32-Bit Arm® Cortex®-M0+ 5V Microcontroller

HT32F50020/HT32F50030
User Manual

Revision: V1.00 Date: February 08, 2023

www.holtek.com

Table of Contents

List of Tables

List of Figures

1 Introduction

Overview

This user manual provides detailed information including how to use the HT32F50020/HT32F50030 devices, system and bus architecture, memory organization and peripheral instructions. The target audiences for this document are software developers, application developers and hardware developers. For more information regarding pin assignment, package and electrical characteristics, please refer to the HT32F50020/HT32F50030 datasheet.

The devices are high performance and low power consumption 32-bit microcontrollers based around an Arm® Cortex®-M0+ processor core. The Cortex®-M0+ is a next-generation processor core which is tightly coupled with Nested Vectored Interrupt Controller (NVIC), SysTick timer and advanced debug support.

The devices operate at a frequency of up to 16 MHz to obtain maximum efficiency. It provides up to 32 KB of embedded Flash memory for code/data storage and up to 2 KB of embedded SRAM memory for system operation and application program usage. A variety of peripherals, such as ADC, I^2C, UART, SPI, SCTM, BFTM, LEDC, RTC, WDT and SW-DP (Serial Wire Debug Port), etc., are also implemented in the device series. Several power saving modes provide the flexibility for maximum optimization between wakeup latency and power consumption, which is an especially important consideration in low power applications.

The above features ensure that the devices are suitable for use in a wide range of applications, especially in areas such as white goods application control, power monitors, alarm systems, consumer products, handheld equipment, data logging applications, motor control and so on.

Features

- Core
  - 32-bit Arm® Cortex®-M0+ processor core
  - Up to 16 MHz operating frequency
  - Single-cycle multiplication
  - Integrated Nested Vectored Interrupt Controller (NVIC)
  - 24-bit SysTick timer
- On-Chip Memory
  - Up to 32 KB on-chip Flash memory for instruction/data and option storage
  - Up to 2 KB on-chip SRAM
  - Supports multiple booting modes
- Flash Memory Controller – FMC
  - 32-bit word programming with In System Programming (ISP) and In Application Programming (IAP)
  - Flash protection capability to prevent illegal access

- Reset Control Unit – RSTCU
  - Supply supervisor:
    - Power-On Reset / Power-Down Reset – POR / PDR
    - Brown-Out Detector – BOD
    - Programmable Low Voltage Detector – LVD
- Clock Control Unit – CKCU
  - External 4 to 16 MHz crystal oscillator
  - External 32,768 Hz crystal oscillator
  - Internal 16 MHz RC oscillator trimmed to ±1 % accuracy at 25 °C operating temperature
  - Internal 32 kHz RC oscillator
  - Independent clock divider and gating bits for peripheral clock sources
- Power Management – PWRCU
  - Single V_{DD} power supply: 2.5 V to 5.5 V
  - Integrated 1.5 V LDO regulator for MCU core, peripheral and memory power supply
  - Two power domains: V_{DD} and V_{CORE}
  - Three power saving modes: Sleep, Deep-Sleep1 and Deep-Sleep2 modes
- External Interrupt/Event Controller – EXTI
  - Up to 8 EXTI lines with configurable trigger source and type
  - All GPIO pins can be selected as EXTI trigger source
  - Source trigger type can be high level, low level, negative edge, positive edge or both edges
  - Individual interrupt enable, wakeup enable and status bits for each EXTI line
  - Software interrupt trigger mode for each EXTI line
  - Integrated deglitch filter for short pulse blocking
- Analog to Digital Converter – ADC
  - 12-bit SAR ADC engine
  - Up to 500 ksps conversion rate
  - Up to 12 external analog input channels
- I/O Ports – GPIO
  - Up to 42 GPIOs
  - Port A, B, C, F are mapped on 16 external interrupts – EXTI
  - Almost all I/O pins have configurable output driving current
- Single-Channel Timer – SCTM
  - 16-bit auto-reload up-counter
  - One channel for each timer
  - 8-bit programmable prescaler that allows division of the prescaler clock source by any factor between 1 and 256 to generate the counter clock frequency
  - One Input Capture function
  - Two Compare Match Output
  - PWM waveform generation with Edge-aligned
  - Single Pulse Mode Output
- Basic Function Timer – BFTM
  - 16-bit compare/match count-up counter – no I/O control
  - One shot mode – counter stops counting when compare match occurs
  - Repetitive mode – counter restarts when compare match occurs

- Watchdog Timer – WDT
  - 12-bit down-counter with a 3-bit prescaler
  - Provides reset to the system
  - Programmable watchdog timer window function
  - Register write protection function
- Real-Time Clock – RTC
  - 24-bit up-counter with a programmable prescaler
  - Alarm function
  - Interrupt and wakeup event
- Inter-integrated Circuit – I^2C
  - Supports both master and slave modes with a frequency of up to 1 MHz
  - Supports 7-bit addressing mode and general call addressing
  - Supports two 7-bit slave addresses
- Serial Peripheral Interface – SPI
  - Supports both master and slave mode
  - Frequency of up to (f_{PCLK}/2) MHz for master mode and (f_{PCLK}/3) MHz for slave mode
  - Programmable data frame length up to 8 bits
  - FIFO Depth: 4 levels
- Universal Asynchronous Receiver Transmitter – UART
  - Asynchronous serial communication operating baud rate clock frequency of up to (f_{PCLK}/16) MHz
  - Capability of full duplex communication
  - Fully programmable serial communication characteristics including
    - Word length: 7, 8 or 9-bit character
    - Parity: Even, odd or no-parity bit generation and detection
    - Stop bit: 1 or 2 stop bits generation
    - Bit order: LSB-first or MSB-first transfer
  - Error detection: Parity, overrun and frame error
- LED Controller – LEDC
  - Supports 8-segment digital displays up to 8
  - Supports 8-segment digital displays with common anode or common cathode
  - Supports frame interrupt
  - Three frequency sources: LSI, LSE and PCLK
  - The LED light on/off times can be controlled using the dead time setting
- Debug Support
  - Serial Wire Debug Port – SW-DP
  - 4 comparators for hardware breakpoint or code/literal patch
  - 2 comparators for hardware watchpoints
- Package and Operation Temperature
  - 24 / 28-pin SSOP, 24 / 32 / 46-pin QFN and 48-pin LQFP packages
  - Operation temperature range: -40 ºC to 85 ºC

Device Information

Table 1. Features and Peripheral List

Peripherals		HT32F50020	HT32F50030
Main Flash (KB)		16	31
Option Bytes Flash (KB)		1	1
SRAM (KB)		2	
Timers	SCTM	3	
	BFTM	1	
	WDT	1	
	RTC	1	
Communication	SPI	1	
	UART	2	
	I^2C	1	
EXTI		8	
12-bit ADC		1	
Number of channels		Max.12 Channels	
GPIO		Up to 42	
LEDC		Up to 8 × 8-segment	
CPU frequency		Up to 16 MHz	
Operating voltage		2.5 V ~ 5.5 V	
Operating temperature		-40 °C ~ 85 °C	
Package		24 / 28-pin SSOP, 24 / 32 / 46-pin QFN and 48-pin LQFP	

Block Diagram

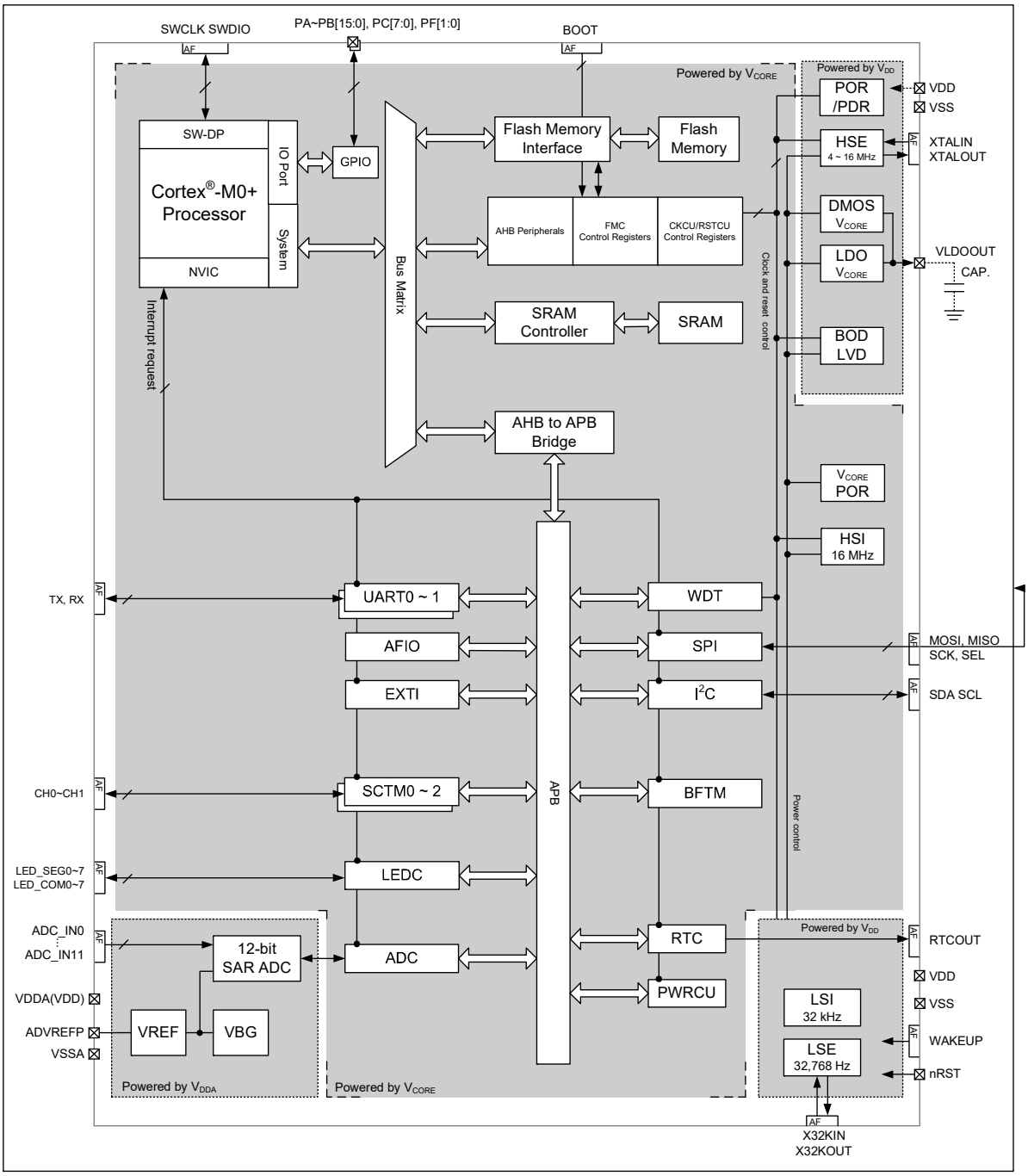


Figure 1. Block Diagram

2 Document Conventions

The conventions used in this document are shown in the following table.

Table 2. Document Conventions

Notation	Example	Description
0x	0x5a05	The number string with a 0x prefix indicates a hexadecimal number.
0xnnnn_nnnn	0x2000_0100	32-bit Hexadecimal address or data.
b	b0101	The number string with a lowercase b prefix indicates a binary number.
NAME [n]	ADDR [5]	Specific bit of NAME. NAME can be a register or field of register. For example, ADDR [5] means bit 5 of ADDR register (field).
NAME [m:n]	ADDR [11:5]	Specific bits of NAME. NAME can be a register or field of register. For example, ADDR [11:5] means bit 11 to 5 of ADDR register (field).
X	b10X1	Don't care notation which means any value is allowed.
RW	19 18 HSEEN | PLLEN RW 0 RW 0	Software can read and write to this bit.
RO	3 2 HSIRDY | HSERDY RO 1 RO 0	Software can only read this bit. A write operation will have no effect.
RC	1 0 PDF | PORF RC 0 RC 1	Software can only read this bit. A read operation will clear it to 0 automatically.
WC	3 2 | CKSF WC 0 WC 0	Software can read this bit or clear it by writing 1. Writing 0 to it will have no effect.
W0C	1 0 | MIF W0C 0	Software can read this bit or clear it by writing 0. Writing 1 to it will have no effect.
WO	1 0 CH1CCG | CH0CCG WO 0 WO 0	Software can only write to this bit. A read operation always returns 0.
Reserved	7 6 | Reserved	Reserved bit(s) for future use. Data read from these bits is not well defined and should be treated as random data. Normally these reserved bits should be cleared to a 0 value. Note that reserved bit must be kept at reset value.
Word		Data length of a word is 32-bit.
Half-word		Data length of a half-word is 16-bit.
Byte		Data length of a byte is 8-bit.

3 System Architecture

The system architecture of the device that includes the Arm® Cortex®-M0+ processor, bus architecture and memory organization will be described in the following sections. The Cortex®-M0+ is a next generation processor core which offers many new features. Integrated and advanced features make the Cortex®-M0+ processor suitable for market products that require microcontrollers with high performance and low power consumption. In brief, The Cortex®-M0+ processor includes the AHB-Lite bus interface. All memory access of the Cortex®-M0+ processor are executed on the AHB-Lite bus according to the different purposes and the target memory spaces. The memory organization uses a Harvard architecture, pre-defined memory map and up to 4 GB of memory space, making the system flexible and extendable.

Arm® Cortex®-M0+ Processor

The Cortex®-M0+ processor is a very low gate count, highly energy efficient processor that is intended for microcontroller and deeply embedded applications that require an area optimized, low power processor. The processor is based on the ARMv6-M architecture and supports Thumb® instruction sets, single-cycle I/O ports, hardware multiplier and low latency interrupt respond time. Some system peripherals listed below are also provided by Cortex®-M0+:

- Internal Bus Matrix connected with AHB-Lite Interface, Single-cycle I/O ports and Debug Access Port (DAP)
- Nested Vectored Interrupt Controller (NVIC)
- Optional Wakeup Interrupt Controller (WIC)
- Breakpoint and Watchpoint Unit
- Optional Memory Protection Unit (MPU)
- Serial Wire debug Port (SW-DP)
- Optional Micro Trace Buffer Interface (MTB)

The following figure shows the Cortex®-M0+ processor block diagram. For more information, refer to the Arm® Cortex®-M0+ Technical Reference Manual.

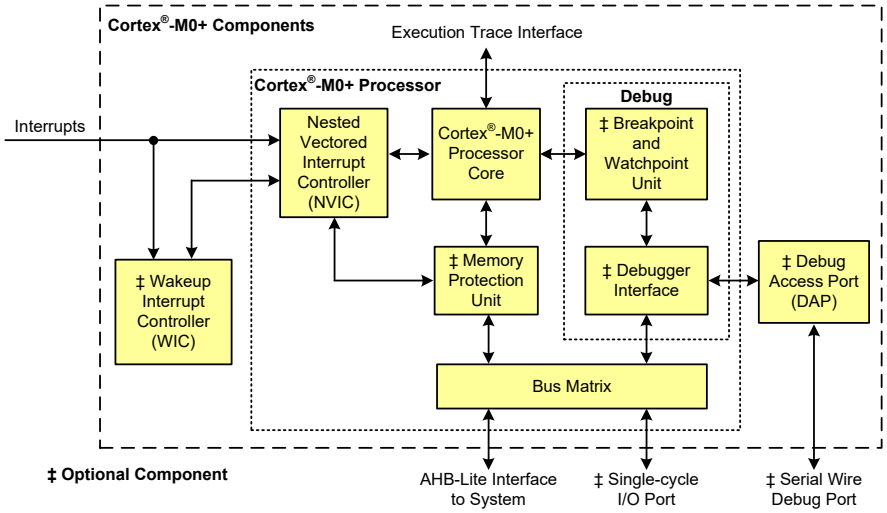


Figure 2. Cortex®-M0+ Block Diagram

Bus Architecture

The devices consist of one master and four slaves in the bus architecture. The Cortex®-M0+ AHB-Lite bus is the master while the internal SRAM access bus, the internal Flash memory access bus, the AHB peripheral access bus and the AHB to APB bridge are the slaves. The single 32-bit AHB-Lite system interface provides simple integration to all system regions include the internal SRAM region and the peripheral region. All of the master buses are based on 32-bit Advanced High-performance Bus-Lite (AHB-Lite) protocol. The following figure shows the bus architecture of the devices.

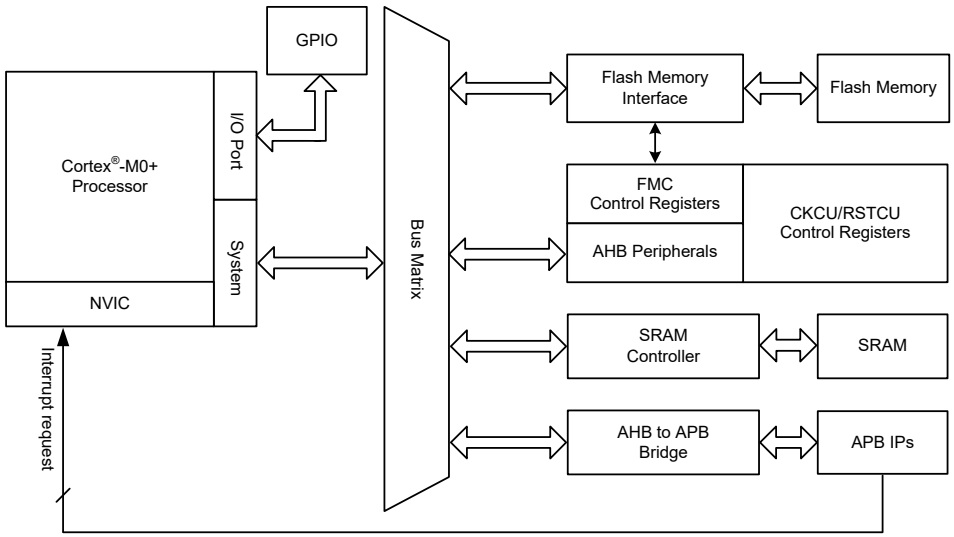


Figure 3. Bus Architecture

Memory Organization

The Arm® Cortex®-M0+ processor access and debug access share the single external interface to external AHB peripherals. The processor access takes priority over debug access. The maximum address range of the Cortex®-M0+ is 4 GB since it has 32-bit bus address width. Additionally, a pre-defined memory map is provided by the Cortex®-M0+ processor to reduce the software complexity of repeated implementation of different device vendors. However, some regions are used by the Arm® Cortex®-M0+ system peripherals. Refer to the Arm® Cortex®-M0+ Technical Reference Manual for more information. The following figure shows the memory map of the HT32F50020/HT32F50030 series of devices, including Code, SRAM, peripheral, and other pre-defined regions.

Memory Map

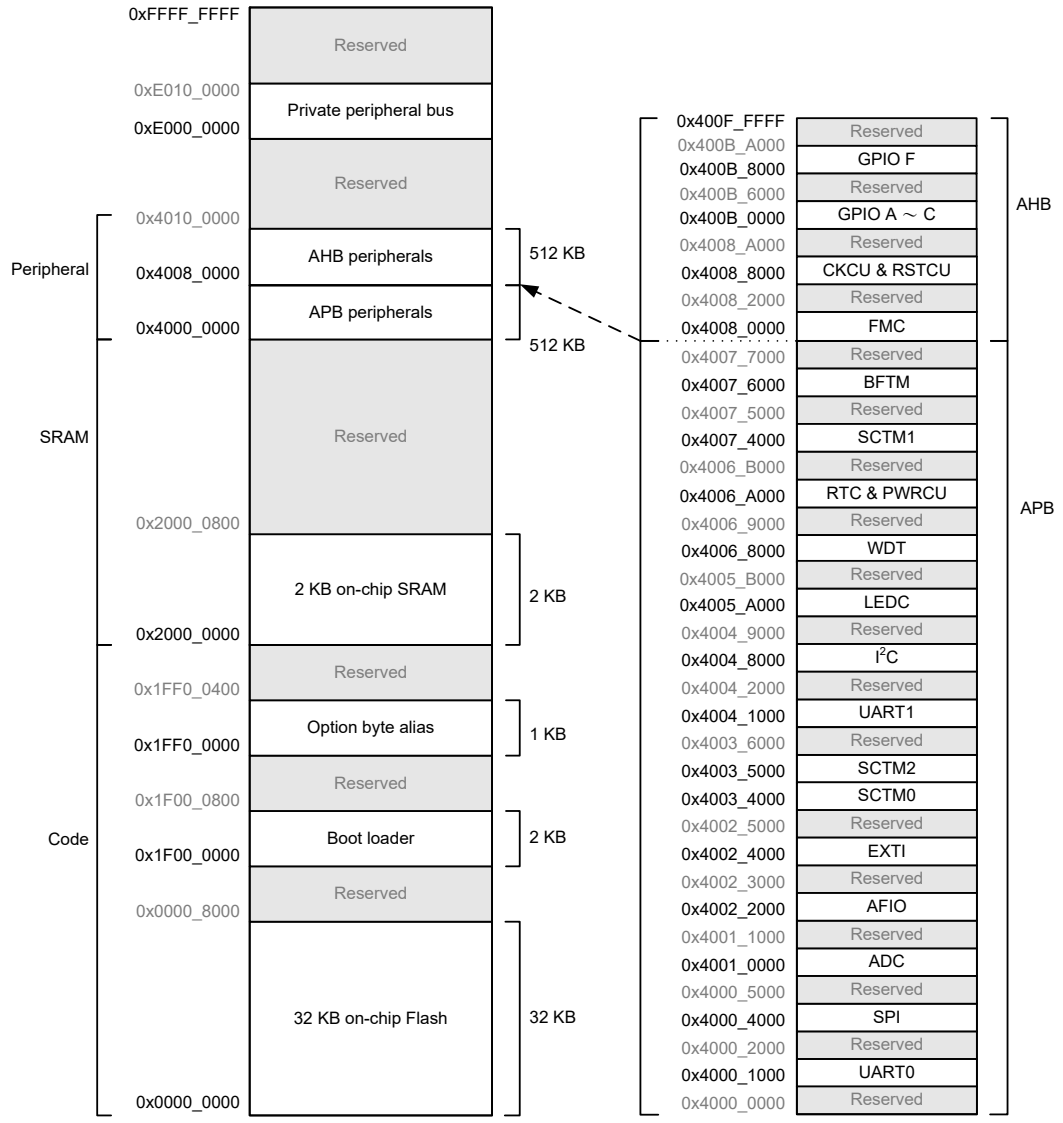


Figure 4. Memory Map

Table 3. Register Map

Start Address	End Address	Peripheral	Bus
0x4000_0000	0x4000_0FFF	Reserved	APB
0x4000_1000	0x4000_1FFF	UART0	
0x4000_2000	0x4000_3FFF	Reserved	
0x4000_4000	0x4000_4FFF	SPI	
0x4000_5000	0x4000_FFFF	Reserved	
0x4001_0000	0x4001_0FFF	ADC	
0x4001_1000	0x4002_1FFF	Reserved	
0x4002_2000	0x4002_2FFF	AFIO	
0x4002_3000	0x4002_3FFF	Reserved	
0x4002_4000	0x4002_4FFF	EXTI	
0x4002_5000	0x4003_3FFF	Reserved	
0x4003_4000	0x4003_4FFF	SCTM0	
0x4003_5000	0x4003_5FFF	SCTM2	
0x4003_6000	0x4004_0FFF	Reserved	
0x4004_1000	0x4004_1FFF	UART1	
0x4004_2000	0x4004_7FFF	Reserved	
0x4004_8000	0x4004_8FFF	I^2C	
0x4004_9000	0x4005_9FFF	Reserved	
0x4005_A000	0x4005_AFFF	LEDC	
0x4005_B000	0x4006_7FFF	Reserved	
0x4006_8000	0x4006_8FFF	WDT	
0x4006_9000	0x4006_9FFF	Reserved	
0x4006_A000	0x4006_AFFF	RTC & PWRCU	
0x4006_B000	0x4007_3FFF	Reserved	
0x4007_4000	0x4007_4FFF	SCTM1	
0x4007_5000	0x4007_5FFF	Reserved	
0x4007_6000	0x4007_6FFF	BFTM	
0x4007_7000	0x4007_FFFF	Reserved	
0x4008_0000	0x4008_1FFF	FMC	AHB
0x4008_2000	0x4008_7FFF	Reserved	
0x4008_8000	0x4008_9FFF	CKCU & RSTCU	
0x4008_A000	0x400A_FFFF	Reserved	
0x400B_0000	0x400B_1FFF	GPIO A	
0x400B_2000	0x400B_3FFF	GPIO B	
0x400B_4000	0x400B_5FFF	GPIO C	
0x400B_6000	0x400B_7FFF	Reserved	
0x400B_8000	0x400B_9FFF	GPIO F	
0x400B_A000	0x400F_FFFF	Reserved	

Embedded Flash Memory

The HT32F50020/HT32F50030 device series provide an up to 32 KB on-chip Flash memory which is located at address 0x0000_0000. It supports byte, half-word and word access operations. Note that the Flash memory only supports read operations for the bus access. Any write operations to the Flash memory will cause a bus fault exception. The Flash memory has up to 32 pages. Each page has a memory capacity of 1 KB and can be erased independently. A 32-bit programming interface provides the capability of changing bits from 1 to 0. A data storage or firmware upgrade can be implemented using several methods such as In System Programming (ISP), In Application Programming (IAP) or In Circuit Programming (ICP). For more information, refer to the Flash Memory Controller section.

Embedded SRAM Memory

The HT32F50020/HT32F50030 device series contain an up to 2 KB on-chip SRAM which is located at address 0x2000_0000. It supports byte, half-word and word access operations.

AHB Peripherals

The address of the AHB peripherals ranges from 0x4008_0000 to 0x400F_FFFF. Some peripherals such as Clock Control Unit, Reset Control Unit and Flash Memory Controller are connected to the AHB bus directly. The AHB peripherals clocks are always enabled after a system reset. Access to registers for these peripherals can be achieved directly via the AHB bus. Note that all peripheral registers in the AHB bus support only word access.

APB Peripherals

The address of APB peripherals ranges from 0x4000_0000 to 0x4007_FFFF. An APB to AHB Bridge provides access capability between the CPU and the APB peripherals. Additionally, the APB peripheral clocks are disabled after a system reset. Software must enable the peripheral clocks by setting up the APBCCRn register in the Clock Control Unit before accessing the corresponding peripheral register. Note that the APB to AHB Bridge will duplicate the half-word or byte data to word width when a half-word or byte access is performed on the APB peripheral registers. In other words, the access result of a half-word or byte access on the APB peripheral register will vary depending on the data bit width of the access operation on the peripheral registers.

4 Flash Memory Controller (FMC)

Introduction

The Flash Memory Controller, FMC, provides all the necessary functions for the embedded on-chip Flash memory. The figure below shows the block diagram of FMC which includes programming interface, control register and access interface. Since the access speed of the Flash memory is slower than the CPU, a wide access interface is provided to the Flash memory in order to reduce the CPU waiting time which will cause CPU instruction execution delay. The Flash memory word programming/page erase functions are also provided for instruction/data storage.

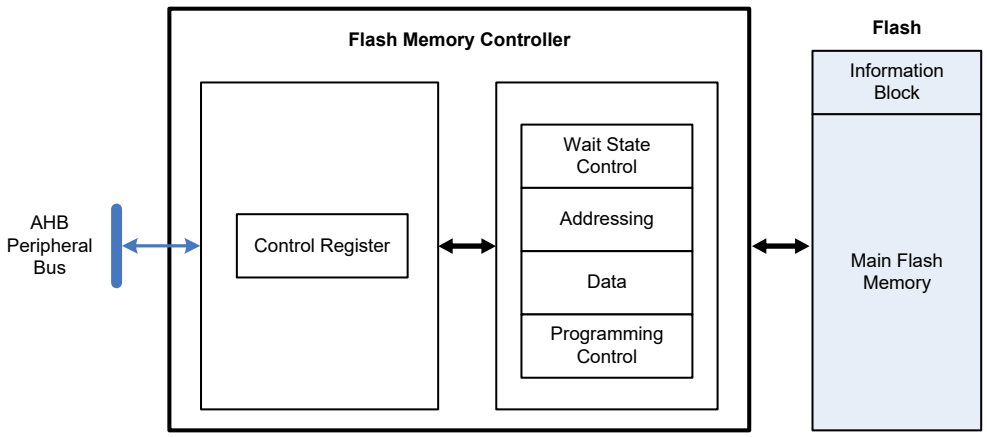


Figure 5. Flash Memory Controller Block Diagram

Features

- Up to 32 KB of on-chip Flash memory for storing instruction/data and option bytes
  - 32 KB (instruction/data + Option Byte) for the HT32F50030
  - 16 KB (instruction/data + Option Byte) for the HT32F50020
- Page size of 1 KB, totally up to 32 pages depending on the main Flash size
- Wide access interface to reduce instruction gaps
- Page erase and mass erase capability
- 32-bit word programming
- Interrupt function to indicate end of Flash memory operations or an error occurrence
- Flash read protection to prevent illegal code/data access
- Page erase/program protection to prevent unexpected operations

Functional Descriptions

Flash Memory Map

The following figure is the Flash memory map of the system. The address ranges from 0x0000_0000 to 0x1FFF_FFFF (0.5 GB). The address from 0x1F00_0000 to 0x1F00_07FF is mapped to the Boot Loader Block with a capacity of 2 KB. Additionally, the region addressed from 0x1FF0_0000 to 0x1FF0_03FF is the alias of the Option Byte block with a capacity of 1 KB, which physically locates at the last page of the main Flash. The memory mapping on system view is shown below.

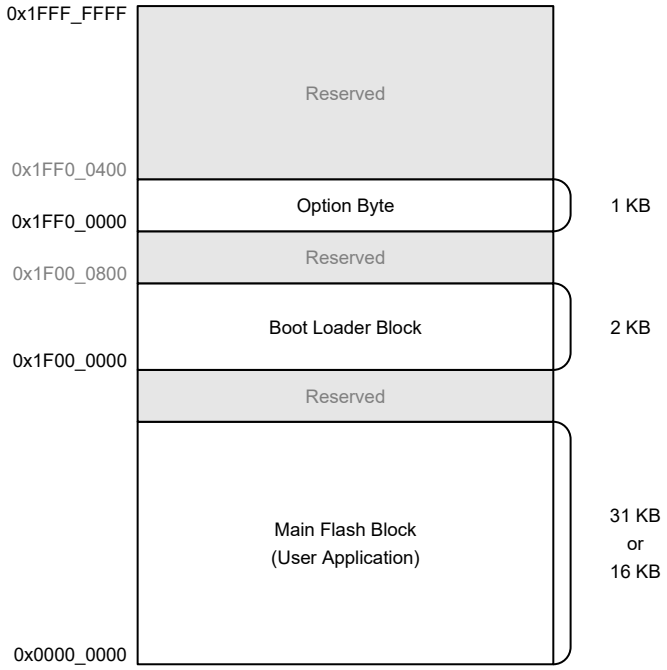


Figure 6. Flash Memory Map

Flash Memory Architecture

The Flash memory consists of up to 32 KB main Flash Block with 1 KB per page and 2 KB Information Block for Boot Loader. The main Flash memory contains a total of 32 pages (or 16 pages for 16 KB device) which can be erased individually. The following table shows the base address, size, and protection setting bit of each page.

Table 4. Flash Memory and Option Byte

Block	Name	Address	Page Protection Bit	Size
Main Flash Block	Page 0	0x0000_0000 ~ 0x0000_03FF	OB_PP [0]	1 KB
	Page 1	0x0000_0400 ~ 0x0000_07FF	OB_PP [1]	1 KB
	Page 2	0x0000_0800 ~ 0x0000_0BFF	OB_PP [2]	1 KB
	Page 3	0x0000_0C00 ~ 0x0000_0FFF	OB_PP [3]	1 KB
	:	:	:	:
	Page 28	0x0000_7000 ~ 0x0000_73FF	OB_PP [28]	1 KB
	Page 29	0x0000_7400 ~ 0x0000_77FF	OB_PP [29]	1 KB
	Page 30	0x0000_7800 ~ 0x0000_7BFF	OB_PP [30]	1 KB
	Page 31 (Option Byte)	Physical address: 0x0000_7C00 ~ 0x0000_7FFF Alias address: 0x1FF0_0000 ~ 0x1FF0_03FF	OB_CP [1]	1 KB
Information Block	Boot Loader	0x1F00_0000 ~ 0x1F00_07FF	NA	2 KB

Notes: 1. The Information Block stores the boot loader and this block cannot be programmed or erased by users.
2. The Option Byte is always located at the last page of the Main Flash Block.

Booting Configuration

The system provides two kinds of booting modes which can be selected using the BOOT pin. The BOOT pin status is sampled during the power-on reset or system reset. Once the logic value is decided, the first 4 words of vector will be remapped to the corresponding source according to the booting mode. The booting modes are shown in the following table.

Table 5. Booting Modes

Booting Mode Selection Pin	Mode	Descriptions
BOOT		
0	Boot Loader	The vector source is Boot Loader
1	Main Flash	The vector source is main Flash

The Flash Vector Mapping Control Register, VMCR, is provided to change the vector remapping setting temporarily after the chip reset. The initial reset value of the VMCR register is determined by the BOOT pin status which will be sampled during the reset duration.

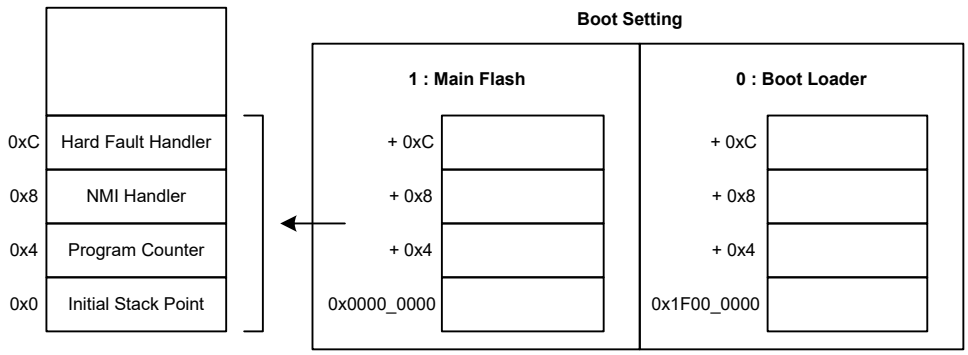


Figure 7. Vector Remapping

Page Erase

The FMC provides a page erase function which is used to reset partial content of the Flash memory. Any page can be erased independently without affecting others. The following steps show the page erase operation register access sequence.

1. Check the OPCR register to confirm that no Flash memory operation is in progress (OPM [3:0] is equal to 0xE or 0x6). Otherwise, wait until the previous operation has been finished.
2. Write the page address to the TADR register.
3. Write the page erase command to the OCMR register (Set CMD [3:0] = 0x8).
4. Commit the page erase command to the FMC by setting the OPCR register (Set OPM [3:0] = 0xA).
5. Wait until all the operations have been completed by checking the value of the OPCR register (OPM [3:0] is equal to 0xE).
6. Read and verify the page if required.

Note that a correct address of the target page must be confirmed. The software may run out of control if the target erase page is under the code fetching or data accessing status. The FMC will not provide any notification when this happens. Additionally, the page erase operation will be ignored on the protected pages. When this occurs, the OREF bit will be set by the FMC and then a Flash Operation Error interrupt will be generated if the OREIEN bit in the OIER register is set. The software can check the PPEF bit in the OISR register to detect this condition in the interrupt handler. The following figure shows the page erase operation flow.

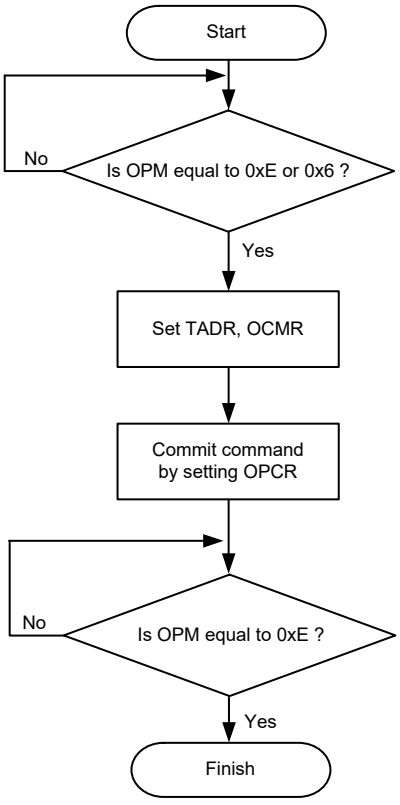


Figure 8. Page Erase Operation Flowchart

Mass Erase

The FMC provides a mass erase function which is used to initialize all the main Flash memory contents to a high state. The following steps show the mass erase operation register access sequence.

1. Check the OPCR register to confirm that no Flash memory operation is in progress (OPM [3:0] is equal to 0xE or 0x6). Otherwise, wait until the previous operation has been finished.
2. Write the mass erase command to the OCMR register (Set CMD [3:0] = 0xA).
3. Commit the mass erase command to the FMC by setting the OPCR register (Set OPM [3:0] = 0xA).
4. Wait until all the operations have been finished by checking the value of the OPCR register (OPM [3:0] is equal to 0xE).
5. Read and verify the Flash memory if required.

Since all Flash data will be reset as 0xFFFF_FFFF, the mass erase operation can be implemented by the program that runs on the SRAM or by the debugging tool that accesses the FMC registers directly. The application program that is executed on the Flash memory will not trigger a mass erase operation. The following figure shows the mass erase operation flow.

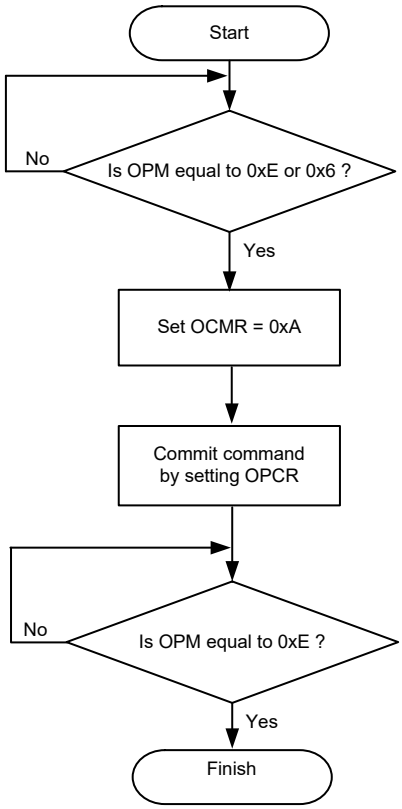


Figure 9. Mass Erase Operation Flowchart

Word Programming

The FMC provides a 32-bit word programming function which is used to modify the Flash memory contents. The following steps show the word programming operation register access sequence.

1. Check the OPCR register to confirm that no Flash memory operation is in progress (OPM [3:0] is equal to 0xE or 0x6). Otherwise, wait until the previous operation has been finished.
2. Write the word address to the TADR register. Write the word data to the WRDR register.
3. Write the word programming command to the OCMR register (Set CMD [3:0] = 0x4).
4. Commit the word programming command to the FMC by setting the OPCR register (Set OPM [3:0] = 0xA).
5. Wait until all the operations have been finished by checking the value of the OPCR register (OPM [3:0] is equal to 0xE).
6. Read and verify the Flash memory if required.

Note that the word programming operation cannot be successively applied to the same address twice. Successive word programming operation to the same address must be separated by a page erase operation. Additionally, the word programming operation will be ignored on the protected pages. When this occurs, the OREF bit will be set by the FMC and then a Flash Operation Error interrupt will be generated if the OREIEN bit in the OIER register is set. The software can check the PPEF bit in the OISR register to detect this condition in the interrupt handler. The following figure shows the word programming operation flow.

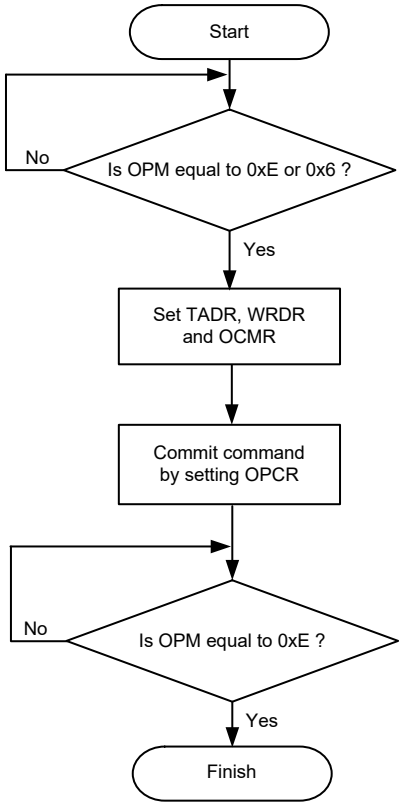


Figure 10. Word Programming Operation Flowchart

Option Byte Description

The Option Byte area can be treated as an independent Flash memory of which the base address is 0x1FF0_0000. The following table shows the functional description and the Option Byte memory map.

Table 6. Option Byte Memory Map

Option Byte	Offset	Description	Reset Value
Option Byte Base Address = 0x1FF0_0000			
OB_PP	0x000 0x004 0x008 0x00C	Flash Page Erase/Program Protection OB_PP [n] (n = 0 ~ X) 0: Flash Page n Erase / Program Protection is enabled 1: Flash Page n Erase / Program Protection is disabled OB_PP [n] (n = (X + 1) ~ 127) Reserved For the HT32F50020 the variable "X" is equal to 15. For the HT32F50030 the variable "X" is equal to 30.	0xFFFF_FFFF 0xFFFF_FFFF 0xFFFF_FFFF 0xFFFF_FFFF
OB_CP	0x010	Flash Security Protection OB_CP [0] 0: Flash Security protection is enabled 1: Flash Security protection is disabled Option Byte Protection OB_CP [1] 0: Option Byte protection is enabled 1: Option Byte protection is disabled OB_CP [31:2]: Reserved	0xFFFF_FFFF
OB_CK	0x020	Flash Option Byte Checksum OB_CK [31:0] OB_CK should be set as the sum of 5 words of Option Byte content, of which the offset address ranges from 0x000 to 0x010 (0x000 + 0x004 + 0x008 + 0x00C + 0x010), when the OB_PP or OB_CP register content is not equal to 0xFFFF_FFFF. Otherwise, both page erase/program protection and security protection will be enabled.	0xFFFF_FFFF
OB_WDT	0x02C	Flash Option Watchdog Timer Enable OB_WDT [15:0]: 0x7A92 If the OB_WDT [15:0] is set to 0x7A92, the WDT will be enabled immediately when the MCU power on reset or system reset occurs. The WDT can be disabled by software. OB_WDT [31:16]: Reserved	0xFFFF_FFFF
OB_TOOL	0x030 ~ 0x04C	Reserved for Flash writer tool and boot loader.	0xFFFF_FFFF

Page Erase/Program Protection

The FMC provides a page erase/program protection function to prevent unexpected operations on the protected Flash memory area. The page erase (CMD [3:0] = 0x8 in the OCMR register) or word programming (CMD [3:0] = 0x4) command will not be accepted by the FMC on the protected pages. When the page erase or word programming command aimed at the protected pages is sent to the FMC, the PPEF bit in the OISR register will then be set by the FMC and the Flash Operation Error interrupt will be triggered to inform the CPU if the OREIEN bit in the OIER register is set. The page protection function can be individually enabled for each page by configuring the OB_PP registers in the Option Byte area. The following table shows the access permission of the main Flash page when the page protection is enabled.

Table 7. Access Permission of Protected Main Flash Page

Mode / Operation	ISP/IAP	ICP/Debug Mode
Read	O	O
Programming	X	X
Page Erase	X	X
Mass Erase	O	O

Notes: 1. Each write protection bit setting is for one specific page. The above access permission only affects the pages of which the protection function has been enabled. Other pages are not affected.
2. The main Flash page protection is configured by OB_PP [30:0]. Option Byte is physically located at the last page of the main Flash. The Option Byte page protection is configured by the OB_CP [1] bit.
3. The page erase operation on the Option Byte area can disable the page protection of the main Flash.
4. The page protection of the Option Byte can only be disabled by a mass erase operation.

The following steps show the register access sequence for the page erase/program protection procedure.

1. Check the OPCR register to confirm that no Flash memory operation is in progress (OPM [3:0] is equal to 0xE or 0x6). Otherwise, wait until the previous operation has been finished.
2. Write the OB_PP address to the TADR register (Set TADR = 0x1FF0_0000).
3. Write the desired data, which indicates the protection function of the corresponding page is to be enabled or disabled, into the WRDR register (0: Enabled, 1: Disabled).
4. Write the word programming command to the OCMR register (Set CMD [3:0] = 0x4).
5. Commit the word programming command to the FMC by setting the OPCR register (Set OPM [3:0] = 0xA).
6. Wait until all the operations have been finished by checking the value of the OPCR register (OPM [3:0] is equal to 0xE).
7. Read and verify the Option Byte if required.
8. The OB_CK field in the Option Byte area must be updated according to the Option Byte checksum rule.
9. Apply a system reset to activate the new OB_PP setting.

Security Protection

The FMC provides a security protection function to prevent illegal code/data access to the Flash memory. This function is useful for protecting the software/firmware from illegal users. The function is activated by setting OB_CP [0] in the Option Byte. Once the function has been enabled, all the main Flash data access through ICP/Debug mode, programming and page erase operation will not be allowed except the user's application. However the mass erase operation will still be accepted by the FMC in order to disable this security protection function. The following table shows the access permission of the Flash memory when the security protection is enabled.

Table 8. Access Permission When Security Protection is Enabled

Operation \ Mode	User Application (1)	ICP/Debug Mode
Read	O	X (read as 0)
Programming	O (1)	X
Page Erase	O (1)	X
Mass Erase	O	O

Notes: 1. User application means the software that is executed or booted from the main Flash memory with the SW debugger being disconnected. However, the Option Byte area and page 0 are still under protection, where the Programming/Page Erase operations are not accepted.
2. The Mass Erase operation can erase the Option Byte area and disable the security protection.

The following steps show the register access sequence for the security protection procedure.

1. Check the OPCR register to confirm that no Flash memory operation is in progress (OPM [3:0] is equal to 0xE or 0x6). Otherwise, wait until the previous operation has been finished.
2. Write the OB_CP address to the TADR register (Set TADR = 0x1FF0_0010).
3. Write data to the WRDR register to set OB_CP [0] to 0.
4. Write the word programming command to the OCMR register (Set CMD [3:0] = 0x4).
5. Commit the word programming command to the FMC by setting the OPCR register (Set OPM = 0xA).
6. Wait until all the operations have been finished by checking the value of the OPCR register (OPM [3:0] is equal to 0xE).
7. Read and verify the Option Byte if required.
8. The OB_CK field in the Option Byte area must be updated according to the Option Byte checksum rule.
9. Apply a system reset to active the new OB_CP setting.

Register Map

The following table shows the FMC registers and reset values.

Table 9. FMC Register Map

Register	Offset	Description	Reset Value
TADR	0x000	Flash Target Address Register	0x0000_0000
WRDR	0x004	Flash Write Data Register	0x0000_0000
OCMR	0x00C	Flash Operation Command Register	0x0000_0000
OPCR	0x010	Flash Operation Control Register	0x0000_000C
OIER	0x014	Flash Operation Interrupt Enable Register	0x0000_0000
OISR	0x018	Flash Operation Interrupt and Status Register	0x0001_0000
PPSR	0x020 0x024 0x028 0x02C	Flash Page Erase/Program Protection Status Register	0xXXXX_XXXX 0xXXXX_XXXX 0xXXXX_XXXX 0xXXXX_XXXX
CPSR	0x030	Flash Security Protection Status Register	0x0000_000X
VMCR	0x100	Flash Vector Mapping Control Register	0x0000_000X
MDID	0x180	Flash Manufacturer and Device ID Register	0x0376_XXXX
PNSR	0x184	Flash Page Number Status Register	0x0000_00XX
PSSR	0x188	Flash Page Size Status Register	0x0000_0400
DIDR	0x18C	Device ID Register	0x000X_XXXX
CIDR0	0x310	Custom ID Register 0	0xXXXX_XXXX
CIDR1	0x314	Custom ID Register 1	0xXXXX_XXXX
CIDR2	0x318	Custom ID Register 2	0xXXXX_XXXX
CIDR3	0x31C	Custom ID Register 3	0xXXXX_XXXX

Note: “X” means various reset values which depend on the Device, Flash value, Option Byte value or power on reset setting.

Register Descriptions

Flash Target Address Register – TADR

This register specifies the target address of the page erase and word programming operations.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	TADB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	TADB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	TADB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	TADB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:0]	TADB	Flash Target Address Bits For programming operations, the TADR register specifies the address where the data is written to. Since the programming length is 32-bit, the TADR register should be set as word-aligned (4 bytes). The TADB [1:0] bits will be ignored during programming operations. For page erase operations, the TADR register contains the page address which is to be erased. Since the page size is 1 KB, the TADB [9:0] bits will be ignored in order to limit the target address as 1 Kbyte-aligned. For 32 KB main Flash addressing, TADB [31:15] should be zero and TADB [31:14] should be zero for 16 KB and so on. The region of which the address ranges from 0x1FF0_0000 to 0x1FF0_03FF is the 1 KB Option Byte. This field for available Flash address must be within the range of 0x0000_0000 to 0x1FFF_FFFF. Otherwise, an Invalid Target Address interrupt will be generated if the corresponding interrupt enable bit is set.

Flash Write Data Register – WRDR

This register specifies the data to be written for the programming operation.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	WRDB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	WRDB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	WRDB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	WRDB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:0]	WRDB	Flash Write Data Bits The data value for the programming operation.

Flash Operation Command Register – OCMR

This register is used to specify the Flash operation commands that include word programming, page erase and mass erase.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved				CMD			
Type/Reset					RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[3:0]	CMD	Flash Operation Command The following table shows the definitions of the operation command field, CMD [3:0], which specifies the Flash memory operation. If an invalid command is set and the IOCMIEN bit is set to 1, an Invalid Operation Command interrupt will be generated.

CMD [3:0]	Description
0x0	Idle (default)
0x4	Word programming
0x8	Page erase
0xA	Mass erase
Others	Reserved

Flash Operation Control Register – OPCR

This register is used for controlling the command commitment and checking the status of the FMC operations.

Offset: 0x010

Reset value: 0x0000_000C

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved			OPM				Reserved
Type/Reset				RW 0	RW 1	RW 1	RW 0	

Bits	Field	Descriptions
[4:1]	OPM	Operation Mode The following table shows the FMC operation modes. Users can commit command which is set by the OCMR register to the FMC according to the address alias setting in the TADR register. The contents of the TADR, WRDR and OCMR registers should be prepared before setting this register. After all the operations have been finished, the OPM field will be set to 0xE by the FMC hardware. The Idle mode can be set when all the operations have been finished for power saving purpose. Note that the operation status should be checked before executing next operation. The contents of the TADR, WRDR, OCMR and OPCR registers should not be changed until the previous operation has been finished.

OPM [3:0]	Description
0x6	Idle (default)
0xA	Commit command to main Flash
0xE	All operation finished on main Flash
Others	Reserved

Flash Operation Interrupt Enable Register – OIER

This register is used to enable or disable the FMC interrupt function. The FMC will generate the interrupt when the corresponding interrupt enable bit is set and the interrupt condition occurs.

Offset: 0x014

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved			OREIEN	IOCMIEN	OBEIEN	ITADIEN	ORFIEN
Type/Reset				RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[4]	OREIEN	Operation Error Interrupt Enable 0: Operation Error Interrupt is disabled 1: Operation Error Interrupt is enabled
[3]	IOCMIEN	Invalid Operation Command Interrupt Enable 0: Invalid Operation Command Interrupt is disabled 1: Invalid Operation Command Interrupt is enabled
[2]	OBEIEN	Option Byte Check Sum Error Interrupt Enable 0: Option Byte Check Sum Error Interrupt is disabled 1: Option Byte Check Sum Error Interrupt is enabled
[1]	ITADIEN	Invalid Target Address Interrupt Enable 0: Invalid Target Address Interrupt is disabled 1: Invalid Target Address Interrupt is enabled
[0]	ORFIEN	Operation Finished Interrupt Enable 0: Operation Finished Interrupt is disabled 1: Operation Finished Interrupt is enabled

Flash Operation Interrupt and Status Register – OISR

This register indicates the FMC interrupt status which is used to check if a Flash operation has been finished or if an error has occurred. The status bits, bit [4:0], if set high, are available to trigger the interrupts when the corresponding enable bits in the OIER register are set high.

Offset: 0x018

Reset value: 0x0001_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved						PPEF	RORFF
Type/Reset							RO 0	RO 1
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved			OREF	IOCMF	OBEF	ITADF	ORFF
Type/Reset				WC 0	WC 0	WC 0	WC 0	WC 0

Bits	Field	Descriptions
[17]	PPEF	Page Erase/Program Protected Error Flag 0: Page Erase/Program Protected Error does not occur 1: Operation error occurs due to an invalid erase/program operation being applied to a protected page This bit is reset by hardware once a new flash operation command is committed.
[16]	RORFF	Raw Operation Finished Flag 0: The last Flash operation command is not finished 1: The last Flash operation command is finished The RORFF bit is directly connected to the Flash memory for debugging purpose.
[4]	OREF	Operation Error Flag 0: No Flash operation error occurred 1: The last Flash operation is failed This bit will be set high when any Flash operation error occurs such as an invalid command, program error and erase error, etc. The Operation Error interrupt will be generated if the OREIEN bit in the OIER register is set. Reset this bit by writing 1.
[3]	IOCMF	Invalid Operation Command Flag 0: No invalid Flash operation command has been written into the OCMR register 1: An invalid Flash operation command has been written into the OCMR register This bit will be set when an invalid Flash operation command has been written into the OCMR register. Then the Invalid Operation Command interrupt will be generated if the IOCMIEN bit in the OIER register is set. Reset this bit by writing 1.

Bits	Field	Descriptions
[2]	OBEF	Option Byte Checksum Error Flag 0: Option Byte checksum is correct 1: Option Byte checksum is incorrect This bit will be set high when the Option Byte checksum is incorrect. The Option Byte Checksum Error interrupt will be generated if the OBEIEN bit in the OIER register is set. This bit is cleared to zero by software writing “1” into it. However, the Option Byte Checksum Error Flag cannot be cleared by software until the interrupt condition is released, which means that the Option Byte checksum value has be correctly modified or the corresponding interrupt control is disabled. Otherwise, the interrupt will be continually generated.
[1]	ITADF	Invalid Target Address Flag 0: The target address is valid 1: The target address is invalid The data in the TADR field must be within the range from 0x0000_0000 to 0x1FFF_FFFF. Otherwise, this bit will be set high and an Invalid Target Address interrupt will be generated if the ITADIEN bit in the OIER register is set. Reset this bit by writing 1.
[0]	ORFF	Operation Finished Flag 0: Last Flash operation has not finished 1: Last Flash operation has finished This bit will be set when the last Flash operation has finished. Then the Operation Finished interrupt will be generated if the ORFIEN bit in the OIER register is set. Reset this bit by writing 1.

Flash Page Erase/Program Protection Status Register – PPSR

This register indicates the page protection status of the Flash page erase/program protection functions.

Offset: 0x020 (0) ~ 0x02C (3)

Reset value: 0xXXXX_XXXX

	31	30	29	28	27	26	25	24
	PPSBn							
Type/Reset	RO X	RO X	RO X	RO X	RO X	RO X	RO X	RO X
	23	22	21	20	19	18	17	16
	PPSBn							
Type/Reset	RO X	RO X	RO X	RO X	RO X	RO X	RO X	RO X
	15	14	13	12	11	10	9	8
	PPSBn							
Type/Reset	RO X	RO X	RO X	RO X	RO X	RO X	RO X	RO X
	7	6	5	4	3	2	1	0
	PPSBn							
Type/Reset	RO X	RO X	RO X	RO X	RO X	RO X	RO X	RO X

Bits	Field	Descriptions
[127:0]	PPSBn	Page Erase/Program Protection Status Bits (n = 0 ~ 127) PPSB[n] = OB_PP[n] 0: The corresponding page is protected 1: The corresponding page is not protected The content of this register is not dynamically updated and will only be reloaded from the Option Byte when any kind of reset occurs. The erase or program function of the specific pages is not allowed when the corresponding bits of the PPSR registers are reset. The reset value of PPSR [127:0] is determined by the Option Byte OB_PP [127:0] bits. Since the maximum page number of the main Flash is various and dependent on the chip specification. Therefore, the every page erase/program protection status bit may protect one or two pages, depending on the chip specification. Other bits of the OB_PP and PPSR registers are reserved for future use.

Flash Security Protection Status Register – CPSR

This register indicates the Flash memory security protection status. The content of this register is not dynamically updated and will only be reloaded by the Option Byte loader which is activated when any kind of reset occurs.

Offset: 0x030

Reset value: 0x0000_000X

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						OBPSB	CPSB
Type/Reset							RO X	RO X

Bits	Field	Descriptions
[1]	OBPSB	Option Byte Page Erase/Program Protection Status Bit 0: The Option Byte page is protected 1: The Option Byte page is not protected The reset value of the OBPSB bit is determined by the Option Byte OB_CP [1] bit.
[0]	CPSB	Flash Security Protection Status Bit 0: Flash Security protection is enabled 1: Flash Security protection is not enabled The reset value of the CPSB bit is determined by the Option Byte OB_CP [0] bit.

Flash Vector Mapping Control Register – VMCR

This register is used to control the vector mapping. The reset value of the VMCR register is determined by the external booting pin, BOOT, during the power-on reset period.

Offset: 0x100

Reset value: 0x0000_000X

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						VMCB	Reserved
Type/Reset							RW	X

Bits	Field	Descriptions
[1]	VMCB	Vector Mapping Control Bit The VMCB bit is used to control the mapping source of the first 4 words of vector addressed from 0x0 to 0xC. The following table shows the vector mapping setting.

BOOT	VMCB	Descriptions
Low	0	Boot Loader mode The vector mapping source is the boot loader area.
High	1	Main Flash mode The vector mapping source is the main Flash area.

The reset value of the VMCB bit is determined by the pin status of the external BOOT pin during power-on reset and system reset. The vector mapping setting can be changed temporarily by configuring the VMCB bit when the application program is executed.

Flash Manufacturer and Device ID Register – MDID

This register specifies the manufacture ID and device part number information which can be used as the product identity.

Offset: 0x180

Reset value: 0x0376_XXXX

	31		30		29		28		27		26		25		24	
	MFID															
Type/Reset	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	1	RO	1
	23		**22**		**21**		**20**		**19**		**18**		**17**		**16**	
	MFID															
Type/Reset	RO	0	RO	1	RO	1	RO	1	RO	0	RO	1	RO	1	RO	0
	15		**14**		**13**		**12**		**11**		**10**		**9**		**8**	
	ChipID															
Type/Reset	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X
	7		**6**		**5**		**4**		**3**		**2**		**1**		**0**	
	ChipID															
Type/Reset	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X

Bits	Field	Descriptions
[31:16]	MFID	Manufacturer ID Read as 0x0376.
[15:0]	ChipID	Chip ID The last 4 digital codes of the MCU device part number.

Flash Page Number Status Register – PNSR

This register indicates the Flash memory page number.

Offset: 0x184

Reset value: 0x0000_00XX

	31	30	29	28	27	26	25	24
	PNSB							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PNSB							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	PNSB							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PNSB							
Type/Reset	RO X	RO X	RO X	RO X	RO X	RO X	RO X	RO X

Bits	Field	Descriptions
[31:0]	PNSB	Flash Page Number Status Bits 0x0000_0010: Totally 16 pages for the on-chip Flash memory device 0x0000_0020: Totally 32 pages for the on-chip Flash memory device 0x0000_0040: Totally 64 pages for the on-chip Flash memory device 0x0000_0080: Totally 128 pages for the on-chip Flash memory device 0x0000_00FF: Totally 255 pages for the on-chip Flash memory device

Flash Page Size Status Register – PSSR

This register indicates the page size in bytes.

Offset: 0x188

Reset value: 0x0000_0400

	31	30	29	28	27	26	25	24
	PSSB							
Type/Reset	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PSSB							
Type/Reset	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	PSSB							
Type/Reset	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	1	0	0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PSSB							
Type/Reset	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0

Bits	Field	Descriptions
[31:0]	PSSB	Flash Page Size Status Bits 0x200: The page size is 512 Bytes per page 0x400: The page size is 1 K Bytes per page 0x800: The page size is 2 K Bytes per page

Device ID Register – DIDR

This register specifies the device part number information which can be used as the product identity.

Offset: 0x18C

Reset value: 0x000X_XXXX

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	Reserved				ChipID			
Type/Reset					RO	RO	RO	RO
Reset					X	X	X	X
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	ChipID							
Type/Reset	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	ChipID							
Type/Reset	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X

Bits	Field	Descriptions
[19:0]	ChipID	Chip ID The complete 5 digital codes of the MCU device part number.

Custom ID Register n – CIDRn (n = 0 ~ 3)

This register specifies the custom ID information which can be used as the custom identity.

Offset: 0x310 (0) ~ 0x31C (3)

Reset value: 0xXXXX_XXXX – Various depending on Flash Manufacture Privilege Information Block

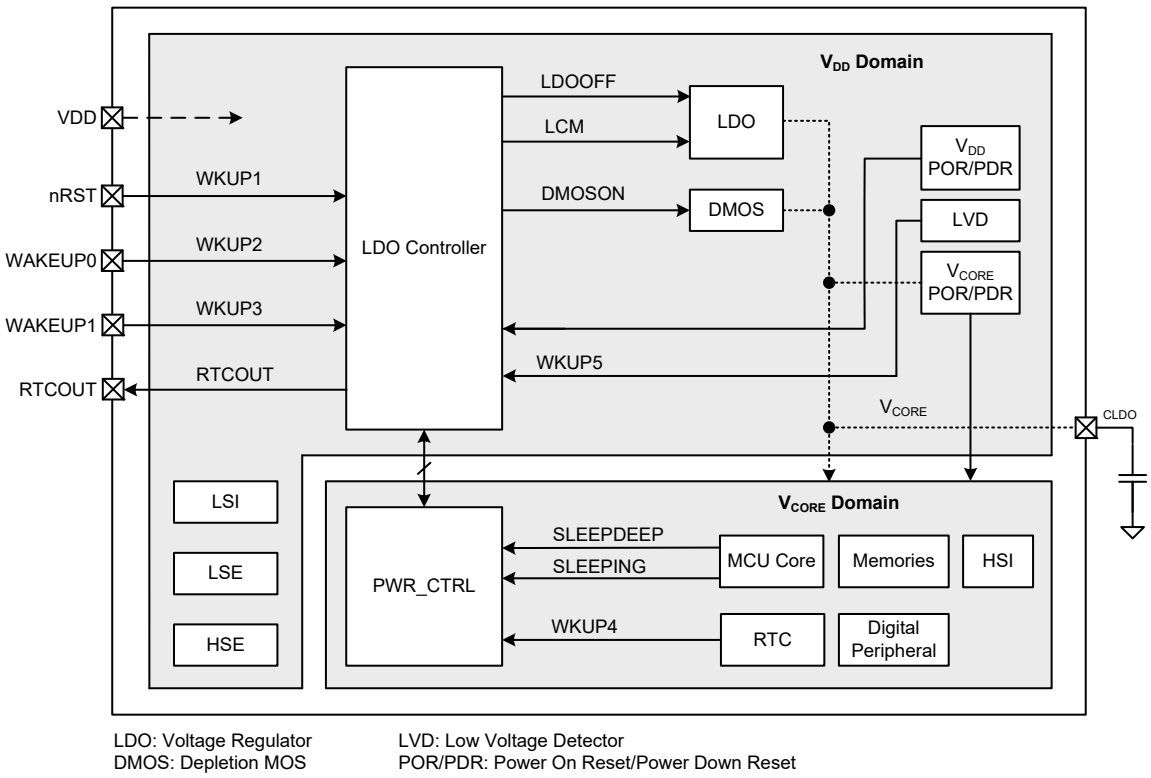
	31		30		29		28		27		26		25		24	
	CID															
Type/Reset	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X
	23		22		21		20		19		18		17		16	
	CID															
Type/Reset	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X
	15		14		13		12		11		10		9		8	
	CID															
Type/Reset	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X
	7		6		5		4		3		2		1		0	
	CID															
Type/Reset	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X

Bits	Field	Descriptions
[31:0]	CIDn	Custom ID Read as the CIDn[31:0] (n = 0 ~ 3) field in the Custom ID registers in Flash Manufacture Privilege Block.

5 Power Control Unit (PWRCU)

Introduction

The power consumption can be regarded as one of the most important issues for many embedded system applications. Accordingly the Power Control Unit, PWRCU, provides many types of power saving modes such as Sleep, Deep-Sleep1 and Deep-Sleep2 modes. These modes reduce the power consumption and allow the application to achieve the best trade-off between the conflicting demands of CPU operating time, speed and power consumption. The shadow in the following figure indicates the power supply source of two digital power domains.



LDO: Voltage Regulator
DMOS: Depletion MOS
LVD: Low Voltage Detector
POR/PDR: Power On Reset/Power Down Reset

Figure 11. PWRCU Block Diagram

Features

- Two power domains: V_{DD} and V_{CORE} power domains
- Three power saving modes: Sleep, Deep-Sleep1 and Deep-Sleep2 modes
- Internal Voltage regulator supplies V_{CORE} voltage source
- Additional Depletion MOS supplies V_{CORE} voltage source with low leakage and low operating current
- A power reset is generated when one of the following events occurs:
  - Power-on / Power-down reset (POR / PDR reset)
  - The control bits BODEN = 1, BODRIS = 0 and the supply power $V_{DD} \leq V_{BOD}$
- The Brown-Out Detector can issue a system reset or an interrupt when V_{DD} power source is lower than the Brown-Out Detector voltage V_{BOD}.
- The Low Voltage Detector can issue an interrupt or wakeup event when V_{DD} is lower than a programmable threshold voltage V_{LVD}.

Functional Descriptions

V_{DD} Power Domain

LDO Power Control

The LDO will be automatically switched off when the following condition occurs:

- The Deep-Sleep 2 mode is entered.

The LDO will be automatically switched on by hardware when the supply power $V_{DD} > V_{POR}$ if any of the following conditions occurs:

- Resume operation from the power saving mode – RTC wakeup, LVD wakeup, EXTI wakeup and WAKEUPn pin wakeup
- Detect a falling edge on the external reset pin (nRST)
- The control bit BODEN = 1 and the supply power $V_{DD} > V_{BOD}$

To enter the Deep-Sleep1 mode, the PWRCU will request the LDO to operate in a low current mode, LCM. To enter the Deep-Sleep 2 mode, the PWRCU will turn off the LDO and turn on the DMOS to supply an alternative V_{CORE} power.

Voltage Regulator

The voltage regulator, LDO, Depletion MOS, DMOS, Low voltage Detector, LVD, Low Speed Internal RC oscillator, LSI, Low Speed external Crystal oscillator, LSE and high Speed external Crystal oscillator, HSE are operated under the V_{DD} power domain. The LDO can be configured to operate in either normal mode (LDOOFF = 0, LDOLCM = 0, I_{OUT} = High current mode) or low current mode (LDOOFF = 0, LDOLCM =1, I_{OUT} = Low current mode) to supply the V_{CORE} power. An alternative V_{CORE} power source is the output of the DMOS which has low static and driving current characteristics. It is controlled using the DMOSON bit in the PWRCR register. The DMOS output has weak output current and regulation capability and only operate in the Deep-Sleep 2 mode for data retention purposes in the V_{CORE} power domain.

Power-On Reset (POR) / Power-Down Reset (PDR)

The devices have an integrated POR/PDR circuitry that allows proper operation starting from V_{POR} or down to V_{PDR}. For more details concerning the power-on/power-down reset threshold voltage, refer to the electrical characteristics of the corresponding datasheet.

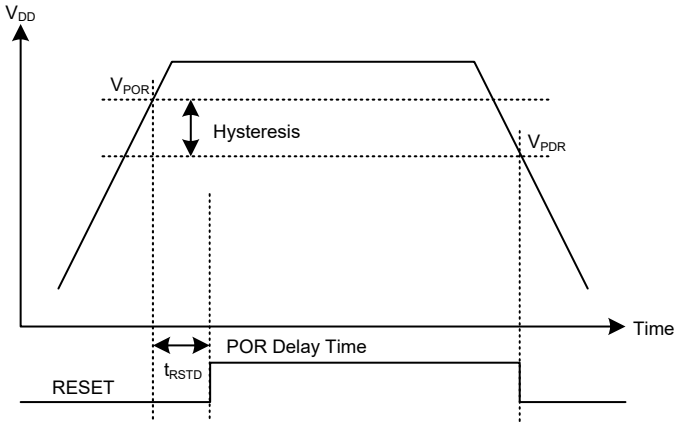


Figure 12. Power-On Reset / Power-Down Reset Waveform

Low Voltage Detector / Brown-Out Detector

The Low Voltage Detector, LVD, can detect whether the supply voltage V_{DD} is lower than a programmable threshold voltage V_{LVD}. It is selected by the LVDS field in the LVDCSR register. When a low voltage on the VDD power pin is detected, the LVDF flag will be active and an interrupt will be generated and sent to the MCU core if the LVDEN and LVDIWEN bits in the LVDCSR register are set. For more details concerning the LVD programmable threshold voltage V_{LVD}, refer to the electrical characteristics of the corresponding datasheet.

The Brown-Out Detector, BOD, is used to detect if the V_{DD} supply voltage is equal to or lower than V_{BOD}. When the BODEN bit in the LVDCSR register is set to 1 and the V_{DD} supply voltage is lower than V_{BOD} then the BODF flag is active. The PWRCU will regard this as a power-down reset situation and then immediately disable the internal LDO regulator when the BODRIS bit is cleared to 0 or issue an interrupt to notify the CPU to execute a power-down procedure when the BODRIS bit is set to 1. For more details concerning the Brown-Out Detector voltage V_{BOD}, refer to the electrical characteristics of the corresponding datasheet.

High Speed External Oscillator

The High Speed External Oscillator, HSE, is located in the V_{DD} power domain. The HSE crystal oscillator can be switched on or off using the HSEEN bit in the Global Clock Control Register, GCCR. The HSE clock can then be used directly as the system clock source.

LSE and LSI

The Low Speed Internal RC oscillator, LSI, or the Low Speed External Crystal oscillator, LSE, ether are located in the V_{DD} power domain. They can be used directly as the system clock source or the RTC clock source. Only LSE can be switched on or off using the LSEEN bit in the RTC Circuitry Control Register(RTCCR).

V_{CORE} Power Domain

The main functions that include high speed internal oscillator HSI, RTC, MCU core logic, AHB/APB peripherals and memories and so on are located in this power domain. Once the V_{CORE} is powered up, the POR will generate a reset sequence on the V_{CORE} power domain. Subsequently, to enter the expected power saving mode, the associated control bits including the LDOOFF, DMOSON and LDOLCM bits must be configured. Then, once a WFI or WFE instruction is executed, the device will enter the expected power saving mode which will be discussed in the following section.

High Speed Internal Oscillator

The High Speed Internal Oscillator, HSI, is located in the V_{CORE} power domain. When exiting from the Deep-Sleep mode, the HSI clock can be configured as the system clock for a certain period by setting the PSRCEN bit to 1. This bit is located in the Global Clock Control Register, GCCR, in the Clock Control Unit, CKCU. The system clock will not be switched back to the original clock source used before entering the Deep-Sleep mode until the original clock source.

RTC

The Real-Time Clock Timer clock source can be derived from either the Low Speed Internal RC oscillator, LSI, or the Low Speed External Crystal oscillator, LSE. Before entering the power saving mode by executing WFI/WFE instruction, the MCU needs to setup the compare register with an expected wakeup time and enable the wakeup function to achieve the RTC timer wakeup event. After entering the power saving mode for a certain amount of time, the Compare Match flag, CMFLAG, will be asserted to wake up the device when the compare match event occurs. The details of the RTC configuration for wakeup timer will be described in the RTC chapter.

Operation Modes

Run Mode

In the Run mode, the system operates with full functions and all power domains are active. There are two ways to reduce the power consumption in this mode. The first is to slow down the system clock by setting the AHBPRE field in the CKCU AHBCFGR register, and the second is to turn off the unused peripherals clock by setting the APBCCR0 and APBCCR1 registers or slow down the peripherals clock by setting the related bit field to meet the application requirement. Reducing the system clock speed before entering the sleep mode will also help to minimize power consumption.

Additionally, there are several power saving modes to provide maximum optimization between device performance and power consumption.

Table 10. Operation Mode Definitions

Mode Name	Hardware Action
Run	After system reset, CPU fetches instructions to execute.
Sleep	CPU clock will be stopped. Peripherals, Flash and SRAM clocks can be stopped by setting.
Deep-Sleep1	Stop all clocks in the V_{CORE} power domain. Disable HSI and HSE. Turn on the LDO low current mode to reduce the V_{CORE} power domain current.
Deep-Sleep2	Turn off the LDO and turn on the DMOS to reduce the V_{CORE} power domain current.

Sleep Mode

By default, only the CPU clock will be stopped in the Sleep mode. Clearing the FMCEN or SRAMEN bit in the CKCU AHBCCR register to zero will have the effect of stopping the Flash clock or SRAM clock after the system enters the Sleep mode. If it is not necessary for the CPU to access the Flash memory and SRAM in the Sleep mode, it is recommended to clear the FMCEN and SRAMEN bits in the AHBCCR register to minimize power consumption. To enter the Sleep mode, it is only necessary to execute a WFI or WFE instruction and let the SLEEPDEEP bit to be 0. The system will exit from the Sleep mode via any interrupt or event trigger. The accompanying table provides more information about the power saving modes.

Table 11. Enter/Exit Power Saving Modes

Mode	Mode Entry				Mode Exit
	CPU Instruction	CPU SLEEPDEEP	LDOOFF	DMOSON	
Sleep	WFI or WFE (Takes effect)	0	X	X	WFI: Any interrupt WFE: Any wakeup event(1) or Any interrupt (NVIC on) or Any interrupt with SEVONPEND = 1 (NVIC off)
Deep-Sleep1		1	0	0	Any EXTI in event mode or RTC wakeup or LVD wakeup(2) or WAKEUPn pin wakeup
Deep-Sleep2		1	X	1	Any EXTI in event mode or RTC wakeup or LVD wakeup(2) or WAKEUPn pin wakeup

Notes: 1. Wakeup event means EXTI line trigger event, RTC event, LVD event or WAKEUPn pin wakeup.
2. If the system allows the LVD activity to wake it up after the system has entered the power saving mode, the LVDEWEN and LVDEN bits in the LVDCSR register must be set to 1 to make sure that the system can be woken up by an LVD event and then the LDO can be turned on when the system is woken up from the Deep-Sleep2 mode.

Deep-Sleep Mode

To enter the Deep-Sleep mode, configure the registers as shown in the preceding table and execute the WFI or WFE instruction. In the Deep-Sleep mode, all clocks including high speed oscillators, known as HSI and HSE, will be stopped. In addition, Deep-Sleep1 turns the LDO into low current mode while Deep-Sleep2 turns off the LDO and uses a DMOS to keep V_{CORE} power. Once the PWRCU receives a wakeup event or an interrupt as shown in the preceding Mode-Exiting table, the LDO will then operate in normal mode and the high speed oscillators will be enabled. Finally, the CPU will return to the Run mode to handle the wakeup interrupt if required. A Low Voltage Detection also can be regarded as a wakeup event if the corresponding wakeup control bit LVDEWEN in the LVDCSR register is enabled. The last wakeup event is a transition from low to high on the external WAKEUPn pin sent to the PWRCU to resume from the Deep-Sleep mode. During the Deep-Sleep mode, retaining the register and memory contents will shorten the wakeup latency.

Table 12. Power Status after System Reset

PORF	PORSTF	Description
1	1	Power-up for the first time after the V_{DD} domain is reset: Power-on reset when V_{DD} is applied for the first time or executing software reset command on the V_{DD} domain.
0	1	Restart from unexpected loss of the V_{CORE} power or other reset (nRST, WDT, …)

WAKEUPn Pin Wakeup

The software can set the WUPnEN bit in register PWRCR to 1 to enable the WAKEUPn pin function before entering the power saving mode, waiting for a wakeup trigger signal occurrence on the WAKEUPn pin to wake up the system from the power saving mode. The external WAKEUPn pin interrupt shares the same exception number with the EXTI event wakeup interrupt. The software can set the EXTI Event Wakeup Interrupt Enable bit (EVWUPIEN) to 1 to assert the WKUP interrupt in the NVIC unit when both the WUPnEN and WUPFn bits are set to 1.

Although the WUPnEN bit is located in the V_{DD} domain, it also can be reset by the nRST reset pin. After the reset there will be a delay before the WUPnEN bit is active. This bit will not be active until the system reset is finished and the V_{DD} domain ISO signal is disabled. This means that the bit cannot be immediately set by software after a system reset is finished and the V_{DD} domain ISO signal is disabled. The delay time requires at least three 32 kHz clock periods after the WUPnEN bit reset has been finished.

Register Map

The following table shows the PWRCU registers and reset values. Note all the registers in this unit are located in the V_{DD} power domain.

Table 13. PWRCU Register Map

Register	Offset	Description	Reset Value
PWRSR	0x100	Power Control Status Register	0x0000_0010
PWRCR	0x104	Power Control Register	0x0000_0000
LVDCSR	0x110	Low Voltage/Brown-Out Detect Control and Status Register	0x0000_0000

Register Descriptions

Power Control Status Register – PWRSR

This register indicates power control status.

Offset: 0x100

Reset value: 0x0000_0010

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved						WUPF1	WUPF0
Type/Reset							RC 0	RC 0
	7	6	5	4	3	2	1	0
	Reserved			PORF	Reserved			
Type/Reset				RC 1				

Bits	Field	Descriptions
[9]	WUPF1	External WAKEUP1 Pin Flag 0: The WAKEUP1 pin is not asserted 1: The WAKEUP1 pin is asserted This bit is set by hardware when the WAKEUP1 pin asserts and is cleared by software read. Software should read this bit to clear it after a system wake up from the power saving mode.
[8]	WUPF0	External WAKEUP0 Pin Flag 0: The WAKEUP0 pin is not asserted 1: The WAKEUP0 pin is asserted This bit is set by hardware when the WAKEUP0 pin asserts and is cleared by software read. Software should read this bit to clear it after a system wake up from the power saving mode.
[4]	PORF	Power On Reset Flag 0: V_{CORE} Power Domain reset does not occur 1: V_{CORE} Power Domain reset occurs This bit is set by hardware when V_{CORE} power on reset occurs, either a hardware power on reset or software reset. The bit is cleared by software read. This bit must be cleared after the system is first powered on, otherwise it will be impossible to detect when a V_{CORE} Power Domain reset has been triggered. When this bit is read as 1, a read software loop must be implemented until the bit returns again to 0.

Power Control Register – PWRCR

This register provides power control bits for the different kinds of power saving modes.

Offset: 0x104

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved				WUP1TYPE		WUP0TYPE	
Type/Reset					RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	DMOSSTS	Reserved				WUP1EN	Reserved	WUP0EN
Type/Reset	RO 0					RW 0		RW 0
	7	6	5	4	3	2	1	0
	DMOSON	Reserved			LDOOFF	LDOLCM	Reserved	PWCURST
Type/Reset	RW 0				RW 0	RW 0		WO 0

Bits	Field	Descriptions
[19:18]	WUP1TYPE	WAKEUP1 Signal Trigger Type

WUP1TYPE [1:0]		WAKEUP1 Signal Trigger Type
0	0	Positive-edge Triggered
0	1	Negative-edge Triggered
1	0	High-level Sensitive
1	1	Low-level Sensitive

[17:16] WUP0TYPE WAKEUP0 Signal Trigger Type

WUP0TYPE [1:0]		WAKEUP0 Signal Trigger Type
0	0	Positive-edge Triggered
0	1	Negative-edge Triggered
1	0	High-level Sensitive
1	1	Low-level Sensitive

[15] DMOSSTS Depletion MOS Status

This bit is set to 1 if the DMOSON bit in this register has been set to 1.

This bit is cleared to 0 if the DMOSON bit has been set to 0 or if a POR/PDR reset occurred.

[10] WUP1EN External WAKEUP1 Pin Enable

0: Disable WAKEUP1 pin function

1: Enable WAKEUP1 pin function

The software can set the WUP1EN bit as 1 to enable the WAKEUP1 pin function before entering the power saving mode. When WUP1EN = 1, a change on the WAKEUP1 pin wakes up the system from the power saving mode. If the WAKEUP1 pin is active high, this bit will set an input pull down mode.

Bits	Field	Descriptions
[8]	WUP0EN	External WAKEUP0 Pin Enable 0: Disable WAKEUP0 pin function 1: Enable WAKEUP0 pin function The software can set the WUP0EN bit as 1 to enable the WAKEUP0 pin function before entering the power saving mode. When WUP0EN = 1, a change on the WAKEUP0 pin wakes up the system from the power saving mode. If the WAKEUP0 pin is active high, this bit will set an input pull down mode.
[7]	DMOSON	DMOS Control 0: DMOS is OFF 1: DMOS is ON A DMOS is implemented to provide an alternative voltage source for the V_{CORE} power domain when the CPU enters the Deep-Sleep mode (SLEEPDEEP = 1). The control bit DMOSON is set by software and cleared by software or V_{DD} power domain reset. If the DMOSON bit is set to 1, the LDO will automatically be turned off when the CPU enters the Deep-Sleep mode.
[3]	LDOOFF	LDO Operating Mode Control 0: The LDO operates in a low current mode when MCU enters the Deep-Sleep mode (SLEEPDEEP = 1). The V_{CORE} power is available 1: The LDO is turned off when the MCU enters the Deep-Sleep mode (SLEEPDEEP = 1). The V_{CORE} power is not available Note: This bit is only available when the DMOSON bit is cleared to 0.
[2]	LDOLCM	LDO Low Current Mode 0: The LDO is operated in normal current mode 1: The LDO is operated in low current mode Note: This bit is only available when CPU is in the run mode. The LDO output current capability will be limited at 10 mA below and lower static current when the LDOLCM bit is set. It is suitable for CPU is operated at lower speed system clock to get a lower current consumption. This bit will be clear to 0 when the LDO is power down or V_{DD} power domain reset.
[0]	PWCURST	Power Control Unit Software Reset 0: No action 1: Power Control Unit Software Reset is activated. When this bit is set, it will reset all the related RTC and PWRCU registers.

Low Voltage / Brown Out Detect Control and Status Register – LVDCSR

This register specifies flags, enable bits and option bits for low voltage detector.

Offset: 0x110

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved	LVDS [2]	LVDEWEN	LVDIWEN	LVDF		LVDS [1:0]	LVDEN
Type/Reset		RW 0	RW 0	RW 0	RO 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved				BODF	Reserved	BODRIS	BODEN
Type/Reset					RO 0		RW 0	RW 0

Bits	Field	Descriptions
[21]	LVDEWEN	LVD Event Wakeup Enable 0: LVD event wakeup is disabled 1: LVD event wakeup is enabled Setting this bit to 1 will enable the LVD event wakeup function to wake up the system when an LVD condition occurs which result in the LVDF bit being asserted. If the system requires to be woken up from the Deep-Sleep mode by an LVD condition, this bit must be set to 1.
[20]	LVDIWEN	LVD Interrupt Wakeup Enable 0: LVD interrupt wakeup is disabled 1: LVD interrupt wakeup is enabled Setting this bit to 1 will enable the LVD interrupt function. When an LVD condition occurs and the LVDIWEN bit is set to 1, an LVD interrupt will be generated and sent to the MCU NVIC unit.
[19]	LVDF	Low Voltage Detect Status Flag 0: V_{DD} is higher than the specific voltage level 1: V_{DD} is equal to or lower than the specific voltage level When the LVD condition occurs, the LVDF flag will be asserted. When the LVDF flag is asserted, an LVD interrupt will be generated for MCU if the LVDIWEN bit is set to 1. However, if the LVDEWEN bit is set to 1 and the LVDIWEN bit is cleared to 0, only an LVD event will be generated rather than an LVD interrupt when the LVDF flag is asserted.
[22], [18:17]	LVDS [2:0]	Low Voltage Detect Level Selection For more details concerning the LVD programmable threshold voltage, refer to the electrical characteristics of the corresponding datasheet.

Bits	Field	Descriptions
[16]	LVDEN	Low Voltage Detect Enable 0: Disable Low Voltage Detect 1: Enable Low Voltage Detect Setting this bit to 1 will generate an LVD event when the V_{DD} power is equal to or lower than the voltage set by LVDS bits. Therefore when the LVD function is enabled before the system is into the Deep-Sleep2 (DMOS is turn on and LDO is power down), the LVDEWEN bit has to be enabled to avoid the LDO does not activate in the meantime when the MCU is woken up by the low voltage detection activity.
[3]	BODF	Brown Out Detect Flag 0: $V_{DD} > V_{BOD}$ 1: $V_{DD} \leq V_{BOD}$
[1]	BODRIS	BOD Reset or Interrupt Selection 0: Reset the whole chip 1: Generate Interrupt
[0]	BODEN	Brown Out Detector Enable 0: Disable Brown Out Detector 1: Enable Brown Out Detector

6 Clock Control Unit (CKCU)

Introduction

The Clock Control unit, CKCU, provides functions of High Speed Internal RC oscillator (HSI), High Speed External crystal oscillator (HSE), Low Speed Internal RC oscillator (LSI), Low Speed External crystal oscillator (LSE), HSE clock monitor, clock prescaler, clock multiplexer and clock gating. The clocks of AHB, APB, and CPU are derived from system clock (CK_SYS) which can come from HSI, HSE, LSI or LSE. Watchdog Timer and Real-Time Clock (RTC) use either LSI or LSE as their clock source. The maximum operating frequency of system clock f_{CK_AHB} can be up to 16 MHz.

A variety of internal clocks can also be wired out through CKOUT for debugging purpose. The clock monitor can be used to get clock failure detection of HSE. Once the clock of HSE does not function such as being broken down or removed, etc., CKCU will force to switch the system clock source to the HSI clock to prevent system halt.

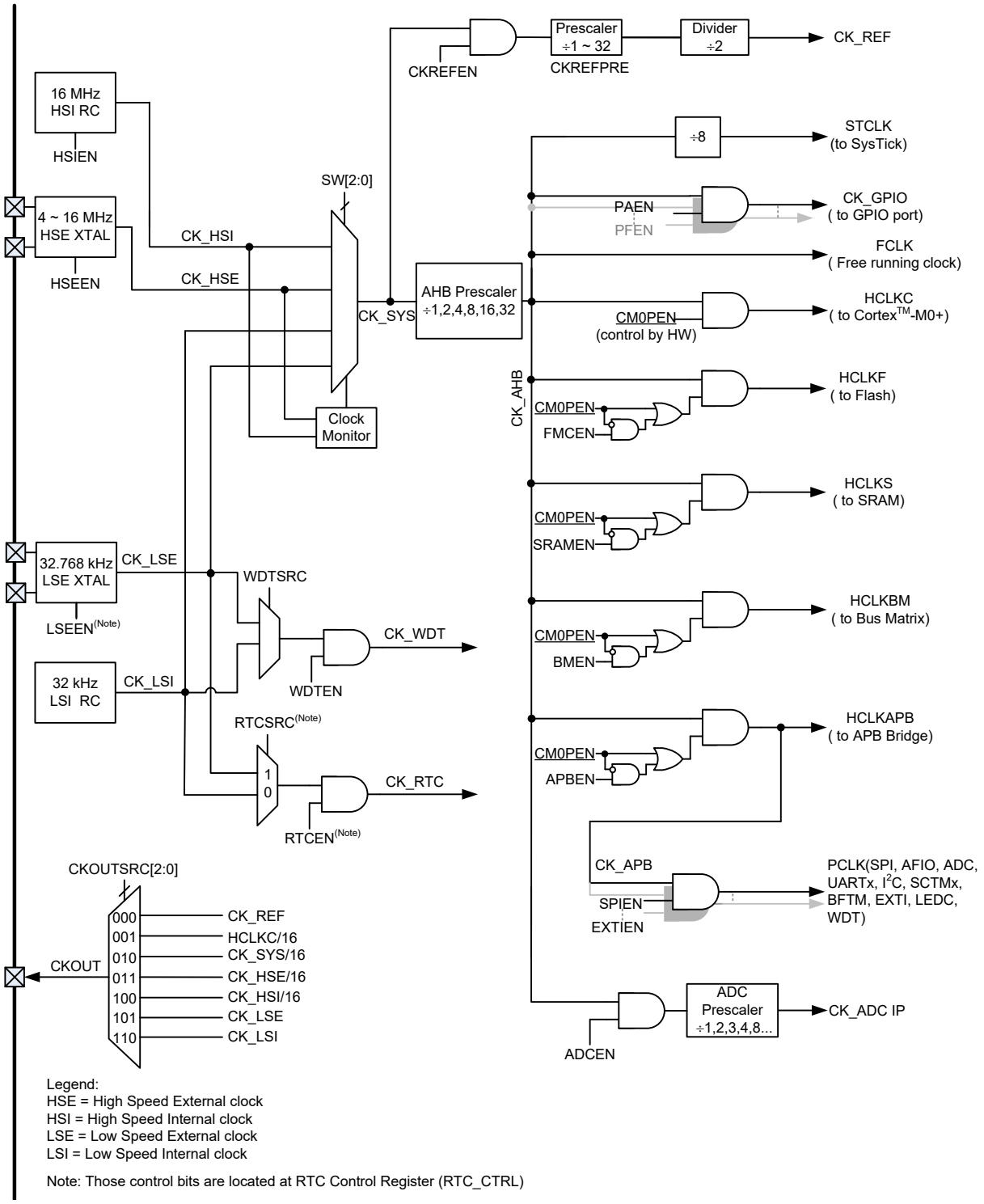


Figure 13. CKCU Block Diagram

Features

- 4 ~ 16 MHz external crystal oscillator (HSE)
- Internal 16 MHz RC oscillator (HSI) with configuration option calibration and custom trimming capability
- 32,768 Hz external crystal oscillator (LSE) for Watchdog Timer, RTC or system clock
- Internal 32 kHz RC oscillator (LSI) for Watchdog Timer, RTC or system clock
- HSE clock monitor

Functional Descriptions

High Speed External Crystal Oscillator – HSE

The high speed external 4 to 16 MHz crystal oscillator (HSE) produces a highly accurate clock source to the system clock. The related hardware configuration is shown in the following figure. The crystal with specific frequency must be placed across the two HSE pins (XTALIN / XTALOUT) and the external components such as resistors and capacitors are necessary to make it oscillate properly.

The following guidelines are provided to improve the stability of the crystal circuit PCB layout.

- The crystal oscillator should be located as close as possible to the MCU so that the trace lengths are kept as short as possible to reduce any parasitic capacitances.
- Shield any lines in the vicinity of the crystal by using a ground plane to isolate signals and reduce noise.
- Keep frequently switching signal lines away from the crystal area to prevent crosstalk.

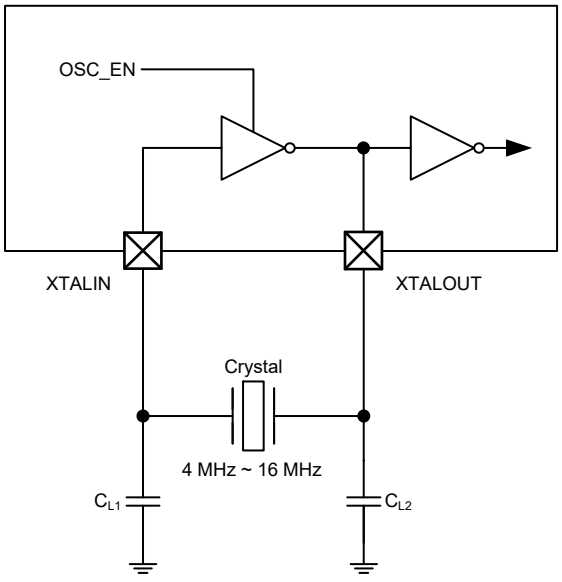


Figure 14. External Crystal, Ceramic and Resonators for HSE

The HSE crystal oscillator can be switched on or off using the HSEEN bit in the Global Clock Control Register (GCCR). The HSERDY flag in the Global Clock Status Register (GCSR) will indicate if the high speed external crystal oscillator is stable. While switching on the HSE, the HSE clock will still not be released until this HSERDY bit is set by the hardware. The specific delay period is well-known as "Start-up time". As the HSE becomes stable, the HSE clock can then be used directly as the system clock source.

High Speed Internal RC Oscillator – HSI

The high speed internal 16 MHz RC oscillator (HSI) is the default selection of clock source for the CPU when the device is powered up. The HSI RC oscillator provides a clock source in a lower cost because no external components are required. The HSI RC oscillator can be switched on or off using the HSIEN bit in the Global Clock Control Register (GCCR). The HSIRDY flag in the Global Clock Status Register (GCSR) will indicate if the internal RC oscillator is stable. The start-up time of HSI is shorter than the HSE crystal oscillator.

The accuracy of the frequency of the high speed internal RC oscillator HSI can be calibrated via the configuration options, but it is still less accurate than the HSE crystal oscillator. The applications, the environments and the cost will determine the use of the oscillators.

Software could configure the Power Saving Wakeup RC Clock Enable bit, PSRCEN, to 1 to force the HSI clock to be the system clock when waking up from the Deep-Sleep1 or Deep-Sleep2 mode. Subsequently, the system clock is back to the original clock source (HSE) if the original clock source ready flag is asserted. This function can reduce the wakeup time when using HSE as system clock.

Low Speed External Crystal Oscillator – LSE

The low speed external crystal or ceramic resonator oscillator with a 32,768 Hz frequency produces a low power but highly accurate clock source for the circuits of Real-Time Clock peripheral, Watchdog Timer or system clock. The associated hardware configuration is shown in the following figure. The crystal or ceramic resonator must be placed across the two LSE pins (X32KIN / X32KOUT) and the external components such as resistors and capacitors are necessary to make it oscillate properly. The LSE oscillator can be switched on or off by using the LSEEN bit in the RTC Control Register (RTCCR). The LSERDY flag in the Global Clock Status Register (GCSR) will indicate if the LSE clock is stable.

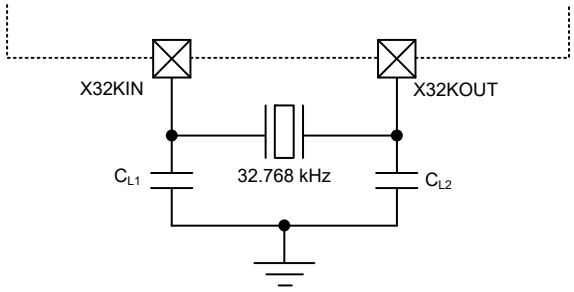


Figure 15. External Crystal, Ceramic and Resonators for LSE

Low Speed Internal RC Oscillator – LSI

The low speed internal RC oscillator with a frequency of about 32 kHz produces a low power clock source for the circuits of Real-Time Clock peripheral, Watchdog Timer or system clock. The LSI is also a low cost clock source because no external component is needed to make it oscillate. The frequency accuracy of the low speed internal RC oscillator LSI is shown in the corresponding data sheet. The LSIRDY flag in the Global Clock Status Register (GCSR) will indicate if the LSI clock is stable.

Clock Ready Flag

The CKCU provides clock ready flags for HSI, HSE, LSI and LSE to confirm these clocks are stable before using them as system clock source or other purposes. Software can check specific clock is ready or not by polling separate clock ready status bits in GCSR register.

System Clock (CK_SYS) Selection

After a system reset occurs, the default source of the system clock CK_SYS will be the high speed internal RC oscillator HSI. The CK_SYS clock may come from the HSI, HSE, LSE or LSI output clock and it can be switched from one clock source to another via the System Clock Switch field, SW, in the Global Clock Control Register (GCCR). The system will still run under the original clock until the destination clock gets ready. The corresponding clock ready status bit in the Global Clock Status Register (GCSR) will indicate whether the selected clock is ready to use or not. The CKCU also contains the clock source status bits in the Clock Source Status Register (CKST) to indicate which clock is currently used as the system clock. More details about clock enable function are described below.

If any following action takes effect, the HSI is always under enable state.

- Enable Clock monitor. (CKMEN)
- Configure clock switch field to select HSI. (SW)
- Configure HSI enable bit to 1. (HSIEN)

If any following action takes effect, the HSE is always under enable state.

- Configure clock switch field to select HSE. (SW)
- Configure HSE enable bit to 1. (HSEEN)

Programming guide of system clock selection is listed below.

1. Enable any source clock which will become the system clock.
2. Configure the SW field to change the system clock source after the corresponding clock ready flag is asserted. Note that the system clock will force to HSI if the clock monitor is enabled and HSE clock configured as system clock is stuck at 0 or 1.

HSE Clock Monitor

The main function of the oscillator check is enabled by the HSE Clock Monitor Enable bit CKMEN in the Global Clock Control Register (GCCR). The HSE clock monitor should be enabled after the HSE oscillator start-up delay and be disabled when the HSE oscillator is stopped. Once the HSE oscillator failure is detected, the HSE oscillator will automatically be disabled. The HSE clock stuck flag CKSF in the Global Clock Interrupt Register (GCIR) will be set and an interrupt of main oscillator failure will be generated if the clock stuck interrupt enable bit CKSIE in the GCIR is set. This failure interrupt is connected to the exception vector of CPU Non-Maskable Interrupt (NMI). If the HSE is directly used as the system clock source, when the HSE oscillator failure occurs, the HSE will be turned off and the system clock will be switched to the HSI automatically by the hardware.

Clock Output Capability

The device has the clock output capability to allow the clocks to be output on the specific external output pin CKOUT. The configuration registers of the corresponding GPIO port must be well configured in the Alternate Function I/O section, AFIO, to output the selected clock signal. There are seven output clock signals to be selected via the device clock output source selection field CKOUTSRC in the Global Clock Configuration Register (GCFGR).

Table 14. CKOUT Clock Source

CKOUTSRC[2:0]	Clock Source
000	CK_REF = CK_SYS / (CKREFPRE + 1) / 2
001	HCLKC / 16
010	CK_SYS / 16
011	CK_HSE / 16
100	CK_HSI / 16
101	CK_LSE
110	CK_LSI

Register Map

The following table shows the CKCU registers and reset values.

Table 15. CKCU Register Map

Register	Offset	Description	Reset Value
GCFGR	0x000	Global Clock Configuration Register	0x0000_0002
GCCR	0x004	Global Clock Control Register	0x0000_0803
GCSR	0x008	Global Clock Status Register	0x0000_0028
GCIR	0x00C	Global Clock Interrupt Register	0x0000_0000
AHBCFGR	0x020	AHB Configuration Register	0x0000_0000
AHBCCR	0x024	AHB Clock Control Register	0x0000_0065
APBCFGR	0x028	APB Configuration Register	0x0000_0000
APBCCR0	0x02C	APB Clock Control Register 0	0x0000_0000
APBCCR1	0x030	APB Clock Control Register 1	0x0000_0000
CKST	0x034	Clock Source Status Register	0x0100_0003
MCUDBGCR	0x304	MCU Debug Control Register	0x0000_0000

Register Descriptions

Global Clock Configuration Register – GCFGR

This register specifies the low power mode status and clock source for CKOUT.

Offset: 0x000

Reset value: 0x0000_0002

	31	30	29	28	27	26	25	24
	LPMOD			Reserved				
Type/Reset	WC	0 WC	0 RO	0				
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	CKREFPRE					Reserved		
Type/Reset	RW	0 RW	0 RW	0 RW	0 RW	0		
	7	6	5	4	3	2	1	0
	Reserved					CKOUTSRC		
Type/Reset						RW	0 RW	1 RW 0

Bits	Field	Descriptions
[31:29]	LPMOD	Lower Power Mode Status 000: When chip is in running mode 001: When chip once entered Sleep mode 010: When chip once entered Deep-Sleep1 mode 011: When chip once entered Deep-Sleep2 mode Others: Reserved Set by hardware. Reset by software writing b11x.
[15:11]	CKREFPRE	CK_REF Clock Prescaler Selection CK_REF = CK_SYS / (CKREFPRE + 1) / 2 00000: CK_REF = CK_SYS / 2 00001: CK_REF = CK_SYS / 4 ... 11111: CK_REF = CK_SYS / 64 Set and reset by software to control the CK_REF clock prescaler setting.
[2:0]	CKOUTSRC	CKOUT Clock Source Selection 000: CK_REF is selected, CK_REF = CK_SYS / (CKREFPRE + 1) / 2 001: (HCLKC / 16) is selected 010: (CK_SYS / 16) is selected 011: (CK_HSE / 16) is selected 100: (CK_HSI / 16) is selected 101: CK_LSE is selected 110: CK_LSI is selected 111: Reserved Set and reset by software to control the CKOUT clock source.

Global Clock Control Register – GCCR

This register specifies the clock enable bits.

Offset: 0x004

Reset value: 0x0000_0803

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved						PSRCEN	CKMEN
Type/Reset							RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved				HSIEN	HSEEN	Reserved	HSEGAIN
Type/Reset					RW 1	RW 0		RW 0
	7	6	5	4	3	2	1	0
	Reserved					SW		
Type/Reset						RW 0	RW 1	RW 1

Bits	Field	Descriptions
[17]	PSRCEN	Power Saving Wakeup RC Clock Enable 0: No action 1: Use Internal 16 MHz RC clock (HSI) as system clock after a Deep-Sleep1/2 mode wakeup Software can set PSRCEN high before entering the Deep-Sleep1/2 mode in order to reduce the waiting time after wakeup. When PSRCEN = 1, hardware will select HSI as clock source after the system wakeup from Deep-Sleep1/2 mode. Meanwhile, instruction can start execution since the HSI clock is provided to CPU. After the original clock source, which is selected as CK_SYS before entering the Deep-Sleep1/2 mode, is ready, hardware will switch back the clock source as originally.
[16]	CKMEN	HSE Clock Monitor Enable 0: Disable external 4 ~ 16 MHz crystal oscillator clock monitor 1: Enable external 4 ~ 16 MHz crystal oscillator clock monitor When hardware detects the HSE clock stuck at low/high state, internal hardware will switch the system clock to the internal high speed RC clock (HSI).
[11]	HSIEN	Internal High Speed Clock Enable 0: Internal 16 MHz RC oscillator clock is disabled 1: Internal 16 MHz RC oscillator clock is enabled Set and reset by software. This bit cannot be reset if the HSI clock is used as system clock.
[10]	HSEEN	External High Speed Clock Enable 0: External 4 ~ 16 MHz crystal oscillator clock is disabled 1: External 4 ~ 16 MHz crystal oscillator clock is enabled Set and reset by software. This bit cannot be reset if the HSE clock is used as system clock.
[8]	HSEGAIN	External High Speed Clock Gain Selection 0: HSE low gain mode 1: HSE high gain mode

Bits	Field	Descriptions
[2:0]	SW	System Clock Switch 010: CK_HSE as system clock 011: CK_HSI as system clock 110: CK_LSE as system clock 111: CK_LSI as system clock Others: CK_HSI as system clock These bits are used to select the CK_SYS source. If the HSE oscillator is used directly or indirectly as the system clock and the HSE clock monitor function is enabled, once the HSE failure is detected, these bits will be set by hardware to force HSI (b011) as the system clock. Note: When switching the system clock using the SW field, the system clock will not be immediately switched and a certain delay is necessary. Software can monitor the CKSWST field in the clock source status register CKST to make sure which clock is currently used as the system clock.

Global Clock Status Register – GCSR

This register indicates the clock ready status.

Offset: 0x008

Reset value: 0x0000_0028

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
		Reserved	LSIRDY	LSERDY	HSIRDY	HSERDY		Reserved
Type/Reset			RO 1	RO 0	RO 1	RO 0		

Bits	Field	Descriptions
[5]	LSIRDY	Internal Low Speed Clock Ready Flag 0: Internal 32 kHz RC oscillator clock is not ready 1: Internal 32 kHz RC oscillator clock is ready Set by hardware to indicate that the LSI is stable to be used.
[4]	LSERDY	External Low Speed Clock Ready Flag 0: External 32,768 Hz crystal oscillator clock is not ready 1: External 32,768 Hz crystal oscillator clock is ready Set by hardware to indicate that the LSE is stable to be used.
[3]	HSIRDY	Internal High Speed Clock Ready Flag 0: Internal 16 MHz RC oscillator clock is not ready 1: Internal 16 MHz RC oscillator clock is ready Set by hardware to indicate whether the HSI is stable or not.

Bits	Field	Descriptions
[2]	HSERDY	External High Speed Clock Ready Flag 0: External 4 ~ 16 MHz crystal oscillator clock is not ready 1: External 4 ~ 16 MHz crystal oscillator clock is ready Set by hardware to indicate that the HSE is stable to be used.

Global Clock Interrupt Register – GCIR

This register specifies interrupt enable and flag bits.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							CKSIE
Type/Reset								RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved							CKSF
Type/Reset								WC 0

Bits	Field	Descriptions
[16]	CKSIE	Clock Stuck Interrupt Enable 0: Disable clock stuck interrupt 1: Enable clock stuck interrupt Set and reset by software to enable/disable interrupt caused by clock monitor.
[0]	CKSF	Clock Stuck Interrupt Flag 0: Clock works normally 1: HSE clock is stuck Set by hardware when the HSE clock stuck and CKMEN is set. Reset by software writing 1.

AHB Configuration Register – AHBCFGR

This register specifies the system clock frequency.

Offset: 0x020

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					AHBPRE		
Type/Reset						RW 0	RW 0	RW 0

Bits	Field	Descriptions
[2:0]	AHBPRE	AHB Pre-scaler 000: CK_AHB = CK_SYS 001: CK_AHB = CK_SYS / 2 010: CK_AHB = CK_SYS / 4 011: CK_AHB = CK_SYS / 8 100: CK_AHB = CK_SYS / 16 101: CK_AHB = CK_SYS / 32 110: CK_AHB = CK_SYS / 32 111: CK_AHB = CK_SYS / 32 Set and reset by software to control the division factor of the AHB clock.

AHB Clock Control Register – AHBCCR

This register specifies clock enable bits of AHB peripherals.

Offset: 0x024

Reset value: 0x0000_0065

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
		Reserved	PFEN		Reserved	PCEN	PBEN	PAEN
Type/Reset			RW 0			RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved				CKREFEN	Reserved		
Type/Reset					RW 0			
	7	6	5	4	3	2	1	0
	Reserved	APBEN	BMEN		Reserved	SRAMEN	Reserved	FMCEN
Type/Reset		RW 1	RW 1			RW 1		RW 1

Bits	Field	Descriptions
[21]	PFEN	GPIO Port F Clock Enable 0: Port F clock is disabled 1: Port F clock is enabled Set and reset by software
[18]	PCEN	GPIO Port C Clock Enable 0: Port C clock is disabled 1: Port C clock is enabled Set and reset by software.
[17]	PBEN	GPIO Port B Clock Enable 0: Port B clock is disabled 1: Port B clock is enabled Set and reset by software.
[16]	PAEN	GPIO Port A Clock Enable 0: Port A clock is disabled 1: Port A clock is enabled Set and reset by software.
[11]	CKREFEN	CK_REF Clock Enable 0: CK_REF clock is disabled 1: CK_REF clock is enabled Set and reset by software.
[6]	APBEN	APB bridge Clock Enable 0: APB bridge clock is automatically disabled by hardware during Sleep mode 1: APB bridge clock is always enabled during Sleep mode Set and reset by software. Users can set APBEN as 0 to reduce power consumption if the APB bridge is unused during Sleep mode.
[5]	BMEN	Bus Matrix Clock Enable 0: Bus Matrix clock is automatically disabled by hardware during Sleep mode 1: Bus Matrix clock is always enabled during Sleep mode Set and reset by software. Users can set BMEN as 0 to reduce power consumption if the bus matrix is unused during Sleep mode.

Bits	Field	Descriptions
[2]	SRAMEN	SRAM Clock Enable 0: SRAM clock is automatically disabled by hardware during Sleep mode 1: SRAM clock is always enabled during Sleep mode Set and reset by software. Users can set SRAMEN as 0 to reduce power consumption if the SRAM is unused during Sleep mode.
[0]	FMCEN	Flash Memory Controller Clock Enable 0: FMC clock is automatically disabled by hardware during Sleep mode 1: FMC clock is always enabled during Sleep mode Set and reset by software. Users can set FMCEN as 0 to reduce power consumption if the Flash Memory is unused during Sleep mode.

APB Configuration Register – APBCFGR

This register specifies the ADC conversion clock frequency.

Offset: 0x028

Reset value: 0x0001_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved					ADCDIV		
Type/Reset						RW 0	RW 0	RW 1
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved							
Type/Reset								

Bits	Field	Descriptions
[18:16]	ADCDIV	ADC Clock Frequency Division Selection 000: CK_ADC = CK_AHB 001: CK_ADC = CK_AHB / 2 010: CK_ADC = CK_AHB / 4 011: CK_ADC = CK_AHB / 8 100: CK_ADC = CK_AHB / 16 101: CK_ADC = CK_AHB / 32 110: CK_ADC = CK_AHB / 64 111: CK_ADC = CK_AHB / 3 Set and reset by software to control the ADC conversion clock division factor.

APB Clock Control Register 0 – APBCCR0

This register specifies clock enable bits of APB peripherals.

Offset: 0x02C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
		Reserved	LEDCEN	Reserved				
Type/Reset			RW 0					
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	EXTIEN	AFIOEN	Reserved		UR1EN	UR0EN	Reserved	
Type/Reset	RW 0	RW 0			RW 0	RW 0		
	7	6	5	4	3	2	1	0
	Reserved			SPIEN	Reserved			I2CEN
Type/Reset				RW 0				RW 0

Bits	Field	Descriptions
[29]	LEDCEN	LED Controller Clock Enable 0: LEDC clock is disabled 1: LEDC clock is enabled Set and reset by software.
[15]	EXTIEN	External Interrupt Clock Enable 0: EXTI clock is disabled 1: EXTI clock is enabled Set and reset by software.
[14]	AFIOEN	Alternate Function I/O Clock Enable 0: AFIO clock is disabled 1: AFIO clock is enabled Set and reset by software.
[11]	UR1EN	UART1 Clock Enable 0: UART1 clock is disabled 1: UART1 clock is enabled Set and reset by software.
[10]	UR0EN	UART0 Clock Enable 0: UART0 clock is disabled 1: UART0 clock is enabled Set and reset by software.
[4]	SPIEN	SPI Clock Enable 0: SPI clock is disabled 1: SPI clock is enabled Set and reset by software.
[0]	I2CEN	I^2C Clock Enable 0: I^2C clock is disabled 1: I^2C clock is enabled Set and reset by software.

APB Clock Control Register 1 – APBCCR1

This register specifies clock enable bits of APB peripherals.

Offset: 0x030

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved	SCTM2EN	SCTM1EN	SCTM0EN	Reserved			ADCCEN
Type/Reset		RW 0	RW 0	RW 0				RW 0
	23	22	21	20	19	18	17	16
	Reserved							BFTMEN
Type/Reset								RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved	VDDREN	Reserved	WDTREN	Reserved			
Type/Reset		RW 0		RW 0				

Bits	Field	Descriptions
[30]	SCTM2EN	SCTM2 Clock Enable 0: SCTM2 clock is disabled 1: SCTM2 clock is enabled Set and reset by software.
[29]	SCTM1EN	SCTM1 Clock Enable 0: SCTM1 clock is disabled 1: SCTM1 clock is enabled Set and reset by software.
[28]	SCTM0EN	SCTM0 Clock Enable 0: SCTM0 clock is disabled 1: SCTM0 clock is enabled Set and reset by software.
[24]	ADCCEN	ADC Controller Clock Enable 0: ADC clock is disabled 1: ADC clock is enabled Set and reset by software.
[16]	BFTMEN	BFTM Clock Enable 0: BFTM clock is disabled 1: BFTM clock is enabled Set and reset by software.
[6]	VDDREN	V_{DD} Domain Clock Enable for Register Access 0: Register access clock is disabled 1: Register access clock is enabled Set and reset by software.
[4]	WDTREN	Watchdog Timer Clock Enable for Register Access 0: Register access clock is disabled 1: Register access clock is enabled Set and reset by software.

Clock Source Status Register – CKST

This register specifies status of various clock sources.

Offset: 0x034

Reset value: 0x0100_0003

	31	30	29	28	27	26	25	24
	Reserved					HSIST		
Type/Reset						RO 0	RO 0	RO 1
	23	22	21	20	19	18	17	16
	Reserved						HSEST	
Type/Reset							RO 0	RO 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					CKSWST		
Type/Reset						RO 0	RO 1	RO 1

Bits	Field	Descriptions
[26:24]	HSIST	Internal High Speed Clock Occupation Status (CK_HSI) xx1: HSI is used by System Clock (CK_SYS) (SW = 0x3) x1x: Reserved 1xx: HSI is used by Clock Monitor
[17:16]	HSEST	External High Speed Clock Occupation Status (CK_HSE) x1: HSE is used by System Clock (CK_SYS) (SW = 0x2) 1x: Reserved
[2:0]	CKSWST	Clock Switch Status 00x: Reserved 010: CK_HSE as system clock 011: CK_HSI as system clock 110: CK_LSE as system clock 111: CK_LSI as system clock The fields are status to indicate which clock source is being used as system clock currently.

MCU Debug Control Register – MCUDBGCR

This register specifies the debug control of MCU.

Offset: 0x304

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							DBSCTM2
Type/Reset								RW 0
	23	22	21	20	19	18	17	16
	DBSCTM1	DBSCTM0		Reserved	DBUR1	DBUR0	Reserved	DBBFTM
Type/Reset	RW 0	RW 0			RW 0	RW 0		RW 0
	15	14	13	12	11	10	9	8
	Reserved	DBDSLP2	Reserved	DBI2C	Reserved	DBSPI		Reserved
Type/Reset		RW 0		RW 0		RW 0		
	7	6	5	4	3	2	1	0
	Reserved				DBWDT	Reserved	DBDSLP1	DBSLP
Type/Reset					RW 0		RW 0	RW 0

Bits	Field	Descriptions
[24]	DBSCTM2	SCTM2 Debug Mode Enable 0: SCTM2 counter continues even if the core is halted 1: SCTM2 counter is stopped when the core is halted Set and reset by software.
[23]	DBSCTM1	SCTM1 Debug Mode Enable 0: SCTM1 counter continues even if the core is halted 1: SCTM1 counter is stopped when the core is halted Set and reset by software.
[22]	DBSCTM0	SCTM0 Debug Mode Enable 0: SCTM0 counter continues even if the core is halted 1: SCTM0 counter is stopped when the core is halted Set and reset by software.
[19]	DBUR1	UART1 Debug Mode Enable 0: Same behavior as in normal mode 1: UART1 FIFO timeout is frozen when the core is halted Set and reset by software.
[18]	DBUR0	UART0 Debug Mode Enable 0: Same behavior as in normal mode 1: UART0 FIFO timeout is frozen when the core is halted Set and reset by software.
[16]	DBBFTM	BFTM Debug Mode Enable 0: BFTM counter continues to count even if the core is halted 1: BFTM counter is stopped when the core is halted Set and reset by software.
[14]	DBDSLP2	Debug Deep-Sleep2 Mode 0: LDO = Off (but turn on DMOS), FCLK = Off, and CM0PEN = 0 in Deep-Sleep2 mode 1: LDO = On, FCLK = On, and CM0PEN = 1 in Deep-Sleep2 mode Set and reset by software.

Bits	Field	Descriptions
[12]	DBI2C	I²C Debug Mode Enable 0: Same behavior as in normal mode 1: I²C timeout is frozen when the core is halted Set and reset by software.
[10]	DBSPI	SPI Debug Mode Enable 0: Same behavior as in normal mode 1: SPI FIFO timeout is frozen when the core is halted Set and reset by software.
[3]	DBWDT	Watchdog Timer Debug Mode Enable 0: Watchdog Timer counter continues to count even if the core is halted 1: Watchdog Timer counter is stopped when the core is halted Set and reset by software.
[1]	DBDSLP1	Debug Deep-Sleep1 Mode 0: LDO = Low power mode, FCLK = Off, and CM0PEN = 0 in Deep-Sleep1 mode 1: LDO = On, FCLK = On, and CM0PEN = 1 in Deep-Sleep1 mode Set and reset by software.
[0]	DBSLP	Debug Sleep Mode 0: LDO = On, FCLK = On, and CM0PEN = 0 in Sleep mode 1: LDO = On, FCLK = On, and CM0PEN = 1 in Sleep mode Set and reset by software.

7 Reset Control Unit (RSTCU)

Introduction

The Reset Control Unit, RSTCU, has three kinds of reset, the power-on reset, system reset and APB unit reset. The power-on reset, known as a cold reset, resets the full system during a power up. A system reset resets the processor core and peripheral IP components with the exception of the debug port controller. The resets can be triggered by an external signal, internal events and the reset generators. More information about these resets will be described in the following section.

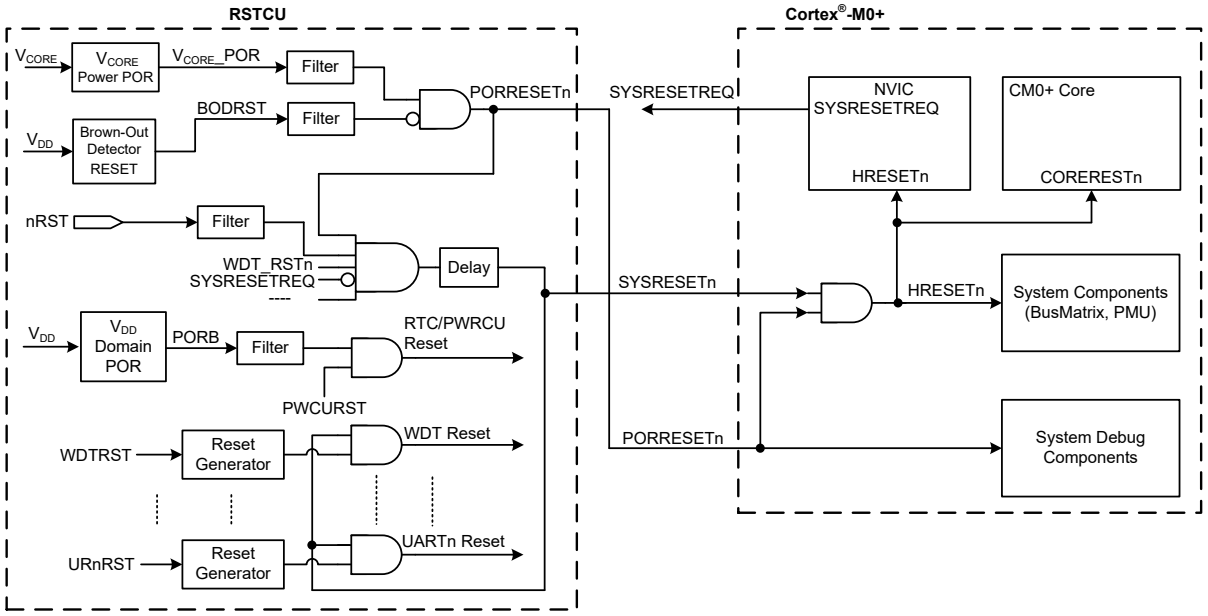


Figure 16. RSTCU Block Diagram

Functional Descriptions

Power-On Reset

The Power-on reset, POR, is generated by either an external reset or the internal reset generator. Both types have an internal filter to prevent glitches from causing erroneous reset operations. By referring to Figure 17, the V_{CORE}_POR active low signal will be de-asserted when the internal LDO voltage regulator is ready to provide the V_{CORE} power. In addition to the V_{CORE}_POR signal, the Power Control Unit, PWRCU, will assert the BODF signal as a Power-Down Reset, PDR, when the BODEN bit in the LVDCSR register is set and the brown-out event occurs. For more details about the PWRCU function, refer to the PWRCU chapter.

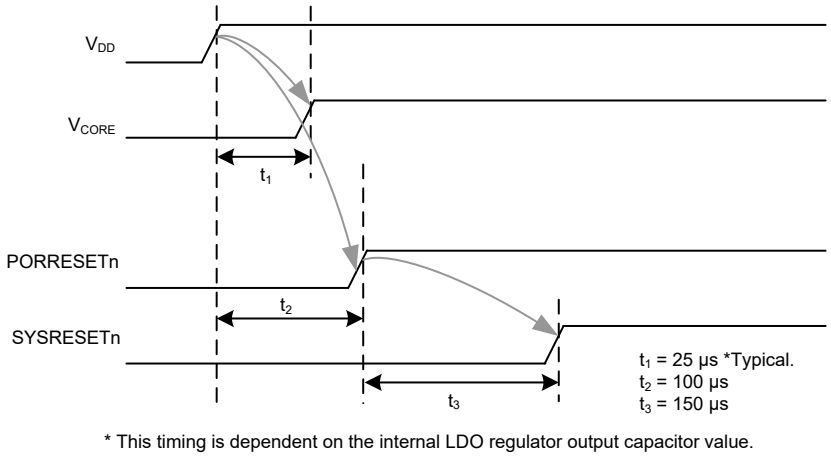


Figure 17. Power-On Reset Sequence

System Reset

A system reset is generated by a power-on reset (PORRESETn), a Watchdog Timer reset (WDT_RSTn), an nRST pin event or a software reset (SYSRESETREQ) event. For more information about SYSRESETREQ event, refer to the related chapter in the Cortex®-M0+ reference manual.

AHB and APB Unit Reset

The AHB and APB unit reset can be divided into hardware and software resets. A hardware reset can be generated by either power on reset or system reset for all AHB and APB units. Each functional IP connected to the AHB and APB buses can be reset individually through the associated software reset bits in the RSTCU. For example, the application software can generate a UART0 reset via the UR0RST bit in the APBPRSTR0 register.

Register Map

The following table shows the RSTCU registers and reset values.

Table 16. RSTCU Register Map

Register	Offset	Description	Reset Value
GRSR	0x100	Global Reset Status Register	0x0000_0008
AHBPRSTR	0x104	AHB Peripheral Reset Register	0x0000_0000
APBPRSTR0	0x108	APB Peripheral Reset Register 0	0x0000_0000
APBPRSTR1	0x10C	APB Peripheral Reset Register 1	0x0000_0000

Register Descriptions

Global Reset Status Register – GRSR

This register specifies a variety of reset status conditions.

Offset: 0x100

Reset value: 0x0000_0008

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved				PORSTF	WDTRSTF	EXTRSTF	NVICRSTF
Type/Reset					WC 1	WC 0	WC 0	WC 0

Bits	Field	Descriptions
[3]	PORSTF	V_{CORE} Power On Reset Flag 0: No POR occurred 1: POR occurred This bit is set by hardware when a power-on reset occurs and reset by writing 1 into it.
[2]	WDTRSTF	Watchdog Timer Reset Flag 0: No Watchdog Timer reset occurred 1: Watchdog Timer occurred This bit is set by hardware when a watchdog timer reset occurs and reset by writing 1 into it or by hardware when a power-on reset occurs.
[1]	EXTRSTF	External Pin Reset Flag 0: No pin reset occurred 1: Pin reset occurred This bit is set by hardware when an external pin reset occurs and reset by writing 1 into it or by hardware when a power-on reset occurs.
[0]	NVICRSTF	NVIC Reset Flag 0: No NVIC asserting system reset occurred 1: NVIC asserting system reset occurred This bit is set by hardware when a system reset occurs and reset by writing 1 into it or by hardware when a power-on reset occurs.

AHB Peripheral Reset Register – AHBPRSTR

This register specifies several AHB peripherals software reset control bits.

Offset: 0x104
Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
		Reserved	PFRST		Reserved	PCRST	PBRST	PARST
Type/Reset			RW 0			RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	Reserved							
Type/Reset								

Bits	Field	Descriptions
[13]	PFRST	GPIO Port F Reset Control 0: No reset 1: Reset Port F This bit is set by software and cleared to 0 by hardware automatically.
[10]	PCRST	GPIO Port C Reset Control 0: No reset 1: Reset Port C This bit is set by software and cleared to 0 by hardware automatically.
[9]	PBRST	GPIO Port B Reset Control 0: No reset 1: Reset Port B This bit is set by software and cleared to 0 by hardware automatically.
[8]	PARST	GPIO Port A Reset Control 0: No reset 1: Reset Port A This bit is set by software and cleared to 0 by hardware automatically.

APB Peripheral Reset Register 0 – APBPRSTR0

This register specifies several APB peripherals software reset control bits.

Offset: 0x108

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
		Reserved	LEDCRST	Reserved				
Type/Reset			RW 0					
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	Reserved							
Type/Reset								
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	EXTIRST	AFIORST	Reserved		UR1RST	UR0RST	Reserved	
Type/Reset	RW 0	RW 0			RW 0	RW 0		
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	Reserved			SPIRST	Reserved			I2CRST
Type/Reset				RW 0				RW 0

Bits	Field	Descriptions
[29]	LEDCRST	LED Controller Reset Control 0: No reset 1: Reset LED Controller This bit is set by software and cleared to 0 by hardware automatically.
[15]	EXTIRST	External Interrupt Controller Reset Control 0: No reset 1: Reset EXTI This bit is set by software and cleared to 0 by hardware automatically.
[14]	AFIORST	Alternate Function I/O Reset Control 0: No reset 1: Reset Alternate Function I/O This bit is set by software and cleared to 0 by hardware automatically.
[11]	UR1RST	UART1 Reset Control 0: No reset 1: Reset UART1 This bit is set by software and cleared to 0 by hardware automatically.
[10]	UR0RST	UART0 Reset Control 0: No reset 1: Reset UART0 This bit is set by software and cleared to 0 by hardware automatically.
[4]	SPIRST	SPI Reset Control 0: No reset 1: Reset SPI This bit is set by software and cleared to 0 by hardware automatically.
[0]	I2CRST	I^2C Reset Control 0: No reset 1: Reset I^2C This bit is set by software and cleared to 0 by hardware automatically.

APB Peripheral Reset Register 1 – APBPRSTR1

This register specifies several APB peripherals software reset control bits.

Offset: 0x10C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved	SCTM2RST	SCTM1RST	SCTM0RST	Reserved			ADCRST
Type/Reset		RW 0	RW 0	RW 0				RW 0
	23	22	21	20	19	18	17	16
	Reserved							BFTMRST
Type/Reset								RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved			WDTRST	Reserved			
Type/Reset				RW 0				

Bits	Field	Descriptions
[30]	SCTM2RST	SCTM2 Reset Control 0: No reset 1: Reset SCTM2 This bit is set by software and cleared to 0 by hardware automatically.
[29]	SCTM1RST	SCTM1 Reset Control 0: No reset 1: Reset SCTM1 This bit is set by software and cleared to 0 by hardware automatically.
[28]	SCTM0RST	SCTM0 Reset Control 0: No reset 1: Reset SCTM0 This bit is set by software and cleared to 0 by hardware automatically.
[24]	ADCRST	ADC Reset Control 0: No reset 1: Reset A/D Converter This bit is set by software and cleared to 0 by hardware automatically.
[16]	BFTMRST	BFTM Reset Control 0: No reset 1: Reset BFTM This bit is set by software and cleared to 0 by hardware automatically.
[4]	WDTRST	Watchdog Timer Reset Control 0: No reset 1: Reset Watchdog Timer This bit is set by software and cleared to 0 by hardware automatically.

8 General Purpose I/O (GPIO)

Introduction

There are up to 42 General Purpose I/O ports, GPIO, named PA0 ~ PA15, PB0 ~ PB15, PC0 ~ PC7 and PF0 ~ PF1 for the devices to implement the logic input/output functions. Each of the GPIO ports has related control and configuration registers to satisfy the requirement of specific applications. The actual available General Purpose I/O port numbers are dependent on the device specification and package type. Refer to the device datasheet for detailed information.

The GPIO ports are pin-shared with other alternative functions (AFs) to obtain maximum flexibility on the package pins. The GPIO pins can be used as alternative functional pins by configuring the corresponding registers regardless of the AF input or output pins.

The external interrupts on the GPIO pins of the device have related control and configuration registers in the External Interrupt Control Unit (EXTI).

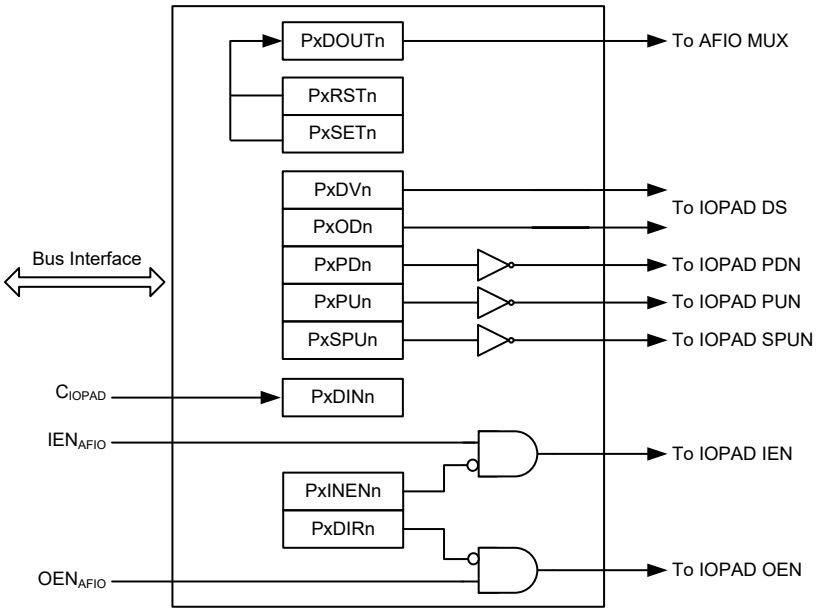


Figure 18. GPIO Block Diagram

Features

- Input/output direction control
- Input weak pull-up/pull-down and strong pull-up control
- Output push-pull/open-drain enable control
- Output set/reset control
- Output drive current selection
- External interrupt with programmable trigger edge – using EXTI configuration registers
- Analog input/output configurations – using AFIO configuration registers
- Alternate function input/output configurations – using AFIO configuration registers
- Port configuration lock

Functional Descriptions

Default GPIO Pin Configuration

During or just after the reset period, the alternative functions are all inactive and the GPIO ports are configured into the input disable floating mode, i.e. input disabled without pull-up/pull-down resistors. Only the boot and Serial-Wired Debug pins which are pin-shared with the I/O pins are active after a device reset.

- PA9_BOOT: Input enable with internal pull-up
- SWCLK: Input enable with internal pull-up
- SWDIO: Input enable with internal pull-up

General Purpose I/O – GPIO

The GPIO pins can be configured as inputs or outputs via the data direction control registers PxDIRCR (where x = A ~ C, F). When the GPIO pins are configured as input pins, the data on the external pads can be read if the enable bits in the input enable function register PxINER are set. The GPIO pull-up/pull-down registers PxPUR/PxPDR can be configured to fit specific applications. When the pull-up and pull-down functions are both enabled, the pull-up function has the higher priority while the pull-down function will be blocked until the pull-up function is released.

The GPIO pins can be configured as output pins where the output data is latched into the data register PxDOUTR. The output type can be setup to be either push-pull or open-drain by the open-drain selection register PxODR. Only one or several specific bits of the output data will be set or reset by configuring the port output set/reset control register PxSRR or the port output reset register PxRR without affecting the unselected bits. As the port output set and reset functions are both enabled, the port output set function has the higher priority and the port output reset function will be blocked. The output driving current of the GPIO pins can be selected by configuring the drive current selection register PxDRVR.

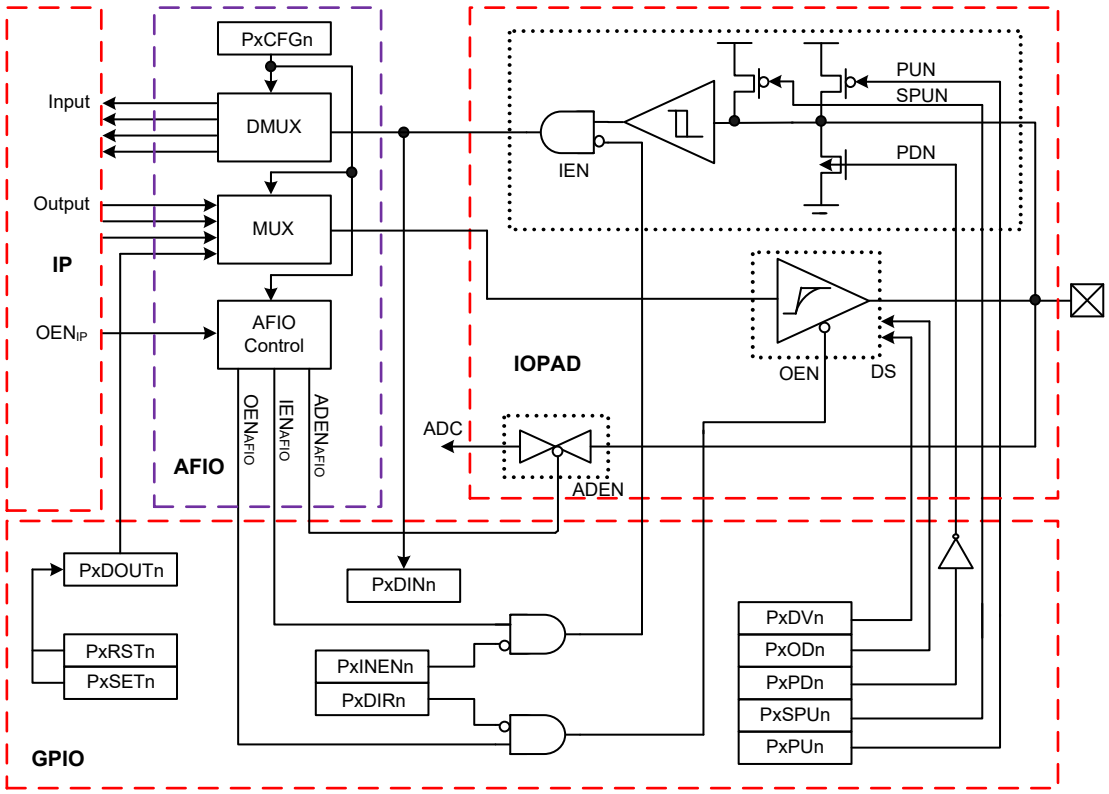


Figure 19. AFIO/GPIO Control Signal

PxDINn / PxDOUTn (x = A ~ C, F): Data Input / Data Output

PxRSTn / PxSETn (x = A ~ C, F): Reset / Set

PxDIRn (x = A ~ C, F): Direction

PxINENn (x = A ~ C, F): Input Enable

PxDVn (x = A ~ C, F): Output Drive

PxODn (x = A ~ C, F): Open-Drain

PxPDn / PxPUn / PxSPUn (x = A ~ C, F): Weak Pull-Down / Pull-Up / Strong Pull-Up

PxCFGn (x = A ~ C, F): AFIO Configuration

Table 17. AFIO, GPIO and I/O Pad Control Signal True Table

Type	AFIO			GPIO		PAD		
	$ADEN_{AFIO}$	OEN_{AFIO}	IEN_{AFIO}	PxDIRn	PxINENn	ADEN	OEN	IEN
GPIO Input (Note)	1	1	1	0	1	1	1	0
GPIO Output (Note)	1	1	1	1	0 (1 if need)	1	0	1 (0)
AFIO Input	1	1	0	0	X	1	1	0
AFIO Output	1	0	1	X	0 (1 if need)	1	0	1 (0)
ADC Input	0	1	1	0	0 (1 if need)	0	1	1 (0)
OSC Output	0	1	1	0	0 (1 if need)	0	1	1 (0)

Note: The signals, IEN and OEN, for I/O pads are derived from the GPIO register bits PxINENn and PxDIRn respectively when the associated pin is configured in the GPIO input/output mode.

GPIO Locking Mechanism

The GPIO also offers a lock function to lock the port until a reset event occurs. The PxLOCKR (x = A ~ C, F) registers are used to lock the port x and lock control options. The value 0x5FA0 is written into the PxLKEY field in the PxLOCKR registers to freeze the PxDIRCR, PxINER, PxPUR, PxSPUR, PxPDR, PxODR, PxDRVR control and AFIO mode configuration (GPxCFGHR or GPxCFGLR, where x = A ~ C, F). If the value in the PxLOCKR register is 0x5FA0_0001, it means that the Port x Lock function is enabled and the Port x pin 0 is frozen.

Register Map

The following table shows the GPIO registers and reset values of the Port A ~ C, F.

Table 18. GPIO Register Map

Register	Offset	Description	Reset Value
GPIO A Base Address = 0x400B_0000			
PADIRCR	0x000	Port A Data Direction Control Register	0x0000_0000
PAINER	0x004	Port A Input Function Enable Control Register	0x0000_0200
PAPUR	0x008	Port A Pull-Up Selection Register	0x0000_3200
PAPDR	0x00C	Port A Pull-Down Selection Register	0x0000_0000
PAODR	0x010	Port A Open-Drain Selection Register	0x0000_0000
PADRVR	0x014	Port A Drive Current Selection Register	0x0000_0000
PALOCKR	0x018	Port A Lock Register	0x0000_0000
PADINR	0x01C	Port A Data Input Register	0x0000_3200
PADOUTR	0x020	Port A Data Output Register	0x0000_0000
PASRR	0x024	Port A Output Set/Reset Control Register	0x0000_0000
PARR	0x028	Port A Output Reset Control Register	0x0000_0000
PASCER	0x02C	Port A Sink Current Enhanced Selection Register	0x0000_0000
GPIO B Base Address = 0x400B_2000			
PBDIRCR	0x000	Port B Data Direction Control Register	0x0000_0000
PBINER	0x004	Port B Input Function Enable Control Register	0x0000_0000
PBPUR	0x008	Port B Pull-Up Selection Register	0x0000_0000
PBPDR	0x00C	Port B Pull-Down Selection Register	0x0000_0000
PBODR	0x010	Port B Open-Drain Selection Register	0x0000_0000
PBDRVR	0x014	Port B Drive Current Selection Register	0x0000_0000
PBLOCKR	0x018	Port B Lock Register	0x0000_0000
PBDINR	0x01C	Port B Data Input Register	0x0000_0000
PBDOUTR	0x020	Port B Data Output Register	0x0000_0000
PBSRR	0x024	Port B Output Set/Reset Control Register	0x0000_0000
PBRR	0x028	Port B Output Reset Control Register	0x0000_0000
PBSCER	0x02C	Port B Sink Current Enhanced Selection Register	0x0000_0000
GPIO C Base Address = 0x400B_4000			
PCDIRCR	0x000	Port C Data Direction Control Register	0x0000_0000
PCINER	0x004	Port C Input Function Enable Control Register	0x0000_0000
PCPUR	0x008	Port C Pull-Up Selection Register	0x0000_0000
PCPDR	0x00C	Port C Pull-Down Selection Register	0x0000_0000
PCODR	0x010	Port C Open-Drain Selection Register	0x0000_0000
PCDRVR	0x014	Port C Drive Current Selection Register	0x0000_0000

Register	Offset	Description	Reset Value
PCLOCKR	0x018	Port C Lock Register	0x0000_0000
PCDINR	0x01C	Port C Data Input Register	0x0000_0000
PCDOUTR	0x020	Port C Data Output Register	0x0000_0000
PCSRR	0x024	Port C Output Set/Reset Control Register	0x0000_0000
PCRR	0x028	Port C Output Reset Control Register	0x0000_0000
PCSCER	0x02C	Port C Sink Current Enhanced Selection Register	0x0000_0000
GPIO F Base Address = 0x400B_A000			
PFDIRCR	0x000	Port F Data Direction Control Register	0x0000_0000
PFINER	0x004	Port F Input Function Enable Control Register	0x0000_0000
PFPUR	0x008	Port F Pull-Up Selection Register	0x0000_0000
PFPDR	0x00C	Port F Pull-Down Selection Register	0x0000_0000
PFODR	0x010	Port F Open-Drain Selection Register	0x0000_0000
PFDRVR	0x014	Port F Drive Current Selection Register	0x0000_0000
PFLOCKR	0x018	Port F Lock Register	0x0000_0000
PFDINR	0x01C	Port F Data Input Register	0x0000_0000
PFDOUTR	0x020	Port F Data Output Register	0x0000_0000
PFSRR	0x024	Port F Output Set and Reset Control Register	0x0000_0000
PFRR	0x028	Port F Output Reset Control Register	0x0000_0000
PFSCER	0x02C	Port F Sink Current Enhanced Selection Register	0x0000_0000

Register Descriptions

Port A Data Direction Control Register – PADIRCR

This register is used to control the direction of the GPIO Port A pin as input or output.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PADIR							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PADIR							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PADIRn	GPIO Port A pin n Direction Control Bits (n = 0 ~ 15) 0: Pin n is in input mode 1: Pin n is in output mode

Port A Input Function Enable Control Register – PAINER

This register is used to enable or disable the GPIO Port A input function.

Offset: 0x004

Reset value: 0x0000_0200

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PAINEN							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 1	RW 0
	7	6	5	4	3	2	1	0
	PAINEN							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PAINENn	GPIO Port A pin n Input Enable Control Bits (n = 0 ~ 15) 0: Pin n input function is disabled 1: Pin n input function is enabled When the pin n input function is disabled, the input Schmitt trigger will be turned off and the Schmitt trigger output will remain at a zero state.

Port A Pull-Up Selection Register – PAPUR

This register is used to enable or disable the GPIO Port A pull-up function. There are two pull-up functions, including weak pull-up and strong pull-up.

Offset: 0x008

Reset value: 0x0000_3200

	31	30	29	28	27	26	25	24
	PASPU							
Type/Reset	RW 0	RW 0	RW 1	RW 1	RW 0	RW 0	RW 1	RW 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PASPU							
Type/Reset	RW 0	RW 0	RW 1	RW 1	RW 0	RW 0	RW 1	RW 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	PAPU							
Type/Reset	RW 0	RW 0	RW 1	RW 1	RW 0	RW 0	RW 1	RW 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PAPU							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:16]	PASPUn	GPIO Port A pin n Strong Pull-Up Selection Control Bits (n = 0 ~ 15) 0: Pin n strong pull-up function is disabled 1: Pin n strong pull-up function is enabled The strong pull-up function provides stronger pull-up capability than weak pull-up function. It can be configured individually as weak pull-up or strong pull-up function by setting the corresponding PAPUn or PASPUn bit. When the strong pull-up and weak pull-up functions are both enabled can provide a maximum internal pull-up effect in GPIO port. Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.
[15:0]	PAPUn	GPIO Port A pin n Weak Pull-Up Selection Control Bits (n = 0 ~ 15) 0: Pin n weak pull-up function is disabled 1: Pin n weak pull-up function is enabled Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.

Port A Pull-Down Selection Register – PAPDR

This register is used to enable or disable the GPIO Port A pull-down function.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PAPD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PAPD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PAPDn	GPIO Port A pin n Pull-Down Selection Control Bits (n = 0 ~ 15) 0: Pin n pull-down function is disabled 1: Pin n pull-down function is enabled Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.

Port A Open-Drain Selection Register – PAODR

This register is used to enable or disable the GPIO Port A open-drain function.

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PAOD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PAOD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PAODn	GPIO Port A pin n Open-Drain Selection Control Bits (n = 0 ~ 15) 0: Pin n Open-Drain output is disabled (The output type is CMOS output) 1: Pin n Open-Drain output is enabled (The output type is open-drain output) Note: When the open-drain function is enabled, the pin n internal pull-up or pull-down configuration will be invalid.

Port A Drive Current Selection Register – PADRVR

This register specifies the GPIO Port A output driving current.

Offset: 0x014

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	PADV15		PADV14		PADV13		PADV12	
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	PADV11		PADV10		PADV9		PADV8	
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	PADV7		PADV6		PADV5		PADV4	
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PADV3		PADV2		PADV1		PADV0	
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:0]	PADVn[1:0]	GPIO Port A pin n Drive Current Selection Control Bits (n = 0 ~ 15) 00: 4 mA source/sink current 01: 8 mA source/sink current 10: 12 mA source/sink current 11: 16 mA source/sink current

Port A Lock Register – PALOCKR

This register specifies the GPIO Port A lock configuration.

Offset: 0x018

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	PALKEY							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	PALKEY							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	PALOCK							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PALOCK							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:16]	PALKEY	GPIO Port A Lock Key 0x5FA0: Port A Lock function is enabled Others: Port A Lock function is disabled To lock the Port A function, a value of 0x5FA0 should be written into the PALKEY field in this register. To execute a successful write operation on this lock register, the value written into the PALKEY field must be 0x5FA0. If the value written into this field is not equal to 0x5FA0, any write operations on the PALOCKR register will be aborted. The result of a read operation on the PALKEY field returns the GPIO Port A Lock Status which indicates whether the GPIO Port A is locked or not. If the read value of the PALKEY field is 0, this indicates that the GPIO Port A Lock function is disabled. Otherwise, it indicates that the GPIO Port A Lock function is enabled as the read value is equal to 1.
[15:0]	PALOCKn	GPIO Port A Pin n Lock Control Bits (n = 0 ~ 15) 0: Port A Pin n is not locked 1: Port A Pin n is locked The PALOCKn bits are used to lock the configurations of corresponding GPIO Pins when the correct Lock Key is applied to the PALKEY field. The locked configurations including PADIRn, PAINENn, PAPUn, PASPUn, PAPDn, PAODn and PADVn setting in the related GPIO registers. Additionally, the GPACFGHR or GPACFGLR register which is used to configure the alternative function of the associated GPIO pin will also be locked. Note that the PALOCKR register can only be written once which means that PALKEY and PALOCKn (lock control bit) should be written together and cannot be changed until a system reset or GPIO Port A reset occurs.

Port A Data Input Register – PADINR

This register specifies the GPIO Port A input data.

Offset: 0x01C

Reset value: 0x0000_3200

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PADIN							
Type/Reset	RO 0	RO 0	RO 1	RO 1	RO 0	RO 0	RO 1	RO 0
	7	6	5	4	3	2	1	0
	PADIN							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

Bits	Field	Descriptions
[15:0]	PADINn	GPIO Port A pin n Data Input Bits (n = 0 ~ 15) 0: The input data of the corresponding pin is 0 1: The input data of the corresponding pin is 1

Port A Output Data Register – PADOUTR

This register specifies the GPIO Port A output data.

Offset: 0x020

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PADOUT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PADOUT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PADOUTn	GPIO Port A pin n Data Output Bits (n = 0 ~ 15) 0: Data to be output on pin n is 0 1: Data to be output on pin n is 1

Port A Output Set/Reset Control Register – PASRR

This register is used to set or reset the corresponding bit of the GPIO Port A output data.

Offset: 0x024

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	PARST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PARST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	PASET							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PASET							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

Bits	Field	Descriptions
[31:16]	PARSTn	GPIO Port A pin n Output Reset Control Bits (n = 0 ~ 15) 0: No effect on the PADOUTn bit 1: Reset the PADOUTn bit Note that when the PARSTn bit in this register or the PARSTn bit in the PARR register is enabled, the reset function on the PADOUTn bit will take effect.
[15:0]	PASETn	GPIO Port A pin n Output Set Control Bits (n = 0 ~ 15) 0: No effect on the PADOUTn bit 1: Set the PADOUTn bit Note that the function enabled by the PASETn bit has the higher priority if both the PASETn and PARSTn bits are set at the same time.

Port A Output Reset Register – PARR

This register is used to reset the corresponding bit of the GPIO Port A output data.

Offset: 0x028

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PARST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
	PARST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

Bits	Field	Descriptions
[15:0]	PARSTn	GPIO Port A pin n Output Reset Bits (n = 0 ~ 15) 0: No effect on the PADOUTn bit 1: Reset the PADOUTn bit

Port A Sink Current Enhanced Selection Register – PASCER

This register specifies the GPIO Port A enhanced sink driving current.

Offset: 0x02C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PASCE15	PASCE14	PASCE13	PASCE12	PASCE11	PASCE10	PASCE9	PASCE8
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PASCE7	PASCE6	PASCE5	PASCE4	PASCE3	PASCE2	PASCE1	PASCE0
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PASCEn	GPIO Port A pin n Sink Current Enhanced Selection Control Bits (n = 0 ~ 15) 0: No enhanced sink current 1: Enhanced sink current

Port B Data Direction Control Register – PBDIRCR

This register is used to control the direction of GPIO Port B pin as input or output.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	Reserved							
Type/Reset								
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	PBDIR							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PBDIR							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PBDIRn	GPIO Port B pin n Direction Control Bits (n = 0 ~ 15) 0: Pin n is in input mode 1: Pin n is in output mode

Port B Input Function Enable Control Register – PBINER

This register is used to enable or disable the GPIO Port B input function.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PBINEN							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PBINEN							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PBINENn	GPIO Port B pin n Input Enable Control Bits (n = 0 ~ 15) 0: Pin n input function is disabled 1: Pin n input function is enabled When the pin n input function is disabled, the input Schmitt trigger will be turned off and the Schmitt trigger output will remain at a zero state.

Port B Pull-Up Selection Register – PBPUR

This register is used to enable or disable the GPIO Port B pull-up function. There are two pull-up functions, weak pull-up and strong pull-up.

Offset: 0x008

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	PBSPU							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PBSPU							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	PBPU							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PBPU							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:16]	PBSPUn	GPIO Port B pin n Strong Pull-Up Selection Control Bits (n = 0 ~ 15) 0: Pin n strong pull-up function is disabled 1: Pin n strong pull-up function is enabled The strong pull-up function provides stronger pull-up capability than weak pull-up function. It can be configured individually as weak pull-up or strong pull-up function by setting the corresponding PBPUn or PBSPUn bit. When the strong pull-up and weak pull-up functions are both enabled can provide a maximum internal pull-up effect in GPIO port. Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.
[15:0]	PBPUn	GPIO Port B pin n Weak Pull-Up Selection Control Bits (n = 0 ~ 15) 0: Pin n weak pull-up function is disabled 1: Pin n weak pull-up function is enabled Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.

Port B Pull-Down Selection Register – PBPDR

This register is used to enable or disable the GPIO Port B pull-down function.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PBPD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PBPD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PBPDn	GPIO Port B pin n Pull-Down Selection Control Bits (n = 0 ~ 15) 0: Pin n pull-down function is disabled 1: Pin n pull-down function is enabled Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.

Port B Open-Drain Selection Register – PBODR

This register is used to enable or disable the GPIO Port B open-drain function.

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PBOD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PBOD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PBODn	GPIO Port B pin n Open-Drain Selection Control Bits (n = 0 ~ 15) 0: Pin n Open-Drain output is disabled (The output type is CMOS output) 1: Pin n Open-Drain output is enabled (The output type is open-drain output) Note: When the open-drain function is enabled, the pin n internal pull-up or pull-down configuration will be invalid.

Port B Drive Current Selection Register – PBDRVR

This register specifies the GPIO Port B output driving current.

Offset: 0x014

Reset value: 0x0000_0000

	31		30		29		28		27		26		25		24	
	PBDV15				PBDV14				PBDV13				PBDV12			
Type/Reset	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	23		22		21		20		19		18		17		16	
	PBDV11				PBDV10				PBDV9				PBDV8			
Type/Reset	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	15		14		13		12		11		10		9		8	
	PBDV7				PBDV6				PBDV5				PBDV4			
Type/Reset	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7		6		5		4		3		2		1		0	
	PBDV3				PBDV2				PBDV1				PBDV0			
Type/Reset	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

Bits	Field	Descriptions
[31:0]	PBDVn[1:0]	GPIO Port B pin n Drive Current Selection Control Bits (n = 0 ~ 15) 00: 4 mA source/sink current 01: 8 mA source/sink current 10: 12 mA source/sink current 11: 16 mA source/sink current

Port B Lock Register – PBLOCKR

This register specifies the GPIO Port B lock configuration.

Offset: 0x018

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	PBLKEY							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PBLKEY							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	PBLOCK							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PBLOCK							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:16]	PBLKEY	GPIO Port B lock Key 0x5FA0: Port B lock function is enabled Others: Port B Lock function is disabled To lock the Port B function, a value of 0x5FA0 should be written into the PBLKEY field in this register. To execute a successful write operation on this lock register, the value written into the PBLKEY field must be 0x5FA0. If the value written into this field is not equal to 0x5FA0, any write operations on the PBLOCKR register will be aborted. The result of a read operation on the PBLKEY field returns the GPIO Port B Lock Status which indicates whether the GPIO Port B is locked or not. If the read value of the PBLKEY field is 0, this indicates that the GPIO Port B Lock function is disabled. Otherwise, it indicates that the GPIO Port B Lock function is enabled as the read value is equal to 1.
[15:0]	PBLOCKn	GPIO Port B pin n Lock Control Bits (n = 0 ~ 15) 0: Port B pin n is not locked 1: Port B pin n is locked The PBLOCKn bits are used to lock the configurations of corresponding GPIO Pins when the correct Lock Key is applied to the PBLKEY field. The locked configurations including PBDIRn, PBINENn, PBPUn, PBSPUn, PBPDn, PBODn and PBDVn setting in the related GPIO registers. Additionally, the GPBCFGHR or GPBCFGLR register which is used to configure the alternative function of the associated GPIO pin will also be locked. Note that the PBLOCKR register can only be written once which means that PBLKEY and PBLOCKn (lock control bit) should be written together and cannot be changed until a system reset or GPIO Port B reset occurs.

Port B Data Input Register – PBDINR

This register specifies the GPIO Port B input data.

Offset: 0x01C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PBDIN							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	PBDIN							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

Bits	Field	Descriptions
[15:0]	PBDINn	GPIO Port B pin n Data Input Bits (n = 0 ~ 15) 0: The input data of corresponding pin is 0 1: The input data of corresponding pin is 1

Port B Output Data Register – PBDOUTR

This register specifies the GPIO Port B output data.

Offset: 0x020

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PBDOUT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PBDOUT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PBDOUTn	GPIO Port B pin n Data Output Bits (n = 0 ~ 15) 0: Data to be output on pin n is 0 1: Data to be output on pin n is 1

Port B Output Set/Reset Control Register – PBSRR

This register is used to set or reset the corresponding bit of the GPIO Port B output data.

Offset: 0x024

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	PBRST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PBRST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	PBSET							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PBSET							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

Bits	Field	Descriptions
[31:16]	PBRSTn	GPIO Port B pin n Output Reset Control Bits (n = 0 ~ 15) 0: No effect on the PBDOUTn bit 1: Reset the PBDOUTn bit Note that when the PBRSTn bit in this register or the PBRSTn bit in the PBRR register is enabled, the reset function on the PBDOUTn bit will take effect.
[15:0]	PBSETn	GPIO Port B pin n Output Set Control Bits (n = 0 ~ 15) 0: No effect on the PBDOUTn bit 1: Set the PBDOUTn bit Note that the function enabled by the PBSETn bit has the higher priority if both the PBSETn and PBRSTn bits are set at the same time.

Port B Output Reset Register – PBRR

This register is used to reset the corresponding bit of the GPIO Port B output data.

Offset: 0x028

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PBRST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
	PBRST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

Bits	Field	Descriptions
[15:0]	PBRSTn	GPIO Port B pin n Output Reset Bits (n = 0 ~ 15) 0: No effect on the PBDOUTn bit 1: Reset the PBDOUTn bit

Port B Sink Current Enhanced Selection Register – PBSCER

This register specifies the GPIO Port B enhanced sink driving current.

Offset: 0x02C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PBSCE15	PBSCE14	PBSCE13	PBSCE12	PBSCE11	PBSCE10	PBSCE9	PBSCE8
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PBSCE7	PBSCE6	PBSCE5	PBSCE4	PBSCE3	PBSCE2	PBSCE1	PBSCE0
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PBSCEn	GPIO Port B pin n Sink Current Enhanced Selection Control Bits (n = 0 ~ 15) 0: No enhanced sink current 1: Enhanced sink current

Port C Data Direction Control Register – PCDIRCR

This register is used to control the direction of GPIO Port C pin as input or output.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCDIR							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	PCDIRn	GPIO Port C pin n Direction Control Bits (n = 0 ~ 7) 0: Pin n is in input mode 1: Pin n is in output mode

Port C Input Function Enable Control Register – PCINER

This register is used to enable or disable the GPIO Port C input function.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCINEN							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	PCINENn	GPIO Port C pin n Input Enable Control Bits (n = 0 ~ 7) 0: Pin n input function is disabled 1: Pin n input function is enabled When the pin n input function is disabled, the input Schmitt trigger will be turned off and the Schmitt trigger output will remain at a zero state.

Port C Pull-Up Selection Register – PCPUR

This register is used to enable or disable the GPIO Port C pull-up function. There are two pull-up functions, weak pull-up and strong pull-up.

Offset: 0x008

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	PCSPU							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCPU							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[23:16]	PCSPUn	GPIO Port C pin n Strong Pull-Up Selection Control Bits (n = 0 ~ 7) 0: Pin n strong pull-up function is disabled 1: Pin n strong pull-up function is enabled The strong pull-up function provides stronger pull-up capability than weak pull-up function. It can be configured individually as weak pull-up or strong pull-up function by setting the corresponding PCPUn or PCSPUn bit. When the strong pull-up and weak pull-up functions are both enabled can provide a maximum internal pull-up effect in GPIO port. Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.
[7:0]	PCPUn	GPIO Port C pin n Weak Pull-Up Selection Control Bits (n = 0 ~ 7) 0: Pin n weak pull-up function is disabled 1: Pin n weak pull-up function is enabled Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.

Port C Pull-Down Selection Register – PCPDR

This register is used to enable or disable the GPIO Port C pull-down function.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCPD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	PCPDn	GPIO Port C pin n Pull-Down Selection Control Bits (n = 0 ~ 7) 0: Pin n pull-down function is disabled 1: Pin n pull-down function is enabled Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.

Port C Open-Drain Selection Register – PCODR

This register is used to enable or disable the GPIO Port C open-drain function.

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCOD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	PCODn	GPIO Port C pin n Open-Drain Selection Control Bits (n = 0 ~ 7 0: Pin n Open-Drain output is disabled (The output type is CMOS output) 1: Pin n Open-Drain output is enabled (The output type is open-drain output) Note: When the open-drain function is enabled, the pin n internal pull-up or pull-down configuration will be invalid.

Port C Drive Current Selection Register – PCDRVR

This register specifies the GPIO Port C output driving current.

Offset: 0x014

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PCDV7		PCDV6		PCDV5		PCDV4	
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PCDV3		PCDV2		PCDV1		PCDV0	
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PCDVn[1:0]	GPIO Port C pin n Drive Current Selection Control Bits (n = 0 ~ 7) 00: 4 mA source/sink current 01: 8 mA source/sink current 10: 12 mA source/sink current 11: 16 mA source/sink current

Port C Lock Register – PCLOCKR

This register specifies the GPIO Port C lock configuration.

Offset: 0x018

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	PCLKEY							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PCLKEY							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	Reserved							
Type/Reset								
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	PCLOCK							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:16]	PCLKEY	GPIO Port C lock Key 0x5FA0: Port C Lock function is enable Others: Port C Lock function is disable To lock the Port C function, a value of 0x5FA0 should be written into the PCLKEY field in this register. To execute a successful write operation on this lock register, the value written into the PCLKEY field must be 0x5FA0. If the value written into this field is not equal to 0x5FA0, any write operations on the PCLOCKR register will be aborted. The result of a read operation on the PCLKEY field returns the GPIO Port C Lock Status which indicates whether the GPIO Port C is locked or not. If the read value of the PCLKEY field is 0, this indicates that the GPIO Port C Lock function is disabled. Otherwise, it indicates that the GPIO Port C Lock function is enabled as the read value is equal to 1.
[7:0]	PCLOCKn	GPIO Port C pin n Lock Control Bits (n = 0 ~ 7) 0: Port C pin n is not locked 1: Port C pin n is locked The PCLOCKn bits are used to lock the configurations of corresponding GPIO Pins when the correct Lock Key is applied to the PCLKEY field. The locked configurations including PCDIRn, PCINENn, PCPUn, PCSPUn, PCPDn, PCODn and PCDVn setting in the related GPIO registers. Additionally, the GPCCFGHR or GPCCFGLR register which is used to configure the alternative function of the associated GPIO pin will also be locked. Note that the PCLOCKR register can only be written once which means that PCLKEY and PCLOCKn (lock control bit) should be written together and cannot be changed until a system reset or GPIO Port C reset occurs.

Port C Data Input Register – PCDINR

This register specifies the GPIO Port C input data.

Offset: 0x01C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCDIN							
Type/Reset	RO	RO	RO	RO	RO	RO	RO	RO
	0	0	0	0	0	0	0	0

Bits	Field	Descriptions
[7:0]	PCDINn	GPIO Port C pin n Data Input Bits (n = 0 ~ 7) 0: The input data of corresponding pin is 0 1: The input data of corresponding pin is 1

Port C Output Data Register – PCDOUTR

This register specifies the GPIO Port C output data.

Offset: 0x020

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCDOUT							
Type/Reset	RW	RW	RW	RW	RW	RW	RW	RW
	0	0	0	0	0	0	0	0

Bits	Field	Descriptions
[7:0]	PCDOUTn	GPIO Port C pin n Data Output Bits (n = 0 ~ 7) 0: Data to be output on pin n is 0 1: Data to be output on pin n is 1

Port C Output Set/Reset Control Register – PCSRR

This register is used to set or reset the corresponding bit of the GPIO Port C output data.

Offset: 0x024

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	PCRST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCSET							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

Bits	Field	Descriptions
[23:16]	PCRSTn	GPIO Port C pin n Output Reset Control Bits (n = 0 ~ 7) 0: No effect on the PCDOUTn bit 1: Reset the PCDOUTn bit Note that when the PCRSTn bit in this register or the PCRSTn bit in the PCRR register is enabled, the reset function on the PCDOUTn bit will take effect.
[7:0]	PCSETn	GPIO Port C pin n Output Set Control Bits (n = 0 ~ 7) 0: No effect on the PCDOUTn bit 1: Set the PCDOUTn bit Note that the function enabled by the PCSETn bit has the higher priority if both the PCSETn and PCRSTn bits are set at the same time.

Port C Output Reset Register – PCRR

This register is used to reset the corresponding bit of the GPIO Port C output data.

Offset: 0x028

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCRST							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

Bits	Field	Descriptions
[7:0]	PCRSTn	GPIO Port C pin n Output Reset Control Bits (n = 0 ~ 7) 0: No effect on the PCDOUTn bit 1: Reset the PCDOUTn bit

Port C Sink Current Enhanced Selection Register – PCSCER

This register specifies the GPIO Port C enhanced sink driving current.

Offset: 0x02C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PCSCE7	PCSCE6	PCSCE5	PCSCE4	PCSCE3	PCSCE2	PCSCE1	PCSCE0
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	PCSCEn	GPIO Port C pin n Sink Current Enhanced Selection Control Bits (n = 0 ~ 7) 0: No enhanced sink current 1: Enhanced sink current

Port F Data Direction Control Register – PFDIRCR

This register is used to control the direction of GPIO Port F pin as input or output.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFDIR	
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[1:0]	PFDIRn	GPIO Port F pin n Direction Control Bits (n = 0 ~ 1) 0: Pin n is in input mode 1: Pin n is in output mode

Port F Input Function Enable Control Register – PFINER

This register is used to enable or disable the GPIO Port F input function.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFINEN	
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[1:0]	PFINENn	GPIO Port F pin n Input Enable Control Bits (n = 0 ~ 1) 0: Pin n input function is disabled 1: Pin n input function is enabled When the pin n input function is disabled, the input Schmitt trigger will be turned off and the Schmitt trigger output will remain at a zero state.

Port F Pull-Up Selection Register – PFPUR

This register is used to enable or disable the GPIO Port F pull-up function. There are two pull-up functions, weak pull-up and strong pull-up.

Offset: 0x008

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved						PFSPU	
Type/Reset							RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFPU	
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[17:16]	PFSPUn	GPIO Port F pin n Strong Pull-Up Selection Control Bits (n = 0 ~ 1) 0: Pin n strong pull-up function is disabled 1: Pin n strong pull-up function is enabled The strong pull-up function provides stronger pull-up capability than pull-up function. It can be configured individually as weak pull-up or strong pull-up function by setting the corresponding PFPUn or PFSPUn bit. When the strong pull-up and weak pull-up functions are both enabled can provide a maximum internal pull-up effect in GPIO port. Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.
[1:0]	PFPUn	GPIO Port F pin n Weak Pull-Up Selection Control Bits (n = 0 ~ 1) 0: Pin n weak pull-up function is disabled 1: Pin n weak pull-up function is enabled Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.

Port F Pull-Down Selection Register – PFPDR

This register is used to enable or disable the GPIO Port F pull-down function.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFPD	
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[1:0]	PFPDn	GPIO Port F pin n Pull-Down Selection Control Bits (n = 0 ~ 1) 0: Pin n pull-down function is disabled 1: Pin n pull-down function is enabled Note: When the pull-up and pull-down functions are both enabled, the pull-up function will have the higher priority and therefore the pull-down function will be blocked and disabled.

Port F Open-Drain Selection Register – PFODR

This register is used to enable or disable the GPIO Port F open-drain function.

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFOD	
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[1:0]	PFODn	GPIO Port F pin n Open-Drain Selection Control Bits (n = 0 ~ 1) 0: Pin n Open-Drain output is disabled. (The output type is CMOS output) 1: Pin n Open-Drain output is enabled. (The output type is open-drain output) Note: When the open-drain function is enabled, the pin n internal pull-up or pull-down configuration will be invalid.

Port F Drive Current Selection Register – PFDRVR

This register specifies the GPIO Port F output driving current.

Offset: 0x014

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved				PFDV1		PFDV0	
Type/Reset					RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[3:0]	PFDVn[1:0]	GPIO Port F pin n Drive Current Selection Control Bits (n = 0 ~ 1) 00: 4 mA source/sink current 01: 8 mA source/sink current 10: 12 mA source/sink current 11: 16 mA source/sink current

Port F Lock Register – PFLOCKR

This register specifies the GPIO Port F lock configuration.

Offset: 0x018

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	PFLKEY							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	PFLKEY							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	Reserved							
Type/Reset								
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	Reserved						PFLOCK	
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[31:16]	PFLKEY	GPIO Port F Lock Key 0x5FA0: Port F Lock function is enabled Others: Port F Lock function is disabled To lock the Port F function, a value of 0x5FA0 should be written into the PFLKEY field in this register. To execute a successful write operation on this lock register, the value written into the PFLKEY field must be 0x5FA0. If the value written into this field is not equal to 0x5FA0, any write operations on the PFLOCKR register will be aborted. The result of a read operation on the PFLKEY field returns the GPIO Port F Lock Status which indicates whether the GPIO Port F is locked or not. If the read value of the PFLKEY field is 0, this indicates that the GPIO Port F Lock function is disabled. Otherwise, it indicates that the GPIO Port F Lock function is enabled as the read value is equal to 1.
[1:0]	PFLOCKn	GPIO Port F pin n Lock Control Bits (n = 0 ~ 1) 0: Port F pin n is not locked 1: Port F pin n is locked The PFLOCKn bits are used to lock the configurations of corresponding GPIO Pins when the correct Lock Key is applied to the PFLKEY field. The locked configurations including PFDIRn, PFINENn, PFPUn, PFSPUn, PFPDn, PFODn and PFDVn setting in the related GPIO registers. Additionally, the GPFCFGHR or GPFCFGLR register which is used to configure the alternative function of the associated GPIO pin will also be locked. Note that the PFLOCKR register can only be written once which means that PFLKEY and PFLOCKn (lock control bit) should be written together and can not be changed until a system reset or GPIO Port F reset occurs.

Port F Data Input Register – PFDINR

This register specifies the GPIO Port F input data.

Offset: 0x01C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFDIN	
Type/Reset							RO 0	RO 0

Bits	Field	Descriptions
[1:0]	PFDINn	GPIO Port F pin n Data Input Bits (n = 0 ~ 1) 0: The input data of corresponding pin is 0 1: The input data of corresponding pin is 1

Port F Output Data Register – PFDOUTR

This register specifies the GPIO Port F output data.

Offset: 0x020

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFDOUT	
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[1:0]	PFDOUTn	GPIO Port F pin n Data Output Bits (n = 0 ~ 1) 0: Data to be output on pin n is 0 1: Data to be output on pin n is 1

Port F Sink Current Enhanced Selection Register – PFSCER

This register specifies the GPIO Port F enhanced sink driving current.

Offset: 0x02C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFSCE1	PFSCE0
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[1:0]	PFSCEn	GPIO Port F pin n Sink Current Enhanced Selection Control Bits (n = 0 ~ 1) 0: No enhanced sink current 1: Enhanced sink current

Port F Output Set/Reset Control Register – PFSRR

This register is used to set or reset the corresponding bit of the GPIO Port F output data.

Offset: 0x024

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved						PFRST	
Type/Reset							WO 0	WO 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFSET	
Type/Reset							WO 0	WO 0

Bits	Field	Descriptions
[17:16]	PFRSTn	GPIO Port F pin n Output Reset Control Bits (n = 0 ~ 1) 0: No effect on the PFDOUTn bit 1: Reset the PFDOUTn bit Note that when the PFRSTn bit in this register or the PFRSTn bit in the PFSRR register is enabled, the reset function on the PFDOUTn bit will take effect.
[1:0]	PFSETn	GPIO Port F pin n Output Set Control Bits (n = 0 ~ 1) 0: No effect on the PFDOUTn bit 1: Set the PFDOUTn bit Note that the function enabled by the PFSETn bit has the higher priority if both the PFSETn and PFRSTn bits are set at the same time.

Port F Output Reset Register – PFRR

This register is used to reset the corresponding bit of the GPIO Port F output data.

Offset: 0x028

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						PFRST	
Type/Reset							WO 0	WO 0

Bits	Field	Descriptions
[1:0]	PFRSTn	GPIO Port F pin n Output Reset Bits (n = 0 ~ 1) 0: No effect on the PFDOUTn bit 1: Reset the PFDOUTn bit

9 Alternate Function Input / Output Control Unit (AFIO)

Introduction

In order to expand the flexibility of the GPIO or the usage of peripheral functions, each I/O pin can be configured to have up to sixteen different functions such as GPIO or IP functions by setting the GPxCFGLR or GPxCFGHR register where x is the different port name. According to the usage of the IP resource and application requirements, suitable pin-out locations can be selected by using the peripheral I/O remapping mechanism. Additionally, various GPIO pins can be selected to be the EXTI interrupt line by setting the EXTInPIN [3:0] field in the ESSR register to trigger an interrupt or event. Please refer to the EXTI section for more details.

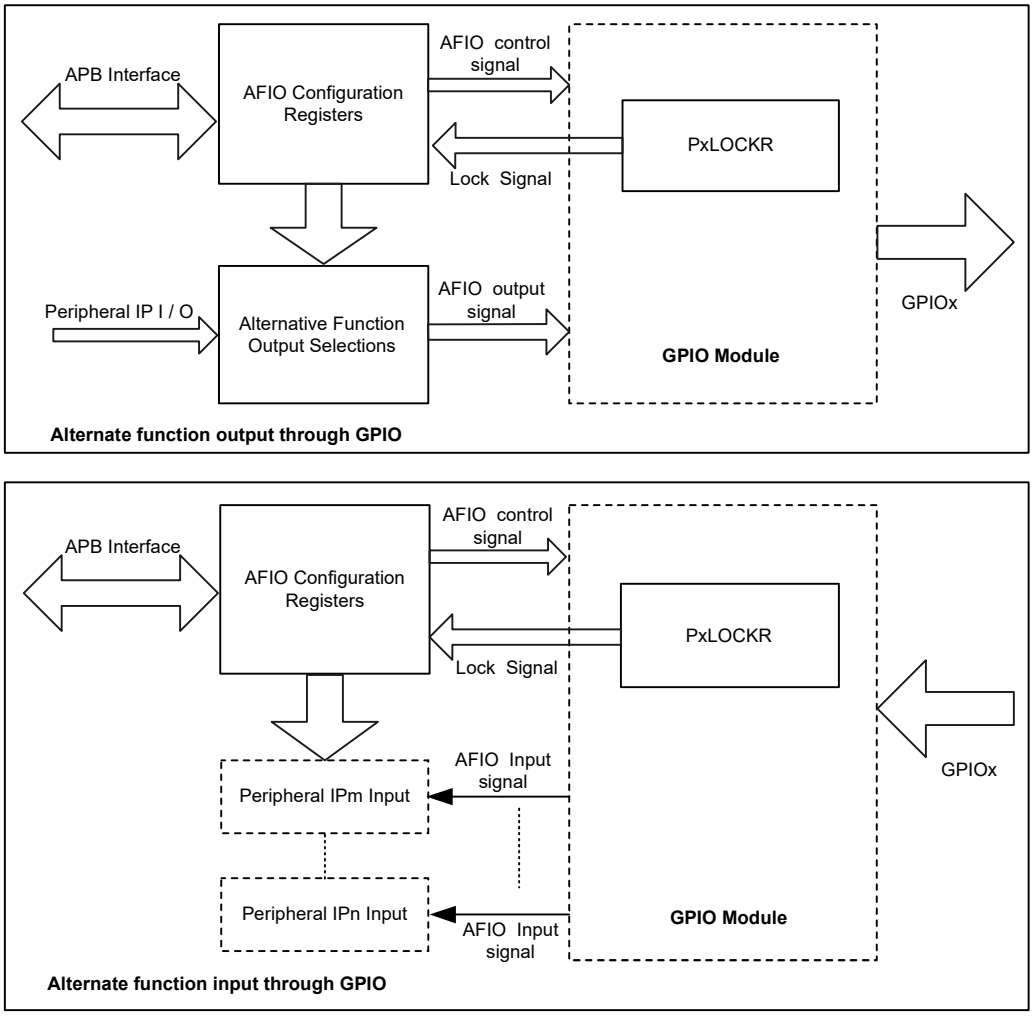


Figure 20. AFIO Block Diagram

Features

- APB slave interface for register access
- EXTI source selection
- Configurable pin function for each GPIO, up to 8 alternative functions on each pin
- AFIO lock mechanism

Functional Descriptions

External Interrupt Pin Selection

The GPIO pins are connected to the 8 EXTI lines as shown in the accompanying figure. For example, the user can set the EXTI0PIN [3:0] field in the ESSR register to b0000 to select the GPIO PA0 pin as EXTI line 0 input. Since not all the pins of the Port A ~ C, F are available in all package types, refer to the pin assignment section for detailed pin information. The setting of the EXTInPIN [3:0] field is invalid when the corresponding pin is not available.

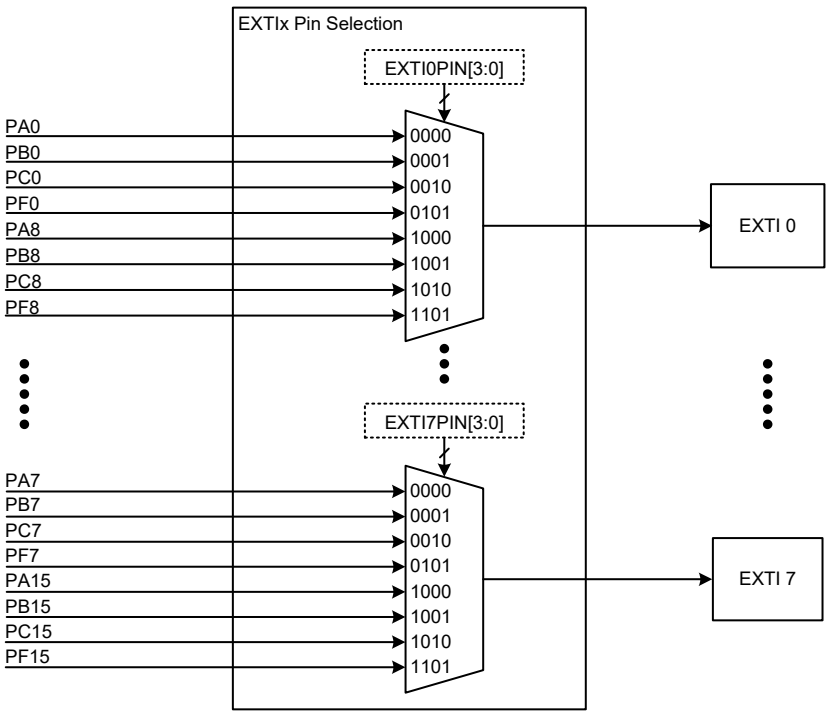


Figure 21. EXTI Channel Input Selection

Alternate Function

Up to eight alternative functions can be chosen for each I/O pad by setting the PxCFGn [2:0] field in the GPxCFGLR or GPxCFGHR (n = 0 ~ 15, x = A ~ C, F) registers. If the pin is selected as unavailable item which is noted as "N/A" in the "Alternate Function Mapping" table of the device datasheet, this pin will be defined as default alternate function. Refer to the "Alternate Function Mapping" table in the device datasheet for detailed mapping of the alternate function I/O pins. In addition to this flexible I/O multiplexing architecture, each peripheral has alternate functions mapped onto different I/O pins to optimize the number of peripherals available in smaller packages. The following description shows the setting of the PxCFGn [2:0] field.

- PxCFGn [2:0] = 000: The default alternated function (after reset, AF0)
- PxCFGn [2:0] = 001: Alternate Function 1 (AF1)
- PxCFGn [2:0] = 010: Alternate Function 2 (AF2)
- …….
- PxCFGn [2:0] = 110: Alternate Function 6 (AF6)
- PxCFGn [2:0] = 111: Alternate Function 7 (AF7)

Table 19. AFIO Selection for Peripheral Map Example

AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
System Default	GPIO	ADC	LEDC	SCTM	SPI	UART	I^2C

Lock Mechanism

The device also offers a lock function to lock the AFIO configuration using the GPIO lock register, PxLOCKR (x = A ~ C, F), until a reset event occurs. Refer to the GPIO Locking Mechanism section in the GPIO chapter for more details.

Register Map

The following table shows the AFIO registers and reset values.

Table 20. AFIO Register Map

Register	Offset	Description	Reset Value
ESSR	0x000	EXTI Source Selection Register	0x0000_0000
GPACFGLR	0x020	GPIO Port A Configuration Low Register	0x0000_0000
GPACFGHR	0x024	GPIO Port A Configuration High Register	0x0000_0000
GPBCFGLR	0x028	GPIO Port B Configuration Low Register	0x0000_0000
GPBCFGHR	0x02C	GPIO Port B Configuration High Register	0x0000_0000
GPCCFGLR	0x030	GPIO Port C Configuration Low Register	0x0000_0000
GPCCFGHR	0x034	GPIO Port C Configuration High Register	0x0000_0000
GPFCFGLR	0x048	GPIO Port F Configuration Low Register	0x0000_0000
GPFCFGHR	0x04C	GPIO Port F Configuration High Register	0x0000_0000

Register Descriptions

EXTI Source Selection Register – ESSR

This register specifies the I/O selection of EXTI0 ~ EXTI7.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	EXTI7PIN				EXTI6PIN			
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	EXTI5PIN				EXTI4PIN			
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	EXTI3PIN				EXTI2PIN			
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	EXTI1PIN				EXTI0PIN			
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31:0]	EXTInPIN[3:0]	EXTIn Pin Selection (n = 0 ~ 7) 0000: PA Bit n is selected as EXTIn source signal 0001: PB Bit n is selected as EXTIn source signal 0010: PC Bit n is selected as EXTIn source signal 0101: PF Bit n is selected as EXTIn source signal 1000: PA Bit (n + 8) is selected as EXTIn source signal 1001: PB Bit (n + 8) is selected as EXTIn source signal 1010: PC Bit (n + 8) is selected as EXTIn source signal 1101: PF Bit (n + 8) is selected as EXTIn source signal Others: Reserved Note: Since not all GPIO pins are available in all products and package types, refer to the pin assignment section for detailed pin information. The EXTInPIN [3:0] field setting is invalid when the corresponding pin is not available.

GPIO x Configuration Low Register – GPxCFGLR, x = A, B, C, F

This low register specifies the alternate function of GPIO Port x, x = A, B, C, F.

Offset: 0x020, 0x028, 0x030, 0x048

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved	PxCFG7			Reserved	PxCFG6		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	Reserved	PxCFG5			Reserved	PxCFG4		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved	PxCFG3			Reserved	PxCFG2		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	Reserved	PxCFG1			Reserved	PxCFG0		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0

Bits	Field	Descriptions
[4n+2:4n]	PxCFGn[2:0]	Alternate function selection for port x pin n (n = 0 ~ 7) 000: Port x pin n is selected as AF0 001: Port x pin n is selected as AF1 : : 110: Port x pin n is selected as AF6 111: Port x pin n is selected as AF7 If the pin is selected as unavailable item which is noted as “N/A” in the “Alternate Function Mapping” table of the device datasheet, this pin will be defined as default alternate function. Refer to the “Alternate Function Mapping” table in the device datasheet for detailed mapping of the alternate function I/O pins.

GPIO x Configuration High Register – GPxCFGHR, x = A, B, C, F

This high register specifies the alternate function of GPIO Port x, x = A, B, C, F.

Offset: 0x024, 0x02C, 0x034, 0x04C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved	PxCFG15			Reserved	PxCFG14		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	Reserved	PxCFG13			Reserved	PxCFG12		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved	PxCFG11			Reserved	PxCFG10		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	Reserved	PxCFG9			Reserved	PxCFG8		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0

Bits	Field	Descriptions
[4n-30:4n-32]	PxCFGn[2:0]	Alternate function selection for port x pin n (n = 8 ~ 15) 000: Port x pin n is selected as AF0 001: Port x pin n is selected as AF1 : : 110: Port x pin n is selected as AF6 111: Port x pin n is selected as AF7 If the pin is selected as unavailable item which is noted as “N/A” in the “Alternate Function Mapping” table of the device datasheet, this pin will be defined as default alternate function. Refer to the “Alternate Function Mapping” table in the device datasheet for detailed mapping of the alternate function I/O pins.

10 Nested Vectored Interrupt Controller (NVIC)

Introduction

In order to reduce the latency and increase the interrupt processing efficiency, a tightly coupled integrated section, which is named as Nested Vectored Interrupt Controller (NVIC) is provided by the Cortex®-M0+. The NVIC controls the system exceptions and the peripheral interrupts which include functions such as the enable/disable control, priority, clear-pending, active status report, software trigger and vector table remapping. Refer to the Technical Reference Manual of Cortex®-M0+ for more details.

Additionally, an integrated simple, 24-bit down-count timer (SysTick) is provided by the Cortex®-M0+ to be used as a tick timer for the Real-Time Operating System (RTOS) or as a simple counter. The SysTick counts down from the reloaded value and generates a system interrupt when it reaches zero. The accompanying table lists the system exceptions types and a variety of peripheral interrupts.

Table 21. Exception Types

Interrupt Number	Exception Number	Exception Type	Priority	Vector Address	Description
—	0	—	—	0x000	Initial Stack Point value
—	1	Reset	-3 (Highest)	0x004	Reset
-14	2	NMI	-2	0x008	Non-Maskable Interrupt. The clock stuck interrupt signal (clock monitor function provided by Clock Control Unit) is connected to the NMI input
-13	3	Hard Fault	-1	0x00C	All fault classes
—	4-10	Reserved	—	—	—
-5	11	SVCall	Configurable(1)	0x02C	SVC instruction System service call
—	12-13	Reserved	—	—	—
-2	14	PendSV	Configurable(1)	0x038	System Service Pendable request
-1	15	SysTick	Configurable(1)	0x03C	SysTick timer decreased to zero
0	16	LVD & BOD	Configurable(2)	0x040	Low voltage & Brown Out detection interrupt
1	17	RTC	Configurable(2)	0x044	RTC global interrupt
2	18	FMC	Configurable(2)	0x048	FMC global interrupt
3	19	WKUP	Configurable(2)	0x04C	EXTI event wakeup or external WAKEUPn pin interrupt (3)
4	20	EXTI0 ~ 1	Configurable(2)	0x050	EXTI Line 0 & 1 interrupt
5	21	EXTI2 ~ 3	Configurable(2)	0x054	EXTI Line 2 & 3 interrupt
6	22	EXTI4 ~ 7	Configurable(2)	0x058	EXTI Line 4 ~ 7 interrupt
7	23	Reserved	—	0x05C	—
8	24	ADC	Configurable(2)	0x060	ADC global interrupt
9	25	Reserved	—	0x064	—
10	26	Reserved	—	0x068	—

Interrupt Number	Exception Number	Exception Type	Priority	Vector Address	Description
11	27	Reserved	—	0x06C	—
12	28	Reserved	—	0x070	—
13	29	SCTM0	Configurable(2)	0x074	SCTM0 global interrupt
14	30	SCTM1	Configurable(2)	0x078	SCTM1 global interrupt
15	31	SCTM2	Configurable(2)	0x07C	SCTM2 global interrupt
16	32	Reserved	—	0x080	—
17	33	BFTM	Configurable(2)	0x084	BFTM global interrupt
18	34	Reserved	—	0x088	—
19	35	I²C	Configurable(2)	0x08C	I²C global interrupt
20	36	Reserved	—	0x090	—
21	37	SPI	Configurable(2)	0x094	SPI global interrupt
22	38	Reserved	—	0x098	—
23	39	Reserved	—	0x09C	—
24	40	Reserved	—	0x0A0	—
25	41	UART0	Configurable(2)	0x0A4	UART0 global interrupt
26	42	UART1	Configurable(2)	0x0A8	UART1 global interrupt
27	43	Reserved	—	0x0AC	—
28	44	Reserved	—	0x0B0	—
29	45	LEDC	Configurable(2)	0x0B4	LED Controller global interrupt
30	46	Reserved	—	0x0B8	—
31	47	Reserved	—	0x0BC	—

Notes: 1. The exception priority can be changed using the NVIC System Handler Priority Registers. For more information, refer to the Arm "Cortex®-M0+ Devices Generic User Guide" document.
2. The interrupt priority can be changed using the NVIC Interrupt Priority Registers. For more information, refer to the Arm "Cortex®-M0+ Devices Generic User Guide" document.
3. Refer to the PWRCU chapter for the relevant configuration descriptions about the WAKEUPn pin wakeup interrupt.

Features

- 7 system Cortex®-M0+ exceptions
- Up to 32 Maskable peripheral interrupts
- 4 programmable priority levels (2 bits for interrupt priority setting)
- Non-Maskable interrupt
- Low-latency exception and interrupt handling
- Vector table remapping capability
- Integrated simple, 24-bit system timer, SysTick
  - 24-bit down-counter
  - Auto-reloading capability
  - Maskable system interrupt generation when counter decreases to 0
  - SysTick clock source derived from the HCLK clock divided by 8

Functional Descriptions

SysTick Calibration

The SysTick Calibration Value Register (SYST_CALIB) is provided by the NVIC to give a reference time base of 1 ms for the RTOS tick timer or other purposes. The TENMS field in the SYST_CALIB register has a fixed value of 2000 which is the Counter-Reload value to indicate 1 ms when the clock source comes from the SysTick reference input clock STCLK with a frequency of 2 MHz (16 MHz divide by 8).

Register Map

The following table shows the NVIC registers and reset values.

Table 22. NVIC Register Map

Register	Offset	Description	Reset Value
NVIC Base Address = 0xE000_E000			
SYST_CSR	0x010	SysTick Control and Status Register	0x0000_0000
SYST_RVR	0x014	SysTick Reload Value Register	Unpredictable
SYST_CVR	0x018	SysTick Current Value Register	Unpredictable
SYST_CALIB	0x01C	SysTick Calibration Value Register	0x4000_07D0
NVIC_ISER	0x100	Interrupt Set Enable Register	0x0000_0000
NVIC_ICER	0x180	Interrupt Clear Enable Register	0x0000_0000
NVIC_ISPR	0x200	Interrupt Set Pending Register	0x0000_0000
NVIC_ICPR	0x280	Interrupt Clear Pending Register	0x0000_0000
NVIC_IPR0	0x400	Interrupt 0 ~ 3 Priority Register	0x0000_0000
NVIC_IPR1	0x404	Interrupt 4 ~ 7 Priority Register	0x0000_0000
NVIC_IPR2	0x408	Interrupt 8 ~ 11 Priority Register	0x0000_0000
NVIC_IPR3	0x40C	Interrupt 12 ~ 15 Priority Register	0x0000_0000
NVIC_IPR4	0x410	Interrupt 16 ~ 19 Priority Register	0x0000_0000
NVIC_IPR5	0x414	Interrupt 20 ~ 23 Priority Register	0x0000_0000
NVIC_IPR6	0x418	Interrupt 24 ~ 27 Priority Register	0x0000_0000
NVIC_IPR7	0x41C	Interrupt 28 ~ 31 Priority Register	0x0000_0000
CPUID	0xD00	CPUID register	0x410C_C601
ICSR	0xD04	Interrupt Control and State Register	0x0000_0000
VTOR	0xD08	Vector Table Offset Register	0x0000_0000
AIRCR	0xD0C	Application Interrupt and Reset Control Register	0xFA05_0000
SCR	0xD10	System Control Register	0x0000_0000
CCR	0xD14	Configuration and Control Register	0x0000_0208
SHPR2	0xD1C	System Handlers Priority Register 2	0x0000_0000
SHPR3	0xD20	System Handlers Priority Register 3	0x0000_0000

Note: For more detailed descriptions of the above registers, refer to the "Cortex®-M0+ Devices Generic User Guide" document from Arm.

11 External Interrupt / Event Controller (EXTI)

Introduction

The External Interrupt/Event Controller, EXTI, comprises 8 edge detectors which can generate wakeup events or interrupt requests independently. In the interrupt mode there are five trigger types which can be selected as the external interrupt trigger type, low level, high level, negative edge, positive edge and both edges, selectable using the SRCnTYPE field in the EXTICFGRn (n = 0 ~ 7) register. In the wakeup event mode, the wakeup event polarity can be configured by setting the EXTInWPOL (n = 0 ~ 7) field in the EXTIWAKUPPOLR register. If the EVWUPIEN bit in the EXTIWAKUPCR Register is set, the WKUP interrupt can be generated when the associated wakeup event occurs and the corresponding EXTI wakeup enable bit is set. Each EXTI line can also be masked independently.

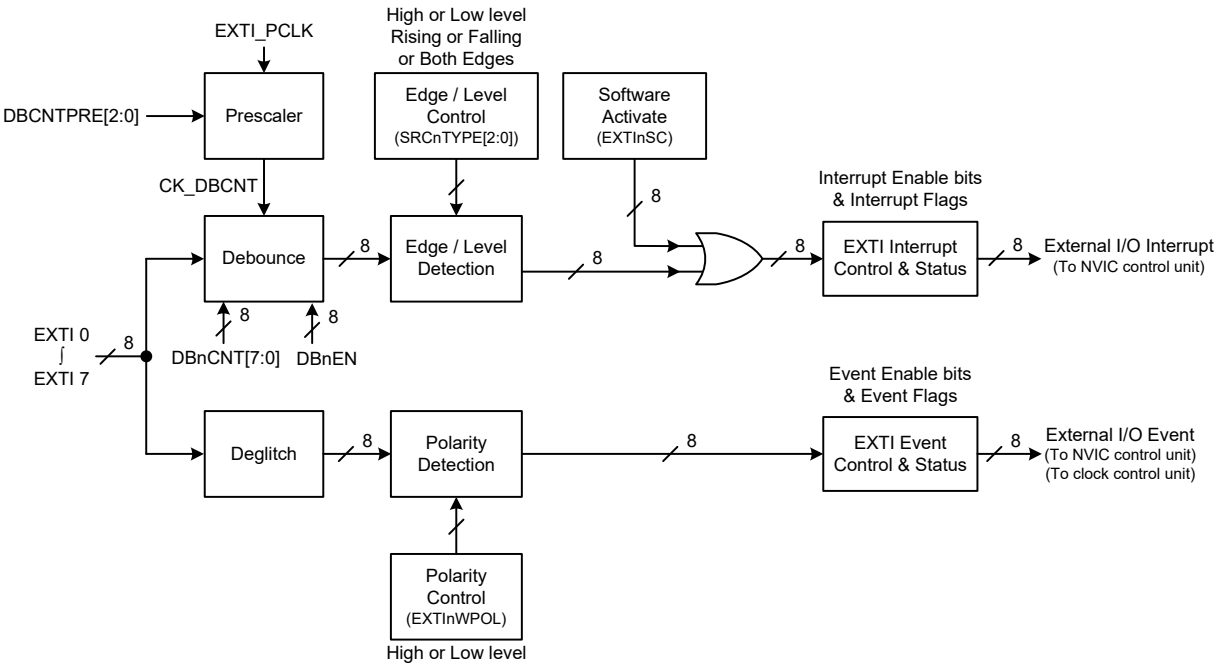


Figure 22. EXTI Block Diagram

Features

- Up to 8 EXTI lines with configurable trigger source and type
  - All GPIO pins can be selected as EXTI trigger source
  - Source trigger type includes high level, low level, negative edge, positive edge or both edges
- Individual interrupt enable, wakeup enable and status bits for each EXTI line
- Software interrupt trigger mode for each EXTI line
- Integrated deglitch filter for short pulse blocking

Functional Descriptions

Wakeup Event Management

In order to wake up the system from the power saving mode, the EXTI controller provides a function which can monitor external events and send them to the MCU core and the Clock Control Unit, CKCU. These external events include EXTI events, Low Voltage Detection, WAKEUPn input pin, Comparator and RTC wakeup functions. By configuring the wakeup event enable bit in the corresponding peripheral, the wakeup signal will be sent to the MCU core and the CKCU via the EXTI controller when the corresponding wakeup event occurs. Additionally, the software can enable the event wakeup interrupt function by setting the EVWUPIEN bit in the EXTIWAKUPCR register and the EXTI controller will then assert an interrupt when the wakeup event occurs.

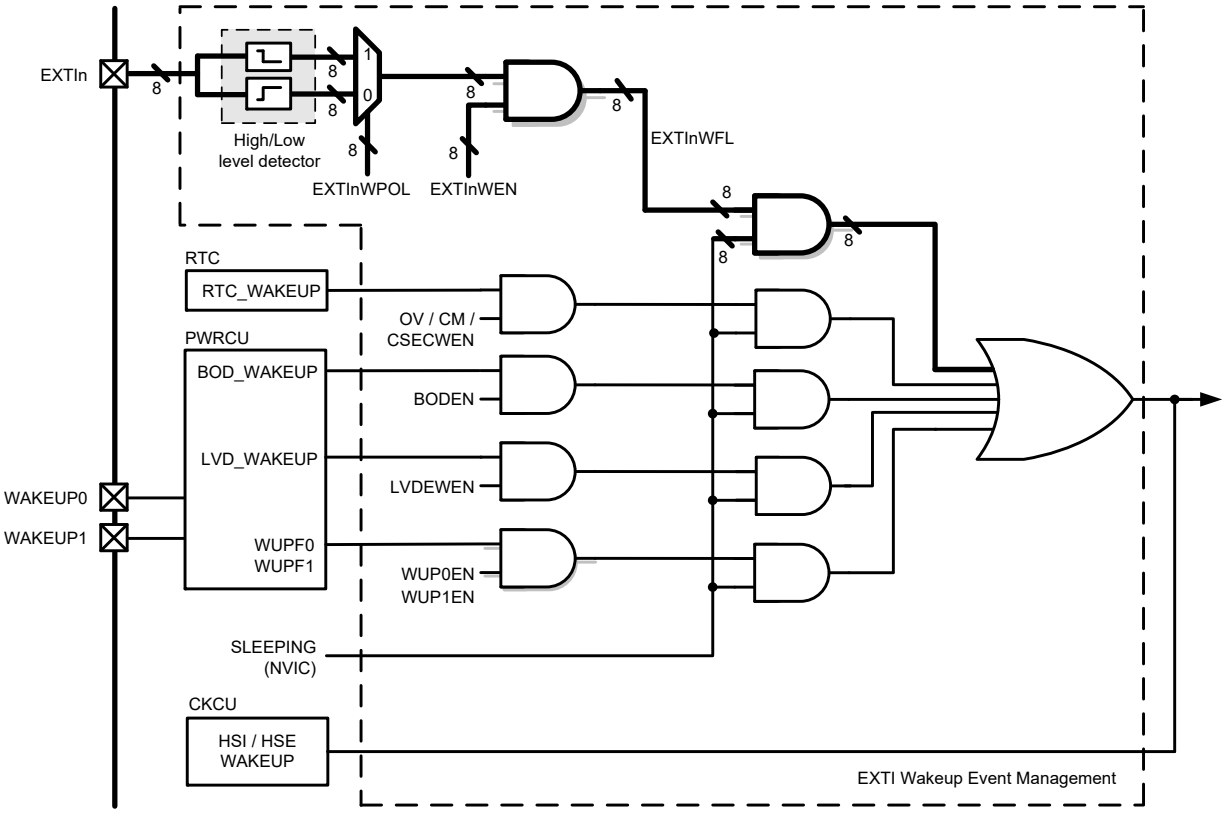


Figure 23. EXTI Wakeup Event Management

External Interrupt/Event Line Mapping

All GPIO pins can be selected as EXTI trigger sources by configuring the EXTInPIN[3:0] field in the AFIO ESSR register to trigger an interrupt or event. Refer to the AFIO section for more details.

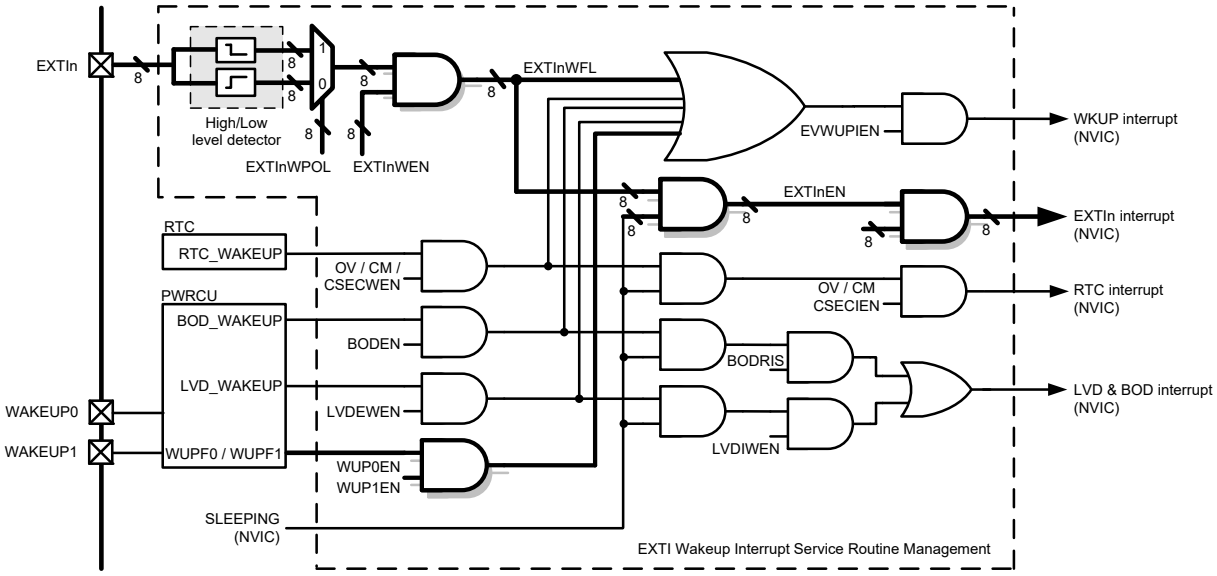


Figure 24. EXTI Wakeup Interrupt Service Routine Management

Interrupt and Debounce

The application software can set the DBnEN bit in the EXTIn Interrupt Configuration Register EXTICFGRn (n = 0 ~ 7) to enable the corresponding pin debounce function and configure the DBnCNT field in the EXTICFGRn register so as to select an appropriate debounce time for specific applications. The interrupt signal will however be delayed due to the debounce function. When the device is woken up from the power saving mode by an external interrupt, an interrupt request will be generated by the EXTI wakeup flag. After the device has been woken up and the clock has recovered, the EXTI wakeup flag that was triggered by the EXTI line must be read and then cleared by application software. The accompanying diagram shows the relationship between the EXTI input signal and the EXTI interrupt/event request signal.

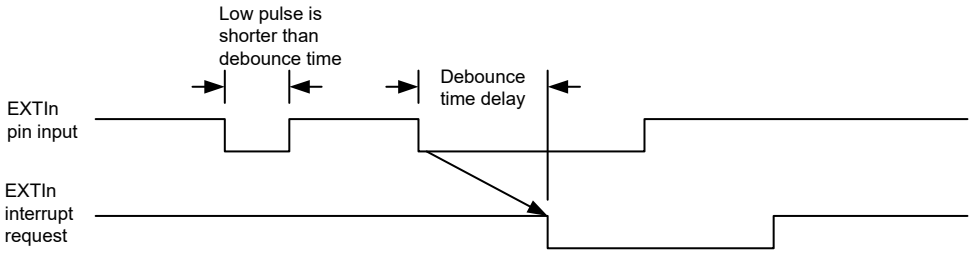


Figure 25. EXTI Interrupt Debounce Function

Register Map

The following table shows the EXTI registers and reset values.

Table 23. EXTI Register Map

Register	Offset	Description	Reset Value
EXTICFGR0	0x000	EXTI Interrupt 0 Configuration Register	0x0000_0000
EXTICFGR1	0x004	EXTI Interrupt 1 Configuration Register	0x0000_0000
EXTICFGR2	0x008	EXTI Interrupt 2 Configuration Register	0x0000_0000
EXTICFGR3	0x00C	EXTI Interrupt 3 Configuration Register	0x0000_0000
EXTICFGR4	0x010	EXTI Interrupt 4 Configuration Register	0x0000_0000
EXTICFGR5	0x014	EXTI Interrupt 5 Configuration Register	0x0000_0000
EXTICFGR6	0x018	EXTI Interrupt 6 Configuration Register	0x0000_0000
EXTICFGR7	0x01C	EXTI Interrupt 7 Configuration Register	0x0000_0000
EXTICR	0x040	EXTI Interrupt Control Register	0x0000_0000
EXTIEDGEFLGR	0x044	EXTI Interrupt Edge Flag Register	0x0000_0000
EXTIEDGESR	0x048	EXTI Interrupt Edge Status Register	0x0000_0000
EXTISSCR	0x04C	EXTI Interrupt Software Set Command Register	0x0000_0000
EXTIWAKUPCR	0x050	EXTI Interrupt Wakeup Control Register	0x0000_0000
EXTIWAKUPPOLR	0x054	EXTI Interrupt Wakeup Polarity Register	0x0000_0000
EXTIWAKUPFLG	0x058	EXTI Interrupt Wakeup Flag Register	0x0000_0000

Register Descriptions

EXTI Interrupt n Configuration Register – EXTICFGRn, n = 0 ~ 7

This register is used to specify the debounce function and select the trigger type.

Offset: 0x000 (0) ~ 0x01C (7)

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	DBnEN	SRCnTYPE			Reserved	DBCNTPRE		
Type/Reset	RW 0	RW 0	RW 0	RW 0		RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	DBnCNT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31]	DBnEN	EXTIn Debounce Circuit Enable Bit (n = 0 ~ 7) 0: Debounce circuit is disabled 1: Debounce circuit is enabled

Bits	Field	Descriptions
[30:28]	SRCnTYPE	EXTIn Interrupt Source Trigger Type (n = 0 ~ 7)

SRCnTYPE [2:0]			Interrupt Source Type
0	0	0	Low-level Sensitive
0	0	1	High-level Sensitive
0	1	0	Negative-edge Triggered
0	1	1	Positive-edge Triggered
1	X	X	Both-edge Triggered

[26:24]	DBCNTPRE	CK_DBCNT Clock Prescaler Selection 000: CK_DBCNT = EXTI_PCLK 001: CK_DBCNT = EXTI_PCLK / 2 010: CK_DBCNT = EXTI_PCLK / 4 ... 111: CK_DBCNT = EXTI_PCLK / 128 The CK_DBCNT clock prescaler is controlled by software setting DBCNTPRE bit field. The control bits is used to select the prescaler of the all EXTI channels de-bounce counter to extend the EXTI de-bounce counter period. This control field is only located at the EXTI Interrupt Configuration Register 0 (EXTICFGR0) and effected on all EXTI channels de-bounce counter.
[7:0]	DBnCNT	EXTIn Debounce Counter (n = 0 ~ 7) The debounce time is calculated with DBnCNT × CK_DBCNT clock period and should be long enough to take effect on the input signal.

EXTI Interrupt Control Register – EXTICR

This register is used to control the EXTI interrupt.

Offset: 0x040
Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	EXTI7EN	EXTI6EN	EXTI5EN	EXTI4EN	EXTI3EN	EXTI2EN	EXTI1EN	EXTI0EN
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	EXTInEN	EXTIn Interrupt Enable Bit (n = 0 ~ 7) 0: EXTI line n interrupt is disabled 1: EXTI line n interrupt is enabled

EXTI Interrupt Edge Flag Register – EXTIEDGEFLGR

This register is used to indicate if an EXTI edge has been detected.

Offset: 0x044

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								

	7	6	5	4	3	2	1	0
	EXTI7EDF	EXTI6EDF	EXTI5EDF	EXTI4EDF	EXTI3EDF	EXTI2EDF	EXTI1EDF	EXTI0EDF
Type/Reset	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0

Bits	Field	Descriptions
[7:0]	EXTInEDF	EXTIn Edge Detection Flag (n = 0 ~ 7) 0: No edge is detected 1: Positive or negative edge is detected This bit is set by the hardware circuitry when a positive or negative edge is detected on the corresponding EXTI line. Software should write 1 to clear it.

EXTI Interrupt Edge Status Register – EXTIEDGESR

This register indicates the polarity of a detected EXTI edge.

Offset: 0x048

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	EXTI7EDS	EXTI6EDS	EXTI5EDS	EXTI4EDS	EXTI3EDS	EXTI2EDS	EXTI1EDS	EXTI0EDS
Type/Reset	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0

Bits	Field	Descriptions
[7:0]	EXTInEDS	EXTIn Edge Detection Status (n = 0 ~ 7) 0: Negative edge is detected 1: Positive edge is detected Software should write 1 to clear it.

EXTI Interrupt Software Set Command Register – EXTISSCR

This register is used to activate the EXTI interrupt.

Offset: 0x04C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								

	7	6	5	4	3	2	1	0
	EXTI7SC	EXTI6SC	EXTI5SC	EXTI4SC	EXTI3SC	EXTI2SC	EXTI1SC	EXTI0SC
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	EXTInSC	EXTIn Software Set Command (n = 0 ~ 7) 0: Deactivates the corresponding EXTI interrupt 1: Activates the corresponding EXTI interrupt

EXTI Interrupt Wakeup Control Register – EXTIWAKUPCR

This register is used to control the EXTI interrupt and wakeup function.

Offset: 0x050

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	EVWUPIEN	Reserved						
Type/Reset	RW 0							
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								

	7	6	5	4	3	2	1	0
	EXTI7WEN	EXTI6WEN	EXTI5WEN	EXTI4WEN	EXTI3WEN	EXTI2WEN	EXTI1WEN	EXTI0WEN
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31]	EVWUPIEN	EXTI Event Wakeup Interrupt Enable Bit 0: Disable WKUP interrupt 1: Enable WKUP interrupt

Bits	Field	Descriptions
[7:0]	EXTInWEN	EXTIn Wakeup Enable Bit (n = 0 ~ 7) 0: Power saving mode wakeup is disabled 1: Power saving mode wakeup is enabled

EXTI Interrupt Wakeup Polarity Register – EXTIWAKUPPOLR

This register is used to select the EXTI line interrupt wakeup polarity.

Offset: 0x054

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	EXTI7WPOL	EXTI6WPOL	EXTI5WPOL	EXTI4WPOL	EXTI3WPOL	EXTI2WPOL	EXTI1WPOL	EXTI0WPOL
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	EXTInWPOL	EXTIn Wakeup Polarity (n = 0 ~ 7) 0: EXTIn wakeup is high level active 1: EXTIn wakeup is low level active

EXTI Interrupt Wakeup Flag Register – EXTIWAKUPFLG

This register is the EXTI interrupt wakeup flag register.

Offset: 0x058

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	EXTI7WFL	EXTI6WFL	EXTI5WFL	EXTI4WFL	EXTI3WFL	EXTI2WFL	EXTI1WFL	EXTI0WFL
Type/Reset	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0

Bits	Field	Descriptions
[7:0]	EXTInWFL	EXTIn Wakeup Flag (n = 0 ~ 7) 0: No wakeup occurs 1: System is woken up by EXTIn Software should write 1 to clear it.

12 Analog to Digital Converter (ADC)

Introduction

A 12-bit multi-channel Analog to Digital Converter (ADC) with a Voltage Reference Generator is integrated in the devices. There are a total of 14 multiplexed channels including 12 external channels on which the external analog signal can be supplied and 2 internal channels. There are two conversion modes to convert an analog signal to digital data. The A/D conversion can be operated in one shot conversion and continuous conversion modes. A 16-bit data register is provided to store the data after conversion.

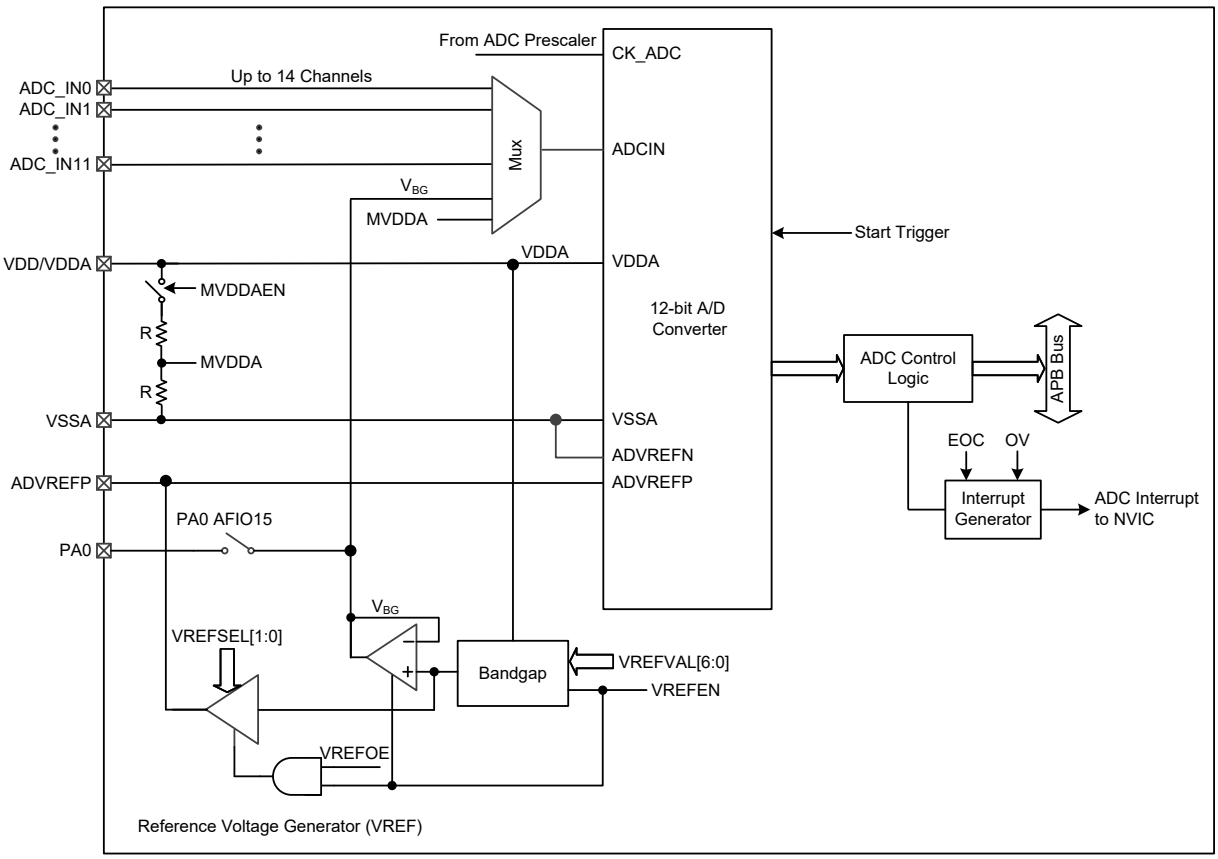


Figure 26. ADC Block Diagram

Features

- 12-bit SAR ADC engine
- Up to 500 ksps conversion rate
- Up to 12 external analog input channels
- 1 channel for internal voltage reference (V_{BG})
- 1 channel for monitor MV_{DDA}

- Programmable sampling time for conversion channel
- Up to 4 programmable conversion channel sequence and dedicated data registers for conversion result
- Two conversion mode
  - One shot conversion mode
  - Continuous conversion mode
- Software trigger start source for conversion modes
- Multiple generated interrupts
  - End of single conversion
  - End of cycle conversion
  - Data register overwriting

Functional Descriptions

ADC Clock Setup

The ADC clock, CK_ADC is provided by the Clock Controller which is synchronous and divided by with the AHB clock known as HCLK. Refer to the Clock Control Unit chapter for more details. Notes that the ADC requires at least two ADC clock cycles to switch between power-on and power-off conditions (ADCEN bit = '0').

Channel Selection

The A/D converter supports 14 multiplexed channels and organizes the conversion results into a specific group. A conversion group can organize a sequence which can be implemented on the channels arranged in a specific conversion sequence length from 1 to 4. For example, conversion can be carried out with the following channel sequence: CH2, CH4, CH7 and CH5 one after another.

A group is composed of up to 4 conversions. The selected channels of the group conversion can be specified in the ADCLST register. The total conversion sequence length is setup using the ADSEQL[1:0] bits in the ADCCR register.

Modifying the ADCCR or ADCLST register during a conversion process will reset the current conversion, after which a new start pulse is required to restart a new conversion.

Conversion Mode

The A/D has two operating conversion modes. The conversion modes are One Shot Conversion Mode and Continuous Conversion Mode. Details are provided later.

One Shot Conversion Mode

In the One Shot Conversion mode, the ADC will perform conversion cycles on the channels specified in the A/D conversion list register ADCLST with a specific sequence when an A/D converter trigger event occurs. When the A/D conversion mode field ADMODE [1:0] in the ADCCR register is set to 0x0, the A/D converter will operate in the One Shot Conversion Mode. This mode can be started by a software trigger.

After Conversion

- The converted data will be stored in the 16-bit ADCDRy (y = 0 ~ 3) registers.
- The ADC single sample end of conversion event raw status flag, ADIRAWS, in the ADCIRAW register will be set when the single sample conversion is finished.

- An interrupt will be generated after a single sample end of conversion if the ADIES bit in the ADCIER register is enabled.
- An interrupt will be generated after a group cycle end of conversion if the ADIEC bit in the ADCIER register is enabled.

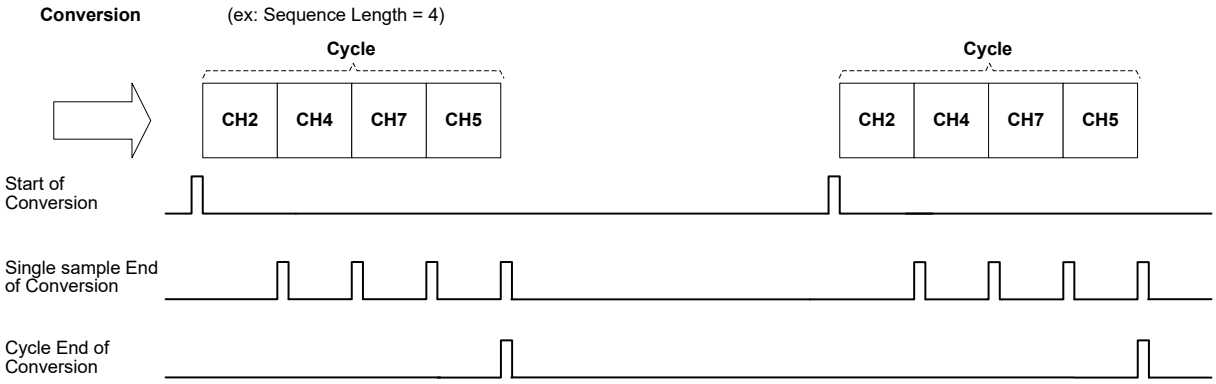


Figure 27. One Shot Conversion Mode

Continuous Conversion Mode

In the Continuous Conversion Mode, repeated conversion cycle will restart automatically without requiring additional A/D start trigger signals after a channel group conversion has completed. When the A/D conversion mode field ADMODE[1:0] is set to 0x2, the A/D converter will operate in the Continuous Conversion Mode which can be started by a software trigger.

After conversion

- The converted data will be stored in the 16-bit ADCDRy (y = 0 ~ 3) registers.
- The ADC group cycle end of conversion event raw status flag, ADIRAWC, in the ADCIRAW register will be set when the conversion cycle is finished.
- An interrupt will be generated after a group cycle end of conversion if the ADIEC bit in the ADCIER register is enabled.

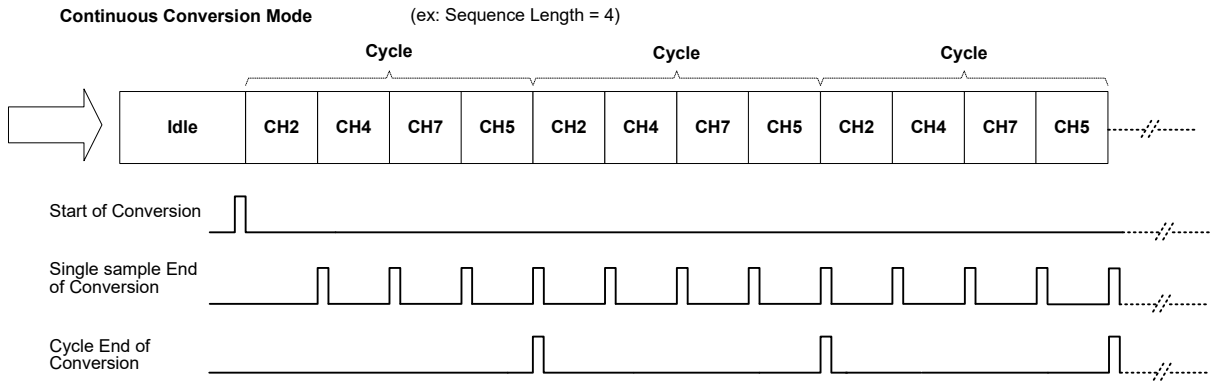


Figure 28. Continuous Conversion Mode

Start Conversion by Software Trigger

An A/D converter conversion can be started by setting the software trigger bit, ADSC, in the ADCTSR register for the group channel when the software trigger enable bit, ADSW, in the ADCTCR register is set to 1. After the A/D converter starts converting the analog data, the corresponding enable bit ADSC will be cleared to 0 automatically.

Sampling Time Setting

The conversion channel sampling time can be programmed according to the input resistance of the input voltage source. This sampling time must be enough for the input voltage source to charge the internal sample and hold capacitor in the A/D converter to the input voltage level. Each conversion channel is sampled with the same sampling time. By modifying the ADST[7:0] bits in the ADCSTR register, the sampling time of the analog input signal can be determined.

The total conversion time (T_{conv}) is calculated using the following formula:

$$T_{conv} = T_{Sampling} + T_{Latency}$$

Where the minimum sampling time $T_{Sampling}$ = 1.5 cycles (when ADST[7:0] = 0) and the minimum channel conversion latency $T_{Latency}$ = 12.5 cycles.

Example:

With the A/D Converter clock CK_ADC = 7 MHz and a sampling time = 1.5 cycles:

$$T_{conv} = 1.5 + 12.5 = 14 \text{ cycles} = 2\ \mu s$$

Data Format

The ADC conversion result can be read in the ADCDRy register and the data format is shown in the following table.

Table 24. Data Format in ADCDR [15:0]

Description	ADCDR register Data Format
Right aligned	"0_0_0_0_d11_d10_d9_d8_d7_d6_d5_d4_d3_d2_d1_d0"

Interrupts

When an A/D conversion is completed, an End of Conversion EOC event will occur. There are two kinds of EOC events which are known as single sample EOC and cycle EOC for A/D conversion. A single sample EOC event will occur and the single sample EOC interrupt raw flag, ADIRAWS bits in the ADCIRAW register, will be asserted when a single channel conversion has completed. A cycle EOC event will occur and the cycle EOC interrupt raw flag, ADIRAWC bits in the ADCIRAW register, will be asserted when a cycle conversion is finished. When a single sample EOC or a cycle EOC raw flag is asserted and the corresponding interrupt enable bit, ADIMC or ADIES bit in the ADCIER register, is set to 1, the associated interrupt will be generated.

After a conversion has completed, the 12-bit digital data will be stored in the associated ADCDRy registers and the value of the data valid flag named as ADVLDy will be changed from low to high. The converted data should be read by the application program, after which the data valid flag ADVLDy will be automatically changed from high to low. Otherwise, a data overwrite event will occur and the data overwrite interrupt raw flag ADIRAWO bit in the ADCIRAW register will be asserted. When the related data overwrite raw flag is asserted, the data overwrite interrupt will be generated if the interrupt enable bit ADIEO in the ADCIER register is set to 1.

The A/D Converter interrupt clear bits are used to clear the associated A/D converter interrupt raw and interrupt status bits. Writing a 1 into the specific A/D converter interrupt clear bit in the A/D converter interrupt clear register ADCICLR will clear the corresponding A/D converter interrupt raw and interrupt status bits. These bits are automatically cleared to 0 by hardware after being set to 1.

Voltage Reference Generator

The internal voltage reference generator provides a stable voltage output to the ADVREFP pin for the ADC reference positive voltage, ADVREFP, when the VREFOE bit is set. The Bandgap Voltage V_{BG} is internally connected to the ADC input channel. The precise voltage of the V_{REF} is individually measured and calculated for each part by manufacture during production test and stored in the Flash Manufacture Privilege Information Block. It can be accessed using the VREFVALR register in the read-only mode. Refer to the datasheet for additional information. When not in use the internal V_{REF} generator can be configured in the power down mode to save power consumption.

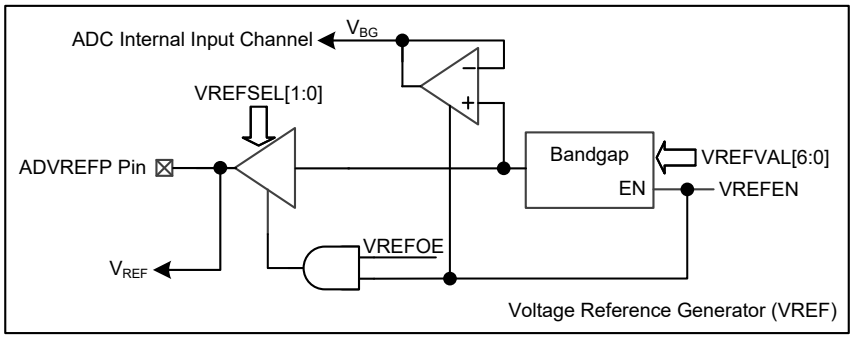


Figure 29. Voltage Reference Generator Block Diagram

V_{DDA} Voltage Monitor

The MVDDAEN bit in the VREFCR register allows the applications to measure the V_{DDA} voltage on the VDDA pin. As the V_{DDA} voltage could be higher than the ADC reference positive voltage, ADVREFP, in order to ensure the correct operation of the ADC, the VDDA pin is internally connected to a bridge divider by 2. This bridge is automatically enabled when the MVDDAEN bit is set, to connect the $V_{DDA}/2$ to the ADC input channel. As a consequence, the converted digital value is half of the V_{DDA} voltage. To prevent any unwanted consumption on the battery, it is recommended to enable the V_{DDA} power divider only when the ADC conversion is required

Register Map

The following table shows the A/D Converter registers and reset values.

Table 25. A/D Converter Register Map

Register	Offset	Description	Reset Value
ADCCR	0x000	ADC Conversion Control Register	0x0000_0000
ADCLST	0x004	ADC Conversion List Register	0x0000_0000
ADCSTR	0x020	ADC Input Sampling Time Register	0x0000_0000
ADCDR0	0x030	ADC Conversion Data Register 0	0x0000_0000
ADCDR1	0x034	ADC Conversion Data Register 1	0x0000_0000
ADCDR2	0x038	ADC Conversion Data Register 2	0x0000_0000
ADCDR3	0x03C	ADC Conversion Data Register 3	0x0000_0000
ADCTSR	0x074	ADC Trigger Source Register	0x0000_0000
ADCIER	0x080	ADC Interrupt Enable register	0x0000_0000
ADCIRAW	0x084	ADC Interrupt Raw Status Register	0x0000_0000
ADCISR	0x088	ADC Interrupt Status Register	0x0000_0000
ADCICLR	0x08C	ADC Interrupt Clear Register	0x0000_0000
VREFCR	0x0A0	Voltage Reference Control Register	0x0000_0000
VREFVALR	0x0A4	Voltage Reference Value Register	0x0000_00XX

Register Descriptions

ADC Conversion Control Register – ADCCR

This register specifies the mode setting and sequence length of the ADC conversion mode. Note that once the content of ADCCR is changed, the conversion in progress will be aborted and the A/D converter will return to an idle state. The application program has to wait for at least one CK_ADC clock before issuing the next command.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved						ADSEQL	
Type/Reset							RW 0	RW 0
	7	6	5	4	3	2	1	0
	ADCEN	ADCRST	Reserved				ADMODE	
Type/Reset	RW 0	RW 0					RW 0	RW 0

Bits	Field	Descriptions
[9:8]	ADSEQL	ADC Conversion Length 00: The channel specified by the ADSEQ0 field in the ADCLST register will be converted Others: Sequence length = ADSEQL[1:0] + 1

Bits	Field	Descriptions
[7]	ADCEN	ADC Enable 0: ADC is disabled 1: ADC is enabled When this bit is cleared to 0, the A/D converter will be disabled and the CK_ADC clock will also be switched off.
[6]	ADCRST	ADC Reset 0: No effect 1: Reset A/D converter except for the A/D Converter controller
[1:0]	ADMODE	ADC Conversion Mode

ADMODE [1:0]	Mode	Descriptions
00	One shot mode	After a start trigger, the conversion will be executed on the specific channels for the whole conversion sequence once.
01	Reserved	—
10	Continuous mode	After a start trigger, the conversion will be executed on the specific channels for the whole sequence continuously until conversion mode is changed.
11	Reserved	—

ADC Conversion List Register – ADCLST

This register specifies the conversion sequence order No.0 ~ No.3 of the ADC.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved			ADSEQ3				
Type/Reset				RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	Reserved			ADSEQ2				
Type/Reset				RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved			ADSEQ1				
Type/Reset				RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	Reserved			ADSEQ0				
Type/Reset				RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[28:24]	ADSEQ3	ADC Conversion Sequence Select 3 Select the ADC input channel for the 3rd ADC conversion sequence. 0x00: ADC_IN0 0x01: ADC_IN1 0x02: ADC_IN2 0x03: ADC_IN3 0x04: ADC_IN4 0x05: ADC_IN5 0x06: ADC_IN6 0x07: ADC_IN7 0x08: ADC_IN8 0x09: ADC_IN9 0x0A: ADC_IN10 0x0B: ADC_IN11 0x0C: Internal Voltage Reference (V_{BG}) 0x0D: Analog power MV_{DDA} (V_{DDA}/2) 0x0E ~ 0x1F: Invalid setting. These values must not be selected as it may cause the ADC abnormal operations.
[20:16]	ADSEQ2	ADC Conversion Sequence Select 2
[12:8]	ADSEQ1	ADC Conversion Sequence Select 1
[4:0]	ADSEQ0	ADC Conversion Sequence Select 0

ADC Input Sampling Time Register – ADCSTR

This register specifies the A/D converter input channel sampling time.

Offset: 0x020

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	ADST							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	ADST	ADC Input Channel Sampling Time Sampling time = (ADST[7:0] + 1.5) × CK_ADC clocks.

ADC Conversion Data Register y – ADCDRy, y = 0 ~ 3

This register is used to store the conversion data of the conversion sequence order No.y which is specified by the ADSEQy field in the ADCLST register.

Offset: 0x030 ~ 0x03C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	ADVLDy	Reserved						
Type/Reset	RC	0						
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	ADDy							
Type/Reset	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0
	7	6	5	4	3	2	1	0
	ADDy							
Type/Reset	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0

Bits	Field	Descriptions
[31]	ADVLDy	ADC Conversion Data of Sequence Order No.y Valid Bit (y = 0 ~ 3) 0: Data are invalid or have been read 1: New data is valid
[15:0]	ADDy	ADC Conversion Data of Sequence Order No.y (y = 0 ~ 3) The conversion result of Sequence Order ADSEQy in the ADCLST register.

ADC Trigger Source Register – ADCTSR

This register contains the software trigger bit of the conversion.

Offset: 0x074

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved							ADSC
Type/Reset								RW 0

Bits	Field	Descriptions
[0]	ADSC	ADC Conversion Software Trigger Bit 0: No operation 1: Start conversion immediately This bit is set by software to start a conversion manually and then cleared by hardware automatically after conversion started.

ADC Interrupt Enable Register – ADCIER

This register contains the ADC interrupt enable bits.

Offset: 0x080

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							ADIEO
Type/Reset								RW 0
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					ADIEC	Reserved	ADIES
Type/Reset						RW 0		RW 0

Bits	Field	Descriptions
[24]	ADIEO	ADC Data Register Overwrite Interrupt enable 0: ADC data register overwrite interrupt is disabled 1: ADC data register overwrite interrupt is enabled
[2]	ADIEC	ADC Cycle EOC Interrupt enable 0: ADC cycle end of conversion interrupt is disabled 1: ADC cycle end of conversion interrupt is enabled
[0]	ADIES	ADC Single EOC Interrupt enable 0: ADC single end of conversion interrupt is disabled 1: ADC single end of conversion interrupt is enabled

ADC Interrupt Raw Status Register – ADCIRAW

This register contains the ADC interrupt raw status bits.

Offset: 0x084

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							ADIRAWO
Type/Reset								RO 0
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					ADIRAWC	Reserved	ADIRAWS
Type/Reset						RO 0		RO 0

Bits	Field	Descriptions
[24]	ADIRAWO	ADC Data Register Overwrite Interrupt Raw Status 0: ADC data register overwrite event does not occur 1: ADC data register overwrite event occurs
[2]	ADIRAWC	ADC Cycle EOC Interrupt Raw Status 0: ADC cycle end of conversion event does not occur 1: ADC cycle end of conversion event occurs
[0]	ADIRAWS	ADC Single EOC Interrupt Raw Status 0: ADC single end of conversion event does not occur 1: ADC single end of conversion event occurs

ADC Interrupt Status Register – ADCISR

This register contains the ADC interrupt masked status bits. The corresponding interrupt status will be set to 1 if the associated interrupt event occurs and the related enable bit is set to 1.

Offset: 0x088

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							ADISRO
Type/Reset								RO 0
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					ADISRC	Reserved	ADISRS
Type/Reset						RO 0		RO 0

Bits	Field	Descriptions
[24]	ADISRO	ADC Data Register Overwrite Interrupt Status 0: ADC data register overwrite interrupt does not occur or the data register overwrite interrupt is disabled 1: ADC data register overwrite interrupt occurs as the data register overwrite interrupt is enabled
[2]	ADISRC	ADC Cycle EOC Interrupt Status 0: ADC cycle end of conversion interrupt does not occur or the cycle end of conversion interrupt is disabled 1: ADC cycle end of conversion interrupt occurs as the cycle end of conversion interrupt is enabled
[0]	ADISRS	ADC Single EOC Interrupt Status 0: ADC single end of conversion interrupt does not occur or the single end of conversion interrupt is disabled 1: ADC single end of conversion interrupt occurs as the single end of conversion interrupt is enabled

ADC Interrupt Clear Register – ADCICLR

This register provides the clear bits used to clear the interrupt raw and masked status of the ADC. These bits are set to 1 by software to clear the interrupt status and automatically cleared to 0 by hardware after being set to 1.

Offset: 0x08C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							ADICLRO
Type/Reset								WO 0
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					ADICLRC	Reserved	ADICLRS
Type/Reset						WO 0		WO 0

Bits	Field	Descriptions
[24]	ADICLRO	ADC Data Register Overwrite Interrupt Status Clear Bit 0: No effect 1: Clear ADISRO and ADIRAWO bits
[2]	ADICLRC	ADC Cycle EOC Interrupt Status Clear Bit 0: No effect 1: Clear ADISRC and ADIRAWC bits
[0]	ADICLRS	ADC Single EOC Interrupt Status Clear Bit 0: No effect 1: Clear ADISRS and ADIRAWS bits

Voltage Reference Control Register – VREFCR

This register contains the internal voltage reference control bits.

Offset: 0x0A0

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							MVDDAEN
Type/Reset								RW 0
	7	6	5	4	3	2	1	0
		Reserved		VREFSEL		Reserved	VREFOE	VREFEN
Type/Reset			RW 0	RW 0			RW 0	RW 0

Bits	Field	Descriptions
[8]	MVDDAEN	Measurement V_{DDA}/2 power Enable 0: Disable 1: Enable measurement V_{DDA}/2 power
[5:4]	VREFSEL	Voltage Reference Output Selection 00: 2.5 V 01: 3.0 V 10: 4.0 V 11: 4.5 V These bits select the Voltage Reference output level.
[1]	VREFOE	Voltage Reference Generator Output Buffer Enable 0: Voltage reference generator output buffer is disabled 1: Voltage reference generator output buffer is enabled The internal voltage reference generator (V_{REF}) provides a stable voltage output to the ADVREFP pin for the ADC reference positive voltage, ADVREFP, when this bit is set.
[0]	VREFEN	Voltage Reference Enable 0: Disable Voltage Reference 1: Enable Voltage Reference

Voltage Reference Value Register – VREFVALR

This register contains the internal voltage reference trim value.

Offset: 0x0A4

Reset value: 0x0000_00XX (Various depending on Flash Manufacture Privilege Information Block)

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved	VREFVAL						
Type/Reset		RO X	RO X	RO X	RO X	RO X	RO X	RO X

Bits	Field	Descriptions
[6:0]	VREFVAL	Voltage Reference Calibration Value During the manufacturing process, the calibration data of the internal voltage reference is stored in the Flash Manufacture Privilege Information Block and downloaded to this field when the system is powered on.

13 Single-Channel Timer (SCTM)

Introduction

The Single-Channel Timer consists of one 16-bit up-counter, a 16-bit Channel 0 Capture / Compare Register, one 16-bit Counter-Reload Register, a 16-bit Channel 1 Capture Register and several control/status registers. It can be used for a variety of purposes including general timer, input signal pulse width measurement or output waveform generation such as single pulse generation or PWM output.

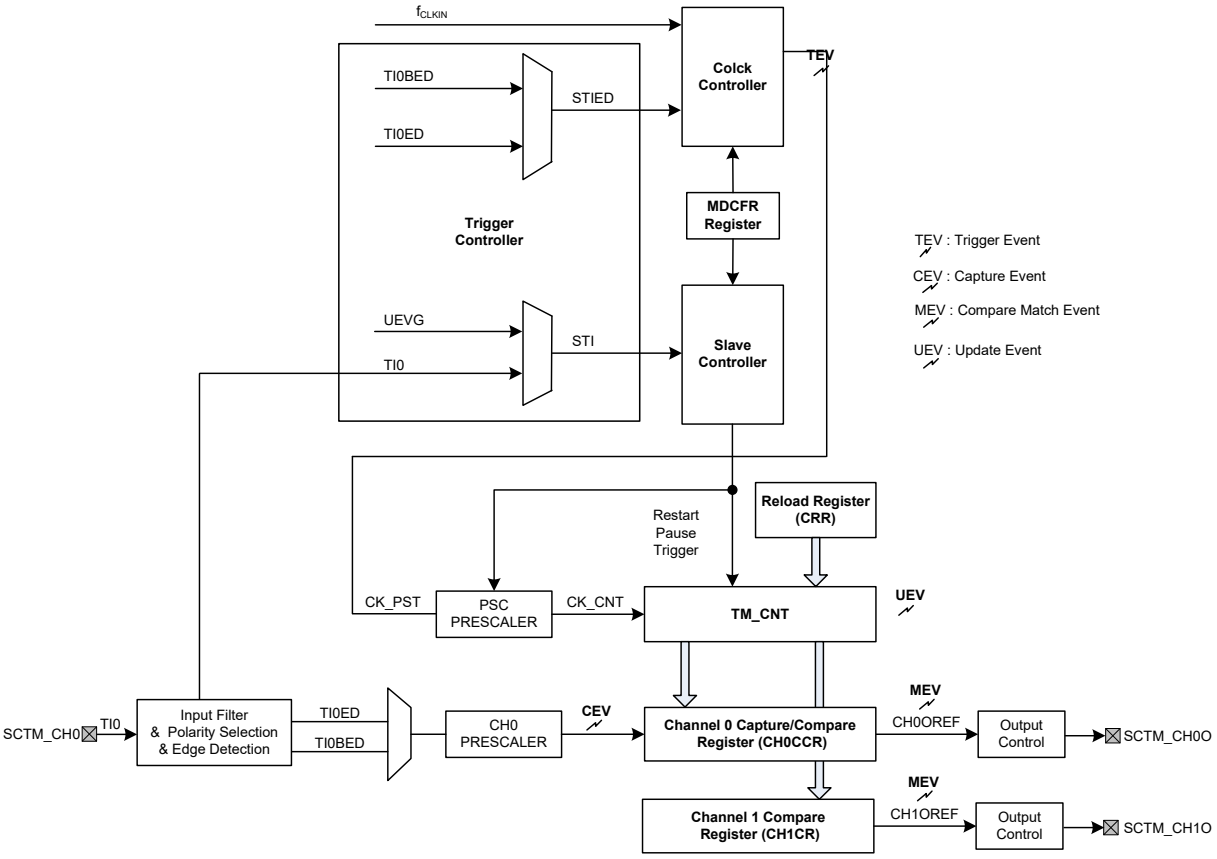


Figure 30. SCTM Block Diagram

Features

- 16-bit auto-reload up counter
- 8-bit programmable prescaler that allows division of the prescaler clock source by any factor between 1 and 256 to generate the counter clock frequency
- Channels for:
  - Input Capture function (for Channel 0 only)
  - Compare Match Output
  - PWM waveform Output
  - Single Pulse Mode Output
- Interrupt generation with the following events:
  - Update event
  - Trigger event
  - Input capture event (for Channel 0 only)
  - Output compare match event

Functional Descriptions

Counter Mode

The counter counts continuously from 0 to the counter-reload value, which is defined in the CRR register, then restarts to count from 0 and generates an overflow event. This action will continue repeatedly. When the update event is generated by setting the UEVG bit in the EVGR register to 1, the counter value will also be initialized to 0.

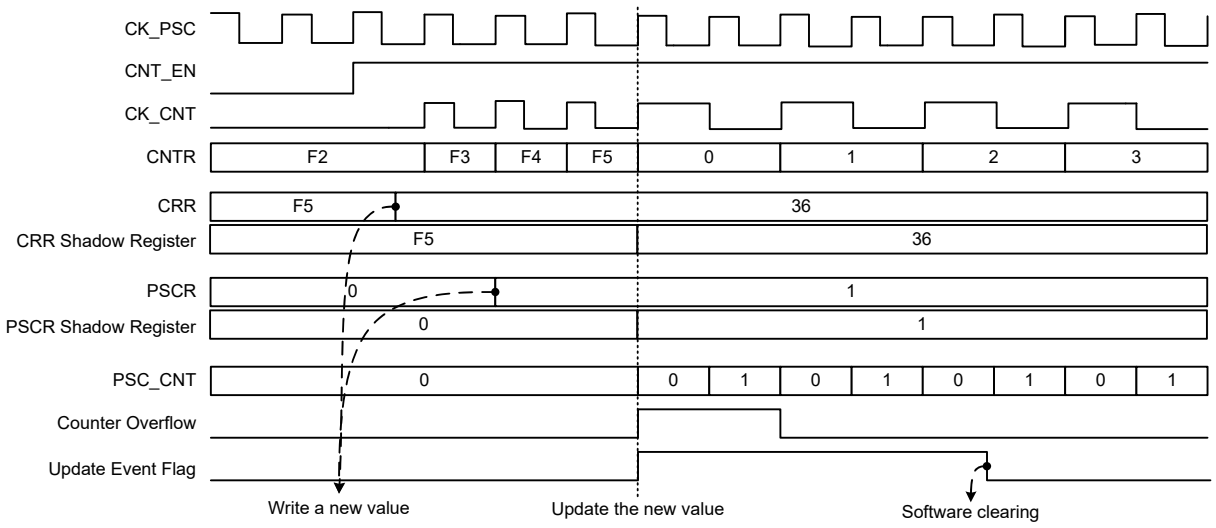


Figure 31. Up-counting Example

Clock Controller

The following describes the Timer Module clock controller which determines the clock source of the internal prescaler counter.

- Internal APB clock f_{CLKIN}

  The default internal clock source is the APB clock f_{CLKIN} used to drive the counter prescaler when the slave mode selection bits SMSEL in the MDCFR register is 0x0 that the slave mode is disabled. When the slave mode selection bits SMSEL are set to 0x4, 0x5 or 0x6, the internal APB clock f_{CLKIN} is the counter prescaler driving clock source. If the slave mode controller is enabled by setting SMSEL field in the MDCFR register to 0x7, the prescaler is clocked by other clock sources selected by the TRSEL field in the TRCFR register and described as follows.

- STIED

  When the STIED mode is selected by setting the SMSEL field to 0x7 in the MDCFR register, the counter prescaler will count on each rising edge of the STI signal. Here the counter will act as an event counter. The input event, known as STI here, can be selected by setting the TRSEL field to an available value except the value of 0x0. When the STI signal is selected as the clock source, the internal edge detection circuitry will generate a clock pulse during each STI signal rising edge to drive the counter prescaler. It is important to note that if the TRSEL field is set to 0x0 to select the software UEVG bit as the trigger source, then when the SMSEL field is set to 0x7, the counter will be updated instead of counting.

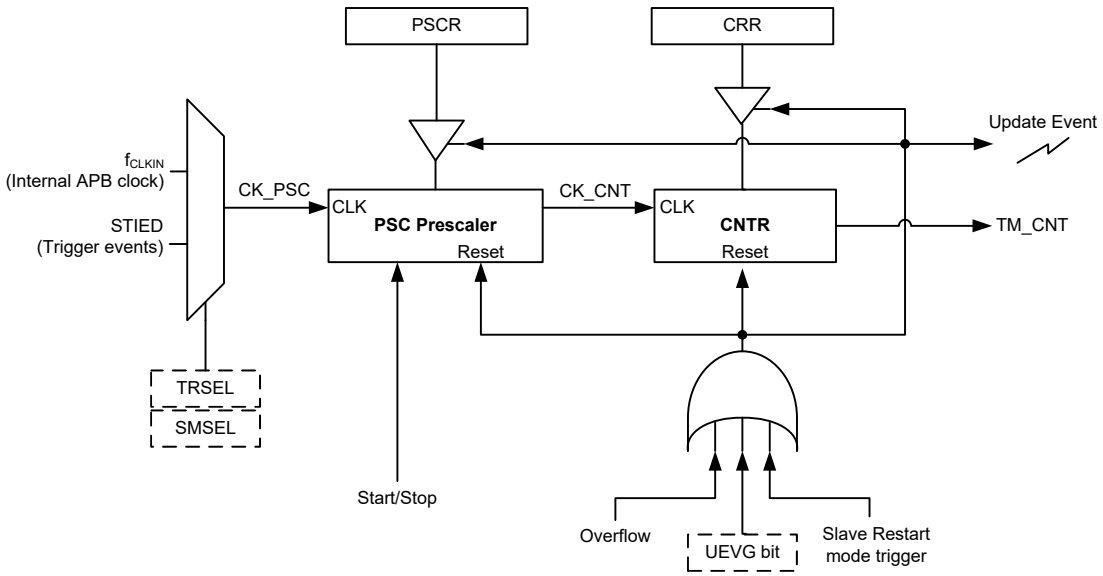


Figure 32. SCTM Clock Source Selection

Trigger Controller

The trigger controller is used to select the trigger source and setup the trigger level or edge trigger condition. For the internal trigger input, it can be selected by the Trigger Selection bits TRSEL in the TRCFR register. For all the trigger sources except the UEVG bit software trigger, the internal edge detection circuitry will generate a clock pulse at each trigger signal rising edge to stimulate some SCTM functions which are triggered by a trigger signal rising edge.

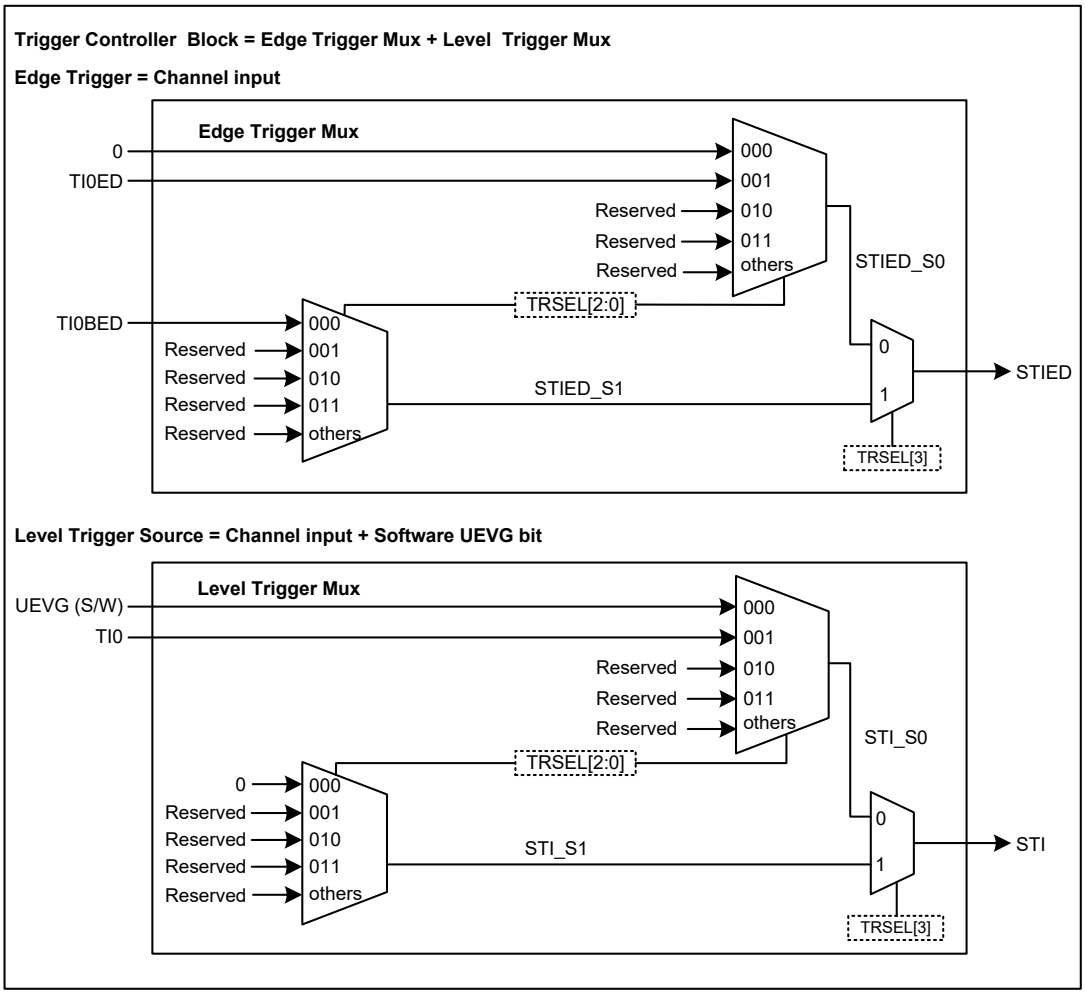


Figure 33. Trigger Controller Block

Slave Controller

The SCTM can be synchronized with an external trigger in several modes including the Restart mode, the Pause mode and the Trigger mode which is selected by the SMSEL field in the MDCFR register. The trigger input of these modes comes from the STI signal which is selected by the TRSEL field in the TRCFR register. The operation modes in the Slave Controller are described in the accompanying sections.

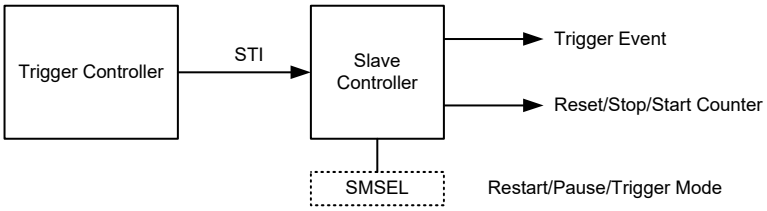


Figure 34. Slave Controller Diagram

Restart Mode

The counter and its prescaler can be reinitialized in response to a rising edge of the STI signal. When an STI rising edge occurs, the update event software generation bit named UEVG will automatically be asserted by hardware and the trigger event flag will also be set. Then the counter and prescaler will be reinitialized. Although the UEVG bit is set to 1 by hardware, the update event does not really occur. It depends upon whether the update event disable control bit UEVDIS in the CNTCFR register is set to 1 or not. If the UEVDIS is set to 1 to disable the update event to occur, there will no update event be generated, however the counter and prescaler are still reinitialized when the STI rising edge occurs. If the UEVDIS bit in the CNTCFR register is cleared to enable the update event to occur, an update event will be generated together with the STI rising edge, then all the preloaded registers will be updated.

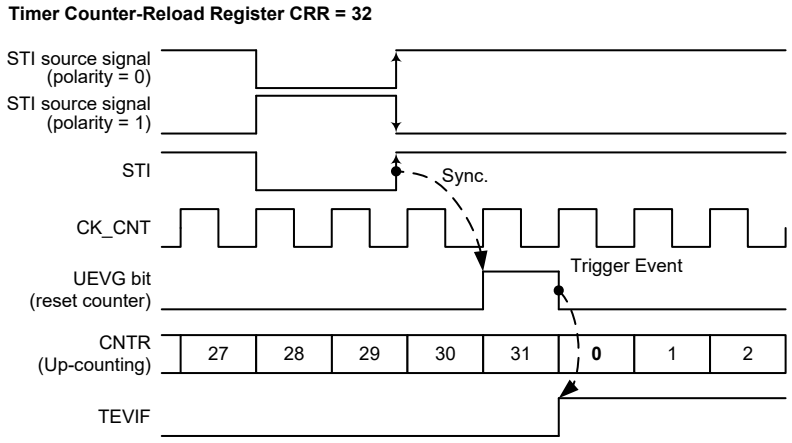


Figure 35. SCTM in Restart Mode

Pause Mode

In the Pause Mode, the selected STI input signal level is used to control the counter start/stop operation. The counter starts to count when the selected STI signal is at a high level and stops counting when the STI signal is changed to a low level, here the counter will maintain its present value and will not be reset. Since the Pause function depends upon the STI level to control the counter stop/start operation, the selected STI trigger signal cannot be derived from the TI0BED signal.

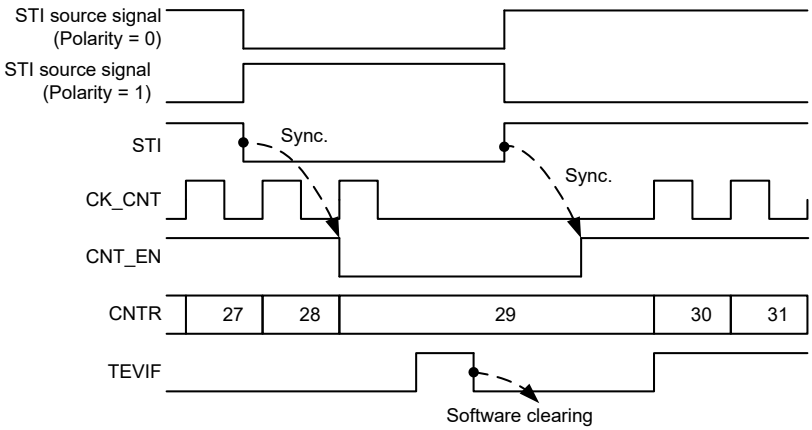


Figure 36. SCTM in Pause Mode

Trigger Mode

After the counter is disabled to count, the counter can resume counting when an STI rising edge signal occurs. When an STI rising edge occurs, the counter will start to count from the current value in the counter. Note that if the STI signal is selected to be derived from the UEVG bit software trigger, the counter will not resume counting. When software triggering using the UEVG bit is selected as the STI source signal, there will be no clock pulse generated which can be used to make the counter resume counting. Note that the STI signal is only used to enable the counter to resume counting and has no effect on controlling the counter to stop counting.

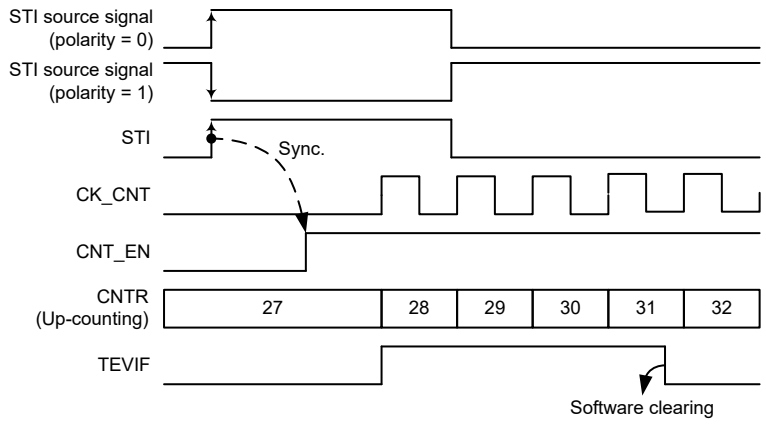


Figure 37. SCTM in Trigger Mode

Channel Controller

The SCTM has channel 0 and channel 1. The channel 0 which is the main channel can be used as capture inputs or compare match output while the channel 1 can be only used as compare match output. The capture input or compare match output channel is composed of a preload register and a shadow register. Data access of the APB bus is always implemented by reading/writing the preload register.

When used in the input capture mode, the counter value is captured into the CH0CCR shadow register first and then transferred into the CH0CCR preload register when the capture event occurs.

When used in the compare match output mode, the contents of the CH0CCR or CH1CR preload register is copied into the associated shadow register, the counter value is then compared with the register value.

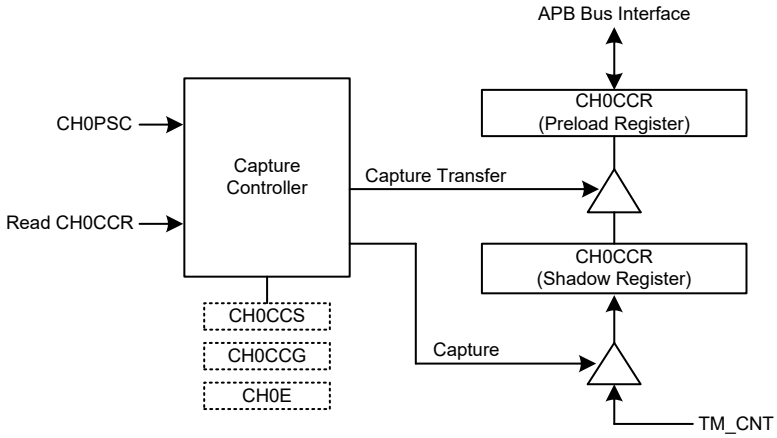


Figure 38. Capture Block Diagram

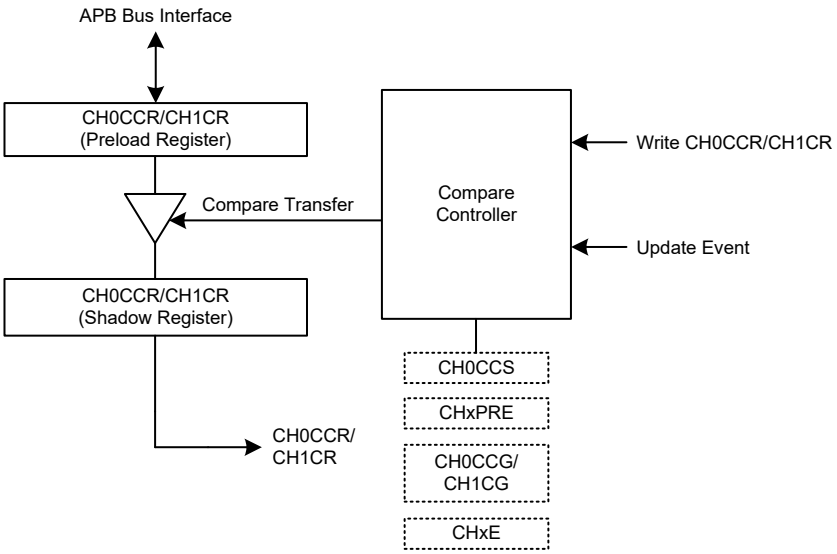


Figure 39. Compare Block Diagram

Capture Counter Value Transferred to CH0CCR

When the CH0 channel is used as a capture input, the counter value is captured into the Channel 0 Capture/Compare Register (CH0CCR) when an effective input signal transition occurs. Once the capture event occurs, the CH0CCIF flag in the INTSR register is set accordingly. If the CH0CCIF bit is already set, i.e., the flag has not yet been cleared by software, and another capture event on this channel occurs, the corresponding channel Over-Capture flag, named CH0OCF, will be set.

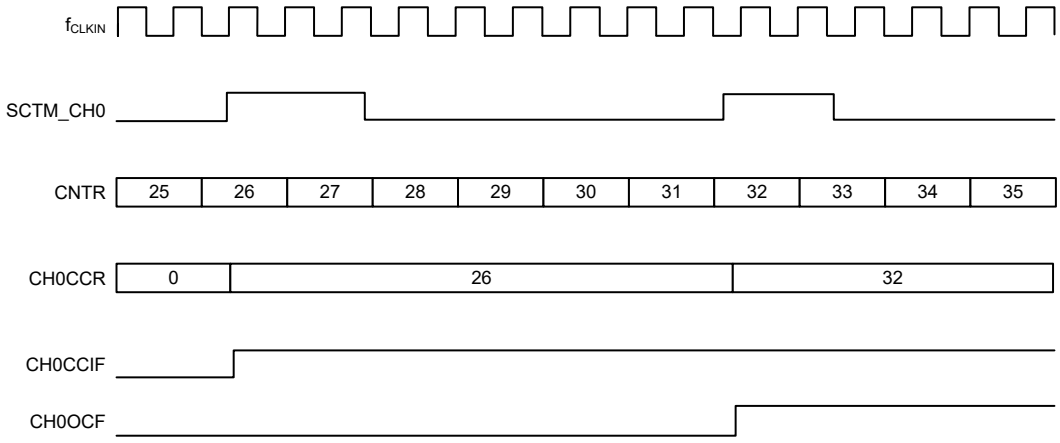


Figure 40. Input Capture Mode on Channel 0

Input Stage

The input stage consists of a digital filter, a channel polarity selection, edge detection and channel prescaler for CH0. The channel input signal (TI0) is sampled by a digital filter to generate a filtered input signal TI0FP. Then the channel polarity and the edge detection block can generate a TI0ED signal for CH0 input capture function. The effective input event number can be set by the channel capture input source prescaler setting the CH0PSC field in the CH0ICFR register.

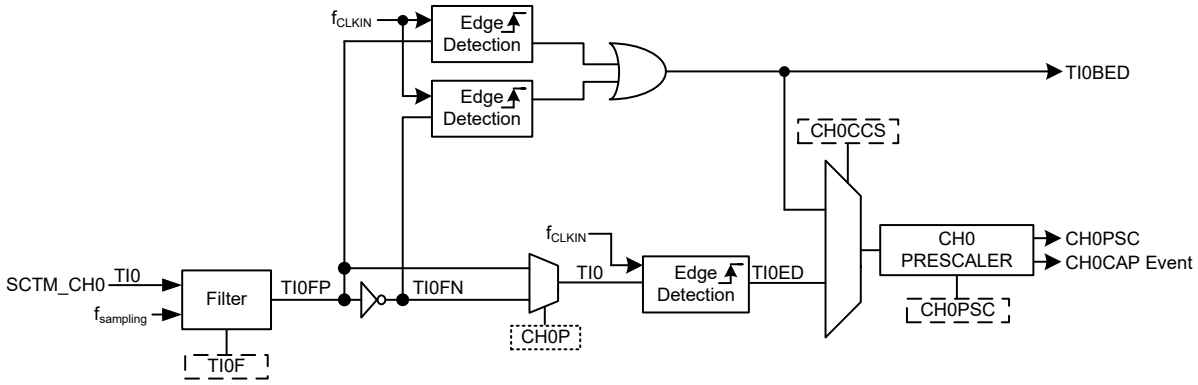


Figure 41. Channel Input Stages

Digital Filter

The digital filter is embedded in the channel input stage. The digital filter in the SCTM is an N-event counter where N refers to how many valid transitions are necessary to output a filtered signal. The N value can be set to 0 or 2 ~ 16 according to the user selection for this digital filter.

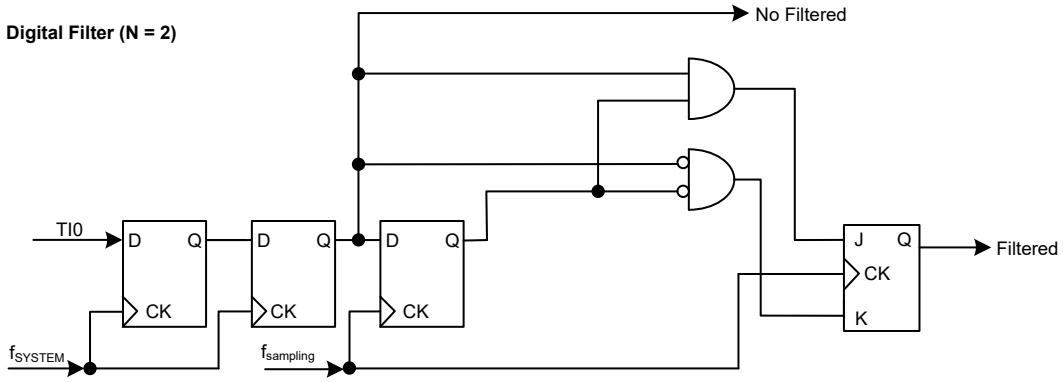


Figure 42. TI0 Digital Filter Diagram with N = 2

Output Stage

The SCTM output has function for compare match, single pulse or PWM output. The channel output SCTM_CHxO is controlled by the CHxOM, CHxP and CHxE bits in the corresponding CHxOCFR, CHPOLR and CHCTR registers.

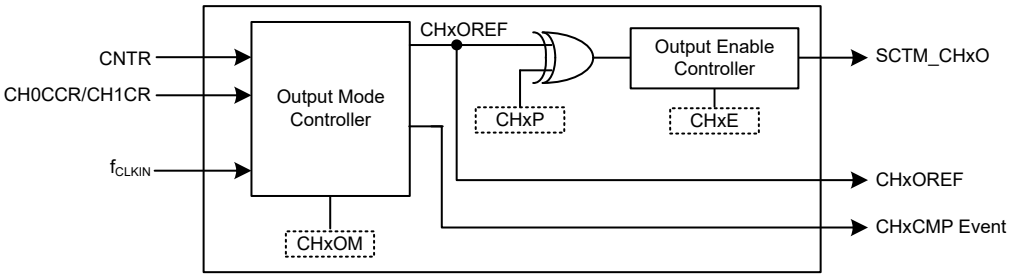


Figure 43. Output Stage Block Diagram

Channel x Output Reference Signal

When the SCTM is used in the compare match output mode, the CHxOREF signal (Channel x Output Reference signal) is defined by the CHxOM bit field setup. The CHxOREF signal has several types of output function which defines what happens to the output when the counter value matches the contents of the CH0CCR or CH1CR register. In addition to the low, high and toggle CHxOREF output types; there are also PWM mode 1 and PWM mode 2 outputs. In these modes, the CHxOREF signal level is changed according to the relationship between the counter value and the CH0CCR or CH1CR content. There are also two modes which will force the output into an inactive or active state irrespective of the CH0CCR or CH1CR content or counter values. With regard to a more detailed description refer to the relative bit field definition. The accompanying table shows a summary of the output type setup.

Table 26. Compare Match Output Setup

CHxOM Value	Compare Match Level
0x0	No change
0x1	Clear Output to 0
0x2	Set Output to 1
0x3	Toggle Output
0x4	Force Inactive Level
0x5	Force Active Level
0x6	PWM Mode 1
0x7	PWM Mode 2

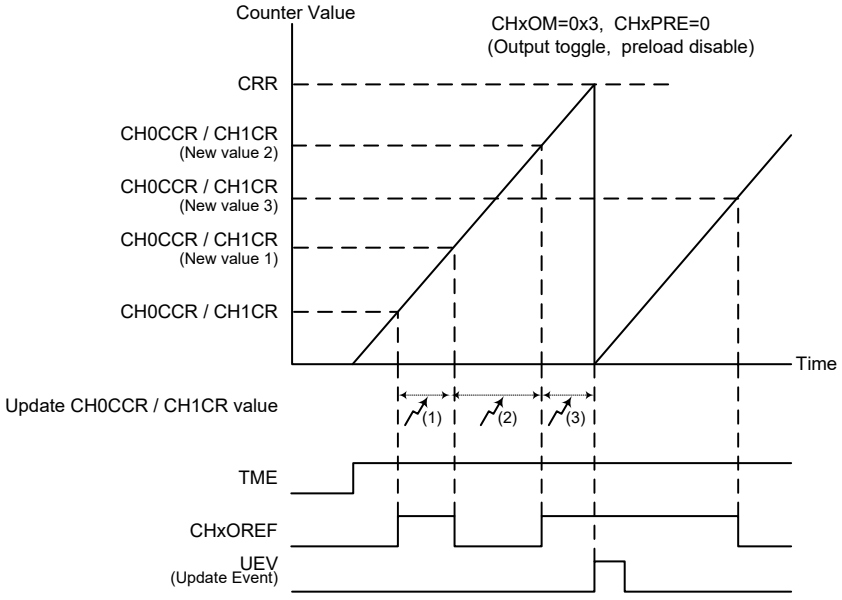


Figure 44. Toggle Mode Channel x Output Reference Signal – CHxPRE = 0

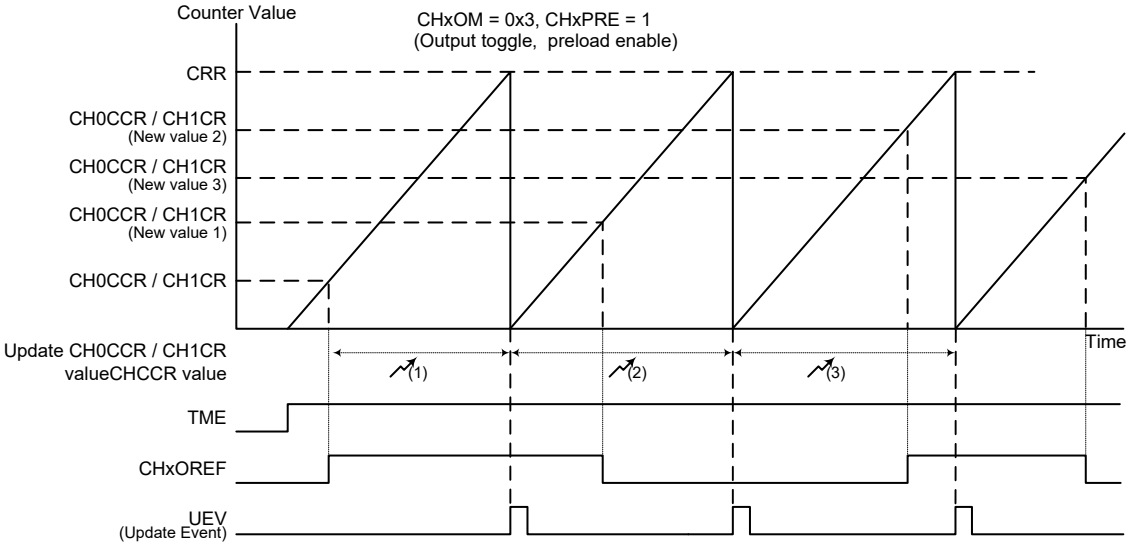


Figure 45. Toggle Mode Channel x Output Reference Signal – CHxPRE = 1

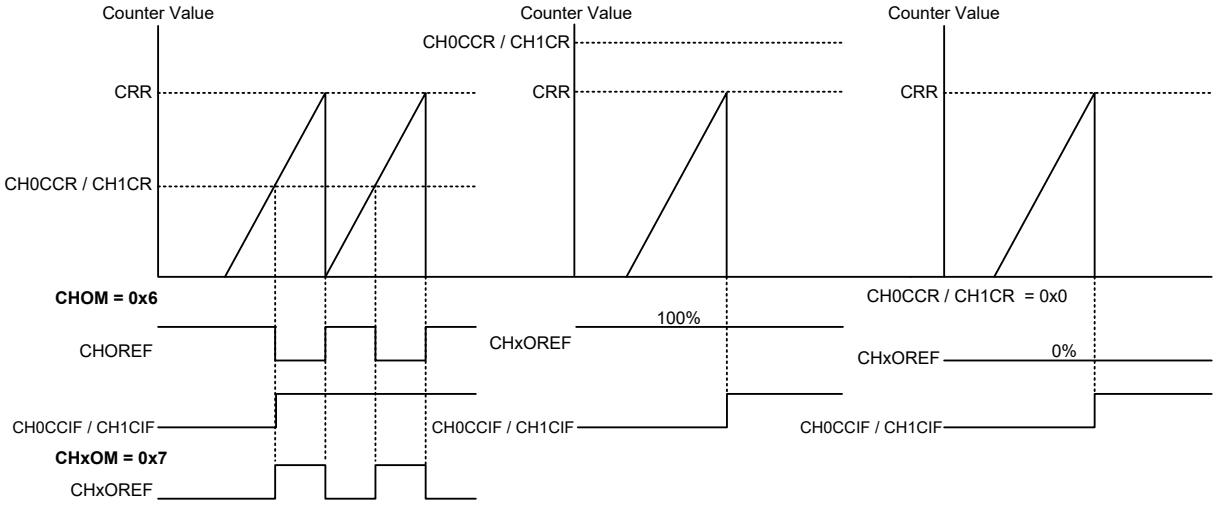


Figure 46. PWM Mode Channel x Output Reference Signal

Update Management

The Update event is used to update the CRR, the PSCR and the CH0CCR or CH1CR values from the actual registers to the corresponding shadow registers. An update event will occur when the counter overflows, the software update control bit is triggered or an update event from the slave controller is generated.

The UEVDIS bit in the CNTCFR register can determine whether the update event occurs or not. When the update event occurs, the corresponding update event interrupt will be generated depending upon whether the update event interrupt generation function is enabled or not by configuring the UGDIS bit in the CNTCFR register. For more detailed description, refer to the UEVDIS and UGDIS bit definition in the CNTCFR register.

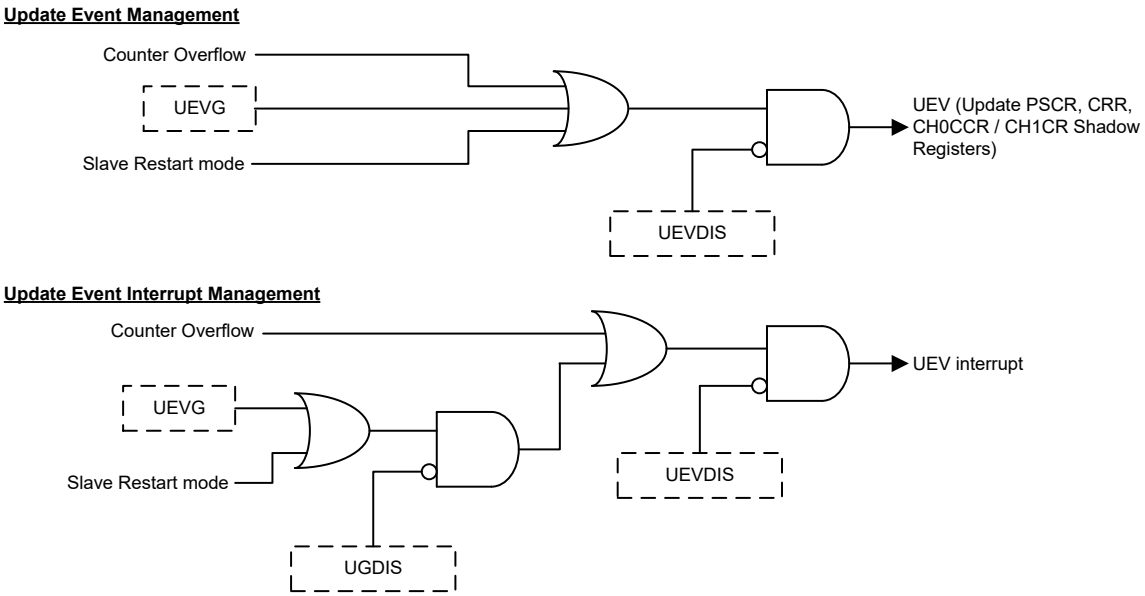


Figure 47. Update Event Setting Diagram

Single Pulse Mode

Once the timer is set to operate in the single pulse mode, it is not necessary to set the timer enable bit TME in the CTR register to 1 to enable the counter. The trigger to generate a pulse can be sourced from the STI signal rising edge or by setting the TME bit to 1 using software. Setting the TME bit to 1 or a trigger from the STI signal rising edge can generate a pulse and then keep the TME bit at a high state until the update event occurs or the TME bit is written to 0 by software. If the TME bit is cleared to 0 using software, the counter will be stopped and its value held. If the TME bit is automatically cleared to 0 by a hardware update event, the counter will be reinitialized.

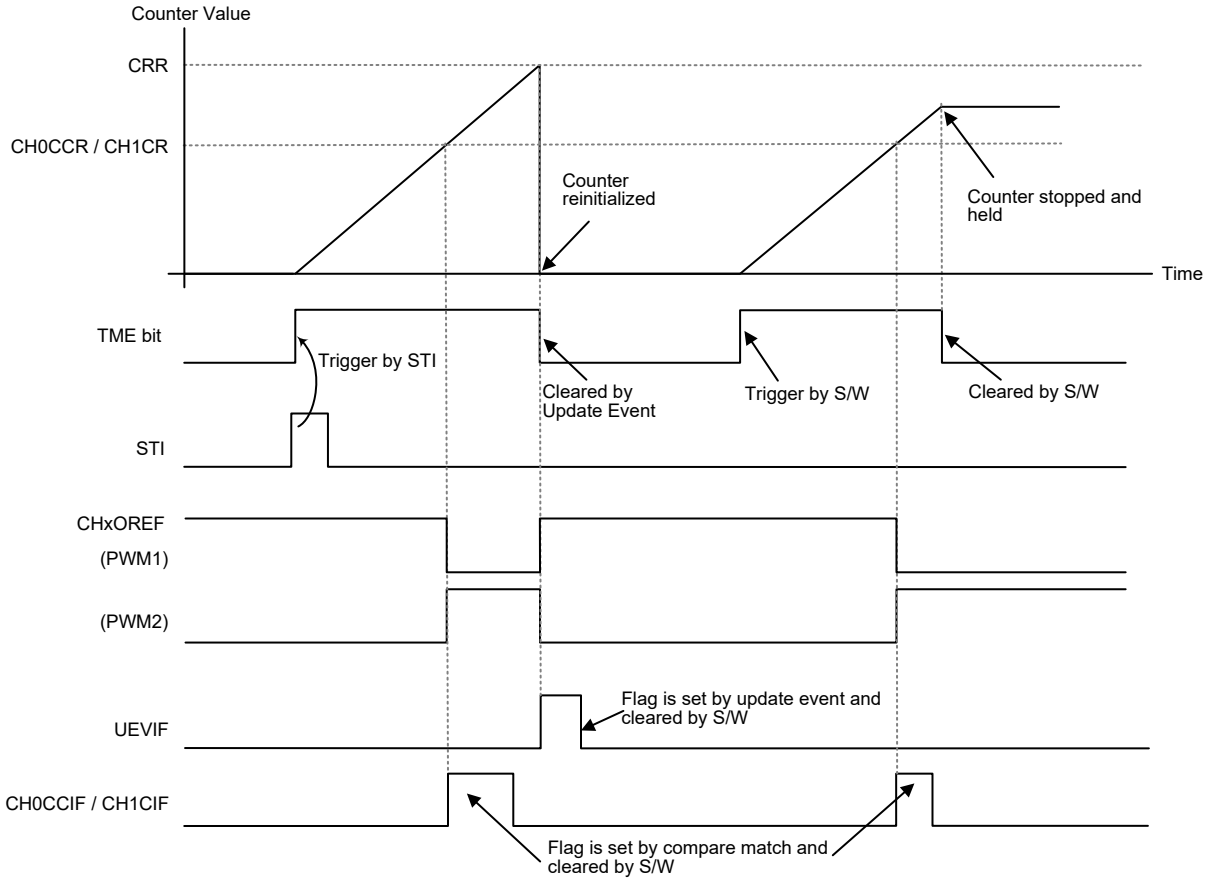


Figure 48. Single Pulse Mode

Register Map

The following table shows the SCTM registers and reset values.

Table 27. SCTM Register Map

Register	Offset	Description	Reset Value
CNTCFR	0x000	Timer Counter Configuration Register	0x0000_0000
MDCFR	0x004	Timer Mode Configuration Register	0x0000_0000
TRCFR	0x008	Timer Trigger Configuration Register	0x0000_0000
CTR	0x010	Timer Control Register	0x0000_0000
CH0ICFR	0x020	Channel 0 Input Configuration Register	0x0000_0000
CH0OCFR	0x040	Channel 0 Output Configuration Register	0x0000_0000
CH1OCFR	0x044	Channel 1 Output Configuration Register	0x0000_0000
CHCTR	0x050	Channel Control Register	0x0000_0000
CHPOLR	0x054	Channel Polarity Configuration Register	0x0000_0000
DICTR	0x074	Timer Interrupt Control Register	0x0000_0000
EVGR	0x078	Timer Event Generator Register	0x0000_0000
INTSR	0x07C	Timer Interrupt Status Register	0x0000_0000
CNTR	0x080	Timer Counter Register	0x0000_0000
PSCR	0x084	Timer Prescaler Register	0x0000_0000
CRR	0x088	Timer Counter-Reload Register	0x0000_FFFF
CH0CCR	0x090	Channel 0 Capture/Compare Register	0x0000_0000
CH1CR	0x094	Channel 1 Compare Register	0x0000_0000

Register Descriptions

Timer Counter Configuration Register – CNTCFR

This register specifies the SCTM counter configuration.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved						CKDIV	
Type/Reset							RW 0	RW 0
	7	6	5	4	3	2	1	0
	Reserved						UGDIS	UEVDIS
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[9:8]	CKDIV	Clock Division These two bits define the frequency ratio between the timer clock (f_{CLKIN}) and the digital filter sampling clock ($f_{sampling}$). 00: $f_{sampling} = f_{CLKIN}$ 01: $f_{sampling} = f_{CLKIN} / 2$ 10: $f_{sampling} = f_{CLKIN} / 4$ 11: $f_{sampling} = f_{CLKIN} / 8$
[1]	UGDIS	Update Event Interrupt Generation Disable Control 0: Any of the following events will generate an update interrupt - Counter overflow - Setting the UEVG bit - Update generation through the slave mode 1: Only counter overflow generates an update interrupt
[0]	UEVDIS	Update Event Disable Control 0: Enable the update event request by one of following events: - Counter overflow - Setting the UEVG bit - Update generation through the slave mode 1: Disable the update event (However the counter and the prescaler are reinitialized if the UEVG bit is set or if a hardware restart is received from the slave mode)

Timer Mode Configuration Register – MDCFR

This register specifies the SCTM master and slave mode selection and single pulse mode setting.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							SPMSET
Type/Reset								RW 0
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved					SMSEL		
Type/Reset						RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	Reserved							
Type/Reset								

Bits	Field	Descriptions
[24]	SPMSET	Single Pulse Mode Setting 0: Counter counts normally irrespective of whether the update event occurred or not 1: Counter stops counting at the next update event and then the TME bit is cleared by hardware
[10:8]	SMSEL	Slave Mode Selection

SMSEL [2:0]	Mode	Descriptions
000	Disable Mode	The prescaler is clocked directly by the internal clock.
100	Restart Mode	The counter value restarts from 0 depending upon the counter mode on the rising edge of the STI signal. The registers will also be updated.
101	Pause Mode	The counter starts to count when the selected trigger input STI is high. The counter stops counting on the instant, not being reset, when the STI signal changes its state to a low level. Both the counter start and stop control are determined by the STI signal.
110	Trigger Mode	The counter starts to count from the original value in the counter on the rising edge of the selected trigger input STI. Only the counter start control is determined by the STI signal.
111	STIED	The rising edge of the selected trigger signal STI will clock the counter.
Others	Reserved	—

Timer Trigger Configuration Register – TRCFR

This register specifies the trigger source selection of SCTM.

Offset: 0x008

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved				TRSEL			
Type/Reset					RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[3:0]	TRSEL	Trigger Source Selection These bits are used to select the trigger input (STI) for counter synchronizing. 0000: Software Trigger by setting the UEVG bit 0001: Filtered input of channel 0 (TI0) 1000: Channel both edge detector (TI0BED) Others: Reserved Note: These bits must be updated only when they are not in use, i.e. the slave mode is disabled by setting the SMSEL field to 0x0.

Timer Control Register – CTR

This register specifies the timer enable bit (TME) and CRR buffer enable bit (CRBE).

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						CRBE	TME
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[1]	CRBE	Counter-Reload Register Buffer Enable 0: Counter-reload register can be updated immediately 1: Counter-reload register cannot be updated until the update event occurs
[0]	TME	Timer Enable bit 0: SCTM off 1: SCTM on – SCTM functions normally When the TME bit is cleared to 0, the counter is stopped and the SCTM consumes no power in any operation mode except for the single pulse mode and the slave trigger mode. In these two modes the TME bit can automatically be set to 1 by hardware which permits all the SCTM registers to function normally.

Channel 0 Input Configuration Register – CH0ICFR

This register specifies the channel 0 input mode configuration.

Offset: 0x020

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved				CH0PSC		CH0CCS	
Type/Reset					RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved				TI0F			
Type/Reset					RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[19:18]	CH0PSC	Channel 0 Capture Input Source Prescaler Setting These bits define the effective events of the channel 0 capture input. Note that the prescaler is reset once the Channel 0 Capture/Compare Enable bit, CH0E, in the Channel Control register named CHCTR is cleared to 0. 00: No prescaler, channel 0 capture input signal is chosen for each active event 01: Channel 0 Capture input signal is chosen for every 2 events 10: Channel 0 Capture input signal is chosen for every 4 events 11: Channel 0 Capture input signal is chosen for every 8 events
[17:16]	CH0CCS	Channel 0 Capture/Compare Selection 00: Channel 0 is configured as an output 01: Channel 0 is configured as an input derived from the TI0 signal 10: Reserved 11: Channel 0 is configured as an input which comes from the TI0BED signal Note: The CH0CCS field can be accessed only when the CH0E bit is cleared to 0.
[3:0]	TI0F	Channel 0 Input Source TI0 Filter N-event Counter Setting The Digital filter in the SCTM is an N-event counter. N is defined as how many valid transitions are necessary to output a filtered signal. 0000: No filter Others: N = TI0F[3:0] + 1

Channel 0 Output Configuration Register – CH0OCFR

This register specifies the channel 0 output mode configuration.

Offset: 0x040

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved			CH0PRE	Reserved	CH0OM		
Type/Reset				RW 0		RW 0	RW 0	RW 0

Bits	Field	Descriptions
[4]	CH0PRE	Channel 0 Capture/Compare Register (CH0CCR) Preload Enable 0: CH0CCR preload function is disabled The CH0CCR register can be immediately assigned a new value when the CH0PRE bit is cleared to 0 and the updated CH0CCR value is used immediately. 1: CH0CCR preload function is enabled The new CH0CCR value will not be transferred to its shadow register until the update event occurs.
[2:0]	CH0OM	Channel 0 Output Mode Setting These bits define the functional types of the output reference signal CH0OREF. 000: No Change 001: Output 0 on compare match 010: Output 1 on compare match 011: Output toggles on compare match 100: Force inactive – CH0OREF is forced to 0 101: Force active – CH0OREF is forced to 1 110: PWM mode 1 - Channel 0 has an active level when CNTR < CH0CCR or otherwise has an inactive level. 111: PWM mode 2 - Channel 0 has an inactive level when CNTR < CH0CCR or otherwise has an active level.

Channel 1 Output Configuration Register – CH1OCFR

This register specifies the channel 1 output mode configuration.

Offset: 0x044

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved			CH1PRE	Reserved	CH1OM		
Type/Reset				RW 0		RW 0	RW 0	RW 0

Bits	Field	Descriptions
[4]	CH1PRE	Channel 1 Compare Register (CH1CR) Preload Enable 0: CH1CR preload function is disabled The CH1CR register can be immediately assigned a new value when the CH1PRE bit is cleared to 0 and the updated CH1CR value is used immediately. 1: CH1CR preload function is enabled The new CH1CR value will not be transferred to its shadow register until the update event occurs.
[2:0]	CH1OM	Channel 1 Output Mode Setting These bits define the functional types of the output reference signal CH1OREF. 000: No Change 001: Output 0 on compare match 010: Output 1 on compare match 011: Output toggles on compare match 100: Force inactive – CH1OREF is forced to 0 101: Force active – CH1OREF is forced to 1 110: PWM mode 1 - Channel 1 has an active level when CNTR < CH1CR or otherwise has an inactive level. 111: PWM mode 2 - Channel 1 has an inactive level when CNTR < CH1CR or otherwise has an active level.

Channel Control Register – CHCTR

This register contains the channel 0 capture input and channel 0 ~ 1 compare output function enable control bits.

Offset: 0x050

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					CH1E	Reserved	CH0E
Type/Reset						RW 0		RW 0

Bits	Field	Descriptions
[2]	CH1E	Channel 1 Compare Enable 0: Off – Channel 1 output signal SCTM_CH1O is not active 1: On – Channel 1 output signal SCTM_CH1O generated on the corresponding output pin
[0]	CH0E	Channel 0 Capture/Compare Enable - Channel 0 is configured as an input (CH0CCS = 0x1/0x3) 0: Input Capture Mode is disabled 1: Input Capture Mode is enabled - Channel 0 is configured as an output (CH0CCS = 0x0) 0: Off – Channel 0 output signal SCTM_CH0O is not active 1: On – Channel 0 output signal SCTM_CH0O generated on the corresponding output pin

Channel Polarity Configuration Register – CHPOLR

This register contains the channel capture input or compare output polarity control.

Offset: 0x054

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					CH1P	Reserved	CH0P
Type/Reset						RW 0		RW 0

Bits	Field	Descriptions
[2]	CH1P	Channel 1 Compare Polarity 0: Channel 1 Output active high 1: Channel 1 Output active low
[0]	CH0P	Channel 0 Capture/Compare Polarity - When Channel 0 is configured as an input (CH0CCS = 0x1/0x3) 0: Capture event occurs on a Channel 0 rising edge 1: Capture event occurs on a Channel 0 falling edge - When Channel 0 is configured as an output (CH0CCS = 0x0) 0: Channel 0 Output active high 1: Channel 0 Output active low

Timer Interrupt Control Register – DICTR

This register contains the timer interrupt enable control bits.

Offset: 0x074

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved					TEVIE	Reserved	UEVIE
Type/Reset						RW 0		RW 0
	7	6	5	4	3	2	1	0
	Reserved						CH1CIE	CH0CCIE
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[10]	TEVIE	Trigger Event Interrupt Enable 0: Trigger event interrupt is disabled 1: Trigger event interrupt is enabled
[8]	UEVIE	Update Event Interrupt Enable 0: Update event interrupt is disabled 1: Update event interrupt is enabled
[1]	CH1CIE	Channel 1 Compare Interrupt Enable 0: Channel 1 interrupt is disabled 1: Channel 1 interrupt is enabled
[0]	CH0CCIE	Channel 0 Capture/Compare Interrupt Enable 0: Channel 0 interrupt is disabled 1: Channel 0 interrupt is enabled

Timer Event Generator Register – EVGR

This register contains the software event generation bits.

Offset: 0x078

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved					TEVG	Reserved	UEVG
Type/Reset						WO 0		WO 0
	7	6	5	4	3	2	1	0
	Reserved						CH1CG	CH0CCG
Type/Reset							WO 0	WO 0

Bits	Field	Descriptions
[10]	TEVG	Trigger Event Generation The trigger event TEV can be generated by setting this bit. It is cleared by hardware automatically. 0: No action 1: TEVIF flag is set
[8]	UEVG	Update Event Generation The update event UEV can be generated by setting this bit. It is cleared by hardware automatically. 0: No action 1: Reinitialize the counter If this bit is set, the counter value returns to 0. An update operation of any related registers will also be performed. For more detail descriptions, refer to the corresponding section.
[1]	CH1CG	Channel 1 Compare Generation A Channel 1 compare event can be generated by setting this bit. It is cleared by hardware automatically. 0: No action 1: Compare event is generated on channel 1 Channel 1 is configured as an output, the CH1CIF bit is set.
[0]	CHCCG	Channel 0 Capture/Compare Generation A Channel 0 capture/compare event can be generated by setting this bit. It is cleared by hardware automatically. 0: No action 1: Capture/compare event is generated on channel 0 If Channel 0 is configured as an input, the counter value is captured into the CH0CCR register and then the CH0CCIF bit is set. If Channel is configured as an output, the CH0CCIF bit is set.

Timer Interrupt Status Register – INTSR

This register stores the timer interrupt status.

Offset: 0x07C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved					TEVIF	Reserved	UEVIF
Type/Reset						W0C 0		W0C 0
	7	6	5	4	3	2	1	0
	Reserved			CH0OCF	Reserved		CH1CIF	CH0CCIF
Type/Reset				W0C 0			W0C 0	W0C 0

Bits	Field	Descriptions
[10]	TEVIF	Trigger Event Interrupt Flag This flag is set by hardware on a trigger event and is cleared by software. 0: No trigger event occurs 1: Trigger event occurs
[8]	UEVIF	Update Event Interrupt Flag. This bit is set by hardware on an update event and is cleared by software. 0: No update event occurs 1: Update event occurs Note: The update event is derived from the following conditions: - The counter overflows - The UEVG bit is asserted - A restart trigger event occurs from the slave trigger input
[4]	CH0OCF	Channel 0 Over-Capture Flag This flag is set by hardware and cleared by software. 0: No over-capture event is detected 1: Capture event occurs again when the CH0CCIF bit is already set and it is not yet cleared by software This bit is set by hardware on an update event and is cleared by software.
[1]	CH1CIF	Channel 1 Compare Interrupt Flag 0: No match event occurs 1: The contents of the counter CNTR have matched the content of the CH1CR register This flag is set by hardware when the counter value matches the CH1CR value. It is cleared by software.

Bits	Field	Descriptions
[0]	CH0CCIF	Channel 0 Capture/Compare Interrupt Flag - Channel 0 is configured as an output: 0: No match event occurs 1: The contents of the counter CNTR have matched the content of the CH0CCR register This flag is set by hardware when the counter value matches the CH0CCR value. It is cleared by software. - Channel 0 is configured as an input: 0: No input capture occurs 1: Input capture occurs This bit is set by hardware on a capture event. It is cleared by software or by reading the CH0CCR register.

Timer Counter Register – CNTR

This register stores the timer counter value.

Offset: 0x080

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	CNTV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	CNTV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	CNTV	Counter Value.

Timer Prescaler Register – PSCR

This register specifies the timer prescaler value to generate the counter clock.

Offset: 0x084

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	PSCV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	PSCV	Prescaler Value These bits are used to specify the prescaler value to generate the counter clock frequency f_{CK_CNT}. $f_{CK\_CNT} = \frac{f_{CK\_PSC}}{PSCV[7:0] + 1}$, where the f_{CK_PSC} is the prescaler clock source.

Timer Counter-Reload Register – CRR

This register specifies the timer counter-reload value.

Offset: 0x088

Reset value: 0x0000_FFFF

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	CRV							
Type/Reset	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1
	7	6	5	4	3	2	1	0
	CRV							
Type/Reset	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1

Bits	Field	Descriptions
[15:0]	CRV	Counter-Reload Value The CRV is the reload value which is loaded into the actual counter register.

Channel 0 Capture/Compare Register – CH0CCR

This register specifies the timer channel capture/compare value.

Offset: 0x090

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	CH0CCV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	CH0CCV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	CH0CCV	Channel 0 Capture/Compare Value - When Channel 0 is configured as an output The CH0CCR value is compared with the counter value and the comparison result is used to trigger the CH0OREF output signal. - When Channel 0 is configured as an input The CH0CCR register stores the counter value captured by the last channel capture event.

Channel 1 Compare Register – CH1CR

This register specifies the timer channel 1 compare value.

Offset: 0x094

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	CH1CV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	CH1CV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	CH1CV	Channel 1 Compare Value - When Channel 1 is configured as an output The CH1CR value is compared with the counter value and the comparison result is used to trigger the CH1OREF output signal.

14 Basic Function Timer (BFTM)

Introduction

The Basic Function Timer is a 16-bit up-counting counter designed to measure time intervals, generate one shot pulses or generate repetitive interrupts. The BFTM can operate in two modes which are repetitive and one shot modes. The repetitive mode restarts the counter at each compare match event which is generated by the internal comparator. The BFTM also supports a one shot mode which will force the counter to stop counting when a compare match event occurs.

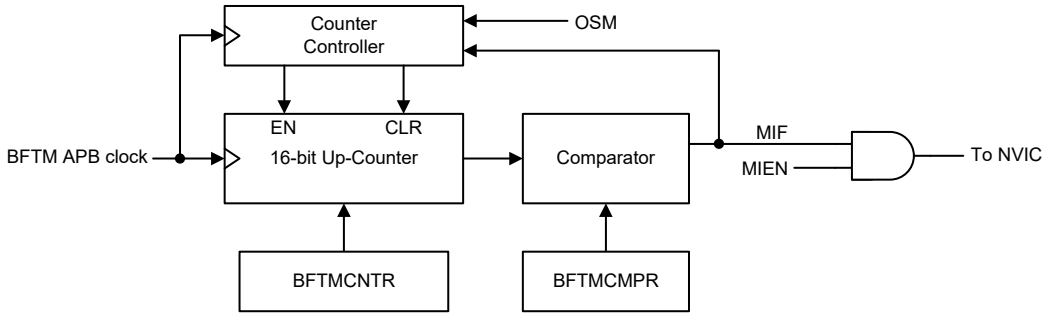


Figure 49. BFTM Block Diagram

Features

- 16-bit up-counting counter
- Compare Match function
- Includes debug mode
- Clock source: BFTM APB clock
- Counter value can be Read/Written on the fly
- One shot mode: counter stops counting when compare match occurs
- Repetitive mode: counter restarts when compare match occurs
- Compare Match interrupt enable/disable control

Functional Description

The BFTM is a 16-bit up-counting counter which is driven by the BFTM APB clock, PCLK. The counter value can be changed or read at any time even when the timer is counting. The BFTM supports two operating modes known as the repetitive mode and one shot mode allowing the measurement of time intervals or the generation of periodic time durations.

Repetitive Mode

The BFTM counts up from zero to a specific compare value which is pre-defined by the BFTMCMPR register. When the BFTM operates in the repetitive mode and the counter reaches a value equal to the specific compare value in the BFTMCMPR register, the timer will generate a compare match event signal, MIF. When this occurs, the counter will be reset to 0 and resume its counting operation. When the MIF signal is generated, a BFTM compare match interrupt will also

be generated periodically if the compare match interrupt is enabled by setting the corresponding interrupt control bit, MIEN, to 1. The counter value will remain unchanged and the counter will stop counting if it is disabled by clearing the CEN bit to 0.

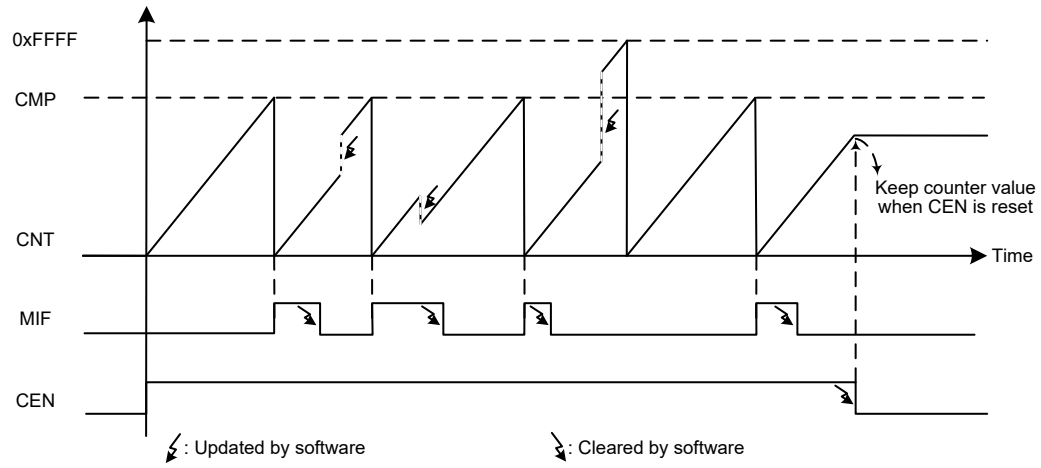


Figure 50. BFTM – Repetitive Mode

One Shot Mode

By setting the OSM bit in BFTMCR register to 1, the BFTM will operate in the one shot mode. The BFTM starts to count when the CEN bit is set to 1 by the application program. The counter value will remain unchanged if the CEN bit is cleared to 0 by the application program. However, the counter value will be reset to 0 and stop counting when the CEN bit is cleared automatically to 0 by the internal hardware when a counter compare match event occurs.

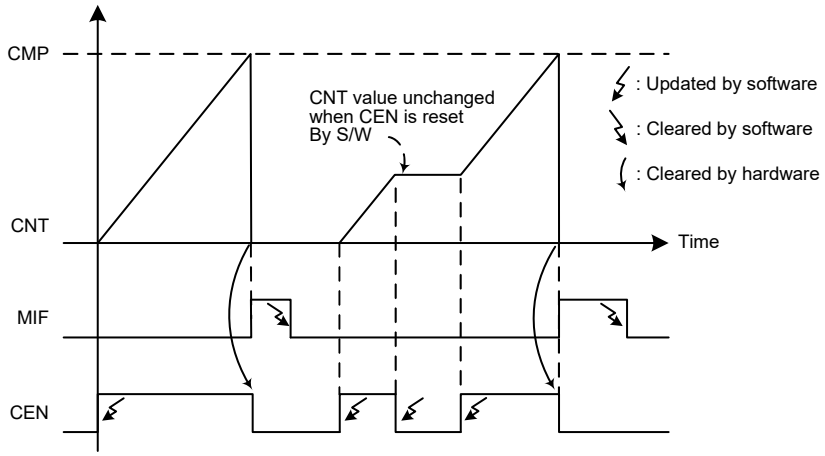


Figure 51. BFTM – One Shot Mode

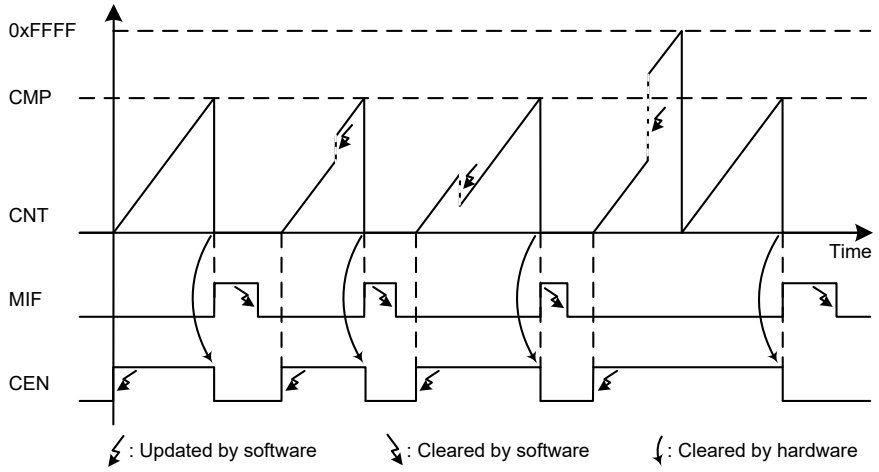


Figure 52. BFTM – One Shot Mode Counter Updating

Register Map

The following table shows the BFTM registers and reset values.

Table 28. BFTM Register Map

Register	Offset	Description	Reset Value
BFTMCR	0x000	BFTM Control Register	0x0000_0000
BFTMSR	0x004	BFTM Status Register	0x0000_0000
BFTMCNTR	0x008	BFTM Counter Value Register	0x0000_0000
BFTMCMPR	0x00C	BFTM Compare Value Register	0x0000_FFFF

Register Descriptions

BFTM Control Register – BFTMCR

This register specifies the overall BFTM control bits.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					CEN	OSM	MIEN
Type/Reset						RW 0	RW 0	RW 0

Bits	Field	Descriptions
[2]	CEN	BFTM Counter Enable Control 0: BFTM is disabled 1: BFTM is enabled When this bit is set to 1, the BFTM counter will start to count. The counter will stop counting and the counter value will remain unchanged when the CEN bit is cleared to 0 by the application program regardless of whether it is in the repetitive or one shot mode. However, in the one shot mode, the counter will stop counting and be reset to 0 when the CEN bit is cleared to 0 by the timer hardware circuitry which results from a compare match event.
[1]	OSM	BFTM One Shot Mode Selection 0: Counter operates in repetitive mode 1: Counter operates in one shot mode
[0]	MIEN	BFTM Compare Match Interrupt Enable Control 0: Compare Match Interrupt is disabled 1: Compare Match Interrupt is enabled

BFTM Status Register – BFTMSR

This register specifies the BFTM status.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved							MIF
Type/Reset								W0C 0

Bits	Field	Descriptions
[0]	MIF	BFTM Compare Match Interrupt Flag 0: No compare match event occurs 1: Compare match event occurs When the counter value, CNT, is equal to the compare register value, CMP, a compare match event will occur and the corresponding interrupt flag, MIF, will be set. The MIF bit is cleared to 0 by writing a data “0”.

BFTM Counter Value Register – BFTMCNTR

This register specifies the BFTM counter value.

Offset: 0x008

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	CNT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	CNT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	CNT	BFTM Counter Value A 16-bit BFTM counter value is stored in this field which can be read or written on the fly.

BFTM Compare Value Register – BFTMCMPR

The register specifies the BFTM compare value.

Offset: 0x00C

Reset value: 0x0000_FFFF

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	CMP							
Type/Reset	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1
	7	6	5	4	3	2	1	0
	CMP							
Type/Reset	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1

Bits	Field	Descriptions
[15:0]	CMP	BFTM Compare Value This register specifies a 16-bit BFTM compare value which is used for comparison with the BFTM counter value.

15 Real-Time Clock (RTC)

Introduction

The Real-Time Clock, RTC, circuitry includes the APB interface, a 24-bit up-counter, a control register, a prescaler, a compare register and a status register. Most of the RTC circuits are located in the V_{CORE} Power Domain, as shown shaded in dotted box in the accompanying figure. When the device enters the power saving mode, the RTC counter is used as a wakeup timer to let the system resume from the power saving modes. The detailed RTC function will be described in the following sections.

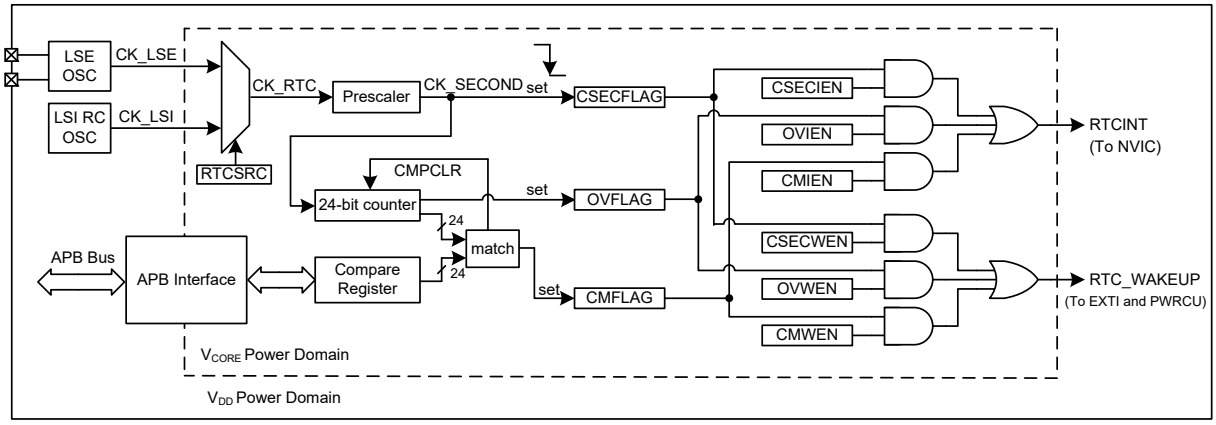


Figure 53. RTC Block Diagram

Features

- 24-bit up-counter for counting elapsed time
- Programmable clock prescaler
  - Division factor: 1, 2, 4, 8…, 32768
- 24-bit compare register for alarm usage
- RTC clock source
  - LSE oscillator clock
  - LSI oscillator clock
- Three RTC Interrupt/wakeup settings
  - RTC second clock interrupt/wakeup
  - RTC compare match interrupt/wakeup
  - RTC counter overflow interrupt/wakeup
- The RTC interrupt/wakeup event can work together with power management to wake up the chip from power saving mode

Functional Descriptions

RTC Related Register Reset

Most of the RTC registers can be reset by either a V_{CORE} Domain power on reset, V_{CORE}_POR, or by a V_{CORE} Domain software reset by setting the PWCURST bit in the PWRCR register. Other reset events have no effect to clear the RTC registers.

Low Speed Clock Configuration

The default RTC clock source, CK_RTC, is derived from the LSI oscillator. The CK_RTC clock can be derived from either the external 32,768 Hz crystal oscillator, named the LSE oscillator, or the internal 32 kHz RC oscillator named the LSI oscillator, by setting the RTCSRC bit in the RTCCR register. A prescaler is provided to divide the CK_RTC by a ratio ranged from 2^0 to 2^{15} determined by the RPRE [3:0] field. For instance, setting the prescaler value RPRE [3:0] to 0xF will generate an exact 1 Hz CK_SECOND clock if the CK_RTC clock frequency is equal to 32,768 Hz. The LSE oscillator can be enabled by the LSEEN control bit in the RTCCR register respectively. In addition, the LSE oscillator startup mode can be selected by configuring the LSESM bit in the RTCCR register. This enables the LSE oscillator to have either a shorter startup time or a lower power consumption, both of which are traded off depending upon specific application requirements. An example of the startup time and the power consumption for different startup modes are shown in the accompanying table for reference.

Table 29. LSE Startup Mode Operating Current and Startup Time

Startup Mode	LSESM Setting in the RTCCR Register	Operating Current	Startup Time
Normal startup	0	2.0 μA	Above 500 ms
Fast startup	1	3.5 μA	Below 300 ms

@ V_{DD} = 3.3 V and LSE clock = 32,768 Hz; these values are only for reference, actual values are dependent on the specification of the external 32.768 kHz crystal.

RTC Counter Operation

The RTC provides a 24-bit up-counter which increases at the falling edge of the CK_SECOND clock and whose value can be read from the RTCCNT register asynchronously via the APB bus. A 24-bit compare register, RTCCMP, is provided to store the specific value to be compared with the RTCCNT content. This is used to define a pre-determined time interval. When the RTCCNT register content is equal to the RTCCMP register value, the match flag CMFLAG in the RTCSR register will be set by hardware and an interrupt or wakeup event can be sent according to the corresponding enable bits in the RTCIWEN register. The RTC counter will be either reset to zero or keep counting when the compare match event occurs, dependent upon the CMPCLR bit in the RTCCR register. For example, if the RPRE [3:0] is set to 0xF, the RTCCMP register content is set to a decimal value of 60 and the CMPCLR bit is set to 1, then the CMFLAG bit will be set every minute. In addition, the OVFLAG bit in the RTCSR register will be set when the RTC counter overflows. A read operation on the RTCSR register clears the status flags including the CSECFLAG, CMFLAG and OVFLAG bits.

Interrupt and Wakeup Control

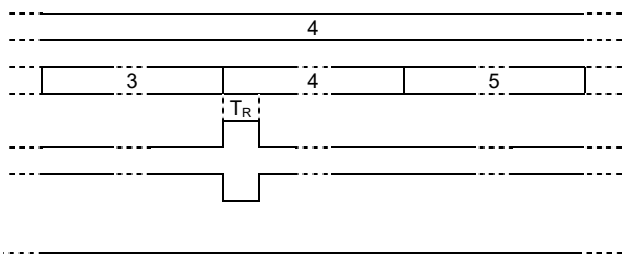
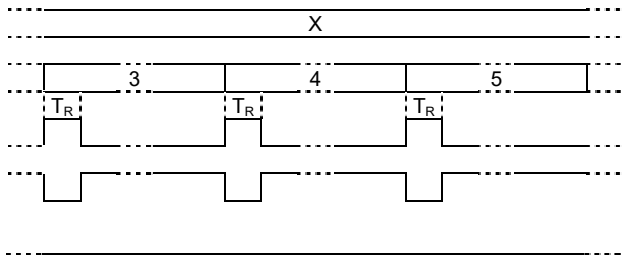
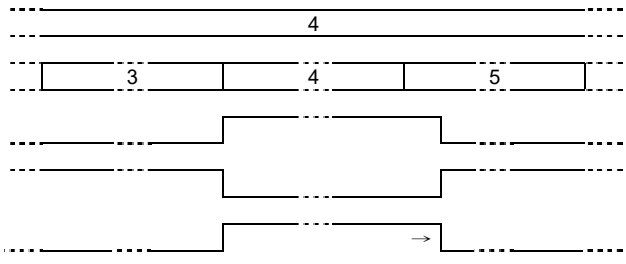
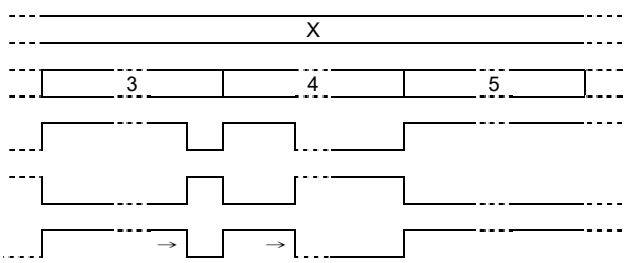
The falling edge of the CK_SECOND clock causes the CSECFLAG bit in the RTCSR register to be set and generates an interrupt if the corresponding interrupt enable bit, CSECIEN, in the RTCIWEN register is set. The wakeup event can also be generated to wake up the HSI/HSE

oscillators, the LDO and the MCU core if the corresponding wakeup enable bit CSECWEN is set. When the RTC counter overflows or a compare match event occurs, it will generate an interrupt or a wake up event determined by the corresponding interrupt or wakeup enable control bits, OVIEN/OVWEN or CMIEN/CMWEN bits, in the RTCIWEN register. Refer to the related register definitions for more details.

RTCOUT Output Pin Configuration

The following table shows the RTCOUT output format according to the mode, polarity and event selection setting.

Table 30. RTCOUT Output Mode and Active Level Setting

ROWM	ROES	RTCOUT Output Waveform
0 (Pulse mode)	0 (Compare match)	
	1 (Second clock)	
1 (Level mode)	0 (Compare match)	
	1 (Second clock)	

T_R: RTCOUT output pulse time = 1 / f_{CK_RTC}
→: Cleared by software reading ROLF bit

Register Map

The following table shows the RTC registers and reset values. Note all the registers in this unit are located at the V_{CORE} power domain.

Table 31. RTC Register Map

Register	Offset	Description	Reset Value
RTCCNT	0x000	RTC Counter Register	0x0000_0000
RTCCMP	0x004	RTC Compare Register	0x0000_0000
RTCCR	0x008	RTC Control Register	0x0000_0F04
RTCSR	0x00C	RTC Status Register	0x0000_0000
RTCIWEN	0x010	RTC Interrupt and Wakeup Enable Register	0x0000_0000

Register Descriptions

RTC Counter Register – RTCCNT

This register defines a 24-bit up-counter which is increased by the CK_SECOND clock.

Address: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	RTCCNTV							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	15	14	13	12	11	10	9	8
	RTCCNTV							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	RTCCNTV							
Type/Reset	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

Bits	Field	Descriptions
[23:0]	RTCCNTV	RTC Counter Value The current value of the RTC counter is returned when reading the RTCCNT register. The RTCCNT register is updated during the falling edge of the CK_SECOND clock. This register is reset by one of the following conditions: - V_{CORE} Domain software reset – set the PWCURST bit in the PWRCR register - V_{CORE} Domain power on reset – V_{CORE}_POR - Compare match (RTCCNT = RTCCMP) when CMPCLR = 1 (in the RTCCR register) - RTCEN bit changed from 0 to 1

RTC Compare Register – RTCCMP

This register defines a specific value to be compared with the RTC counter value.

Address: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	RTCCMPV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	RTCCMPV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	RTCCMPV							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[23:0]	RTCCMPV	RTC Compare Match Value A match condition happens when the value in the RTCCNT register is equal to the RTCCMP value. An interrupt can be generated if the CMIEN bit in the RTCIWEN register is set. When the CMPCLR bit in the RTCCR register is set to 0 and a match condition happens, the CMFLAG bit in the RTCSR register is set while the value in the RTCCNT register is not affected and will continue to count until overflow. When the CMPCLR bit is set to 1 and a match condition happens, the CMFLAG bit in the RTCSR register is set and the RTCCNT register will be reset to zero and then the counter continues to count.

RTC Control Register – RTCCR

This register specifies a range of RTC circuitry control bits.

Address: 0x008

Reset value: 0x0000_0F04

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved			ROLF	ROAP	ROWM	ROES	ROEN
Type/Reset				RC 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved				RPRE			
Type/Reset					RW 1	RW 1	RW 1	RW 1
	7	6	5	4	3	2	1	0
		Reserved	LSESM	CMPCLR	LSEEN	Reserved	RTCSRC	RTCEN
Type/Reset			RW 0	RW 0	RW 0		RW 0	RW 0

Bits	Field	Descriptions
[20]	ROLF	RTCOUT Level Mode Flag 0: RTCOUT Output is inactive 1: RTCOUT Output is holding as active level Set by hardware when in the level mode (ROWM = 1) and a RTCOUT output event occurs. Cleared by software reading this flag. The RTCOUT signal will return to the inactive level after software has read this bit.
[19]	ROAP	RTCOUT Output Active Polarity 0: Active level is high 1: Active level is low
[18]	ROWM	RTCOUT Output Waveform Mode 0: Pulse mode The output pulse duration is one RTC clock (CK_RTC) period. 1: Level mode The RTCOUT signal will remain at an active level until the ROLF bit is cleared by software reading the ROLF bit.
[17]	ROES	RTCOUT Output Event Selection 0: RTC compare match is selected 1: RTC second clock (CK_SECOND) event is selected The ROES bit can be used to select whether the RTCOUT signal is output on the RTCOUT pin when a RTC compare match event or the RTC second clock (CK_SECOND) event occurs.
[16]	ROEN	RTCOUT Output Pin Enable 0: Disable RTCOUT output pin 1: Enable RTCOUT output pin When the ROEN bit is set to 1, the RTCOUT signal will be at an active level once a RTC compare match or the RTC second clock (CK_SECOND) event occurs. The active polarity and output waveform mode can be configured by the ROAP and ROWM bits respectively. When the ROEN bit is cleared to 0, the RTCOUT pin will be in a floating state.

Bits	Field	Descriptions
[11:8]	RPRE	RTC Clock Prescaler Select CK_SECOND = CK_RTC / 2^{RPRE} 0000: CK_SECOND = CK_RTC / 2^0 0001: CK_SECOND = CK_RTC / 2^1 0010: CK_SECOND = CK_RTC / 2^2 … 1111: CK_SECOND = CK_RTC / 2^{15}
[5]	LSESM	LSE oscillator Startup Mode 0: Normal startup and requires less operating power 1: Fast startup but requires higher operating current
[4]	CMPCLR	Compare Match Counter Clear 0: 24-bit RTC counter is not affected when compare match condition occurs 1: 24-bit RTC counter is cleared when compare match condition occurs
[3]	LSEEN	LSE oscillator Enable Control 0: LSE oscillator is disabled 1: LSE oscillator is enabled
[1]	RTCSRC	RTC Clock Source Selection 0: LSI oscillator is selected as the RTC clock source 1: LSE oscillator is selected as the RTC clock source
[0]	RTCEN	RTC Enable Control 0: RTC is disabled 1: RTC is enabled

RTC Status Register – RTCSR

This register stores the counter flags.

Address: 0x00C

Reset value: 0x0000_0000 (It can also be reset by changing the RTCEN bit from 1 to 0)

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved					OVFLAG	CMFLAG	CSECFLAG
Type/Reset						RC 0	RC 0	RC 0

Bits	Field	Descriptions
[2]	OVFLAG	Counter Overflow Flag 0: Counter overflow has not occurred since the last RTCSR register read operation 1: Counter overflow has occurred since the last RTCSR register read operation This bit is set by hardware when the counter value in the RTCCNT register changes from 0xFF_FFFF to 0x00_0000 and cleared by read operation. This bit is suggested to be read in the RTC IRQ handler and should be taken care when software polling is used.
[1]	CMFLAG	Compare Match Condition Flag 0: Compare match condition has not occurred since the last RTCSR register read operation 1: Compare match condition has occurred since the last RTCSR register read operation. This bit is set by hardware on the CK_SECOND clock falling edge when the RTCCNT register value is equal to the RTCCMP register content. It is cleared by software reading this bit. This bit is suggested for access in the corresponding RTC interrupt routine – do not use software polling during software free running.
[0]	CSECFLAG	CK_SECOND Occurrence Flag 0: CK_SECOND has not occurred since the last RTCSR register read operation 1: CK_SECOND has occurred since the last RTCSR register read operation This bit is set by hardware on the CK_SECOND clock falling edge. It is cleared by software reading this bit. This bit is suggested for access in the corresponding RTC interrupt routine – do not use software polling during software free running.

RTC Interrupt and Wakeup Enable Register – RTCIWEN

This register contains the interrupt and wakeup enable bits.

Address: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved					OVWEN	CMWEN	CSECWEN
Type/Reset						RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	Reserved					OVIEN	CMIEN	CSECIEN
Type/Reset						RW 0	RW 0	RW 0

Bits	Field	Descriptions
[10]	OVWEN	Counter Overflow Wakeup Enable 0: Counter overflow wakeup is disabled 1: Counter overflow wakeup is enabled
[9]	CMWEN	Compare Match Wakeup Enable 0: Compare match wakeup is disabled 1: Compare match wakeup is enabled
[8]	CSECWEN	Counter Clock CK_SECOND Wakeup Enable 0: Counter Clock CK_SECOND wakeup is disabled 1: Counter Clock CK_SECOND wakeup is enabled
[2]	OVIEN	Counter Overflow Interrupt Enable 0: Counter Overflow Interrupt is disabled 1: Counter Overflow Interrupt is enabled
[1]	CMIEN	Compare Match Interrupt Enable 0: Compare Match Interrupt is disabled 1: Compare Match Interrupt is enabled
[0]	CSECIEN	Counter Clock CK_SECOND Interrupt Enable 0: Counter Clock CK_SECOND Interrupt is disabled 1: Counter Clock CK_SECOND Interrupt is enabled

16 Watchdog Timer (WDT)

Introduction

The Watchdog Timer is a hardware timing circuitry that can be used to detect a system lock-up due to software trapped in a deadlock. The Watchdog Timer can be operated in a reset mode. The Watchdog Timer will generate a reset when the counter counts down to a zero value. Therefore, the software should reload the counter value before a Watchdog Timer underflow occurs. In addition, a reset is also generated if the software reloads the counter before it reaches a delta value. That means that the Watchdog Timer prevents a software deadlock that continuously triggers the Watchdog, the reload must occur when the Watchdog Timer value has a value within a limited window of 0 and WDTD. The Watchdog Timer counter can be stopped when the processor is in the debug or the three sleep modes. The register write protection function can be enabled to prevent an unexpected change in the Watchdog Timer configuration.

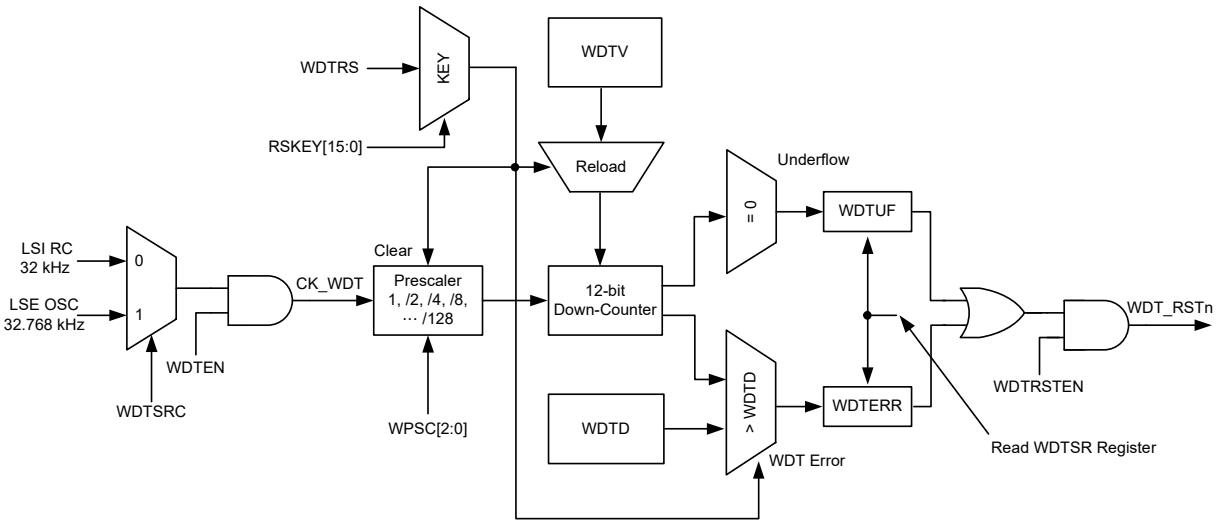


Figure 54. Watchdog Timer Block Diagram

Features

- Clock source from either the internal 32 kHz RC oscillator (LSI) or the external 32,768 Hz oscillator (LSE)
- Can be independently setup to keep running or to stop when entering the Sleep or Deep-Sleep1 mode
- 12-bit down-counter with 3-bit prescaler structure
- Provides reset to the system
- Limited reload window setup function for custom Watchdog Timer reload times
- Watchdog Timer may be stopped when the processor is in the debug mode
- Reload lock key to prevent unexpected operation
- Configuration register write protection function for counter value, reset enable, delta value, and prescaler value

Functional Description

The Watchdog Timer is formed from a 12-bit count-down counter and a fixed 3-bit prescaler. The largest time-out period is 16 seconds, using the LSE or LSI clock and a 1/128 maximum prescaler value.

The Watchdog Timer configuration setup includes programmable Counter-Reload value, reset enable, window value and prescaler value. These configurations are set using the WDTMR0 and WDTMR1 registers which must be properly programmed before the Watchdog Timer starts counting. In order to prevent unexpected write operations to those configurations, a register write protection function can be enabled by writing any value, other than 0x35CA to PROTECT[15:0], in the WDTPR register. A value of 0x35CA can be written to PROTECT[15:0] to disable the register write protection function before accessing any configuration register. A read operation on PROTECT[0] can obtain the enable/disable status of the register write protection function.

During normal operation, the Watchdog Timer counter should be reloaded before it underflows to prevent the generation of a Watchdog reset. The 12-bit count-down counter can be reloaded with the required Watchdog Timer Counter Value (WDTV) by first setting the WDTRS bit to 1 with the correct key, which is 0x5FA0 in the WDTCR register.

If a software deadlock occurs during a Watchdog Timer reload routine, the reload operation will still go ahead and therefore the software deadlock cannot be detected. To prevent this situation from occurring, the reload operation must be executed in such a way that the value of the Watchdog Timer counter is limited to within a delta value (WDTD). If the Watchdog Timer counter value is greater than the delta value and a reload operation is executed, a Watchdog Timer error will occur. The Watchdog Timer error will cause a Watchdog reset if the related functional control is enabled. Additionally, the above features can be disabled by programming a WDTD value greater than or equal to the WDTV value.

The WDTERR and WDTUF flags in the WDTSR register will be set respectively when the Watchdog Timer error occurs or when a Watchdog Timer underflows. A system reset or writing “1” operation on the WDTSR register will clear the WDTERR and WDTUF flags.

The Watchdog Timer uses two clocks: PCLK and CK_WDT. The PCLK clock is used for APB access to the watchdog registers. The CK_WDT clock is used for the Watchdog Timer functionality and counting. There is some synchronization logic between these two clock domains.

When the system enters the Sleep mode or Deep-Sleep1 mode, the Watchdog Timer counter will either continue to count or stop depending on the WDTSHLT field setup in the WDTMR0 register. However, the Watchdog Timer will always stop when the system is in the Deep-Sleep2 mode. When the Watchdog stops counting, the count value is retained so that it continues counting after the system is woken up from these three sleep modes. A Watchdog reset will occur any time when the Watchdog Timer is running and when it has an operating clock source. When the system enters the debug mode, the Watchdog Timer counter will either continue to count or stop depending on the DBWDT bit of the MCUDBGCR register in the Clock Control Unit.

The Watchdog timer should be used in the following manners:

- Set the Watchdog Timer reload value (WDTV) and reset in the WDTMR0 register.
- Set the Watchdog Timer delta value (WDTD) and prescaler in the WDTMR1 register.
- Start the Watchdog Timer by writing to the WDTCR register with WDTRS = 1 and RSKEY = 0x5FA0.

- Write to the WDTPR register to lock all the Watchdog Timer registers except for WDTCR and WDTPR.
- The Watchdog Timer counter should be reloaded again within the delta value (WDTD).

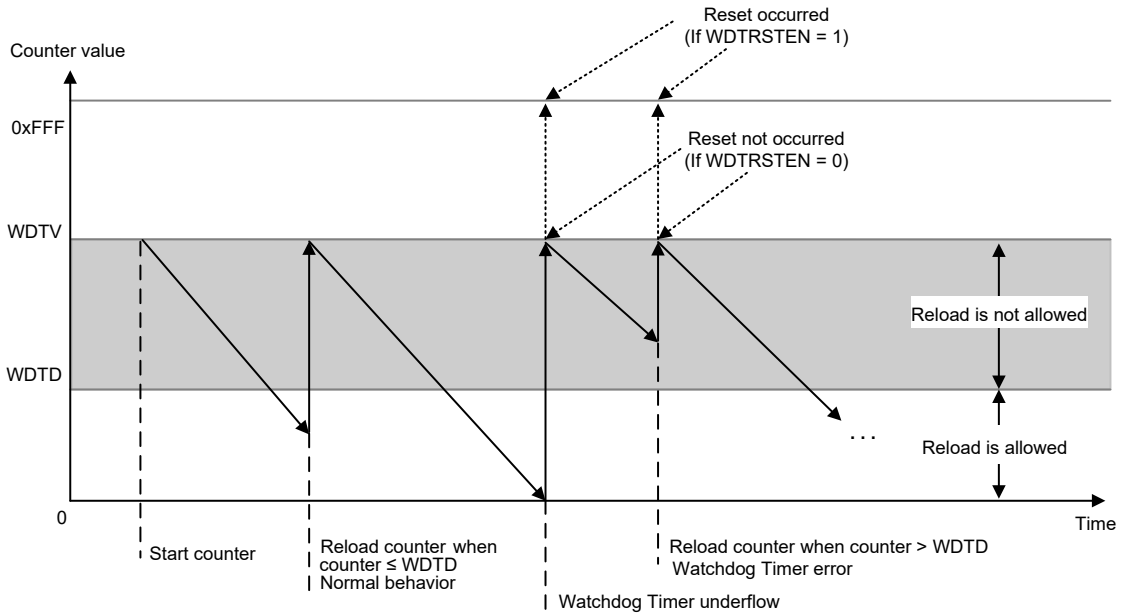


Figure 55. Watchdog Timer Behavior

Register Map

The following table shows the Watchdog Timer registers and reset values.

Table 32. Watchdog Timer Register Map

Register	Offset	Description	Reset Value
WDTCR	0x000	Watchdog Timer Control Register	0x0000_0000
WDTMR0	0x004	Watchdog Timer Mode Register 0	0x0000_0FFF
WDTMR1	0x008	Watchdog Timer Mode Register 1	0x0000_7FFF
WDTSR	0x00C	Watchdog Timer Status Register	0x0000_0000
WDTPR	0x010	Watchdog Timer Protection Register	0x0000_0000
WDTCSR	0x018	Watchdog Timer Clock Selection Register	0x0000_0000

Register Descriptions

Watchdog Timer Control Register – WDTCR

This register is used to reload the Watchdog timer.

Offset: 0x000
Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	RSKEY							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	RSKEY							
Type/Reset	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	Reserved							
Type/Reset								
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	Reserved							WDTRS
Type/Reset								WO 0

Bits	Field	Descriptions
[31:16]	RSKEY	Watchdog Timer Reload Lock Key The RSKEY [15:0] bits should be written with a 0x5FA0 value to enable the WDT reload operation function. Writing any other value except 0x5FA0 in this field will abort the write operation.
[0]	WDTRS	Watchdog Timer Reload 0: No effect 1: Reload Watchdog Timer This bit is used to reload the Watchdog timer counter as a WDTV value which is stored in the WDTMR0 register. It is set to 1 by software and cleared to 0 by hardware automatically.

Watchdog Timer Mode Register 0 – WDTMR0

This register specifies the Watchdog timer Counter-Reload value and reset enable control.

Offset: 0x004

Reset value: 0x0000_0FFF

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							WDTEN
Type/Reset								RW 0
	15	14	13	12	11	10	9	8
		WDTSHLT	WDTRSTEN	Reserved	WDTV			
Type/Reset	RW 0	RW 0	RW 0		RW 1	RW 1	RW 1	RW 1
	7	6	5	4	3	2	1	0
	WDTV							
Type/Reset	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1

Bits	Field	Descriptions
[16]	WDTEN	Watchdog Timer Running Enable 0: Watchdog Timer is disabled 1: Watchdog Timer is enabled to run When the Watchdog Timer is disabled, the counter will be reset to its hardware default condition. When the WDTEN bit is set, the Watchdog Timer will be reloaded with the WDTV value and count down.
[15:14]	WDTSHLT	Watchdog Timer Sleep Halt 00: The Watchdog runs when the system is in the Sleep mode or Deep-Sleep1 mode 01: The Watchdog runs when the system is in the Sleep mode and halts in Deep-Sleep1 mode 10 or 11: The Watchdog halts when the system is in the Sleep mode and Deep-Sleep1 mode Note that the Watchdog timer always halts when the system is in the Deep-Sleep2 mode. The Watchdog timer stops counting when the WDTSHLT field is properly configured in the Sleep mode or Deep-Sleep1 mode. When the Watchdog stops counting, the count value is retained so that it continues counting after the system wakes up from these three sleep modes. If a Watchdog reset occurs in the Sleep or Deep-Sleep1 mode, it will wake up the device.
[13]	WDTRSTEN	Watchdog Timer Reset Enable 0: A Watchdog Timer underflow or error has no effect on the system reset 1: A Watchdog Timer underflow or error triggers a Watchdog Timer system reset
[11:0]	WDTV	Watchdog Timer Counter Value WDTV defines the value loaded into the 12-bit Watchdog down-counter.

Watchdog Timer Mode Register 1 – WDTMR1

This register specifies the Watchdog delta value and the prescaler selection.

Offset: 0x008

Reset value: 0x0000_7FFF

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved	WPSC			WDTD			
Type/Reset		RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1
	7	6	5	4	3	2	1	0
	WDTD							
Type/Reset	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1

Bits	Field	Descriptions
[14:12]	WPSC	Watchdog Timer Prescaler Selection 000: 1/1 001: 1/2 010: 1/4 011: 1/8 100: 1/16 101: 1/32 110: 1/64 111: 1/128
[11:0]	WDTD	Watchdog Timer Delta Value Define the permitted range to reload the Watchdog Timer. If the Watchdog Timer counter value is less than or equal to WDTD, writing to the WDTCR register with WDTRS = 1 and RSKEY = 0x5FA0 will reload the timer. If the Watchdog Timer value is greater than WDTD, then writing WDTCR with WDTRS = 1 and RSKEY = 0x5FA0 will cause a Watchdog Timer error. This feature can be disabled by programming a WDTD value greater then or equal to the WDTV value.

Watchdog Timer Status Register – WDTSR

This register specifies the Watchdog timer status.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						WDTERR	WDTUF
Type/Reset							WC 0	WC 0

Bits	Field	Descriptions
[1]	WDTERR	Watchdog Timer Error 0: No Watchdog Timer error has occurred since the last read of this register 1: A Watchdog Timer error has occurred since the last read of this register Note: A reload operation when the Watchdog Timer counter value is larger than WDTD causes a Watchdog Timer error. Note that this bit is a write-one-clear flag.
[0]	WDTUF	Watchdog Timer Underflow 0: No Watchdog Timer underflow has occurred since the last read of this register 1: A Watchdog Timer underflow has occurred since the last read of this register Note that this bit is a write-one-clear flag.

Watchdog Timer Protection Register – WDTPR

This register specifies the Watchdog timer protect key configuration.

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	PROTECT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	PROTECT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	PROTECT	Watchdog Timer Register Protection For write operation: 0x35CA: Disable the Watchdog Timer register write protection Others: Enable the Watchdog Timer register write protection For read operation: 0x0000: Watchdog Timer register write protection is disabled 0x0001: Watchdog Timer register write protection is enabled This register is used to enable/disable the Watchdog timer configuration register write protection function. All configuration registers become read only except for WDTCR and WDTPR when the register write protection is enabled. Additionally, the read operation of PROTECT[0] can obtain the enable/disable status of the register write protection function.

Watchdog Timer Clock Selection Register – WDTCSR

This register specifies the Watchdog timer clock source selection and lock configuration.

Offset: 0x018

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved			WDTLOCK	Reserved			WDTSRC
Type/Reset				RW 0				RW 0

Bits	Field	Descriptions
[4]	WDTLOCK	Watchdog Timer Lock Mode 0: This bit is only cleared to 0 on any reset, it cannot be cleared by software 1: This bit is set once only by software and locks the Watchdog Timer function Software can set this bit to 1 at any time. Once the WDTLOCK bit is set, the function and registers of the Watchdog Timer cannot be modified or disabled, including the Watchdog Timer clock source. The lock mode can only be disabled until a system reset occurs.
[0]	WDTSRC	Watchdog Timer Clock Source Selection 0: Internal 32 kHz RC oscillator clock is selected (LSI) 1: External 32.768 kHz crystal oscillator clock is selected (LSE) Select using software to control the Watchdog timer clock source.

17 Inter-Integrated Circuit (I²C)

Introduction

The I²C Module is an internal circuit allowing communication with an external I²C interface which is an industry standard two-line serial interface used for connection to external hardware. These two serial lines are known as a serial data line, SDA, and a serial clock line, SCL. The I²C module provides three data transfer rates: 100 kHz in the Standard mode, 400 kHz in the Fast mode and 1 MHz in the Fast mode plus. The SCL period generation registers are used to set different kinds of duty cycle implementation for the SCL pulse.

The SDA line which is connected to the whole I²C bus is a bidirectional data line between the master and slave devices used for the transmission and reception of data.

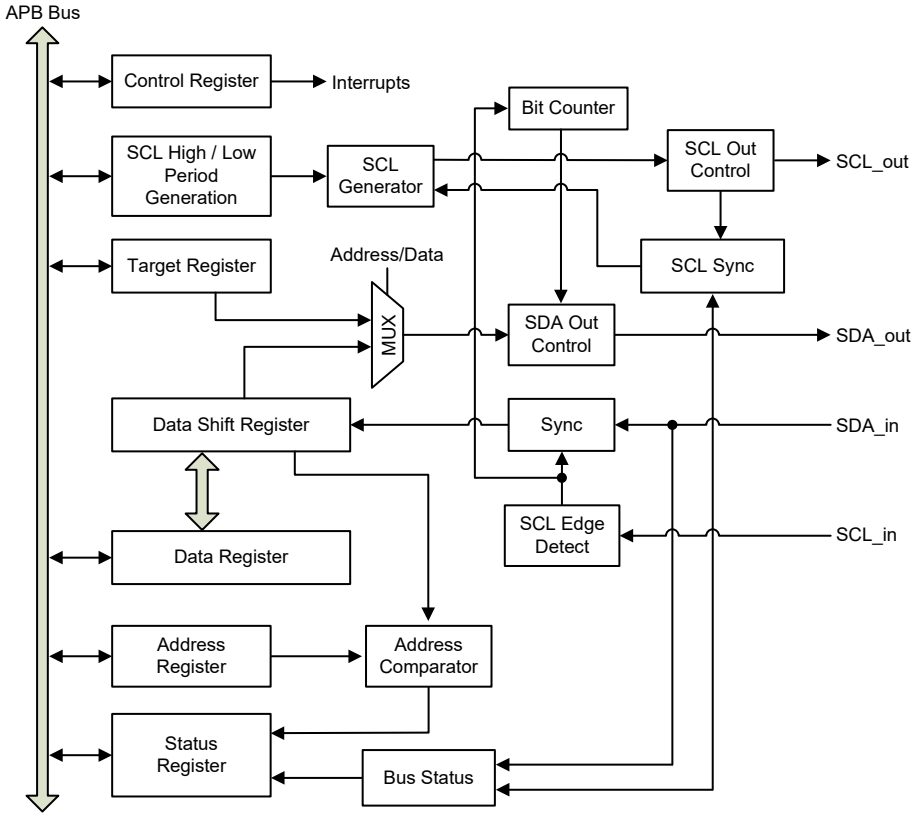


Figure 56. I²C Module Block Diagram

Features

- Two-wire I²C serial interface
  - Serial data line (SDA) and serial clock (SCL)
- Multiple speed modes
  - Standard mode – 100 kHz
  - Fast mode – 400 kHz
  - Fast mode plus – 1 MHz
- Bidirectional data transfer between master and slave
- Supports 7-bit addressing mode and general call addressing
- Supports two 7-bit slave addresses
- Timeout function

Functional Descriptions

Two-Wire Serial Interface

The I²C module has two external lines, the serial data SDA and serial clock SCL lines, to carry information between the interconnected devices connected to the bus. The SCL and SDA lines are both bidirectional and must be connected to a pull-high resistor. When the I²C bus is in the free or idle state, both pins are at a high level to perform the required wired-AND function for multiple connected devices.

START and STOP Conditions

A master device can initialize a transfer by sending a START signal and terminate the transfer with a STOP signal. A START signal is usually referred to as the "S" bit, which is defined as a High to Low transition on the SDA line while the SCL line is high. A STOP signal is usually referred to as the "P" bit, which is defined as a Low to High transition on the SDA line while SCL is high.

A repeated START signal, which is denoted as the "Sr" bit, is functionally identical to the normal START condition. A repeated START signal allows the I²C interface to communicate with another slave device or with the same device but in a different transfer direction without releasing the I²C bus control.

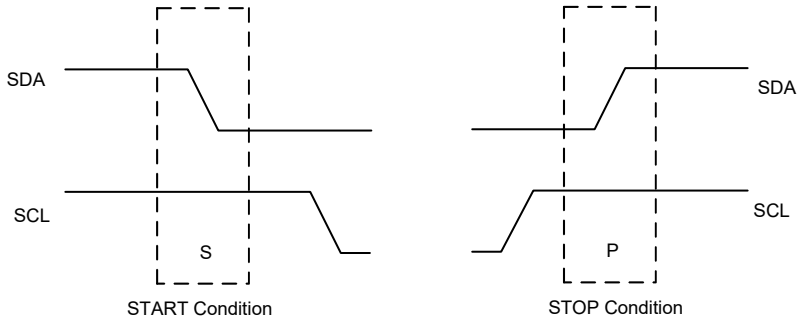


Figure 57. START and STOP Condition

Data Validity

The data on the SDA line must be stable during the high period of the SCL clock. The SDA data state can only be changed when the clock signal on the SCL line is in a low state.

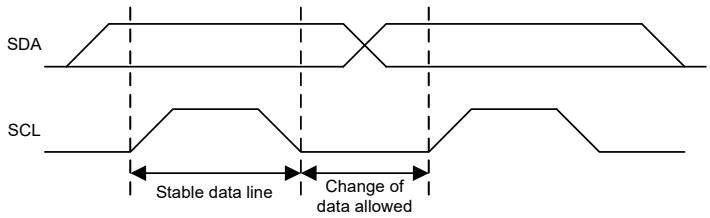


Figure 58. Data Validity

Addressing Format

The I²C interface starts to transfer data after the master device has sent the address to confirm the targeted slave device. The address frame is sent just after the START signal by the master device.

Address Format

The address format is composed of the 7-bit length slave address, which the master device wants to communicate with, a R/$\overline{W}$ bit and an ACK bit. The R/$\overline{W}$ bit defines the direction of the data transfer.

R/$\overline{W}$ = 0 (Write): The master transmits data to the addressed slave.

R/$\overline{W}$ = 1 (Read): The master receives data from the addressed slave.

The slave address can be assigned through the ADDRn field in the I2CADDR register. The ADDRn field indicates the I²C device address. The ADDRnEN bit in the I2CADDR register is used to enabled or disabled the corresponding I²C device address. When the ADDRnEN bit is set, the ADDRn field will be compared with the received address sent from the I²C master device. The slave device sends back the acknowledge bit (ACK) if its slave address matches the transmitted address sent by the master. Note that the ADRSPSEL bit in the I2CCR register is only available to select the time when the ADRS flag is set in the I²C slave mode. It advances the ADRS bit detection timing of the slave address match when the ADRSPSEL bit is set to 1 and then user has more time to process the data transfer to avoid the SCL line be held at a logic low state.

Note that it is forbidden to own the same address for two slave devices.

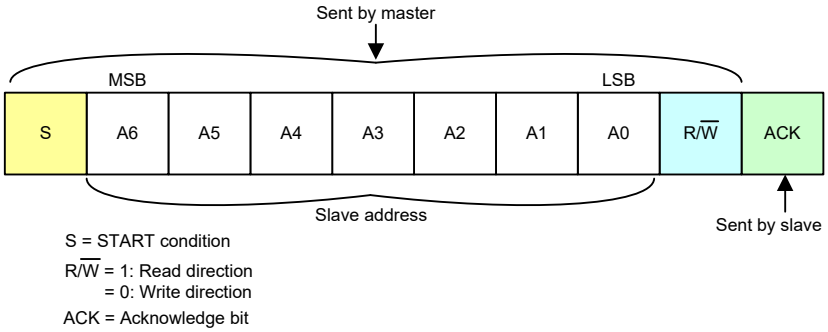


Figure 59. Addressing Mode

Data Transfer and Acknowledge

Once the slave device address has been matched, the data can be transmitted to or received from the slave device according to the transfer direction specified by the R/$\overline{W}$ bit. Each byte is followed by an acknowledge bit on the 9th SCL clock.

If the slave device returns a Not-Acknowledge (NACK) signal to the master device, the master device can generate a STOP signal to terminate the data transfer or generate a repeated START signal to restart the transfer.

If the master device sends a Not-Acknowledge (NACK) signal to the slave device, the slave device should release the SDA line for the master device to generate a STOP signal to terminate the transfer.

General Call Addressing

The general call addressing function can be used to address all the devices connected to the I^{2}C bus. The master device can activate the general call function by writing a value "0x00" into the TAR field and clearing the RWD bit to 0 in the I2CTAR register on the addressing frame.

The device can support the general call addressing function by setting the corresponding enable control bit GCEN to 1. If the GCEN bit is set to 1 to support the general call addressing, the AA bit in the I2CCR register should also be set to 1 to send an acknowledge signal back when the device receives an address frame with a value of 00H. When this condition occurs, the general call flag, GCS, will be set to 1, but the ADRS flag will not be set.

Bus Error

If an unpredictable START or STOP condition occurs when the data is being transferred on the I^{2}C bus, it will be considered as a bus error and the transferring data will be aborted. When a bus error event occurs, the relevant bus error flag BUSERR in the I2CSR register will set to 1 and both the SDA and SCL lines are released. The BUSERR flag should be cleared by writing a 1 to it to initiate the I^{2}C module to an idle state.

Address Snoop

The Address Snoop register, I2CADDSR, is used to monitor the calling address on the I^{2}C bus during the whole data transfer operation no matter if the I^{2}C module operates as a master or a slave device. Note that the I2CADDSR register is a read only register and each calling address on the I^{2}C bus will be stored in the I2CADDSR register automatically even if the I^{2}C device is not addressed.

Operation Mode

The I^{2}C module can operate in the following modes:

- Master Transmitter
- Master Receiver
- Slave Transmitter
- Slave Receiver

The I^{2}C module operates in the slave mode by default. The interface will switch to the master mode automatically after generating a START signal.

Master Transmitter Mode

Start Condition

Users write the target slave device address and communication direction into the I2CTAR register after setting the I2CEN bit in the I2CCR register. The STA flag in the I2CSR register is set by hardware after a start condition occurs. In order to send the following address frame, the STA flag must be cleared to 0 if it has been set to 1. The STA flag is cleared by reading the I2CSR register.

Address Frame

The ADRS flag in the I2CSR register will be set after the address frame is sent by the master device and the acknowledge signal from the address matched slave device is received. In order to send the following data frame, the ADRS flag must be cleared to 0 if it has been set to 1. The ADRS bit is cleared by reading the I2CSR register.

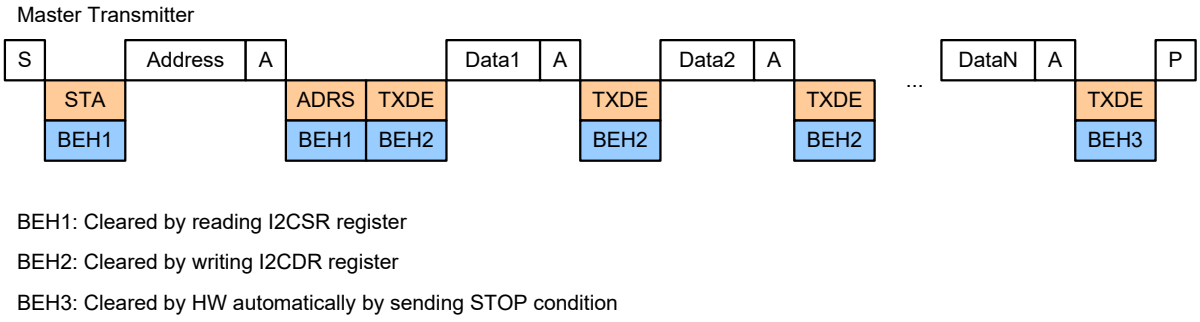
Data Frame

The data to be transmitted to the slave device must be transferred to the I2CDR register.

The TXDE bit in the I2CSR register is set to indicate that the I2CDR register is empty, which results in the SCL line being held at a logic low state. New data must then be transferred to the I2CDR register to continue the data transfer process. Writing a data into the I2CDR register will clear the TXDE flag.

Close / Continue Transmission

After transmitting the last data byte, the STOP bit in the I2CCR register can be set to terminate the transmission or re-assign another slave device by configuring the I2CTAR register to restart a new transfer.



BEH1: Cleared by reading I2CSR register

BEH2: Cleared by writing I2CDR register

BEH3: Cleared by HW automatically by sending STOP condition

Figure 60. Master Transmitter Timing Diagram

Master Receiver Mode

Start Condition

The target slave device address and communication direction must be written into the I2CTAR register. The STA flag in the I2CSR register is set by hardware after a start condition occurs. In order to send the following address frame, the STA flag must be cleared to 0 if it has been set to 1.The STA flag is cleared by reading the I2CSR register.

Address Frame

In the addressing mode: The ADRS flag is set after the address frame is sent by the master device and the acknowledge signal from the address matched slave device is received. In order to receive the following data frame, the ADRS bit must be cleared to 0 if it has been set to 1. The ADRS bit is cleared after reading the I2CSR register.

Data Frame

In the master receiver mode, data is transmitted from the slave device. Once a data is received by the master device, the RXDNE flag in the I2CSR register is set but it will not hold the SCL line. However, if the device receives a complete new data byte and the RXDNE flag has already been set to 1, the RXBF bit in the I2CSR register will be set to 1 and the SCL line will be held at a logic low state. When this situation occurs, data from the I2CDR register should be read to continue the data transfer process. The RXDNE flag can be cleared after reading the I2CDR register.

Close / Continue Transmission

The master device needs to reset the AA bit in the I2CCR register to send an NACK signal to the slave device before the last data byte transfer has been completed. After the last data byte has been received from the slave device, the master device will hold the SCL line at a logic low state following an NACK signal sent by the master device to the slave device. The STOP bit can be set to terminate the data transfer process or re-assign the I2CTAR register to restart a new transfer.

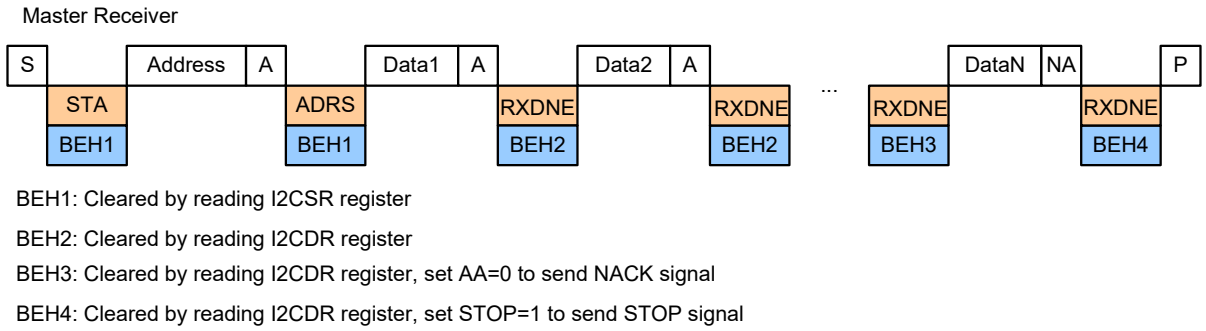


Figure 61. Master Receiver Timing Diagram

Slave Transmitter Mode

Address Frame

In the addressing mode, the ADRS bit in the I2CSR register is set after the slave device receives the calling address which matches with the slave device address. After the ADRS bit has been set to 1, it must be cleared to 0 to continue the data transfer process. The ADRS bit is cleared after reading the I2CSR register.

Data Frame

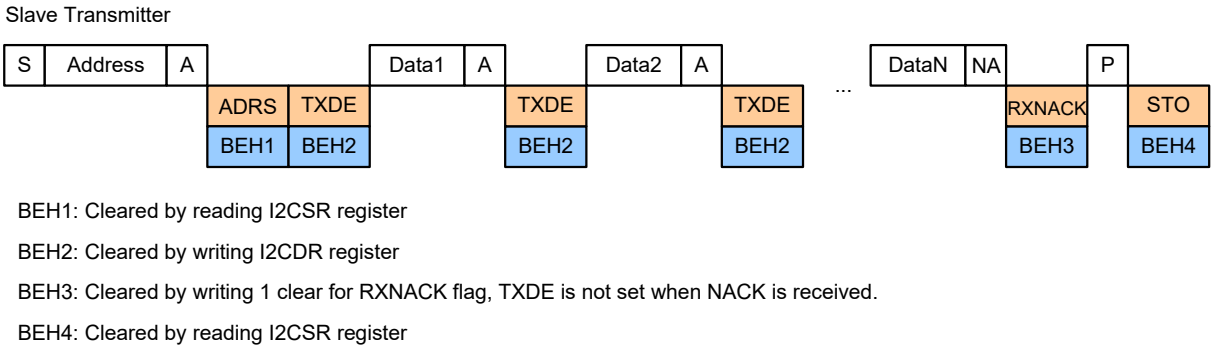
In the Slave transmitter mode, the TXDE bit is set to indicate that the I2CDR is empty, which results in the SCL line being held at a logic low state. New transmission data must then be written into the I2CDR register to continue the data transfer process. Writing a data into the I2CDR register will clear the TXDE bit.

Receive Not-Acknowledge

When the slave device receives a Not-Acknowledge signal, the RXNACK bit in the I2CSR Register is set but it will not hold the SCL line. Writing "1" to RXNACK will clear the RXNACK flag.

STOP Condition

When the slave device detects a STOP condition, the STO bit in the I2CSR register is set to indicate that the I²C interface transmission is terminated. Reading the I2CSR register can clear the STO flag.



BEH1: Cleared by reading I2CSR register

BEH2: Cleared by writing I2CDR register

BEH3: Cleared by writing 1 clear for RXNACK flag, TXDE is not set when NACK is received.

BEH4: Cleared by reading I2CSR register

Figure 62. Slave Transmitter Timing Diagram (ADRSPSEL = 0)

Slave Receiver Mode

Address Frame

The ADRS bit in the I2CSR register is set after the slave device receives the calling address which matches with the slave device address. After the ADRS bit has been set to 1, it must be cleared to 0 to continue the data transfer process. The ADRS flag is cleared after reading the I2CSR register.

Data Frame

In the slave receiver mode, the data is transmitted from the master device. Once a data byte is received by the slave device, the RXDNE flag in the I2CSR register is set but it will not hold the SCL line. However, if the device receives a complete new data byte and the RXDNE bit has been set to 1, the RXBF bit in the I2CSR register will be set to 1 and the SCL line will be held at a logic low state. When this situation occurs, data from the I2CDR register should be read to continue the data transfer process. The RXDNE flag bit can be cleared after reading the I2CDR register.

STOP Condition

When the slave device detects a STOP condition, the STO flag bit in the I2CSR register is set to indicate that the I²C interface transmission is terminated. Reading the I2CSR register can clear the STO flag bit.

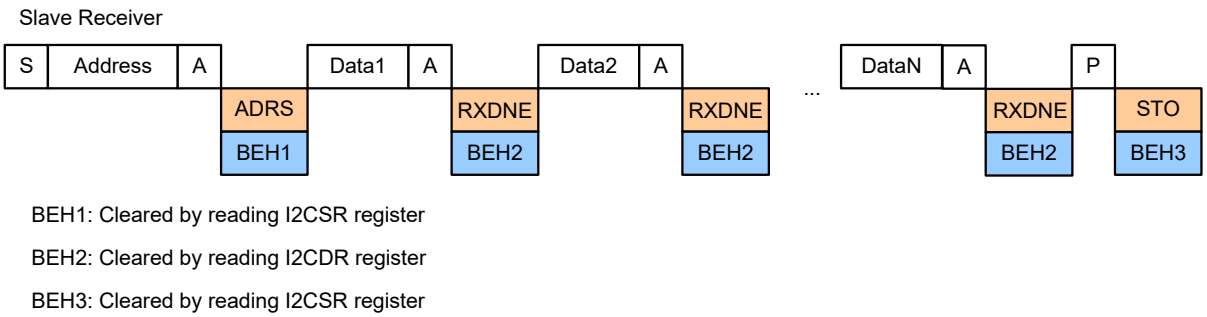


Figure 63. Slave Receiver Timing Diagram (ADRSPSEL = 0)

Conditions of Holding SCL Line

The following conditions will cause the SCL line to be held at a logic low state by hardware resulting in all the I²C transfers being stopped. Data transfer will be continued after the creating conditions are eliminated.

Table 33. Conditions of Holding SCL line

Type	Condition	Description	Eliminating Condition
Flag	TXDE	I²C is used in transmitter mode and I2CDR register needs to have data to transmit. (Note: TXDE won't be asserted after receiving an NACK)	Master case: Writing data to I2CDR register Set TAR Set STOP Slave case: Writing data to I2CDR register
	GCS	I²C is addressed as slave through general call	Reading I2CSR register
	ADRS	Master: I²C address frame is sent and an ACK from slave is returned (Note: Reference Figure 60 and Figure 61) Slave: I²C is addressed as slave device (Note: Reference Figure 62 and Figure 63)	Reading I2CSR register
	STA	Master sends a START signal	Reading I2CSR register
	RXBF	Received a complete new data and meanwhile the RXDNE flag has been set already before.	Reading I2CDR register
Event	Master receives NACK	No matter in address or data frame, once received an NACK signal will hold SCL line in master mode.	Set TAR Set STOP
	Master sends NACK used in receiver mode	Occurred when receiving the last data byte in Master receiver mode (Note: Reference Figure 61, and RXNACK flag won't be asserted in this case)	Set TAR Set STOP

I²C Timeout Function

In order to reduce the occurrence of I²C lockup problem due to the reception of erroneous clock source, a timeout function is provided. If the I²C bus clock source is not received for a certain timeout period, then a corresponding I²C timeout flag will be asserted. This timeout period is determined by a 8-bit down-counting counter with a programmable preload value. The timeout counter is driven by the I²C timeout clock, f_{I2CTO}, which is specified by the timeout prescaler field in the I2CTOUT register. The TOUT field in the I2CTOUT register is used to define the timeout counter preload value. The timeout function is enabled by setting the ENTOUT bit in the I2CCR register. The timeout counter will start to count down from the preloaded value if the ENTOUT bit is set to 1 and one of the following conditions occurs:

- The I²C master module sends a START signal.
- The I²C slave module detects a START signal.
- The RXBF, TXDE, RXDNE, RXNACK, GCS or ADRS flag is asserted.

The timeout counter will stop counting when the ENTOUT bit is cleared. However, the counter will also stop counting when one of the conditions listed as follows occurs:

- The I²C slave module is not addressed.
- The I²C slave module detects a STOP signal.
- The I²C master module sends a STOP signal.
- The BUSERR flag in the I2CSR register is asserted.

If the timeout counter underflows, the corresponding timeout flag, TOUTF, in the I2CSR register will be set to 1 and a timeout interrupt will be generated if the relevant interrupt is enabled.

Register Map

The following table shows the I²C registers and reset values.

Table 34. I²C Register Map

Register	Offset	Description	Reset Value
I2CCR	0x000	I²C Control Register	0x0000_2000
I2CIER	0x004	I²C Interrupt Enable Register	0x0000_0000
I2CADDR	0x008	I²C Address Register	0x0000_0000
I2CSR	0x00C	I²C Status Register	0x0000_0000
I2CSHPGR	0x010	I²C SCL High Period Generation Register	0x0000_0000
I2CSLPGR	0x014	I²C SCL Low Period Generation Register	0x0000_0000
I2CDR	0x018	I²C Data Register	0x0000_0000
I2CTAR	0x01C	I²C Target Register	0x0000_0000
I2CADDSR	0x024	I²C Address Snoop Register	0x0000_0000
I2CTOUT	0x028	I²C Timeout Register	0x0000_0000

Register Descriptions

I²C Control Register – I2CCR

This register specifies the corresponding I²C function enable control

Offset: 0x000

Reset value: 0x0000_2000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
		SEQFILTER	COMBFILTEREN	ENTOUT	Reserved			
Type/Reset	RW 0	RW 0	RW 1	RW 0				
	7	6	5	4	3	2	1	0
	Reserved			ADRSPSEL	I2CEN	GCEN	STOP	AA
Type/Reset				RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:14]	SEQFILTER	SDA or SCL Input Sequential Filter Configuration Bits 00: Sequential filter is disabled 01: 1 PCLK glitch filter 1x: 2 PCLK glitch filter Note: This setting would affect the frequency of SCL. Details are described in I2CSLPGR register.
[13]	COMBFILTEREN	SDA or SCL Input Combinational Filter Enable Bit 0: Combinational filter is disabled 1: Combinational filter is enabled
[12]	ENTOUT	I²C Timeout Function Enable Control 0: Timeout Function is disabled 1: Timeout Function is enabled This bit is used to enable or disable the I²C timeout function. It is recommended that users have to properly configure the PSC and TOUT fields in the I2CTOUT register before the timeout counter starts to count by setting the ENTOUT bit to 1.
[4]	ADRSPSEL	Address Receive Match Position Selection 0: ADRS flag is set at the SCL falling edge of the ACK bit in the Slave address frame 1: ADRS flag is set at the SCL falling edge of the R/W bit in the Slave address frame This bit is only available to select the time when the ADRS flag is set in the I²C slave mode. Because the ADRS bit has been set to 1, it must be cleared to 0 to continue the data transfer process. It advances the ADRS bit detection timing of the slave address match when the bit ADRSPSEL is set to 1 and then user has more time to process the data transfer to avoid the SCL line be held at a logic low state.

Bits	Field	Descriptions
[3]	I2CEN	I²C Interface Enable 0: I²C interface is disabled 1: I²C interface is enabled
[2]	GCEN	General Call Enable 0: General call is disabled 1: General call is enabled When the device receives the calling address with a value of 0x00 and if both the GCEN and the AA bits are set to 1, then the I²C interface is addressed as a slave and the GCS bit in the I2CSR register is set to 1.
[1]	STOP	STOP Condition Control 0: No action 1: Send a STOP condition in master mode This bit is set to 1 by software to generate a STOP condition and automatically cleared to 0 by hardware. The STOP bit is only available for the master device.
[0]	AA	Acknowledge Bit 0: Send a Not-Acknowledge (NACK) signal after a byte is received 1: Send an Acknowledge (ACK) signal after a byte is received

I²C Interrupt Enable Register – I2CIER

This register specifies the corresponding I²C interrupt enable bits.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved					RXBFIE	TXDEIE	RXDNEIE
Type/Reset						RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved				TOUTIE	BUSERRIE	RXNACKIE	Reserved
Type/Reset					RW 0	RW 0	RW 0	
	7	6	5	4	3	2	1	0
	Reserved				GCSIE	ADRSIE	STOIE	STAIE
Type/Reset					RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[18]	RXBFIE	RX Buffer Full Interrupt Enable Bit 0: Interrupt is disabled 1: Interrupt is enabled
[17]	TXDEIE	Data Register Empty Interrupt Enable Bit in Transmitter Mode 0: Interrupt is disabled 1: Interrupt is enabled

Bits	Field	Descriptions
[16]	RXDNEIE	Data Register Not Empty Interrupt Enable Bit in Received Mode 0: Interrupt is disabled 1: Interrupt is enabled
[11]	TOUTIE	Timeout Interrupt Enable Bit 0: Interrupt is disabled 1: Interrupt is enabled
[10]	BUSERRIE	Bus Error Interrupt Enable Bit 0: Interrupt is disabled 1: Interrupt is enabled
[9]	RXNACKIE	Received Not-Acknowledge Interrupt Enable Bit 0: Interrupt is disabled 1: Interrupt is enabled
[3]	GCSIE	General Call Slave Interrupt Enable Bit 0: Interrupt is disabled 1: Interrupt is enabled
[2]	ADRSIE	Slave Address Match Interrupt Enable Bit 0: Interrupt is disabled 1: Interrupt is enabled
[1]	STOIE	STOP Condition Detected Interrupt Enable Bit 0: Interrupt is disabled 1: Interrupt is enabled The bit is used for the I^2C slave mode only.
[0]	STAIE	START Condition Transmit Interrupt Enable Bit 0: Interrupt is disabled 1: Interrupt is enabled The bit is used for the I^2C master mode only.

I²C Address Register – I2CADDR

This register specifies the I²C device address.

Offset: 0x008

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	ADDR1EN	Reserved						
Type/Reset	RW 0							
	23	22	21	20	19	18	17	16
	Reserved	ADDR1						
Type/Reset		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	ADDR0EN	Reserved						
Type/Reset	RW 0							
	7	6	5	4	3	2	1	0
	Reserved	ADDR0						
Type/Reset		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[31]	ADDR1EN	Device Address 1 Enable Bit 0: Device Address 1 is disabled 1: Device Address 1 is enabled
[22:16]	ADDR1	Device Address 1 The ADDR1 field indicates the I²C device address. When the ADDR1EN bit is set, the ADDR1 field will be compared with the received address sent from the I²C master device
[15]	ADDR0EN	Device Address 0 Enable Bit 0: Device Address 0 is disabled 1: Device Address 0 is enabled
[6:0]	ADDR0	Device Address 0 The ADDR0 field indicates the I²C device address. When the ADDR0EN bit is set, the ADDR0 field will be compared with the received address sent from the I²C master device.

I²C Status Register – I2CSR

This register contains the I²C operation status.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
		Reserved	TXNRX	MASTER	BUSBUSY	RXBF	TXDE	RXDNE
Type/Reset			RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	15	14	13	12	11	10	9	8
	Reserved				TOUTF	BUSERR	RXNACK	Reserved
Type/Reset					WC 0	WC 0	WC 0	
	7	6	5	4	3	2	1	0
	Reserved				GCS	ADRS	STO	STA
Type/Reset					RC 0	RC 0	RC 0	RC 0

Bits	Field	Descriptions
[21]	TXNRX	Transmitter / Receiver Mode 0: Receiver mode 1: Transmitter mode Read only bit.
[20]	MASTER	Master Mode 0: I²C is in the slave mode or idle 1: I²C is in the master mode The I²C interface is switched as a master device on the I²C bus when the I2CTAR register is assigned and the I²C bus is idle. The MASTER bit is cleared by hardware when software disables the I²C bus by clearing the I2CEN bit to 0 or sends a STOP condition to the I²C bus or the bus error is detected. This bit is set and cleared by hardware and is a read only bit.
[19]	BUSBUSY	Bus Busy 0: I²C bus is idle 1: I²C bus is busy The I²C interface hardware starts to detect the I²C bus status if the interface is enabled by setting the I2CEN bit to 1. It is set to 1 when the SDA or SCL signal is detected to have a logic low state and cleared when a STOP condition is detected.
[18]	RXBF	Buffer Full Flag in Receiver Mode 0: Data buffer is not full 1: Data buffer is full This bit is set when the data register I2CDR has already stored a data byte and meanwhile the data shift register also has been received a complete new data byte. The RXBF bit is cleared by software reading the I2CDR register.

Bits	Field	Descriptions
[17]	TXDE	Data Register Empty in Transmitter Mode 0: Data register I2CDR is not empty 1: Data register I2CDR is empty This bit is set when the I2CDR register is empty in the Transmitter mode. Note that the TXDE bit will be set after the address frame is being transmitted to inform that the data to be transmitted should be loaded into the I2CDR register. The TXDE bit is cleared by software writing data to the I2CDR register in both the master and slave mode or cleared automatically by hardware after setting the STOP signal to terminate the data transfer or setting the I2CTAR register to restart a new data transfer in the master mode.
[16]	RXDNE	Data Register Not Empty in Receiver Mode 0: Data register I2CDR is empty 1: Data register I2CDR is not empty This bit is set when the I2CDR register is not empty in the receiver mode. The RXDNE bit is cleared by software reading the data byte from the I2CDR register.
[11]	TOUTF	Timeout Counter Underflow Flag 0: No timeout counter underflow has occurred 1: Timeout counter underflow has occurred Writing “1” to this bit will clear the TOUTF flag.
[10]	BUSERR	Bus Error Flag 0: No bus error has occurred 1: Bus error has occurred This bit is set by hardware when the I^2C interface detects a misplaced START or STOP condition in a transfer process. Writing a “1” to this bit will clear the BUSERR flag. In Master Mode: Once the Bus Error event occurs, both the SDA and SCL lines are released by hardware and the BUSERR flag is asserted. The application software has to clear the BUSERR flag before the next address byte is transmitted. In Slave Mode: Once a misplaced START or STOP condition has been detected by the slave device, the software must clear the BUSERR flag before the next address byte is received.
[9]	RXNACK	Received Not-Acknowledge Flag 0: Acknowledge is returned from receiver 1: Not-Acknowledge is returned from receiver The RXNACK bit indicates that the Not-Acknowledge signal is received in master or slave transmitter mode. Writing “1” to this bit will clear the RXNACK flag.
[3]	GCS	General Call Slave Flag 0: No general call slave occurs 1: I^2C interface is addressed by a general call command When the I^2C interface receives an address with a value of 0x00, if both the GCEN and the AA bit are set to 1, then it is switched as a general call slave. This flag is cleared automatically after being read.

Bits	Field	Descriptions
[2]	ADRS	Address Transmit (master mode) / Address Receive (slave mode) Flag Address Sent in Master Mode: 0: Address frame has not been transmitted 1: Address frame has been transmitted For the addressing mode, this bit is set after the master device receives the address frame acknowledge bit sent from the slave device. Address Matched in Slave Mode: 0: I^2C interface is not addressed 1: I^2C interface is addressed as slave When the I^2C interface has received the calling address that matches the address defined in the I2CADDR register together with the AA bit being set to 1 in the I2CCR register, it will be switched to a slave mode. This flag is cleared automatically after the I2CSR register has been read.
[1]	STO	STOP Condition Detected Flag 0: No STOP condition is detected 1: STOP condition is detected in slave mode This bit is only available for the slave mode and is cleared automatically after the I2CSR register is read.
[0]	STA	START Condition Transmit 0: No START condition is detected 1: START condition is transmitted in master mode This bit is only available for the master mode and is cleared automatically after the I2CSR register is read.

I²C SCL High Period Generation Register – I2CSHPGR

This register specifies the I²C SCL clock high period interval.

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	SHPG							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	SHPG	SCL Clock High Period Generation High period duration setting $SCL_{HIGH} = T_{PCLK} \times (SHPG + d)$ where T_{PCLK} is the APB bus peripheral clock (PCLK) period, and d value depends on the setting of SEQFILTER in the I²C Control Register (I2CCR). If SEQFILTER = 00, d = 6 If SEQFILTER = 01, d = 8 If SEQFILTER = 10 or 11, d = 9

I²C SCL Low Period Generation Register – I2CSLPGR

This register specifies the I²C SCL clock low period interval.

Offset: 0x014

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	SLPG							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	SLPG	SCL Clock Low Period Generation Low period duration setting SCL_{LOW} = T_{PCLK} × (SLPG + d) where TPCLK is the APB bus peripheral clock (PCLK) period, and d value depends on the setting of SEQFILTER in the I²C Control Register (I2CCR). If SEQFILTER = 00, d = 6 If SEQFILTER = 01, d = 8 If SEQFILTER = 10 or 11, d = 9

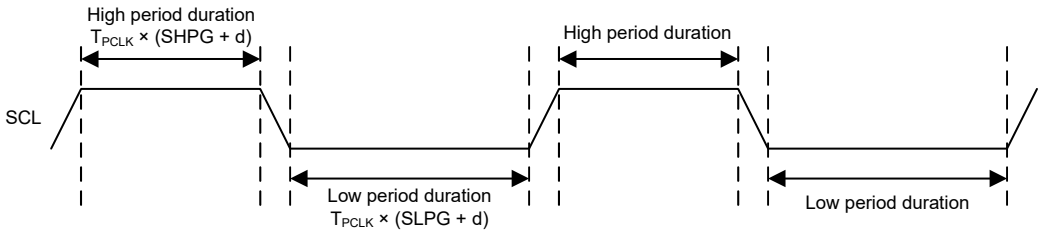


Figure 64. SCL Timing Diagram

Table 35. I²C Clock Setting Example

I²C Clock	T_{SCL} = T_{PCLK} × [(SHPG + d) + (SLPG + d)] (where d = 6) SHPG + SLPG Value at PCLK	
	8 MHz	16 MHz
100 kHz (Standard Mode)	68	148
400 kHz (Fast Mode)	8	28
1 MHz (Fast Mode Plus)	N/A	4

I²C Data Register – I2CDR

This register specifies the data to be transmitted or received by the I²C module.

Offset: 0x018

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	DATA							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	DATA	I²C Data Register For the transmitter mode, a data byte which is transmitted to a slave device can be assigned to these bits. The TXDE flag is cleared if the application software assigns new data to the I2CDR register. For the receiver mode, a data byte is received bit by bit from MSB to LSB through the I²C interface and stored in the data shift register. Once the acknowledge bit is given, the data shift register value is delivered into the I2CDR register if the RXDNE flag is equal to 0.

I²C Target Register – I2CTAR

This register specifies the target device address to be communicated.

Offset: 0x01C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved					RWD		Reserved
Type/Reset						RW 0		
	7	6	5	4	3	2	1	0
	Reserved	TAR						
Type/Reset		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[10]	RWD	Read or Write Direction 0: Write direction to target slave address 1: Read direction from target slave address
[6:0]	TAR	Target Slave Address The I²C interface will assign a START signal and send a target slave address automatically once the data is written to this register. When the system wants to send a repeated START signal to the I²C bus, it is suggested to set the I2CTAR register after a byte transfer is completed. It is not allowed to set TAR in the address frame.

I²C Address Snoop Register – I2CADDSR

This register is used to indicate the address frame value appeared on the I²C bus.

Offset: 0x024

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved	ADDSR						
Type/Reset		RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

Bits	Field	Descriptions
[6:0]	ADDSR	Address Snoop Once the I2CEN bit is enabled, the calling address value on the I²C bus will automatically be loaded into this ADDSR field.

I²C Timeout Register – I2CTOUT

This register specifies the I²C timeout counter preload value and clock prescaler ratio.

Offset: 0x028

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved						PSC	
Type/Reset							RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	TOUT							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[17:16]	PSC	I²C Timeout Counter Prescaler Selection This PSC field is used to specify the I²C timeout counter clock frequency, f_{I2CTO}. The timeout clock frequency is obtained using the following formula. $f_{I2CTO} = \frac{f_{PCLK}}{2^{PSC}}$ PSC = 0 → $f_{I2CTO} = f_{PCLK} / 2^0 = f_{PCLK}$ PSC = 1 → $f_{I2CTO} = f_{PCLK} / 2^1 = f_{PCLK} / 2$ PSC = 2 → $f_{I2CTO} = f_{PCLK} / 2^2 = f_{PCLK} / 4$ PSC = 3 → $f_{I2CTO} = f_{PCLK} / 2^3 = f_{PCLK} / 8$
[7:0]	TOUT	I²C Timeout Counter Preload Value The TOUT field is used to define the counter preloaded value. The counter value is reloaded as any one of the following conditions occurs: 1. The RXBF, TXDE, RXDNE, RXNACK, GCS or ADRS flag in the I2CSR register is asserted. 2. The I²C master module sends a START signal. 3. The I²C slave module detects a START signal. The counter stops counting as any one of the following conditions occurs: 1. The I²C slave device is not addressed. 2. The I²C master module sends a STOP signal. 3. The I²C slave module detects a STOP signal. 4. The BUSERR flag in the I2CSR register is asserted.

18 Serial Peripheral Interface (SPI)

Introduction

The Serial Peripheral Interface, SPI, provides an SPI protocol data transmit and receive functions in both master or slave mode. The SPI interface uses 4 pins, among which are the serial data input and output lines SPI_MISO and SPI_MOSI, the clock line SPI_SCK, and the slave select line SPI_SEL. One SPI device acts as a master who controls the data flow using the SEL and SCK signals to indicate the start of the data communication and the data sampling rate. To receive the data bits, the streamlined data bits which range from 1 bit to 8 bits specified by the DFL field in the SPICR1 register are latched on a specific clock edge and stored in the data register or in the RX FIFO. Data transmission is carried out in a similar way but with the reverse sequence.

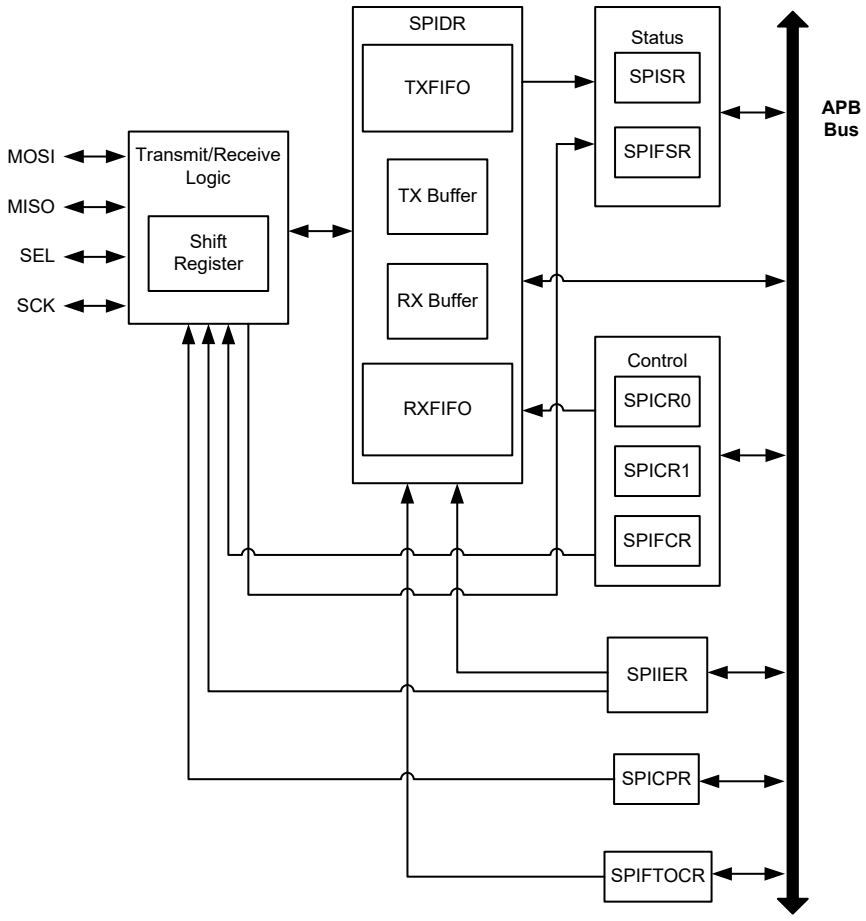


Figure 65. SPI Block Diagram

Features

- Master or slave mode
- Master mode speed up to $f_{PCLK}/2$
- Slave mode speed up to $f_{PCLK}/3$
- Programmable data frame length up to 8 bits
- FIFO Depth: 4 levels
- MSB or LSB first shift selection
- Programmable slave select high or low active polarity
- Three error flags with individual interrupt
  - Read overrun
  - Write collision
  - Slave abort

Functional Descriptions

Master Mode

Each data frame can range from 1 to 8 bits in data length. The first bit of the transmitted data can be either an MSB or LSB determined by the FIRSTBIT bit in the SPICR1 register. The SPI module is configured as a master or a slave by setting the MODE bit in the SPICR1 register. When the MODE bit is set, the SPI module is configured as a master and will generate the serial clock on the SPI_SCK pin. The data stream will transmit data in the shift register to the SPI_MOSI pin on the serial clock edge. The SPI_SEL pin is active during the full data transmission. When the SELAP bit in the SPICR1 register is set, the SPI_SEL pin is active high during the complete data transactions. When the SELM bit in the SPICR1 register is set, the SPI_SEL pin will be driven by the hardware automatically and the time interval between the active SEL edge and the first edge of SCK is equal to half an SCK period.

Slave Mode

In the slave mode, the SPI_SCK pin acts as an input pin and the serial clock will be derived from the external master device. The SPI_SEL pin also acts as an input. When the SELAP bit is cleared to 0, the SEL signal is active low during the full data stream reception. When the SELAP bit is set to 1, the SEL signal will be active high during the full data stream reception.

Note: For the slave mode, the APB clock, known as f_{PCLK}, must be at least 3 times faster than the external SCK clock input frequency.

SPI Serial Frame Format

The SPI interface format is based on the Clock Polarity, CPOL, and the Clock Phase, CPHA, configurations.

- Clock Polarity Bit – CPOL
  When the Clock Polarity bit is cleared to 0, the SCK line idle state is low. When the Clock Polarity bit is set to 1, the SCK line idle state is high.
- Clock Phase Bit – CPHA
  When the Clock Phase bit is cleared to 0, the data is sampled on the first SCK clock transition. When the Clock Phase bit is set to 1, the data is sampled on the second SCK clock transition.

There are four formats contained in the SPI interface. The accompanying table shows how to configure these formats by setting the FORMAT field in the SPICR1 register.

Table 36. SPI Interface Format Setup

FORMAT [2:0]	CPOL	CPHA
001	0	0
010	0	1
110	1	0
101	1	1
Others	Reserved	

CPOL = 0, CPHA = 0

In this format, the received data is sampled on the SCK line rising edge while the transmitted data is changed on the SCK line falling edge. In the master mode, the first bit is driven when data is written into the SPIDR Register. In the slave mode, the first bit is driven when the SEL signal goes to an active level. The accompanying figure shows the single byte data transfer timing of this format.

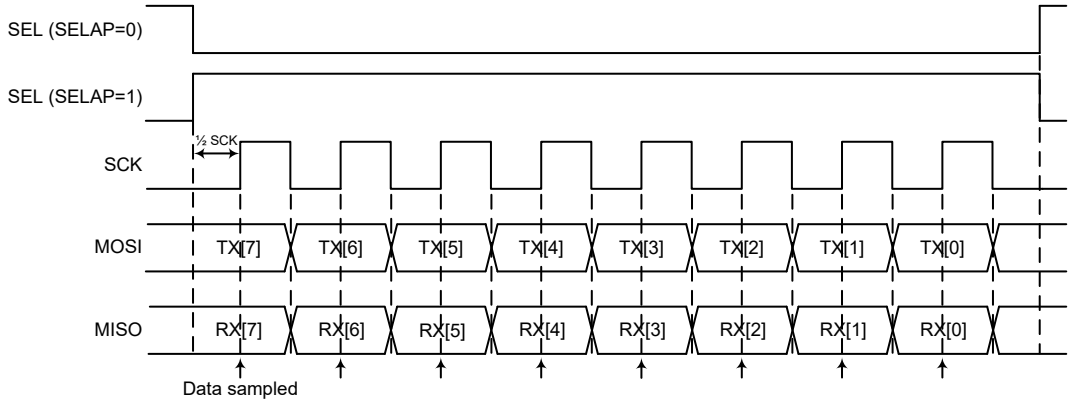


Figure 66. SPI Single Byte Transfer Timing Diagram – CPOL = 0, CPHA = 0

The accompanying figure shows the continuous data transfer timing diagram of this format. Note that the SEL signal must change to an inactive level between each data frame.

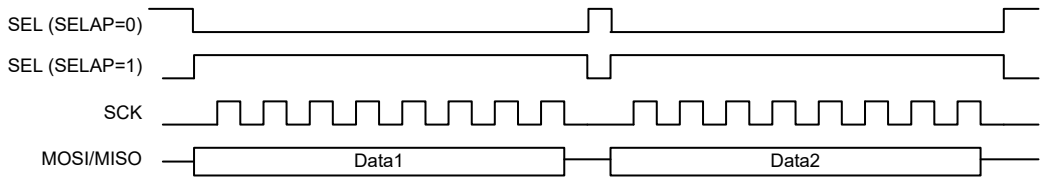


Figure 67. SPI Continuous Data Transfer Timing Diagram – CPOL = 0, CPHA = 0

CPOL = 0, CPHA = 1

In this format, the received data is sampled on the SCK line falling edge while the transmitted data is changed on the SCK line rising edge. In the master mode, the first bit is driven when data is written into the SPIDR register. In the slave mode, the first bit is driven at the first SCK clock rising edge. The accompanying figure shows the single data byte transfer timing.

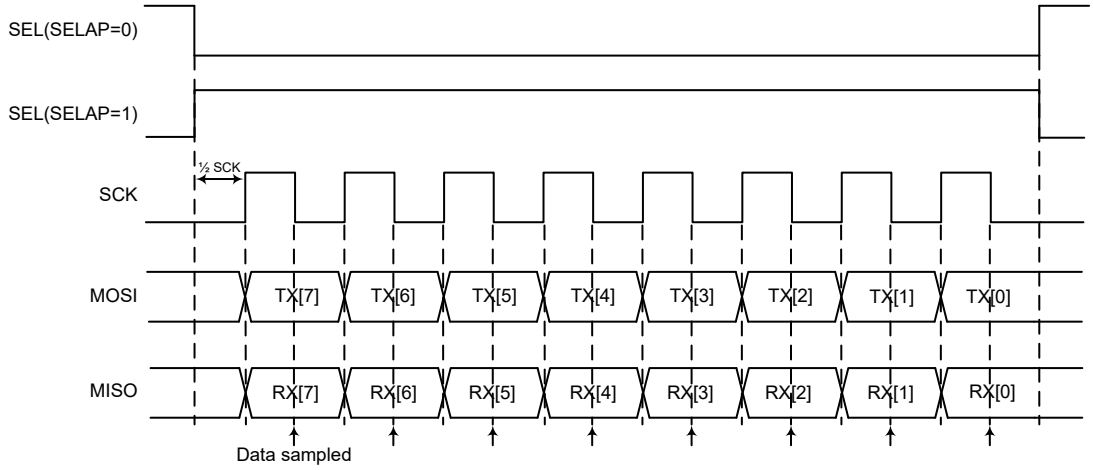


Figure 68. SPI Single Byte Transfer Timing Diagram – CPOL = 0, CPHA = 1

The accompanying figure shows the continuous data transfer diagram timing. Note that the SEL signal must remain active until the last data transfer has completed.

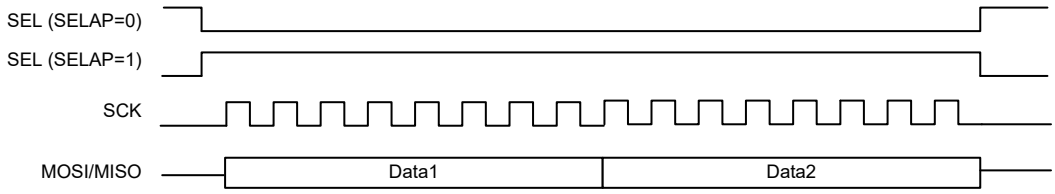


Figure 69. SPI Continuous Transfer Timing Diagram – CPOL = 0, CPHA = 1

CPOL = 1, CPHA = 0

In this format, the received data is sampled on the SCK line falling edge while the transmitted data is changed on the SCK line rising edge. In the master mode, the first bit is driven when data is written into the SPIDR register. In the slave mode, the first bit is driven when the SEL signal changes to an active level. The accompanying figure shows the single byte transfer timing of this format.

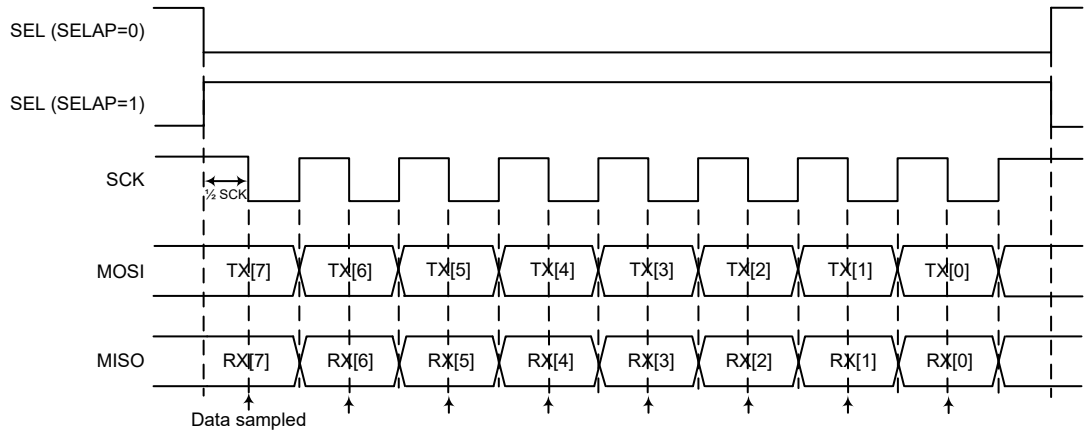


Figure 70. SPI Single Byte Transfer Timing Diagram – CPOL = 1, CPHA = 0

The accompanying figure shows the continuous data transfer timing of this format. Note that the SEL signal must change to an inactive level between each data frame.

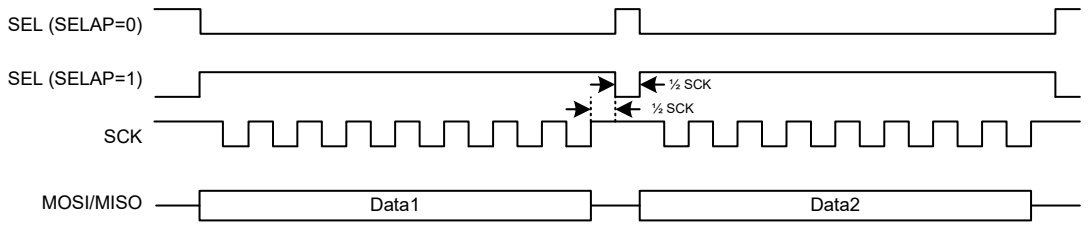


Figure 71. SPI Continuous Transfer Timing Diagram – CPOL = 1, CPHA = 0

CPOL = 1, CPHA = 1

In this format, the received data is sampled on the SCK line rising edge while the transmitted data is changed on the SCK line falling edge. In the master mode, the first bit is driven when data is written into the SPIDR register. In the slave mode, the first bit is driven at the first SCK falling edge. The accompanying figure shows the single byte transfer timing of this format.

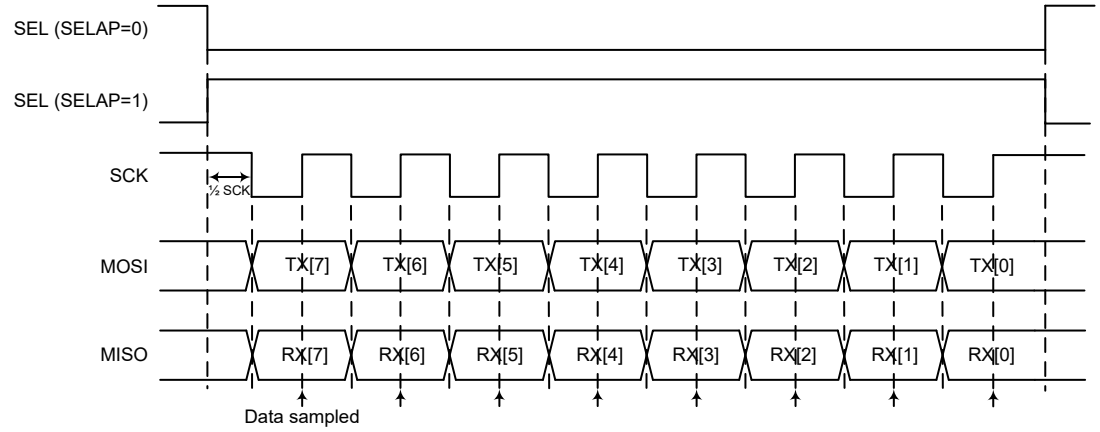


Figure 72. SPI Single Byte Transfer Timing Diagram – CPOL = 1, CPHA = 1

The accompanying figure shows the continuous data transfer timing of this format. Note that the SEL signal must remain active until the last data transfer has completed.

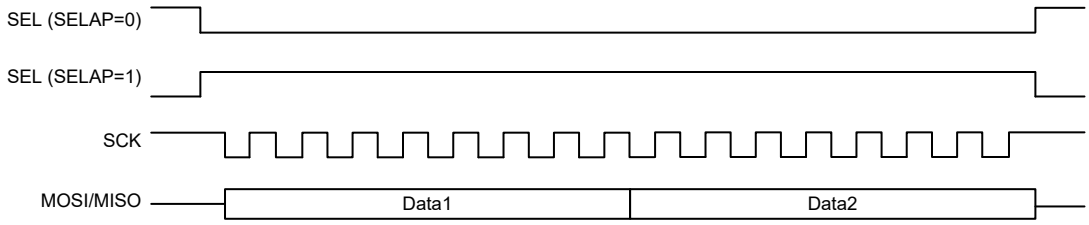


Figure 73. SPI Continuous Transfer Timing Diagram – CPOL = 1, CPHA = 1

Status Flags

TX Buffer Empty – TXBE

This TXBE flag is set when the TX buffer is empty in the non-FIFO mode or when the TX FIFO data length is equal to or less than the TX FIFO threshold level as defined by the TXFTLS field in the SPIFCR register in the FIFO mode. The following data to be transmitted can then be loaded into the buffer again. After this, the TXBE flag will be reset when the TX buffer already contains new data in the non-FIFO mode or when the TX FIFO data length is greater than the TX FIFO threshold level determined by the TXFTLS field in FIFO mode.

Transmission Register Empty – TXE

This TXE flag is set when both the TX buffer and the TX shift registers are empty. It will be reset when the TX buffer or the TX shift register contains new transmitted data.

RX Buffer Not Empty – RXBNE

This RXBNE flag is set when there is valid received data in the RX buffer in the non-FIFO mode or the RX FIFO data length is equal to or greater than the RX FIFO threshold level as defined by the RXFTLS field in the SPIFCR register in the SPI FIFO mode. This flag will be automatically cleared by hardware when the received data have been read out from the RX buffer totally in the non-FIFO mode or when the RX FIFO data length is less than the RX FIFO threshold level set in the RXFTLS field.

Time Out Flag – TO

The time out function is only available in the SPI FIFO mode and is disabled by loading a zero value into the TOC field in the Time Out Counter register. The time out counter will start counting if the SPI RX FIFO is not empty, once data is read from the SPIDR register or new data is received, the time out counter will be reset to 0 and count again. When the time out counter value is equal to the value specified by the TOC field in the SPIFTOCR register, the TO flag will be set. The flag is cleared by writing 1 to this bit.

Write Collision – WC

The following conditions will assert the Write Collision Flag.

- The FIFOEN bit in the SPIFCR register is cleared
  The write collision flag is asserted when new data is written into the SPIDR register while both the TX buffer and the shift register are already full. Any new data written into the TX buffer will be lost.
- The FIFOEN bit in the SPIFCR register is set
  The write collision flag is asserted to indicate that new data is written into the SPIDR register while both the TX FIFO and the TX shift register are already full. Any new data written into the TX FIFO will be lost.

Read Overrun – RO

- The FIFOEN bit in the SPIFCR register is cleared
  The read overrun flag is asserted to indicate that both the RX shift register and the RX buffer are already full, if one more data is received. This will result in the newly received data not being shifted into the SPI shift register. As a result the latest received data will be lost.
- The FIFOEN bit in the SPIFCR register is set
  The read overrun flag is set to indicate that the RX shift register and the RX FIFO are both full, if one more data is received. This means that the latest received data cannot be shifted into the SPI shift register. As a result the latest received data will be lost.

Slave Abort – SA

In the SPI slave mode, the slave abort flag is set to indicate that the SPI_SEL pin suddenly changed to an inactive state during the reception of a data frame transfer. The data frame length is set by the DFL field in the SPICR1 register.

Register Map

The following table shows the SPI registers and reset values.

Table 37. SPI Register Map

Register	Offset	Description	Reset Value
SPICR0	0x000	SPI Control Register 0	0x0000_0000
SPICR1	0x004	SPI Control Register 1	0x0000_0000
SPIIER	0x008	SPI Interrupt Enable Register	0x0000_0000
SPICPR	0x00C	SPI Clock Prescaler Register	0x0000_0001
SPIDR	0x010	SPI Data Register	0x0000_0000
SPISR	0x014	SPI Status Register	0x0000_0003
SPIFCR	0x018	SPI FIFO Control Register	0x0000_0000
SPIFSR	0x01C	SPI FIFO Status Register	0x0000_0000
SPIFTOCR	0x020	SPI FIFO Time Out Counter Register	0x0000_0000

Register Descriptions

SPI Control Register 0 – SPICR0

This register specifies the SEL control and the SPI enable bits.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	SELHT				GUADT			
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	GUADTEN	Reserved		SSELC	Reserved			SPIEN
Type/Reset	RW 0			RW 0				RW 0

Bits	Field	Descriptions
[15:12]	SELHT	Chip Select Hold Time 0x0: 1/2 SCK 0x1: 1 SCK 0x2: 3/2 SCK 0x3: 2 SCK .... Note that SELHT is for master mode only.

Bits	Field	Descriptions
[11:8]	GUADT	Guard Time GUADTEN = 1 0x0: 1 SCK 0x1: 2 SCK 0x2: 3 SCK ... Note that GUADT is for master mode only.
[7]	GUADTEN	Guard Time Enable 0: Guard Time is 1/2 SCK 1: When this bit is set, guard time can be controlled by GUADT Note that GUADTEN is for master mode only.
[4]	SSELC	Software Slave Select Control 0: Set the SEL output to an inactive state 1: Set the SEL output to an active state The application software can set the SEL output to an active or inactive state by configuring the SSELC bit. The active level is configured by the SELAP bit in the SPICR1 register. Note that the SELM bit is cleared to 0 for controlling the SEL signal by software. Otherwise, the SSELC bit has no effect.
[0]	SPIEN	SPI Enable 0: SPI interface is disabled 1: SPI interface is enabled

SPI Control Register 1 – SPICR1

This register specifies the SPI parameters including the data length, the transfer format, the SEL active polarity/mode, the LSB/MSB control and the master/slave mode.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	SCKDUTY	MODE	SELM	FIRSTBIT	SELAP	FORMAT		
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	Reserved					DFL		
Type/Reset						RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15]	SCKDUTY	SCK Pulse High/Low Duty 0: High Duty is longer than Low Duty 1: High Duty is shorter than Low Duty The frequency of SCK is calculated by f_{PCLK}/n. When n is odd, the SCK high/low duty is not 50%. In this case, the ratio of the high/low duty can be selected by this control bit.
[14]	MODE	Master or Slave Mode 0: Slave mode 1: Master mode
[13]	SELM	Slave Select Mode 0: SEL signal is controlled by software – asserted or de-asserted by the SSELC bit 1: SEL signal is controlled by hardware – generated automatically by the SPI hardware Note that the SELM bit is available for master mode only, i.e., MODE = 1.
[12]	FIRSTBIT	LSB or MSB Transmitted First 0: MSB is transmitted first 1: LSB is transmitted first
[11]	SELAP	Slave Select Active Polarity 0: SEL signal is active low 1: SEL signal is active high

Bits	Field	Descriptions
[10:8]	FORMAT	SPI Data Transfer Format These three bits are used to determine the data transfer format of the SPI interface. CPOL: Clock Polarity 0: SCK Idle state is low 1: SCK Idle state is high CPHA: Clock Phase 0: Data is captured on the first SCK clock edge 1: Data is captured on the second SCK clock edge
[2:0]	DFL	Data Frame Length Selects the data transfer frame from 1 bit to 8 bits. 001: 1 bit 010: 2 bits 011: 3 bits 100: 4 bits 101: 5 bits 110: 6 bits 111: 7 bits 000: 8 bits

FORMAT [2:0]	CPOL	CPHA
001	0	0
010	0	1
110	1	0
101	1	1
Others	Reserved	

SPI Interrupt Enable Register – SPIIER

This register contains the corresponding SPI interrupt enable control bit.

Offset: 0x008

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								

	7	6	5	4	3	2	1	0
	TOIEN	SAIEN	Reserved	ROIEN	WCIEN	RXBNEIEN	TXEIEN	TXBEIEN
Type/Reset	RW 0	RW 0		RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7]	TOIEN	Time Out Interrupt Enable 0: Disable 1: Enable
[6]	SAIEN	Slave Abort Interrupt Enable 0: Disable 1: Enable
[4]	ROIEN	Read Overrun Interrupt Enable 0: Disable 1: Enable
[3]	WCIEN	Write Collision Interrupt Enable 0: Disable 1: Enable
[2]	RXBNEIEN	RX Buffer Not Empty Interrupt Enable 0: Disable 1: Enable An interrupt is generated when the RXBNE flag is set and RXBNEIEN is set. In the FIFO mode, the interrupt being generated depends upon the RX FIFO trigger level setting.
[1]	TXEIEN	Transmission Register Empty Interrupt Enable 0: Disable 1: Enable The transmission register empty interrupt request will be generated when the TXE flag and the TXEIEN bit are set.
[0]	TXBEIEN	TX Buffer Empty Interrupt Enable 0: Disable 1: Enable The TX buffer empty interrupt request will be generated when the TXBE flag and the TXBEIEN bit are set. In the FIFO mode, the interrupt request being generated depends upon the TX FIFO trigger level setting.

SPI Clock Prescaler Register – SPICPR

This register specifies the SPI clock prescaler ratio.

Offset: 0x00C

Reset value: 0x0000_0001

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	CP							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 1

Bits	Field	Descriptions
[7:0]	CP	SPI Clock Prescaler The SPI clock (SCK) is determined by the following equation: $f_{SCK} = f_{PCLK} / (CP + 1)$, where the CP ranges is from 0 to 255 Note: For the SPI master mode, the APB clock (f_{PCLK}) must be at least 2 times faster than the SPI SCK output.

SPI Data Register – SPIDR

This register stores the SPI received or transmitted Data.

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	DR							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	DR	Data Register The SPI data register is used to store the serial bus transmitted or received data. In the non-FIFO mode, writing data into the SPI data register will also load the data into the data transmission buffer, known as the TX buffer. Reading data from the SPI data register will return the data held in the data received buffer, named RX buffer.

SPI Status Register – SPISR

This register contains the relevant SPI status.

Offset: 0x014

Reset value: 0x0000_0003

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							BUSY
Type/Reset								RO 0
	7	6	5	4	3	2	1	0
	TO	SA	Reserved	RO	WC	RXBNE	TXE	TXBE
Type/Reset	WC 0	WC 0		WC 0	WC 0	RO 0	RO 1	RO 1

Bits	Field	Descriptions
[8]	BUSY	SPI Busy flag 0: SPI not busy 1: SPI busy In the master mode, this flag is reset when the TX buffer and TX shift register are both empty and is set when the TX buffer or the TX shift register are not empty. In the slave mode, this flag is set when SEL changes to an active level and is reset when SEL changes to an inactive level.
[7]	TO	Time Out flag 0: No RX FIFO time out 1: RX FIFO time out has occurred Once the time out counter value is equal to the TOC field setting in the SPIFTOCR register, the time out flag will be set and an interrupt will be generated if the TOIEN bit in the SPIIER register is enabled. This bit is cleared by writing 1. Note: This Time Out flag function is only available in the SPI FIFO mode.
[6]	SA	Slave Abort flag 0: No slave abort 1: Slave abort has occurred This bit is set by hardware and cleared by writing 1.
[4]	RO	Read Overrun flag 0: No read overrun 1: Read overrun has occurred This bit is set by hardware and cleared by writing 1.
[3]	WC	Write Collision flag 0: No write collision 1: Write collision has occurred This bit is set by hardware and cleared by writing 1.

Bits	Field	Descriptions
[2]	RXBNE	RX Buffer Not Empty flag 0: RX buffer is empty 1: RX buffer is not empty This bit indicates the RX buffer status in the non-FIFO mode. It is also used to indicate if the RX FIFO trigger level has been reached in the FIFO mode. This bit will be cleared when the SPI RX buffer is empty in the non-FIFO mode or if the number of data contained in RX FIFO is less than the trigger level which is specified by the RXFTLS field in the SPIFCR register in the SPI FIFO mode.
[1]	TXE	Transmission Register Empty flag 0: TX buffer or TX shift register is not empty 1: TX buffer and TX shift register both are empty
[0]	TXBE	TX Buffer Empty flag 0: TX buffer is not empty 1: TX buffer is empty In the FIFO mode, this bit if set indicates that the number of data contained in TX FIFO is equal to or less than the trigger level specified by the TXFTLS field in the SPIFCR register.

SPI FIFO Control Register – SPIFCR

This register contains the related SPI FIFO control including the FIFO enable control and the FIFO trigger level selections.

Offset: 0x018

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved					FIFOEN		Reserved
Type/Reset						RW 0		
	7	6	5	4	3	2	1	0
	Reserved	RXFTLS			Reserved	TXFTLS		
Type/Reset		RW 0	RW 0	RW 0		RW 0	RW 0	RW 0

Bits	Field	Descriptions
[10]	FIFOEN	FIFO Enable 0: FIFO is disabled 1: FIFO is enabled This bit cannot be set or reset when the SPI interface is in transmitting.

Bits	Field	Descriptions
[6:4]	RXFTLS	RX FIFO Trigger Level Select 000: Trigger level is 0 001: Trigger level is 1 010: Trigger level is 2 011: Trigger level is 3 100: Trigger level is 4 Others: Reserved The RXFTLS field is used to specify the RX FIFO trigger level. When the number of data contained in the RX FIFO is equal to or greater than the trigger level defined by the RXFTLS field, the RXBNE flag will be set
[2:0]	TXFTLS	TX FIFO Trigger Level Select 000: Trigger level is 0 001: Trigger level is 1 010: Trigger level is 2 011: Trigger level is 3 100: Trigger level is 4 Others: Reserved The TXFTLS field is used to specify the TX FIFO trigger level. When the number of data contained in the TX FIFO is equal to or less than the trigger level defined by the TXFTLS field, the TXBE flag will be set.

SPI FIFO Status Register – SPIFSR

This register contains the relevant SPI FIFO status.

Offset: 0x01C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved	RXFS			Reserved	TXFS		
Type/Reset		RO	0 RO	0 RO	0		RO	0 RO 0 RO 0

Bits	Field	Descriptions
[6:4]	RXFS	RX FIFO Status 000: RX FIFO empty 001: RX FIFO contains 1 data 010: RX FIFO contains 2 data 011: RX FIFO contains 3 data 100: RX FIFO contains 4 data Others: Reserved

Bits	Field	Descriptions
[2:0]	TXFS	TX FIFO Status 000: TX FIFO empty 001: TX FIFO contains 1 data 010: TX FIFO contains 2 data 011: TX FIFO contains 3 data 100: TX FIFO contains 4 data Others: Reserved

SPI FIFO Time Out Counter Register – SPIFTOCR

This register stores the SPI RX FIFO time out counter value.

Offset: 0x020

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	TOC							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	TOC	Time Out Counter Compare Value The time out counter starts to count from 0 after the SPI RX FIFO receives a data, and the counter value is reset once the data is read from the SPIDR register by software or another new data is received. If the FIFO does not receive new data or the software does not read data from the SPIDR register the time out counter value will continuously increase. When the time out counter value is equal to the TOC setting value, the TO flag in the SPISR register will be set and an interrupt will be generated if the TOIEN bit in the SPIIER register is set. The time out counter will be stopped when the RX FIFO is empty. The SPI FIFO time out function can be disabled by setting the TOC field to zero. The time out counter is driven by the system APB clock, named f_{PCLK}.

19 Universal Asynchronous Receiver Transmitter (UART)

Introduction

The Universal Asynchronous Receiver Transceiver, UART, provides a flexible full duplex data exchange using asynchronous transfer. The UART is used to translate data between parallel and serial interfaces, and is also commonly used for RS232 standard communication. The UART peripheral function supports a variety of interrupts.

The UART module includes a transmit data register TDR and transmit shift register TSR, and a receive data register RDR and receive shift register RSR. Software can detect a UART error status by reading the UART Status & Interrupt Flag Register, URSIFR. The status includes the condition of the transfer operations as well as several error conditions resulting from Parity, Overrun, Framing and Break events.

The UART includes a programmable baud rate generator which is capable of dividing the UART clock CK_APB (CK_UART) to produce a baud rate clock for the UART transmitter and receiver.

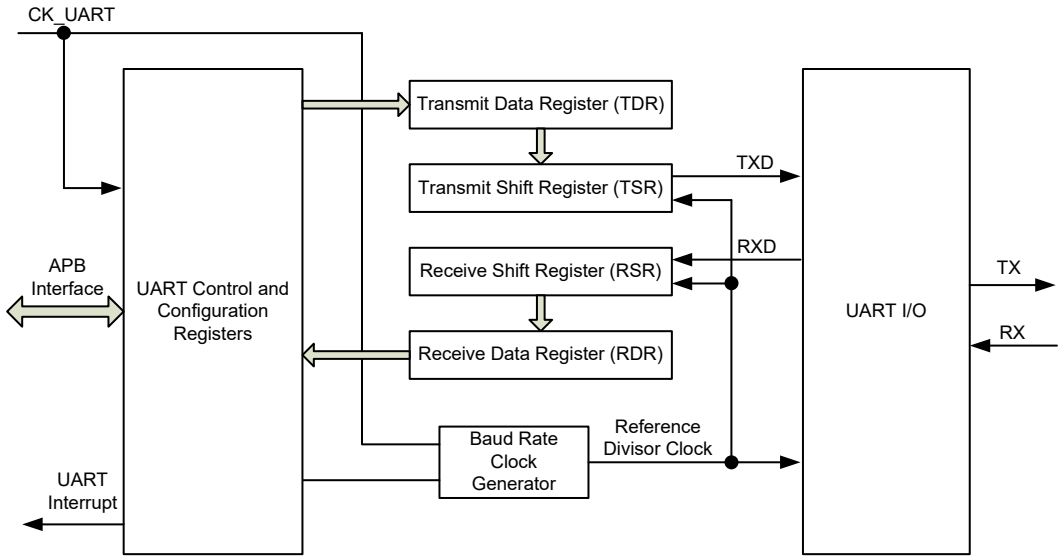


Figure 74. UART Block Diagram

Features

- Supports asynchronous serial communication modes
- Full Duplex Communication Capability
- Programming baud rate clock frequency up to (f_{PCLK}/16) MHz
- Fully programmable serial communication functions including:
  - Word length: 7, 8 or 9-bit character
  - Parity: Even, odd or no-parity bit generation and detection
  - Stop bit: 1 or 2 stop bits generation
  - Bit order: LSB-first or MSB-first transfer
- Error detection: Parity, overrun, and frame error

Functional Descriptions

Serial Data Format

The UART module performs a parallel-to-serial conversion on data that is written to the transmit data register and then sends the data with the following format: Start bit, 7 ~ 9 LSB/MSB first data bits, optional Parity bit and finally 1 ~ 2 Stop bits. The Start bit has the opposite polarity of the data line idle state. The Stop bit is the same as the data line idle state and provides a delay before the next start situation. Both the Start and Stop bits are used for data synchronization during the asynchronous data transmission.

The UART module also performs a serial-to-parallel conversion on the data that is read from the receive data register. It will first check the Parity bit and will then look for a Stop bit. If the Stop bit is not found, the UART module will consider the entire word transmission as failed and respond with a Framing Error.

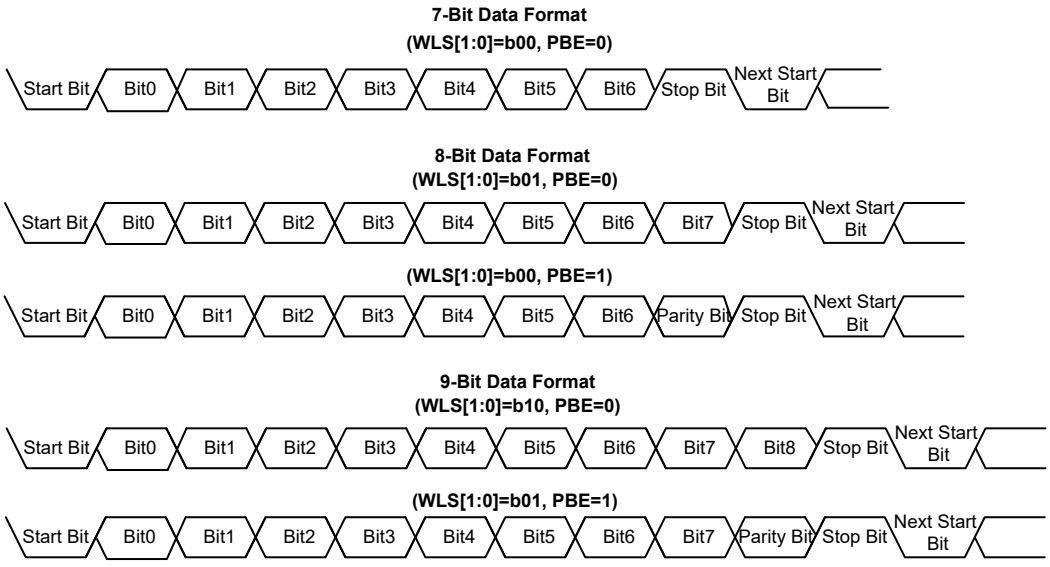


Figure 75. UART Serial Data Format

Baud Rate Generation

The baud rate for the UART receiver and transmitter are both set with the same values. The baud rate divisor, BRD, has the following relationship with the UART clock which is known as CK_UART.

Baud Rate Clock = CK_UART / BRD

Where the CK_UART clock is the APB clock connected to the UART while the BRD range is from 16 to 65535.

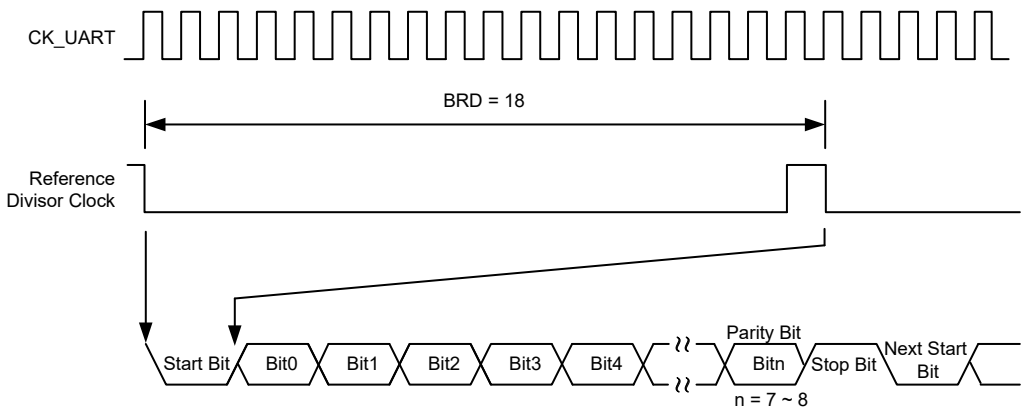


Figure 76. UART Clock CK_UART and Data Frame Timing

Table 38. Baud Rate Deviation Error Calculation – CK_UART = 16 MHz

Baud Rate		CK_UART = 16 MHz		
No.	Kbps	Actual	BRD	Deviation Error Rate
1	2.4	2.4	8333	0.00%
2	9.6	9.6	2083	0.02%
3	19.2	19.2	1042	-0.03%
4	57.6	57.6	347	0.06%
5	115.2	114.9	174	-0.22%
6	230.4	229.9	87	-0.22%
7	460.8	465.1	43	0.94%
8	921.6	909.1	22	-1.36%
9	1250	1250	16	0%

Table 39. Baud Rate Deviation Error Calculation – CK_UART = 10 MHz

Baud Rate		CK_UART = 10 MHz		
No.	Kbps	Actual	BRD	Deviation Error Rate
1	2.4	2.4	4167	-0.01%
2	9.6	9.6	1042	-0.03%
3	19.2	19.2	521	-0.03%
4	57.6	57.6	174	-0.22%
5	115.2	114.9	87	-0.22%
6	230.4	232.6	43	0.94%
7	460.8	454.5	22	-1.36%
8	625	625	16	0%

Interrupts and Status

The UART can generate interrupts when the following events occur and the corresponding interrupt enable bits are set:

- Receiver line status interrupts: The interrupts are generated when the UART receiver overrun error, parity error, framing error and break event occur.
- Transmit data register empty interrupt: An interrupt is generated when the content of the transmit data register is transferred to the transmit shift register (TSR).
- Transmit complete interrupt: An interrupt is generated when the transmit data register (TDR) is empty and the content of the transmit shift register (TSR) is also completely shifted.
- Receive data ready interrupt: An interrupt is generated when the content of the receive shift register (RSR) has been transferred to the URDR register and is ready to read.

Register Map

The following table shows the UART registers and reset values.

Table 40. UART Register Map

Register	Offset	Description	Reset Value
URDR	0x000	UART Data Register	0x0000_0000
URCR	0x004	UART Control Register	0x0000_0000
URIER	0x00C	UART Interrupt Enable Register	0x0000_0000
URSIFR	0x010	UART Status & Interrupt Flag Register	0x0000_0180
URDLR	0x024	UART Divider Latch Register	0x0000_0010
URTSTR	0x028	UART Test Register	0x0000_0000

Register Descriptions

UART Data Register – URDR

The register is used to access the UART transmitted and received data.

Offset: 0x000

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							DB
Type/Reset								RW 0
	7	6	5	4	3	2	1	0
	DB							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[8:0]	DB	By reading this register, the UART will return a 7, 8 and 9-bit received data. The DB field bit 8 is valid for the 9-bit mode only and is fixed at 0 for the 8-bit mode. For the 7-bit mode, the DB[6:0] field contains the available bits. By writing to this register, the UART will send out 7, 8 or 9-bit transmitted data. The DB field bit 8 is valid for the 9-bit mode only and will be ignored for the 8-bit mode. For the 7-bit mode, the DB[6:0] field contains the available bits.

UART Control Register – URCR

The register specifies the serial parameters such as data length, parity and stop bit for the UART.

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved	BCB	SPE	EPE	PBE	NSB	WLS	
Type/Reset		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
		Reserved	URRXEN	URTXEN	Reserved	TRSM		Reserved
Type/Reset			RW 0	RW 0		RW 0		

Bits	Field	Descriptions
[14]	BCB	Break Control Bit When this bit is set 1, the serial data output on the UART TX pin will be forced to the Spacing State (logic 0). This bit acts only on the UART TX output pin and has no effect on the transmitter logic.
[13]	SPE	Stick Parity Enable 0: Disable stick parity 1: Stick Parity bit is transmitted This bit is only available when the PBE bit is set to 1. If both the PBE and SPE bits are set to 1 and the EPE bit is cleared to 0, the transmitted parity bit will be stuck to 1. However, when the PBE and SPE bits are set to 1 and also the EPE bit is set to 1, the transmitted parity bit will be stuck to 0.
[12]	EPE	Even Parity Enable 0: Odd number of logic 1's are transmitted or checked in the data word and parity bits 1: Even number of logic 1's are transmitted or checked in the data word and parity bits This bit is only available when the PBE bit is set to 1.
[11]	PBE	Parity Bit Enable 0: Parity bit is not generated (transmitted data) and checked (receive data) during transfer 1: Parity bit is generated and checked during transfer Note: When the WLS field is set to "10" to select the 9-bit data format, writing to the PBE bit has no effect.
[10]	NSB	Number of STOP bit 0: One STOP bit is generated in the transmitted data 1: Two STOP bits are generated when 8-bit or 9-bit word length is selected

Bits	Field	Descriptions
[9:8]	WLS	Word Length Select 00: 7 bits 01: 8 bits 10: 9 bits 11: Reserved
[5]	URRXEN	UART RX Enable 0: Disable 1: Enable
[4]	URTXEN	UART TX Enable 0: Disable 1: Enable
[2]	TRSM	Transfer Mode Selection This bit is used to select the data transfer protocol. 0: LSB first 1: MSB first

UART Interrupt Enable Register – URIER

This register is used to enable the related UART interrupt function. The UART module generates interrupts to the controller when the corresponding events occur and the corresponding interrupt enable bits are set.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved	BIE	FEIE	PEIE	OEIE	TXCIE	TXDEIE	RXDRIE
Type/Reset		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[6]	BIE	Break Interrupt Enable 0: Disable 1: Enable If this bit is set, an interrupt will be generated when the BII bit in the URSIFR register is set.
[5]	FEIE	Framing Error Interrupt Enable 0: Disable 1: Enable If this bit is set, an interrupt will generated when the FEI bit in the URSIFR register is set.

Bits	Field	Descriptions
[4]	PEIE	Parity Error Interrupt Enable 0: Disable 1: Enable If this bit is set, an interrupt will be generated when the PEI bit in the URSIFR register is set.
[3]	OEIE	Overrun Error Interrupt Enable 0: Disable 1: Enable If this bit is set, an interrupt will be generated when the OEI bit in the URSIFR register is set.
[2]	TXCIE	Transmit Complete Interrupt Enable 0: Disable 1: Enable If this bit is set, an interrupt will be generated when the TXC bit in the URSIFR register is set.
[1]	TXDEIE	Transmit Data Register Empty Interrupt Enable 0: Disable 1: Enable If this bit is set, an interrupt will be generated when the TXDE bit in the URSIFR register is set.
[0]	RXDRIE	Receive Data Ready Interrupt Enable 0: Disable 1: Enable If this bit is set, an interrupt will be generated when the RXDR bit in the URSIFR register is set.

UART Status & Interrupt Flag Register – URSIFR

This register contains the corresponding UART status.

Offset: 0x010

Reset value: 0x0000_0180

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							TXC
Type/Reset								RO 1
	7	6	5	4	3	2	1	0
	TXDE	Reserved	RXDR	BII	FEI	PEI	OEI	Reserved
Type/Reset	RO 1		RO 0	WC 0	WC 0	WC 0	WC 0	

Bits	Field	Descriptions
[8]	TXC	Transmit Complete 0: Either the transmit data register (TDR) or transmit shift register (TSR) is not empty 1: Both the transmit data register (TDR) and transmit shift register (TSR) are empty When this bit is set, an interrupt will be generated if the TXCIE bit in the URIER register is set to 1. This bit is cleared by a write to the URDR register with new data.
[7]	TXDE	Transmit Data Register Empty 0: Transmit data register is not empty 1: Transmit data register is empty The TXDEIE bit is set by hardware when the content of the transmit data register is transferred to the transmit shift register (TSR). An interrupt will be generated if the TXDEIE bit in the URIER register is set to 1. This bit is cleared by a write to the URDR register with new data.
[5]	RXDR	RX Data Ready 0: Receive data register is empty 1: The received data in receive data register is ready to read This bit is set by hardware when the content of the receive shift register (RSR) has been transferred to the URDR register. An interrupt will be generated if the RXDRIE bit in the URIER register is set to 1. It is cleared by a read to the URDR register.
[4]	BII	Break Interrupt Indicator This bit is set to 1 whenever the received data input is held in the “spacing state” (logic 0) for longer than a full character transmission time, which is the total time of “start bit” + data bits + “parity” + “stop bits” duration. Writing 1 to this bit clears the flag.
[3]	FEI	Framing Error Indicator This bit is set 1 whenever the received character does not have a valid stop bit, which means, the stop bit following the last data bit or parity bit is detected as logic 0. Writing 1 to this bit clears the flag.

Bits	Field	Descriptions
[2]	PEI	Parity Error Indicator This bit is set to 1 whenever the received character does not have a valid parity bit. Writing 1 to this bit clears the flag.
[1]	OEI	Overrun Error Indicator An overrun error will occur only after the receive data register is full and when the next character has been completely received in the receive shift register. The character in the receive shift register will be overwritten when a new character is received in the receive shift register after an overrun event occurs, but the data in the receive shift register will not be transferred to the receive data register. The OEI bit is used to indicate event as soon as it happens. Writing 1 to this bit clears the flag.

UART Divider Latch Register – URDLR

The register is used to determine the UART clock divided ratio to generate the appropriate baud rate.

Offset: 0x024

Reset value: 0x0000_0010

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	BRD							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	BRD							
Type/Reset	RW 0	RW 0	RW 0	RW 1	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[15:0]	BRD	Baud Rate Divider The 16 bits define the UART clock divider ratio. Baud Rate = CK_UART / BRD Where the CK_UART clock is the clock connected to the UART module. BRD = 16 ~ 65535 for the UART mode

UART Test Register – URTSTR

This register controls the UART debug mode.

Offset: 0x028

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved						LBM	
Type/Reset							RW 0	RW 0

Bits	Field	Descriptions
[1:0]	LBM	Loopback Test Mode Select 0x: Normal Operation 10: Automatic Echo Mode 11: Loopback Mode

20 LED Controller (LEDC)

Introduction

The LED controller is used to drive 8-segment digital displays. The LED controller can drive up to eight 8-segment digital displays. Users have the flexibility to configure the pin position and number of the COMs according to the digital displays in their application. In a complete frame period, the enabled COMs will be scanned from the lower to the higher. Taking an example of where four 8-segment LEDs are used and where COM0, COM5, COM6 and COM7 are enabled. Here COM0, COM5, COM6 and the COM7 will be scanned successively in this sequence within a complete frame period. The scanning time of each COM port is equal to 1/4 frame, which is subdivided into the dead time duty and the COM duty. Users can adjust the dead time duty to change the LED brightness.

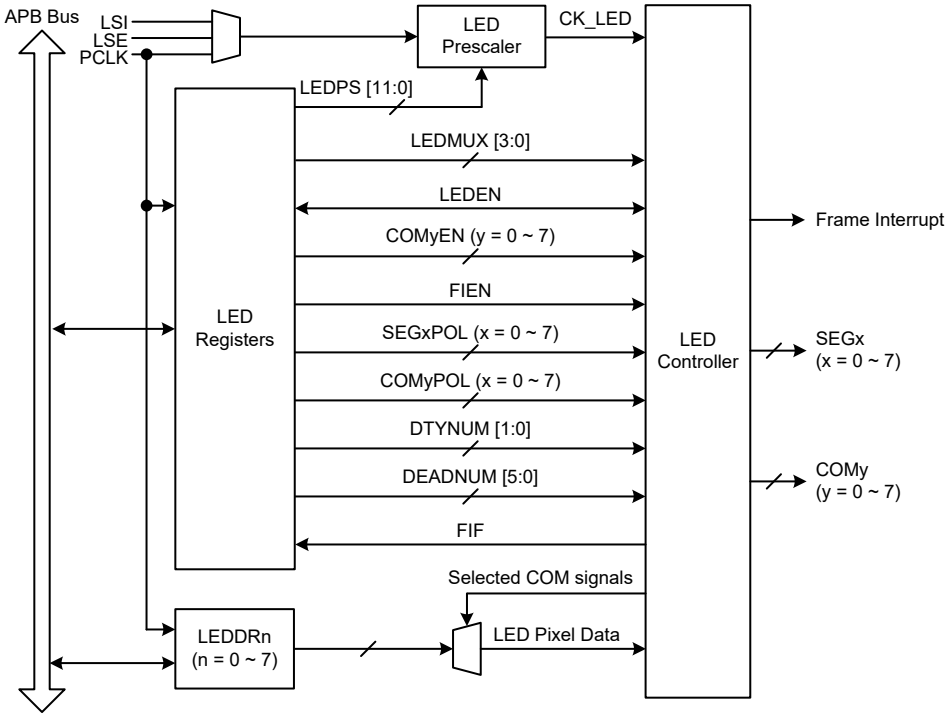


Figure 77. LEDC Block Diagram

Features

- Supports 8-segment digital displays up to 8
- Supports 8-segment digital displays with common anode or common cathode
- Supports frame interrupt
- Three frequency sources: LSI, LSE and PCLK
- The LED light on/off times can be controlled using the dead time setting

Functional Description

The position of each pixel point is represented by SEGx and COMy. The devices can drive up to eight 8-segment digital displays, x = 0 ~ 7, y = 0 ~ 7. The number of COMs to be enabled is N. The following will take N = 4, here COM0, COM5, COM6 and COM7 are enabled as an example to introduce the LEDC functions.

LEDC Basic Setting

The following show the steps to configure the LEDC drive module.

- Set the LEDSRC, LEDPS and DTYNUM bit fields in the LEDCR register.
- The required COMy can be enabled through the COMyEN bit in the LEDCER register.
- Configure the required COMy and SEGx pins for the LEDC functions using the AFIO function.
- Set the current drive capability of the COMy and SEGx.
- Select the COMy and SEGx output polarity by configuring the LEDPCR register.
- Select the dead time clock number by the DEADNUM bit field in the LEDDTCR register.
- Initialise the LEDDR register.
- The LEDEN bit in the LEDCR is set high to enable the LED driver module.

Except for the LEDEN bit which needs to be enabled during the last step, there is no sequence requirement for the setup steps provided above. When the LEDEN bit is set high, the LEDSRC, LEDPS and DTYNUM bit field contents should remain unchanged. However, the LEDDR and DEADNUM bit fields can be modified.

LEDC Clock Source Selection

The LEDC clock can be sourced from LSI, LSE or PCLK, which is selected by the LEDSRC bit field in the LEDCR register. The selected clock source passes through a divider, the division ratio of which is selected by the LEDPS bit field in the LEDCR register. The LED clock, CK_LED, is determined by the following equation:

$$f_{CK\_LED} = (LSI, LSE \text{ or } PCLK) / (LEDPS[11:0] + 1)$$

LEDC Operational Description

Assuming that N digital displays are used, then each digital display has an operating time of 1/N frame. The required CK_LED clock number for each digital display scan can be configured as 8, 16, 32 or 64 by the DTYNUM[1:0] bit field in the LEDDTCR register.

$$1/N \text{ frame} = (8, 16, 32 \text{ or } 64) \times CK\_LED \text{ clock}$$

Each digital display scanning cycle can be subdivided into a dead time duty and a COM duty. The LED will not be illuminated during the dead time. The dead time clock number is selected by the DEADNUM[5:0] bit field. The duty clock number is selected by the DTYNUM[1:0] bit field. The dead time clock number should be less than the duty clock number. The maximum clock number of the dead time varies and is dependent on the duty clock number.

COM duty = 1/N frame – Dead time

Note that 8-segment digital displays have two connection methods, namely common cathode and common anode. For this reason it may be necessary to invert the SEGx or COMy outputs in applications, which can be implemented by configuring the SEGx or COMy polarity control bits.

- The SEGxPOL bit is used to control the SEGx polarity. If SEGxPOL is set to 1, then SEGx will be an inverted output.
- The COMyPOL bit is used to control the COMy polarity. If COMyPOL is set to 1, then COMy will be an inverted output.

The following will introduce several connection methods in the application.

Common Cathode 8-segment Digital Display Connection

- SEGx output high and COMy output low → LED on
- COMy output high → LED off

The register configurations are as follows:

- Set SEGxPOL = 0 and COMyPOL = 0, the output is non-inverted.
- Set DTYNUM[1:0] = b11, the period of each digital display scan is 64 CK_LED clocks.
- The selected COMs are enabled.

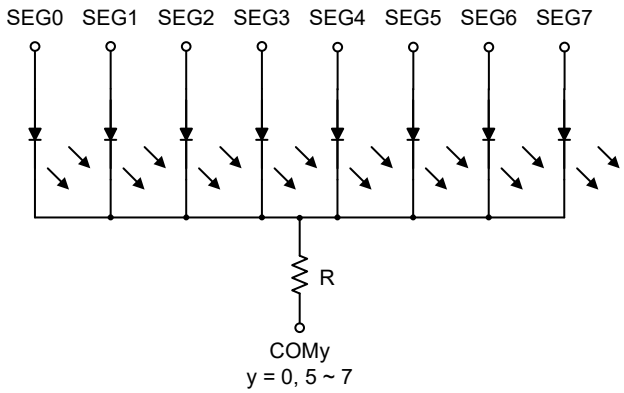


Figure 78. Common Cathode 8-segment Digital Display Connection

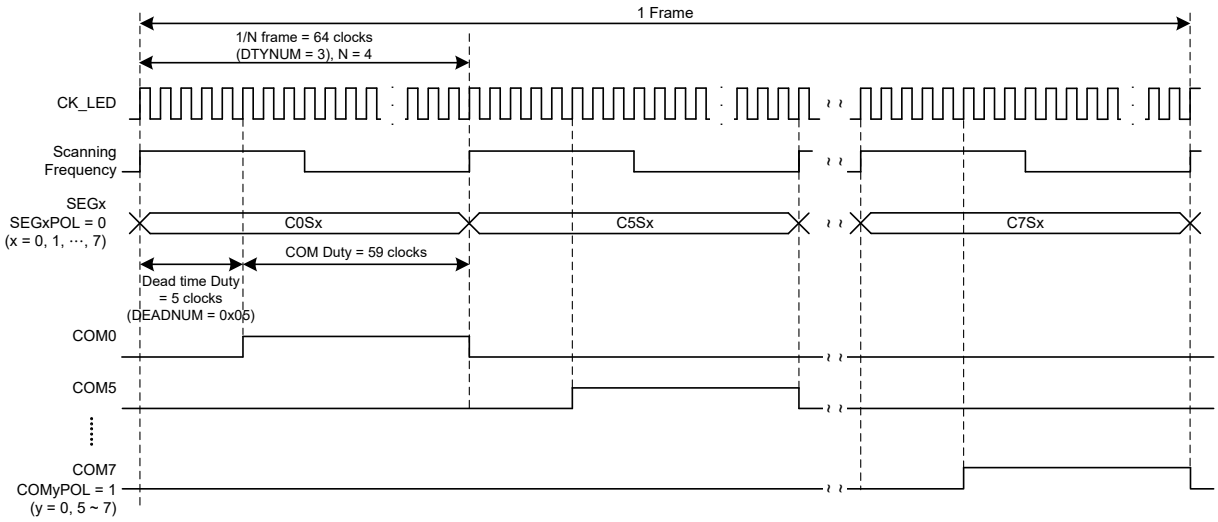


Figure 79. Common Cathode 8-segment Digital Display Timing

Common Anode 8-segment Digital Display + NPN BJT Connection

- SEGx output low and COMy output high → LED on
- COMy output low → LED off

Due to the I/O drive capability limitation, an external transistor is needed to increase the current drive capability if using this connection method.

The register configurations are as follows:

- Set SEGxPOL = 1 and COMyPOL = 1, the output is inverted.
- Set DTYNUM[1:0] = b11, the period of each digital display scan is 64 CK_LED clocks.
- The selected COMs are enabled.

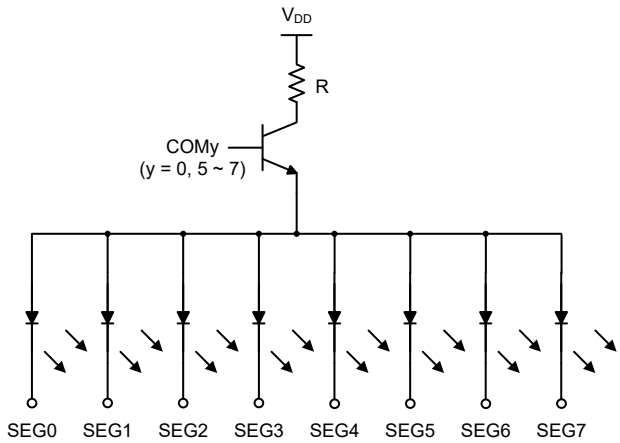


Figure 80. Common Anode 8-segment Digital Display + NPN BJT Connection

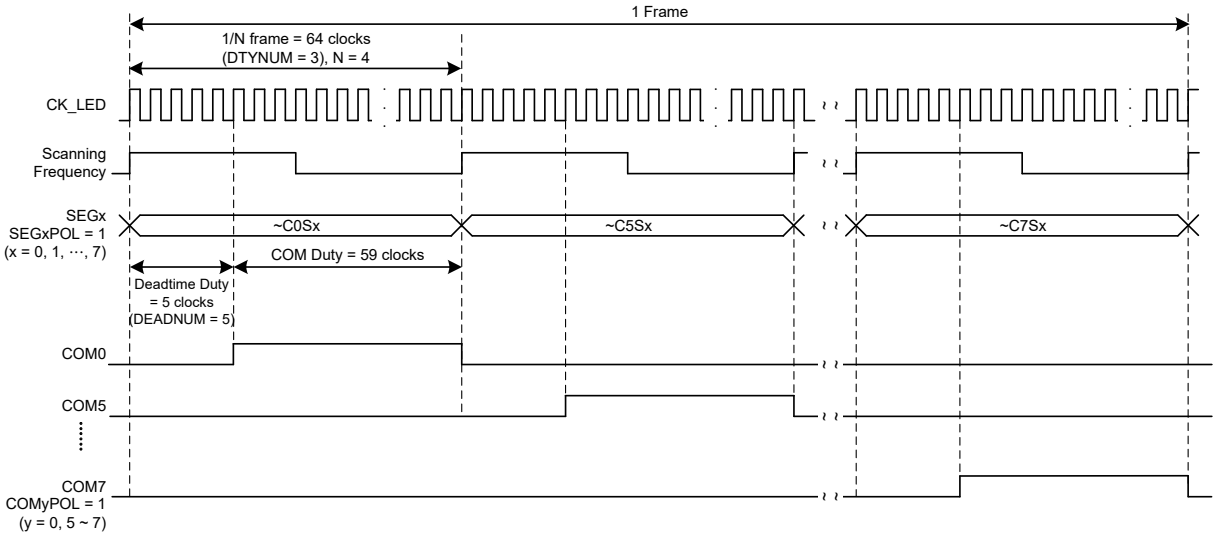


Figure 81. Common Anode 8-segment Digital Display+ NPN BJT Timing

Common Cathode 8-segment Digital Display Connection + NPN Transistor

- SEG output high and COM output high → LED on
- COM output low → LED off

Connect an external transistor to prevent the LED current from affecting the device.

The register configurations are as follows.

- Set SEGxPOL = 1 and COMyPOL = 1, the output is inverted.
- Set DTYNUM[1:0] = b11, the period of each digital display scan is 64 CK_LED clocks.
- The selected COMs are enabled.

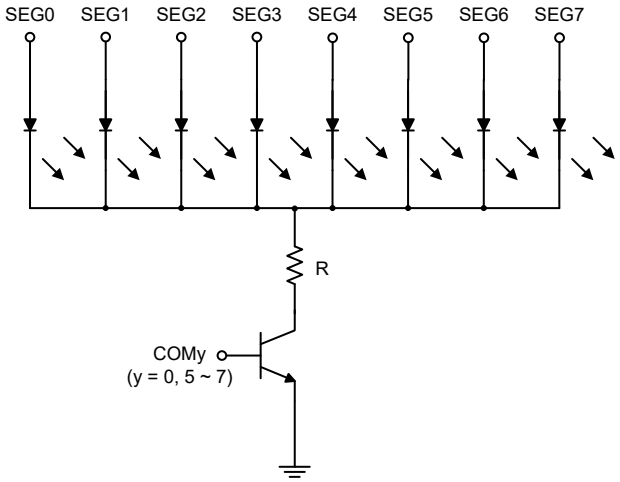


Figure 82. Common Cathode 8-segment Digital Display + NPN Transistor Connection

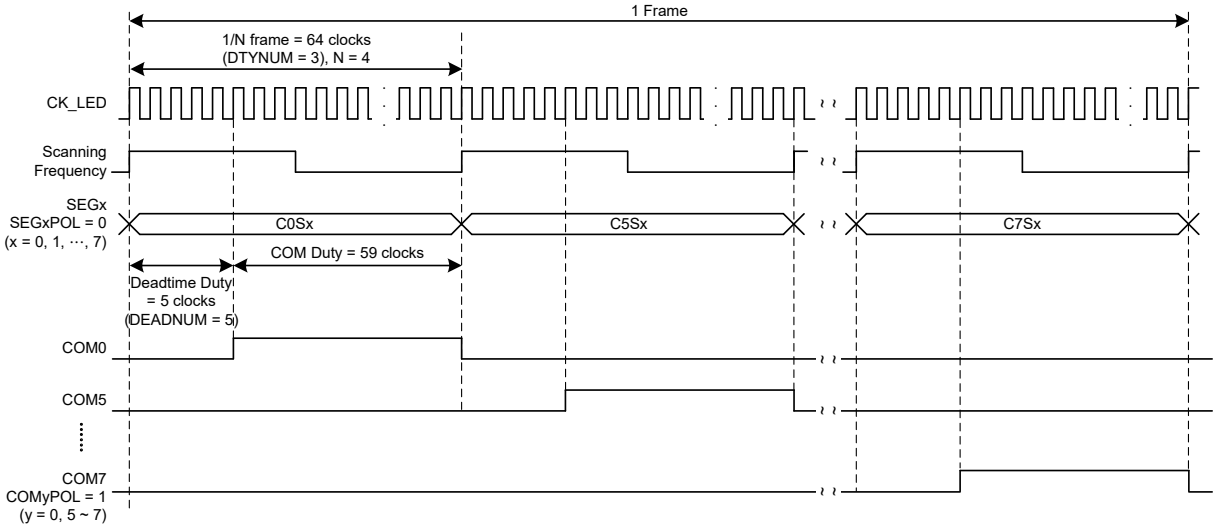


Figure 83. Common Cathode 8-segment Digital Display + NPN Transistor Timing

Common Anode 8-segment Digital Display + PNP BJT Connection

- SEG output low and COM output low → LED on
- COM output high → LED off

The register configurations are as follows:

- Set SEGxPOL = 1, the output is inverted. Set COMyPOL = 0, the output is non-inverted.
- Set DTYNUM[1:0] = b11, the period of each digital display scan is 64 CK_LED clocks.
- The selected COMs are enabled.

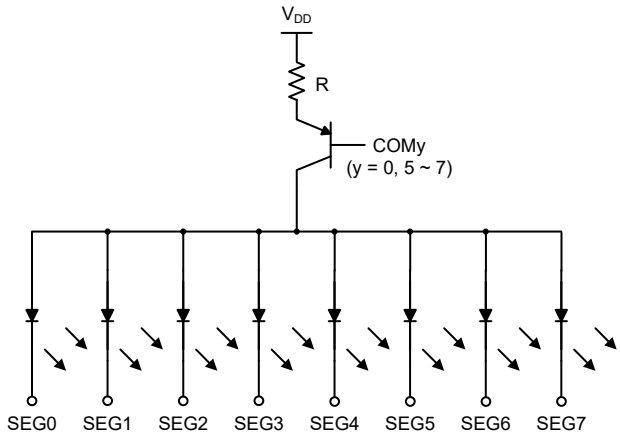


Figure 84. Common Anode 8-segment Digital Display + PNP BJT Connection

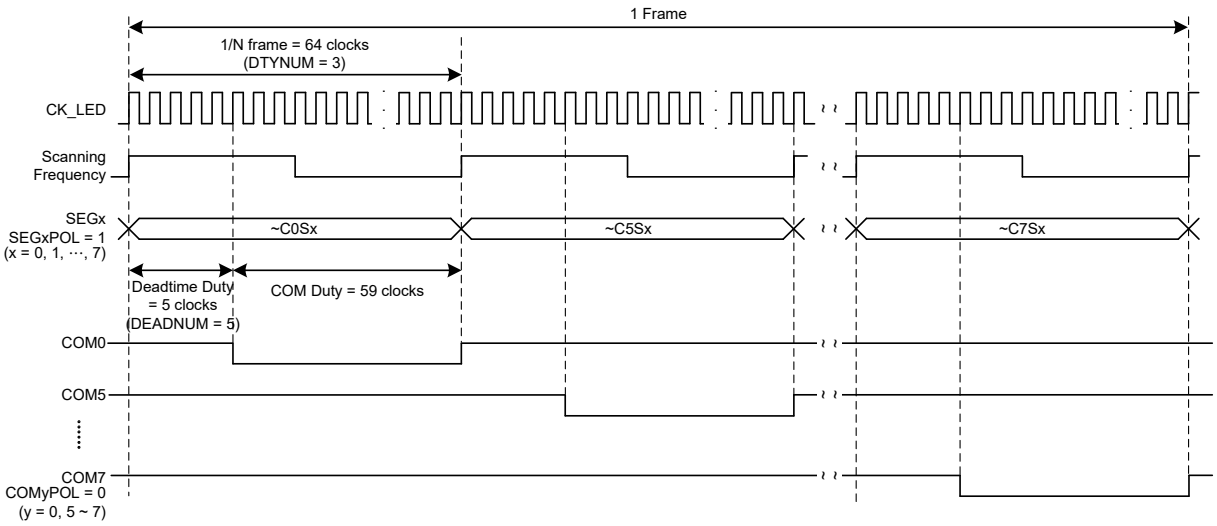


Figure 85. Common Anode 8-segment Digital Display + PNP BJT Timing

LEDC Frame Interrupt

The frame interrupt can be enabled after the pixel data of the last COM duty is latched. Users can update the pixel data or adjust the dead time duty in the interrupt service routine.

The Frame interrupt flag, FIF, in the LEDSR register is set by hardware and reset by software by writing a "1" to it.

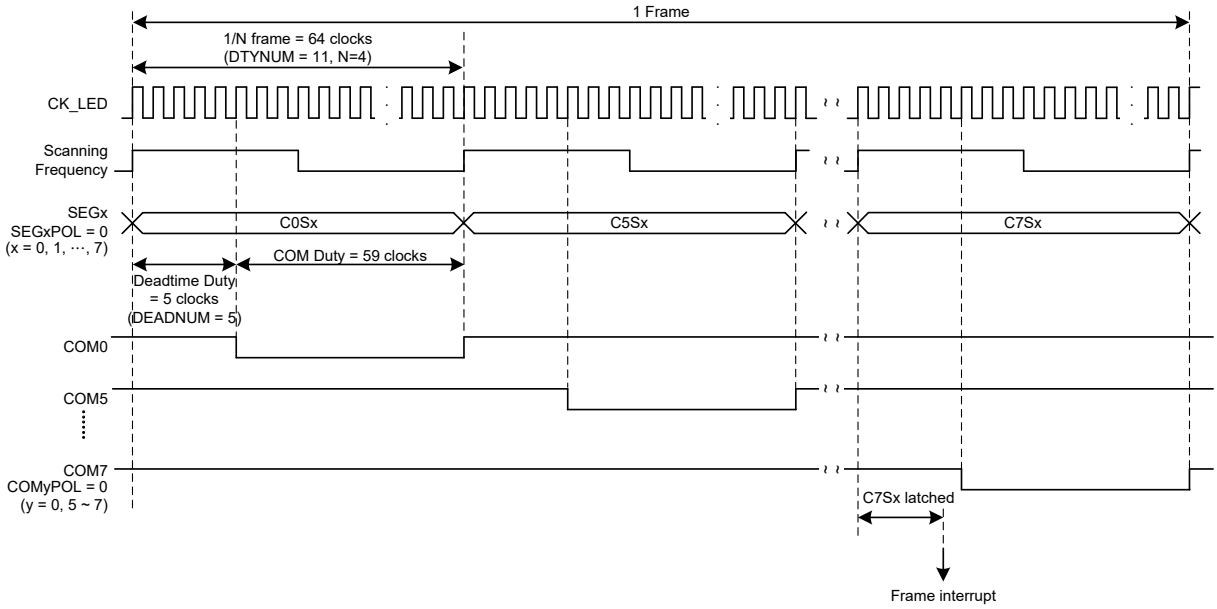


Figure 86. Frame Interrupt Diagram

LEDC Data Update Method

After the LEDC function is enabled, if the displayed characters need to be updated, it is recommended to use a frame interrupt to update the pixel data. Using this method will more effectively maintain the display integrity of each frame. The corresponding relationship between the LED pixel data and (SEGx, COMy) is shown in the table below.

Table 41. LED Pixel Data and (SEGx, COMy) Relationship

LEDDRn	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
COMy	CnS7	CnS6	CnS5	CnS4	CnS3	CnS2	CnS1	CnS0

Note: SEGx corresponds to the CnSx bit in the LEDDRn register and COMy corresponds to the LEDDRn register. The CnSx value represents the pixel data of the 8-segment digital display.

Frame Rate Calculation

The calculation formula of the frame rate frequency is as follows:

$$f_{frame} = f_{CK\_LED} / (\text{Duty clock} \times N)$$

Where N is the number of COMs selected, the maximum value of N is 8. The duty clock number can be configured as 8, 16, 32 or 64 by the DTYNUM[1:0] bit field in the LEDDTCR register. The f_{CK_LED} is calculated by the following formula. The f_{clock_source} can be selected to be sourced from LSI, LSE or PCLK.

$$f_{CK\_LED} = f_{clock\_source} / (LEDPS + 1)$$

Examples

(1) If the CK_LED clock source f_{clock_source} is 32.768 kHz, DTYNUM[1:0] = b11, LEDPS[11:0] = 0x000 and 8 COMs are enabled, then the frame rate can be calculated as follows.

$$\text{Frame rate} = 32768\ \text{Hz} / (64 \times 8) = 64\ \text{Hz}$$

If the operating frequency needs to be greater than 120 Hz, this can be implemented by setting the DTYNUM[1:0] bits to b10. The frame rate can be calculated as follows.

$$\text{Frame rate} = 32768\ \text{Hz} / (32 \times 8) = 128\ \text{Hz}$$

(2) If the CK_LED clock source f_{clock_source} is 16 MHz, DTYNUM[1:0] = b11, LEDPS[11:0] = 0x206 and 8 COMs are enabled, then the frame rate can be calculated as follows.

$$\text{Frame rate} = [16\ \text{MHz} / (518 + 1)] / (64 \times 8) = 60\ \text{Hz}$$

If the operating frequency needs to be greater than 120 Hz, this can be implemented by setting LEDPS[11:0] to 0x103. The frame rate can be calculated as follows.

$$\text{Frame rate} = [16\ \text{MHz} / (259 + 1)] / (64 \times 8) = 120\ \text{Hz}$$

If the operating frequency needs to be greater than 120 Hz, this can also be implemented by setting the DTYNUM[1:0] bits to b10. The frame rate can be calculated as follows.

$$\text{Frame rate} = [16\ \text{MHz} / (518 + 1)] / (32 \times 8) = 120\ \text{Hz}$$

Register Map

The following table shows the LEDC registers and their reset values.

Table 42. LEDC Register Map

Register	Offset	Description	Reset Value
LEDCR	0x000	LED Control Register	0x0000_0000
LEDCER	0x004	LED COM Enable Register	0x0000_0000
LEDPCR	0x008	LED Polarity Control Register	0x0000_0000
LEDIER	0x00C	LED Interrupt Enable Register	0x0000_0000
LEDSR	0x010	LED Status Register	0x0000_0000
LEDDTCR	0x014	LED Dead Time Control Register	0x0000_0000
LEDDR0	0x018	LED Data Register 0	0x0000_0000
LEDDR1	0x01C	LED Data Register 1	0x0000_0000
LEDDR2	0x020	LED Data Register 2	0x0000_0000
LEDDR3	0x024	LED Data Register 3	0x0000_0000
LEDDR4	0x028	LED Data Register 4	0x0000_0000
LEDDR5	0x02C	LED Data Register 5	0x0000_0000
LEDDR6	0x030	LED Data Register 6	0x0000_0000
LEDDR7	0x034	LED Data Register 7	0x0000_0000

Register Descriptions

LED Control Register – LEDCR

This register is used to control the LEDC clock source, prescaler, duty clock number and LEDC function enable.

Offset: 0x000
Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved				LEDPS			
Type/Reset					RW 0	RW 0	RW 0	RW 0
	23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	LEDPS							
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
	Reserved		DTYNUM		Reserved		LEDSRC	
Type/Reset			RW 0	RW 0			RW 0	RW 0
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	Reserved							LEDEN
Type/Reset								RW 0

Bits	Field	Descriptions
[27:16]	LEDPS	LEDC Clock Prescaler 0x000: CK_LED = CLKIN/1 0x001: CK_LED = CLKIN/2 0x002: CK_LED = CLKIN/3 … 0xFFF: CK_LED = CLKIN/4096 The CLKIN clock can be sourced from LSI, LSE or PCLK, which is selected by the LEDSRC bit field in the LEDCR register.

Bits	Field	Descriptions
[13:12]	DTYNUM	Duty Clock Numbers 00: 8 CK_LED clocks 01: 16 CK_LED clocks 10: 32 CK_LED clocks 11: 64 CK_LED clocks Assuming that N digital displays are used, the number of COMs is N. Each digital display operating time is 1/N frame. The required CK_LED clock number for each digital display scanning cycle can be configured as 8, 16, 32 or 64 by the DTYNUM[1:0] bit field. Each digital display scanning cycle can be subdivided into a dead time duty and a COM duty.
[9:8]	LEDSRC	LEDC Clock Source Selection 00: PCLK 01: LSI 10: LSE 11: Reserved
[0]	LEDEN	LEDC Enable Bit 0: Disable 1: Enable The LEDC state machine will be enabled when this bit is set to 1. If this bit is set to 0, the state machine will continue to output until the current frame operation is completed, then the state of the state machine will be cleared and finally the LEDEN bit will be cleared to zero by the hardware. Therefore, the LEDEN bit must be polled continuously after clearing the LEDEN bit has been executed, the LED controller will not actually cease operation until the LEDEN becomes 0. When the LEDEN bit is set high, the LEDSRC, LEDPS and DTYNUM bit field contents should remain unchanged. Even if new data is written into these bit fields, they are invalid, however the LEDDR and DEADNUM bit fields can be modified. It is recommended to update in the frame interrupt subroutine.

LED COM Enable Register – LEDCER

This register is used to control the COMy enable, (y = 0 ~ 7).

Offset: 0x004

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	COM7EN	COM6EN	COM5EN	COM4EN	COM3EN	COM2EN	COM1EN	COM0EN
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	COMyEN	COMy Enable Bit (y = 0 ~ 7) 0: COMy is disabled 1: COMy is enabled In a complete frame period, the enabled COMs will be scanned from low number to high number. Assuming that four 8-segment LEDs are used, the COM7EN, COM6EN, COM5EN and COM0EN bits are set to 1 and the remaining bits are set to zero. Here COM0, COM5, COM6 and COM7 will be scanned successively within a complete frame period. The scanning time of for each COM port is equal to 1/4 frame, which is subdivided into dead time duty and COM duty. Users can adjust the dead time duty to change the LED brightness. Therefore, a complete frame scans COM0 → COM5 → COM6 → COM7 in sequence. If no COMyEN bit is set, the LEDC will not operate after LEDEN is enabled.

LED Polarity Control Register – LEDPCR

This register controls the polarity of the COMy and SEGx. (x = 0 ~ 7, y = 0 ~ 7)

Offset: 0x008

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	SEG7POL	SEG6POL	SEG5POL	SEG4POL	SEG3POL	SEG2POL	SEG1POL	SEG0POL
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	COM7POL	COM6POL	COM5POL	COM4POL	COM3POL	COM2POL	COM1POL	COM0POL
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[23:16]	SEGxPOL	SEGx Output Polarity (x = 0 ~ 7) 0: Output non-inverted 1: Output inverted
[7:0]	COMyPOL	COMy Output Polarity (y = 0 ~ 7) 0: Output non-inverted 1: Output inverted

LED Interrupt Enable Register – LEDIER

This register is used to control the frame interrupt enable.

Offset: 0x00C

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved							FIEN
Type/Reset								RW 0

Bits	Field	Descriptions
[0]	FIEN	Frame Interrupt Enable 0: Disable 1: Enable

LED Status Register – LEDSR

This register specifies the frame interrupt flag.

Offset: 0x010

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved							FIF
Type/Reset								WC 0

Bits	Field	Descriptions
[0]	FIF	Frame interrupt flag 0: No frame interrupt occurs 1: Frame interrupt occurs Set by hardware and reset by software writing 1.

LED Dead Time Control Register – LEDDTCR

This register specifies the dead time duty.

Offset: 0x014

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								
	7	6	5	4	3	2	1	0
	Reserved		DEADNUM					
Type/Reset			RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[5:0]	DEADNUM	Dead time Clock Numbers 000000: 0 CK_LED clock 000001: 1 CK_LED clock 000010: 2 CK_LED clocks 000011: 3 CK_LED clocks … 111111: 63 CK_LED clocks COM duty clock number = Duty clock number – Dead time clock number The dead time clock number should be less than duty clock number. The maximum clock number of the dead time varies and is dependent upon the duty clock number. If DTYNUM bits are set to 00, that is, the duty cycle clock is 8 CK_LED clocks, only bit 2 ~ bit 0 of the DEADNUM bit field is valid. If DTYNUM bits are set to 01, that is, the duty cycle clock is 16 CK_LED clocks, only bit 3 ~ bit 0 of the DEADNUM bit field is valid. If DTYNUM bits are set to 10, that is, the duty cycle clock is 32 CK_LED clocks, only bit 4 ~ bit 0 of the DEADNUM bit field is valid. If DTYNUM bits are set to 11, that is, the duty cycle clock is 64 CK_LED clocks, the bits of the DEADNUM bit field are all valid.

LED Data Register n – LEDDRn (n = 0 ~ 7)

This register specifies that the LCD pixel points that are represented by SEGx and COMy are to be illuminated. (x = 0 ~ 7, y = n = 0 ~ 7)

offset: 0x018 ~ 0x044

Reset value: 0x0000_0000

	31	30	29	28	27	26	25	24
	Reserved							
Type/Reset								
	23	22	21	20	19	18	17	16
	Reserved							
Type/Reset								
	15	14	13	12	11	10	9	8
	Reserved							
Type/Reset								

	7	6	5	4	3	2	1	0
	CnS7	CnS6	CnS5	CnS4	CnS3	CnS2	CnS1	CnS0
Type/Reset	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

Bits	Field	Descriptions
[7:0]	CnSx	LED Pixel Data (x = 0 ~ 7, n = 0 ~ 7) 0: Output low 1: Output high