver.
  The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte.
- Data transfer from a slave transmitter to a master receiver.
  The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. Next follows the data bytes transmitted by the slave to the master. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a “not acknowledge” is returned. The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a Repeated START condition. Since a Repeated START condition is also the beginning of the next serial transfer, the I2C bus will not be released.

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

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

11.2 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.
- Monitor mode allows observing all I2C-bus traffic, regardless of slave address.

- I2C-bus can be used for test and diagnostic purposes.
- Generation and detection of 7-bit/10-bit addressing and General Call.

11.3 PIN DESCRIPTION

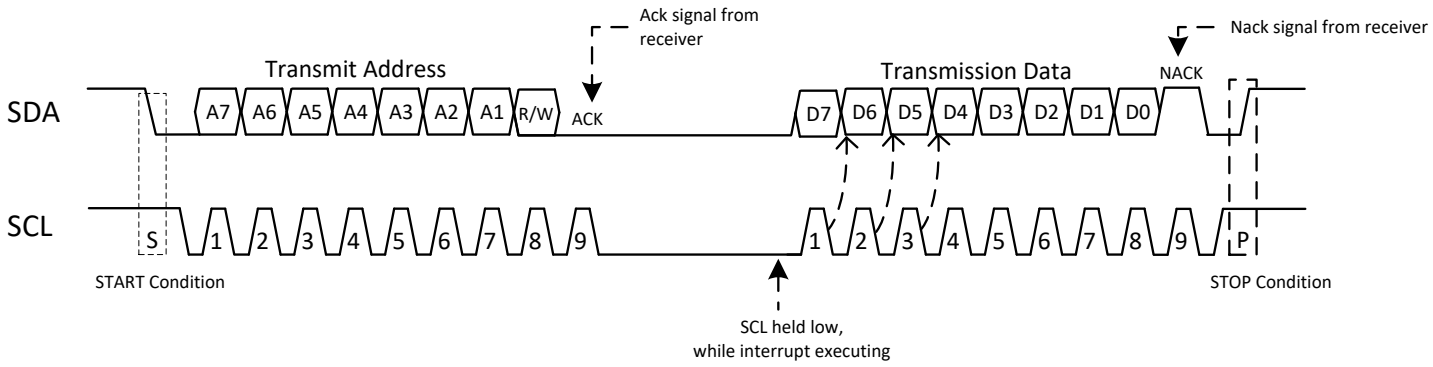
Pin Name	Type	Description	GPIO Configuration
SCLn	I/O	I2C Serial clock	Output with Open-drain Input depends on GPIOn_CFG
SDAn	I/O	I2C Serial data	Output with Open-drain Input depends on GPIOn_CFG

11.4 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.

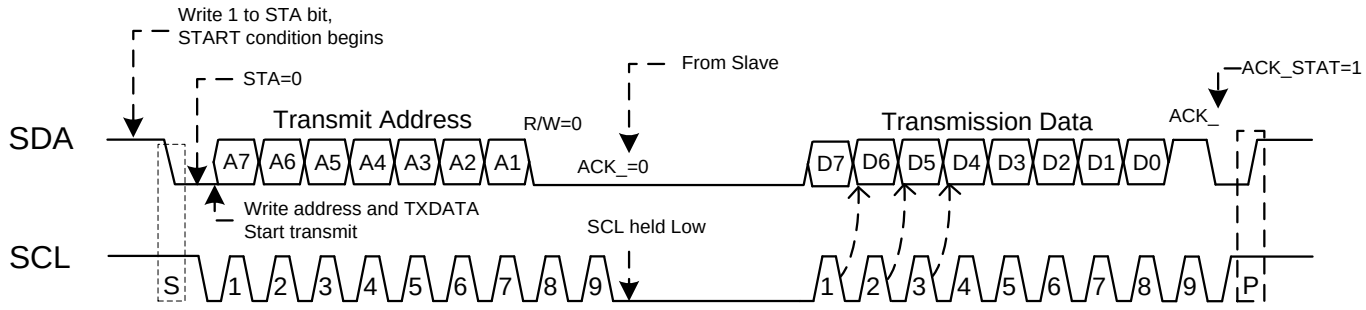
Data is transmitted with the Most Significant Bit (MSB) first. In address byte, the higher 7-bit is address bit and the lowest bit is data direction (R/W) bit. When R/W=0, it assigns a “WRITE” operation. When R/W=1, it assigns a “READ” operation.

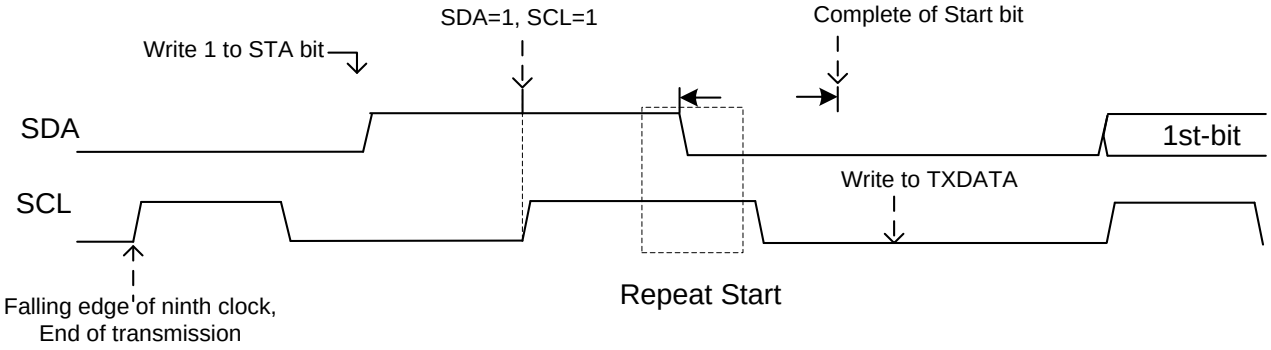
After each byte is received, the receiver (a master or a slave) must send an acknowledge (ACK) bit. If transmitter can’t receive an ACK, it will recognize a not acknowledge (NACK). In WRITE operation, the master will transmit data to the slave and then waits for ACK from slave. In READ operation, the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



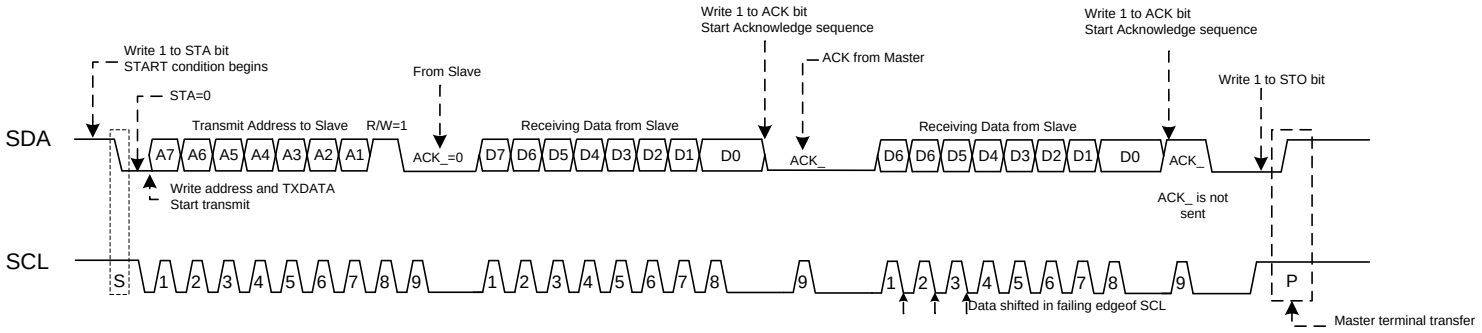
11.4.1 7-BIT ADDRESSING MODES

11.4.1.1 MASTER TRANSMITTER MODE

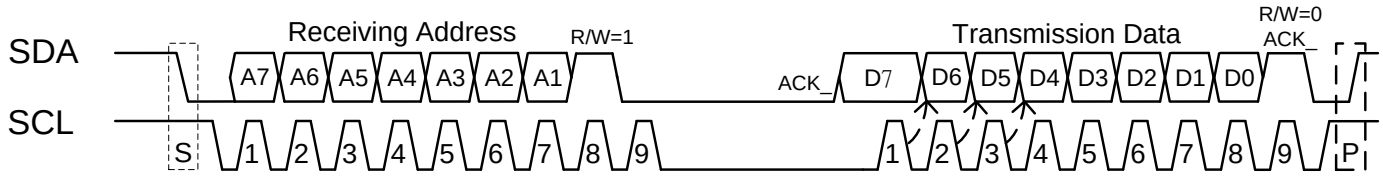




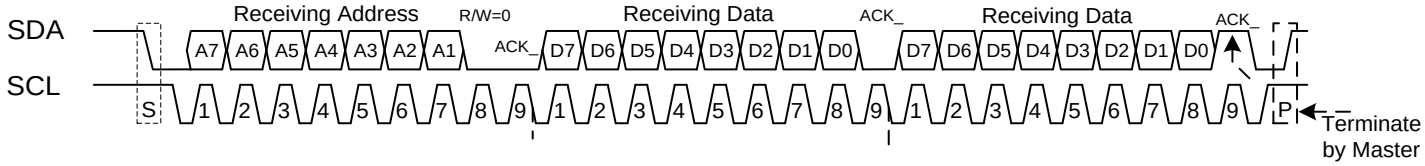
11.4.1.2 MASTER RECEIVER MODE



11.4.1.3 SLAVE TRANSMITTER MODE

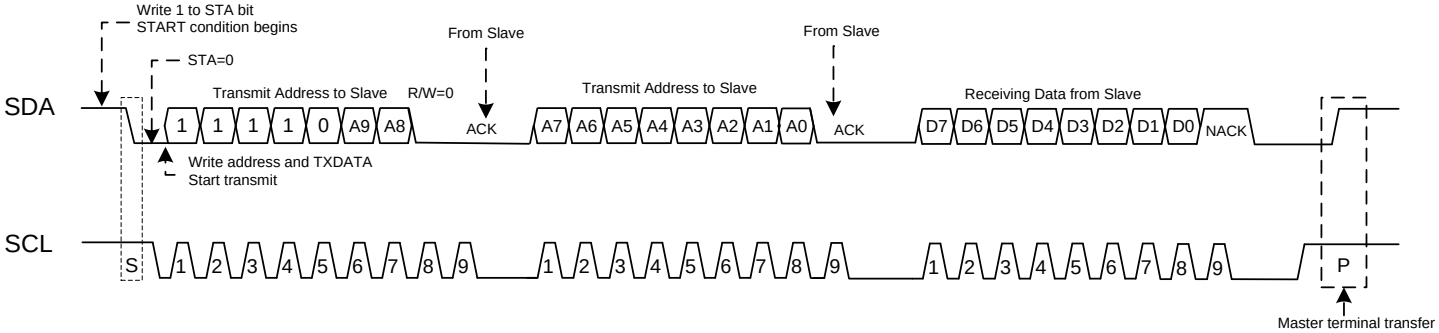


11.4.1.4 SLAVE RECEIVER MODE

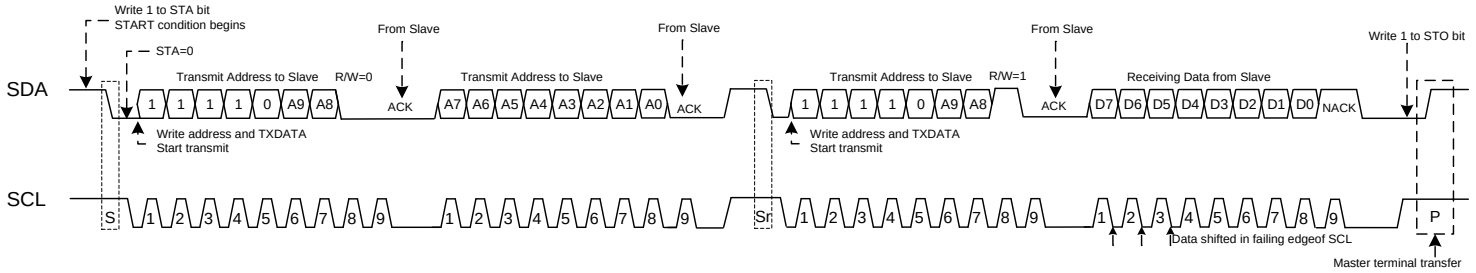


11.4.2 10-BIT ADDRESSING MODES

11.4.2.1 MASTER TRANSMITTER MODE



11.4.2.2 MASTER RECEIVER MODE



11.5 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 anther 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 anther 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 anther 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 mater 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.

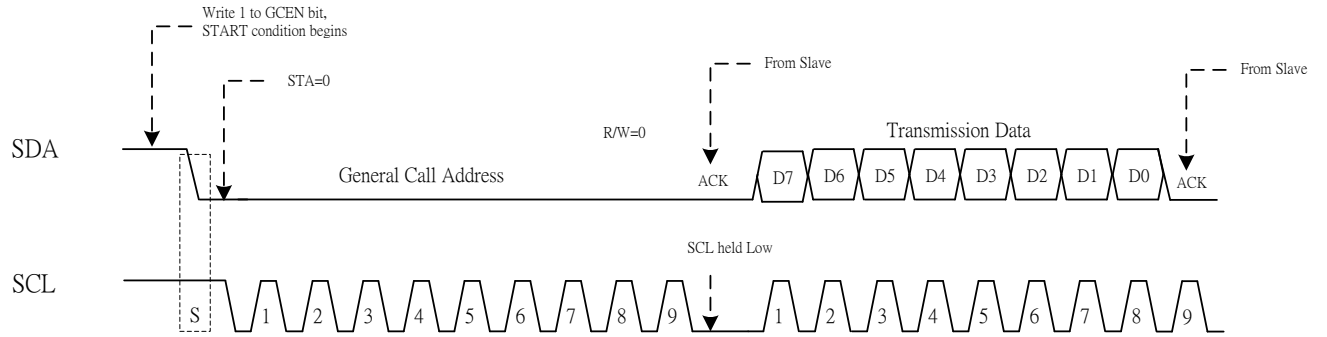
11.6 CLOCK STRETCHING

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.

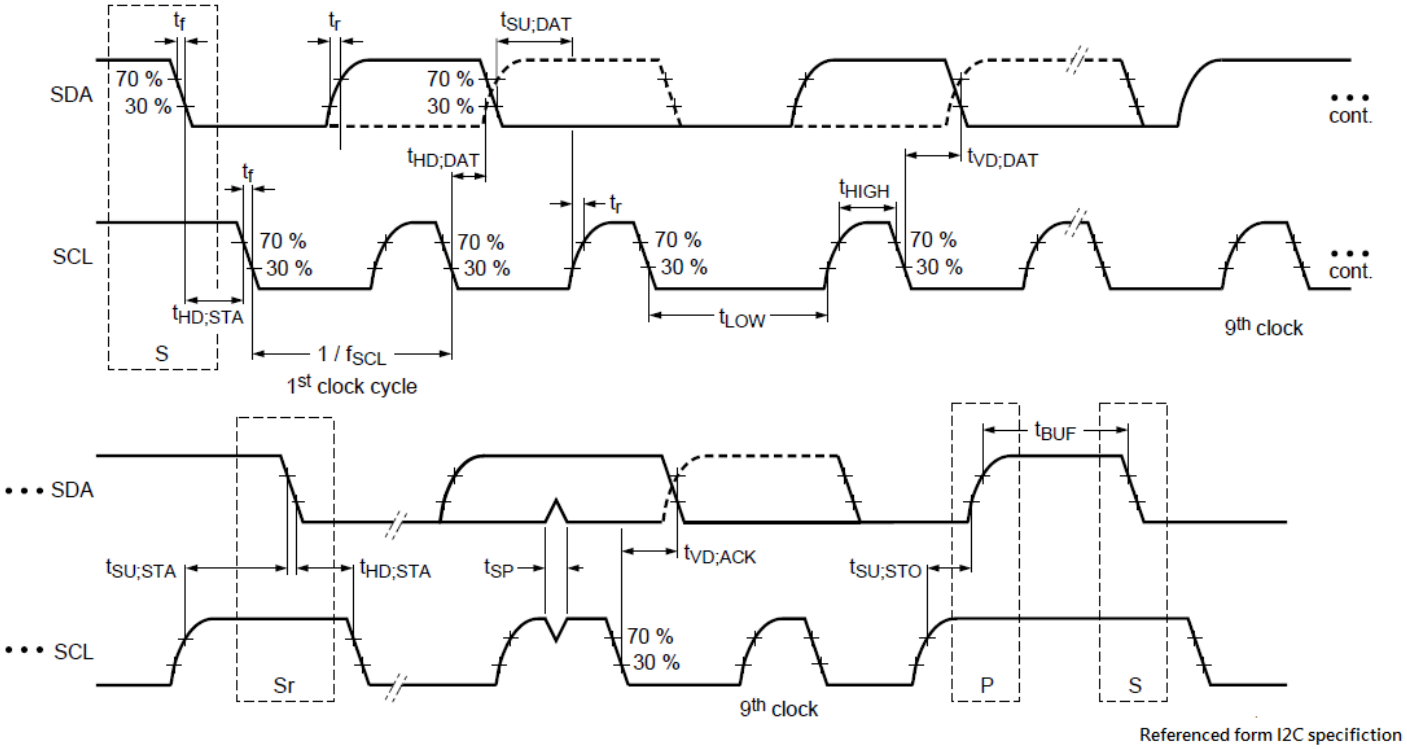
On the byte level, a device may be able to receive bytes of data at a fast rate, but needs more time to store a received byte or prepare another byte to be transmitted. Slaves can then pulls the SCL line LOW after reception and acknowledgment of a byte to force the master into a wait state until the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

11.7 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



11.8 TIMING CHARACTERISTICS



11.8.1 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

11.8.2 SLAVE TRANSMITTER MODE

t_{HIGH}: controlled by Master

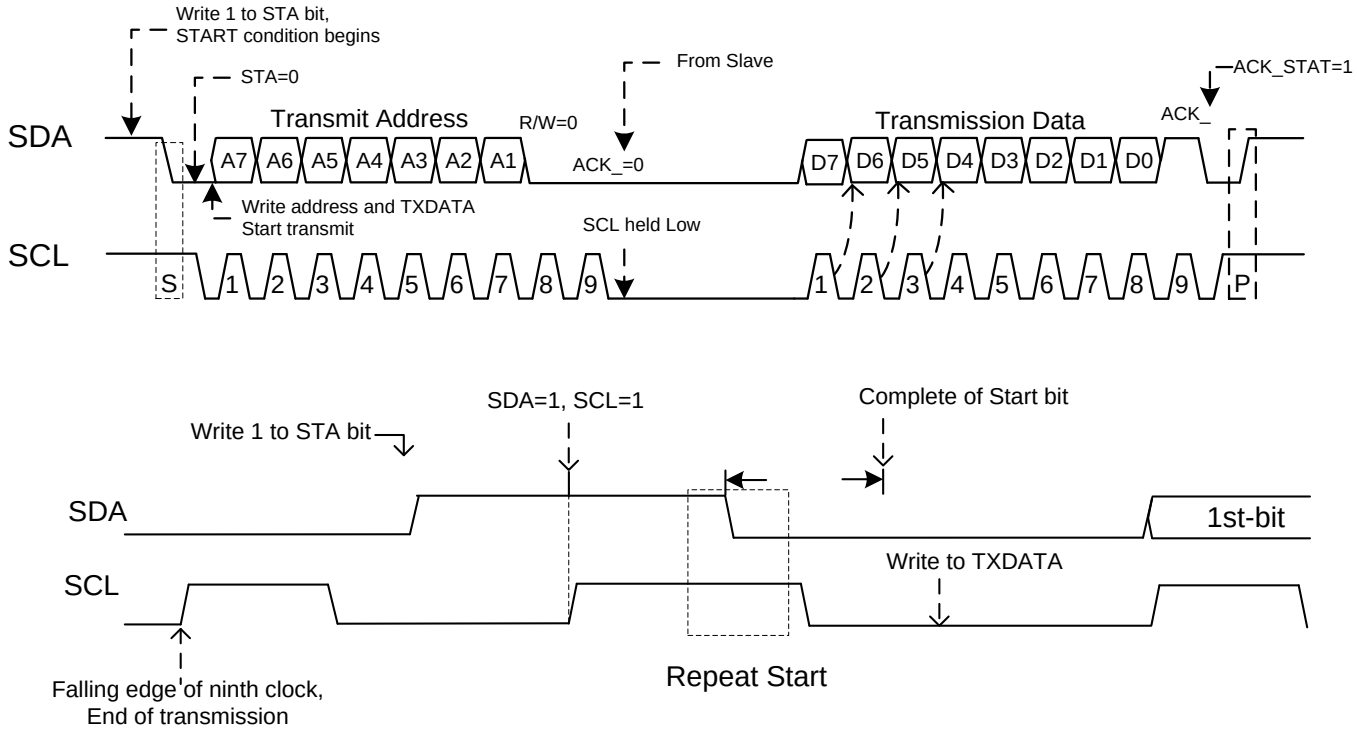
t_{LOW}: controlled by Master

$t_{HD;DAT}$: 2 * I2C_PCLK cycle ~ 3 * I2C_PCLK cycle

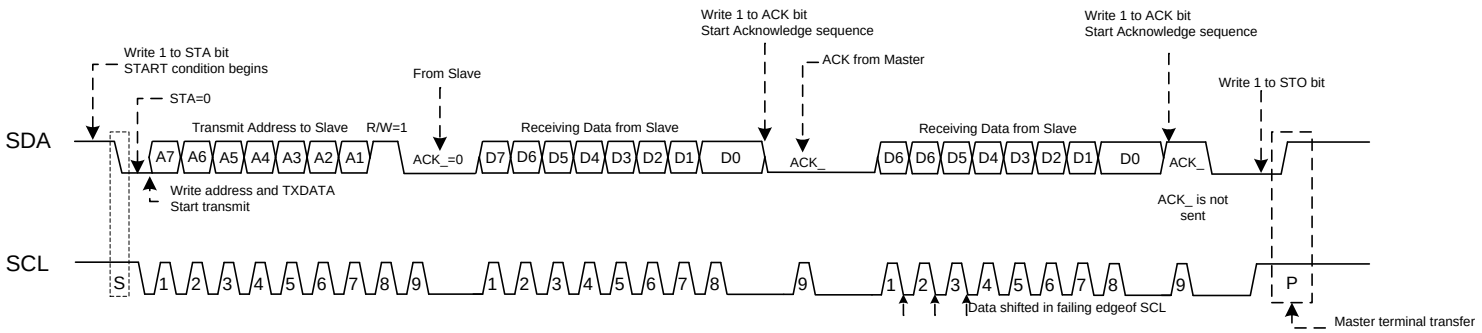
$t_{SU;DAT}$: t_{LOW} - 2 * I2C_PCLK cycle ~ t_{LOW} - 3 * I2C_PCLK cycle

11.9 I2C MASTER MODES

11.9.1 MASTER TRANSMITTER MODE



11.9.2 MASTER RECEIVER MODE



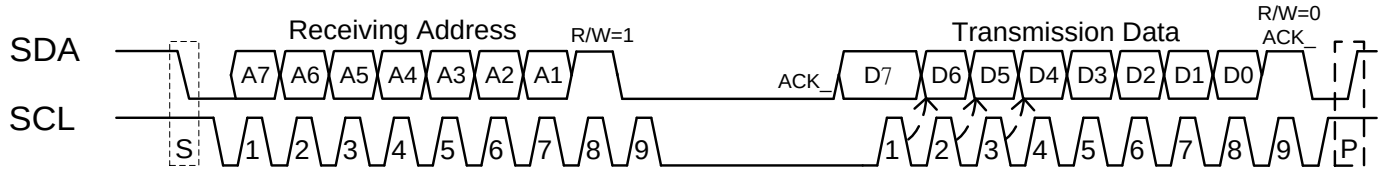
11.9.3 ARBITRATION

In the master transmitter mode, the arbitration logic checks that every transmitted logic 1 actually appears as a logic 1 on the I2C bus. If another device on the bus overrules a logic 1 and pulls the SDA line low, arbitration is lost, and the I2C block immediately changes from master transmitter to slave receiver. The I2C block will continue to output clock pulses (on SCL) until transmission of the current serial byte is complete.

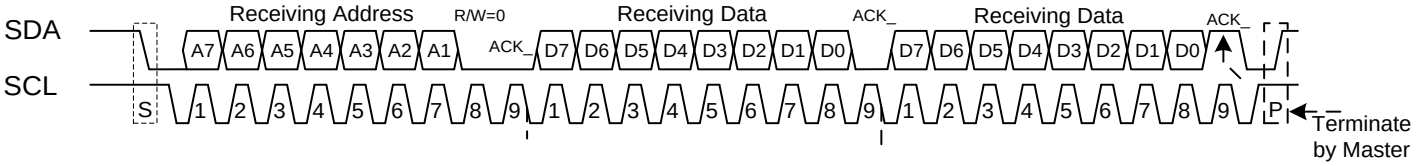
Arbitration may also be lost in the master receiver mode. Loss of arbitration in this mode can only occur while the I2C block is returning a “not acknowledge” to the bus. Arbitration is lost when another device on the bus pulls this signal low. Since this can occur only at the end of a serial byte, the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

11.10 I2C SLAVE MODES

11.10.1 SLAVE TRANSMITTER MODE



11.10.2 SLAVE RECEIVER MODE



11.11 I2C REGISTERS

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

11.11.1 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 it 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/SCL1 and SDA0/SDA1 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. I2EN 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.	R/W	0
7	MODE	I2C mode selection bit. 0: Standard/Fast mode. 1: Reserved.	R	0
6	Reserved			
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.	R/W	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.	R/W	0
3	Reserved		R	0
2	ACK	Assert ACK (Low level to SDA) flag. HW will clear after issuing ACK automatically. 0: Master mode→ No function. Slave mode→Return a NACK after receiving address or data.	R/W	0

		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 Call bit (GC) in the ADR 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.		
1	NACK	Assert NACK (HIGH level to SDA) flag. HW will clear after issuing NACK automatically. 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.	R/W	0
0	Reserved		R	0

11.11.2 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.

While I2CIF =1, the low period of the serial clock on the SCL line is stretched, and the serial transfer is suspended. When SCL is HIGH, it is unaffected by the state of I2CIF.

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.	R/W	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.	R	0

		1: MASTER mode→a STOP condition was issued. SLAVE mode→a STOP condition was received.		
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

11.11.3I2C 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

11.11.4I2C 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

11.11.5I2C 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

11.11.6 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

11.11.7 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. SCL High Period Time = (SCLH+1) * I2C0_PCLK cycle	R/W	0x04

11.11.8 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. SCL Low Period Time = (SCLL+1) * I2C0_PCLK cycle.	R/W	0x04

11.11.9 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

12 Universal Asynchronous Receiver and Transmitter (UART)

12.1 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.

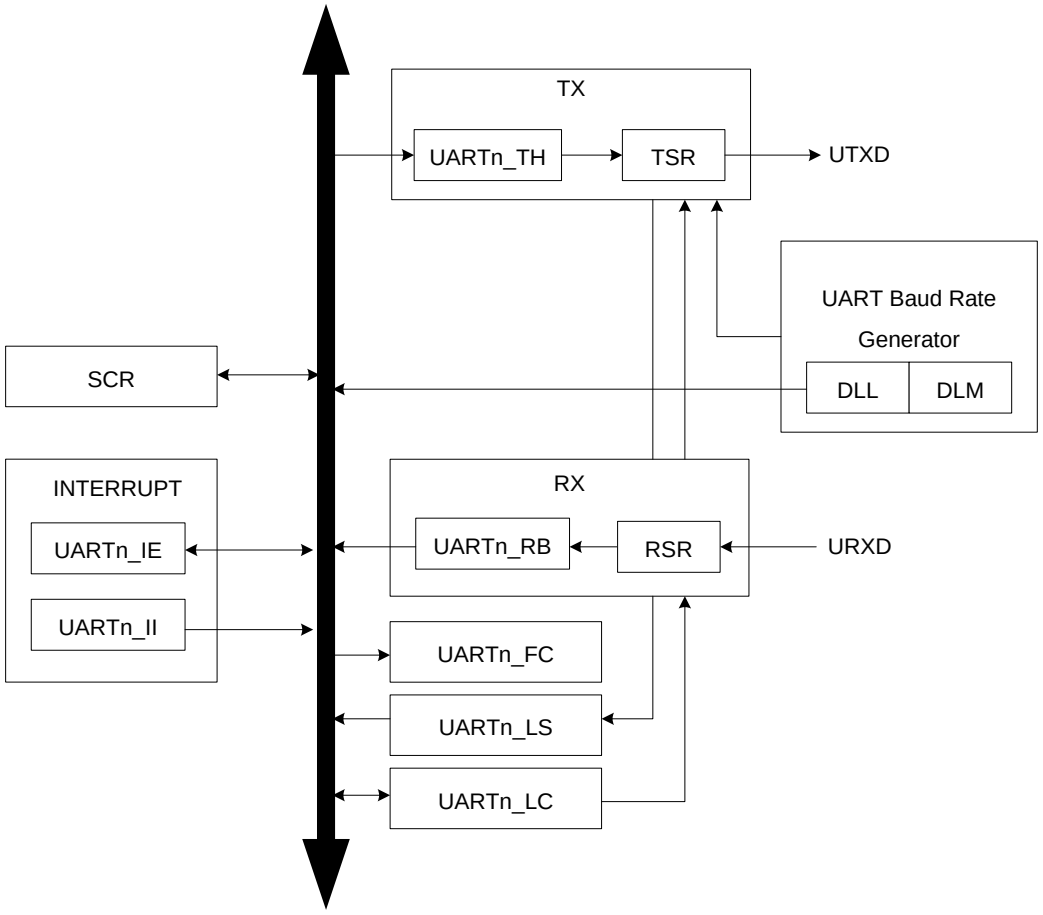
12.2 FEATURES

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

12.3 PIN DESCRIPTION

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

12.4 BLOCK DIAGRAM



12.5 BAUD RATE CALCULATION

The UART baud rate is calculated as:

$$UART_{BAUDRATE} = \frac{UARTn\_PCLK}{Oversampling \times (256 \times DLM + DLL) \times (1 + DIVADDVAL / MULVAL)}$$

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.

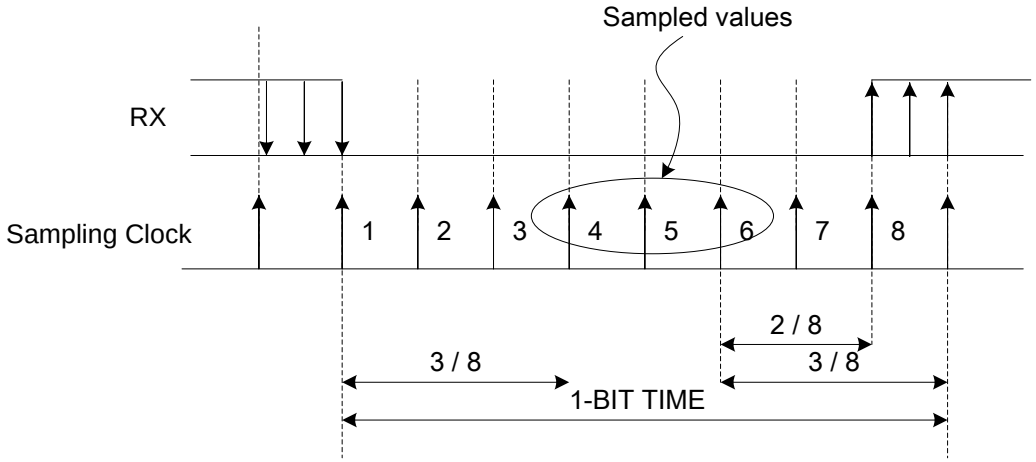
The value of MULVAL and DIVADDVAL should comply with the following conditions:

1. 1 ≤ MULVAL ≤ 15
2. 0 ≤ DIVADDVAL ≤ 14
3. DIVADDVAL< MULVAL
4. Oversampling is 8 or 16

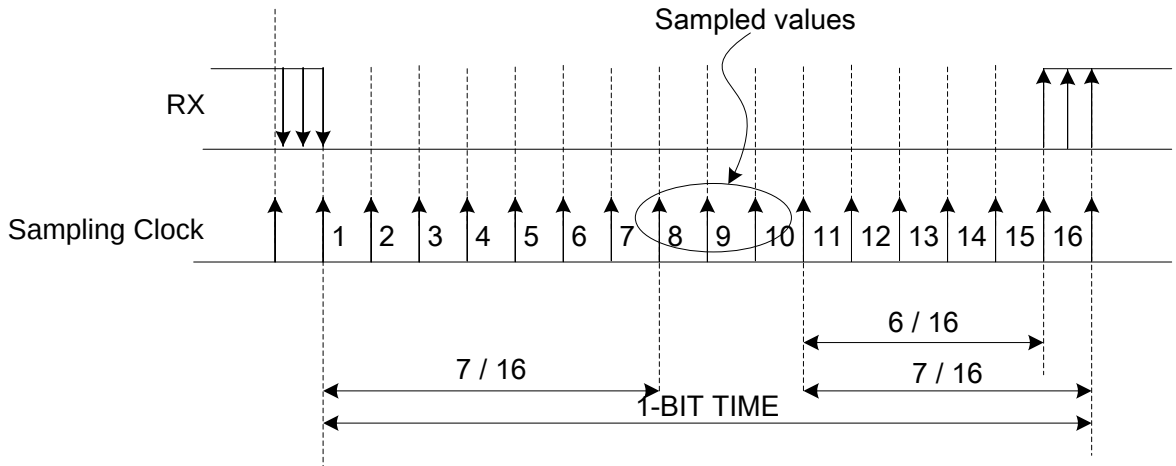
The value of the UARTn_FD register should not be modified while transmitting/receiving data or data may be lost or corrupted.

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, MULVAL, 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, MULVAL, DLM, and DLL to generate BR = 115200 when UARTn_PCLK = 12 MHz, and Oversampling = 16.

$$UART_{BAUDRATE} = \frac{UARTn\_PCLK}{Oversampling \times (256 \times DLM + DLL) \times (1 + DIVADDVAL / MULVAL)}$$

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

$$(256 \times DLM + DLL) \times (1 + DIVADDVAL / MULVAL) = 6.51$$

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

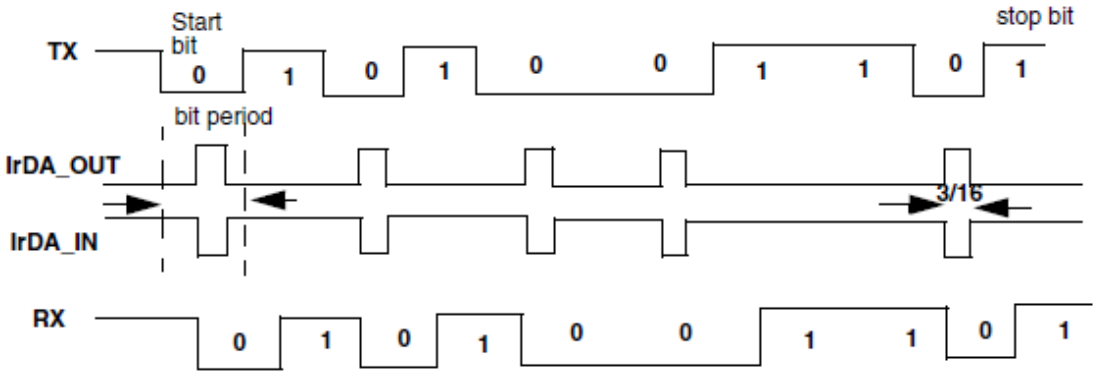
1. 1 ≤ MULVAL ≤ 15
2. 0 ≤ DIVADDVAL ≤ 14
3. DIVADDVAL< MULVAL

Thus, the suggested UART settings would be: DLM = 0, DLL = 4, DIVADDVAL = 5, and MULVAL = 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.

12.6 SERIAL IrDA MODE

The IrDA mode is enabled by setting the UARTEN bit to 1 and MODE[2:0] = 010b in UARTn_CTRL register.
The SIR Transmit encoder modulates the Non Return to Zero (NRZ) transmit bit stream output from UART. The output pulse stream is transmitted to an external output driver and infrared LED. UART supports only bit rates up to 115.2Kbps for the SIR ENDEC. In normal mode the transmitted pulse width is specified as 3/16 of a bit period.
The SIR receive decoder demodulates the Return-to-Zero bit stream from the infrared detector and outputs the received NRZ serial bit stream to UART. The decoder input is normally HIGH (marking state) in the Idle state. The transmit encoder output has the opposite polarity to the decoder input. A start bit is detected when the decoder input is low.

- IrDA is a half duplex communication protocol. If the Transmitter is busy (i.e. the UART is sending data to the IrDA encoder), any data on the IrDA receive line will be ignored by the IrDA decoder and if the Receiver is busy (UART is receiving decoded data from the UART), data on the TX from the UART to IrDA will not be encoded by IrDA. While receiving data, transmission should be avoided as the data to be transmitted could be corrupted.
- A '0 is transmitted as a high pulse and a '1 is transmitted as a '0. The width of the pulse is specified as 3/16th of the selected bit period in normal mode.



- The SIR decoder converts the IrDA compliant receive signal into a bit stream for UART.
- The SIR receive logic interprets a high state as a logic one and low pulses as logic zeros.
- The transmit encoder output has the opposite polarity to the decoder input. The SIR output is in low state when

Idle.

- In IrDA mode, the STOP bits must be configured to “1 stop bit”.
- The IrDA specification requires the acceptance of pulses greater than 1.41 us. The acceptable pulse width is programmable. Glitch detection logic on the receiver filters out pulses of width less than 2 times low-power baud rate. Pulses of width greater than 2 low-power baud rate will be accepted as a pulse.
- The receiver can communicate with a low-power transmitter. In low-power mode the pulse width is not maintained at 3/16 of the bit period. Instead, the width of the pulse is 3 times the low-power baud rate which can be a minimum of 1.42 MHz. Generally the low-power baud rate is 1.8432 MHz (1.42 MHz < low-power baud rate < 2.12 MHz).
- If FIXPULSEEN = 0, the low level pulse width shall > (2/16 *baud cycle) for receiver to accepted as a low pulse; if FIXPULSEEN = 1, the low level pulse width shall > (1/2 * PULSEDIV setting PCLK) for receiver to accepted as a low pulse.
- The PULSEDIV bits are used to select the pulse width when the fixed pulse width mode is used in IrDA mode when FIXPULSEEN = 1. The value of these bits should be set so that the resulting pulse width is at least 1.63 µs. Table shows the possible pulse widths.
- OVER8 bit in UARTn_FD register must be 0 in IrDA mode, oversampling by 8 is not supported in IrDA mode.

IrDA Pulse Width			
FIXPULSEEN	PULSEDIV[2:0]	OVER8	IrDA Transmitter Pulse Width (us)
0	X	0	3 / (16 x Baud rate)
1	0	0	2 x TPCLK
1	1	0	4 x TPCLK
1	2	0	8 x TPCLK
1	3	0	16 x TPCLK
1	4	0	32 x TPCLK
1	5	0	64 x TPCLK
1	6	0	128 x TPCLK
1	7	0	256 x TPCLK

12.7 UART REGISTERS

Base Address: 0x4001 6000 (UART0)
0x4001 4000 (UART1)
0x4001 2000 (UART2)

12.7.1 UART n Receiver Buffer register (UARTn_RB) (n=0,1,2)

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

12.7.2 UART n Transmitter Holding register (UARTn_TH) (n=0,1,2)

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

12.7.3 UART n Divisor Latch LSB register (UARTn_DLL) (n=0,1,2)

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.	R/W	0

12.7.4 UART n Divisor Latch MSB register (UARTn_DLM) (n=0,1,2)

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 DLM register, determines the baud rate of the UART.	R/W	0

12.7.5 UART n Interrupt Enable register (UARTn_IE) (n=0,1,2)

Address Offset: 0x04

The DLAB bit in 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

12.7.6 UART n Interrupt Identification register (UARTn_II) (n=0,1,2)

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	0
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 UARTn_II[3:1].	R	1

		0: At least one interrupt is pending. 1: No interrupt is pending.		

Bits UARTn_II[9:8] are set by the auto-baud function and signal a time-out or end of auto-baud condition. The auto-baud interrupt conditions are cleared by setting the corresponding Clear bits in the Auto-baud Control Register.

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

Interrupt	UARTn_II [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	2nd Level	RX data in FIFO reached trigger level (FCR0=1)	Read UARTn_RB register or UART FIFO drops below trigger level
THRE	0010	3rd Level	THRE	Read UARTn_II register (if source of interrupt) or Write THR register
TEMT	1110	3rd Level	TEMT	Read UARTn_II register (if source of interrupt) or Write THR register

12.7.7 UART n FIFO Control register (UARTn_FIFOCTRL) (n=0,1,2)

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

12.7.8 UART n Line Control register (UARTn_LC) (n=0,1,2)

Address Offset: 0x0C

The UARTn_LC 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.	R/W	0

		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.		
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

12.7.9 UART n Line Status register (UARTn_LS) (n=0,1,2)

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_FIFOCTRL register. Note: The break interrupt is associated with the character at the top of the UARTn_RB FIFO. 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 logic 0, a framing error occurs. A UARTn_LS register read clears FE bit. The time of the framing error	R	0

		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. Note: A framing error is associated with the character at the top of the UARTn_RB FIFO. 0: Framing error status is inactive. 1: Framing error status is active.		
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 FIFOENbit 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

12.7.10 UART n Scratch Pad register (UARTn_SP) (n=0,1,2)

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

12.7.11 UART n IrDA Control register (UARTn_IRDACTRL) (n=0,1,2)

Address Offset: 0x24

The UARTn_IRDACTRL register enables and configures the IrDA mode. The value of the UARTn_IRDACTRL register should not be changed while transmitting or receiving data, or data loss or corruption may occur.

Bit	Name	Description	Attribute	Reset
31:6	Reserved		R	0
5:3	PULSEDIV[2:0]	Configures the pulse width when FIXPULSEEN = 1. 000: 2 x TPCLK 001: 4 x TPCLK 010: 8 x TPCLK 011: 16 x TPCLK 100: 32 x TPCLK 101: 64 x TPCLK 110: 128 x TPCLK 111: 256 x TPCLK	R/W	0
2	FIXPULSEEN	IrDA fixed pulse width mode enable.	R/W	0

		0: Disable. Pulse width = 3 / (Oversampling x baud rate) 1: Enable. Pulse width is set by PULSEDIV bits.		
1:0	Reserved		R	0

12.7.12 UART n Fractional Divider register (UARTn_FD) (n=0,1,2)

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: If the fractional divider is active (DIVADDVAL > 0) and UARTn_DLM = 0, the value of the UARTn_DLL register must ≥ 3.***

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 = 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

12.7.13 UART n Control register (UARTn_CTRL) (n=0,1,2)

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 hardware 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	When this bit is 1, data written to the UARTn_TH register is output on the TXD pin as soon as any preceding data has been sent. If this bit is cleared to 0 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.	R/W	1
6	RXEN	0: Disable RX function 1: Enable RX	R/W	1
5:4	Reserved		R	0
3:1	MODE[2:0]	UARTn Mode 000: UART mode. 010: IrDA mode. Other: Reserved	R/W	0
0	UARTEN	UART enable 0: Disable 1: Enable. HW switches GPIO to UART pin according to MODE bits automatically.	R/W	0

12.7.14 UART n Half-duplex Enable register (UARTn_HDEN) (n=0,1,2)

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

13 CYCLIC REDUNDANCY CHECK (CRC)

13.1 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.

13.2 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

13.3 CRC REGISTERS

Base Address: 0x4003 8000

13.3.1 CRC Control register (CRC_CTRL)

Address offset: 0x00

Bit	Name	Description	Attribute	Reset
31:5	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

13.3.2 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

14 USB FS DEVICE INTERFACE

14.1 OVERVIEW

The USB is the answer to connectivity for the PC architecture. A fast, bi-directional interrupt pipe, low-cost, dynamically attachable serial interface is consistent with the requirements of the PC platform of today and tomorrow. The SONiX USB microcontrollers are optimized for human-interface computer peripherals such as a mouse, keyboard, joystick, and game pad.

USB Specification Compliance

— Conforms to USB specifications, Version 2.0.

— Supports 1 Full-speed USB device address.

— Supports 1 control endpoint and 7 configurable endpoints for interrupt/bulk transfer.

— Integrated USB transceiver.

— 5V to 3.3V regulator output for D+ 1.5K ohm internal resistor pull up.

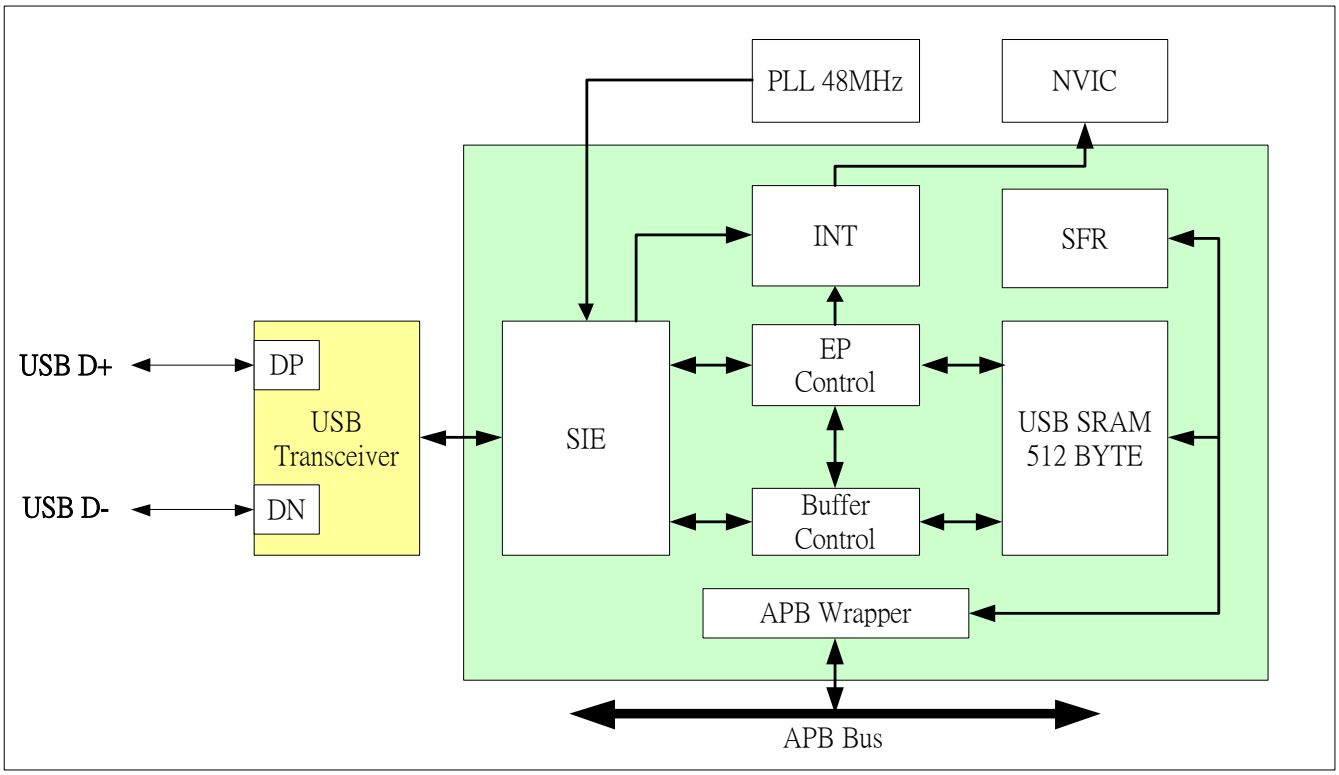
14.2 FEATURES

- Conforms to USB specifications, Version 2.0.
- Supports 1 Full-speed USB device address.
- Supports 1 control endpoint with maximum packet size 8 bytes, 16 bytes, 32 bytes, or 64 bytes.
- Supports 7 endpoints configurable for interrupt/bulk transfer.
- Supports USB SRAM size 512 bytes shared by all 8 endpoints.
- Flexible data FIFO offset setting for endpoints except endpoint 0.
- 5V to 3.3V regulator output for D+ 1.5K ohm internal resistor pull up.
- Integrated USB transceiver.
- FS USB function work under system clock SYSCLK/1, SYSCLK/2, SYSCLK/4.

14.3 PIN DESCRIPTION

Pin Name	Type	Description	GPIO Configuration
D+	I/O	USB differential signal D+	N/A
D-	I/O	USB differential signal D-	N/A

14.4 BLOCK DIAGRAM



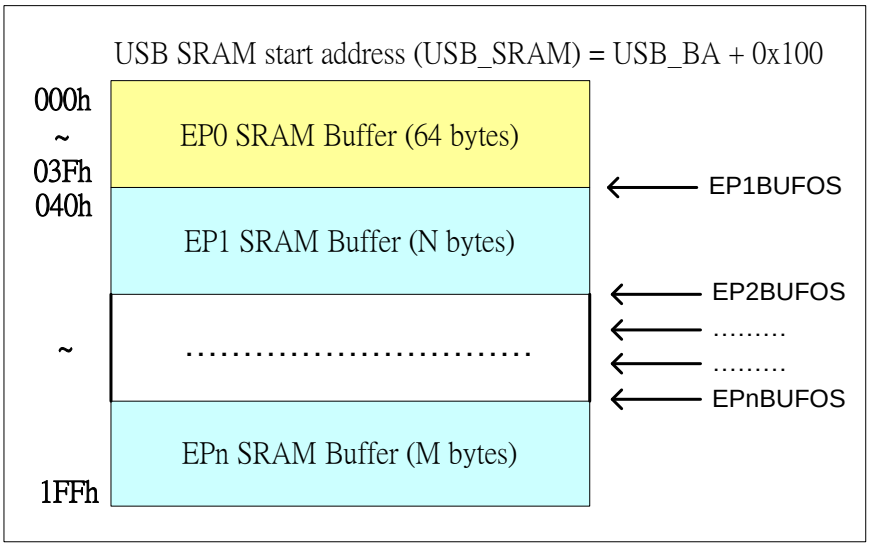
14.5 USB SRAM ACCESS

There is 512 bytes SRAM in the controller and the 8 endpoints share this buffer. The user shall configure each endpoint's effective starting address in the buffer offset register before the USB function active. The USB_EPnBUFOS block is used to control each endpoint's effective starting address.

The principles to access USB SRAM are as below.

– Each EPnBUFOS setting must be word-aligned, with 2 LSB bits equal to '0'.

– The maximum length of EPn SRAM buffer is defined by user. However, each endpoint should have its own EPn SRAM buffer without overlapping each other.



14.6 USB MACHINE

The USB machine allows the microcontroller to communicate with the USB host. The hardware handles the following USB bus activity independently of the microcontroller.

The USB machine will do:

– Translate the encoded received data and format the data to be transmitted on the bus.
– CRC checking and generation by hardware. If CRC is not correct, hardware will not send any response to USB host.
– Send and update the data toggle bit (Data1/0) automatically by hardware.
– Send appropriate ACK/NAK/STALL handshakes.
– SETUP, IN, or OUT Token type identification. Set the appropriate bit once a valid token is received.
– Place valid received data in the appropriate endpoint FIFOs.
– Bit stuffing/unstuffing.
– Address checking. Ignore the transactions not addressed to the device.
– Endpoint checking. Check the endpoint's request from USB host, and set the appropriate bit of registers.

Firmware is required to handle the rest of the following tasks:

– Coordinate enumeration by decoding USB device requests.
– Fill and empty the FIFOs.
– Reset/Suspend/Resume coordination.
– Remote wake up function.
– Determine the right interrupt request of USB communication

14.7 USB INTERRUPT

The USB function will accept the USB host command and generate the relative interrupts, and enter USB_IRQ_Handler. Firmware is required to check the USB status bit to realize what request comes from the USB host.

The USB function interrupt is generated when:

– The endpoint 0 is set to accept a SETUP token.
– The device receives an ACK handshake after a successful read transaction (IN) from the host.
– If the endpoint is in ACK OUT modes, an interrupt is generated when data is received.
– The USB host sends USB suspend request to the device.
– USB bus reset/resume event occurs.
– The USB endpoints interrupt after a USB transaction complete is on the bus.
– The NAK handshaking when the NAK interrupt enables.

14.8 USB ENUMERATION

A typical USB enumeration sequence is shown below.

1. The host computer sends a SETUP packet followed by a DATA packet to USB address 0 requesting the Device descriptor.
2. Firmware decodes the request and retrieves its Device descriptor from the program memory tables.
3. The host computer performs a control read sequence and firmware responds by sending the Device descriptor over the USB bus, via the on-chip USB SRAM.
4. After receiving the descriptor, the host sends a SETUP packet followed by a DATA packet to address 0 assigning a new USB address to the device.
5. Firmware stores the new address in its USB Device Address Register after the no-data control sequence completes.
6. The host sends a request for the Device descriptor using the new USB address.
7. Firmware decodes the request and retrieves the Device descriptor from program memory tables.
8. The host performs a control read sequence and firmware responds by sending its Device descriptor over the USB bus.
9. The host generates control reads from the device to request the Configuration and Report descriptors.
10. Once the device receives a Set Configuration request, its functions may now be used.
11. Firmware should take appropriate action for Endpoint 0~N transactions, which may occur from this point.

14.9 USB REGISTERS

Base Address: 0x4005 C000

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
USB_INTEN	0x00	R/W	USB Interrupt Enable Register.	0x0000_0000
USB_INSTS	0x04	R	USB Interrupt Event Status Register.	0x0000_0000
USB_INSTSC	0x08	W	USB Interrupt Event Status Clear Register.	0x0000_0000
USB_ADDR	0x0C	R/W	USB Device Address Register.	0x0000_0000
USB_CFG	0x10	R/W	USB Configuration Register.	0x0000_0000
USB_SGCTL	0x14	R/W	USB Signal Control Register.	0x0000_0000
USB_EP0CTL	0x18	R/W	USB Endpoint 0 Control Register.	0x0000_0000
USB_EP1CTL	0x1C	R/W	USB Endpoint 1 Control Register.	0x0000_0000
USB_EP2CTL	0x20	R/W	USB Endpoint 2 Control Register.	0x0000_0000
USB_EP3CTL	0x24	R/W	USB Endpoint 3 Control Register.	0x0000_0000
USB_EP4CTL	0x28	R/W	USB Endpoint 4 Control Register.	0x0000_0000
USB_EP5CTL	0x2C	R/W	USB Endpoint 5 Control Register.	0x0000_0000
USB_EP6CTL	0x30	R/W	USB Endpoint 6 Control Register.	0x0000_0000
USB_EP7CTL	0x34	R/W	USB Endpoint 7 Control Register.	0x0000_0000
USB_EPTOGGLE	0x3C	R/W	USB Endpoint Data Toggle Register.	0x0000_007F
USB_EP1BUFOS	0x48	R/W	USB Endpoint 1 Buffer Offset Register.	0x0000_0040
USB_EP2BUFOS	0x4C	R/W	USB Endpoint 2 Buffer Offset Register.	0x0000_0080
USB_EP3BUFOS	0x50	R/W	USB Endpoint 3 Buffer Offset Register.	0x0000_00C0
USB_EP4BUFOS	0x54	R/W	USB Endpoint 4 Buffer Offset Register.	0x0000_0100
USB_EP5BUFOS	0x58	R/W	USB Endpoint 5 Buffer Offset Register.	0x0000_0140
USB_EP6BUFOS	0x5C	R/W	USB Endpoint 6 Buffer Offset Register.	0x0000_0180
USB_EP7BUFOS	0x60	R/W	USB Endpoint 7 Buffer Offset Register.	0x0000_01C0
USB_FRMNO	0x64	R	USB Frame Number Register.	0x0000_0000
USB_PHYPRM	0x68	R/W	USB PHY Parameter Register.	0x0000_0000
USB_PHYPRM2	0x70	R/W	USB PHY Parameter Register 2.	0x0000_0000
USB_PS2CTL	0x74	R/W	USB PS/2 Control Register.	0x0000_0000
USB_RWADDR	0x78	R/W	USB FIFO Read/Write Address Register	0x0000_0000
USB_RWDATA	0x7C	R/W	USB FIFO Data Register	0x0000_0000
USB_RWSTATUS	0x80	R/W	USB FIFO Read/Write Status Register	0x0000_0000
USB_RWADDR2	0x84	R/W	USB FIFO Read/Write Address Register2	0x0000_0000
USB_RWDATA2	0x88	R/W	USB FIFO Data Register2	0x0000_0000
USB_RWSTATUS2	0x8C	R/W	USB FIFO Read/Write Status Register2	0x0000_0000
USB_SRAM	0x100	R/W	USB 512 byte SRAM	Undefined

14.9.1 USB Interrupt Enable Register (USB_INTEN)

Address Offset: 0x00

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31	BUS_IE	Bus Event Interrupt Enable. 0: Disable BUS event interrupt. 1: Enable Bus event interrupt. Any bus event including BUS_RESET, BUS_SUSPEND, and BUS_RESUME triggers USB interrupt.	R/W	0
30	USB_SOF_IE	USB SOF Interrupt Enable. 0: Disable USB SOF interrupt. 1: Enable USB SOF interrupt.	R/W	0
29	USB_IE	USB Event Interrupt Enable. 0: Disable USB event interrupt. 1: Enable USB event interrupt. Any USB event except EP1~EP7's NAK triggers USB interrupt.	R/W	0

28	BUSWK_IE	BUSWK_IE: Bus Wake Up Interrupt Enable. 0: Disable Wake Up event interrupt. 1: Enable Wake Up event interrupt.	R/W	0
27:8	Reserved		R	0
7	EPN_ACK_EN	Enable all of EP(1~7) ACK Interrupt. 0: Disable EP1 to 7 ACK interrupt function. 1: Enable EP1 to 7 ACK interrupt function.	R/W	0
6	EP7_NAK_EN	EP7 NAK Interrupt Enable. 0: Disable EP7 NAK interrupt function. 1: Enable EP7 NAK interrupt function.	R/W	0
5	EP6_NAK_EN	EP6 NAK Interrupt Enable. 0: Disable EP6 NAK interrupt function. 1: Enable EP6 NAK interrupt function.	R/W	0
4	EP5_NAK_EN	EP5 NAK Interrupt Enable. 0: Disable EP5 NAK interrupt function. 1: Enable EP5 NAK interrupt function.	R/W	0
3	EP4_NAK_EN	EP4 NAK Interrupt Enable. 0: Disable EP4 NAK interrupt function. 1: Enable EP4 NAK interrupt function.	R/W	0
2	EP3_NAK_EN	EP3 NAK Interrupt Enable. 0: Disable EP3 NAK interrupt function. 1: Enable EP3 NAK interrupt function.	R/W	0
1	EP2_NAK_EN	EP2 NAK Interrupt Enable. 0: Disable EP2 NAK interrupt function. 1: Enable EP2 NAK interrupt function.	R/W	0
0	EP1_NAK_EN	EP1 NAK Interrupt Enable. 0: Disable EP1 NAK interrupt function. 1: Enable EP1 NAK interrupt function.	R/W	0

14.9.2 USB Interrupt Event Status Register (USB_INSTS)

Address Offset: 0x04

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31	BUS_RESET	USB Bus Reset signal (>2.5us SE0) flag. 0: No bus reset signal is detected. 1: Bus reset signal is detected. Cleared by write 1 to USB_INSTSC[31].	R	0
30	BUS_SUSPEND	USB Bus Suspend signal (>3ms idle state) flag. 0: No bus suspend is detected. 1: Bus suspend is detected.	R	0
29	BUS_RESUME	USB Bus Resume signal flag. 0: No bus resume signal is detected. 1: Bus resume signal from suspend mode is detected. Cleared by write 1 to USB_INSTSC[29].	R	0
28:27	Reserved		R	0
26	USB_SOF	USB SOF packet received flag. 0: No USB SOF packet. 1: USB SOF packet is received. Cleared by write 1 to USB_INSTSC[26].	R	0
25	BUS_WAKEUP	Bus Wake Up flag 0: No wakeup from suspend mode. 1: Wakeup from suspend mode. Cleared by write 1 to USB_INSTSC[25] *This flag will be set to '1'when wakeup from suspend mode under USB PLL 48MHz is off.	R	0
24	EP0_PRESETUP	EP0 Setup token packet flag. This flag will not trigger USB interrupt. 0: No EP0 Setup token packet. 1: EP0 Setup token packet is received. Cleared by write 1 to USB_INSTSC[24]	R	0
23	EP0_SETUP	EP0 Setup transaction flag. 0: No EP0 Setup transaction.	R	0

		1: EP0 Setup transaction is completed. Cleared by write 1 to USB_INSTSC[23].		
22	EP0_IN	EP0 IN ACK transaction flag. 0: No EP0 IN ACK Transaction. 1: EP0 IN ACK transaction is completed. Cleared by write 1 to USB_INSTSC[22].	R	0
21	EP0_OUT	EP0 OUT ACK transaction flag. 0: No EP0 OUT ACK transaction. 1: EP0 OUT ACK transaction is completed. Cleared by write 1 to USB_INSTSC[21].	R	0
20	EP0_IN_STALL	EP0 IN STALL transaction flag. 0: No EP0 IN STALL transaction. 1: EP0 IN STALL transaction is completed. Cleared by write 1 to USB_INSTSC[20].	R	0
19	EP0_OUT_STALL	EP0 OUT STALL transaction flag. 0: No EP0 OUT STALL transaction. 1: EP0 OUT STALL transaction is completed. Cleared by write 1 to USB_INSTSC[19].	R	0
18	ERR_SETUP	Wrong Setup data received. This flag will not trigger USB interrupt. 0: Normal 8-byte Setup DATA0 is received. 1: Setup data is not 8-byte or is not DATA0. Cleared by write 1 to USB_INSTSC[18].	R	0
17	ERR_TIMEOUT	Timeout status. This flag will not trigger USB interrupt. 0: No timeout. 1: Host ACK response timeout after IN data packet is sent. Cleared by write 1 to USB_INSTSC[17].	R	0
16:15	Reserved		R	0
14	EP7_ACK	Endpoint 7 ACK transaction flag. 0: No Endpoint 7 ACK transaction. 1: Endpoint 7 ACK transaction completes. Cleared by write 1 to USB_INSTSC[14].	R	0
13	EP6_ACK	Endpoint 6 ACK transaction flag. 0: No Endpoint 6 ACK transaction. 1: Endpoint 6 ACK transaction completes. Cleared by write 1 to USB_INSTSC[13].	R	0
12	EP5_ACK	Endpoint 5 ACK transaction flag. 0: No Endpoint 5 ACK transaction. 1: Endpoint 5 ACK transaction completes. Cleared by write 1 to USB_INSTSC[12].	R	0
11	EP4_ACK	Endpoint 4 ACK transaction flag. 0: No Endpoint 4 ACK transaction. 1: Endpoint 4 ACK transaction completes. Cleared by write 1 to USB_INSTSC[11].	R	0
10	EP3_ACK	Endpoint 3 ACK transaction flag. 0: No Endpoint 3 ACK transaction. 1: Endpoint 3 ACK transaction completes. Cleared by write 1 to USB_INSTSC[10].	R	0
9	EP2_ACK	Endpoint 2 ACK transaction flag. 0: No Endpoint 2 ACK transaction. 1: Endpoint 2 ACK transaction completes. Cleared by write 1 to USB_INSTSC[9].	R	0
8	EP1_ACK	Endpoint 1 ACK transaction flag. 0: No Endpoint 1 ACK transaction. 1: Endpoint 1 ACK transaction completes. Cleared by write 1 to USB_INSTSC[8].	R	0
7	Reserved		R	0
6	EP7_NAK	Endpoint 7 NAK transaction flag. 0: No EP7 NAK transaction. 1: EP7 NAK transaction completes. Cleared by write 1 to USB_INSTSC[6].	R	0
5	EP6_NAK	Endpoint 6 NAK transaction flag. 0: No EP6 NAK transaction. 1: EP6 NAK transaction completes. Cleared by write 1 to USB_INSTSC[5].	R	0
4	EP5_NAK	Endpoint 5 NAK transaction flag. 0: No EP5 NAK transaction. 1: EP5 NAK transaction completes. Cleared by write 1 to USB_INSTSC[4].	R	0
3	EP4_NAK	Endpoint 4 NAK transaction flag.	R	0

		0: No EP4 NAK transaction. 1: EP4 NAK transaction completes. Cleared by write 1 to USB_INSTSC[3].		
2	EP3_NAK	Endpoint 3 NAK transaction flag. 0: No EP3 NAK transaction. 1: EP3 NAK transaction completes. Cleared by write 1 to USB_INSTSC[2].	R	0
1	EP2_NAK	Endpoint 2 NAK transaction flag. 0: No EP2 NAK transaction. 1: EP2 NAK transaction completes. Cleared by write 1 to USB_INSTSC[1].	R	0
0	EP1_NAK	Endpoint 1 NAK transaction flag. 0: No EP1 NAK transaction. 1: EP1 NAK transaction completes. Cleared by write 1 to USB_INSTSC[0].	R	0

14.9.3 USB Interrupt Event Status Clear Register (USB_INSTSC)

Address Offset: 0x08

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31	BUS_RESETC	0: No effect. 1: Clear BUS_RESET bit.	W	0
30	Reserved		R	-
29	BUS_RESUMEC	0: No effect. 1: Clear BUS_RESUME bit.	W	0
28:27	Reserved		R	-
26	USB_SOFC	0: No effect. 1: Clear USB_SOF bit.	W	0
25	BUS_WAKEUPC	0: No effect. 1: Clear BUS_WAKEUP bit.	W	0
24	EP0_PRESETUPC	0: No effect 1: Clear EP0_PRESETUP bit.	W	0
23	EP0_SETUPC	0: No effect 1: Clear EP0_SETUP bit.	W	0
22	EP0_INC	0: No effect. 1: Clear EP0_IN bit.	W	0
21	EP0_OUTC	0:.No effect. 1: Clear EP0_OUT bit.	W	0
20	EP0_IN_STALLC	0: No effect. 1: Clear EP0_IN_STALL bit.	W	0
19	EP0_OUT_STALLC	0: No effect. 1: Clear EP0_OUT_STALL bit.	W	0
18	ERR_SETUPC	0: No effect. 1: Clear ERR_SETUP bit.	W	0
17	ERR_TIMEOUTC	0: No effect. 1: Clear ERR_TIMEOUT bit.	W	0
16:15	Reserved		R	-
14	EP7_ACKC	0: No effect. 1: Clear EP7_ACK bit.	W	0
13	EP6_ACKC	0: No effect. 1: Clear EP6_ACK bit.	W	0
12	EP5_ACKC	0: No effect. 1: Clear EP5_ACK bit.	W	0
11	EP4_ACKC	0: No effect. 1: Clear EP4_ACK bit.	W	0
10	EP3_ACKC	0: No effect. 1: Clear EP3_ACK bit.	W	0
9	EP2_ACKC	0: No effect. 1: Clear EP2_ACK bit.	W	0
8	EP1_ACKC	0: No effect. 1: Clear EP1_ACK bit.	W	0
7	Reserved		R	-

6	EP7_NAKC	0: No effect. 1: Clear EP7_NAK bit.	W	0
5	EP6_NAKC	0: No effect. 1: Clear EP6_NAK bit.	W	0
4	EP5_NAKC	0: No effect. 1: Clear EP5_NAK bit.	W	0
3	EP4_NAKC	0: No effect. 1: Clear EP4_NAK bit.	W	0
2	EP3_NAKC	0: No effect. 1: Clear EP3_NAK bit.	W	0
1	EP2_NAKC	0: No effect. 1: Clear EP2_NAK bit.	W	0
0	EP1_NAKC	0: No effect. 1: Clear EP1_NAK bit.	W	0

14.9.4 USB Device Address Register (USB_ADDR)

Address Offset: 0x0C

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6:0	UADDR	USB device's address.	R/W	0

14.9.5 USB Configuration Register (USB_CFG)

Address offset: 0x10

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31	VREG33_EN	Internal VREG33 output function. If VREG33_EN is disabled, VREG33 will be switched to IC_VDD. 0: Disable 1: Enable *If VREG33 is disabled, the VREG33 will be switched to IC_VDD.	R/W	1
30	PHY_EN	PHY transceiver function. PHY will be automatically disabled if entering sleep mode, deep-sleep mode. 0: Disable PHY transceiver function. 1: Enable PHY transceiver function.	R/W	0
29	DPPU_EN	Internal D+ 1.5k pull-up resistor function. 0: Disable internal D+ pull-up resistor. 1: Enable internal D+ pull-up resistor.	R/W	0
28	SIE_EN	USB serial interface engine enable. 0: Disable USB SIE function. 1: Enable USB SIE function.	R/W	0
27	ESD_EN	USB ESD protection enable. 0: Disable ESD protection. 1: Enable ESD protection.	R/W	0
26	DIS_PDEN	Enable internal D+ and D - 175k pull-down resistor. 0: Disable. 1: Enable.	R/W	0
25:7	Reserved		R	0
6	EP7_DIR	Endpoint 7 IN/OUT direction setting. 0: EP7 only handshakes to IN token packet. 1: EP7 only handshakes to OUT token packet.	R/W	0
5	EP6_DIR	Endpoint 6 IN/OUT direction setting.	R/W	0

		0: EP6 only handshakes to IN token packet. 1: EP6 only handshakes to OUT token packet.		
4	EP5_DIR	Endpoint 5 IN/OUT direction setting. 0: EP5 only handshakes to IN token packet. 1: EP5 only handshakes to OUT token packet.	R/W	0
3	EP4_DIR	Endpoint 4 IN/OUT direction setting. 0: EP4 only handshakes to IN token packet. 1: EP4 only handshakes to OUT token packet.	R/W	0
2	EP3_DIR	Endpoint 3 IN/OUT direction setting. 0: EP3 only handshakes to IN token packet. 1: EP3 only handshakes to OUT token packet.	R/W	0
1	EP2_DIR	Endpoint 2 IN/OUT direction setting. 0: EP2 only handshakes to IN token packet. 1: EP2 only handshakes to OUT token packet.	R/W	0
0	EP1_DIR	Endpoint 1 IN/OUT direction setting. 0: EP1 only handshakes to IN token packet. 1: EP1 only handshakes to OUT token packet.	R/W	0

14.9.6 USB Signal Control Register (USB_SGCTL)

Address offset: 0x14

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:3	Reserved		R	0
2	BUS_DRVEN	Enable to drive USB bus. 0: Not drive USB bus. Write operation to BUS_DP or BUS_DN has no effect. 1: Drive USB bus. The D+/D- bus state can be set by set BUS_DP and BUS_DN.	R/W	0
1	BUS_DP	USB D+ state. 0: D+ state is low. 1: D+ state is high.	R/W	0
0	BUS_DN	USB D- state. 0: D- state is low. 1: D- state is high.	R/W	0

14.9.7 USB Endpoint 0 Control Register (USB_EP0CTL)

Address Offset: 0x18

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31	ENDP_EN	Enable Endpoint 0 function. 0: Disable endpoint 0 function. No handshake to endpoint0 SETUP/IN/ OUT token. 1: Enable endpoint 0 function.	R/W	0
30:29	ENDP_STATE[1:0]	Endpoint Handshake State. 00: NAK. 01: ACK. For IN transaction, device will handshake data0/1 to IN transaction. For OUT transaction, device will handshake ACK to OUT token and the following data0/1. After IN/OUT ACK transaction completes, the ENDP_STATE will automatically return to NAK state. 10/11: INOUT_STALL: Device will handshake STALL to both IN or OUT token. ENDP_STATE will automatically return to NAK state after USB Setup transaction has completed.	R/W	00

28	IN_STALL_EN	Enable EP0 to handshake STALL to EP0 IN transaction. 0: Disable 1: Enable IN_STALL_EN enable is only effective to EP0 IN token. The EP0 handshake for EP0 OUT transaction depends on OUT_STALL_EN and ENDP_STATE setting. This bit will be automatically cleared to '0' after USB setup transaction has completed.	R/W	0
27	OUT_STALL_EN	Enable EP0 to handshake STALL to EP0 OUT transaction. 0: Disable 1: Enable OUT_STALL_EN enable is only effective to EP0 OUT token. The EP0 handshake state to EP0 IN transaction depends on IN_STALL_EN and ENDP_STATE setting. This bit will be automatically cleared to '0' after USB setup transaction has completed.	R/W	0
26:7	Reserved	-	R	0
6:0	ENDP_CNT[6:0]	Endpoint Byte Count For IN transaction, the ENDP_CNT indicates the byte count to be uploaded to host. The maximum count for IN transaction should depend on the bMaximumPacketSize0 declaration in USB Device Descriptor and cannot exceed 64 bytes for USB FS device. For OUT transaction, the ENDP_CNT indicates the byte count received from host.	R/W	0

14.9.8 USB Endpoint n Control Register (USB_EPnCTL, n = 1 ~ 7)

Address Offset: 0x1C, 0x20, 0x24, 0x28, 0x2C, 0x30, 0x34

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31	ENDP_EN	EPn function enable bit. 0: Disable EPn function. No handshake to EPn IN/OUT token. 1: Enable EPn function.	R/W	0
30:29	ENDP_STATE[1:0]	Endpoint Handshake State. 00: NAK For IN direction usage, device will handshake NAK to IN token. For OUT direction usage, device will handshake NAK to OUT token. 01: ACK For IN direction usage, device will handshake data0/1 to IN token. For OUT direction usage, device will handshake ACK to OUT token and the following data0/1. After IN/OUT ACK transaction completes, the ENDP_STATE will automatically return to NAK state. 10/11: STALL For IN direction usage, device will handshake STALL to IN token. For OUT direction usage, device will handshake STALL to OUT token and the following data0/1.	R/W	0
28:7	Reserved	-	-	0
6:0	ENDP_CNT[6:0]	Endpoint Byte Count For IN direction usage, the ENDP_CNT indicates the byte count to be uploaded to host. For OUT direction usage, the ENDP_CNT indicates the byte count received from host.	R/W	0

14.9.9 USB Endpoint Data Toggle Register (USB_EPTOGGLE)

Address Offset: 0x3C

Reset value: 0x0000 007F

Bit	Name	Description	Attribute	Reset
31:7	Reserved		R	0
6	EP7_DATA01	0: Clear EP7's toggle bit to DATA0. 1: HW sets toggle bit automatically.	R/W	1
5	EP6_DATA01	0: Clear EP6's toggle bit to DATA0. 1: HW sets toggle bit automatically.	R/W	1
4	EP5_DATA01	0: Clear EP5's toggle bit to DATA0. 1: HW sets toggle bit automatically.	R/W	1
3	EP4_DATA01	0: Clear EP4's toggle bit to DATA0. 1: HW sets toggle bit automatically.	R/W	1
2	EP3_DATA01	0: Clear EP3's toggle bit to DATA0. 1: HW sets toggle bit automatically.	R/W	1
1	EP2_DATA01	0: Clear EP2's toggle bit to DATA0. 1: HW sets toggle bit automatically.	R/W	1
0	EP1_DATA01	0: Clear EP1's toggle bit to DATA0. 1: HW sets toggle bit automatically.	R/W	1

14.9.10 USB Endpoint n Buffer Offset Register (USB_EPnBUFOS, n = 1 ~ 7)

Address Offset: 0x48, 0x4C, 0x50, 0x54, 0x58, 0x5C, 0x60

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:2	OFFSET[5:0]	The offset address for each endpoint data buffer. The effective offset address is: USB_SRAM address + {EPnBUFOS[7:2], 2'b00} Where USB_SRAM address = USB_BA + 0x100 For endpoint 0, the offset address is fixed as USB_SRAM address.	R/W	40, 80, C0, 100, 140, 180, 1C0
1:0	Reserved		R	0

14.9.11 USB Frame Number Register (USB_FRMNO)

Address Offset: 0x64

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:11	Reserved		R	0
10:0	FRAME_NO[10:0]	The 11-bit frame number of the Start-Of-Frame(SOF) packet. This number is updated by H/W automatically when SOF packet is received.	R	0

14.9.12 USB PHY Parameter Register (USB_PHYPRM)

Address Offset: 0x68

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:26	PHY_PARAM[5:0]	The USB PHY parameter value. The suggested settings would be 0x20.	R/W	0
25:0	Reserved		R	0

14.9.13 USB PHY Parameter Register 2(USB_PHYPRM2)

Address Offset: 0x70

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:15	Reserved		R	0
14:0	PHY_PARAM2[14:0]	The USB PHY parameter value.	R/W	0

14.9.14 PS/2 Control Register (USB_PS2CTL)

Address Offset: 0x74

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31	PS2ENB	PS/2 internal 5kohm pull-up resistor control bit.	R/W	0
30:4	Reserved		R	0
3	SDA	PS/2 SDA data buffer.	R/W	0
2	SCK	PS/2 SCK data buffer.	R/W	0
1	SDAM	PS/2 SDA mode control bit.	R/W	0
0	SCKM	PS/2 SCK mode control bit.	R/W	0

14.9.15 USB Read/Write Address Register (USB_RWADDR)

Address Offset: 0x7C

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:2	RWADDR[5:0]	USB FIFO address to be read or written from/to USB FIFO.	R/W	0
1:0	Reserved		R	0

14.9.16 USB Read/Write Data Register (USB_RWDATA)

Address Offset: 0x80

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:0	RWDATA[31:0]	Data to be read or written from/to USB FIFO.	R/W	0

14.9.17 USB Read/Write Status Register (USB_RWSTATUS)

Address Offset: 0x84

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1	R_STATUS	Read status of USB FIFO. *If F/W is to read the data from USB FIFO, set this bit as '1 '. When hardware has completed the read action (RWDATA content has been written by the new data read from USB FIFO with address RWADDR.), this bit is automatically cleared as '0' by hardware.	R/W	0
0	W_STATUS	Write status of USB FIFO. *If F/W is to write data into USB FIFO, set this bit as '1 '. When hardware has completed the write action (RWDATA content has been read as the new data, and the new data is written into USB FIFO with address RWADDR.), this bit is automatically cleared as '0' by hardware.	R/W	0

14.9.18 USB Read/Write Address Register2 (USB_RWADDR2)

Address Offset: 0x88

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:8	Reserved		R	0
7:2	RWADDR[5:0]	USB FIFO address to be read or written from/to USB FIFO.	R/W	0
1:0	Reserved		R	0

14.9.19 USB Read/Write Data Register2 (USB_RWDATA2)

Address Offset: 0x8C

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:0	RWDATA[31:0]	Data to be read or written from/to USB FIFO.	R/W	0

14.9.20 USB Read/Write Status Register 2(USB_RWSTATUS2)

Address Offset: 0x90

Reset value: 0x0000 0000

Bit	Name	Description	Attribute	Reset
31:2	Reserved		R	0
1	R_STATUS	Read status of USB FIFO. *If F/W is to read the data from USB FIFO, set this bit as '1 '. When hardware has completed the read action (RWDATA content has been written by the new data read from USB FIFO with address RWADDR.), this bit is automatically cleared as '0' by hardware.	R/W	0
0	W_STATUS	Write status of USB FIFO. *If F/W is to write data into USB FIFO, set this bit as '1 '. When hardware has completed the write action (RWDATA content has been read as the new data, and the new data is written into USB FIFO with address RWADDR.), this bit is automatically cleared as '0' by hardware.	R/W	0

15 FLASH

15.1 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.

15.2 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	128K Bytes divided into 128 pages of 1024 Bytes
Boot ROM	4K Bytes divided into 4 pages of 1024 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.

15.3 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.

15.4 ORGANIZATION

Block	Name	Base Address	Size (Byte)
User ROM	Page 0	0x00000000 ~ 0x000003FF	1024
	Page 1	0x00000400 ~ 0x000007FF	1024
	. . .	. . .	. . .
	Page 127	0x0001FC00 ~ 0x0001FFFF	1024
Boot Loader	Page 0	0x1FFF0000 ~ 0x1FFF03FF	1024
	Page 1	0x1FFF0400 ~ 0x1FFF07FF	1024
	Page 2	0x1FFF0800 ~ 0x1FFF0BFF	1024
	Page 3	0x1FFF0C00 ~ 0x1FFF0FFF	1024

15.5 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.

15.6 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.

15.7 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.

15.8 FLASH MEMORY CONTROLLER (FMC)

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

15.8.1 CODE SECURITY (CS)

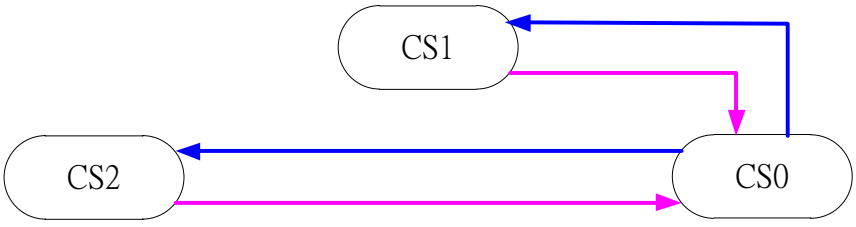
Code Security is a mechanism that allows the user to enable different levels of security in the system so that access to the on-chip Flash and use of the ISP can be restricted.

✸ ***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 CS3 to CS0, 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.***



includes:
- New Code Security bytes programming

includes:
- Erase Code Security bytes
- Mass Erase User ROM

15.8.2 PROGRAM FLASH MEMORY

The Flash memory can be programmed 64 bits (8 bytes) at a time, and SHALL program more or equal to 8 bytes once. CPU can program the main Flash memory by performing standard word write operations. The PG bit in the FLASH_CTRL register must be set. FMC preliminarily reads the value at the addressed main Flash memory location and checks that it has been erased. If not, the program operation is skipped and a warning is issued by the PGERR bit in FLASH_STATUS register. The end of the program operation is indicated by the EOP bit in the FLASH_STATUS register.

The main Flash memory programming sequence in standard mode is as follows:
1. Set the PG bit in the FLASH_CTRL register.
2. Perform the data write at the desired address.
3. Wait for the BUSY bit to be reset.
4. (Optional) Read the programmed value and verify.

✸ ***Note: User SHALL fill in more or equal to 8 bytes to FLASH_DATA register when programming the ROM.***

15.8.3 ERASE

The Flash memory can be erased page by page or completely (Mass Erase).

15.8.3.1 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 STARTE bit in the FLASH_CTRL register
4. Wait for the BUSY bit to be reset
5. (Optional) Read the erased page and verify

15.8.3.2 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.

15.9 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.

15.10 HW CHECKSUM

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

15.11 FMC REGISTERS

Base Address: 0x4006 2000

15.11.1 Flash Low Power Control register (FLASH_LPCTRL)

Address offset: 0x00

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 enable bit 000000b: HCLK<24MHz 101001b: 24MHz≦HCLK≦48MHz 111001b: HCLK>48MHz Other: Reserved (May cause unexpected error to force MCU enter Hard fault handler)	R/W	0

15.11.2 Flash Status register (FLASH_STATUS)

Address offset: 0x04

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. ✸ The address to be programmed contains a value different from 0xFFFFFFFF before programming.	R/W	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

15.11.3 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. 1: Triggers Checksum calculation. 0: Checksum calculation is done.	R/W	0
6	START	Start Erase/Program operation. 1: Triggers an Erase/Program operation when set. This bit is set only by SW and resets when the BUSY bit resets. 0: Erase process is done.	R/W	0
5:3	Reserved		R	0

2	MER	Mass erase chosen mode bit. Erase of all user pages chosen. This bit is set only by SW and reset when the BUSY bit resets.	R/W	0
1	PER	Page Erase chosen. This bit is set only by SW and reset when the BUSY bit resets.	R/W	0
0	PG	Flash Programming chosen. This bit is set only by SW and reset when the BUSY bit resets.	R/W	0

15.11.4 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.

✸ ***Note: User SHALL fill in more or equal to 8 bytes to FLASH_DATA register when programming the ROM.***

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

15.11.5 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.

✸ ***Note:***
1. Write access to this register is blocked when the BUSY bit in the FLASH_STATUS register is set.
2. Value must be multiples of 8 in the FLASH_ADDR register.

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

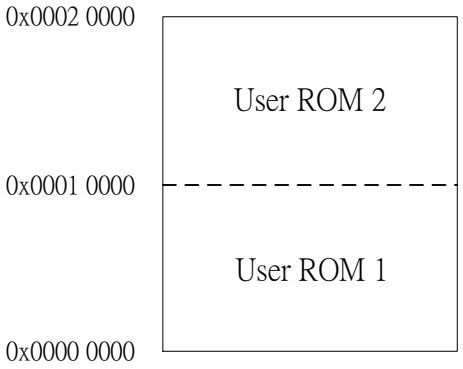
15.11.6 Flash Checksum register (FLASH_CHKSUM)

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

15.11.7 Flash Checksum register 1 (FLASH_CHKSUM1)

Address offset: 0x18



Bit	Name	Description	Attribute	Reset
31:16	UR2CHKSUM[15:0]	Checksum of User ROM 2.	R	0
15:0	UR1CHKSUM[15:0]	Checksum of User ROM 1.	R	0

16 SERIAL-WIRE DEBUG (SWD)

16.1 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.

16.2 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.

16.3 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.	

16.4 DEBUG NOTE

16.4.1 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.

16.4.2 DEBUG RECOVERY

User code may disable SWD function in order to use P3.5 and P3.6 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 P2.2 (BOOT pin) during boot procedure. If P2.2 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 P2.2 as other functions such as GPIO.

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

16.4.3 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

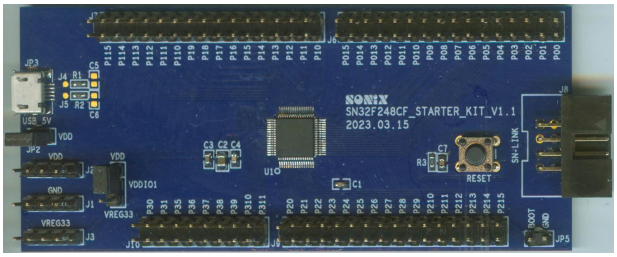
Once a SWD function is disabled by SW, the GPIO controller takes control again.

17 DEVELOPMENT TOOL

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

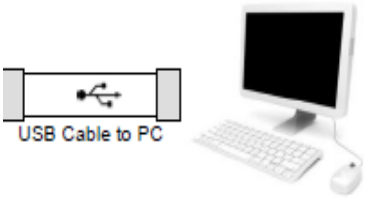
SONiX 32-bit series Embedded ICE Emulator System includes:

- 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: 2.5V~5.5V.
- Up to 4 hardware break points.
- System clock rate up to 48MHz.
- Oscillator supports IHRC, ILRC.

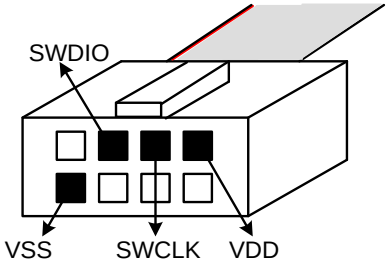
SONiX 32-bit series Embedded ICE Emulator Limitation:

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

17.1 SN-LINK-V3

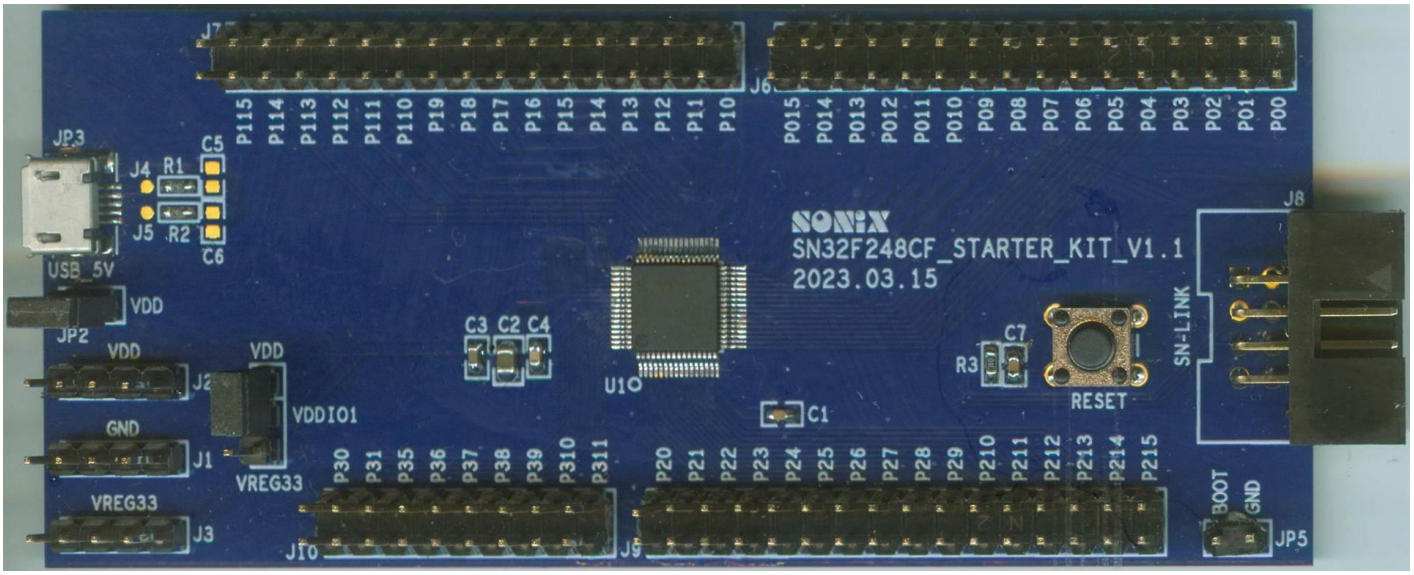
SN-LINK-V3 is a high speed emulator for SONiX 32-bit MCU. It debugs and programs based on SWD protocol. In addition to debugger functions, the SN-LINK-V3 also may be used as a programmer to load firmware from PC to MCU for engineering production, even mass production.

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



17.2 SN32F240C 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.



- JP3 : Micro USB connector.
- JP2 : USB Power connector.
- JP1 : VDDIO1 power connector: Choose the source of VDDIO1 (P0.0~P0.7) (5.0V/3.3V on board).
- J1 : GND connector.
- J2 : VDD connector.
- J3 : VREG33 output connector.
- U1 : SN32F248CF real chip.
- RESET button : External reset trigger source.
- J8 : SN-LINK connector
- JP5 : Short to force MCU stay in Boot loader.

18 ELECTRICAL CHARACTERISTIC

18.1 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 ~ + 85°C
Storage ambient temperature (Tstor) ..... –40°C ~ + 125°C

18.2 ELECTRICAL CHARACTERISTIC

Standard Operating Conditions (Typical temperature Ta = 25℃)
Operating Temperature -40℃≦ Ta ≦ +85℃ for Industrial Class
The below data covers process corner range (SS~TT~FF).

PARAMETER	SYM.	DESCRIPTION		MIN.	TYP.	MAX.	UNIT
Operating Voltage	Vdd1	Supply voltage for core and external rail		2.5	3.3	5.5	V
	Vdd2	USB mode		3.1	5.0	5.25	V
VDD rise rate	V_{POR}	VDD rise rate to ensure internal power-on reset		0.05	-	-	V/ms
VDDIO1 Voltage	V_{DDIO1}	I/O driver power for P0.0~P0.7		1.8		Vdd	V
Power Consumption							
Supply Current	Idd1	Normal mode	System clock = 12MHz [1][2][3]	-	3.5	-	mA
			System clock = 24MHz [1][3][4]	-	6	-	mA
			System clock = 48MHz [1][3][4]	-	11	-	mA
		Normal mode(USB Mode)	System clock = IHRC 12MHz [3][6][7]	-	6.3	-	mA
			System clock = PLL 12MHz [3][6][7]	-	6.5	-	mA
			System clock = PLL 24MHz [3][6][7]	-	9	-	mA
			System clock = PLL 48MHz [3][6][7]	-	12	-	mA
	Idd2	Sleep Mode(USB Mode)	System clock = 32KHz [3][6][8]	-	200	-	uA
	Idd3	Deep-sleep Mode	Vdd=3.3V [1][3][5]	-	2	-	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		40	50	60	KΩ
		Vin = Vss, Vdd = 3.3V		50	75	100	KΩ
I/O High-level output source current	I_{OH}	V_{OP} = Vdd – 0.5V		12	20	-	mA
I/O Low-level output sink current	I_{OL1}	Standard port and RESET pins, except P0.8~P0.15, P1.0~P1.11	V_{OP} = Vss + 0.5V	12	20	-	mA
	I_{OL2}	P0.8~P0.15, P1.0~P1.11	V_{OP} = Vss + 1.5V	-	420	-	mA
ADC							

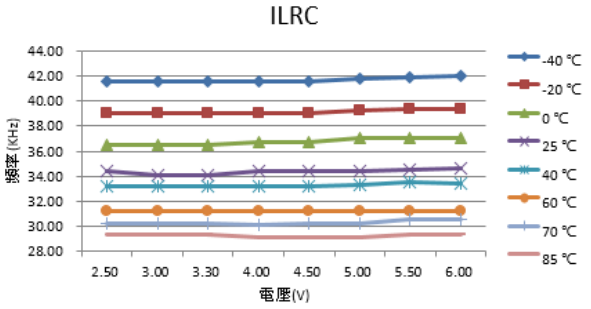
ADC Operating Voltage	V_{ADC}			2.5		5.5	V
External Reference Voltage	V_{EREF}	Vdd=5.0V		2.5	-	Vdd	V
Internal VDD Reference Voltage	V_{IREF1}	Vdd=2.5V~5.0V		-	Vdd	-	V
Internal 4.5V Reference Voltage	V_{IREF2}	Vdd=5.0V		4.41	4.5	4.59	V
Internal 3V Reference Voltage	V_{IREF3}	Vdd=3.5V~5.0V		2.94	3	-3.06	V
Internal 2V Reference Voltage	V_{IREF4}	Vdd=2.5V~5.0V		1.96	2	2.04	V
AVREFH Pin Input Voltage	V_{IREFH}	Vdd=5.0V		2	-	Vdd	V
ADC Current Consumption	I_{ADC}	Vdd=5.0V		-	100	-	uA
Resolution	N_r	No missing code.		10	11	12	Bit
AIN0 ~ AIN15 Input Voltage	V_{AIN}			0	-	V_{REFH}	V
Integral Nonlinearity	INL	Vdd=5.0V		-1	-	+1	LSB
Differential Nonlinearity	DNL	Vdd=5.0V		-1	-	+1	LSB
ADC Clock Frequency	F_{ADCLK}	Vdd=5.0V		-	-	12M	Hz
ADC Offset Voltage	V_{OFFSET}	Non-trimmed		-10	0	+10	mV
		Trimmed		-2	0	+2	mV
ADC Enable Time	T_{ADEN}	Ready to start convert after set ADENB = “1”		100	-	-	V
ADC Conversion Cycle Time	F_{ADCYL}	Vdd=2.5V~5.5V		64	-	-	$1/F_{ADCLK}$
FLASH							
Supply Voltage	Vdd1			1.35	1.50	1.65	V
Endurance Time	T_{EN}	Erase + Program		20K	100K	-	Cycle
Page Erase Cent	I_{PER}			-	3	5	mA
Program Current	I_{PG}			-	3	5	mA
Page Erase Time	T_{PE}	1-Page (1024 bytes)		-	1	2	ms
Mass Erase Time	T_{MER}			-	7	10	ms
1-Word Programming Time	T_{PG}	1 -Word (64 bits)		-	10	20	us
MISC							
Low Voltage Detector	LVD	Interrupt/Reset	Level 0	2.10	2.20	2.30	V
			Level 1	2.60	2.70	2.80	V
			Level 2	3.50	3.60	3.70	V
IHRC Freq.	F_{IHRC}	*T=25*℃, Vdd=2.5V~5.5V		11.76	12	12.24	MHz
		T=-40℃*~85*℃, Vdd=2.5V~5.5V		11.70	12	12.30	MHz
3.3V Regulator Output voltage	V_{REG33}	VCC ≧ 3.60V		3.0	-	3.4	V
ESD_HBM	V_{ESD_HBM}	ESD human body mode		2000	-	-	V
ESD_MM	V_{ESD_MM}	ESD machine mode		200	-	-	V

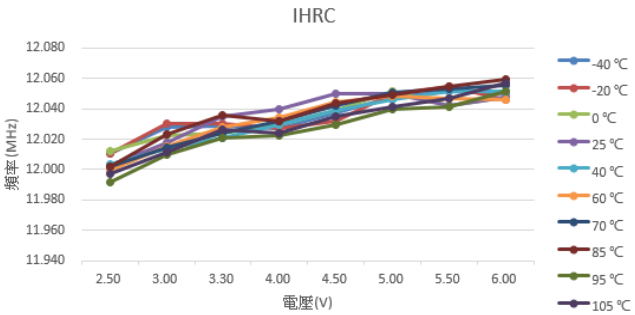
*** Parameters with star mark are non-verified design reference.**

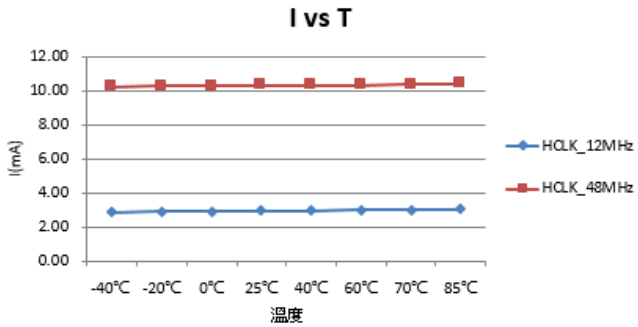
[1] IDD measurements were performed with all pins configured as GPIO outputs driven LOW and pull-up resistors disabled and Vdd=3.3V
[2] IHRC and ILRC are enabled, and PLL is disabled.
[3] LVD and all peripherals are disabled.
[4] IHRC is disabled, and PLL is enabled.
[5] All oscillators and analog blocks are turned off.
[6] Idd measurements were performed with all pins configured as GPIO outputs driven LOW and pull-up resistors disabled and Vdd=5V
[7] IHRC and ILRC are enabled, and PLL is enabled.
[8] IHRC is disabled, ILRC is enabled, and PLL is disabled.

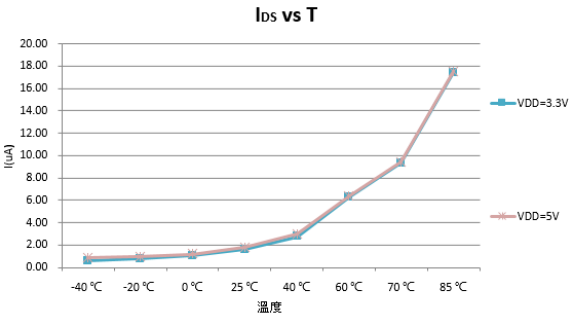
18.3 CHARACTERISTIC GRAPHS

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.









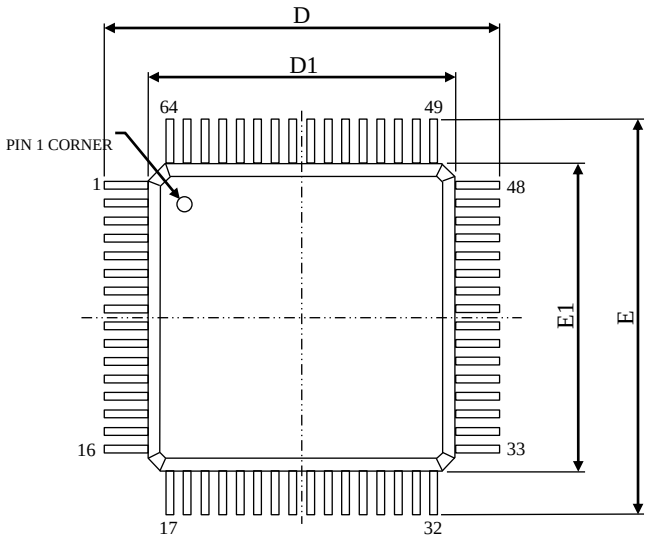
*Supply Current v.s. Temperature (Operating Conditions: All pins configured as GPIO outputs driven Low and pull-up resistors disabled)

19 FLASH ROM PROGRAMMING PIN

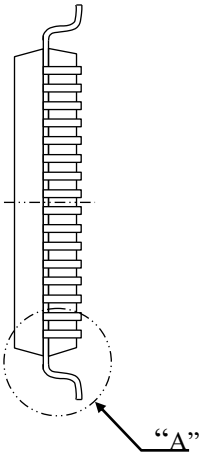
Programming Information of SN32F240C Series									
Chip Name		SN32F248CF		SN32F247CF		SN32F246CJ		SN32F2451CJ	
Writer Connector JP5		Flash IC / JP3 Pin Assignment							
Number	Name	Number	Pin	Number	Pin	Number	Pin	Number	Pin
1	VDD	*16, 33, 55*	VDD	*14, 25, 40*	VDD	*11,22, 37*	VDD	*8,16, 27*	VDD
2	GND	*56*	VSS	*41*	VSS	*38*	VSS	*15, 28*	VSS
3	CLK	*61*	P3.9	*46*	P3.9	*43*	P3.9	*1*	P3.9
4	CE								
5	PGM	*58*	P3.6	*43*	P3.6	*40*	P3.6	*30*	P3.6
6	OE	*57*	P3.5	*42*	P3.5	*39*	P3.5	*29*	P3.5
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	*60*	P3.8	*45*	P3.8	*42*	P3.8	*32*	P3.8

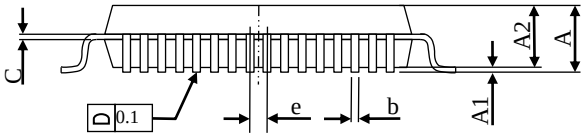
20 PACKAGE INFORMATION

20.1 LQFP 64 PIN

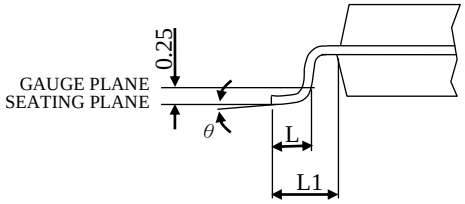


TOP VIEW





SIDE VIEW



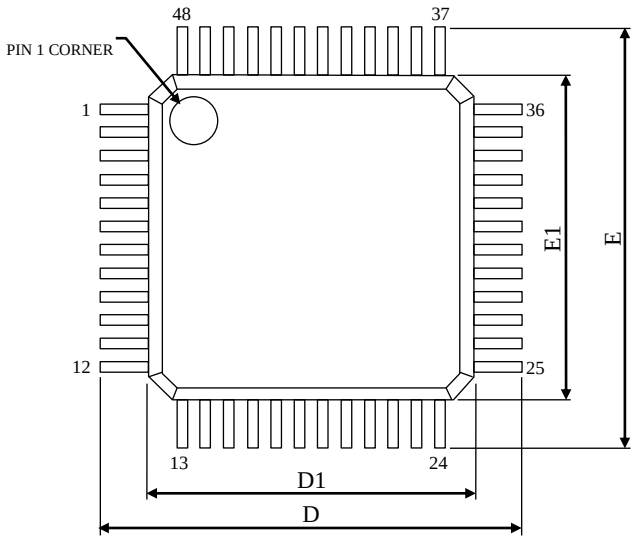
DETAIL "A"

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	0.13	0.19	0.25	0.005	0.007	0.010
c	0.09	--	0.20	0.004	--	0.008
D	9.00 BSC			0.354 BSC		
D1	7.00 BSC			0.276 BSC		
e	0.40 BSC			0.016 BSC		
E	9.00 BSC			0.354 BSC		
E1	7.00 BSC			0.276 BSC		
L	0.4	0.60	0.8	0.016	0.024	0.032
L1	1.00 REF			0.039 REF		
θ	0˚	3.5˚	7˚	0˚	3.5˚	7˚

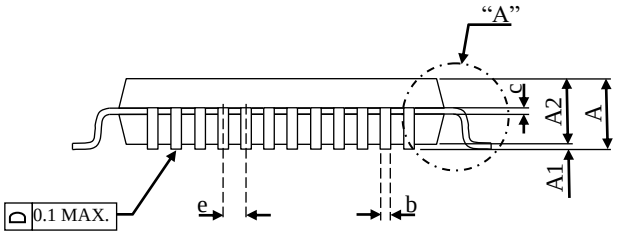
Notes：

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

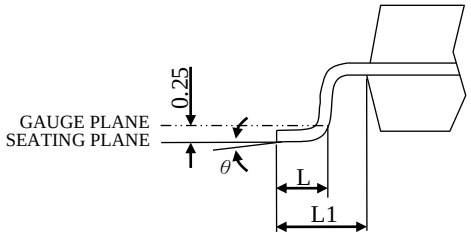
20.2 LQFP 48 PIN



TOP VIEW



SIDE VIEW



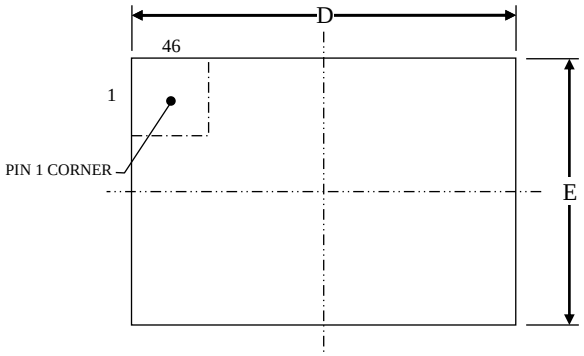
DETAIL "A"

SYMBOLS	Dimension in mm			Dimension in inch		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	--	--	1.60	--	--	0.063
A1	0.05	--	0.15	0.002	--	0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
b	0.17	0.22	0.27	0.007	0.009	0.011
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	0.50 BSC			0.020 BSC		
L	0.40	0.60	0.80	0.016	0.024	0.031
L1	1.00 REF			0.039 REF		
θ	0°	3.5°	7°	0°	3.5°	7°

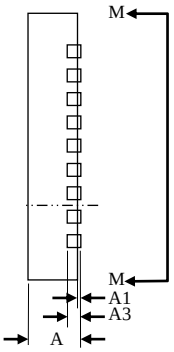
Notes：

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

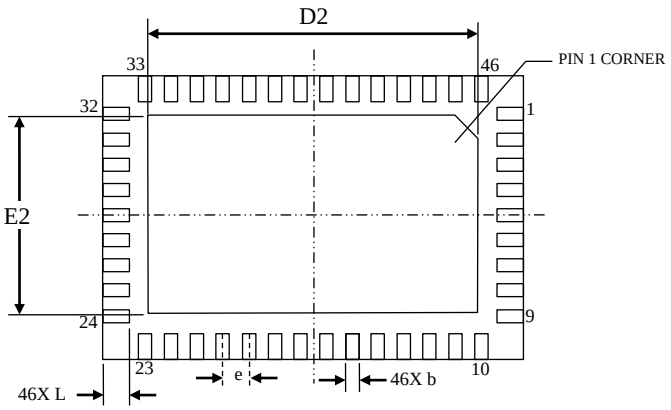
20.3 QFN 46 PIN



TOP VIEW



SIDE VIEW



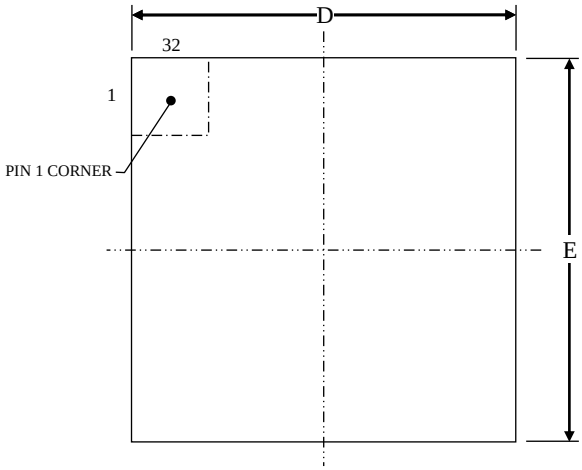
BOTTOM VIEW
VIEW M－M

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	0.001	0.002
A3	0.203 REF			0.008 REF		
b	0.20	0.25	0.30	0.008	0.010	0.012
D	6.50 BSC			0.256 BSC		
E	4.50 BSC			0.177 BSC		
e	0.40 BSC			0.016 BSC		
D2	5.00	5.10	5.20	0.197	0.201	0.205
E2	3.00	3.10	3.20	0.118	0.122	0.126
L	0.30	0.40	0.50	0.012	0.016	0.020

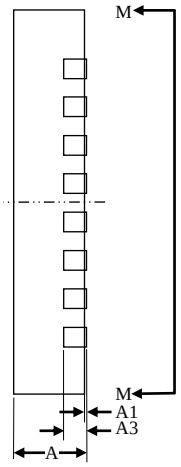
Notes：

1. CONTROLLING DIMENSION：MILLIMETER (mm)

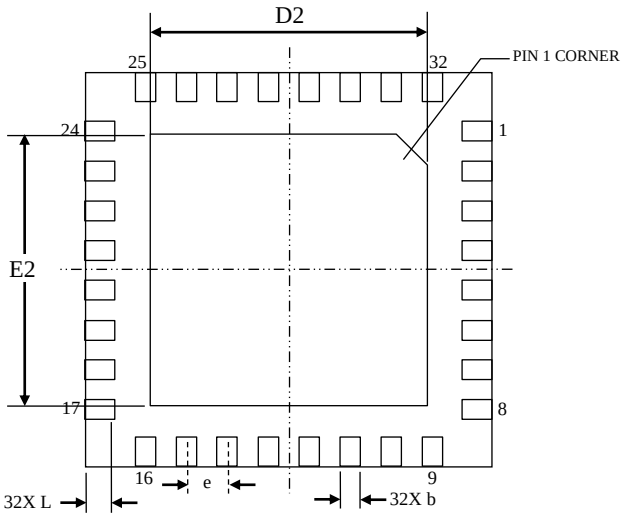
20.4 QFN 32 PIN 4x4



TOP VIEW



SIDE VIEW



BOTTOM VIEW
VIEW M－M

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	2.90	0.080	0.096	0.114
E2	2.00	2.45	2.90	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)

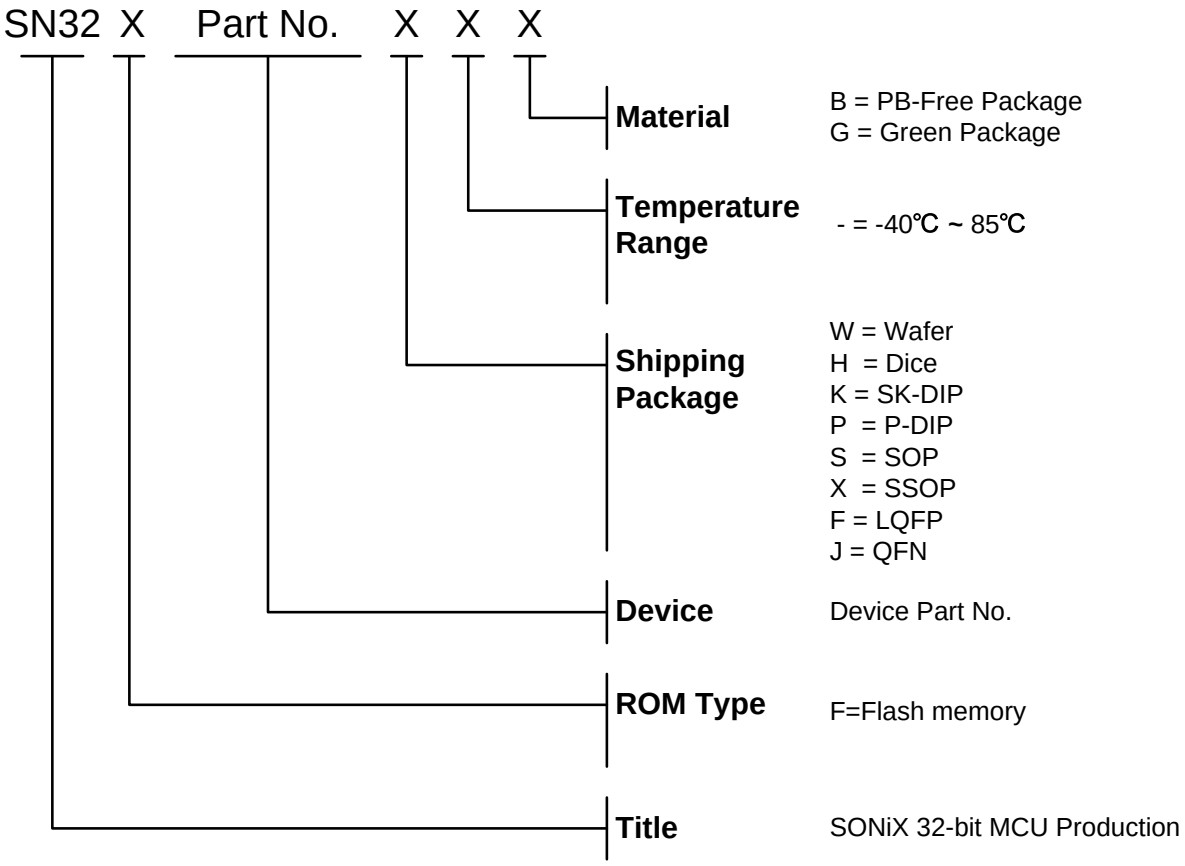
21 MARKING DEFINITION

21.1 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.

21.2 MARKING INDETIFICATION SYSTEM

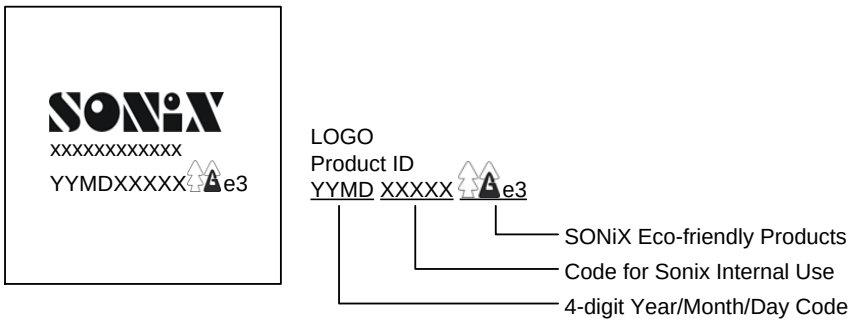


21.3 MARKING EXAMPLE

Name	ROM Type	Device	Package	Temperature	Material
SN32F248CFG	Flash memory	248C	LQFP	-40℃~85℃	Green Package
SN32F247CFG	Flash memory	248C	LQFP	-40℃~85℃	Green Package
SN32F246CJG	Flash memory	248C	QFN	-40℃~85℃	Green Package
SN32F2451CJG	Flash memory	248C	QFN	-40℃~85℃	Green Package
SN32F248CW	Flash memory	248C	Wafer	-40℃~85℃	-
SN32F248CH	Flash memory	248C	Dice	-40℃~85℃	-

21.4 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.



Example of Device Marking

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