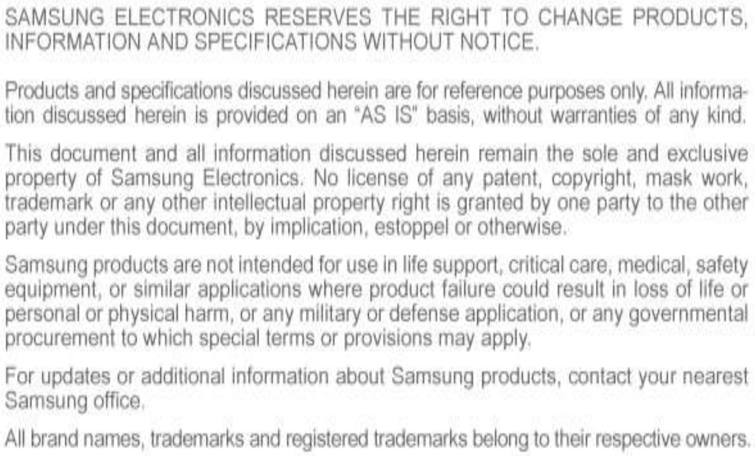
S3FN429

32-bit CMOS Microcontrollers

Revision 1.20

May 2012

User's Manual



SAMSUNG ELECTRONICS RESERVES THE RIGHT TO CHANGE PRODUCTS, INFORMATION AND SPECIFICATIONS WITHOUT NOTICE.

Products and specifications discussed herein are for reference purposes only. All information discussed herein is provided on an "AS IS" basis, without warranties of any kind.

This document and all information discussed herein remain the sole and exclusive property of Samsung Electronics. No license of any patent, copyright, mask work, trademark or any other intellectual property right is granted by one party to the other party under this document, by implication, estoppel or otherwise.

Samsung products are not intended for use in life support, critical care, medical, safety equipment, or similar applications where product failure could result in loss of life or personal or physical harm, or any military or defense application, or any governmental procurement to which special terms or provisions may apply.

For updates or additional information about Samsung products, contact your nearest Samsung office.

All brand names, trademarks and registered trademarks belong to their respective owners.

SAMSUNG ELECTRONICS

Important Notice

Samsung Electronics Co. Ltd. ("Samsung") reserves the right to make changes to the information in this publication at any time without prior notice. All information provided is for reference purpose only. Samsung assumes no responsibility for possible errors or omissions, or for any consequences resulting from the use of the information contained herein.

This publication on its own does not convey any license, either express or implied, relating to any Samsung and/or third-party products, under the intellectual property rights of Samsung and/or any third parties.

Samsung makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Samsung assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation any consequential or incidental damages.

Customers are responsible for their own products and applications. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by the customer's technical experts.

Samsung products are not designed, intended, or authorized for use in applications intended to support or sustain life, or for any other application in which the failure of the Samsung product could reasonably be expected to create a situation where personal injury or death may occur. Customers acknowledge and agree that they are solely responsible to meet all other legal and regulatory requirements regarding their applications using Samsung products notwithstanding any information provided in this publication. Customer shall indemnify and hold Samsung and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim (including but not limited to personal injury or death) that may be associated with such unintended, unauthorized and/or illegal use.

WARNING No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electric or mechanical, by photocopying, recording, or otherwise, without the prior written consent of Samsung. This publication is intended for use by designated recipients only. This publication contains confidential information (including trade secrets) of Samsung protected by Competition Law, Trade Secrets Protection Act and other related laws, and therefore may not be, in part or in whole, directly or indirectly publicized, distributed, photocopied or used (including in a posting on the Internet where unspecified access is possible) by any unauthorized third party. Samsung reserves its right to take any and all measures both in equity and law available to it and claim full damages against any party that misappropriates Samsung's trade secrets and/or confidential information.

警 告 本文件仅向经韩国三星电子株式会社授权的人员提供，其内容含有商业秘密保护相关法规规定并受其保护的三星电子株式会社商业秘密，任何直接或间接非法向第三人披露、传播、复制或允许第三人使用该文件全部或部分内容的行为（包括在互联网等公开媒介刊登该商业秘密而可能导致不特定第三人获取相关信息的行为）皆为法律严格禁止。此等违法行为一经发现，三星电子株式会社有权根据相关法规对其采取法律措施，包括但不限于提出损害赔偿请求。

Samsung Electronics Co., Ltd.
San #24 Nongseo-Dong, Giheung-Gu
Yongin-City, Gyeonggi-Do, Korea 446-711

Contact Us: como.jin@samsung.com
TEL: (82)-(31)-209-4956
FAX: (82)-(31)-8000-2005

Home Page: http://www.samsungsemi.com

Trademarks

All brand names, trademarks and registered trademarks belong to their respective owners.

- Exynos, Exynos4210, FlexOneNAND, and OneNAND are trademarks of Samsung Electronics.
- ARM, Jazelle, TrustZone, and Thumb are registered trademarks of ARM Limited. Cortex, ETM, ETB, Coresight, ISA, and Neon are trademarks of ARM Limited.
- Java is a trademark of Sun Microsystems, Inc.
- SD is a registered trademark of Toshiba Corporation.
- MMC and eMMC are trademarks of MultiMediaCard Association.
- JTAG is a registered trademark of JTAG Technologies, Inc.
- Synopsys is a registered trademark of Synopsys, Inc.
- I2S is a trademark of Phillips Electronics.
- I2C is a trademark of Phillips Semiconductor Corp.
- MIPI and Slimbus are registered trademarks of the Mobile Industry Processor Interface (MIPI) Alliance.

All other trademarks used in this publication are the property of their respective owners.

Chip Handling Guide

Precaution against Electrostatic Discharge

When using semiconductor devices, ensure that the environment is protected against static electricity:

1. Wear antistatic clothes and use earth band.
2. All objects that are in direct contact with devices must be made up of materials that do not produce static electricity.
3. Ensure that the equipment and work table are earthed.
4. Use ionizer to remove electron charge.

Contamination

Do not use semiconductor products in an environment exposed to dust or dirt adhesion.

Temperature/Humidity

Semiconductor devices are sensitive to:

- Environment
- Temperature
- Humidity

High temperature or humidity deteriorates the characteristics of semiconductor devices. Therefore, do not store or use semiconductor devices in such conditions.

Mechanical Shock

Do not to apply excessive mechanical shock or force on semiconductor devices.

Chemical

Do not expose semiconductor devices to chemicals because exposure to chemicals leads to reactions that deteriorate the characteristics of the devices.

Light Protection

In non- Epoxy Molding Compound (EMC) package, do not expose semiconductor IC to bright light. Exposure to bright light causes malfunctioning of the devices. However, a few special products that utilize light or with security functions are exempted from this guide.

Radioactive, Cosmic and X-ray

Radioactive substances, cosmic ray, or X-ray may influence semiconductor devices. These substances or rays may cause a soft error during a device operation. Therefore, ensure to shield the semiconductor devices under environment that may be exposed to radioactive substances, cosmic ray, or X-ray.

EMS (Electromagnetic Susceptibility)

Strong electromagnetic wave or magnetic field may affect the characteristic of semiconductor devices during the operation under insufficient PCB circuit design for Electromagnetic Susceptibility (EMS).

Revision History

Revision No.	Date	Description	Author(s)
1.00	Jan. 17, 2012	• Promoted from preliminary to V1.0 and released	YH Jin
1.10	Feb. 22, 2012	• Added guide notes (caution) related to the pin (XIN and XOUT) connection for unused external oscillator. • Page 2-7, 5-8, 5-21, 5-41, 5-59, and 19-7 • Corrected the errata. • The offset address of PPD_SCTVHR register in page 7-37 was fixed from 0x006C to 0x005C.	MCU solution part
1.20	May. 16, 2012	• Added guide notes related to the pin (OPAMP output pin) configuration when OPAMP is enabled. Page 12-5, 13-8	MCU solution part

Table of Contents

List of Figures

List of Tables

Table Number **Title** **Page Number**

List of Conventions

Register RW Access Type Conventions

Type	Definition	Description
R	Read Only	The application has permission to read the Register field. Writes to read-only fields have no effect.
W	Write Only	The application has permission to write in the Register field.
RW	Read & Write	The application has permission to read and writes in the Register field. The application sets this field by writing 1'b1 and clears it by writing 1'b0.

Register Value Conventions

Expression	Description
x	Undefined bit
X	Undefined multiple bits
?	Undefined, but depends on the device or pin status
Device dependent	The value depends on the device
Pin value	The value depends on the pin status

Reset Value Conventions

Expression	Description
0	Clears the register field
1	Sets the register field
x	Don't care condition

Warning: Some bits of control registers are driven by hardware or write operation only. As a result the indicated reset value and the read value after reset might be different.

1 Product Overview

1.1 Introduction

The S3FN429 user manual describes the complete reference specification of S3FN429.

The improved features of S3FN429 are:

- ARM Cortex-M0 Core
- Built-in 32 KB flash memory
- Internal 2 KB SRAM for stack, data memory, or code memory
- 32 General Purpose IOs (GPIO)
- Operating temperature: – 40 to 105 °C
- Operating voltage range: 2.5 to 5.5 V
- Interrupt controller: Dynamically reconfigurable Nested Vectored Interrupt Controller (NVIC)
- Clock and Power Management Controller (CM)
- Watchdog Timer (WDT)
- 3 × 16-bit Timer/Counter (TC)
- 4 × 16-bit Pulse Width Modulation (PWM)
- 1 × 16-bit Pulse Position Decoder (PPD)
- 1 × 3-Phase Inverter Motor Controller (IMC)
- 1 × USART, 1 × SPI
- 12-bit Analog to Digital Converter (ADC) with external input AINx 10 Channel
- 1 × Operational Amplifier (OP-AMP)
- 4 × Comparator (COMP)
- Supports idle and stop mode for reducing current

1.2 Features

S3FN429 device contains:

- CPU
- Memory
- Interrupt Controller
- Clock Manager (CM)
- Watchdog Timer (WDT)
- 16-bit Timer/Counter (TC)
- Pulse Width Modulation (PWM)
- Pulse Position Decoder (PPD)
- Inverter Motor Controller (IMC)
- Universal Synchronous/Asynchronous Receiver/Transmitter (USART)
- Serial Peripheral Interface (SPI)
- Analog to Digital Converter (ADC)
- Operational Amplifier (OP-AMP)
- Comparator (COMP)
- General Purpose IO (GPIO)
- Two Low Power Modes
- Power-On Reset (POR)
- Low Voltage Detection (LVD)
- Phase-Locked Loop (PLL)
- Operating Voltage Range
- Operating Frequency Range
- Operating Temperature Range
- Available in 44 QFP Package

CPU

The CPU contains:

- 32-bit RISC ARM Cortex-M0 Core
- Serial Wire Debug (SWD)

Memory

The Memory supports:

- 32 KB internal program full flash
- 2 KB internal SRAM
- Only little-endian

Interrupt Controller

The Interrupt Controller supports:

- NVIC of Cortex-M0
- Dynamically reconfigurable interrupt priority (four priority levels)
- 32 device interrupt vectors
- Selectable eight External Interrupts EXI (n)
- Programmable eight wake-up sources from stop

Clock Manager (CM)

The Clock Manager (CM) supports:

- External Main Oscillator Clock (EMCLK) 1 to 12 MHz
- Internal Main Oscillator Clock (IMCLK) 40 MHz
- Phase-Locked Loop (PLL) control , from 12 to 40 MHz
- Clock monitor to detect an external main oscillator failure
- Low power mode (IDLE/STOP) by clock gating control
- Programmable clock dividers (SDIV and PDIV)
- Reset management
- Basic timer for reset generation

Watchdog Timer (WDT)

The Watchdog Timer (WDT) contains:

- Configurable microcontroller reset event
- Programmable 16-bit down counter

16-bit Timer/Counter (TC)

The 16-bit Timer/Counter (TC) contains:

- Operation in an interval, capture, match, and overflow or PWM mode
- Match and overflow interrupt
- Selectable an internal or external clock

Pulse Width Modulation (PWM)

The Pulse Width Modulation (PWM) supports:

- 16-bit PWM signal generation
- Interval mode
- Programmable idle level
- Extension PWM function

Pulse Position Decoder (PPD)

The Pulse Position Decoder (PPD) contains:

- 3 input signals are:
  - PHASEA
  - PHASEB
  - PHASEZ
- Position counter and position capture timer
- Speed counter and speed capture timer
- Up/Down counter

Inverter Motor Controller (IMC)

The Inverter Motor Controller (IMC) supports:

- 3-Phase 16-bit PWM generation
- Programmable dead time insertion
- Output off-control by fault input signals
- ADC conversion start signal generation

Universal Synchronous/Asynchronous Receiver/Transmitter (USART)

The Universal Synchronous/Asynchronous Receiver/Transmitter (USART) supports:

- 5, 6, 7, 8 and 9-bit data length
- Programmable baud rate generator
- Parity, framing, and overrun error detection
- Loop-back mode
- Full duplex
- Idle flag for J1587 protocol
- Smart-card protocol: error signaling and re-transmission

Serial Peripheral Interface (SPI)

The Serial Peripheral Interface (SPI) supports:

- Programmable data frame from 4 to 16-bit
- Master and slave mode
- Programmable clock pre-scale
- Separate 8/8 × 16-bit width Transmit/Receive First-In First-Out (FIFO)

Analog to Digital Converter (ADC)

The Analog to Digital Converter (ADC) contains:

- 12-bit resolution
- 10 input pins for conversion input signal, AIN[10:1]
- ADC has a conversion channel for OP-AMP (AIN0)
- Various conversion start sources - START (software), ADTRG (external input signal), and internal peripherals (IMC, Timer)

Operational Amplifier (OP-AMP)

The Operational Amplifier (OP-AMP) supports:

- The operation with ADC
- Both an internal and external gain control

Comparator (COMP)

The Comparator contains:

- Configurable reference voltage selection
- Control signal for IMC output or input signal for PPD

General Purpose IO (GPIO)

The General Purpose IO (GPIO) contains:

- Input or Output configuration
- Output open-drain/push-pull configuration
- GPIO interrupt

Low Power Modes

The Two Low Power Mode supports:

- IDLE: Only CPU clock stops
- STOP: All clocks stop
- Fast wake-up with internal main oscillator (from stop mode to normal mode)
- Programmable external event/interrupt sources for wake-up

Power-On Reset (POR)

The Power-On Reset (POR) contains:

- Built-in POR circuit

Low Voltage Detection (LVD)

The Low Voltage Detection (LVD) contains:

- LVD for reset with configurable voltage levels
- LVD for interrupt with configurable voltage levels
- LVD reset/interrupt enable/disable can be controllable

Phase-Locked Loop (PLL)

The Phase-Locked Loop (PLL) contains:

- Input clock source: EMCLK
- Input frequency: 1 to 12 MHz
- Output frequency: 12 to 40 MHz

Operating Voltage Range

The Operating Voltage Range contains:

- 2.5 to 5.5 V

Operating Frequency Range

The Operating Frequency Range contains:

- Up to 40 MHz
- EMCLK: 1 to 12 MHz
- IMCLK: 40 MHz
- PLL clock: 12 to 40 MHz

Operating Temperature Range

The Operating Temperature Range contains:

- – 40 °C to 105 °C

Package

The Package contains:

- Available in 44-QFP
- Width × Length: 10.0 mm × 10.0 mm
- Lead pitch: 0.8 mm

1.3 Block Diagram

Figure 1-1 illustrates the block diagram of S3FN429 device.

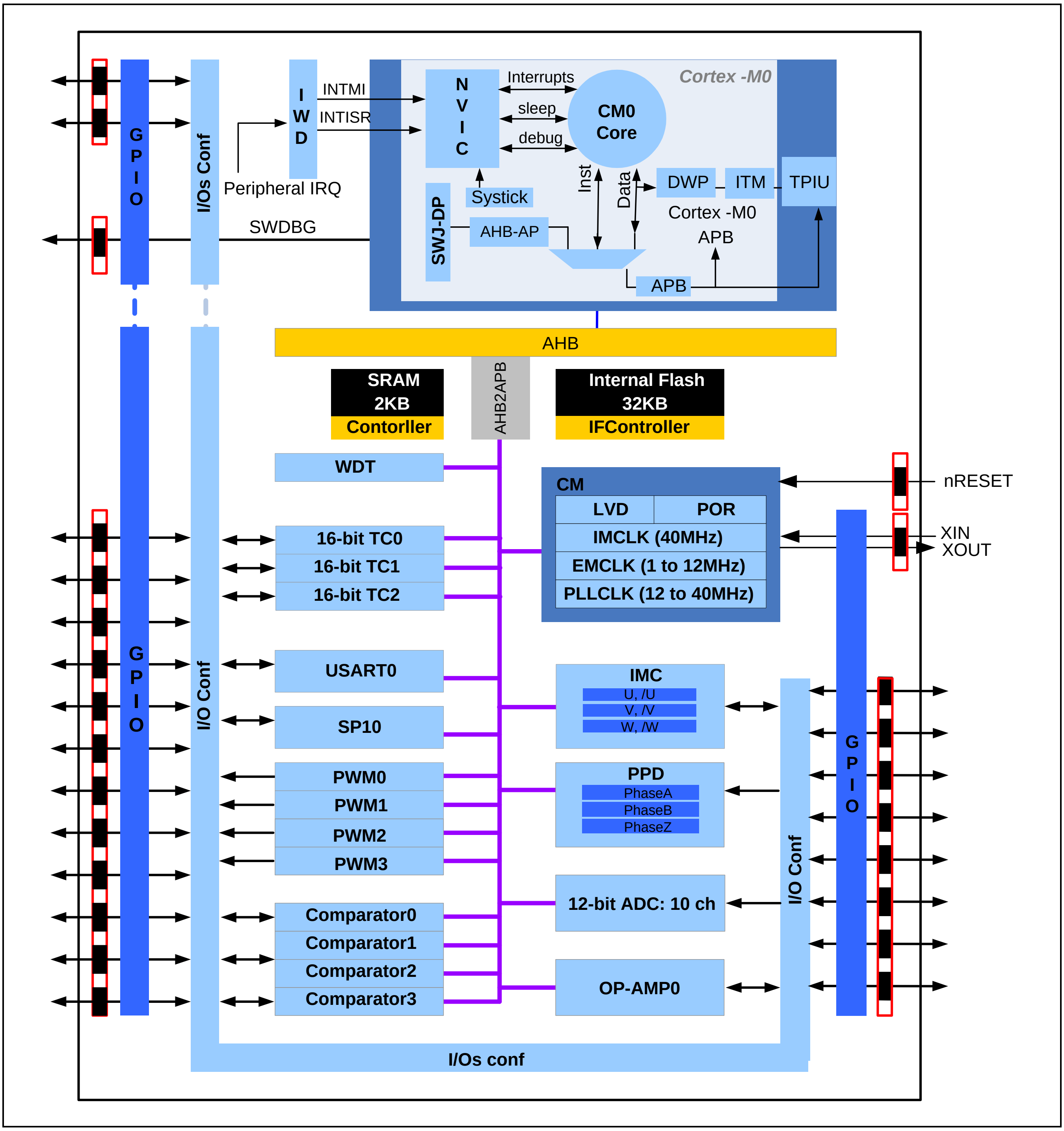


Figure 1-1 Block Diagram

2 Pin Configuration

2.1 Overview

The Pin Configuration chapter describes the pin information of S3FN429.

This chapter includes:

- Pin map diagram
- Pin assignment table
- Mirror pins
- Pin description
  - Power pins
  - System pins
  - Function pins
  - Debug interface pins
  - Flash serial program pins
- Pin circuit type
  - Type A
  - Type B
  - Mode pins
  - nReset pin

2.2 Pin Map

Figure 2-1 illustrates the pin map for S3FN429 device.

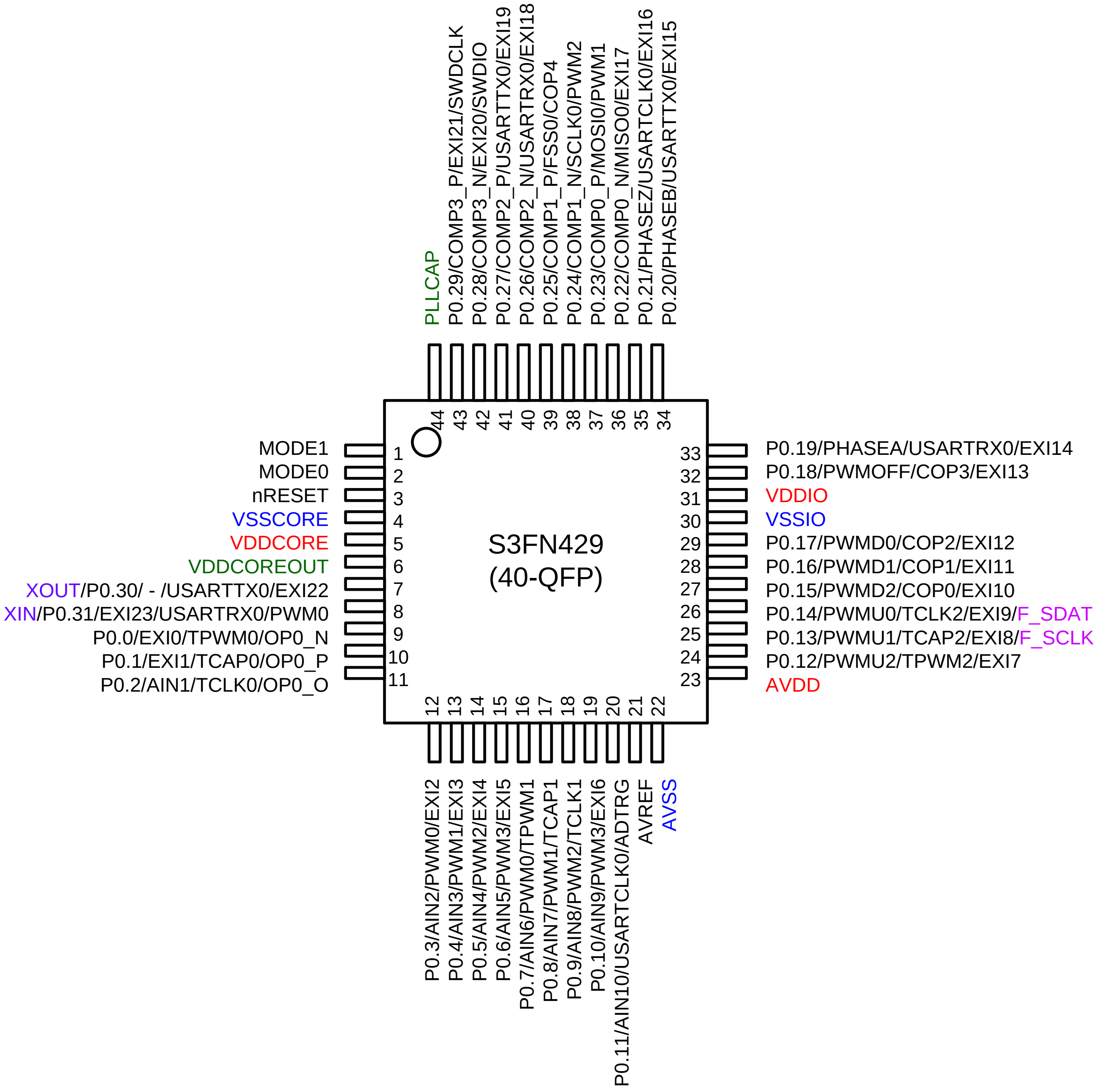

S3FN429
(40-QFP)
1 MODE1
2 MODE0
3 nRESET
4 VSSCORE
5 VDDCORE
6 VDDCOREOUT
7 XOUT/P0.30/ - /USARTTX0/EXI22
8 XIN/P0.31/EXI23/USARTRX0/PWM0
9 P0.0/EXI0/TPWM0/OP0_N
10 P0.1/EXI1/TCAP0/OP0_P
11 P0.2/AIN1/TCLK0/OP0_O
12 P0.3/AIN2/PWM0/EXI2
13 P0.4/AIN3/PWM1/EXI3
14 P0.5/AIN4/PWM2/EXI4
15 P0.6/AIN5/PWM3/EXI5
16 P0.7/AIN6/PWM0/TPWM1
17 P0.8/AIN7/PWM1/TCAP1
18 P0.9/AIN8/PWM2/TCLK1
19 P0.10/AIN9/PWM3/EXI6
20 P0.11/AIN10/USARTCLK0/ADTRG
21 AVREF
22 AVSS
23 AVDD
24 P0.12/PWMU2/TPWM2/EXI7
25 P0.13/PWMU1/TCAP2/EXI8/F_SCLK
26 P0.14/PWMU0/TCLK2/EXI9/F_SDAT
27 P0.15/PWMD2/COP0/EXI10
28 P0.16/PWMD1/COP1/EXI11
29 P0.17/PWMD0/COP2/EXI12
30 VSSIO
31 VDDIO
32 P0.18/PWMOFF/COP3/EXI13
33 P0.19/PHASEA/USARTRX0/EXI14
34 P0.20/PHASEB/USARTTX0/EXI15
35 P0.21/PHASEZ/USARTCLK0/EXI16
36 P0.22/COMP0_N/MISO0/EXI17
37 P0.23/COMP0_P/MOSI0/PWM1
38 P0.24/COMP1_N/SCLK0/PWM2
39 P0.25/COMP1_P/FSS0/COP4
40 P0.26/COMP2_N/USARTRX0/EXI18
41 P0.27/COMP2_P/USARTTX0/EXI19
42 P0.28/COMP3_N/EXI20/SWDIO
43 P0.29/COMP3_P/EXI21/SWDCLK
44 PLLCAP

Figure 2-1 Pin Map Diagram

NOTE: If you use OP amp, AIN1 should not be used for ADC input.

2.3 Pin Assignment

Table 2-1 describes the pin assignment details.

Table 2-1 Pin Assignment by Pin Number Order

Pin Number	1st	2nd	3rd	4th	Flash
1	MODE1	MODE1	MODE1	MODE1	MODE1
2	MODE0	MODE0	MODE0	MODE0	MODE0
3	nRESET	nRESET	nRESET	nRESET	nRESET
4	VSSCORE	VSSCORE	VSSCORE	VSSCORE	VSSCORE
5	VDDCORE	VDDCORE	VDDCORE	VDDCORE	VDDCORE
6	VDDCOREOUT	VDDCOREOUT	VDDCOREOUT	VDDCOREOUT	VDDCOREOUT
7	XOUT/P0.30	XOUT/–	XOUT/USARTTX0	XOUT/EXI22	–
8	XIN/P0.31	XIN/EXI23	XIN/USARTRX0	XIN/PWM0	–
9	P0.0	EXI0	TPWM0	OP0_N	–
10	P0.1	EXI1	TCAP0	OP0_P	–
11	P0.2	AIN1	TCLK0	OP0_O	–
12	P0.3	AIN2	PWM0	EXI2	–
13	P0.4	AIN3	PWM1	EXI3	–
14	P0.5	AIN4	PWM2	EXI4	–
15	P0.6	AIN5	PWM3	EXI5	–
16	P0.7	AIN6	PWM0	TPWM1	–
17	P0.8	AIN7	PWM1	TCAP1	–
18	P0.9	AIN8	PWM2	TCLK1	–
19	P0.10	AIN9	PWM3	EXI6	–
20	P0.11	AIN10	USARTCLK0	ADTRG	–
21	AVREF	AVREF	AVREF	AVREF	–
22	AVSS	AVSS	AVSS	AVSS	–
23	AVDD	AVDD	AVDD	AVDD	–
24	P0.12	PWMU2	TPWM2	EXI7	–
25	P0.13	PWMU1	TCAP2	EXI8	F_SCLK
26	P0.14	PWMU0	TCLK2	EXI9	F_SDAT
27	P0.15	PWMD2	COP0	EXI10	–
28	P0.16	PWMD1	COP1	EXI11	–
29	P0.17	PWMD0	COP2	EXI12	–
30	VSSIO	VSSIO	VSSIO	VSSIO	VSSIO
31	VDDIO	VDDIO	VDDIO	VDDIO	VDDIO
32	P0.18	PWMOFF	COP3	EXI13	–

Pin Number	1st	2nd	3rd	4th	Flash
33	P0.19	PHASEA	USARTRX0	EXI14	–
34	P0.20	PHASEB	USARTTX0	EXI15	–
35	P0.21	PHASEZ	USARTCLK0	EXI16	–
36	P0.22	COMP0_N	MISO0	EXI17	–
37	P0.23	COMP0_P	MOSI0	PWM1	–
38	P0.24	COMP1_N	SCLK0	PWM2	–
39	P0.25	COMP1_P	FSS0	COP4	–
40	P0.26	COMP2_N	USARTRX0	EXI18	–
41	P0.27	COMP2_P	USARTTX0	EXI19	–
42	P0.28	COMP3_N	EXI20	SWDIO	–
43	P0.29	COMP3_P	EXI21	SWDCLK	–
44	PLLCAP	PLLCAP	PLLCAP	PLLCAP	–

2.4 Mirror Pins

Several pins have mirror pins. This helps in flexibility in arranging. These pins are, however, defined for one instance (IP) only.

Table 2-2 describes the summary of mirror pins.

Table 2-2 Summary of Mirror Pins

IP (Instance)	Pin
PWM0	P0.31/EXI23/USARTRX0/PWM0 P0.3/AIN2/PWM0/EXI2 P0.7/AIN6/PWM0/TPWM1
PWM1	P0.4/AIN3/PWM1/EXI3 P0.8/AIN7/PWM1/TCAP1 P0.23/COMP0_P/MOSI0/PWM1
PWM2	P0.5/AIN4/PWM2/EXI4 P0.9/AIN8/PWM2/TCLK1 P0.24/COMP1_N/SCLK0/PWM2
PWM3	P0.6/AIN5/PWM3/EXI5 P0.10/AIN9/PWM3/EXI6
USART0	P0.30/–/USARTTX0/EXI22 P0.31/EXI23/USARTRX0/PWM0 P0.11/AIN10/USARTCLK0/ADTRG P0.19/PHASEA/USARTRX0/EXI14 P0.20/PHASEB/USARTTX0/ExI15 P0.21/PHASEZ/USARTCLK0/EXI16 P0.26/COMP2_N/USARTRX0/EXI18 P0.27/COMP2_P/USARTTX0/EXI19

2.5 Pin Description

The pin description section describes:

- Power Pins
- System Pins
- Function Pins
- Debug Interface Pins
- Flash Serial Program Pins

NOTE:

1. D/A: Digital or Analog, D: Digital, A: Analog
2. I/O: Input or Output, I: Input, O: Output
3. PIN[Z:A]: The same function pin group, PIN.A, PIN.B, to ,PIN.Z

2.5.1 Power Pins

Table 2-3 describes the power pin description.

Table 2-3 Power Pin Description

Module	Pin Name	Function Description	Comments
Power	VDDCORE	Core DC Supply Voltage	See recommended operating condition
	VSSCORE	Core Ground Voltage	–
	VDDIO	I/O DC Supply Voltage	See recommended operating condition
	VSSIO	I/O Ground Voltage	–
	AVDD	ADC Supply Voltage	See recommended operating condition
	AVSS	ADC Ground Voltage	–
	AVREF	ADC Reference Input Voltage	–
	VDDCOREOUT	Cap Output Port from Internal Regulator	Connected to GND through a 0.1 μF capacitor (From internal regulator)
	PLLCAP	Cap Output Port for PLL	Connected to GND through a 220 pF capacitor for PLL

2.5.2 System Pins

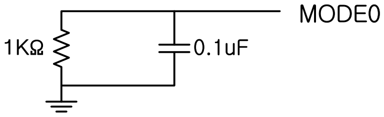
Table 2-4 System Pin Description

<table>
<tr><th>Module</th><th>Pin Name</th><th>Function</th><th>D/A</th><th>I/O</th><th>Comments</th></tr>
<tr><td>RESET</td><td>nRESET</td><td>Hardware Reset Input
This nRESET pin contains an internal pull up resistor typical 250 kΩ. Setting this pin to low level initialize the internal state of the device. Thereafter, setting the input to high release the reset status. The S3FN429 waits for the system clock to be stabled, and the PC to the reset interrupt vector. Internal Reset is generated after clock stabilization.</td><td>D</td><td>I</td><td>Internal filter</td></tr>
<tr><td rowspan="3">CM</td><td>XIN [1]</td><td>External Main Oscillator Input</td><td>A</td><td>I</td><td>–</td></tr>
<tr><td>XOUT [2]</td><td>External Main Oscillator Output</td><td>A</td><td>O</td><td>–</td></tr>
<tr><td>COP[4:0]</td><td>Internal Clock Out Port
COP0 = EMCLK/8
COP1 = IMCLK/8
COP2 = PLLCLK/8
COP3 = CORECLK/8
COP4 = PCLK/8
These pins help you to check clock status when your system has some problem or you are sure about programmed clock configuration.</td><td>D</td><td>O</td><td>–</td></tr>
<tr><td rowspan="2">MODE</td><td>MODE1</td><td>Mode Selection
MODE1 is for a factory test, so it should be connected with ground in user mode.</td><td>D</td><td>I</td><td>Internal pull-down</td></tr>
<tr><td>MODE0</td><td>This pin decides either normal mode or tool mode. If on-board writing (programming in tool mode) is needed, 0.1 uF capacitor should be connected between MODE0 pin and ground. Parallel 1 kΩ resistor should be connected for better noise immunity. [3]
But if you use S3FN429 only in normal mode without on-board writing, this pin should be connected to ground directly.<table><tr><th>Mode 1</th><th>Mode 0</th><th>Mode Setting</th></tr><tr><td>0</td><td>0</td><td>Normal Mode</td></tr><tr><td>0</td><td>1</td><td>Tool Mode</td></tr></table></td><td>D</td><td>I</td><td>Internal pull-down</td></tr>
</table>

Caution: If you don't the external main-oscillator when pin7 and pin8 are defined XOUT and XIN function, you should tie XIN to ground. XOUT should be opened.

NOTE:

1. After reset, the default function of pin8 can be one of two. One is XIN, the other is P0.30. The hardware reset condition is decided by smart option. That means you need to program the chip configuration smart option if you want to change that pin's reset function. You can program and erase the smart option in normal or tool mode. Refer to Chapter 9. Internal Flash Controller IFC).
2. After reset, the default function of pin7 can be one of two. One is XOUT, the other is P0.31. The hardware reset condition is decided by smart option. That means you need to program the chip configuration smart option if you want to change that pin's reset function. You can program and erase the smart option in normal or tool mode. Refer to Chapter 9. Internal Flash Controller (IFC).
3.



2.5.3 Function Pins

Table 2-5 describes pin description.

Table 2-5 Function Pin Description

Module	Pin Name	Function	D/A	I/O	Comments
GPIO	P0[31:0]	General purpose I/O multiplexed	D	I/O	Multiplexed with peripheral module
Interrupt	EXI[23:0]	External interrupt request	D	I	–
16-bit timer	TCLK[2:0]	External clock input for timer	D	I	–
	TCAP[2:0]	Capture input for timer	D	I	–
	TPWM[2:0]	PWM output for timer	D	O	–
PWM	PWM[3:0]	Pulse width modulation output	D	O	–
IMC	PWMU[2:0]	PWM output for inverter motor	D	O	–
	PWMD[2:0]	PWM output for inverter motor	D	O	–
	PWMOFF	Input pin for PWM output off	D	I	–
PPD	PHASEA	Phase A input	D	I	–
	PHASEB	Phase B input	D	I	–
	PHASEZ	Phase Z input	D	I	–
SPI	SCLK0	SPI serial clock	D	I/O	–
	MISO0	Master in slave out	D	I/O	–
	MOSI0	Master out slave in	D	I/O	–
	FSS0	Frame or slave select (master) frame input (slave)	D	I/O	–
USART	USARTRX0	Received signal input	D	I	–
	USARTTX0	Transmit signal output	D	O	–
	USARTCLK0	Clock signal	D	I/O	–
ADC	AIN[10:1]	Analog Input channels	A	I	–
	ADTRG	ADC external Trigger input pin	D	I	–
OP-AMP	OP0_N	OP-AMP negative input	A	I	–
	OP0_P	OP-AMP positive input	A	I	–
	OP0_O	OP-AMP out	A	O	–
Comparator	COMP[3:0]_N	Comparator negative input	A	I	–
	COMP[3:0]_P	Comparator positive input	A	I	–

2.5.4 Debug Interface Pins

Table 2-6 Debug Interface Pin Description

Module	Pin Name	Function	D/A	I/O	Comments
DEBUG	SWDIO	Select/serial wire data input output	D	I/O	Internal pull-up 55 kΩ
	SWDCLK	Serial wire clock	D	I	–

2.5.5 Flash Serial Program Pins

Table 2-6 describes pin description. Refer to 9.2.3.2 Tool Program Mode for more details.

Table 2-7 Flash Serial Program Pin Description

Module	Pin Name	Function	D/A	I/O	Comments
FLASH	F_SDAT	Serial Data pin (output when reading, input when writing)	–	I/O	Input and push-pull output
	F_SCLK	Serial Clock	–	I	–

2.6 Pin Circuit Type

2.6.1 Block Diagram

Figure 2-2, *Figure 2-3*, *Figure 2-4*, and *Figure 2-5* illustrate the Type A, Type B, MODEx and nRESET pin circuits respectively.

2.6.1.1 Type A

Bi-directional buffer with B4 (NOTE) output driver and enables schmitt trigger cmos input with controllable 55K pull-up resistor and 250Ω analog input. The status of input data is high when this is an output mode.

NOTE: B4: Output Max. Operation Freq. 8 MHz

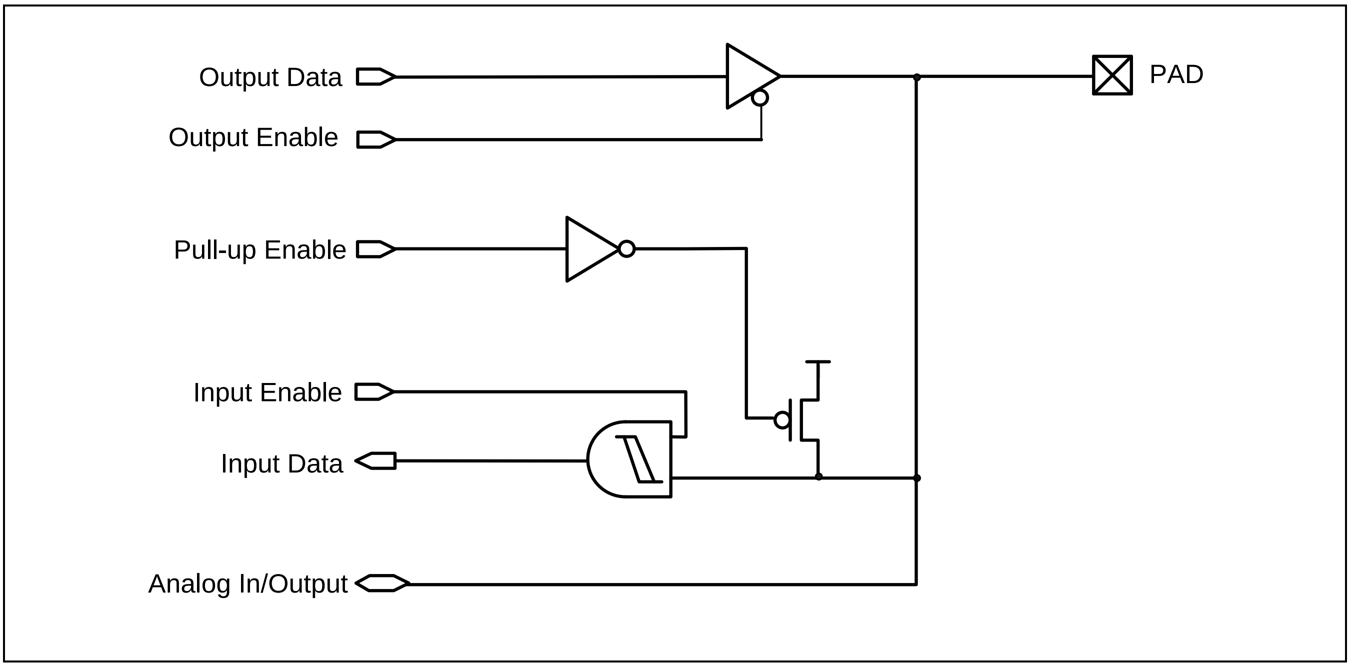


Figure 2-2 Pin Circuit Type A (P0.0 to P0.11 and P0.18 to P0.31)

2.6.1.2 Type B

Bi-directional buffer with Iol = 24 mA (@1 V in 5 V operation at 25 °C) output driver and enables schmitt trigger cmos input with controllable 55K pull-up resistor and 250 Ω analog input.

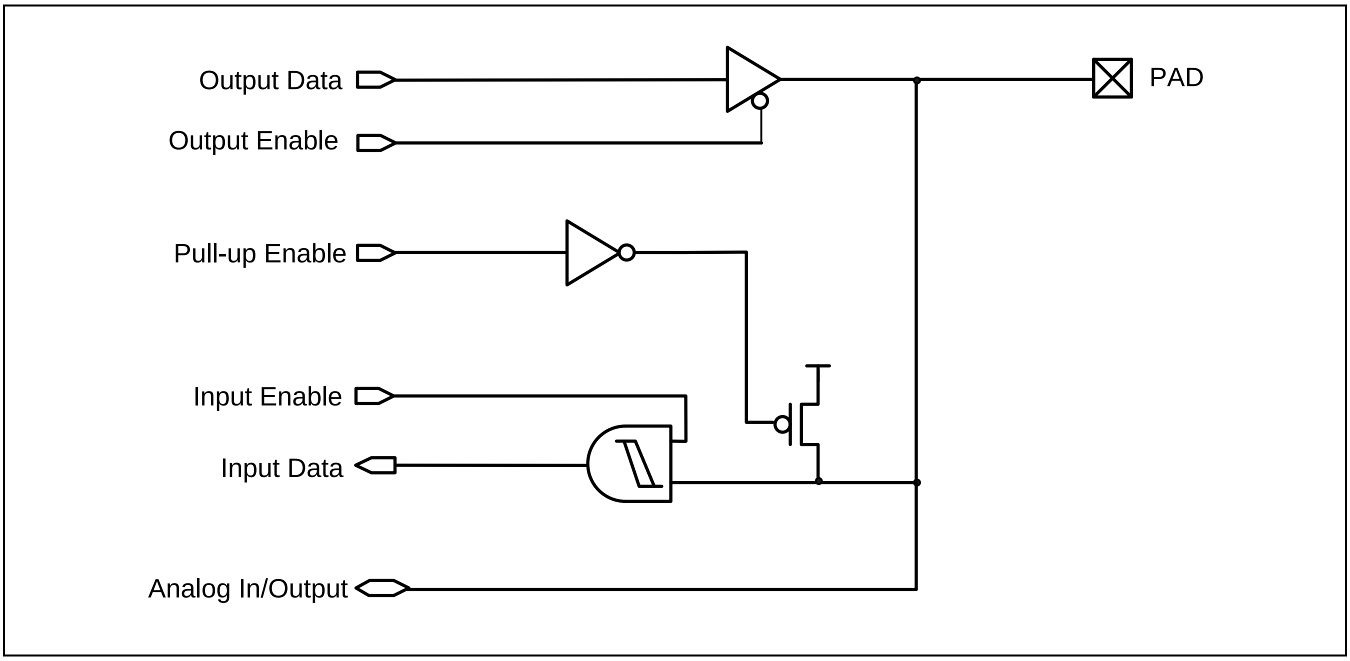


Figure 2-3 Pin Circuit Type B (P0.12 to P0.17)

2.6.1.3 MODE1 and MODE0

Schmitt trigger cmos input with 55K pull-down resistor

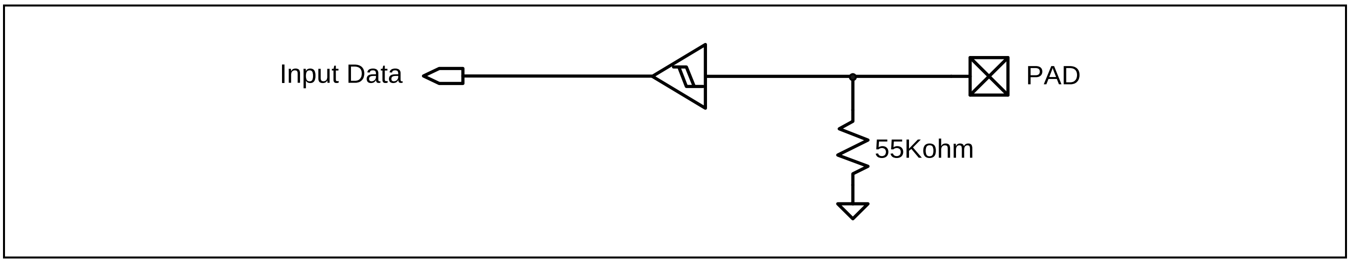


Figure 2-4 Pin Circuit Type C (MODE1 and MODE0)

2.6.1.4 nRESET

Bi-directional buffer with B8 (NOTE) output driver and enable schmitt trigger cmos input with always 250K pull-up resistor.

NOTE: B8: Output Max. Operating Freq. 15 MHz

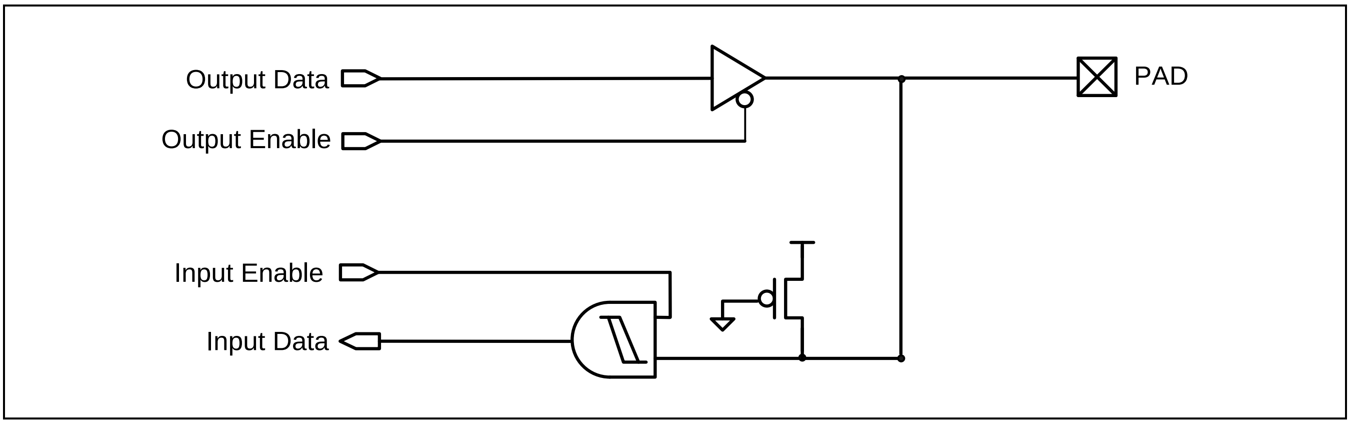


Figure 2-5 Pin Circuit Type D (nRESET)

3 System Memory Management

3.1 Overview

The System Memory Management chapter describes the system memory management for S3FN429.

The chapter includes:

- Memory map
- Special function register map. The register maps are:
  - Core special function register map
  - Peripheral special function register map

3.2 Default Memory Map

Table 3-1 describes the S3FN429 memory space allocation.

Table 3-1 S3FN429 Memory Map

Address	Memory
Reserved	Reserved
0xE00F_FFFF to 0xE000_0000	Cortex-M0 internal peripheral registers
Reserved	Reserved
0x400F_FFFF to 0x4000_0000	Special function registers
Reserved	Reserved
0x2000_07FF to 0x2000_0000	2 KB internal SRAM memory
Reserved	Reserved
0x0000_7FFF to 0x0000_0000	32 KB internal program flash memory (Including smart option area)

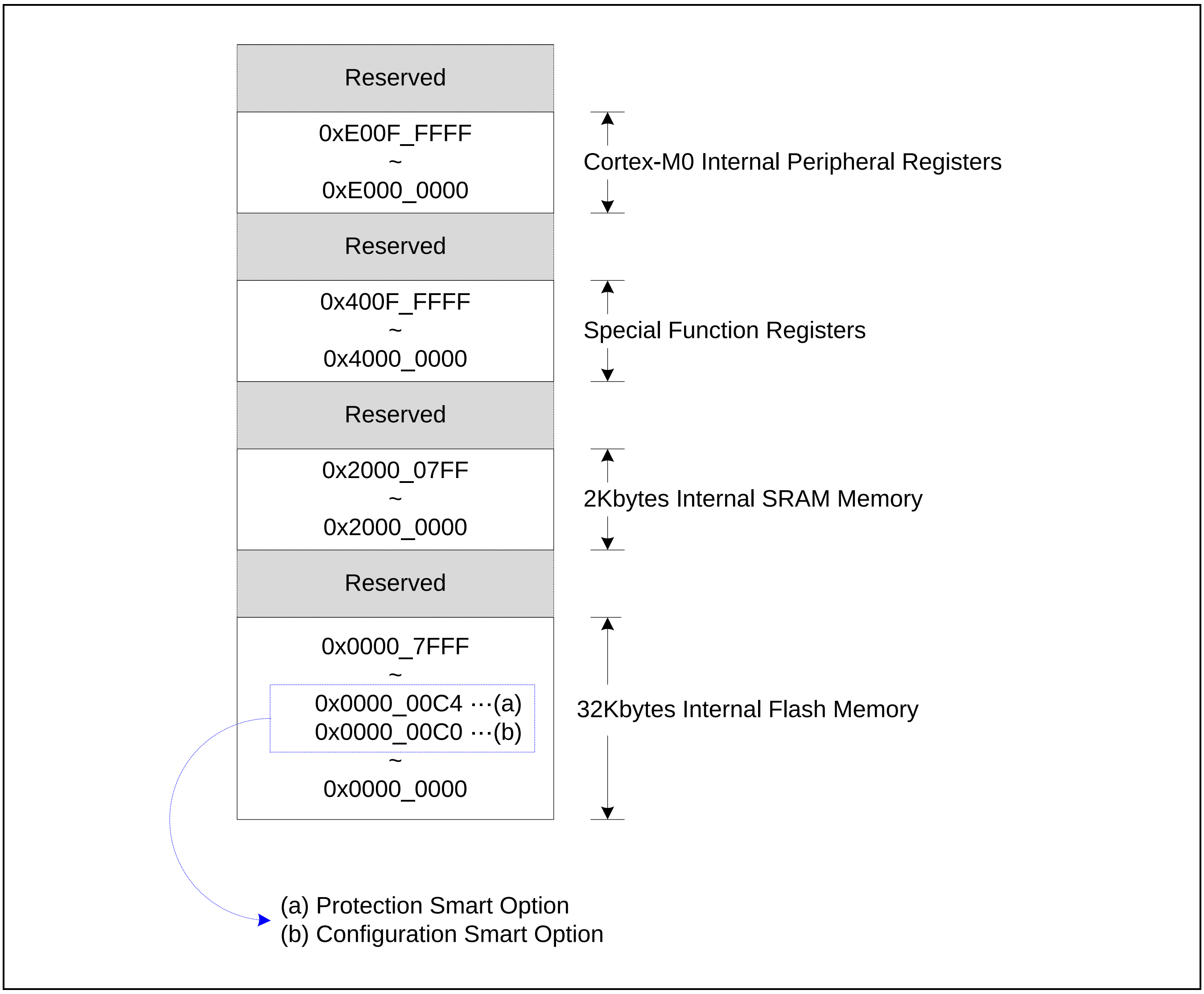

Reserved
0xE00F_FFFF
~
0xE000_0000
Cortex-M0 Internal Peripheral Registers
Reserved
0x400F_FFFF
~
0x4000_0000
Special Function Registers
Reserved
0x2000_07FF
~
0x2000_0000
2Kbytes Internal SRAM Memory
Reserved
0x0000_7FFF
~
0x0000_00C4 ···(a)
0x0000_00C0 ···(b)
~
0x0000_0000
32Kbytes Internal Flash Memory
(a) Protection Smart Option
(b) Configuration Smart Option

Figure 3-1 S3FN429 Memory Map

3.3 Special Function Register Map

The two types of special function register maps are:

- Core special function register map
- Peripheral special function register map

3.3.1 Core Special Function Register Map

Table 3-2 describes the core special function register map.

Table 3-2 Core Special Function Register Map

Base Address	Peripheral	Description
0xE00F_F000	ROM Table	ROM Memory Table
0xE004_2000	External PPB	Private Peripheral Bus
0xE004_1000	Reserved	–
0xE004_0000	TPIU	Trace Port Interface
0xE000_F000	Reserved	–
0xE000_E000	SCS	System Control Space
0xE000_3000	Reserved	–
0xE000_2000	FPB	Flash Patch and Break Pint
0xE000_1000	DWT	Data Watch Point and Trace
0xE000_0000	ITM	Instrumentation Trace Macro-cell

3.3.2 Peripheral Special Function Register Map

Table 3-3 describes the peripheral special function register map.

Table 3-3 Peripheral Memory Map

Module	Base Address	Peripheral	Description
PPD	0x400C_0000	PPD	Pulse Position Decoder
IMC	0x400B_0000	IMC	Inverter Motor Controller
SPI	0x4009_0000	SPI	Serial Peripheral Interface
USART	0x4008_0000	USART	Universal Synchronous/Asynchronous Receiver/Transmitter
PWM	0x4007_3000	PWM3	Pulse Width Modulation 3 (16-bit)
	0x4007_2000	PWM2	Pulse Width Modulation 2 (16-bit)
	0x4007_1000	PWM1	Pulse Width Modulation 1 (16-bit)
	0x4007_0000	PWM0	Pulse Width Modulation 0 (16-bit)
TC	0x4006_2000	TC2	Timer/Counter 2 (16-bit)
	0x4006_1000	TC1	Timer/Counter 1 (16-bit)
	0x4006_0000	TC0	Timer/Counter 0 (16-bit)
GPIO	0x4005_8000	IOCONF	IO Configuration
	0x4005_0000	GPIO0	General Purpose IO Group 0
Comparator	0x4004_2000	COMP	Comparator
OP-AMP	0x4004_1000	OP-AMP	Operational Amplifier
ADC	0x4004_0000	ADC	Analog to Digital Converter
WDT	0x4003_0000	WDT	Watchdog Timer (16-bit)
SYSTEM	0x4002_0000	CM	Clock Manager
MEMORY	0x4001_0000	IFC	Internal Flash Controller

4 Analog to Digital Converter (ADC)

4.1 Overview

This chapter describes the complete functional description of the Analog to Digital Converter (ADC) controller and the operation of design from the end user perspective.

4.1.1 Features

The distinctive features of ADC are:

- Resolution: 12-bit
- One input channel (AIN0) is assigned to Operational Amplifier (OP-AMP)
- 10 external input channels, AIN[10:1]
- Conversion start sources
  - Software start
  - External trigger input (ADTRG)
  - Internal peripheral trigger signals (Inverter Motor Controller (IMC) and Timer/Counter (TC))

- Maximum conversion rate: 5 MHz clock
- Analog input voltage range: 0 to V_{AVREF}
- Differential linearity error: ± 1.5 LSB (Max.) at 2.5 to 5.5 V
- Integral linearity error: ± 3.5 LSB (Max.) at 2.5 to 5.5 V

4.1.2 Pin Description

Table 4-1 describes the pin description of ADC.

Table 4-1 ADC Pin Description

Pin Name	Function	I/O Type	Comments
AVREF	Reference top voltage	Analog Input	–
AIN[10:1]	Analog inputs	Analog Input	–
ADTRG	External start trigger signal	Digital Input	–

NOTE: You should write "AIN1" value in ICNUM field of CCSRx register to convert analog signal asserted on AIN1 pin.

4.1.3 Block Diagram

Figure 4-1 illustrates the block diagram of ADC.

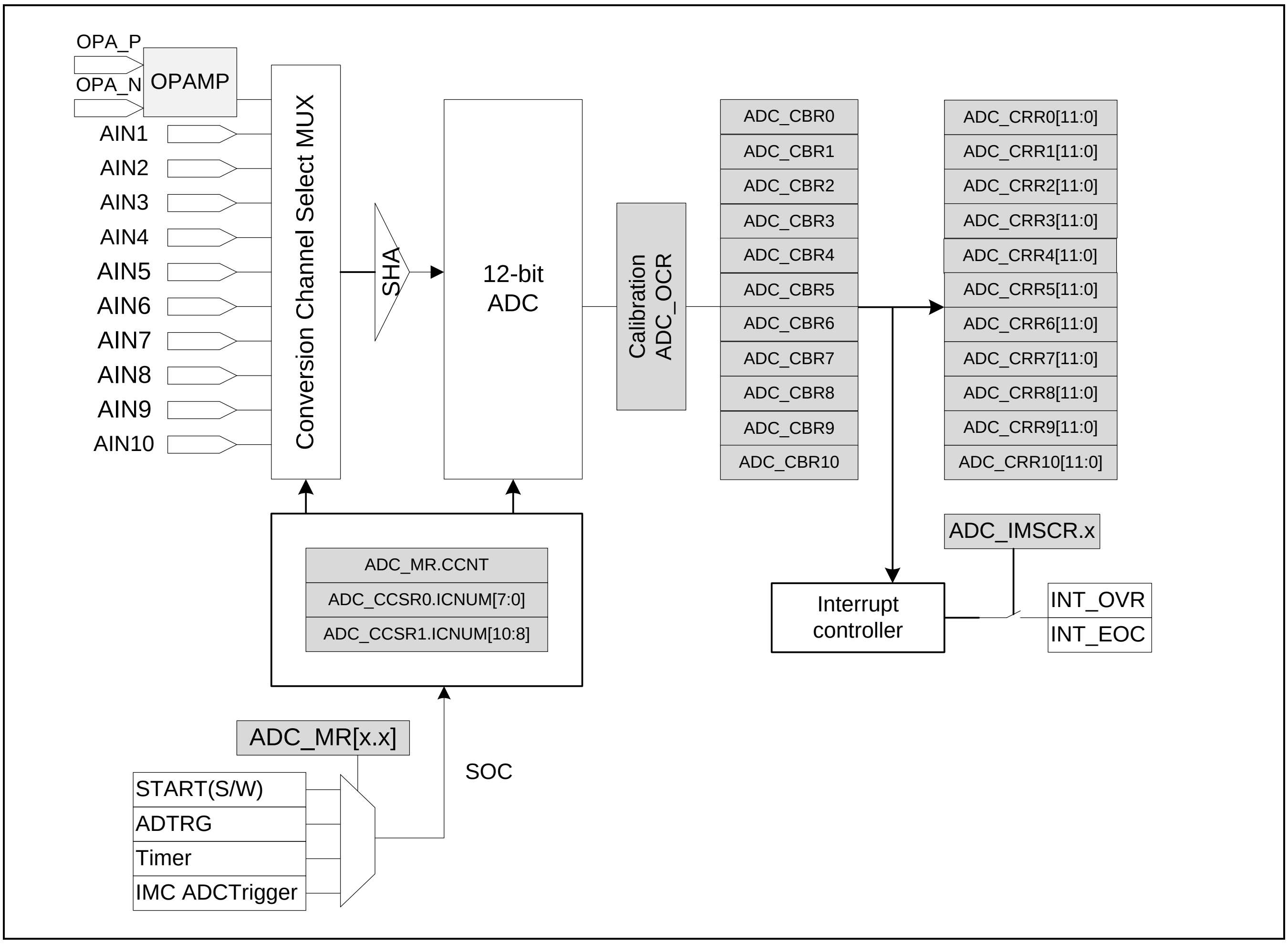


Figure 4-1 ADC Block Diagram

4.1.4 Input and Output

ADC operation converts the signal asserted on AINx input pin to digital data. The valid input signals are:

- Voltage range is from reference bottom to Top. ADC uses AINx function pins to convert an analog input source.
- Input signal range is from reference top to reference bottom.

Input Voltage Range: 0.0 V – V_{AVREF}

Reference Bottom = 0.0 V

Reference Top = V_{AVREF}

V_{AVREF} can be from 2.5 to 5.5 V (typical 5 V).

Assume V_{AVREF} = 5 V.

$$1\ LSB = \frac{Reference\ Top - Reference\ Bottom}{2^{Resolution}} = \frac{5.0V - 0.0V}{2^{12}} = \frac{5.0V}{4096} \approx 1.22mV$$

Table 4-2 describes the ADC Input and digital Output values.

Table 4-2 ADC Input and Digital Output

Index	AINx Input Voltage (V)	Digital Output (Binary)	Digital Output (HEX)
0	– 0.001220703	0000 0000 0000	0x000
1	0.001221 to 0.002441406	0000 0000 0001	0x001
2	0.002441 to 0.003662109	0000 0000 0010	0x002
–	–	–	–
4093	4.996338 – 4.997558594	1111 1111 1101	–
4094	4.997559 – 4.998779297	1111 1111 1110	0xFFE
4095	4.998779 to	1111 1111 1111	0xFFF

4.1.5 Clock Frequency and Conversion Time

ADC operation obtains clock from PCLK. ADC clock is not more than 5 MHz. This feature is effective up to a 5 MHz maximum ADC clock. If ADC operates with 5 MHz, the conversion per channel is done in the minimum time (1 μs).

Figure 4-2 illustrates the ADC operation timing.

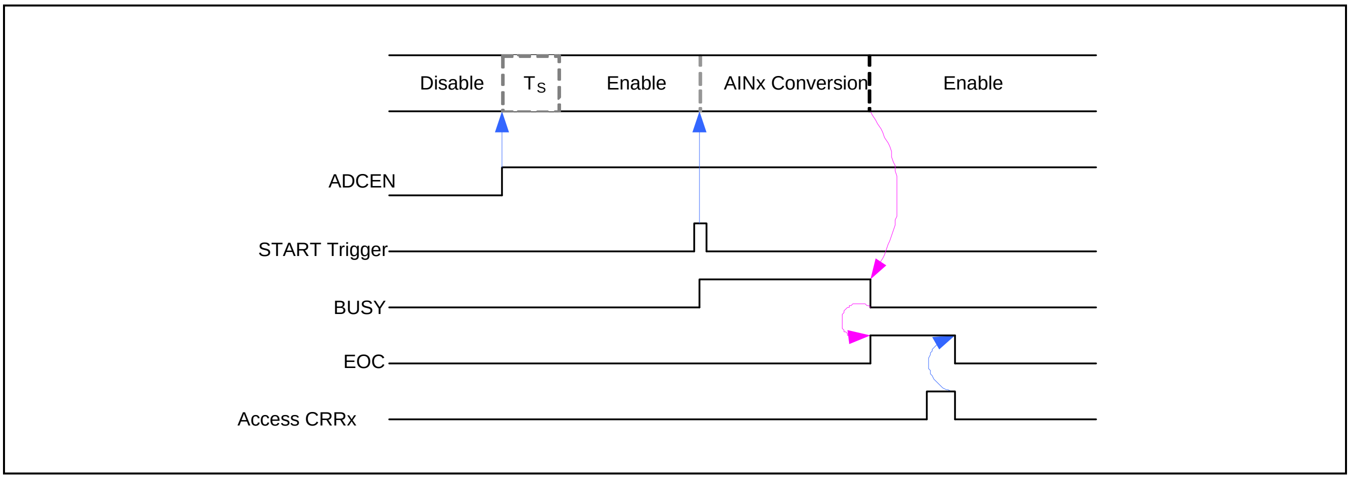


Figure 4-2 ADC Operation Timing Diagram

4.1.6 Conversion Mode

This section describes Conversion Mode and Sequence of ADC.

4.1.6.1 Conversion Sequence

A conversion sequence is a sequence of conversion of analog inputs. You can configure ADC block to make conversions of some of the 10 inputs and an input by OP-AMP in its own order. The setting of the CCNT field in the ADC Mode Register (ADC_MR) defines the length of the sequence (the number of conversions).

Table 4-3 describes the relation between the CCNT field and the number of conversion performed in a sequence.

Table 4-3 CCNT Values and the Number of Conversions

CCNT[3:0]	Count Value	Description
0000	1	One conversion operation of ICNUM0[3:0]
0001	2	2 times conversion operation from ICNUM0[3:0] to ICNUM1[3:0]
0010	3	3 times conversion operation from ICNUM0[3:0] to ICNUM2[3:0]
0011	4	4 times conversion operation from ICNUM0[3:0] to ICNUM3[3:0]
0100	5	5 times conversion operation from ICNUM0[3:0] to ICNUM4[3:0]
0101	6	6 times conversion operation from ICNUM0[3:0] to ICNUM5[3:0]
0110	7	7 times conversion operation from ICNUM0[3:0] to ICNUM6[3:0]
0111	8	8 times conversion operation from ICNUM0[3:0] to ICNUM7[3:0]
1000	9	9 times conversion operation from ICNUM0[3:0] to ICNUM8[3:0]
1001	10	10 times conversion operation from ICNUM0[3:0] to ICNUM9[3:0]
1010	11	11 times conversion operation from ICNUM0[3:0] to ICNUM10[3:0]

When configured in "one shot" mode, the ADC performs the specified number of conversion after a start request. At the end of each conversion sequence, the data register gets updated with the conversion result.

You can program the conversion of a sequence in ADC_CCSR0 and ADC_CCSR1 (Conversion Channel Sequence Register). The ICNUM0 field defines the first input conversion in sequence. The ICNUM1 field defines the second input conversion and this sequence continues.

Table 4-4 describes the relation between the ICNUMx values and input selected.

Table 4-4 ICNUMx Value and Selected Input

ICNUMx Values	Selected Pin
0000	AIN 0 for op-amp
0001	AIN 1
0010	AIN 2
0011	AIN 3
0100	AIN 4
0101	AIN 5
0110	AIN 6
0111	AIN 7
1000	AIN 8
1001	AIN 9
1010	AIN 10

For example, assume:

- CCNT[3:0] = 0x2
- ICNUM0[3:0] = 0x5 (AIN05)
- ICNUM1[3:0] = 0x2 (AIN02)
- ICNUM2[3:0] = 0x1 (AIN01)

After a start conversion request, ADC converts input 5 (AIN05), then input 2 (AIN02) and it finishes by converting input 1 (AIN01). Each converted data is saved in CRR5, CRR2, and CRR1 register.

You can get information of current conversion channel and conversion count value from ADC Sequence State Register (ADC_SSR).

4.1.6.2 One-Shot Conversion Mode

The ADC programs in two modes. The two modes are:

- One-Shot conversion mode
- Continuous conversion mode

Enable one-shot conversion mode by setting control register bit to 0. In this mode, the ADC performs the complete conversion of sequence at conversion start request and it stops. It waits for another start request. The ADC does not stops until it finishes the conversion sequence

Figure 4-3 illustrates single channel conversion for one-shot mode.

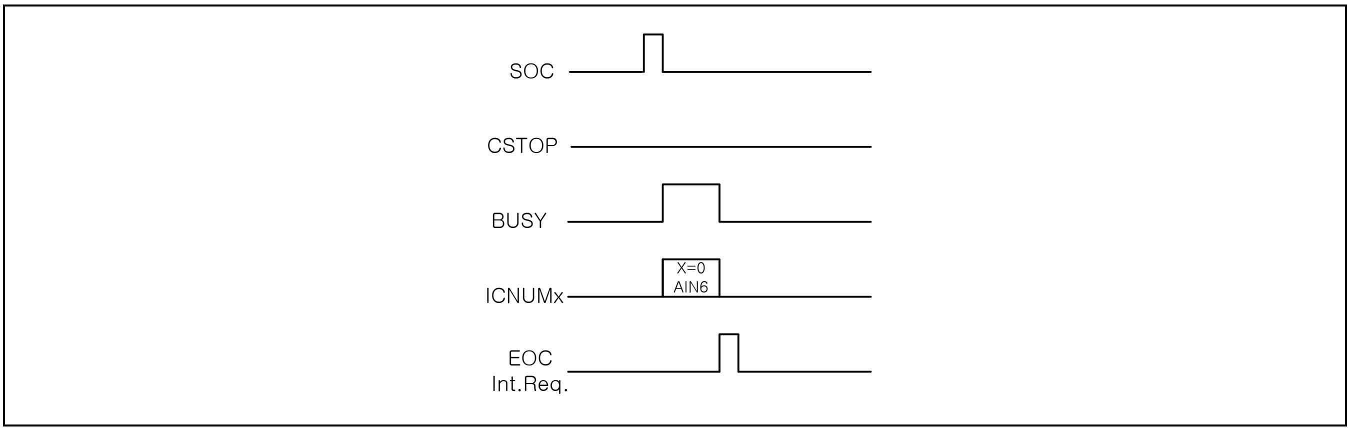


Figure 4-3 One-Shot Mode, Single Channel Conversion

Figure 4-4 illustrates multi-channel conversion for one-shot mode.

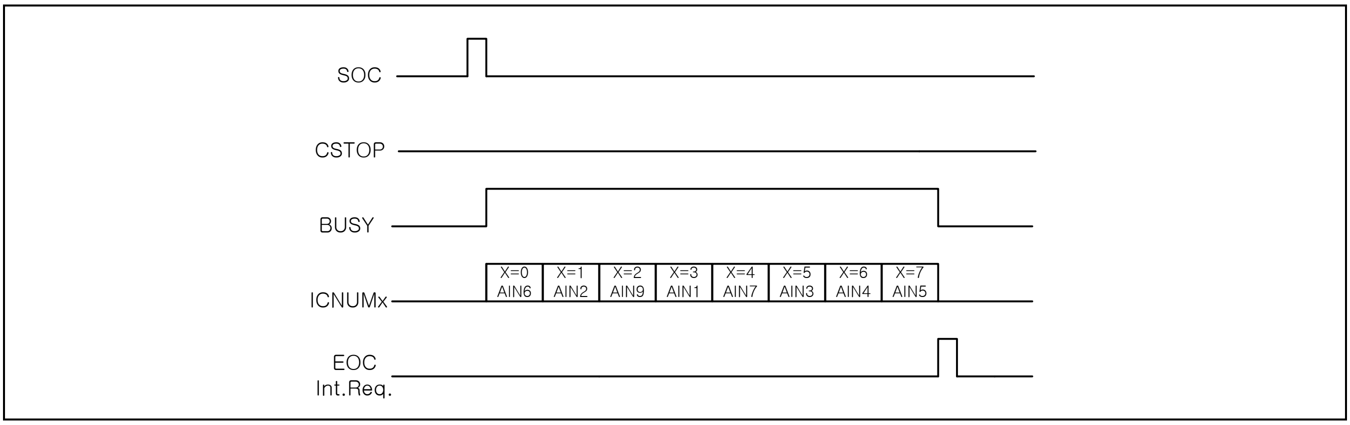


Figure 4-4 One-Shot Mode, Multi-Channel Conversion

4.1.6.3 Continuous Mode

Enable continuous conversion mode by setting control register bit to 1. In this mode, the ADC repetitively performs conversions sequences until it is forced to stop.

To stop continuous conversion, the CPU writes CSTOP bit in control register. When it requests a stop, ADC finishes its current conversion and updates data register with last conversion result. The CPU does not perform any other conversion even if the sequence remains unfinished.

Figure 4-5 illustrates single-channel conversion for continuous mode.

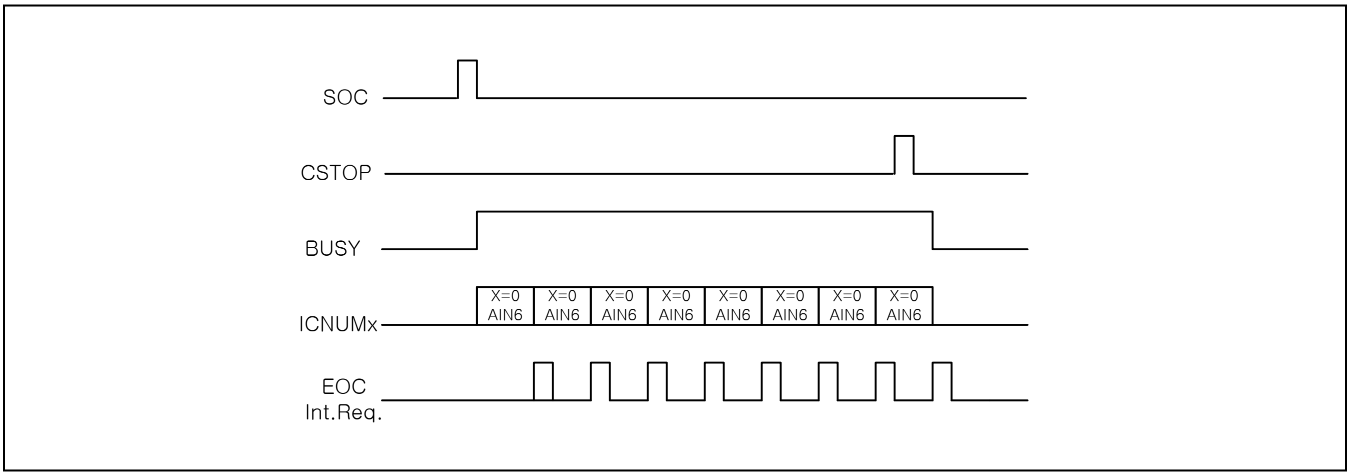


Figure 4-5 Continuous Mode, Single-Channel Conversion

Figure 4-6 illustrates multi-channel conversion for continuous mode.

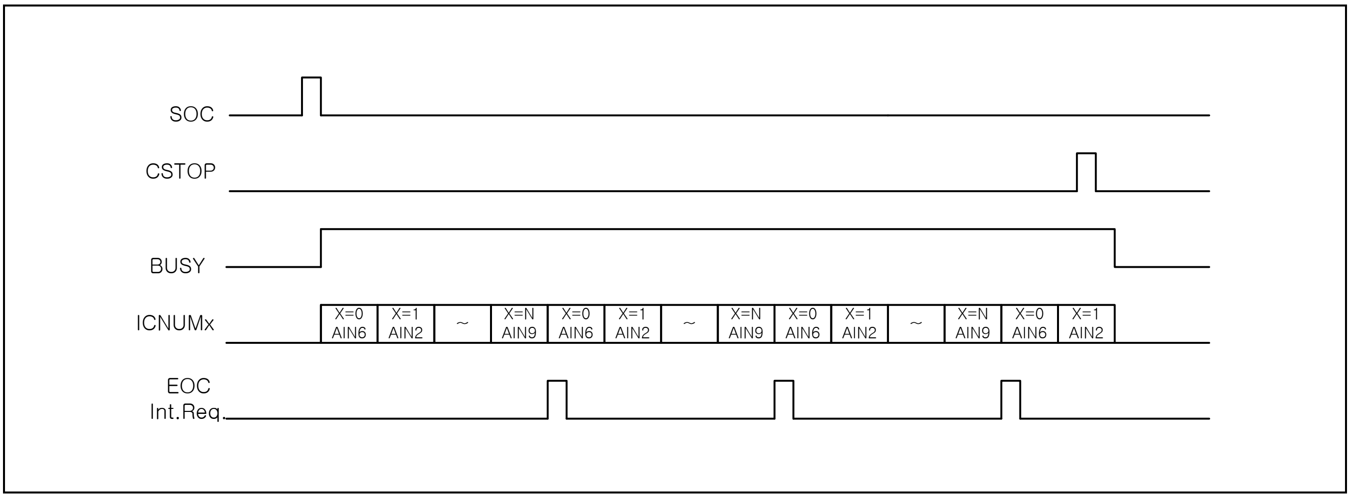


Figure 4-6 Continuous Mode, Multi-Channel Conversion

4.1.7 Conversion Start Trigger

You can select start signal for conversion by selecting TRIG[2:0] field in ADC_MR (Mode Register). There are five types of start triggers:

- Software (START bit in ADC_CR)
- ADTRG – Rising
- ADTRG – Falling
- ADTRG – Rising or Falling (both)
- TCx
- Software trigger by control register bit setting.
- External input trigger. Signal which is asserted from ADTRG pin. You should configure ADTRG pin for use before ADC conversion. When you select ADTRG (TRIG [2:0] = 10'b) define edge type of ADTRG.
- Trigger which is generated by Timer/Counter (TC).
- Trigger which is generated by Inverter Motor Controller (IMC) block

TRIG[2:0]	Start Trigger Source
000'b	Software (START bit in ADC_CR)
001'b	ADTRG – Rising
010'b	ADTRG – Falling
011'b	ADTRG – Rising or Falling (both)
100'b	TCx
101'b	IMC

- START: Conversions are started by the CPU (writing the START bit in the ADC_CR).
- ADTRG: Conversions are started by an external signal using a dedicated input pin (ADTRG).
- Peripheral TC: Conversions are started by the internal hardware signal, TC period match signal.
- Peripheral IMC: Conversions are started by the internal hardware signal, ADC trigger signal in the IMC

4.1.8 Conversion Data

This section describes the Conversion Data process.

4.1.8.1 Conversion Result and Buffer Register

Conversion buffer register is used for saving the converted data temporarily while converting a sequence. After completing the conversion, the ADC generates End of Conversion (EOC) event/interrupt and saves converted data into conversion result registers.

During conversion, use conversion buffer registers to view conversion data. ADC_SSR shows the channel to be converted and the count value. However, after conversion ends, the data in a conversion buffer register is invalid.

4.1.9 Interrupt and Flag

The ADC generates an interrupt if any one of the EOC or OVR is active (set to "1") in the masked interrupt status register and corresponding bit in the ADC_MISR is set to active (set to"1"). Use Interrupt Mask Set/Clear Register (ADC_IMSCR) to enable or disable each interrupt bit.

4.1.9.1 End of Conversion (EOC)

ADC generates an EOC interrupt when conversion sequence is completed while ADC interrupt is enabled. You can check whether interrupt has occurred by reading interrupt status register. Use ADC_IMSCR register to enable or disable interrupt bit respectively.

You can clear EOC interrupt by writing 1 into EOC bit in ADC_ICR register. It is auto-cleared when you read the conversion result register.

4.1.9.2 Over-Run (OVR)

OVR indicates that new data overwrites previously converted data that is not read and is lost. OVR flag is cleared by CPU (writing OVR bit in interrupt clear register). The channel also has an overrun data and you can get this data from status register (ADC_SR).

4.1.9.3 BUSY Flag

The BUSY bit indicates whether ADC is converting analog data or not. The other conversion trigger signal asserted in busy status (BUSY = 1) is ignored.

4.1.10 Calibration

This section explains about Calibration Below registers are related to calibration.

- DAT_NCAL: Converted result data without calibration
- DAT_CAL: Converted result data with calibration
- ADC_GCC: Gain calibration constant
- ADC_OCC: Offset calibration constant

If you want to use calibration function, then you should get constants for a calibration unit. Constants have different value depending on the noise level of real target system.

You can determine ADC Gain Calibration Constant (ADC_GCC) and ADC Offset Calibration Constant (ADC_OCC) by taking two samples of known reference voltages and use these samples to calculate their values. After calculation store ADD_GCC values in ADC_GCCR and ADC_OCC values in ADC_OCCR. A conversion result is calibrated according to the status of CALEN bit. You should set CALEN bit to 1 to get calibrate conversion result of data.

ADC automatically calculates calibrate result before sending result to ADC Convert Data Register (ADC_DR) if CALEN bit is 1. It is directly send to ADC_DR if CALEN bit is 0 and it bypasses calibration unit.

4.1.10.1 Calibration Unit

Figure 4-7 illustrates the ADC calibration scheme.

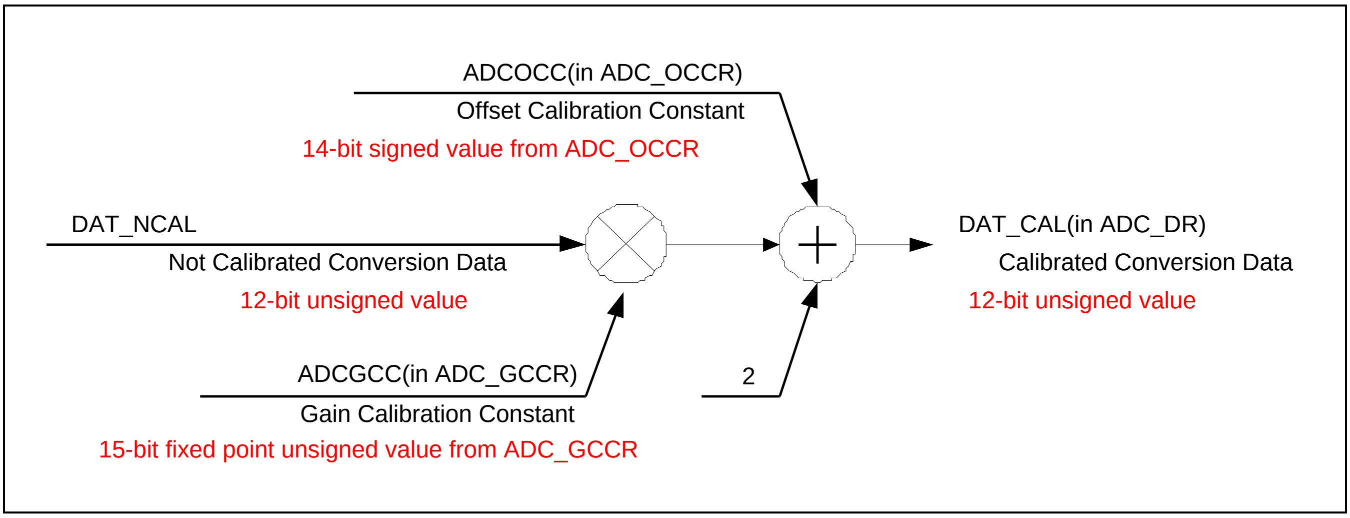

ADCOCC(in ADC_OCCR)
Offset Calibration Constant
14-bit signed value from ADC_OCCR
DAT_NCAL
Not Calibrated Conversion Data
12-bit unsigned value
DAT_CAL(in ADC_DR)
Calibrated Conversion Data
12-bit unsigned value
ADCGCC(in ADC_GCCR)
Gain Calibration Constant
15-bit fixed point unsigned value from ADC_GCCR
2

Figure 4-7 ADC Calibration Scheme

The process to get adjusted ADC conversion data consists of two steps. The two steps are:

- Determine the gain and offset calibration constants.
- Run calibration unit with the non-calibrated data generated by direct conversion on ADC macro-cell.

4.1.10.2 Calibration Unit Operation

DAT_CAL = ADC_GCC × DAT_NCAL + ADC_OCC + 2

You should determine these constants and write corresponding register (ADC_GCCR and ADC_OCCR) values before using calibration function. You should determine two pairs of expected (DAT_CAL) and measured (DAT_NCAL) values that are available for two reference voltages. You can select between internal and external voltage. Reference voltages are 25 percent and 75 percent of VREF.

- VREF: Reference voltage value to convert any ADC input voltage selected by you in voltage range.
- 1/4VREF: 25 percent point voltage of VREF voltage value.
- 3/4VREF: 75 percent point voltage of VREF voltage value.

Figure 4-8 illustrates 2 point calibration.

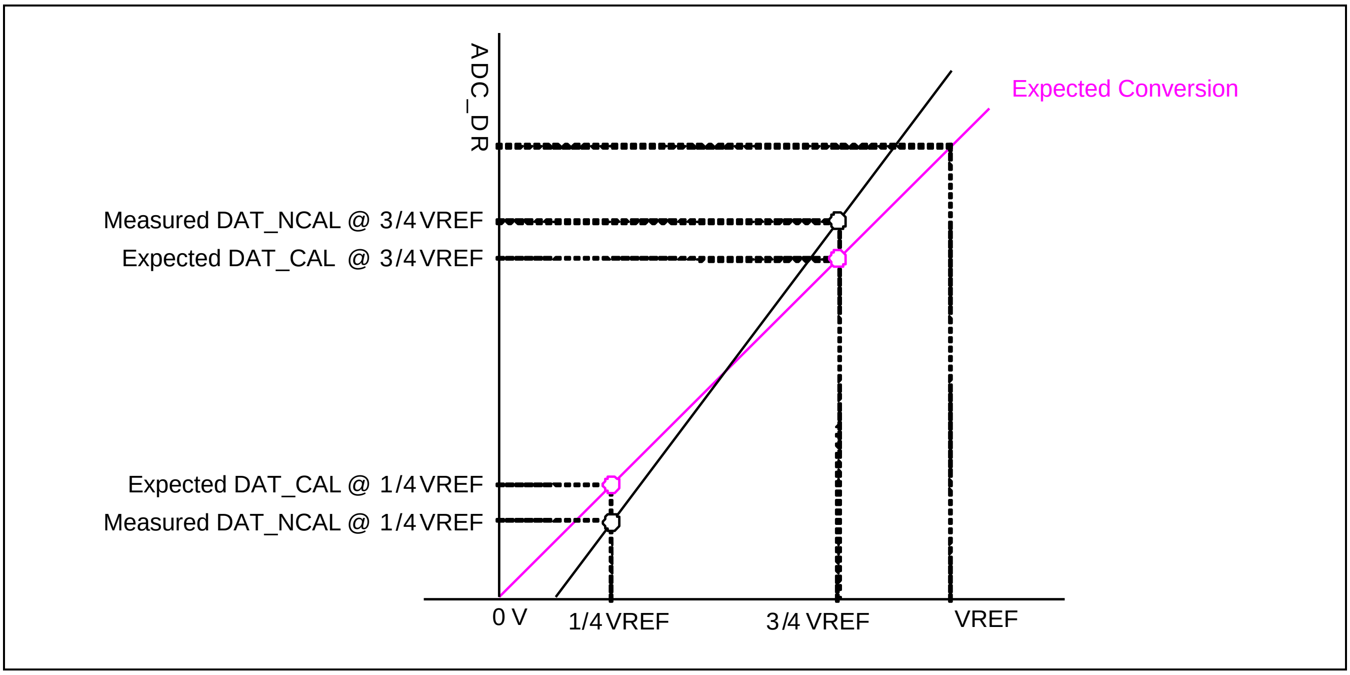


Figure 4-8 2 Point Calibration

The non-calibrated results for these input voltages are obtained by converting these channels with conversion commands when CALEN bit is 0.

The transfer equations when sampling these reference voltages are:

DAT_CAL@3/4VREF = ADC_GCC × DAT_NCAL@3/4VREF + ADC_OCC + 2
DAT_CAL@1/4VREF = ADC_GCC × DAT_NCAL@1/4VREF + ADC_OCC + 2

Thus;

ADC_GCC = (DAT_CAL@3/4VREF – DAT_CAL@1/4VREF)/(DAT_NCAL@3/4VREF – DAT_NCAL@1/4VREF)
ADC_OCC = DAT_CAL@3/4VREF – ADC_GCC × DAT_NCAL@3/4VREF – 2

Or
ADC_OCC = DAT_CAL@1/4VREF – ADC_GCC × DAT_NCAL@1/4VREF – 2

After calculating above equation, the ADC_GCC and ADC_OCC values are written to ADC_GCCR and ADC_OCCR register respectively.

When you set "CALEN" bit, the ADC automatically calibrates the results using the ADCGCC and ADCOCC values stored in ADC calibration registers.

To configure the calibration hardware, the steps are:

- Determine the values of the gain and offset calibration constants.
- Write these constants to the calibration registers

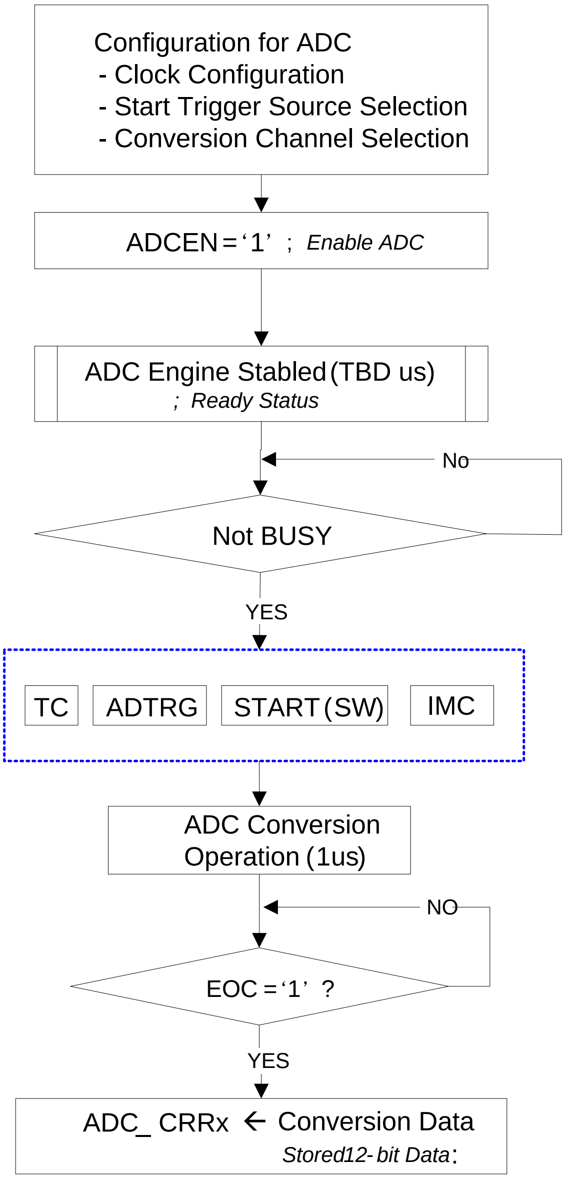
4.1.10.3 Power Management

To minimize power consumption, ADC peripheral contains power management features. Power can be saved on two sides: Analog and Digital.

- Analog Power Saving: To reduce analog power consumption, CPU disables ADC module (ADCDIS bit in SR register should be "1" by writing "1" to ADCDIS bit in control register) that sets analog cell to "standby" mode.
- Digital Power Saving: To reduce digital power consumption, CPU disables ADC clock (write CLKEN bit to "0") that disables all incoming clocks. Then, digital consumption is reduced close to 0.

4.1.11 Operation Sequence

Figure 4-9 illustrates ADC flowchart.


Configuration for ADC
- Clock Configuration
- Start Trigger Source Selection
- Conversion Channel Selection
ADCEN=‘1’ ; Enable ADC
ADC Engine Stabled(TBD us)
; Ready Status
No
Not BUSY
YES
TC
ADTRG
START(SW)
IMC
ADC Conversion Operation (1us)
NO
EOC=‘1’ ?
YES
ADC_ CRRx ← Conversion Data
Stored12-bit Data:

NOTE: User should configure ADC start conversion by only one trigger Signal among ADTRG input signal, START bit, or IMC ADC trigger signal

Figure 4-9 ADC Flowchart

4.1.11.1 Software Sequence for Conversion

The basic sequences of operations after reset for using ADC peripheral are:

- Enable ADC clock, CLKEN in ADC_CEDR.
- Enable ADC engine for ADC operation.
- Configure ADC clock in the ADC_CDR. Program CDIV fields to have analog frequency clock less than 5 MHz.
- Configure conversion channel (ICNUMx in ADC_CCSRx).
- Select the conversion start trigger source (TRIG in ADC_MR).
- Check BUSY bit in ADC_SR. If flag is 0, then ADC is ready to start conversion.
- Start conversion by writing START bit in ADC_CR.
- Sample analog input voltage and every channel completes conversion with 5 ADC clock cycles.
- When the channels defined as a sequence are converted, the EOC bit in ADC_RISR sets to "1".
- CPU then reads the digital value in ADC_CRRx, that automatically clears the EOC. You can clear EOC by writing 1 in ADC_ICR.

4.2 Register Description

4.2.1 Register Map Summary

- Base Address: 0x4004_0000

Register	Offset	Description	Reset Value
ADC_IDR	0x0000	ID register	0x0001_001F
ADC_CEDR	0x0004	Clock enable/disable register	0x0000_0000
ADC_SRR	0x0008	Software reset register	0x0000_0000
ADC_CSR	0x000C	Control set register	0x0000_0000
ADC_CCR	0x0010	Control clear register	0x0000_0000
ADC_CDR	0x0014	Clock divider register	0x0000_0000
ADC_MR	0x0018	Mode register	0x0000_0000
RSVD	–	Reserved	0x0000_0000
ADC_CCSR0	0x0040	Conversion channel sequence register 0	0x0000_0000
ADC_CCSR1	0x0044	Conversion channel sequence register 1	0x0000_0000
ADC_SSR	0x0048	Sequence state register	0x0000_0000
RSVD	–	Reserved	0x0000_0000
ADC_SR	0x0060	Status register	0x0000_0000
ADC_IMSCR	0x0064	Interrupt mask set/clear register	0x0000_0000
ADC_RISR	0x0068	Raw interrupt status register	0x0000_0000
ADC_MISR	0x006C	Masked interrupt status register	0x0000_0000
ADC_ICR	0x0070	Interrupt clear register	0x0000_0000
RSVD	–	Reserved	0x0000_0000
ADC_CRR0	0x0080	Conversion result register 0	0x0000_0000
ADC_CRR1	0x0084	Conversion result register 1	0x0000_0000
ADC_CRR2	0x0088	Conversion result register 2	0x0000_0000
ADC_CRR3	0x008C	Conversion result register 3	0x0000_0000
ADC_CRR4	0x0090	Conversion result register 4	0x0000_0000
ADC_CRR5	0x0094	Conversion result register 5	0x0000_0000
ADC_CRR6	0x0098	Conversion result register 6	0x0000_0000
ADC_CRR7	0x009C	Conversion result register 7	0x0000_0000
ADC_CRR8	0x00A0	Conversion result register 8	0x0000_0000
ADC_CRR9	0x00A4	Conversion result register 9	0x0000_0000
ADC_CRR10	0x00A8	Conversion result register 10	0x0000_0000
ADC_GCR	0x00AC	Gain calibration register	0x0000_0000
ADC_OCR	0x00B0	Offset calibration register	0x0000_0000
RSVD	–	Reserved	0x0000_0000
ADC_CBR0	0x0100	Conversion buffer register 0	0x0000_0000

Register	Offset	Description	Reset Value
ADC_CBR1	0x0104	Conversion buffer register 1	0x0000_0000
ADC_CBR2	0x0108	Conversion buffer register 2	0x0000_0000
ADC_CBR3	0x010C	Conversion buffer register 3	0x0000_0000
ADC_CBR4	0x0110	Conversion buffer register 4	0x0000_0000
ADC_CBR5	0x0114	Conversion buffer register 5	0x0000_0000
ADC_CBR6	0x0118	Conversion buffer register 6	0x0000_0000
ADC_CBR7	0x011C	Conversion buffer register 7	0x0000_0000
ADC_CBR8	0x0120	Conversion buffer register 8	0x0000_0000
ADC_CBR9	0x0124	Conversion buffer register 9	0x0000_0000
ADC_CBR10	0x0128	Conversion buffer register 10	0x0000_0000

4.2.1.1 ADC_IDR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0000, Reset Value = 0x0001_001F

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores the ID code for the corresponding IP.	0x0001_001F

4.2.1.2 ADC_CEDR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBGEN	RSVD																														CLKEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	RW	Debug Mode Enable 0 = Disables debug mode. ADC not halted during processor debug mode. 1 = Enables debug mode. ADC is halted during processor debug mode.	0'b
RSVD	[30:1]	R	Reserved	0
CLKEN	[0]	RW	ADC Controller Clock Enable/Disable Bit 0 = Disables ADC Clock. 1 = Enables ADC Clock. ADC software reset does not affect CLKEN bit status.	0'b

4.2.1.3 ADC_SRR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Performs ADC software reset operation.	0'b

4.2.1.4 ADC_CSR

- Base Address: 0x4004_0000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																													CCSTOP	START	ADCEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	R	Reserved	0
CCSTOP	[2]	W	ADC Continuous Conversion Stop 0 = No effect 1 = Stops the continuous conversion NOTE: Before starting the conversion, you should ensure that ADC is ready for conversion (Ready bit is set to 1 in ADC_SR).	0'b
START	[1]	W	ADC Conversion Start Bit 0 = No effect 1 = Starts ADC conversion This is one of the ADC trigger sources, software trigger.	0'b
ADCEN	[0]	W	ADC Core Enable Bit 0 = No effect 1 = Enables ADC core bit. NOTE: After enabling ADC block (ADCEN == 1), ADC block converts analog value after stabilization time. ADCSTABLE bit in Status Register (SR) is set after stabilization.	0'b

4.2.1.5 ADC_CCR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															ADCEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
ADCEN	[0]	W	ADC Core Enable Clear Bit 0 = No effect 1 = Disables ADC core bit.	0'b

4.2.1.6 ADC_CDR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																										CDIV					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	R	Reserved	0
CDIV	[5:0]	RW	ADC Clock Divider Selection Field This field determines the ADC clock frequency. The frequency is from 1 to 5 MHz. For fast conversion, you generate maximum frequency with clock divider. Clock source of ADC is PCLK. FADC = PCLK/(2 × (CDIV + 1)) Ex) PCLK = 20 MHz, CDIV = 1 → FADC (ADC clock) = 5 MHz PCLK = 20 MHz, CDIV = 9 → FADC (ADC clock) = 1 MHz	00000'b

NOTE: The clock for ADC should not exceed 5 MHz.

4.2.1.7 ADC_MR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD					EICR	ICRV	CALEN	RSVD												CCNT				CMODE	RSVD				TRIG		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R W	R W	R W	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R	R	R	R	R W	R W	R W

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="4">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:27]</td><td>R</td><td colspan="4">Reserved</td><td>0</td></tr>
<tr><td rowspan="6">EICR</td><td rowspan="6">[26]</td><td rowspan="6">RW</td><td colspan="4">Calibration Reference Voltage Source
0 = Selects external input voltage sources (1/4 AV_{DD} and 3/4 AV_{DD}) to determine calibration constants through ADC input channel. In this case, it is same as normal operation mode. The difference is that the input voltage for conversion is 1/4 AV_{DD} and 3/4 AV_{DD}.
1 = Selects internal reference voltage sources (1/4 AV_{DD} and 3/4 AV_{DD}) to determine the calibration constants.
NOTE: If you cannot use ADC without calibration, then the CALEN bit is 0. In normal operation, EICR is 0.</td><td rowspan="6">0'b</td></tr>
<tr><td><b>–</b></td><td><b>EICR</b></td><td><b>ICRV</b></td><td><b>Remark</b></td></tr>
<tr><td>Use 1/4, 3/4, or both reference voltage supplied externally</td><td>0</td><td>X</td><td>You can use any ADC channel (AINx) to obtain external reference voltage. You can get the conversion data by 1/4 reference voltage with AINx. Use the same channel or other channels for conversion data of 3/4 reference voltage.</td></tr>
<tr><td>Use 1/4 reference voltage supplied internally</td><td>1</td><td>0</td><td>Do not use ADC channel (AINx)</td></tr>
<tr><td>Use 3/4 reference voltage supplied internally</td><td>1</td><td>1</td><td>Do not use ADC channel (AINx)</td></tr>
<tr><td colspan="4">X means "don't care".</td></tr>
<tr><td>ICRV</td><td>[25]</td><td>RW</td><td colspan="4">Internal Calibration Reference (Voltage) Value</td><td>0'b</td></tr>
</table>

Name	Bit	Type	Description	Reset Value
			0 = Selects internal voltage source 1/4 AV_{DD} to determine calibration constants. 1 = Selects internal voltage source 3/4 AV_{DD} to determine calibration constants.	
CALEN	[24]	RW	Calibration Enable/Disable Control Bit 0 = Disables calibration bit. The conversion data of ADC is generated without calibration. 1 = Enables calibration bit. The conversion data of ADC is adjusted. To use calibration, you should initialize before enabling calibration. For initialization, you should obtain conversion data of 1/4 and 3/4 reference voltage. That is used to predefine configuration "ICRV" or "CRVS" control bits.	0'b
RSVD	[23:12]	R	Reserved	0
CCNT	[11:8]	RW	Conversion Count Field NOTE: Even in One-Shot mode, ADC will run multiple conversions if the CCNT is greater than 0000'b. 0000 = 1 time conversion operation 0001 = 2 times conversion operation 0010 = 3 times conversion operation 0011 = 4 times conversion operation 0100 = 5 times conversion operation 0101 = 6 times conversion operation 0110 = 7 times conversion operation 0111 = 8 times conversion operation 1000 = 9 times conversion operation 1001 = 10 times conversion operation 1010 = 11 times conversion operation Others = Not used	0000'b
CMODE	[7]	RW	Conversion Mode Bit 0 = Selects One-Shot mode. ADC converts as much inputs as specified by the ICNUMx [3:0] in the order specified in the ADC_CCSR, and stops. 1 = Selects Continuous mode. ADC converts as much inputs as specified by the ICNUMx [3:0] in the order specified in the ADC_CCSR, and repeats. This bit is initialized to 0. In Continuous mode, after a Stop command, ADC finishes the on-going conversion and this looks like an extra conversion.	0'b
RSVD	[6:3]	R	Reserved	0
TRIG	[2:0]	RW	ADC Start Trigger Signal Selection Bits This field determines the trigger signal for ADC. 000 = Selects Software (START bit in ADC_CR). 001 = Selects ADTRG – Rising. 010 = Selects ADTRG – Falling. 011 = Selects ADTRG – Rising or Falling (Both). 100 = Selects TCx – Timer/Counter Match.	000'b

Name	Bit	Type	Description	Reset Value
			101 = Selects IMC0 – ADC Trigger Value. Other = Reserved ADC conversion by one or several timer period match can be started.	

4.2.1.8 ADC_CCSR0

- Base Address: 0x4004_0000
- Address = Base Address + 0x0040, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ICNUM7				ICNUM6				ICNUM5				ICNUM4				ICNUM3				ICNUM2				ICNUM1				ICNUM0			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
ICNUM7 ICNUM6 ICNUM5 ICNUM4 ICNUM3 ICNUM2 ICNUM1 ICNUM0	[31:28] [27:24] [23:20] [19:16] [15:12] [11:8] [7:4] [3:0]	RW	Analog Input Channel Number Selection Field 0000 = AIN0,an input channel for OP-AMP 0001 = AIN1 0010 = AIN2 0011 = AIN3 0100 = AIN4 0101 = AIN5 0110 = AIN6 0111 = AIN7 1000 = AIN8 1001 = AIN9 1010 = AIN10 Others = Invalid	0

4.2.1.9 ADC_CCSR1

- Base Address: 0x4004_0000
- Address = Base Address + 0x0044, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RVDS																				ICNUM10				ICNUM9				ICNUM8			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:12]	R	Reserved	0
ICNUM10 ICNUM9 ICNUM8	[11:8] [7:4] [3:0]	RW	Analog Input Channel Number Selection Field 0000 = AIN0, an input for OP-AMP 0001 = AIN1 0010 = AIN2 0011 = AIN3 0100 = AIN4 0101 = AIN5 0110 = AIN6 0111 = AIN7 1000 = AIN8 1001 = AIN9 1010 = AIN10 Others = Invalid	0

4.2.1.10 ADC_SSR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0048, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	R	Reserved	0
CCCV	[27:24]	R	Conversion Count Value 0000 = No 0001 = 1st conversion is ended. 0010 = 2nd conversion is ended. 0011 = 3rd conversion is ended. 0100 = 4th conversion is ended. 0101 = 5th conversion is ended. 0110 = 6th conversion is ended. 0111 = 7th conversion is ended. 1000 = 8th conversion is ended. 1001 = 9th conversion is ended. 1010 = 10th conversion is ended. 1011 = 11th conversion is ended. When ADC is not in "busy" status, the value in this register is invalid.	0
RSVD	[23:0]	R	Reserved	0

NOTE: The CCCV is maintained the last value in the continuous mode and is set to 0 in the multiple one shot mode at the end of last conversion.

4.2.1.11 ADC_SR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0060, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD					OVR10	OVR9	OVR8	OVF7	OVR6	OVR5	OVR4	OVF3	OVR2	OVR1	OVR0	RSVD								CMODE	RSVD					BUSY	ADCEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:27]	R	Reserved	0
OVR10 OVR9 OVR8 OVR7 OVR6 OVR5 OVR4 OVR3 OVR2 OVR1 OVR0	[26] [25] [24] [23] [22] [21] [20] [19] [18] [17] [16]	R	Overrun Status 0 = No overrun 1 = Overrun occurs	0'b
RSVD	[15:8]	R	Reserved	0
CMODE	[7]	R	Conversion Mode Status 0 = Selects One-Shot mode with the help of microprocessor. 1 = Selects Continuous mode, the peripheral is stand-alone. Initialize this bit to 0 and the bit changes when there is change of mode. This bit never generates any interrupts.	0'b
RSVD	[6:2]	R	Reserved	0
BUSY	[1]	R	ADC Status Monitoring Bit This bit notifies the status of ADC. 0 = ADC is not on a conversion operation. 1 = ADC is on a conversion operation. NOTE: To change the configuration of ADC, you should check ADC Status Register (ADC_SR).	0'b
ADCEN	[0]	R	ADC Stabilization Status	0'b

Name	Bit	Type	Description	Reset Value
			0 = Disables ADC Core or ADC is not stabilized even if ADC Core is enabled. 1 = ADC is stabilized. This bit is set after initialization time. When this bit is 1, ADC can convert data.	

4.2.1.12 ADC_IMSCR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0064, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																														OVR	EOC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	R	Reserved	0
OVR	[1]	RW	Overrun Interrupt Mask 0 = OVR interrupt is masked. (Disables the interrupt) 1 = OVR interrupt is not masked. (Enables the interrupt) OVR includes from OVR0 to OVR10.	0'b
EOC	[0]	RW	End of Conversion Interrupt Mask 0 = EOC interrupt is masked. (Disables the interrupt) 1 = EOC interrupt is not masked. (Enables the interrupt) Hardware sets this bit and software clears it.	0'b

NOTE: On a Read, the ADC_IMSCR register gives current value of mask on the relevant interrupt.
A Write of 1 to a particular bit, sets mask and enables the interrupt to be read.
A Write of 0 to a particular bit clears the corresponding mask.

4.2.1.13 ADC_RISR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0068, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																														OVR	EOC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
OVR	[1]	R	Overrun Raw Interrupt Status Provides raw interrupt state (prior to masking) of OVR interrupt.	0
EOC	[0]	R	End of Conversion Raw Interrupt Status Provides raw interrupt state (prior to masking) of EOC interrupt.	0

NOTE: On a Read, the ADC_RISR register gives the current raw status value of the corresponding interrupt prior to masking. A Write makes no effect.

4.2.1.14 ADC_MISR

- Base Address: 0x4004_0000
- Address = Base Address + 0x006C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																														OVR	EOC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
OVR	[1]	R	Overrun Masked Interrupt State Provides masked interrupt status (after masking) of OVR interrupt.	0
EOC	[0]	R	End of Conversion Masked Interrupt State Provides masked interrupt status (after masking) of EOC interrupt.	0

NOTE: On a Read, the ADC_MISR register gives the current masked status value of the corresponding interrupt. A Write has no effect.

4.2.1.15 ADC_ICR

- Base Address: 0x4004_0000
- Address = Base Address + 0x0070, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																														OVR	EOC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
OVR	[1]	W	Overrun Interrupt Clear 0 = No effect 1 = Clears the OVR interrupt.	0
EOC	[0]	W	End of Conversion Interrupt Clear 0 = No effect 1 = Clears the EOC interrupt.	0

NOTE: On a Write of 1, the corresponding interrupt is cleared. A Write of 0 makes no effect.

4.2.1.16 ADC_CRRn

- Base Address: 0x4004_0000
- Address = Base Address + 0x0080, Reset Value = 0x0000_0000
- Address = Base Address + 0x0084, Reset Value = 0x0000_0000
- Address = Base Address + 0x0088, Reset Value = 0x0000_0000
- Address = Base Address + 0x008C, Reset Value = 0x0000_0000
- Address = Base Address + 0x0090, Reset Value = 0x0000_0000
- Address = Base Address + 0x0094, Reset Value = 0x0000_0000
- Address = Base Address + 0x0098, Reset Value = 0x0000_0000
- Address = Base Address + 0x009C, Reset Value = 0x0000_0000
- Address = Base Address + 0x00A0, Reset Value = 0x0000_0000
- Address = Base Address + 0x00A4, Reset Value = 0x0000_0000
- Address = Base Address + 0x00A8, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:12]	R	Reserved	0
DATA	[11:0]	R	Conversion Result Data of ADC A/D converter Output Data Result: 0x000 – 0xFFF When A/D sequence conversion is finished, the conversion result can be read from the ADC_CRR register.	0x000

4.2.1.17 ADC_GCR

- Base Address: 0x4004_0000
- Address = Base Address + 0x00AC, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:15]	R	Reserved	0
GCC_INT	[14]	RW	–	0
GCC_FRAC	[13:0]	RW	ADC Gain Calibration Constant Value ADCGCC field is fixed point unsigned value. ADCGCC[14:0] = Fixed point unsigned value GCC × 2^{14} The fixed point unsigned value GCC gets from the following (For more detail, Refer to Section *4.1.10.1* *Calibration Unit*) GCC = (IDLE75 % – IDLE25 %)/(RAW75 % – RAW25 %) The calculated result (GCC) should be limited from 0.5 to 1.99999	0x0000

4.2.1.18 ADC_OCR

- Base Address: 0x4004_0000
- Address = Base Address + 0x00B0, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	R	Reserved	0
ADCOCC	[13:0]	RW	ADC Offset Calibration Constant Value ADCOCC field is signed value. Negative values should be expressed using the 2's complement.	0x0000

4.2.1.19 ADC_CBRn

- Base Address: 0x4004_0000
- Address = Base Address + 0x0100, Reset Value = 0x0000_0000
- Address = Base Address + 0x0104, Reset Value = 0x0000_0000
- Address = Base Address + 0x0108, Reset Value = 0x0000_0000
- Address = Base Address + 0x010C, Reset Value = 0x0000_0000
- Address = Base Address + 0x0110, Reset Value = 0x0000_0000
- Address = Base Address + 0x0114, Reset Value = 0x0000_0000
- Address = Base Address + 0x0118, Reset Value = 0x0000_0000
- Address = Base Address + 0x011C, Reset Value = 0x0000_0000
- Address = Base Address + 0x0120, Reset Value = 0x0000_0000
- Address = Base Address + 0x0124, Reset Value = 0x0000_0000
- Address = Base Address + 0x0128, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:12]	R	Reserved	0
DATA	[11:0]	R	Conversion Buffer Data of ADC Data will be from 0x000 to 0xFFF. After conversion starts you can watch converting data before the end of conversion.	0x000

5 Clock and Power Manager

5.1 Overview

The Clock and Power Manager (CM) chapter describes the management for system clock and power according to the operation mode.

The System Clock tree consists of two clock sources, they are:

- EMCLK: External Main Clock, you can select the frequency range from 1 to 12 MHz
- IMCLK: Internal Main Clock, 40 MHz

The Clock Control Logic generates the required clock signals including the HCLK/FCLK for CPU. The HCLK generates the required clock signals for the AHB bus peripherals and the PCLK for the APB bus peripherals and so on. The Clock Manager has a Phase Locked Loop (PLL) for a System Clock.

By using the software, the Clock Control Logic connects or disconnects the clock to each peripheral block, which reduces the power consumption. The Power Control Logic has several power management schemes to maintain an optimal power consumption for a given task. The Power Management Block activates several modes.

5.1.1 Features

The features section includes:

- Clock Management
- Core Clock Sources
- PLL
- Reset Sources
- Power Management

5.1.1.1 Clock Management

The features of Clock Management are:

- External Oscillator 1 to 12 MHz (EMCLK: External Main Clock)
- Programmable PLL with operational frequency from 12 to 40 MHz
- Programmable Clock Divider (SDIV and PDIV) for SYSCLK and PCLK
- EMCLK failure detection with an Internal Main Clock (Clock Monitor function)
- Clock Out Port (COP[4:0])

5.1.1.2 Core Clock Sources

The features of Core Clock sources are:

- EMCLK: External Main Clock, External 1 to 12 MHz Oscillator
- IMCLK: Internal Main Clock, Internal 40 MHz RC Oscillator
- PLLCLK: From 12 to 40 MHz

5.1.1.3 PLL

The features of PLL are:

- Obtain an Input Clock from the EMCLK (1 to 12 MHz)
- Configurable output frequency from 12 to 40 MHz
- Configurable counter for the PLL stabilization time

5.1.1.4 Reset Sources

The features of Reset Sources are:

- NRST: External Input Pin Reset, nRESET
- EMCMRST: External Main Clock Monitor Fail Reset
- LVDRST: LVD Reset (also called as an LVR)
- WDTRST: Watchdog Timer Reset
- SWRST: Software Reset
- PORST: Power-On Reset
- SYSRST: Reset by CPU Request

5.1.1.5 Power Management

The features of Power Management are:

- NORMAL MODE: CPU runs by one of the clock sources (EMCLK or IMCLK), except the PLLCLK.
- PLL MODE: The CPU runs by the PLLCLK
- IDLE MODE: The CPU halts the operations of:
  - HCLK Stop
  - Configure STCLK or PCLK by enabling or disabling the appropriate bits in the registers under software control.
  - Enters by sleep command that belongs to the Cortex^{TM}-M0
- STOP MODE: This mode stops all clocks and operations.

5.1.2 Block Diagram

Figure 5-1 illustrates the system clock tree block diagram.

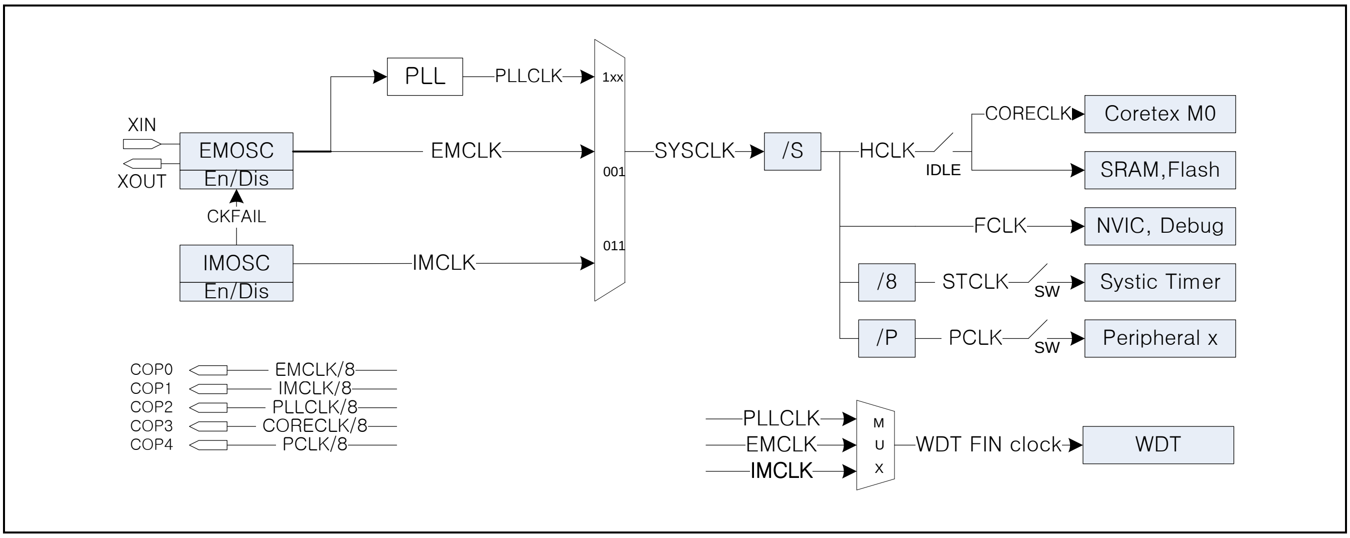


Figure 5-1 System Clock Tree Block Diagram

Table 5-1 describes the clock definition.

Table 5-1 Clock Definition

Name	Definition	Description
EMCLK	External main clock or external main oscillator clock	From 1 to 12 MHz
IMCLK	Internal main clock or internal main oscillator clock	40 MHz
PLLCLK	PLL output clock	From 12 to 40 MHz
SYSCLK	System clock	From 1 to 40 MHz
FCLK	Free running clock for Cortex-M0	SYSCLK/SDIV
CORECLK HCLK	Cortex-M0 clock and AHB bus peripherals clock	SYSCLK/SDIV
STCLK	Sys-tick timer clock in cortex-M0	SYSCLK/SDIV/8
PCLK	Peripherals clock	SYSCLK/SIDV/PDIV

NOTE:

1. Configurable SDIV (1, 2, 3, 4, 5, 6, 7, or 8)
2. Configurable PDIV (1, 2, 4, 8, or 16)

5.2 Clocks

5.2.1 SYSCLK

The System Clock (SYSCLK) is the symbol name of central bridge in clock tree. The manufacture initial clock source is the IMCLK. But the SYSCLK can be selected among three clock sources (IMCLK, EMCLK, or PLLCLK), and it can be explained with two different views.

- Initial default SYSCLK at reset
- In this case, the available clock sources are IMCLK and EMCLK. The SYSCLK is decided between IMCLK and EMCLK by configuration smart option. It is the same as like hardware configuration. If you want to change the SYSCLK's reset value, you need to change the smart option. It can be done by smart option erase and program. Refer to the flash memory controller related to the smart option erase and program.
- If you select EMCLK as the SYSCLK, the pin (XIN or XIN/XOUT) to obtain external clock source should be configured into configuration smart option.
- SYSCLK when the CPU executes code
- This case is to change the SYSCLK from initial clock defined at reset to other different clock source. The available clock sources are IMCLK, EMCLK, and PLLCLK. To change you should use the control bit in Mode Register (CM_MR). It is the control by user and software. When you switch the SYSCLK, the precise sequence is required because of critical event to microcontroller. Refer to 1.3.2 SYSCLK Change

Figure 5-2 illustrates the system clock selection diagram.

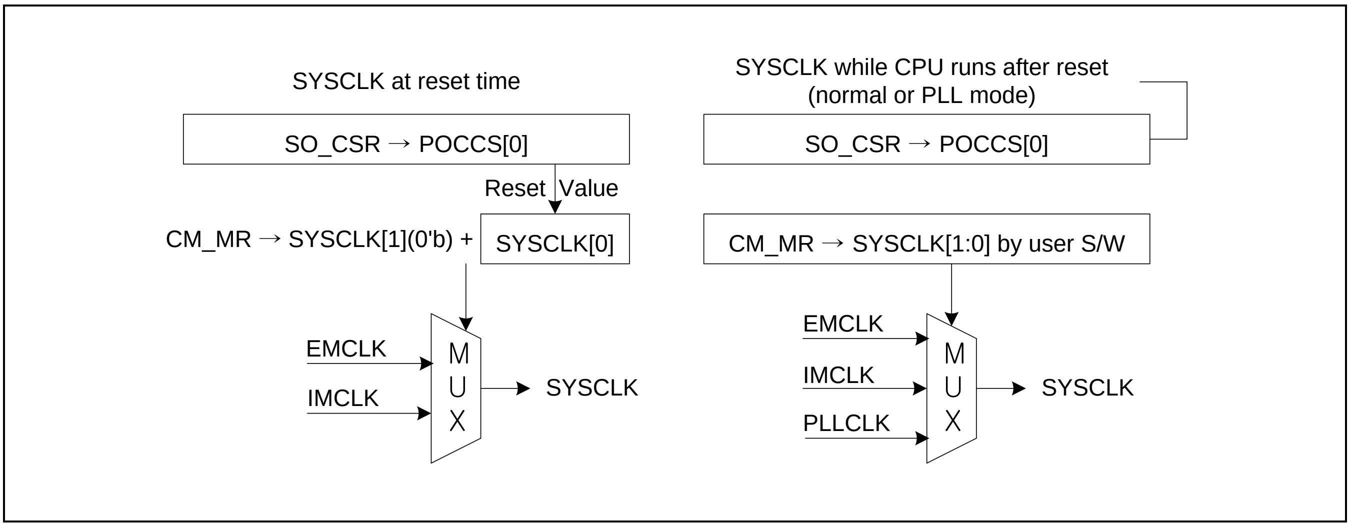


Figure 5-2 System Clock Selection

Table 5-2 describes the summary of smart option for clock manager. To have the configuration for your system, you should use this configuration smart option. They decide the reset value of SYSCLK, Pin8 (XIN/P0.31), Pin7 (XOUT/P0.30), and BTDIV.

Table 5-2 Summary of Smart Option for Clock Manager

SO_CSR	Bit name and value	The result after reset

[1:0]	POCCS[0] = 0
	POCCS[0] = 1
		SYSCLK = EMCLK
		SYSCLK = IMCLK

[2]	XIN	0 = GPIO (P0.31) pin 1 = XIN pin
[3]	XOUT	0 = GPIO (P0.30) pin 1 = XOUT pin
[15:12]	BTDIV[3:0] = 3 BTDIV[3:0] = 4 BTDIV[3:0] = 5 … BTDIV[3:0] = 15	BT DIVIDER = 1 BT DIVIDER = 2 BT DIVIDER = 4 … BT DIVIDER = 4096

NOTE:

1. SO_CSR: Smart Option Configuration Status Register
2. When you program on flash operation, other bits except the upper control bits (field) should be "1".
(For more information, refer to the IFC (Internal Flash Controller) chapters)
3. Smart option should be programmed before the operation.

Table 5-3 describes the clock status at reset and Wake-Up.

Table 5-3 Clock Status at Reset and Wake-Up

Clock	Reset	Wake-Up from STOP
EMCLK	It depends on smart option	Status before STOP
IMCLK	RUN	If FWAKE is 0, status before STOP If FWAKE is 1, RUN
PLLCLK	STOP (Disable)	Status before STOP
SYSCLK	RUN	RUN
FCLK	RUN	RUN
HCLK	RUN	RUN
STCLK	STOP (Disable)	Status before STOP
PCLK	RUN	Status before STOP

NOTE: The "RUN" is explained in the Clock Manager view.

5.2.2 IMCLK

The IMCLK means an Internal Main Clock 40 MHz. After any reset, this clock is always enabled by default. While chip operates, you can enable or disable the IMCLK by controlling the IMCLK bit in the CM_CCR/CM_CSR register.

The IMCLK is the SYSCLK with reset value by manufacture. IMCLK can become SYSCLK by the POCCS bit of the SO_CSR register at reset time (Refer to IFC chapter for more details). After reset, the IMCLK supplies clock to the SYSCLK when the SYSCLK field in the CM_MR register is "01'b".

The IMCLK is used as the reference clock for the EMCLK monitor function. If you want to check the functionality (clocking or not) of the EMCLK, then you should enable the IMCLK and the associated Clock Monitor function. If an enabled clock monitor function detects EMCLK clock fail and EMCMRST (External Main Clock Monitor Reset) occurs, the IMCLK runs as the SYSCLK after reset. When the clock monitor function is enabled, enabled IMCLK can't be disabled although you write "'1'b" to IMCLK bit in CM_CCR register.

5.2.3 EMCLK

EMCLK means an External Main Clock. The acceptable frequency range of External Clock Oscillator is from 1 to 12 MHz. EMCLK can be generated the crystal/ceramic resonator or the external clock. When you use the resonators and load capacitors as components of an External Oscillator Circuit, you should place them close to the chip. This proximity to the chip provides a stable clock and minimizes the stabilization time. You should choose the capacitance value of the load capacitor according to the external oscillator frequency.

If XIN/XOUT or XIN pin is defined after any reset, it has the enabled EMCLK by default. That is decided by the configuration smart option. While microcontroller is operating, you can enable or disable the EMCLK by controlling the EMCLK bit in the CM_CCR/CM_CSR register.

You can use EMCLK for the SYSCLK and several blocks. Especially, the PLL has only the EMCLK as an Input Clock source. EMCLK can become SYSCLK by the POCCS bit of the SO_CSR register at reset time (Refer to IFC chapter for more details). After enabling and stabilizing the External Oscillator, the EMCLK supplies clock to the SYSCLK when the SYSCLK field in the CM_MR register is "00'b".

Figure 5-3 illustrates the crystal/ceramic resonator or the external clock circuit diagram.

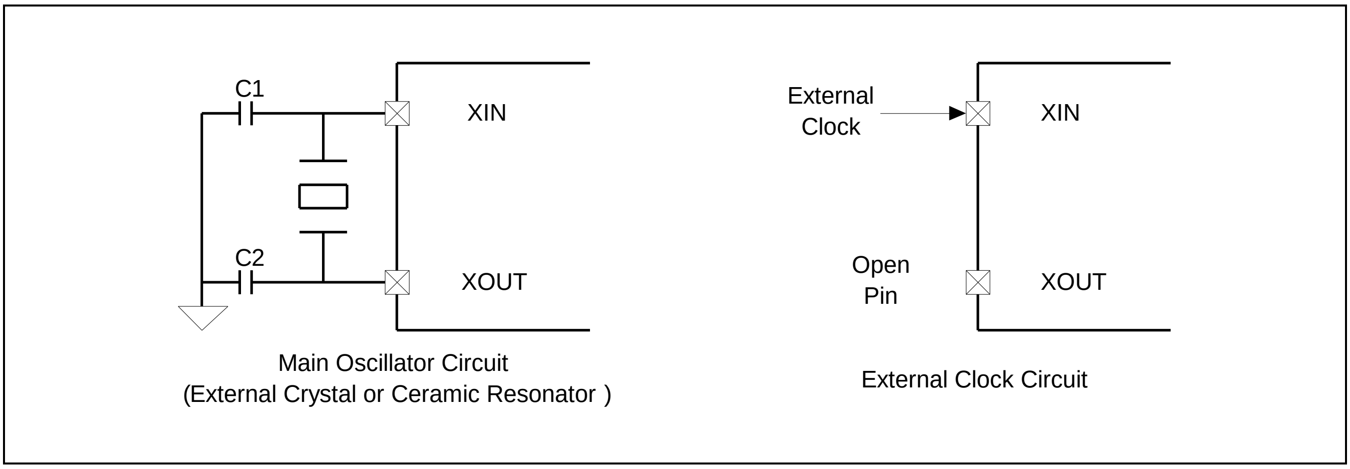


Figure 5-3 Crystal/Ceramic Resonator or External Clock Circuit

Caution: If you don't the external main-oscillator when pin7 and pin8 are defined XOUT and XIN function, you should tie XIN to ground. XOUT should be opened.

5.2.4 PLL

The PLL (Phase Locked Loop) is a frequency synthesizer and provides frequency multiplication capabilities. The PLLs Output Clock Frequency (FOUT) is related to the Input Clock Frequency (FIN) .The FOUT equation is:

```
FOUT = ((m + 8) × FIN)/((p+2) × 2^s)        (when LFPASS=0)
FOUT = ((m + 8) × FIN)/2^s           (when LFPASS=1)
@ m = M[7:0], p = P[5:0], s = S[1:0]
```

Where FOUT is the Output Clock Frequency and FIN is the Input Clock Frequency "m", "p", and "s" are the decimal values for programmable dividers. The PLL contains a Phase Frequency Detector (PFD), a Charge Pump, a Voltage Controlled Oscillator (VCO), a 6-bit Pre-Divider, an 8-bit Main-Divider, and a 2-bit Post-Scaler. *Figure 5-4* illustrates this.

The PLL multiplies the EMCLK and provides the clock source for SYSCLK. The input range of source is from 1 to 12 MHz and the PLL amplifies the source clock to the PLLCLK.

Finish the PLL configuration before enabling the PLL by writing "1" to the PLL bit in the CM_CSR register. Additionally, check the stabilization of the Input Clock (EMCLK) before the PLL is on. After enabling and stabilizing the PLL, it allows you to configure the CM_MR register to supply the PLL clock to the SYSCLK. When the clock transition completes, the STABLE bit in the CM_SR register is set to "1". After the reset, it has the disable PLL by default.

F_{IN} = Input frequency of PLL = EMCLK	PLL multipliers = M[7:0] (16 to 255)	Do not set the value P[5:0] or M[7:0] to all zeros.
	PLL pre-divider = P[5:0] (1 to 63)	
	PLL post-scaler = S[1:0] (0 to 3)	
F_{OUT} = Output frequency of PLL = PLLCLK	((M[7:0]+8) × FIN)/((P[5:0]+2) × 2^{S}[1:0])	when LFPASS = 0
	((M[7:0]+8) × FIN)/2^{S}[1:0]	when LFPASS = 1

Figure 5-4 illustrates the PLL block diagram.

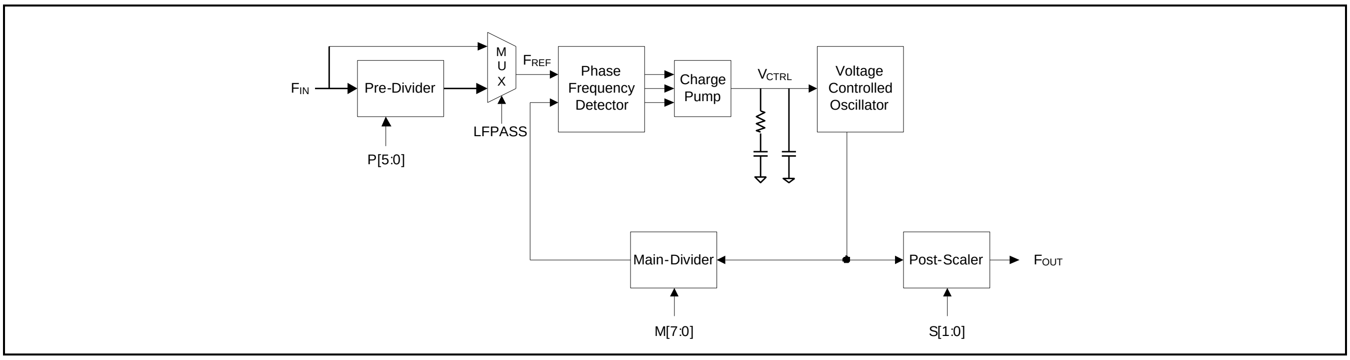


Figure 5-4 PLL (Phase-Locked Loop) Block Diagram

The PLL within the clock generator is the circuit that synchronizes the Output Signal with a reference or Input Signal in the frequencies and phases. It includes the VCO to generate the Output Frequency, the divider "P" to divide the reference frequency by "p", the divider "M" to divide the VCO Output Frequency by "m", the divider "S" to divide the VCO Output Frequency, the phase detector, charge pump, and loop filter.

5.2.4.1 Phase Frequency Detector

The Phase Frequency Detector monitors the phase difference between the F_{REF} (the reference frequency) and the F_{VCO} (the feedback frequency). The Phase Frequency Detector generates a control signal when it detects a difference between the two frequencies.

5.2.4.2 Charge Pump

The Charge Pump converts the Phase Frequency Detector control signal to a charge in voltage across the internal filter. This controls the VCO.

5.2.4.3 Voltage Controlled Oscillator (VCO)

The VCO controls the output voltage from the loop filter. This results into its oscillation frequency to increase or decrease as a function of variations in voltage.

When the VCO output matches the System Clock in frequency and phase, the Phase Frequency-Detector stops sending a control signal to the charge pump. Then the VCO frequency remains constant and the PLL remains locked onto the System Clock.

5.2.4.4 PLL Value Change Steps

If the PLL setting requires change while using F_{OUT} (PLLCLK) as a SYSCLK, then the PLL transition noise may be asserted to system. Therefore, you should change PLL configuration in the NORMAL mode. Steps to change the PLL configuration are:

1. After Reset

   - Enable an External Main Oscillator by controlling the EMCLK bit in the CM_CSR register if EMCLK is disabled.
   - Check an External Main Stable by monitoring the EMCLK bit in the CM_SR register.
   - Set the PLL stabilization time to prevent an abnormal operation.
   - Change the PMS value in the CM_PDPR register.
   - Enable the PLL by controlling the PLL bit in the CM_CSR register.
   - Check the PLL stable by monitoring the PLL bit in the CM_SR or CM_RISR [(1)] register.
   - If a PLL is stable, then set the SYSCLK fields in the CM_MR register. Later SYSCLK is fed from the PLLCLK.
   - Check the switching status by monitoring the STABLE bit in the CM_SR or CM_RISR [(2)] register.

2. PLL Configuration Change in the PLL Mode

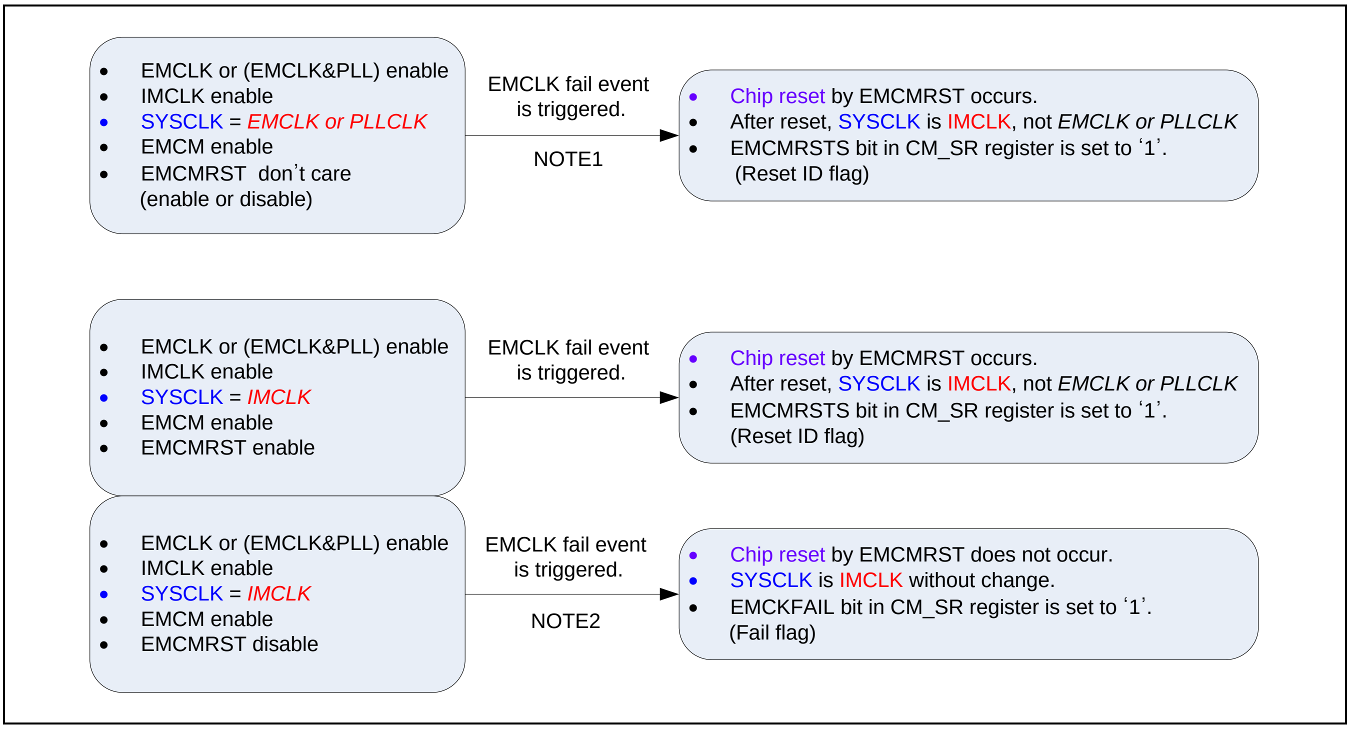
   - Change the System Clock to the External Main Clock Oscillator (EMCLK) or Internal Main Clock (IMCLK)
   - Disable the PLL by setting the PLL bit in the CM_CCR register
   - Change the PMS value in the CM_PDPR register
   - Enable the PLL by controlling the PLL bit in the CM_CSR register
   - Check the PLL stable by monitoring the PLL bit in the CM_SR or CM_RISR [(1)] register.
   - If a PLL is stable, then set the SYSCLK fields in the CM_MR register Later SYSCLK is fed from the PLLCLK.
   - Check the switching status by monitoring the STABLE bit in the CM_SR or CM_RISR [(2)] register.

NOTE:

1. If you use CM_RISR register, it needs to check that PLL bit in CM_RISR register is cleared or not before enabling PLL. If it is set to "1", clear by writing "1" to PLL bit of ICR register.
2. If you use CM_RISR register, it needs to check that STABLE bit in CM_RISR register is cleared or not before changing the SYSCLK. If it is set to "1", clear by writing "1" to STABLE bit of ICR register.

5.2.5 Clock Monitor

The function of clock monitor is to monitor the availability of External Main Clock Oscillator. When you enable the clock monitor, you should also enable the Internal Main Clock Oscillator. The Clock Monitor has clock monitor function enable/disable control bit, clock fail detection flag, clock recovery flag, and clock monitor reset control. After entering into the STOP mode, hardware does not affect the clock monitor function. After Wake-Up from STOP mode, the status of the Clock Monitor function will stay as enabled or disabled. *Figure 5-5* illustrates the clock monitor function diagram.



- EMCLK or (EMCLK&PLL) enable
- IMCLK enable
- SYSCLK = *EMCLK or PLLCLK*
- EMCM enable
- EMCMRST don’t care (enable or disable)
EMCLK fail event is triggered.
NOTE1
- Chip reset by EMCMRST occurs.
- After reset, SYSCLK is IMCLK, not *EMCLK or PLLCLK*
- EMCMRSTS bit in CM_SR register is set to ‘1’. (Reset ID flag)
- EMCLK or (EMCLK&PLL) enable
- IMCLK enable
- SYSCLK = *IMCLK*
- EMCM enable
- EMCMRST enable
EMCLK fail event is triggered.
- Chip reset by EMCMRST occurs.
- After reset, SYSCLK is IMCLK, not *EMCLK or PLLCLK*
- EMCMRSTS bit in CM_SR register is set to ‘1’. (Reset ID flag)
- EMCLK or (EMCLK&PLL) enable
- IMCLK enable
- SYSCLK = *IMCLK*
- EMCM enable
- EMCMRST disable
EMCLK fail event is triggered.
NOTE2
- Chip reset by EMCMRST does not occur.
- SYSCLK is IMCLK without change.
- EMCKFAIL bit in CM_SR register is set to ‘1’. (Fail flag)

Figure 5-5 Clock Monitor Function

NOTE:

1. Although the reset function by a Clock Monitor is disabled, the EMCMRST occurs to make a system safe. But you should enable a Clock Monitor function before detecting clock fail.
2. Although EMCLK fails, the SYSCLK does not have any problem. Therefore, the chip runs continuously except WDT block. WDT block can work normally or not. Because WDT clock source is independent of SYCLK.

Table 5-4 describes the clock monitor control bit.

Table 5-4 Clock Monitor Control Bit

Clock Monitor	Control Bit
Clock monitor fail function enable/disable	EMCM
Clock monitor fail reset function enable/disable	EMCMRST
Clock fail detection flag	EMCKFAIL
Clock recovery flag	EMCKFAIL_END

5.2.5.1 Clock Fail

The Clock Monitor circuit samples the External Main Clock Oscillator using Internal Main Clock (IMCLK). If the sampled value is identical during the three consecutive times, then the Clock Monitor circuit decides the external oscillator failure is detected. After detecting the failure, the operation updates the EMCMRST or EMCKFAIL status.

If SYSCLK is EMCLK or PLLCLK, then the clock manager can reset when the Clock Monitor circuit detects the EMCLK failure. After EMCMRST reset, system clock becomes IMCLK and clock manager updates the reset status register. System runs by IMCLK, and the reset status register is updated by the hardware.

Figure 5-6 illustrates the external main oscillator fail and reset diagram.

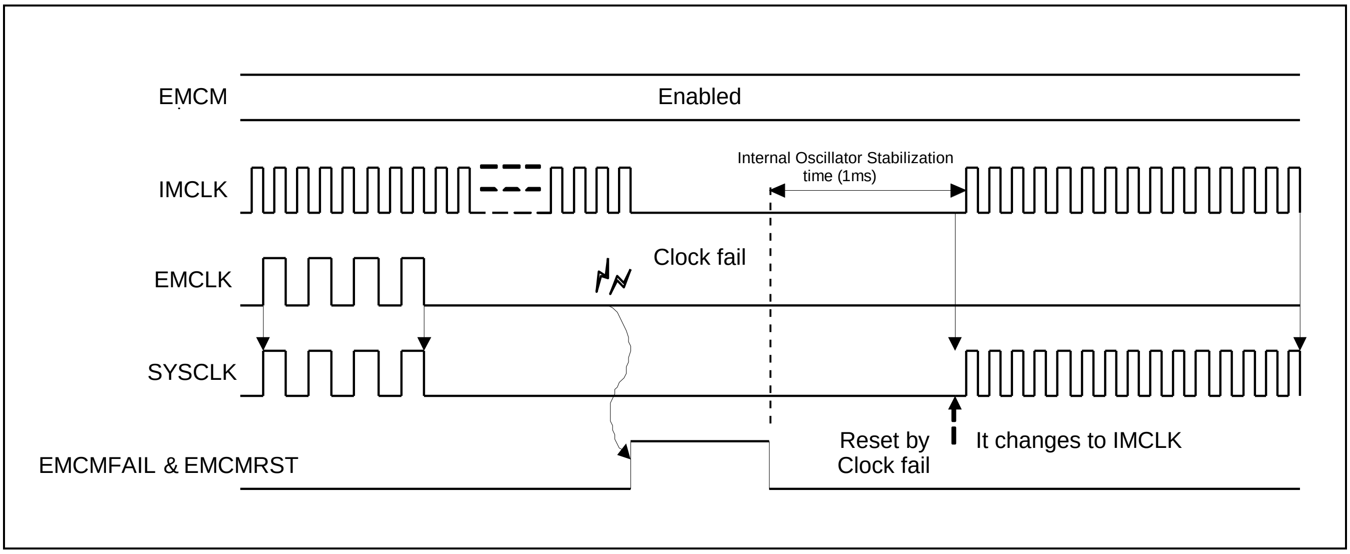


Figure 5-6 External Main Oscillator Fail and Reset

5.2.5.2 End of Clock Fail

The EMCKFAIL_END bit indicates the end of External Main Oscillator Clock failure. When External Main Oscillator Clock is recovered, EMCKFAIL_END bit in CM_SR and CM_RISR register will be set to "1". Also you can use this event with an interrupt.

When the failure disappears (EMCKFAIL_END bit in CM_SR register is set to "1" by edge recognition of the External Oscillator by Hardware), the EMCKFAIL bit clears automatically. The software is able to use the external oscillator again as the source clock. Before switching to EMCLK, the software should refer to these recommendations:

- Check the EMCLK bit in the CM_SR register
- Select the EMCLK by configuring the SYSCLK fields in the CM_MR register (Ensure to stabilize the External Main Clock Oscillator).

Figure 5-7 illustrates the end of clock fail diagram.

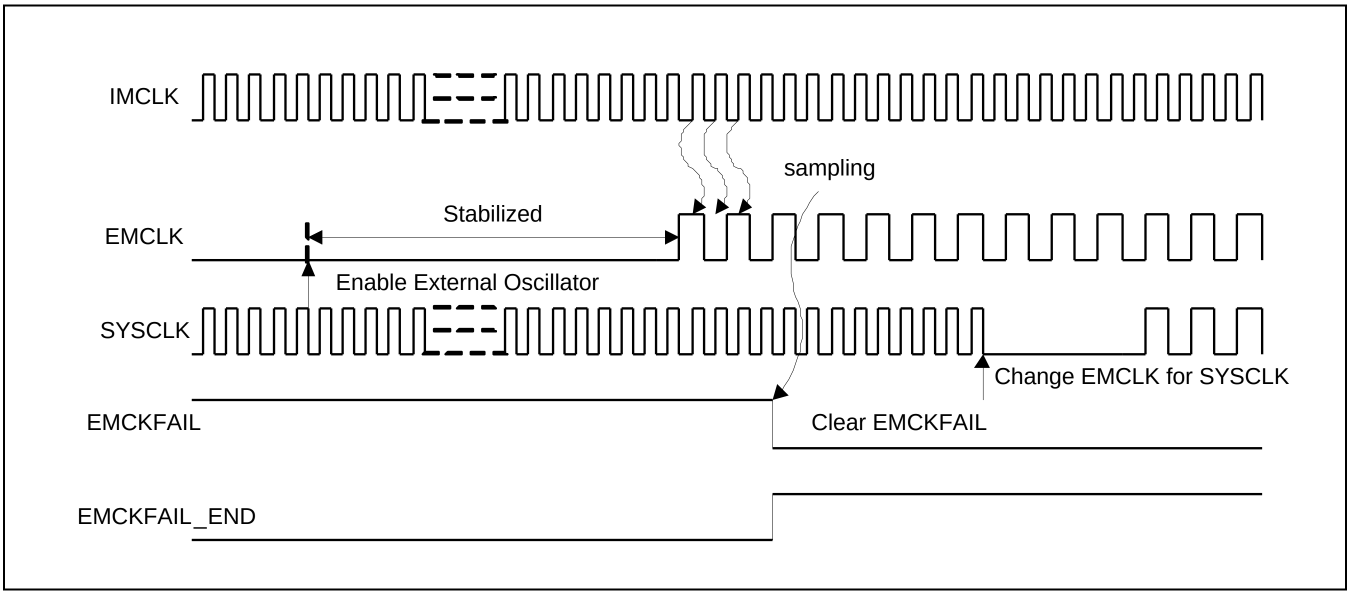


Figure 5-7 End of Clock Fail

5.2.6 Clock Out

S3FN429 includes the five clock-out ports to be configurable as an alternate function. The clock frequency to output onto COPx is the divided speed by 8 because of the maximum IO speed. You can output five different clock sources onto the COP pin using IO configuration. The clock-out ports are as follows;

- COP0: External Main Clock divided by 8 (EMCLK/8)
- COP1: Internal Main Clock divided by 8 (IMCLK/8)
- COP2: PLL Clock divided by 8 (PLLCLK/8)
- COP3: Core Clock divided by 8 (CORECLK/8)
- COP4: Peripheral Clock divided by 8 (PCLK/8)

5.3 Clock Change

5.3.1 Clock State Machine

Figure 5-8 illustrates the clock control state machine diagram.

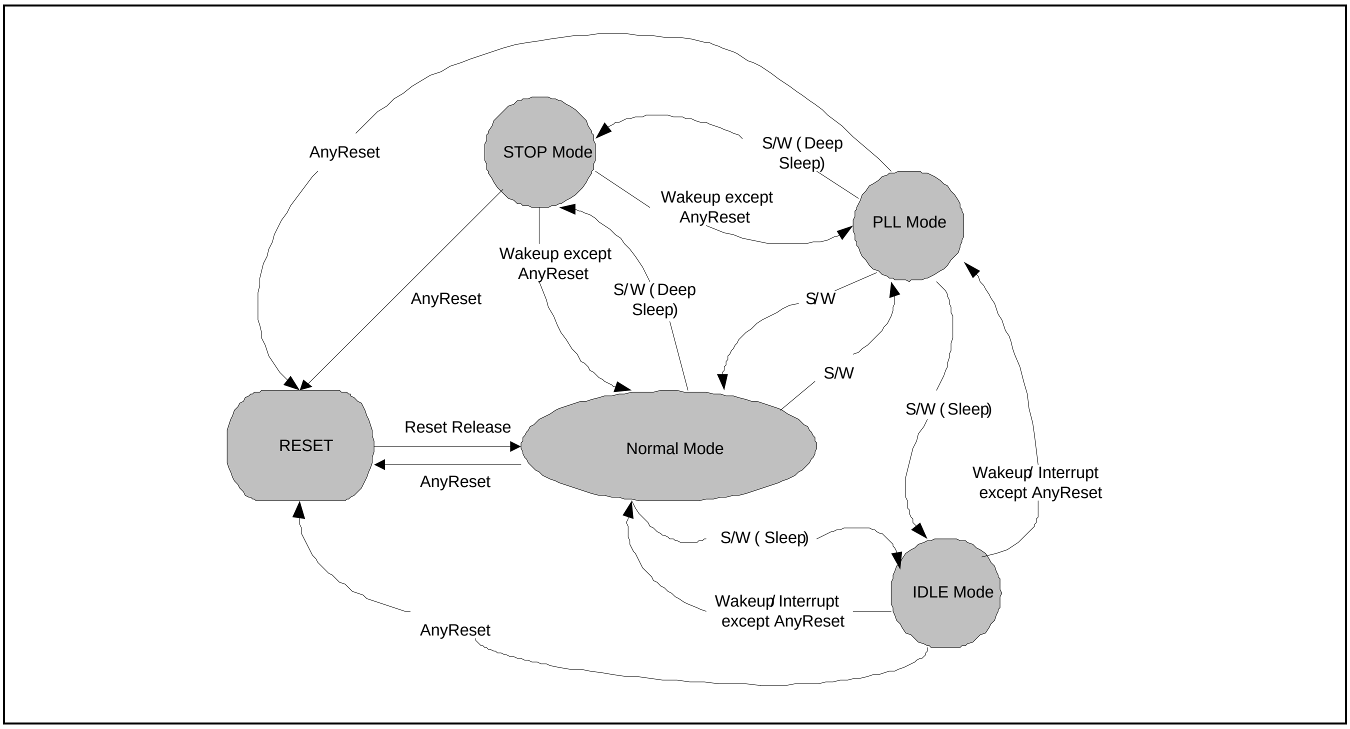


Figure 5-8 Clock Control State Machine

The SYSCLK defines the Operation Mode. The Watchdog Timer Clock is optional.

Table 5-5 describes the operation mode definition.

Table 5-5 Operation Mode Definition

Mode	CORECLK	Always	Optional Alive/Dead Clock
NORMAL MODE	IMCLK	Alive HCLK, FCLK, STCLK, PCLK	EMCLK
	EMCLK	Alive HCLK, FCLK, STCLK, PCLK	IMCLK, PLLCLK
PLL MODE	PLLCLK	Alive EMCLK before PLL enable, FCLK, HCLK, STCLK, PCLK	IMCLK
IDLE MODE	(2)	Alive FCLK Dead HCLK	EMCLK, IMCLK, STCLK, PCLK, PLLCLK
STOP MODE	(2)	Dead HCLK, FCLK, PCLK, STCLK, PLLCLK, EMCLK, IMCLK	–

NOTE:

1. The clock sources are EMCLK and IMCLK.
2. The Core (Cortex-M0) cannot take the CORECLK because the IDLE and the STOP modes disconnect the CORECLK.

The System Clock (SYSCLK) input source can use PLLCLK, EMCLK, or IMCLK as its clock source by setting SYSCLK fields in CM_MR register. If you change the System Clock, then the both source and destination clock should be stabled. Otherwise, you can put microcontroller in an unexpected status. It might generate a command error also.

Figure 5-9 illustrates the case that changes the clock source for the SYSCLK diagram. The operation of the other case remains the same.

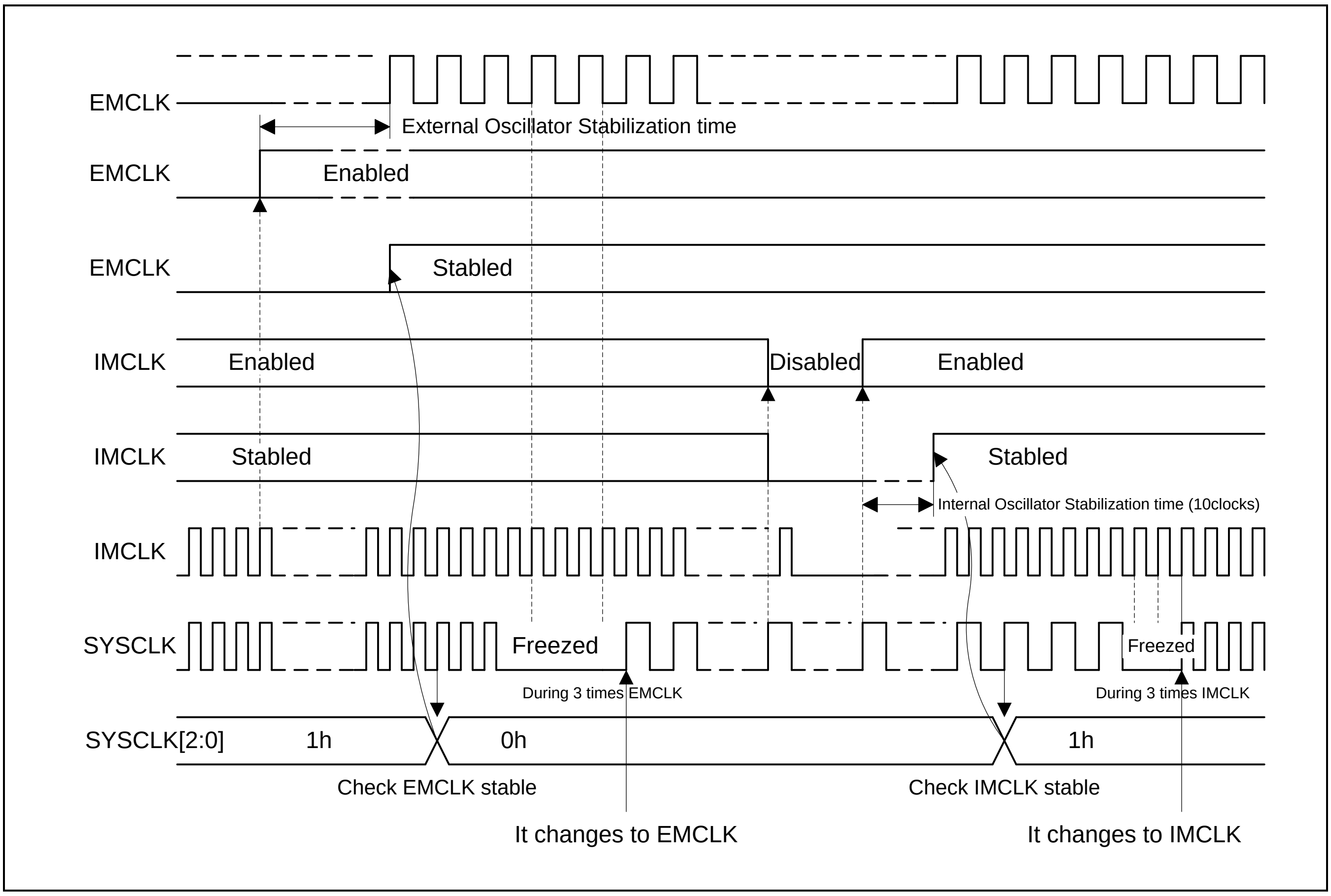


Figure 5-9 The Change Clock Source of SYSCLK

5.3.2 SYSCLK Change

SYSCLK uses for core and peripherals except the watchdog timer.

- When you use the status polling method, the sequence for SYSCLK change is the followings;
  - Check the destination clock to change is stabled or not.
    : PLL, IMCLK, or EMCLK status bits in SR register
  - If STABLE bit in CM_RISR register is "1", write "1" to STABLE bit in CM_ICR register to be clear
  - Change SYSCLK[1:0] field in CM_MR1 register
  - Read CM_STABLE bin in CM_RISR register to check the change is completed or not.
  - Write "1" to CM_STABEL bit in CM_ICR register to be clear.

When you use the interrupt method, the sequence for SYSCLK change is the followings;

- Check the destination clock to change is stabled or not.
  : PLL, IMCLK, or EMCLK status bits in SR register
- If STABLE bit in CM_MISR register is "1", write "1" to STABLE bit in CM_ICR register to be clear
- Register the interrupt handler and enable STABLE interrupt
- Change SYSCLK[1:0] field in CM_MR1 register
- The STABLE interrupt will occur when the change is completed.
- Check STABLE bin in CM_MISR register in ISR (Interrupt Service Routine).
- Write "1" to STABEL bit in CM_ICR register to be clear in ISR.

5.3.3 WDTCLK Source Change

- The sequence for WDTCLK change is the followings;
  - Check the destination clock to change is enabled and stabled or not.
  - If target clock is disabled, enable that.
    : PLL, IMCLK, or EMCLK status bits in SR register
  - Change WDTCLK[1:0] field in CM_MR1 register
  - Read WDTCLKS bin in CM_SR register to check the change is completed or not.
  - The time for switching is (3 x current source clock) + (4 x next destination clock).

5.4 Power Management

There are four operation modes. Two modes among them are the low power mode. One is IDLE, and the other is STOP mode. In low power modes, you can disable the watchdog and LVD when it is not necessary. This reduces the current consumption. After reset, the default mode is NORMAL mode.

5.4.1 Operation Modes

The four types of operation modes are:

- **NORMAL Mode**: Use this mode to supply one of the four clock sources (EMCLK or IMCLK) to the CPU as well as to all the peripherals. Power consumption increases when all the peripherals are enabled. Appropriate actions on the corresponding clock source enable bit of the CM_PCSR register controls the ON and OFF on the clock gate of the individual clock source for each peripheral device.
- **PLL Mode**: Use this mode to supply the PLLCLK for the system. That means the SYSCLK becomes the PLLCLK.
- **IDLE Mode**: This mode is one of the low power modes. In the IDLE mode, disconnecting the Clock to a CPU halts the operation. Some peripherals remain active using the software.
- **STOP Mode**: This mode halts all logics. You can activate the external interrupt, the USARTRX0, the SPI0, or a chip reset to perform the Wake-Up from the STOP mode. The microcontroller can enter the STOP mode from Normal or the PLL mode.

5.4.2 Low Power Modes and Wake-Up

Low power modes are invoked by WFI or WFE instructions. WFI stands for Wait-For-Interrupt, and WFE stands for Wait-For-Events. Events can be interrupts, a previously triggered interrupt, or an external event signal pulse via the RXEV signal. Inside the processor there is a latch for events, so a past event can wake up a processor from WFE.

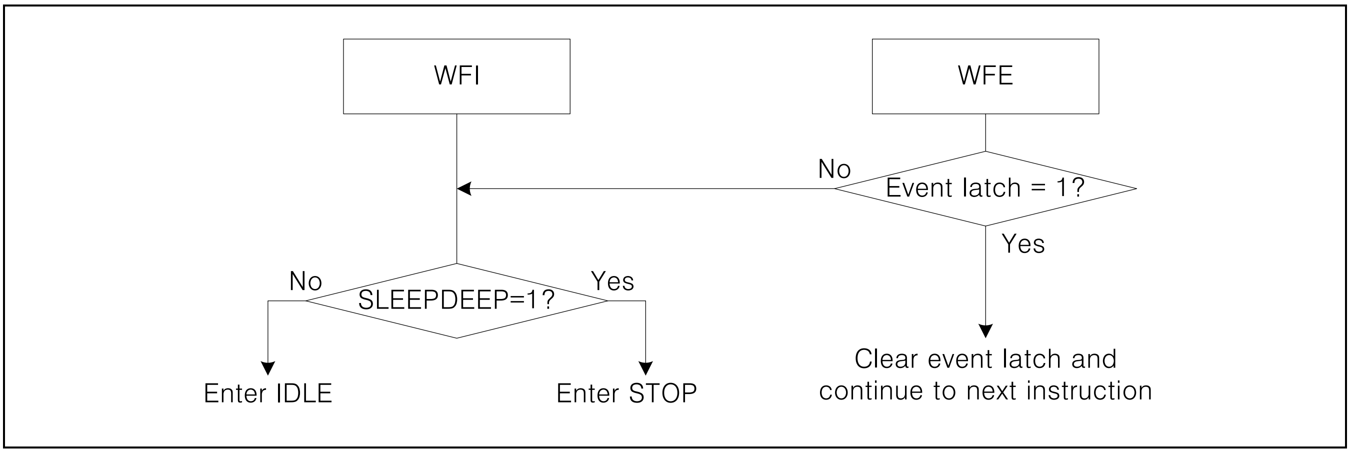


Figure 5-10 Instruction for IDLE and STOP

- WFE (Wait for Event): The WFE instruction. The WFE causes an entry into the IDLE/STOP mode conditional on the value of a one-bit event register. When the processor executes a WFE instruction, it checks this register:
  - If the register is 0, then the processor stops executing the instructions and enters IDLE/STOP mode.
  - If the register is 1, then the processor clears the register to 0, and continues executing the instructions without entering IDLE/STOP mode.

- WFI (Wait for Interrupt): The WFI instruction. The WFI causes immediate entry into the IDLE/STOP mode. When the processor executes a WFI instruction, the WFI stops executing the instructions and enters the IDLE/STOP mode.

5.4.2.1 Enter IDLE Mode

The Idle mode is applicable when it requires Wake-up with the minimum latency. When it enters IDLE mode, controller disconnects only the core clock.

There are two options available to select the IDLE mode entry mechanism depending on the SLEEPONEXIT bit in the Cortex-M0 System Control Register.

- Entry condition:
  - IDLE (Sleep-Now): If it clears the SLEEPONEXIT bit of System Control Register in Cortex-M0, then the MCU enters IDLE mode as soon as it executes the WFI or WFE instruction.
  - IDLE (Sleep-on-Exit): If the SLEEPONEXIT bit of System Control Register in Cortex-M0 is set, then the MCU enters IDLE mode as soon as it exits the lowest priority Interrupt Status Register (ISR).

- Entry sequence:
  - Configure Wake-Up source
  - Configure clock source
  - Configure Oscillator and PLL.
  - Select the WFE/WFI by setting the IDLEW bit in the CM_CSR and the CM_CCR register
  - If you want to use the WFE, then set the RXEV bit in the CM_MR register
  - If you want to use the WFI, then copy the NVIC Interrupt Set-Enable Registers into the CM_NISR register
  - Execute entry condition

The System Control Register of the Cotex-M0 NVIC has the Sleep-on-Exit bit (SLEEPONEXIT). If it is set, then the processor completes the execution of an exception handler. Then it returns to the thread mode and immediately enters into the IDLE mode. Use this mechanism in applications that only requires the processor to run when an exception occurs.

After exiting from IDLE mode, if you use the IMCLK without the external oscillator, then you should not perform the accuracy-critical code and the PLL. The processor should wait until EMCLK is enabled and stabilized.

By configuring the PCLK in the CM_CSR register, it is possible to disconnect the clock supply to peripherals. Thus, you can further optimize the power.

5.4.2.2 Exit IDLE Mode

If you use the WFI instruction to enter the IDLE mode, then any peripheral interrupt acknowledge by the Nested Vectored Interrupt Controller (NVIC) wakes up the device from the IDLE mode. If you use the WFE instruction to enter the IDLE mode, then the MCU exits the IDLE mode as soon as an event occurs.

Table 5-6 describes the IDLE on Sleep-Now.

Table 5-6 IDLE on Sleep-Now

Sleep-Now Mode	Description
Mode Entry	Use the WFI or WFE while: • SLEEPDEEP = 0 and • SLEEPONEXIT = 0 For more information, refer to Cortex-M0 the System Control Register.
Mode exit	If you use WFI for entry: Interrupt If you use WFE for entry: Wake-Up event
Wakeup latency	None

Table 5-7 describes the IDLE on Sleep-Exit.

Table 5-7 IDLE on Sleep-on-Exit

Sleep-on-Exit Mode	Description
Mode Entry	Use the WFI or WFE: • SLEEPDEEP = 0 and • SLEEPONEXIT = 1 For more information, refer to Cortex-M0 the System Control Register.
Mode exit	Interrupt
Wakeup latency	None

5.4.2.3 Enter STOP Mode

The STOP mode is based on the SLEEPDEEP in the Cortex-M0 core. All clocks stop operating. It disconnects the clock supply to the SYSCLK in entry of the STOP mode and connects in exit of the STOP mode.

- Entry Condition
  - STOP (Sleep-Now): If the SLEEPDEEP bit of the System Control Register in the Cortex-M0 is set to 1, then the MCU enters the STOP mode as soon as it executes the WFI or the WFE instruction.
  - STOP (Sleep-on-Exit): If the SLEEPONEXIT and the SLEEPDEEP bits of the System Control Register in the Cortex-M0 are set to 1, then the MCU enters the STOP mode as soon as it exits the lowest priority of the ISR register.

- Entry Sequence
  - Configure Wake-Up sources
  - Configure Clock Source
  - Configure Oscillator and PLL
  - If you want to use the WFI, then copy the NVIC Interrupt Set-Enable Registers into the CM_NISR register
  - Execute entry condition

The System Control Register of the Cotex-M0 NVIC has the Sleep-on-Exit bit (SLEEPONEXIT). If it is set to 0, then the processor completes the execution of an exception handler. Later it returns to the thread mode and immediately enters into the STOP mode.

Caution: If you don't the external main-oscillator when pin7 and pin8 are defined XOUT and XIN function, you should tie XIN to ground. XOUT should be opened.

5.4.2.4 Exit STOP Mode

The microcontroller can exit the STOP mode by issuing an interrupt or a Wake-Up event. After the exit from STOP mode, the clock source of the SYSCLK can differ by condition before entering the STOP mode.

5.4.2.5 Wake-Up from IDLE/STOP Mode

Before the entry of IDLE or STOP mode, executing WFE or WFI allows the processor to Wake-Up by an event or interrupts. As the previous section describes, enable the interrupt that you use for Wake-Up source before the entry of the IDLE or the STOP mode.

Table 5-8 describes the clock status on STOP and wake-up.

Table 5-8 Clock Status on STOP and Wake-Up

–	Normal or PLL Mode	STOP	Wake-Up
EMCLK	RUN	STOP	RUN
	STOP	STOP	STOP
IMCLK	RUN	STOP	RUN
	STOP	STOP	STOP@FWAKE = 0 RUN@FWAKE = 1

Figure 5-11 illustrates the interrupt and event diagram.

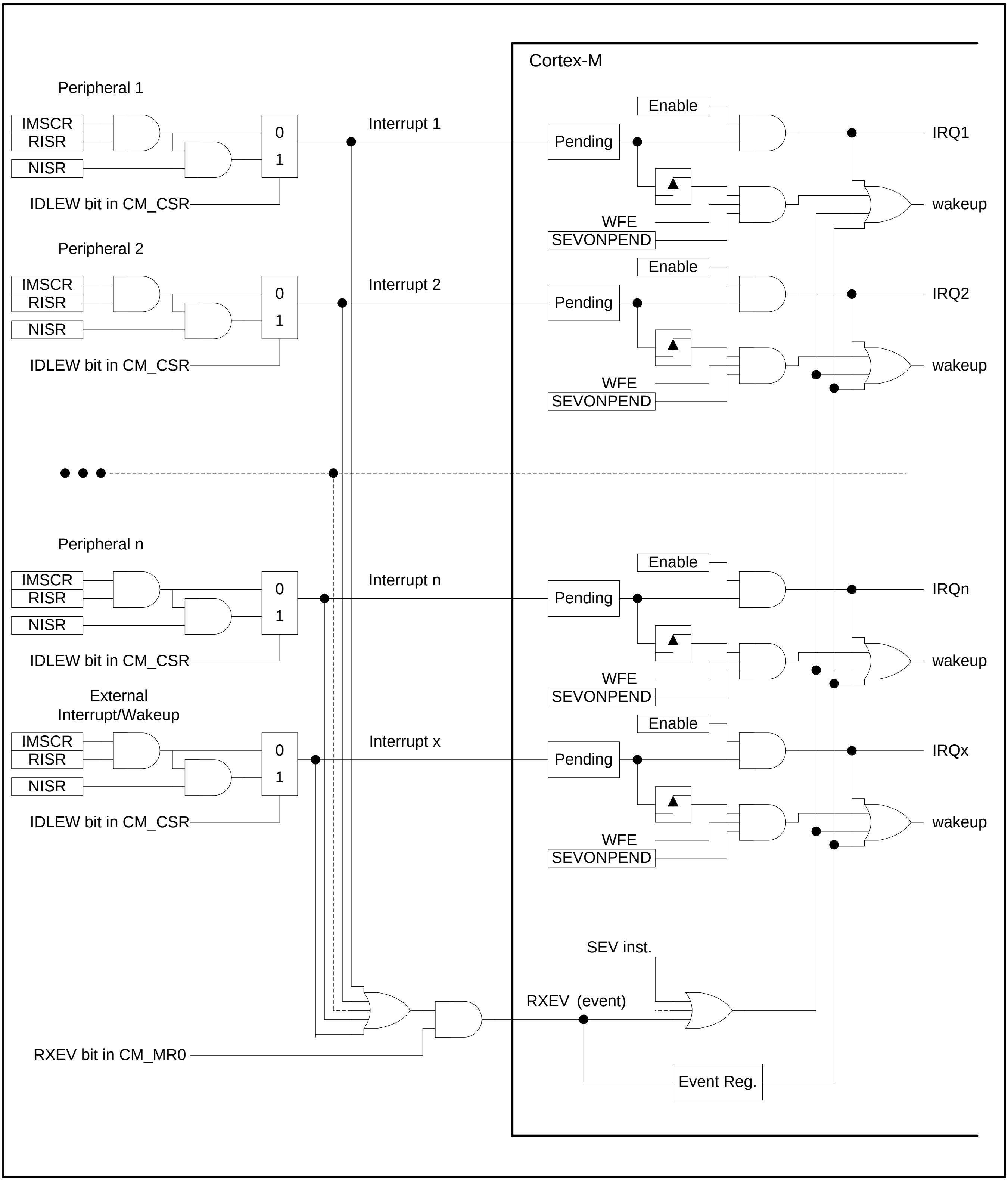


Figure 5-11 Interrupt and Event

Figure 5-12 illustrates the different handling process for interrupt and event in IDLE or STOP mode.

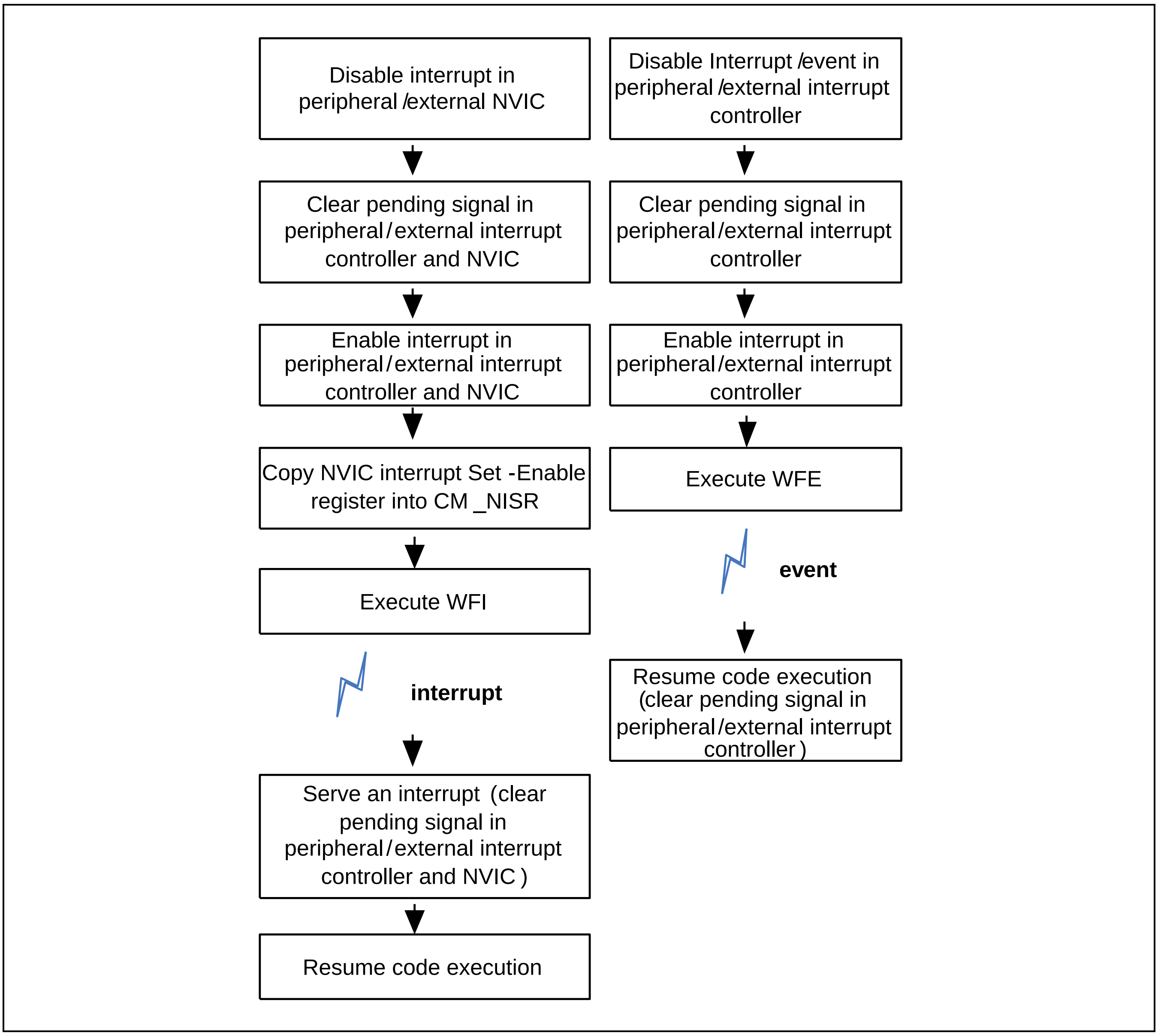

Disable interrupt in peripheral /external NVIC
Clear pending signal in peripheral/external interrupt controller and NVIC
Enable interrupt in peripheral/external interrupt controller and NVIC
Copy NVIC interrupt Set -Enable register into CM _NISR
Execute WFI
interrupt
Serve an interrupt (clear pending signal in peripheral/external interrupt controller and NVIC)
Resume code execution
Disable Interrupt /event in peripheral /external interrupt controller
Clear pending signal in peripheral/external interrupt controller
Enable interrupt in peripheral/external interrupt controller
Execute WFE
event
Resume code execution (clear pending signal in peripheral/external interrupt controller)

Figure 5-12 Different Handling Process for Interrupt and Event in IDLE or Stop Mode

5.5 External Events (External Interrupts and Wake-up Event)

S3FN429 supports 26 external input events including external interrupts (Refer to *Table 5-14*) except other internal peripheral and core event/interrupts. You can use these as a specific event trigger in an operating mode or wake-up trigger in the low power mode (IDLE or STOP). The maximum number that you can use at the same time is eight. You can select any external event using EESRCx field in CM_EECR0 and CM_EECR1 registers. There are the values of EESRCx field into *Table 5-14*.

Table 5-9 External Event Function

Core status	External Event Function
OPERATING	Specific event trigger (External Interrupt)
IDLE (SLEEP)	Core wake-up trigger
STOP (DEEPSLEEP)	Core wake-up trigger

5.5.1 Operating Mode

You can use something among 24 external interrupts in general operating modes. To use external interrupt, the corresponding port should be configured as EXI function. Each external interrupt can be independently configured with CM_EECR0 and CM_EECR1 register. You can select the trigger edge type (rising, falling or both). Their interrupts can be enabled or disable through CM_EIMSCR register. The occurred interrupt status can be read through CM_ERISR or CM_EMISR. To clear interrupt status, you should use CM_EICR register.

The example configuration for EXI2 is the followings;

a) Define interrupt handler (service) routine for EEIA vector. Refer to INTC and CM.
b) EXI2 pin configuration. Refer to IOCONF.
c) Assign EXI2 interrupt source to EE0 interrupt. Refer to CM.
d) Enable EEIA Vector
e) Enable (Unmask) EE0 interrupt defined with an EXI2 interrupt source
f) Check an interrupt pending status. If external interrupt signal is asserted through EXI2 port, EE0 bit in CM_RISR and CM_MISR register will be set to "1".
g) Clear pending interrupt. You can clear when you write "1" to EE0 bit in CM_ICR register. EE0 bit in CM_EEMISR and CM_EERISR register should be 0 (clear).

```
ISR_EEIA() /* Interrupt handler for EE0(EEIA vector) */ --- (a)
{
/* CM_EEMISR → EE0 and CM_EERISR → EE0 should be 1 (set status)*/ --- (f)
CM_EEICR → EE0 = 1; /* Clear interrupt pending bit */ --- (g)
UserDefinedOperation(); /* Handling code for EE0 interrupt */
}
-------------------------------------------------------------------------------------------
IOCONF_MLR0 → IO0_3_FSEL = 11'b /* EXI2 defined as third function of pin12 */ --- (b)
CM_EECR0 = (1'b << 7)|(01'b << 5)|(000010'b << 0) /* (Enable| Edge Type| Source) */ --- (c)
NVIC_ISER0 = (1'b << 16) /* NVIC_INT16 */ --- (d)
CM_EEIMSCR → EE0 = (1'b << 0) --- (e)
```

5.5.2 Idle Modes

IDLE mode halts the core clock. So instruction in user code is not executed any more. But the peripherals such like a timer can either operate or stops by PCLK status in CM_SR register. To wake-up from IDLE mode can be done by internal interrupts such like a timer interrupt or external event interrupts.

5.5.3 Stop Modes

In case of STOP mode, all clocks for device logic halt. Only 26 external event interrupts can wake-up device from STOP mode.

Figure 5-13 is the simple diagram for external interrupt. It shows information about registers and corresponding vector to configure and control external interrupt.

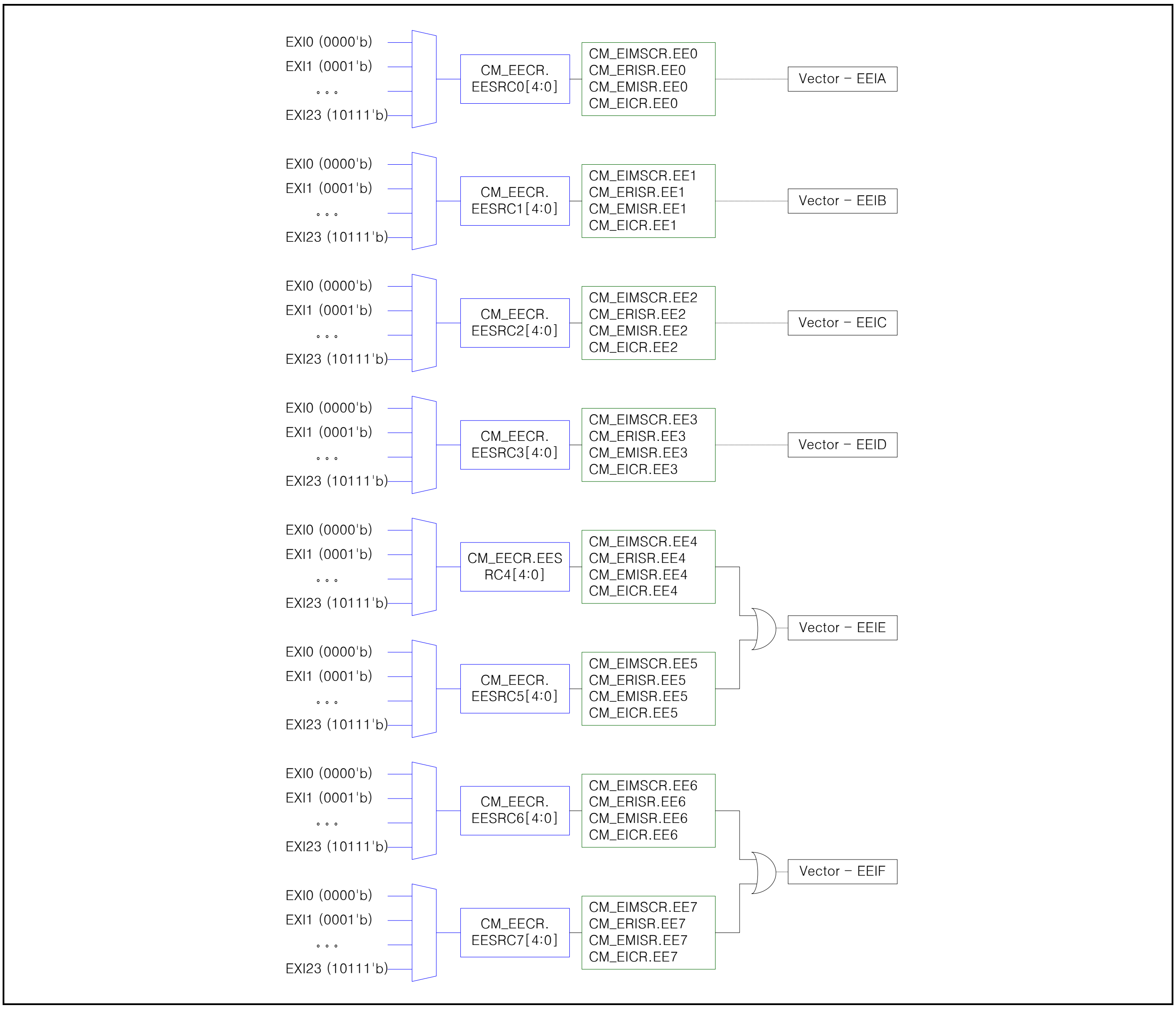


Figure 5-13 External Interrupt Diagram

5.6 Reset Management

The seven sources for a reset are:

- External Pin Reset
- Power-on Reset
- Low Voltage Detect Reset
- External Main Clock Monitor Reset
- Watchdog Timer Reset
- Software Reset
- CPU Request Reset

Figure 5-14 illustrates the reset sources.

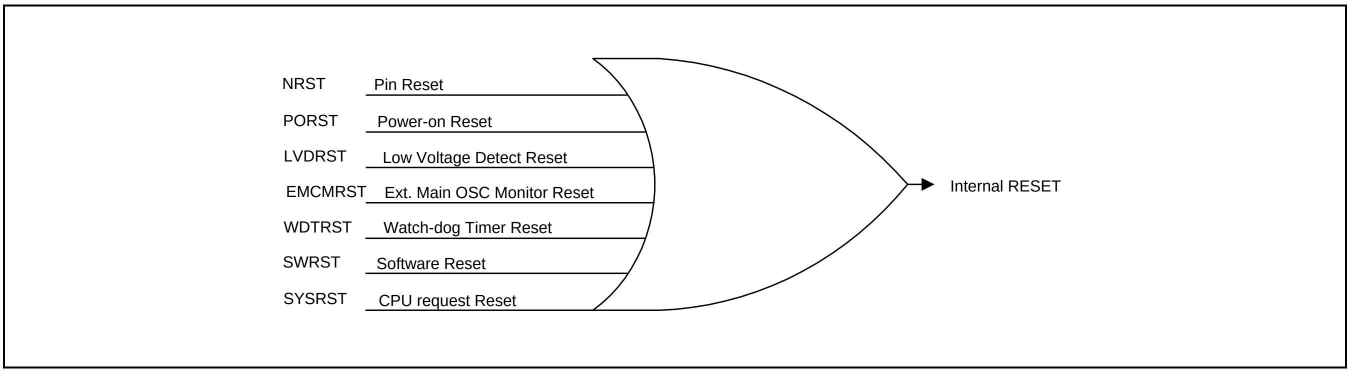


Figure 5-14 Reset Sources

The Clock Manager has status bits that show the reset logging information. The CM_SR (Clock Manager Status Register) includes the information of reset logging ID from 24 to 31-bit. When reset occurs, each source triggers a reset and updates the status of each reset ID. You can clear all reset ID flags by software. Especially you should clear PORRSTS and LVDRSTS bits only by software. Other reset ID flags clear if any reset occurs.

Table 5-10 describes the Reset ID flag.

Table 5-10 Reset ID Flag

Reset Status Bit	CM_SR	Description
SWRSTS	[24]	The software (register control) generates the last reset
NRSTS	[25]	A signal asserted on the nRESET pin generates the last reset
LVDRSTS	[26]	Low voltage detection generates the last reset
WDTRSTS	[27]	Watchdog timer generates the last reset
PORRSTS	[28]	Power-on generates the last reset
EMCMRSTS	[30]	Clock monitor function generates the last reset
SYSRSTS	[31]	CPU request generates the last reset

5.6.1 nRESET Pin Reset (NRST)

When the unmaskable nRESET pin asserts as "Low", then it generates the internal hardware reset signal. After assertion of nRESET pin, the MUC enters into the reset state regardless of the previous states. After the nRESET resets, the NRSTS bit in the CM_SR register sets to "1". User (software) or other resets (hardware) can clear this bit.

5.6.2 Power-On Reset (PORST)

The Power-on Reset circuit is built in the microcontroller. When power initially passes through the microcontroller or VDD drops below the V_{POR} (POR voltage level, typically 1.2 V), then the POR circuit holds the microcontroller in reset until the VDD rises above the V_{LVR} (LVD reset level).
After Power-on Reset, the PORSTS bit in the CM_SR register sets to "1". Although a reset by other reset sources occur, it does not clear the status (1 or 0) of this bit and the status remains the same. Only you can clear this bit by software.

5.6.3 LVD Reset (LVDRST)

By default, it enables the LVD Reset at reset (see the LVDRSTEN bit in the CM_MR register). After a reset, you can configure the LVD reset by using the LVDRST bit in the CM_SR register. You can also configure the level of the LVD reset by using the LVDRL[2:0] fields in the CM_MR register. After the LVD reset, the LVDRSTS bit in the CM_SR register sets to "1". Although, a reset by other reset sources occurs, it does not clear the status (1 or 0) of this bit and the status remains the same. Only you can clear this bit by software.

5.6.3.1 LVD Interrupt

You can enable or disable the LVD interrupt level detection by controlling the LVDINTEN bit in CM_MR register. You can mask or unmask the LVD interrupt by controlling the LVDINT bit in CM_IMSCR register. If LVD detects the interrupt level after LVDINT bit in CM_IMSCR register is set to 1, LVDINT in CM_MISR register is set to 1. You can configure the level of LVD interrupt using LVDIL[2:0] fields in CM_MR register.

To clear the pending interrupt, first clear the status bit (LVDINT) in CM_ICR and then the corresponding pending bit in NVIC.

Figure 5-15 illustrates the LVD block diagram.

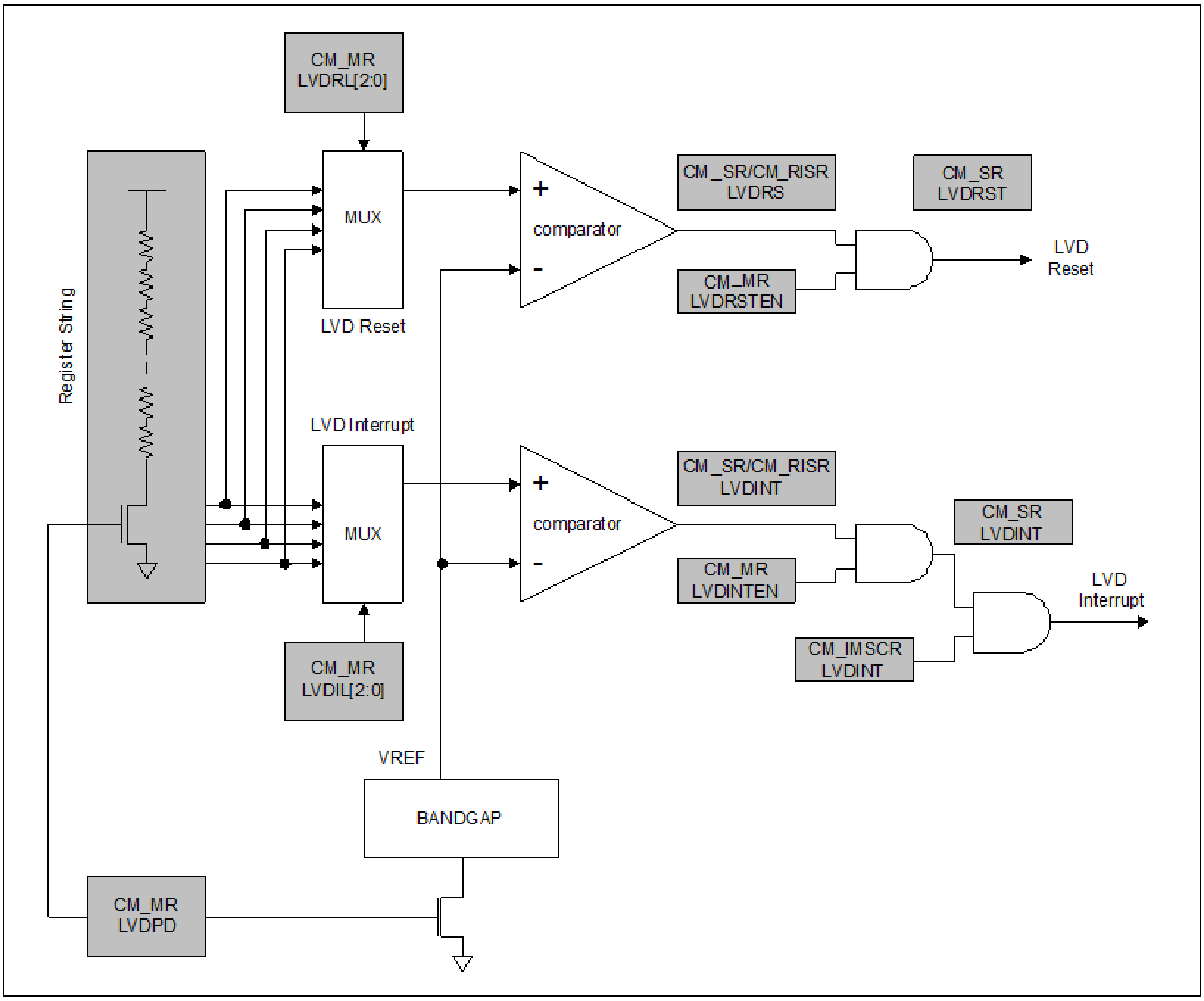


Figure 5-15 LVD Block Diagram

The level for LVD reset should be less than the level of LVD interrupt. If LVD detect the target level when those levels are the same, LVD generates the reset. That means the priority of LVD reset is higher than LVD interrupt.

5.6.4 External Main Clock Monitor Reset (EMCMRST)

For more information, Refer to Chapter 5.2.5 Clock Monitor.

5.6.5 Watchdog Timer Reset (WDTRST)

Watchdog-Timer Reset occurs when Watchdog-Timer counter value reaches 0, overflow event, under the condition that both watchdog timer and reset are enabled.
After Watchdog Timer Reset, WDTRSTS bit in CM_SR register sets to "1". The user (software) or other resets (hardware) can clear this bit.

5.6.6 Software Reset (SWRST)

Software reset occurs when it writes "1" to CM_SRR register. After the software reset, the SWRSTS bit in CM_SR register sets to "1". The user (software) or other resets (hardware) can clear this bit.

5.6.7 CPU Request Reset (SYSRST)

CPU initializes the device state itself when it writes "1" to SYSRESETREQ bit in Application Interrupt and Reset Control Register of Cortex-M0. After system reset, SYSRSTS bit in CM_SR register sets to "1". The user (software) or other resets (hardware) can clear this bit.

5.7 Basic Timer

The Basic Timer (BT) is a release timer at reset or the Wake-Up from the STOP mode. It blocks the clock supply to system and controls the reset/STOP release time for the predefined time. While the power and clock supply becomes unstable as soon as you power on the processor or the oscillator starts to run.

The reference clock for timer is the SYSCLK. Smart option defines the SYSCLK. The smart option generates the source clock for the BT with clock divider. The BT counts with a different clock divider. Smart option defines the reset value of the clock divider.

After a reset, you can control the divider value by changing the BTCDIV[3:0] field in the CM_BTCDR register. For instance, to shorten the Wake-Up latency in the exit of the STOP mode, the software can configure the number of BT divider or count value before the entry of STOP mode.

The reset count value of the BT is 0x100. In other words, when the 8th bit on the BT is sets to "1", it releases system reset or Wake-Up signals.

When the BT count value is 256 and each divider value splits from 1 to 4096, then the frequency of the BT input clock splits into: 1 MHz, 4 MHz, 8 MHz, 16 MHz, 20 MHz, and 40 MHz.

Table 5-11 describes the 256 counting time by BT.

Table 5-11 256 Counting Time by BT

256 Count Divider	40 MHz (μs)	20 MHz (μs)	16 MHz (μs)	8 MHz (μs)	4 MHz (μs)	1 MHz (μs)
/4096	26214.4	52428.8	65536	131072	262144	1048576
/2048	13107.2	26214.4	32768	65536	131072	524288
/1024	6553.6	13107.2	16384	32768	65536	262144
/512	3276.8	6553.6	8192	16384	32768	131072
/256	1638.4	3276.8	4096	8192	16384	65536
/128	819.2	1638.4	2048	4096	8192	32768
/64	409.6	819.2	1024	2048	4096	16384
/32	204.8	409.6	512	1024	2048	8192
/16	102.4	204.8	256	512	1024	4096
/8	51.2	102.4	128	256	512	2048
/4	25.6	51.2	64	128	256	1024
/2	12.8	25.6	32	64	128	512
/1	6.4	12.8	16	32	64	256

The SYSCLK source and the BT clock divider in smart option decide the BT counting frequency. MCU releases the chip reset when BT completes 256 counting. To optimize on system, you can change the reset release time by BT. You can do this by re-programming the flash smart option.

Figure 5-16 illustrates the BT count in reset sequence.

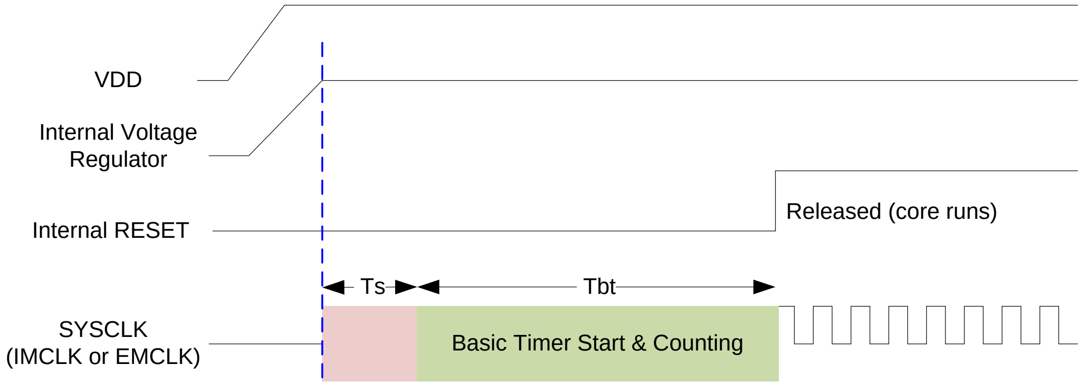
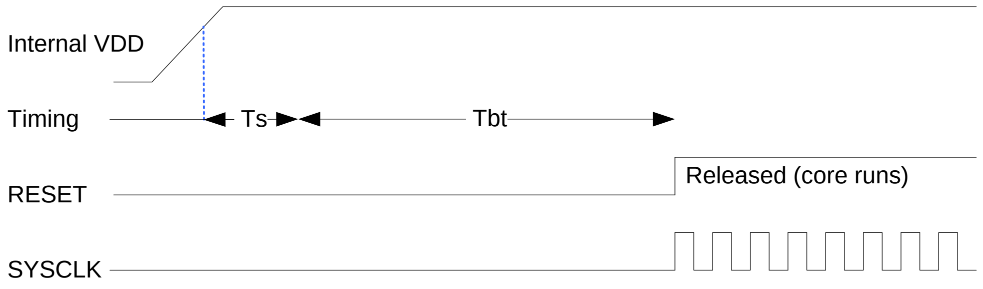


Figure 5-16 RESET@EMCLK



Ts : The time for an internal initialization ≈ 1ms
(init-clock stabilization, smart option configuration, internal trimming etc.)

Tbt : The time for a system initialization by basic timer
It can be configurable because a clock divider for basic timer is decided by smart option.
If user wants to optimize the start-up time, user can use it considering the condition
(i.e. clock source) of a target system.

Figure 5-17 Start-Up with Basic Timer

The Ts + Tbt value which is defined as 1st code run from exit of reset source is dependent on both the smart option value which controls the start-up clock source and the Basic Timer clock division ratio and the reset source. The most optimized Ts + Tbt value can be obtained with the IMCLK as start-up clock source and 1/1 as Basic Timer clock division ratio. With minimum setting of clock division ratio using smart option, the Tbt value can be less than 20us, and Ts value is the most important value to affect the start-up time.

The Most Optimized Ts + Tbt value: Max. 1 ms (Reset source: POR or LVD Reset, Start-up clock source: IMCLK)
The Most Optimized Ts + Tbt value: Max. 200 us (Reset source: Others, Start-up clock source: IMCLK)

5.8 Fast Wake-up

For fast Wake-Up from STOP mode, you can use the "FWAKE" control bit. You should enable the FWAKE bit before the STOP mode. When fast Wake-Up releases the system from STOP mode, the SYSCLK is the IMCLK regardless of enable/disable status before the STOP mode. It is not required to use the fast Wake-Up when the SYSCLK is the IMCLK before the STOP mode.

Figure 5-18 illustrates the fast wake-up.

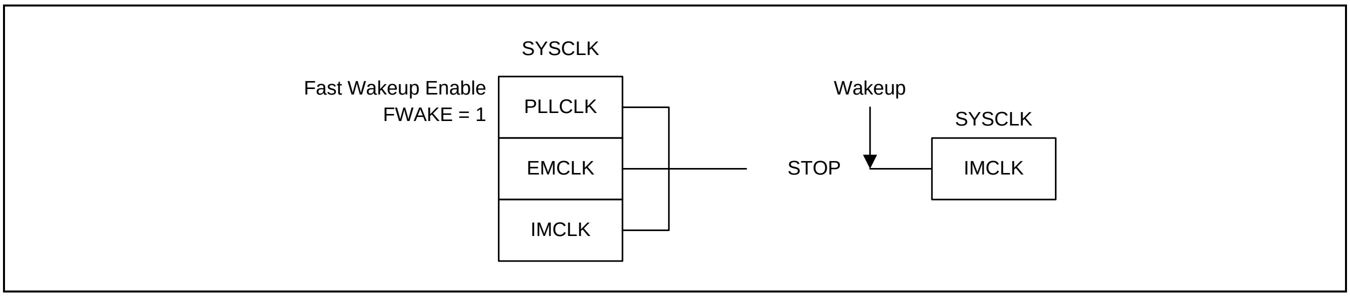


Figure 5-18 Fast Wake-Up

Figure 5-19 illustrates the BT and exit of stop mode when FWAKE is "0".

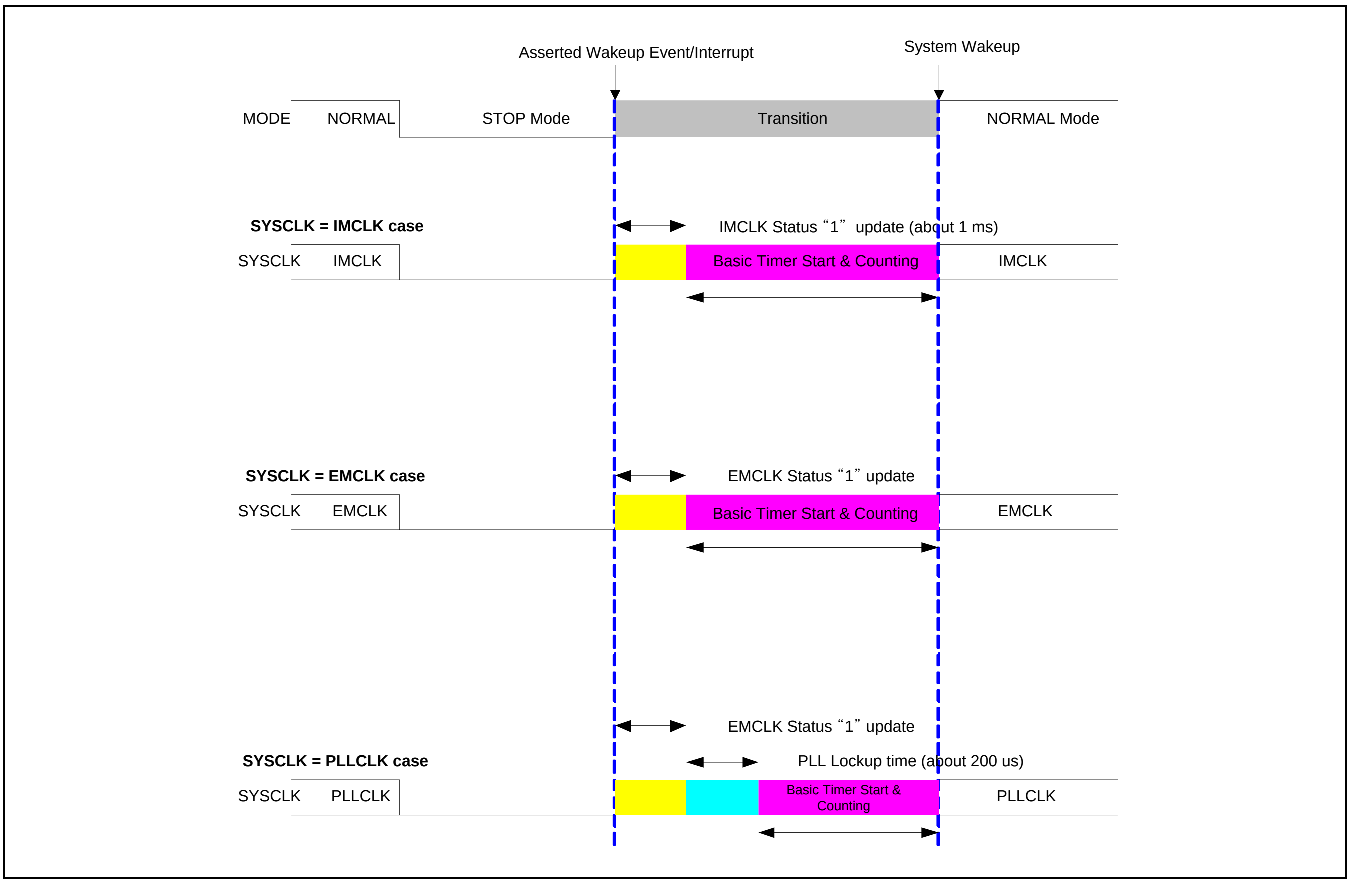


Figure 5-19 Basic Timer and Exit of Stop Mode when FWAKE is "0"

Figure 5-20 illustrates the BT and exit of stop mode when FWAKE is "1" (SYSCLK = IMCLK).

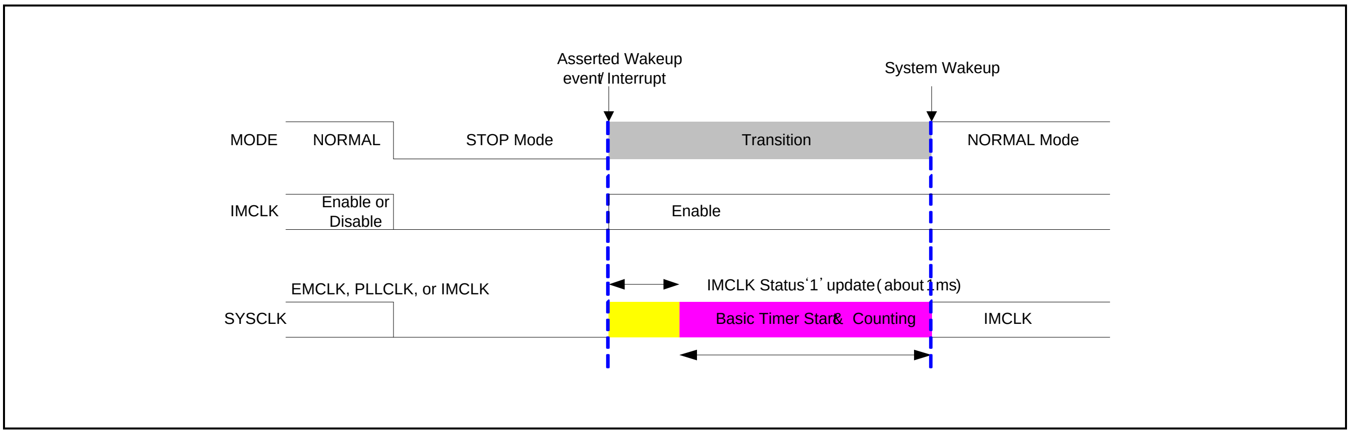


Figure 5-20 Basic Timer and Exit of Stop Mode when FWAKE is "1" (SYSCLK = IMCLK)

5.9 Register Description

5.9.1 Register Map Summary

- Base Address: 0x4002_0000

Register	Offset	Description	Reset Value
CM_IDR	0x0000	ID register	0x0001_001C
CM_SRR	0x0004	Software reset register	0x0000_0000
CM_CSR	0x0008	Control set register	0x0000_0000
CM_CCR	0x000C	Control clear register	0x0000_0000
CM_PCSR0	0x0010	Peripheral clock set register 0	0x0000_0000
RSVD	0x0014	Reserved	0x0000_0000
CM_PCCR0	0x0018	Peripheral clock clear register 0	0x0000_0000
RSVD	0x001C	Reserved	0x0000_0000
CM_PCKSR0	0x0020	Peripheral clock status register 0	0x0000_0002
RSVD	0x0024	Reserved	0x0000_0000
CM_MR0	0x0028	Mode register 0	0x0000_0828
CM_MR1	0x002C	Mode register 1	0x0000_001X
CM_IMSCR	0x0030	Interrupt mask set/clear register	0x0000_0000
CM_RISR	0x0034	Raw interrupt status register	0x0000_001B
CM_MISR	0x0038	Masked interrupt status register	0x0000_0000
CM_ICR	0x003C	Interrupt clear register	0x0000_0000
CM_SR	0x0040	Status register	0xYY80_001B
CM_SCDR	0x0044	System clock divider register	0x0000_0007
CM_PCDR	0x0048	Peripheral clock divider register	0x0000_0000
RSVD	0x004C	Reserved	0x0000_0000
RSVD	0x0050	Reserved	0x0000_0000
RSVD	0x0054	Reserved	0x0000_0000
CM_PSTR	0x0058	PLL stabilization time register	0x0000_0154
CM_PDPR	0x005C	PLL divider parameter register	0x0000_0000
RSVD	0x0060	Reserved	0x0000_0000
RSVD	0x0064	Reserved	0x0000_0000
RSVD	0x0068	Reserved	0x0000_0000
RSVD	0x006C	Reserved	0x0000_0000
CM_BTCDR	0x0070	Basic timer clock divider register	0x0000_000X
CM_BTR	0x0074	CM basic timer register	0x0000_0100
CM_EECR0	0x0078	External event control register 0	0x0000_0000
CM_EECR1	0x007C	External event control register 1	0x0000_0000
RSVD	0x0080	Reserved	0x0000_0000

Register	Offset	Description	Reset Value
RSVD	0x0084	Reserved	0x0000_0000
CM_EEIMSCR	0x0088	External event interrupt mask set/clear register	0x0000_0000
CM_EERISR	0x008C	External event raw interrupt status register	0x0000_0000
CM_EEMISR	0x0090	External event masked interrupt status register	0x0000_0000
CM_EEICR	0x0094	External event interrupt clear register	0x0000_0000
CM_NISR	0x0098	NVIC interrupt status register	0x0000_0000
RSVD	0x009C	Reserved	Undefined
RSVD	0x00A0	Reserved	Undefined
CM_PSR	0x00A4	Power status register	0x0000_0002

NOTE: Smart option decides X value and real reset source decides Y at reset time.

5.9.1.1 CM_IDR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0000, Reset Value = 0x0001_001C

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	Identification Code Register This field stores the ID code for the corresponding IP.	0x0001_001C

5.9.1.2 CM_SRR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWRSTKEY																RSVD															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
SWRSTKEY	[31:16]	W	Software Reset Key This is the key for Write access into the CM_SRR register. Writing in CM_SRR register is only effective if the SWRSTKEY is equal to 0xA66A. In other words, to generate a software chip reset, you should write the SWRST bit with SWRSTKEY value (0xA66A).	0
RSVD	[15:1]	R	Reserved	0
SWRST	[0]	W	CM Software Reset 0 = No effect 1 = Performs software reset operation and auto-clears This reset generates chip reset. It is one of the reset sources. When the software reset occurs, SWRSTS bit in CM_SR register will be set to "1" after reset.	0

5.9.1.3 CM_CSR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD								EMCM	EMCMRST	RSVD										IDLEW	RSVD	PCLK	STCLK	PLL	RSVD	FWAKE	RSVD			IMCLK	EMCLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	W	W	R	R	R	R	R	R	R	R	R	R	W	R	W	W	W	R	W	R	R	R	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	R	Reserved	0
EMCM	[23]	W	External Main Clock Monitor Function Enable Control Bit (2) 1 = Enables the Main Clock Monitor function This function is to detect the failure of an external main clock.	0
EMCMRST	[22]	W	External Main Clock Monitor Reset Enable Control Bit 1 = Enables Reset by Main Clock Monitor When it detects a clock failure, a clock monitor circuit generates the chip reset.	0
RSVD	[21:12]	R	Reserved	0
IDLEW	[11]	W	IDLEW Control Bit 1 = Sets IDLEW	0
RSVD	[10]	R	Reserved	0
PCLK	[9]	W	PCLK Enable Control Bit in IDLE Mode 1 = Connects PCLK supplied to peripherals in IDLE Mode PCLK is supplied to peripherals in IDLE Mode.	0
STCLK	[8]	W	STCLK Enables Control in IDLE Mode 1 = Enables STCLK in IDLE mode This is one of the options with the IDLE mode. If the STCLK bit sets to "1" before entering the IDLE mode, then the STCLK supplies to the systick timer in the IDLE mode.	0
PLL	[7]	W	PLL ON Control Bit 1 = Enables PLL After enabling PLL, you should check the PLL status bit in CM_SR register to use the PLL Clock which is stable.	0
RSVD	[6]	R	Reserved	0
FWAKE	[5]	W	Fast Wake-Up Enable Control Field 1 = Enables FWAKE	0

Name	Bit	Type	Description	Reset Value
			If a microcontroller enters stop mode after being enable a fast Wake-Up, SYSCLK becomes IMCLK after Wake-Up.	
RSVD	[4:2]	R	Reserved	0
IMCLK	[1]	W	Internal Main Clock Enable Control Bit (1) 1 = Enables IMCLK clocking	0
EMCLK	[0]	W	External Main Clock Enable Control Bit (1) 1 = Enables EMCLK clocking	0

NOTE:

1. The complement of "Enable" action means that each clock (EMCLK or IMCLK) becomes stable after you enable each clock. The status register (CM_SR) reflects the result of this execution by hardware.
2. Before you enable the clock monitor function, you should check IMCLK is enabled or disabled. If IMCLK is disabled, you should enable IMCLK before enabling the clock monitor function. Because the clock monitor circuit uses IMCLK as a reference clock to monitor the external main clock. If you enable a clock monitor without enabling IMCLK, the command error generates.
3. The CM_CSR register is Write-only. The opposite control should use CM_CCR.
   All defined control bits have no-effect on writing "0".

5.9.1.4 CM_CCR

- Base Address: 0x4002_0000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD								EMCM	EMCMRST	RSVD										IDLEW	RSVD	PCLK	STCLK	PLL	RSVD	FWAKE	RSVD			IMCLK	EMCLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	W	W	R	R	R	R	R	R	R	R	R	R	W	R	W	W	W	R	W	R	R	R	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	R	Reserved	0
EMCM	[23]	W	External Main Clock Monitor Function Disable Control Bit 1 = Disables the Main Clock Monitor function	0
EMCMRST	[22]	W	External Main Clock Monitor Reset Disable Control Bit 1 = Disables reset by Clock Monitor. When it detects a Clock fail, it reports a Clock fail flag.	0
RSVD	[21:12]	R	Reserved	0
IDLEW	[11]	W	IDLEW Control Bit 1 = Clears the IDLEW	0
RSVD	[10]	R	Reserved	0
PCLK	[9]	W	PCLK Disable Control Bit in IDLE Mode 1 = Disconnects the PCLK supplied to peripherals in an IDLE Mode	0
STCLK	[8]	W	STCLK Disable Control in IDLE Mode 1 = Disconnects the STCLK supplied to a SYSTICK timer of Cortex-M0 in IDLE mode	0
PLL	[7]	W	PLL OFF Control Bit 1 = Disables PLL	0
RSVD	[6]	R	Reserved	0
FWAKE	[5]	W	Fast Wake-Up Disable Control Field 1 = Disables the FWAKE (after Wake-Up from the STOP mode, the SYSCLK source is the same before entry of the STOP mode). If a microcontroller enters the STOP mode after disabling a Fast Wake-Up, the SYSCLK is the same before entry of STOP mode.	0
RSVD	[4:2]	R	Reserved	0

Name	Bit	Type	Description	Reset Value
IMCLK	[1]	W	Internal Main Clock Disable Control Bit 1 = Disables the IMCLK clocking If EMCM bit in the CM_SR register is "1", there is no effect (not even disable) even if this bit is set to "1".	0
EMCLK	[0]	W	External Main Clock Disable Control Bit 1 = Disables the EMCLK clocking	0

NOTE:

1. The CM_CCR register is Write-only. The opposite control should use the CM_CSR register. All defined control bits have no-effect on writing "0".
2. For each bit, you should write "1". Each clock is a source of any block to operate. Before disabling any clock, you should check the usage of the clock by any block. For example, if the WDTCLK uses the IMCLK as a timer clock source, you should stop the operation of the WDTCLK before disabling the IMCLK. If the operating clock of watchdog timer stops while the watchdog timer runs, then the CMDERR bit of the CM_SR register is set to "1".

Caution: If you don't the external main-oscillator when pin7 and pin8 are defined XOUT and XIN function, you should tie XIN to ground. XOUT should be opened.

5.9.1.5 CM_PCSR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD	IOCLK	IFCCLK	RSVD				SPI0CLK	COMPCLK	ADCCLK	RSVD					USART0CLK	RSVD					TC2CLK	TC1CLK	TC0CLK	IMCCLK	PPDCLK	PWM3CLK	PWM2CLK	PWM1CLK	PWM0CLK	WDTCLK	OPACLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	W	W	R	R	R	R	W	W	W	R	R	R	R	R	W	R	R	R	R	R	W	W	W	W	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
IPxCLK	[x]	W	PCLK (APB Clock) Gating Control 0 = No effect 1 = Enables each peripheral clock	0

NOTE: The IOCLK includes the I/O Configuration (IOCONF) and the GPIO pin.

5.9.1.6 CM_PCCR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD	IOCLK	IFCCLK	RSVD				SPI0CLK	COMPCLK	ADCCLK	RSVD					USART0CLK	RSVD					TC2CLK	TC1CLK	TC0CLK	IMCCLK	PPDCLK	PWM3CLK	PWM2CLK	PWM1CLK	PWM0CLK	WDTCLK	OPACLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	W	W	R	R	R	R	W	W	W	R	R	R	R	R	W	R	R	R	R	R	W	W	W	W	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
IPxCLK	[x]	W	PCLK (APB Clock) Gating Control 0 = No effect 1 = Disables each peripheral clock	0

NOTE: The IOCLK includes the IOCONF and the GPIO.

5.9.1.7 CM_PCKSR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD	IOCLK	IFCCLK	RSVD				SPI0CLK	COMPCLK	ADCCLK	RSVD					USART0CLK	RSVD					TC2CLK	TC1CLK	TC0CLK	IMCCLK	PPDCLK	PWM3CLK	PWM2CLK	PWM1CLK	PWM0CLK	WDTCLK	OPACLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31]	R	Reserved	0
IOCLK IFCCLK	[30] [29]	R	Peripheral Clock Status 0 = Disables each peripheral clock (disconnected). It is impossible to control (Write) SFR register of a target peripheral. 1 = Enables each peripheral clock (connected). It is possible to control (Write) SFR register of a target peripheral.	0 0
RSVD	[28:25]	R	Reserved	0
SPI0CLK COMPCLK ADCCLK	[24] [23] [22]	R	Peripheral Clock Status 0 = Disables each peripheral clock (disconnected). It is impossible to control (Write) SFR register of a target peripheral. 1 = Enables each peripheral clock (connected). It is possible to control (Write) SFR register of a target peripheral.	0 0 0
RSVD	[21:17]	R	Reserved	0
USART0CLK	[16]	R	Peripheral Clock Status 0 = Disables each peripheral clock (disconnected). It is impossible to control (Write) SFR register of a target peripheral. 1 = Enables each peripheral clock (connected). It is possible to control (Write) SFR register of a target peripheral.	0
RSVD	[15:11]	R	Reserved	0
TC2CLK TC1CLK TC0CLK IMCCLK	[10] [9] [8] [7]	R	Peripheral Clock Status 0 = Disables each peripheral clock (disconnected). It is impossible to control (Write) SFR register of a target peripheral.	0 0 0 0

Name	Bit	Type	Description	Reset Value
PPDCLK	[6]		1 = Enables each peripheral clock (connected). It is possible to control (Write) SFR register of a target peripheral. NOTE: After reset, WDTCLK is enabled by default and WDT (watchdog timer) counts automatically.	0
PWM3CLK	[5]			0
PWM2CLK	[4]			0
PWM1CLK	[3]			0
PWM0CLK	[2]			0
WDTCLK	[1]			1
OPACLK	[0]			0

5.9.1.8 CM_MR0

- Base Address: 0x4002_0000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0828

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																				LVDPD	STCLKEN	RXEV	RSVD	LVDINTEN	LVDIL			LVDRSTEN	LVDRL		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:12]	R	Reserved	0
LVDPD	[11]	RW	LVD Power Down Control Bit 0 = LVD powers down 1 = LVD powers up When you set to 0, then you cannot use the LVD interrupt and the LVD reset function. To reduce power, you can configure this bit. To use LVD Reset (LVDRST) or/and LVD Interrupt (LVDINT), you should set the LVDPD bit. On reset, LVD is in the power-up state.	1
STCLKEN	[10]	RW	SYSTIC Timer Clock Enable/Disable Control Bit 0 = Disables 1 = Enables	0
RXEV	[9]	RW	RXEV(Receive Event) Enable/Disable Control Bit 0 = Disables RXEV 1 = Enables RXEV This bit causes the Event register defined processor (Cortex-M0) to be set. This causes WFE instruction to complete. It also awakens the processor if it is in sleep mode by the result of encountering a WFE instruction when the Event Register is clear.	0
RSVD	[8]	R	Reserved	0
LVDINTEN	[7]	RW	LVD Interrupt Enable/Disable Control Bit 0 = Disables LVD Interrupt 1 = Enables LVD Interrupt LVD detects the configured voltage level. At this time, interrupt may or may not occur with LVDINTEN value. LVD detects the target voltage and LVDIL[2:0] field decides the target voltage.	0
LVDIL	[6:4]	RW	LVD Interrupt Threshold Level (refer to the Electrical Data	010

Name	Bit	Type	Description	Reset Value
			chapter) 000 = LVD_LEVEL5 (Typical 2.4 V) 001 = LVD_LEVEL4 (Typical 2.6 V) 010 = LVD_LEVEL3 (Typical 2.8 V) 011 = LVD_LEVEL2 (Typical 3.8 V) 100 = LVD_LEVEL1 (Typical 4.3 V) Others = You should not set other values. NOTE: If the LVDRL and LVDIL field have the same value and LVD detects the voltage, then LVD Reset occurs. Before LVD detects the voltage, LVRSTEN bit should be set to "1". If LVDIL[2:0] and LVDRl[2:0] have the same value, LVD reset occurs when LVD detects the configured level.	
LVDRSTEN	[3]	RW	LVD Reset Enable/Disable Control Bit 0 = Disables 1 = Enables This bit controls the LVD Reset. To use the LVD Reset, you should set this bit to "1" LVDRL fields decide the target voltage to detect.	1
LVDRL	[2:0]	RW	LVD Reset Level (refer to Electrical Data chapter) 000 = LVD_LEVEL5 (Typical 2.4 V) - Reset Value 001 = LVD_LEVEL4 (Typical 2.6 V) 010 = LVD_LEVEL3 (Typical 2.8 V) 011 = LVD_LEVEL2 (Typical 3.8 V) 100 = LVD_LEVEL1 (Typical 4.3 V) Others = You should not set other values.	000

5.9.1.9 CM_MR1

- Base Address: 0x4002_0000
- Address = Base Address + 0x002C, Reset Value = 0x0000_001X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																										WDTCLK		RSVD		SYSCLK	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	X
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R	R	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	R	Reserved	0
WDTCLK	[5:4]	RW	Watchdog Timer Clock Source Selection Bits 00 = EMCLK 01 = IMCLK 10 = PLLCLK 11 = Disconnects WDTCLK NOTE: 1. If you try to change the clock source when the target clock source to exchange is in disable state, then the value does not change and command error occurs. When the command error occurs, the CMDERR bit becomes "1". 2. When switched clock becomes stable after you switch on the clock source, the WDTCLK interrupt occurs. You can use WDTCLK interrupt (event) to check that WDTCLK switching completes or not. If you change the value of the WDTCLK to 11'b (disconnect WDTCLK), then the WDTCLK interrupt does not occur. 3. When you do not run the WDTCLK and want to reduce the power, it is recommended to use the "Disconnect WDTCLK".	01
RSVD	[3:2]	R	Reserved	0
SYSCLK	[1:0]	RW	SYSCLK Selection Select one among different clock for SYSCLK 00 = EMCLK 01 = IMCLK 10 = PLLCLK 11 = Prohibits NOTE: 1. If you modify the SYSCLK, it changes your system clock. Be cautious while changing SYSCLK. If you try to	0X

Name	Bit	Type	Description	Reset Value
			write 11'b to SYSCLK, then the SYSCLK does not change and the CMDERR bit is "1". 2. When the clock change completes, the STABLE bit in CM_SR register becomes "1". 3. The SYSCLK[0] reset value depends on the POCCS[0] in smart option configuration status register (refer to the Program Flash chapter).	

5.9.1.10 CM_IMSCR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD													CMDERR	RSVD	LVDINT	EMCKFAIL	EMCKFAIL_END	RSVD						PLL	RSVD		STABLE	RSVD		IMCLK	EMCLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	R	R	R	R	R	RW	R	R	RW	RW	R	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:19]	R	Reserved	0
CMDERR	[18]	RW	Command error interrupt	0
RSVD	[17]	R	Reserved	0
LVDINT	[16]	RW	Interrupt level detect of LVD interrupt	0
EMCKFAIL	[15]	RW	External main clock failure interrupt	0
EMCKFAIL_END	[14]	RW	External main clock failure end interrupt	0
RSVD	[13:8]	R	Reserved	0
PLL	[7]	RW	PLL stable interrupt	0
RSVD	[6:5]	R	Reserved	0
STABLE	[4]	RW	SYSCLK clock switching stable interrupt	0
RSVD	[3:2]	R	Reserved	0
IMCLK	[1]	RW	Internal main clock stable interrupt	0
EMCLK	[0]	RW	External main clock stable interrupt 0 = Mask the stable interrupt (Disables an interrupt) 1 = Unmask the stable interrupt (Enables an interrupt)	0
Interrupt Source X	[X]	RW	Interrupt mask set/clear control Bit 0 = Disables Source X interrupt 1 = Enables Source X interrupt	–

NOTE: On a Read, CM_IMSCR register provides the current value of the mask on the relevant interrupt.
A Write of "1" to the particular bit clears the mask.
A Write of "0" sets the corresponding mask.

5.9.1.11 CM_RISR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0034, Reset Value = 0x0000_001B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD													CMDERR	LVDRS	LVDINT	EMCKFAIL	EMCKFAIL_END	RSVD						PLL	RSVD		STABLE	RSVD		IMCLK	EMCLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:19]	R	Reserved	0
CMDERR	[18]	R	Command Error Interrupt Gives the raw interrupt state (prior to masking) of the CMDERR interrupt.	0
LVDRS	[17]	R	Reset Level Detect Status This bit is only to show the status. It does not generate the interrupt when it is set to "1". This bit is set to "1" when the LVD detect the defined reset voltage level regardless of LVDRSTEN bit status. This bit has the same value with LVDRS bit of CM_SR register. Write "1" into the LVDRS bit of the CM_ICR register to clear this bit.	0
LVDINT	[16]	R	Interrupt Level Detect of LVD Interrupt Gives the raw interrupt state (prior to masking) of the LVDINT interrupt. This bit becomes "1" when LVDINTEN bit in CM_MR register is set to "1".	0
EMCKFAIL	[15]	R	External Main Clock Failure Interrupt Gives the raw interrupt state (prior to masking) of the EMCKFAIL interrupt.	0
EMCKFAIL _END	[14]	R	External Main Clock Failure End Interrupt Gives the raw interrupt state (prior to masking) of the EMCKFAIL_END interrupt.	0
RSVD	[13:8]	R	Reserved	0
PLL	[7]	R	PLL Stable Interrupt Gives the raw interrupt state (prior to masking) of the PLL interrupt.	0
RSVD	[6:5]	R	Reserved	0

Name	Bit	Type	Description	Reset Value
STABLE	[4]	R	SYSCLK Clock Switching Stable Interrupt Gives the raw interrupt state (prior to masking) of the STABLE interrupt.	1
RSVD	[3:2]	R	Reserved	10
IMCLK	[1]	R	Internal Main Clock Stable Interrupt Gives the raw interrupt state (prior to masking) of the IMCLK interrupt.	1
EMCLK	[0]	R	External Main Clock Stable Interrupt Gives the raw interrupt state (prior to masking) of the EMCLK interrupt	1

NOTE:

1. On a Read, CM_RISR register gives the current raw status value of the corresponding interrupt prior to masking.
   A Write has no effect.
2. After reset release, some bits (STABLE/ISCLK/ESCLK/IMCLK/EMCLK) are set to "1".
   Therefore, we recommend clearing those bits above to use the CM_ICR register after the Reset release.

5.9.1.12 CM_MISR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0038, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD													CMDERR	RSVD	LVDINT	EMCKFAIL	EMCKFAIL_END	RSVD						PLL	RSVD		STABLE	RSVD		IMCLK	EMCLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:19]	R	Reserved	0
CMDERR	[18]	R	Command Error Interrupt Gives the unmasked interrupt state (after unmasking) of the CMDERR interrupt 0 = Each interrupt does not occur 1 = Each interrupt occurs	0
RSVD	[17]	R	Reserved	0
LVDINT	[16]	R	Interrupt Level Detect of LVD Interrupt Gives the unmasked interrupt state (after unmasking) of the LVDINT interrupt 0 = Each interrupt does not occur 1 = Each interrupt occurs	0
EMCKFAIL	[15]	R	External Main Clock Failure Interrupt Gives the unmasked interrupt state (after unmasking) of the EMCKFAIL interrupt 0 = Each interrupt does not occur 1 = Each interrupt occurs	0
EMCKFAIL _END	[14]	R	External Main Clock Failure End Interrupt Gives the unmasked interrupt state (after unmasking) of the EMCKFAIL_END interrupt 0 = Each interrupt does not occur 1 = Each interrupt occurs	0
RSVD	[13:8]	R	Reserved	0
PLL	[7]	R	PLL Stable Interrupt Gives the unmasked interrupt state (after unmasking) of the PLL interrupt 0 = Each interrupt does not occur 1 = Each interrupt occurs	0

Name	Bit	Type	Description	Reset Value
RSVD	[6:5]	R	Reserved	0
STABLE	[4]	R	SYSCLK Clock Switching Stable Interrupt Gives the unmasked interrupt state (after unmasking) of the STABLE interrupt 0 = Each interrupt does not occur 1 = Each interrupt occurs	0
RSVD	[3:2]	R	Reserved	0
IMCLK	[1]	R	Internal Main Clock Stable Interrupt Gives the unmasked interrupt state (after unmasking) of the IMCLK interrupt 0 = Each interrupt does not occur 1 = Each interrupt occurs	0
EMCLK	[0]	R	External Main Clock Stable Interrupt Gives the unmasked interrupt state (after unmasking) of the EMCLK interrupt 0 = Each interrupt does not occur 1 = Each interrupt occurs	0

NOTE: On a Read, CM_MISR register gives the current masked status value of the corresponding interrupt. A Write has no effect.

5.9.1.13 CM_ICR

- Base Address: 0x4002_0000
- Address = Base Address + 0x003C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD													CMDERR	LVDRS	LVDINT	EMCKFAIL	EMCKFAIL_END	RSVD						PLL	RSVD		STABLE	RSVD		IMCLK	EMCLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W	R	R	R	R	R	R	W	R	R	W	R	R	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:19]	R	Reserved	0
CMDERR	[18]	W	Command Error Interrupt 1 = Clears the CMDERR interrupt	0
LVDRS	[17]	W	Reset Level Detect of LVD 1 = Clears the LVDRS flag	0
LVDINT	[16]	W	Interrupt Level Detect of LVD Interrupt 1 = Clears the LVDINT Interrupt	0
EMCKFAIL	[15]	W	External Main Clock Failure Interrupt 1 = Clears the EMCKFAIL interrupt	0
EMCKFAIL _END	[14]	W	External Main Clock Failure End Interrupt 1 = Clears the EMCKFAIL_END interrupt	0
RSVD	[13:8]	R	Reserved	0
PLL	[7]	W	PLL Stable Interrupt 1 = Clears the PLL stable interrupt	0
RSVD	[6:5]	R	Reserved	0
STABLE	[4]	W	SYSCLK Clock Switching Stable Interrupt 1 = Clears the Switching stable interrupt	0
RSVD	[3:2]	R	Reserved	0
IMCLK	[1]	W	Internal Main Clock Stable Interrupt 1 = Clears the IMCLK stable interrupt	0
EMCLK	[0]	W	External Main Clock Stable Interrupt 1 = Clears the EMCLK stable interrupt	0

NOTE: On a Write of "1", it clears the corresponding interrupt in the CM_RISR register. A Write of "0" has no effect.

5.9.1.14 CM_SR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0040, Reset Value = 0xYY80_001B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SYSRSTS	EMCMRSTS	RSVD	PORRSTS	WDTRSTS	LVDRSTS	NRSTS	SWRSTS	EMCM	EMCMRST	RSVD			CMDERR	LVDRS	LVDINT	EMCKFAIL	EMCKFAIL_END	RSVD		IDLEW	RSVD	PCLK	STCLK	PLL	RSVD	FWAKE	STABLE	WDTCLKS	RSVD	IMCLK	EMCLK
Y	Y	Y	Y	Y	Y	Y	Y	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	Y
R W	R W	R W	R W	R W	R W	R W	R W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
SYSRSTS	[31]	RW	SysReset Status SYSRST is System (Chip) Reset by CPU request. 0 = CPU reset request does not occur. 1 = CPU request causes the last reset This bit is cleared when NRSTS, PORRSTS or LVDRSTS occurs or you write this bit to "1".	Y
EMCMRSTS	[30]	RW	External Main Clock Monitor Reset Status System (Chip) reset from external main clock monitor fail 0 = External Main-Clock Monitor fail reset does not occur. 1 = External main-clock monitor fail reset causes the last reset. When nReset, POR reset, or LVR reset occur, it clears this bit by hardware. To clear by software, writes this bit to "1".	Y
RSVD	[29]	R	Reserved	0
PORRSTS	[28]	RW	Power-On Reset Status PORRST is System (Chip) Reset by Power-On Reset circuit. 0 = Power-on reset does not occur. 1 = Power-on reset occurs To clear this bit, write to 1.	Y
WDTRSTS	[27]	RW	Watchdog Timer Reset Status WDTRST is System (Chip) Reset by watchdog timer. 0 = Watchdog timer reset is not occurred 1 = The last reset is caused by Watchdog timer reset When nReset, POR reset, or LVR reset occur, it clears this bit by hardware. To clear by software, writes this bit to "1".	Y
LVDRSTS	[26]	RW	LVD Reset Status LVDRST is System (Chip) Reset by LVD.	Y

Name	Bit	Type	Description	Reset Value
			For LVD reset condition, you should enable the LVDRST bit in CM_MR register. 0 = LVD reset is not occurred 1 = LVD reset is occurred To clear this bit, write to 1.	
NRSTS	[25]	RW	nRESET Status NRST is System (Chip) Reset by external reset pin. This bit instructs the reset source that generates reset signal pin. 0 = External pin reset is not occurred 1 = External pin reset (nRESET) is occurred To clear this bit, write to 1.	Y
SWRSTS	[24]	RW	Software Reset Status System (Chip) reset from the software reset (SWRST in CM_SRR register) 0 = Software reset is not occurred 1 = The last reset is caused by software reset When nReset, POR reset, or LVR reset occur, it clears this bit by hardware. To clear by software, writes this bit to "1".	Y
EMCM	[23]	R	External Main Clock Monitor Fail Function Enable/Disable Status 0 = Disables EMCM 1 = Enables EMCM	1
EMCMRST	[22]	R	External Main Clock Monitor Reset Function Enable/Disable Status 0 = Disables reset function by clock fail mode 1 = Enables reset function by clock fail mode	0
RSVD	[21:19]	R	Reserved	0
CMDERR	[18]	R	Command Error Status 0 = Does not occur CMDERR event 1 = Occurs CMDERR event	0
LVDRS	[17]	R	LVD Reset Level Detect Status (same as LVDRS in CM_RISR register) 0 = Does not detect LVD Reset voltage level 1 = Detects LVD Reset voltage level	0
LVDINT	[16]	R	LVD Interrupt Status 0 = Does not detect LVD Interrupt voltage level 1 = Detects LVD Interrupt voltage level	0
EMCKFAIL	[15]	R	External Main Clock Fail Status 0 = The External Main Oscillator failure does not occur 1 = The External Main Oscillator failure has been detected It clears this bit when EMCKFAIL_END or reset occurs. CM_ICR register does not clear this bit.	0
EMCKFAIL	[14]	R	External Main Clock Failure End Status	0

Name	Bit	Type	Description	Reset Value
_END			0 = No end of the External Main Oscillator failure is detected 1 = At least one end of the External Main Oscillator Clock failure is detected To clear this bit, write to "1' into the EMCKFAIL_END bit of CM_ICR register.	
RSVD	[13:12]	R	Reserved	0
IDLEW	[11]	R	IDLE Control Status 0 = Clears IDLEW 1 = Sets IDLEW NOTE: IDLEW bit in the CM_CSR and the CM_CCR registers controls this bit. This bit shows the status of the "SEVONPEND" bit in Cortex^{TM}-M0. When this bit is 1, it can wake-up from WFE if a new interrupt is pending, regardless of whether the interrupt has priority higher than the current level. SEVONPEND means "Send Event on Pending".	0
RSVD	[10]	R	Reserved	0
PCLK	[9]	R	PCLK Control Status in IDLE Mode 0 = Disables(disconnect) PCLK in IDLE mode 1 = Enables(connect) PCLK in IDLE mode	0
STCLK	[8]	R	STCLK Control Status in IDLE Mode 0 = Disables STCLK in IDLE mode 1 = Enables STCLK in IDLE mode	0
PLL	[7]	R	PLL Stable Interrupt 0 = Disables PLL 1 = Enables and stabilizes PLL	0
RSVD	[6]	R	Reserved	0
FWAKE	[5]	R	Fast Wake-Up Control (Enable/Disable) Status 0 = Disables FWAKE 1 = Enables FWAKE (After Wake-Up, SYSCLK becomes IMCLK)	0
STABLE	[4]	R	SYSCLK Clock Stable Status Bit 1 = Clock source switching for SYSCLK completes and status becomes stable.	1
WDTCLKS	[3]	R	WDTCLK Status Bit 0 = WDTCLK inactive status 1 = WDTCLK active status	0'b
RSVD	[2]	R	Reserved	0
IMCLK	[1]	R	Internal Main Clock Status 0 = Disables IMCLK 1 = Enables and stabilizes IMCLK	1
EMCLK	[0]	R	External Main Clock Status	Y

Name	Bit	Type	Description	Reset Value
			0 = Disables EMCLK 1 = Enables and stabilizes EMCLK	

CM_SR [31:24] bits are Read and Write. After checking the status of each reset, you should clear each bit of CM_SR[31:24] by writing "1".

NOTE: Command Error Event.

Command error occurs at these events:

1. If you disable the IMCLK when the Clock Monitor function is enabled, then command error occurs. If you want to disable the IMCLK when the Clock Monitor function is enabled, you should disable the Clock Monitor function.
2. If you enable the clock monitor function when the IMCLK is disabled, then the command error occurs. If you want to enable the Clock Monitor function, then you should enable the IMCLK.

Caution: If you don't the external main-oscillator when pin7 and pin8 are defined XOUT and XIN function, you should tie XIN to ground. XOUT should be opened.

5.9.1.15 CM_SCDR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0044, Reset Value = 0x0000_0007

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDIVKEY																RSVD													SDIV		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Name	Bit	Type	Description	Reset Value
SDIVKEY	[31:16]	W	Key for Write Access into the CM_SCDR Any Write in CM_SCDR register bits is only effective if the SDIVKEY is equal to 0xACDC.	0
RSVD	[15:3]	R	Reserved	0
SDIV	[2:0]	RW	SYSCLK Divider This field selects the division ratio for SYSCLK.	0x7

Table 5-12 SYSCLK Divider Value

SYSCLK	Description	4 MHz	8 MHz	12 MHz	16 MHz	40 MHz
SDIV[2:0]	Division Ratio					
000	1	4	8	12	16	40
001	2	2	4	6	8	20
010	3	1.333333	2.666667	4	5.333333	13.33333
011	4	1	2	3	4	10
100	5	0.8	1.6	2.4	3.2	8
101	6	0.666667	1.333333	2	2.666667	6.666667
110	7	0.571429	1.142857	1.714286	2.285714	5.714286
111	8	0.5	1	1.5	2	5

5.9.1.16 CM_PCDR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0048, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PDIVKEY																RSVD												PDIV			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PDIVKEY	[31:16]	W	PCDR Register Key Key for Write Access into the CM_PCDR Any Write in CM_PCDR register bits is only effective if the PDIVKEY is equal to 0xA3C5.	0
RSVD	[15:4]	R	Reserved	0
PDIV	[3:0]	RW	PCLK Divider This field selects the division ratio for PCLK.	0x0

PDIV[3:0]	Division Ratio	Frequency
0000	1	SYSCLK
0001	2	SYSCLK/2
001X	4	SYSCLK/4
01XX	8	SYSCLK/8
1XXX	16	SYSCLK/16

5.9.1.17 CM_PSTR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0058, Reset Value = 0x0000_0154

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLLSKEY																RSVD					PST										
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PLLSKEY	[31:16]	W	Key for Write access into the CM_PSTR register Any Write in the CM_PSTR register bits will only be effective if the PLLSKEY field is equal to 0x59C1.	0
RSVD	[15:11]	R	Reserved	0
PST	[10:0]	RW	PLL Stabilization Time PST register value = (PLL stabilization time/(EMCLK period × 256))	0x154

NOTE: When you disable the PLL, then the Stabilization Time Register will have Write access. If you enable the PLL, then the register will have Read access.

5.9.1.18 CM_PDPR

- Base Address: 0x4002_0000
- Address = Base Address + 0x005C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLLKEY								LFPASS	RSVD					PLLPOST		RSVD		PLLPRE						PLLMUL							
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
W	W	W	W	W	W	W	W	R W	R	R	R	R	R	R W	R W	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PLLKEY	[31:24]	W	PLL Parameter Control Register Key Value Key for Write access into the CM_PSTR register Any Write in CM_PDPR register bits is only effective if the PLLKEY is equal to 0xC1.	0
LFPASS	[23]	RW	Low Frequency Pass Control Bit 0 = F_{IN} is the same or greater than 4 MHz. ($F_{IN} \geq 4$ MHz) 1 = F_{IN} is less than 4 MHz. ($F_{IN} < 4$ MHz) The pre-divider value, P[5:0], does not effect to output frequency for PLL.	0
RSVD	[22:18]	RW	Reserved	0
PLLPOST	[17:16]	RW	PLL Post-Scaler Value S[1:0] = 0x0 to 0x3 (0 to 3)	00
RSVD	[15:14]	R	Reserved	0
PLLPRE	[13:8]	RW	PLL Pre-Divider Value P[5:0] = 0x01 to 0x3F (1 to 63)	0x00
PLLMUL	[7:0]	RW	PLL Multiplier Value M[7:0] = 0x10 to 0xFF (16 to 255)	0x00

NOTE:

1. If you disable the PLL, then the register will have the Write access. If you enable the PLL, then the register will have Read access.
2. The FIN is input frequency of the PLL and becomes the EMCLK.
3. m = M[7:0], p = P[5:0], s = S[1:0]
4. At the LFPASS = 0, FOUT = $((m + 8) \times \text{FIN})/((p + 2) \times 2^{s})$
5. At the LFPASS = 1, FOUT = $((m + 8) \times \text{FIN})/2^{s}$
6. 18-Bit is reserved. That has no-effect regardless of the value, 0 or 1.
7. F_{OUT} = PLL output frequency, F_{IN} = PLL input frequency = EMCLK

Table 5-13 PMS Value Table

FIN (MHz)	1				2			
FOUT (MHz)	P	M	S	LFPASS	P	M	S	LFPASS
12	0	40	2	1	0	16	2	1
16	0	24	1	1	0	8	1	1
20	0	32	1	1	0	12	1	1
24	0	40	1	1	0	16	1	1
28	0	20	0	1	0	6	0	1
32	0	24	0	1	0	8	0	1
36	0	28	0	1	0	10	0	1
40	0	32	0	1	0	12	0	1

FIN (MHz)	4				6			
FOUT (MHz)	P	M	S	LFPASS	P	M	S	LFPASS
12	2	40	2	0	4	40	2	0
16	2	24	1	0	4	24	1	0
20	2	32	1	0	4	32	1	0
24	2	40	1	0	4	40	1	0
28	2	20	0	0	4	20	0	0
32	2	24	0	0	4	24	0	0
36	2	28	0	0	4	28	0	0
40	2	32	0	0	4	32	0	0

FIN (MHz)	8				10			
FOUT (MHz)	P	M	S	LFPASS	P	M	S	LFPASS
12	6	40	2	0	8	40	2	0
16	6	24	1	0	8	24	1	0
20	6	32	1	0	8	32	1	0
24	6	40	1	0	8	40	1	0
28	6	20	0	0	8	20	0	0
32	6	24	0	0	8	24	0	0
36	6	28	0	0	8	28	0	0
40	6	32	0	0	8	32	0	0

5.9.1.19 CM_BTCDR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0070, Reset Value = 0x0000_000X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BTCDKEY																RSVD												BTCDIV			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X	X	X	X
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
BTCDKEY	[31:16]	W	Key for Write Access into the CM_BTCDR Register Any write in CM_BTCDR register bits is only effective if the BTCDKEY is equal to 0x3569.	0
RSVD	[15:4]	R	Reserved	0
BTCDIV	[3:0]	RW	Basic Timer Clock Divider Control Field Defines the number of Basic Timer clock divider value.	XXXX'b

Firstly, in bear chip status, all bits in the SO_CSR (refer Program Flash Controller (PFC) section) are "1". That means the BT clock is IMCLK (40 MHz) and BT Clock Divider is 4096. At reset, BT counting time is 26.214 ms, 256 count with 40 MHz/4096.

The clock manager block runs BT when chip resets or wakes up from stop mode. BT counts the defined time to stabilize the internal system (logic). BT counting clock source is SYSCLK. SYSCLK can be IMCLK, EMCLK, or PLLCLK. BTCDIV decides the clock frequency of BT count with the clock source. At reset time, smart option value decides SYSCLK and BT clock divider. In wake-up time, the condition before entering the stop mode decides SYSCLK and BTCDIV value in CM_BTCDR register decide BT clock divider.

When 8th bit of BT Timer is set to "1", it releases system reset or Wake-Up signal.

BTCDIV[3:0]	Divider Ratio	BTCDIV[3:0]	Divider Ratio
1111	4096	1000	32
1110	2048	0111	16
1101	1024	0110	8
1100	512	0101	4
1011	256	0100	2
1010	128	0011	1
1001	64	Others	Not used

5.9.1.20 CM_BTR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0074, Reset Value = 0x0000_0100

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
BTCV	[15:0]	RW	Basic Timer Count Value	0x0100

5.9.1.21 CM_EECR0

- Base Address: 0x4002_0000
- Address = Base Address + 0x0078, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EEEN3	EDGE3		EESRC3					EEEN2	EDGE2		EESRC2					EEEN1	EDGE1		EESRC1					EEEN0	EDGE0		EESRC0				
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
EEEN3 EEEN2 EEEN1 EEEN0	[31] [23] [15] [7]	RW	External Event Enable/Disable Control Bit 0 = Disables the target external event source. 1 = Enable the target external event source. "x" means each number from 0 to 3.	0
EDGE3 EDGE2 EDGE1 EDGE0	[30:29] [22:21] [14:13] [6:5]	RW	Edge Type Selection Bit 00 = Selects rising edge trigger for external event 01 = Selects falling edge trigger for external event 10 = Selects both (Rising/Falling) edge trigger for external event 11 = Invalid value	0
EESRC3 EESRC2 EESRC1 EESRC0	[28:24] [20:16] [12:8] [4:0]	RW	External Event Source x External Event Source Selection Field Refer to *Table 5-14*	0x00

NOTE: You should do any configuration in the register before the entry of STOP/IDLE mode.
You should configure either of the corresponding edge or level to enable an event or interrupt.

5.9.1.22 CM_EECR1

- Base Address: 0x4002_0000
- Address = Base Address + 0x007C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EEEN7	EDGE7		EESRC7					EEEN6	EDGE6		EESRC6					EEEN5	EDGE5		EESRC5					EEEN4	EDGE4		EESRC4				
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
EEEN7 EEEN6 EEEN5 EEEN4	[31] [23] [15] [7]	RW	External Event Enable/Disable Control Bit 0 = Disables the target external event source. 1 = Enable the target external event source. "x" means each number from 4 to 7.	0
EDGE7 EDGE6 EDGE5 EDGE4	[30:29] [22:21] [14:13] [6:5]	RW	Edge Type Selection Bit 00 = Selects rising edge trigger for external event 01 = Selects falling edge trigger for external event 10 = Selects both (Rising/Falling) edge trigger for external event 11 = Invalid value	0
EESRC7 EESRC6 EESRC5 EESRC4	[28:24] [20:16] [12:8] [4:0]	RW	External Event Source x External Event Source Selection Field Refer to *Table 5-14*	0x00

NOTE: You should do any configuration in the register before the entry of STOP/IDLE mode.
You should configure either of the corresponding edge or level to enable an event or interrupt.

Table 5-14 The External Event Sources and Pin Assignment

EESRCx[4:0]	External Event Source	Pin Information
00000	EXI0	P0.0/EXI0/TPWM0/OP0_N
00001	EXI1	P0.1/EXI1/TCAP0/OP0_P
00010	EXI2	P0.3/AIN2/PWM0/EXI2
00011	EXI3	P0.4/AIN3/PWM1/EXI3
00100	EXI4	P0.5/AIN4/PWM2/EXI4
00101	EXI5	P0.6/AIN5/PWM3/EXI5
00110	EXI6	P0.10/AIN9/PWM3/EXI6
00111	EXI7	P0.12/PWMU2/TPWM2/EXI7
01000	EXI8	P0.13/PWMU1/TCAP2/EXI8
01001	EXI9	P0.14/PWMU0/TCLK2/EXI9
01010	EXI10	P0.15/PWMD2/COP0/EXI10
01011	EXI11	P0.16/PWMD1/COP1/EXI11
01100	EXI12	P0.17/PWMD0/COP2/EXI12
01101	EXI13	P0.18/PWMOFF/COP3/EXI13
01110	EXI14	P0.19/PHASEA/USARTRX0/EXI14
01111	EXI15	P0.20/PHASEB/USARTTX0/EXI15
10000	EXI16	P0.21/PHASEZ/USARTCLK0/EXI16
10001	EXI17	P0.22/COMP0_N/MISO0/EXI17
10010	EXI18	P0.26/COMP2_N/USARTRX0/EXI18
10011	EXI19	P0.27/COMP2_P/USARTTX0/EXI19
10100	EXI20	P0.28/COMP3_N/EXI20/SWDIO
10101	EXI21	P0.29/COMP3_P/EXI21/SWDCLK
10110	EXI22	P0.30/ – /USARTTX0/EXI22
10111	EXI23	P0.31/EXI23/USARTRX0/PWM0
11000	SPI0	P0.22/COMP0_N/MISO0/EXI17@Master Mode P0.23/COMP0_P/MOSI0/PWM1@Slave Mode
11001	USARTRX0	P0.19/PHASEA/USARTRX0/EXI14 P0.26/COMP2_N/USARTRX0/EXI18
11010 to 11111	RSVD	Reserved

5.9.1.23 CM_EEIMSCR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0088, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																								EE7	EE6	EE5	EE4	EE3	EE2	EE1	EE0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	R	Reserved	0
EE7 EE6 EE5 EE4 EE3 EE2 EE1 EE0	[7] [6] [5] [4] [3] [2] [1] [0]	RW	External Event Interrupt Mask Set/Clear Bit 0 = Mask each interrupt (Disables an interrupt) 1 = Unmask each interrupt (Enables an interrupt)	0

5.9.1.24 CM_EERISR

- Base Address: 0x4002_0000
- Address = Base Address + 0x008C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																								EE7	EE6	EE5	EE4	EE3	EE2	EE1	EE0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	R	Reserved	0
EE7 EE6 EE5 EE4 EE3 EE2 EE1 EE0	[7] [6] [5] [4] [3] [2] [1] [0]	R	External Event Raw Interrupt Status Bit 0 = Each interrupt does not occur (Prior to unmasking) 1 = Each interrupt occurs (Prior to unmasking)	0

NOTE: On a Read, the CM_WRISR register gives the current raw status value of the corresponding interrupt prior to unmasking. A Write has no effect.

5.9.1.25 CM_EEMISR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0090, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																								EE7	EE6	EE5	EE4	EE3	EE2	EE1	EE0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	R	Reserved	0
EE7 EE6 EE5 EE4 EE3 EE2 EE1 EE0	[7] [6] [5] [4] [3] [2] [1] [0]	R	External Event Masked Interrupt Status Bit CM_WMISR = CM_WIMSCR and CM_WRISR 0 = Source X interrupt does not occur 1 = Source X interrupt occurs	0

NOTE: On a Read, the CM_WRISR register gives the current unmasked status value of the corresponding interrupt. A Write has no effect.

5.9.1.26 CM_EEICR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0094, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																								EE7	EE6	EE5	EE4	EE3	EE2	EE1	EE0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	R	Reserved	0
EE7 EE6 EE5 EE4 EE3 EE2 EE1 EE0	[7] [6] [5] [4] [3] [2] [1] [0]	W	External Event Interrupt Clear Bit 0 = No Effect 1 = Clears the external event interrupt	0

NOTE: On a Write of 1, it clears the corresponding interrupt. A Write of 0 has no effect.

5.9.1.27 CM_NISR

- Base Address: 0x4002_0000
- Address = Base Address + 0x0094, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NVIC31	NVIC30	NVIC29	NVIC28	NVIC27	NVIC26	NVIC25	NVIC24	NVIC23	NVIC22	NVIC21	NVIC20	NVIC19	NVIC18	NVIC17	NVIC16	NVIC15	NVIC14	NVIC13	NVIC12	NVIC11	NVIC10	NVIC9	NVIC8	NVIC7	NVIC6	NVIC5	NVIC4	NVIC3	NVIC2	NVIC1	NVIC0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
NVICy	[y]	RW	**C**ontrol Interrupt to NVIC of CORTEX-M0 0 = NVICx interrupt does not occur 1 = NVICx interrupt occurs When IDLEW bit of CM_SR is set to "1" this register is effective. Interrupt signal is handed over to NVIC regardless of CM_NISR register. When IDLEW bit is set to "1" (0xE000_E100 value to CM_NISR), you should copy the values of Interrupt Set Register of CPU to this register to generate interrupt.)	0

5.9.1.28 CM_PSR

- Base Address: 0x4002_0000
- Address = Base Address + 0x00A4, Reset Value = 0x0000_0002

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
NORIVC	[1]	R	Normal IVC Stable Bit 0 = Not stable 1 = IVC for a normal mode is stable	1
RSVD	[0]	R	Reserved	0

5.10 Guide-Clock Initialization

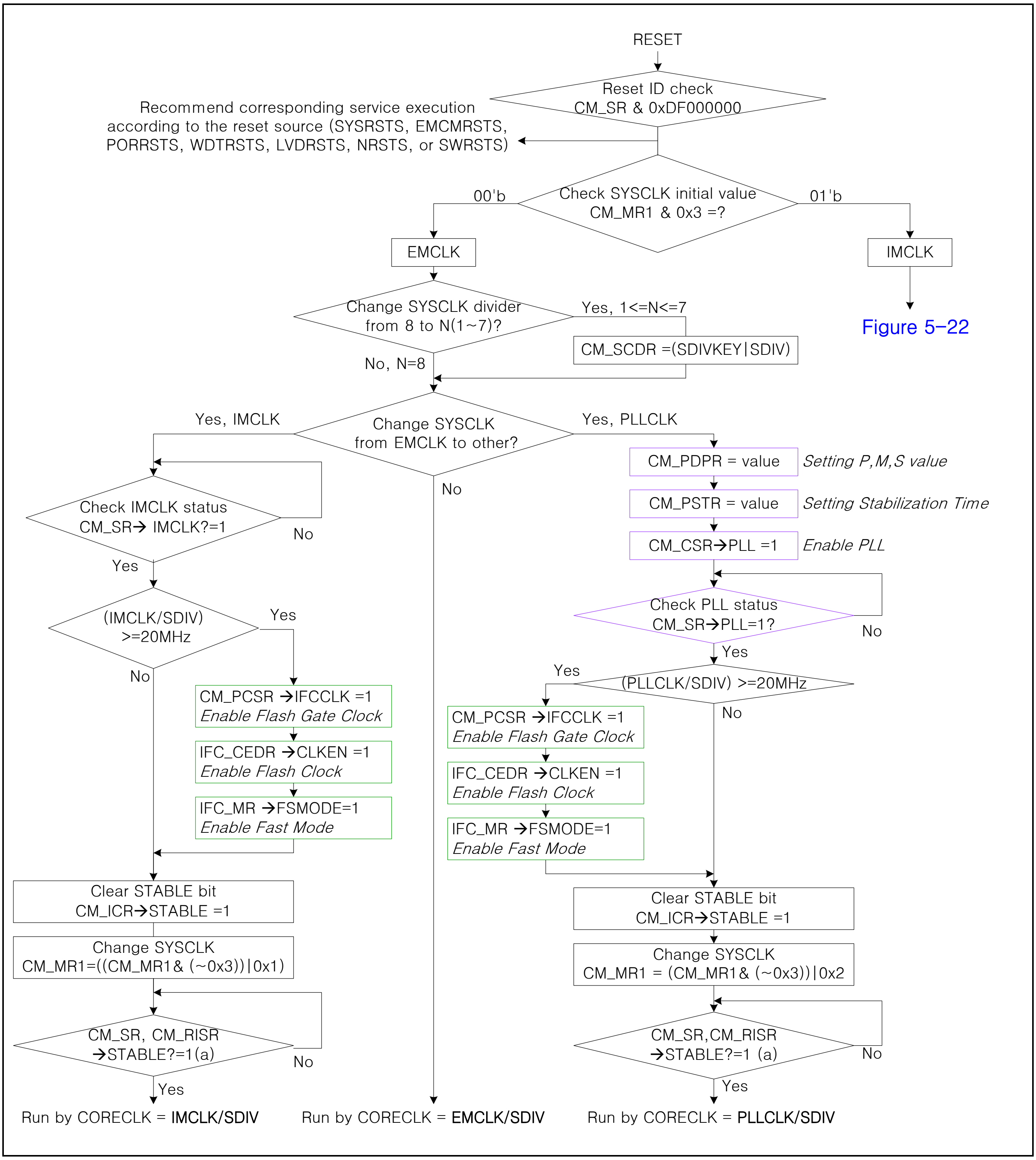


Figure 5-21 Clock Initialization when Reset Value of SYSCLK is EMCLK

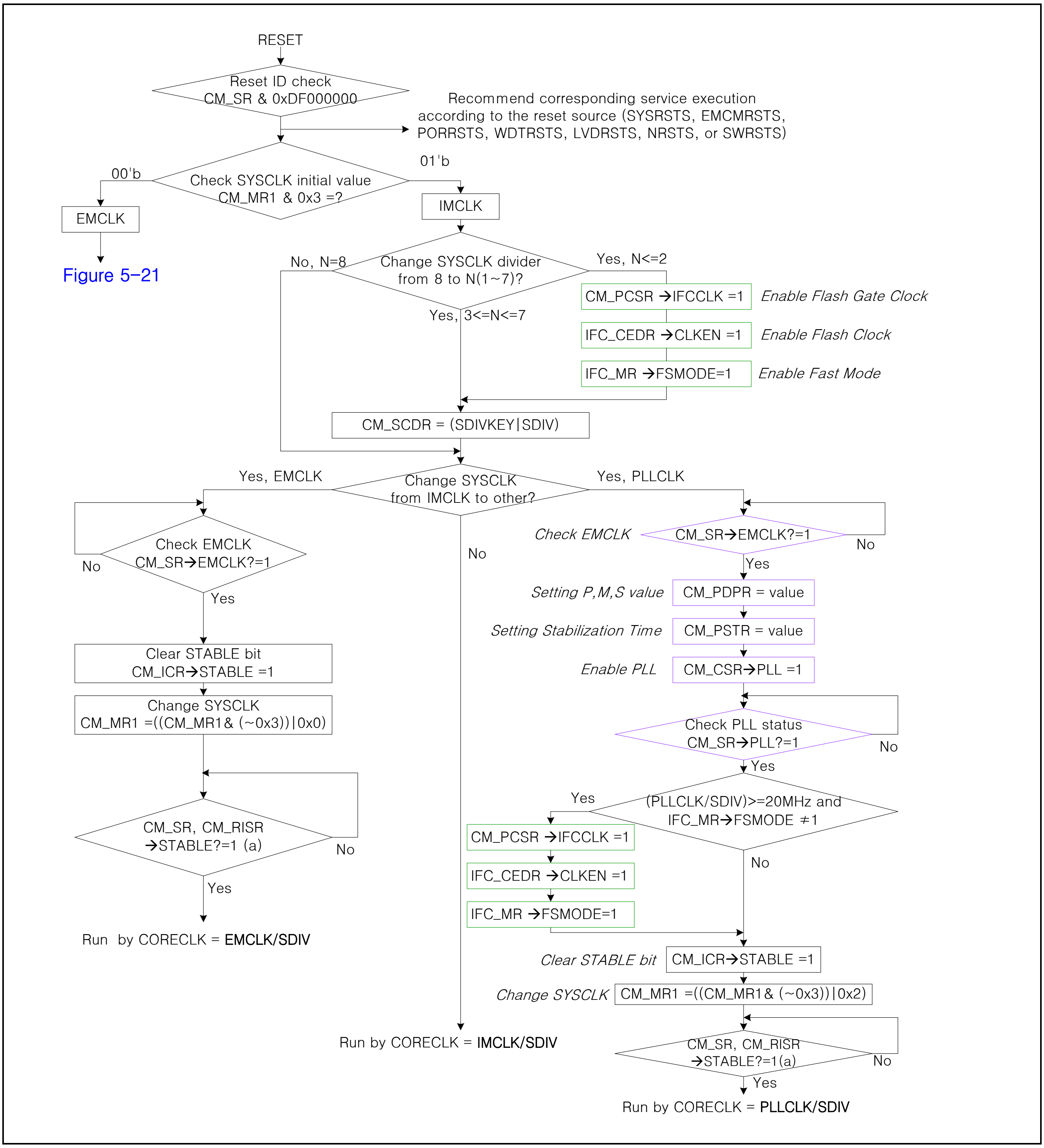


Figure 5-22 Clock Initialization when Reset Value of SYSCLK is IMCLK

If you use an interrupt method, you should configure corresponding interrupt source, interrupt vector, and interrupt service routine before target event execution. In (a) step on Figure 5-21 and Figure 5-22, you need to check CM_MISR instead of CM_RISR register.

For example, if you use STABLE interrupt when changing SYSCLK, the interrupt configuration includes the followings.

- Interrupt handler function (Also be called Interrupt Service Routine (ISR))
- Interrupt source selection and unmask (Interrupt Enable)
- Interrupt vector enable

```
/* Define interrupt handler (service) routine */
Void ISR_CM
{
if((CM_MISR & STABLE)==STABLE)
{
CM_ICR→STABLE =1; /* Clear interrupt pending bit */
}
UserCommand(); /* Handler code for STABLE interrupt*/
}

/* Enable CM (included STABLE) interrupt vector (IRQ14) */
NVIC_ISER→ IRQ14 = 1

/* Enable STABLE interrupt */
CM_IMSCR → STABLE =1
```

6 Comparator

6.1 Overview

The chapter describes comparators in S3FN429 device. This microcontroller has four Comparators. The operation of four Comparators is individually controlled by registers.

6.1.1 Features

The three features of Comparators are:

- Configurable reference voltage selection
- Output voltage of OP-AMP is used for comparator input.
- Comparator output is used for Inverter Motor Controller (IMC) or Pulse Position Decoder (PPD) block.

6.1.2 Pin Description

Table 6-1 describes the pin description of Comparator.

Table 6-1 Pin Description

Pin Name	Function	I/O Type	Comments
COMPx_P	Comparator Positive Input	I	–
COMPx_N	Comparator Negative Input	I	–

NOTE: "x" represents the channel for a Comparator. For example, Comparator 0 has channels represented as COMP0_P and COMP0_N.

6.2 Functional Description

The function of the comparator is to handle:

- Block Diagram
- Comparator input
- Comparator output
- IMC Output-off Control
- Interrupt

6.2.1 Block Diagram

Figure 6-1 illustrates the comparator block diagram.

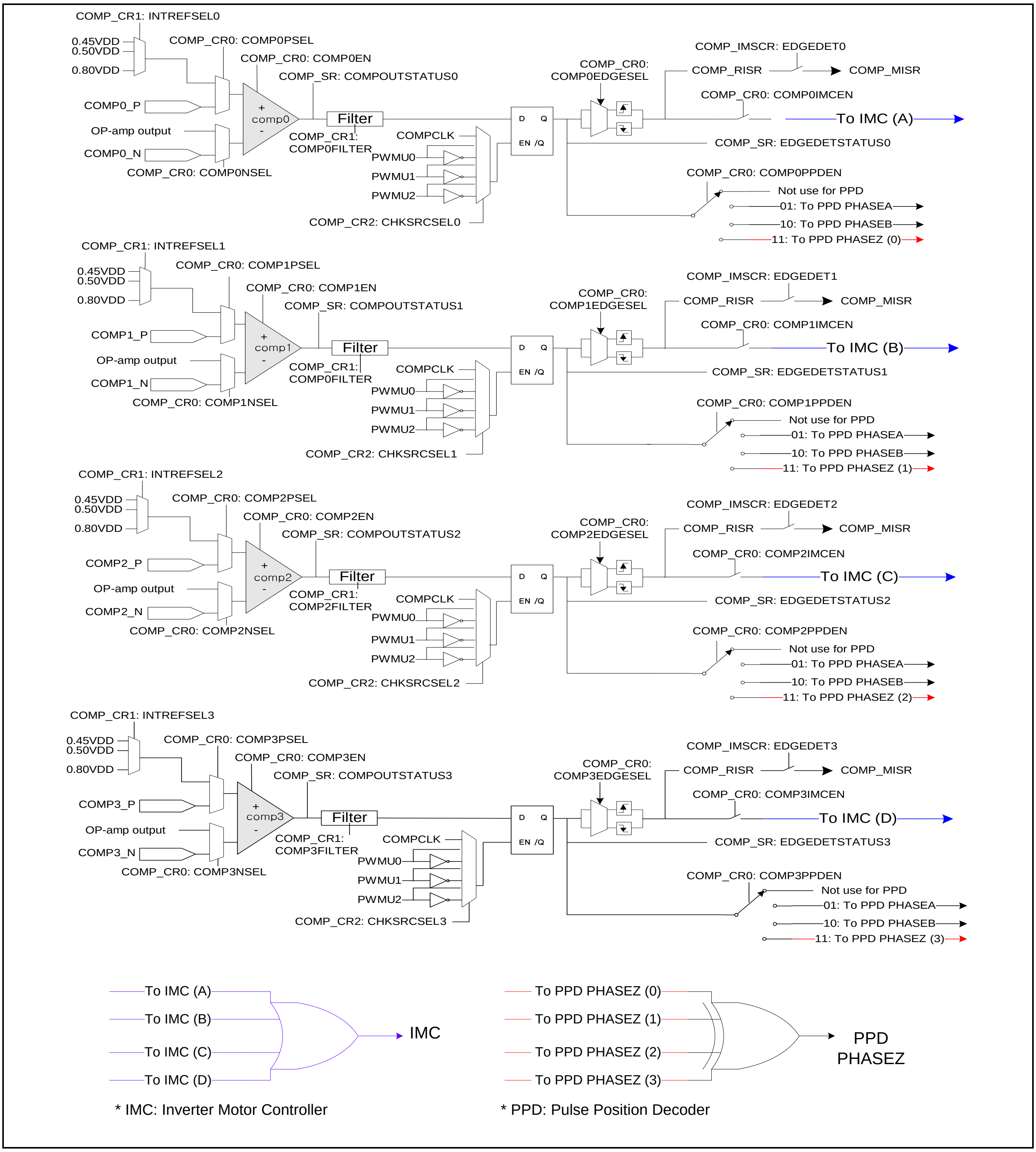


Figure 6-1 Comparator Block Diagram

6.2.2 Comparator Input

The function of comparator is to compare two input signals. The input signals are:

- Reference for signal value to compare
- Target signal

Comparator supports an internal or external reference. Set COMPxPSEL bit to 1 in COMP_CR0 register to use an internal reference value. Range is 0.45 V_{DD} to 0.80 V_{DD}.

6.2.3 Comparator Output

- The output is in two states The two states are:
  - 0 = Voltage of comparator N input > Voltage of comparator P input
  - 1 = Voltage of comparator P input > Voltage of comparator N input

6.2.4 IMC Output-Off Control

The output of comparator is used to cut-off IMC output signal.

6.2.5 Interrupt

There is an edge detection interrupt (EDGEDETx). Use selected interrupt mode edge detection to generate an interrupt.

6.2.5.1 Interruption Handling

The procedure for interrupt handling is:

1. Interrupt Service Routine (ISR) Entry and call C function.
2. Read COMP_IMSR and verify the source of interrupt.
3. Clear the corresponding interrupt at peripheral level by writing in COMP_ICR.
4. Interrupt treatment.
5. Exit ISR.

6.3 Register Description

6.3.1 Register Map Summary

- Base Address: 0x4004_2000

Register	Offset	Description	Reset Value
COMP_IDR	0x0000	ID register	0x0001_0038
COMP_CEDR	0x0004	Clock enable/disable register	0x0000_0000
COMP_SRR	0x0008	Software reset register	0x0000_0000
COMP_CR0	0x000C	Control register 0	0x0000_0000
COMP_CR1	0x0010	Control register 1	0x0000_0000
COMP_CR2	0x0014	Control register 2	0x0000_0000
COMP_SR	0x0018	Status register	0x000F_0000
COMP_IMSCR	0x001C	Interrupt mask set/clear register	0x0000_0000
COMP_RISR	0x0020	Raw interrupt status register	0x0000_0000
COMP_MISR	0x0024	Masked interrupt status register	0x0000_0000
COMP_ICR	0x0028	Interrupt clear register	0x0000_0000

6.3.1.1 COMP_IDR

- Base Address: 0x4004_2000
- Address = Base Address + 0x0000, Reset Value = 0x0001_0038

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores ID code for the corresponding IP.	0x00010038

6.3.1.2 COMP_CEDR

- Base Address: 0x4004_2000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBGEN	RSVD																														CLKEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	RW	Debug Mode Enable/Disable Control Bit 0 = Disables debug mode. In debug mode, Timer/Counter (TC) is not halted. (No influence on the function) 1 = Enables debug mode.	0
RSVD	[30:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable/Disable Control Bit 0 = Disables Counter Clock. 1 = Enables Counter Clock.	0

6.3.1.3 COMP_SRR

- Base Address: 0x4004_2000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Resets Software Comparator IP Block.	0

6.3.1.4 COMP_CR0

- Base Address: 0x4004_2000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD			COMP3EDGESEL		COMP3NINSEL	COMP3PINSEL	COMP3EN	RSVD			COMP2EDGESEL		COMP2NINSEL	COMP2PINSEL	COMP2EN	RSVD			COMP1EDGESEL		COMP1NINSEL	COMP1PINSEL	COMP1EN	RSVD			COMP0EDGESEL		COMP0NINSEL	COMP0PINSEL	COMP0EN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:29] [23:21] [15:13] [7:5]	RW	Reserved	0
COMP3EDGESEL COMP2EDGESEL COMP1EDGESEL COMP0EDGESEL	[28:27] [20:19] [12:11] [4:3]	RW	Comparator x Edge Detection Enable 00 = Selects falling edge. 01 = Selects rising edge. 10 = Selects falling/rising edge. 11 = No effect	00'b
COMP3NINSEL COMP2NINSEL COMP1NINSEL COMP0NINSEL	[26] [18] [10] [2]	RW	Comparator x Negative Input Selection Bit 0 = Selects external pin (COMPx_N). 1 = Selects OP-AMP 0 output.	0'b
COMP3PINSEL COMP2PINSEL COMP1PINSEL COMP0PINSEL	[25] [17] [9] [1]	RW	Comparator x Positive Input Selection Bit 0 = Selects external pin (COMPx_P). 1 = Selects internal reference.	0'b
COMP3EN COMP2EN COMP1EN COMP0EN	[24] [16] [8] [0]	RW	Comparator x Enable Bit 0 = Disables Comparator x. 1 = Enables Comparator x.	0'b

NOTE: Value of "x" from 0 to 3 represents channel of a comparator. For example, COMPx_P have channels represented as COMP0_P, COMP1_P, COMP2_P, and COMP3_P.

6.3.1.5 COMP_CR1

- Base Address: 0x4004_2000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD	COMP3FILTER			RSVD	INTREFSEL3			RSVD	COMP2FILTER			RSVD	INTREFSEL2			RSVD	COMP1FILTER			RSVD	INTREFSEL1			RSVD	COMP0FILTER			RSVD	INTREFSEL0		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31]	R	Reserved	0
COMP3FILTER	[30:28]	RW	Comparator 3 Edge Detection Filter Selection Bit 000 = bypass 001 = PCLK/1 010 = PCLK/16 011 = PCLK/64 100 = PCLK/128 101 = PCLK/256 110 = PCLK/512 111 = PCLK/1024	000'b
RSVD	[27]	R	Reserved	0
INTREFSEL3	[26:24]	RW	Comparator 3 Reference Level Selection Bit 000 = 0.45 V_{DD} 001 = 0.50 V_{DD} 010 = 0.55 V_{DD} 011 = 0.60 V_{DD} 100 = 0.65 V_{DD} 101 = 0.70 V_{DD} 110 = 0.75 V_{DD} 111 = 0.80 V_{DD}	000'b
RSVD	[23]	R	Reserved	0
COMP2FILTER	[22:20]	RW	Comparator 2 Edge Detection Filter Selection Bit 000 = bypass 001 = PCLK/1 010 = PCLK/16 011 = PCLK/64 100 = PCLK/128 101 = PCLK/256 110 = PCLK/512	000'b

Name	Bit	Type	Description	Reset Value
			111 = PCLK/1024	
RSVD	[19]	R	Reserved	0
INTREFSEL2	[18:16]	RW	Comparator 2 Reference Level Selection Bit 000 = 0.45 V_{DD} 001 = 0.50 V_{DD} 010 = 0.55 V_{DD} 011 = 0.60 V_{DD} 100 = 0.65 V_{DD} 101 = 0.70 V_{DD} 110 = 0.75 V_{DD} 111 = 0.80 V_{DD}	000'b
RSVD	[15]	R	Reserved	0
COMP1FILTER	[14:12]	RW	Comparator 1 Edge Detection Filter Selection Bit 000 = bypass 001 = PCLK/1 010 = PCLK/16 011 = PCLK/64 100 = PCLK/128 101 = PCLK/256 110 = PCLK/512 111 = PCLK/1024	000'b
RSVD	[13]	R	Reserved	0
INTREFSEL1	[10:8]	RW	Comparator 1 Reference Level Selection Bit 000 = 0.45 V_{DD} 001 = 0.50 V_{DD} 010 = 0.55 V_{DD} 011 = 0.60 V_{DD} 100 = 0.65 V_{DD} 101 = 0.70 V_{DD} 110 = 0.75 V_{DD} 111 = 0.80 V_{DD}	000'b
RSVD	[7]	R	Reserved	0
COMP0FILTER	[6:4]	RW	Comparator 0 Edge Detection Filter Selection Bit 000 = bypass 001 = PCLK/1 010 = PCLK/16 011 = PCLK/64 100 = PCLK/128 101 = PCLK/256 110 = PCLK/512 111 = PCLK/1024	000'b
RSVD	[3]	R	Reserved	0
INTREFSEL0	[2:0]	RW	Comparator 0 Reference Level Selection Bit 000 = 0.45 V_{DD} 001 = 0.50 V_{DD} 010 = 0.55 V_{DD}	000'b

Name	Bit	Type	Description	Reset Value
			011 = 0.60 V_{DD} 100 = 0.65 V_{DD} 101 = 0.70 V_{DD} 110 = 0.75 V_{DD} 111 = 0.80 V_{DD}	

6.3.1.6 COMP_CR2

- Base Address: 0x4004_2000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD	COMP3PPDEN			COMP3IMCEN	CHKSRCSEL3			RSVD	COMP2PPDEN			COMP2IMCEN	CHKSRCSEL2			RSVD	COMP1PPDEN			COMP1IMCEN	CHKSRCSEL1			RSVD	COMP0PPDEN			COMP0IMCEN	CHKSRCSEL0		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31] [23] [15] [7]	RW	Reserved	0
COMP3PPDEN COMP2PPDEN COMP1PPDEN COMP0PPDEN	[30:28] [22:20] [14:12] [6:4]	RW	Comparator x and PPD Connect Selection 000 = Disconnects to PPD. 001 = Connects a comparator output to PPD PHASEA. 010 = Connects a comparator output to PPD PHASEB. 011 = Connects a comparator output to PPD PHASEZ. 100 = Connects a comparator output to PPD PHASEA and PHASEZ. 101 = Connects a comparator output to PPD PHASEB and PHASEZ. Others = Invalid	000'b
COMP3IMCEN COMP2IMCEN COMP1IMCEN COMP0IMCEN	[27] [19] [11] [3]	RW	IMC Output Signal Floating By Comparator Enable 0 = Disables 1 = Enables	0'b
CHKSRCSEL3 CHKSRCSEL2 CHKSRCSEL1 CHKSRCSEL0	[26:24] [18:16] [10:8] [2:0]	RW	Edge Detect Status Check Source Selection Bit 000 = Selects COMPCLK bit. 001 = Selects PWMU0 signal. 010 = Selects opposite (inverted) PWMU0 signal. 011 = Selects PWMU1 signal. 100 = Selects opposite (inverted) PWMU1 signal. 101 = Selects PWMU2 signal. 110 = Selects opposite (inverted) PWMU2 signal. 111 = Not used	000'b

NOTE: PPD stands for "Pulse Position Decoder" block. IMC stands for "Inverter Motor Controller" block.

6.3.1.7 COMP_SR

- Base Address: 0x4004_2000
- Address = Base Address + 0x0018, Reset Value = 0x000F_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD												COMPOUTSTATUS3	COMPOUTSTATUS2	COMPOUTSTATUS1	COMPOUTSTATUS0	RSVD												EDGEDETSTATUS3	EDGEDETSTATUS2	EDGEDETSTATUS1	EDGEDETSTATUS0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:20]	R	Reserved	0
COMPOUTSTATUS3 COMPOUTSTATUS2 COMPOUTSTATUS1 COMPOUTSTATUS0	[19] [18] [17] [16]	R	Comparator Output Real Time Status 0 = Specifies Voltage of comparator N input > Voltage of comparator P input. 1 = Specifies Voltage of comparator P input > Voltage of comparator N input.	1
RSVD	[15:4]	R	Reserved	0
EDGEDETSTATUS3 EDGEDETSTATUS2 EDGEDETSTATUS1 EDGEDETSTATUS0	[3] [2] [1] [0]	R	Comparator Output Edge Detect Status at Edge Detection 0 = Voltage of comparator N input > Voltage of comparator P input 1 = Voltage of comparator P input > Voltage of comparator N input	0

6.3.1.8 COMP_IMSCR

- Base Address: 0x4004_2000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																												EDGEDET3	EDGEDET2	EDGEDET1	EDGEDET0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
EDGEDET3	[3]	RW	Edge Detection Interrupt Mask 0 = Specifies Edge Detection interrupt is masked. (Disables the interrupt) 1 = Specifies Edge Detection interrupt is not masked. (Enables the interrupt)	0
EDGEDET2	[2]	RW	Edge Detection Interrupt Mask 0 = Specifies Edge Detection interrupt is masked. (Disables the interrupt) 1 = Specifies Edge Detection interrupt is not masked. (Enables the interrupt)	0
EDGEDET1	[1]	RW	Edge Detection Interrupt Mask 0 = Specifies Edge Detection interrupt is masked. (Disables the interrupt) 1 = Specifies Edge Detection interrupt is not masked. (Enables the interrupt)	0
EDGEDET0	[0]	RW	Edge Detection Interrupt Mask 0 = Specifies Edge Detection interrupt is masked. (Disables the interrupt) 1 = Specifies Edge Detection interrupt is not masked. (Enables the interrupt)	0

NOTE: On a Read, the COMP_IMSCR register gives the current value of the mask on the relevant interrupt.
A Write of 1 to a particular bit sets the mask and enables the interrupt.
A Write of 0 to a particular bit clears the corresponding mask.

6.3.1.9 COMP_RISR

- Base Address: 0x4004_2000
- Address = Base Address + 0x0020, Reset Value = 0x0000_000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																												EDGEDET3	EDGEDET2	EDGEDET1	EDGEDET0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
EDGEDET3	[3]	R	Edge Detection Raw Interrupt Status Gives the raw interrupt state (prior to masking) of EDGEDET3 interrupt.	0
EDGEDET2	[2]	R	Edge Detection Raw Interrupt Status Gives the raw interrupt state (prior to masking) of EDGEDET2 interrupt.	0
EDGEDET1	[1]	R	Edge Detection Raw Interrupt Status Gives the raw interrupt state (prior to masking) of EDGEDET1 interrupt.	0
EDGEDET0	[0]	R	Edge Detection Raw Interrupt Status Gives the raw interrupt state (prior to masking) of EDGEDET0 interrupt.	0

NOTE: On a Read, the COMP_RISR register gives the current raw status value of the corresponding interrupt prior to masking. A Write has no effect.

6.3.1.10 COMP_MISR

- Base Address: 0x4004_2000
- Address = Base Address + 0x0024, Reset Value = 0x0000_000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																												EDGEDET3	EDGEDET2	EDGEDET1	EDGEDET0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
EDGEDET3	[3]	R	Edge Detection Masked Interrupt State Provides masked interrupt status of EDGEDET3 interrupt	0
EDGEDET2	[2]	R	Edge Detection Masked Interrupt State Provides masked interrupt status of EDGEDET2 interrupt	0
EDGEDET1	[1]	R	Edge Detection Masked Interrupt State Provides masked interrupt status of EDGEDET1 interrupt	0
EDGEDET0	[0]	R	Edge Detection Masked Interrupt State Provides masked interrupt status of EDGEDET0 interrupt	0

NOTE: On a Read, the COMP_MISR register gives the current masked status value of the corresponding interrupt. A Write has no effect.

6.3.1.11 COMP_ICR

- Base Address: 0x4004_2000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																												EDGEDET3	EDGEDET2	EDGEDET1	EDGEDET0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
EDGEDET3	[3]	W	Edge Detection Interrupt Clear Gives the raw interrupt state (prior to masking) of EDGEDET3 interrupt.	0
EDGEDET2	[2]	W	Edge Detection Interrupt Clear Gives the raw interrupt state (prior to masking) of EDGEDET2 interrupt.	0
EDGEDET1	[1]	W	Edge Detection Interrupt Clear Gives the raw interrupt state (prior to masking) of EDGEDET1 interrupt.	0
EDGEDET0	[0]	W	Edge Detection Interrupt Clear Gives the raw interrupt state (prior to masking) of EDGEDET0 interrupt.	0

NOTE: A Write of 1, clears the corresponding interrupt. A Write of 0 has no effect.

7 Pulse Position Decoder

7.1 Overview

The Pulse Position Decoder (PPD) chapter describes the PPD that you can use for measuring the position and speed.

7.1.1 Features

The features of PPD are:

- The three input signals of PPD are:
  - PHASEA
  - PHASEB
  - PHASEZ
- The two 16-bit up/down counters of PPD are:
  - Position Counter (PCR)
  - Speed Counter (SPCR)
- The two 16-bit capture timers of PPD are:
  - Position Capture Timer (PCT)
  - Speed Capture Timer (SCT)
- PPD has filter in the PHASEZ and edge selector for PHASEZ

7.1.2 Pin Description

Table 7-1 describes the functions of each pin on the PPD.

Table 7-1 PPD Pin Description

Pin Name	Function	I/O Type	Comments
PHASEA	Phase A input	I	–
PHASEB	Phase B input	I	–
PHASEZ	Phase Z input	I	–

7.2 Functional Description

This section describes the functional description of PPD.

7.2.1 Block Diagram

Figure 7-1 illustrates the block diagram for PPD.

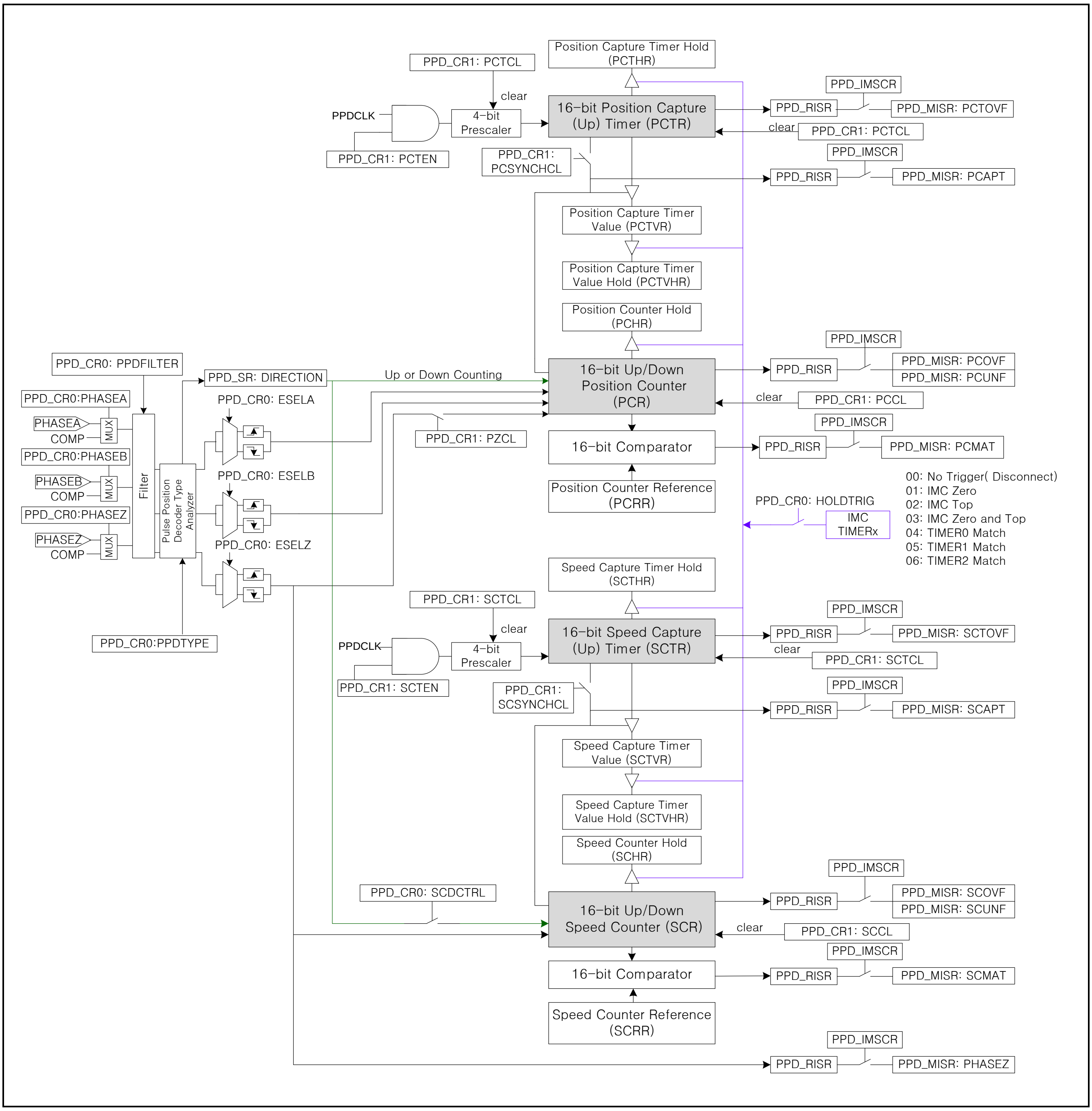


Figure 7-1 PPD Block Diagram

7.2.2 Operating Mode

The Operating Mode section describes the types of operating modes of PPD. The modes are:

- Type 0
- Type 1
- Type 2

7.2.2.1 Type 0

Type 0 includes two modes. The two modes are:

- – 4 Multiplication Mode
- – 1 Multiplication Mode

7.2.2.1.1 4 Multiplication Mode (PPD_CR0.ESELA = 0b10, PPD_CR0.ESELB = 0b10)

Use the three input signals, PHASEA, PHASEB, and PHASEZ to measure the position and speed for Type 0. Use 4 multiplication modes, if the phase difference between PHASEA and PHASEB pulse is 90°. The PHASEZ input generates one-pulse signal at specific position 1 cyclic.

Figure 7-2 illustrates the counter operation for 4 Multiplication Mode.

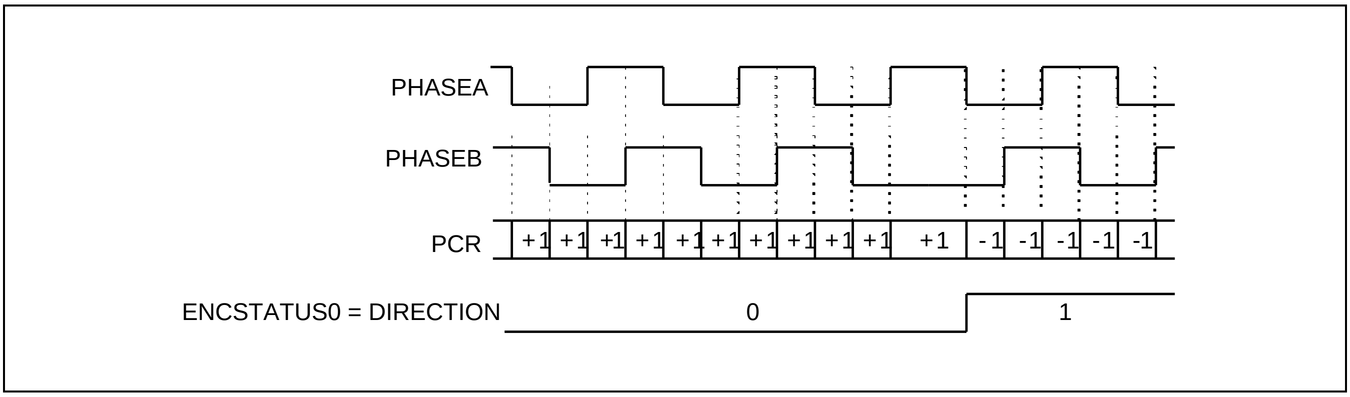


Figure 7-2 Counter Operation (4 Multiplication Mode)

7.2.2.1.2 1 Multiplication Mode (PPD_CR0.ESELA = 0b00, PPD_CR0.ESELB = 0b11)

Use the three input signals, PHASEA, PHASEB, and PHASEZ to measure the position and speed of Type 0. Use 1 multiplication mode if the phase difference between PHASEA and PHASEB pulse is not 90°. The PHASEZ input generates one-pulse signal at specific position 1 cyclic.

Figure 7-3 illustrates the counter operation for 1 Multiplication mode.

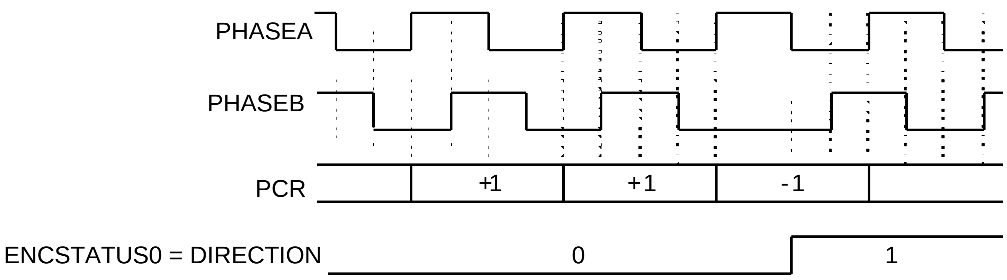


Figure 7-3 Counter Operation (1 Multiplication Mode)

Direction of Rotation

When the DIRECTION bit is 0, the counter value of PCR increases, and when the DIRECTION bit is 1, the counter value of PCR decreases. PCR is an up-down counter.

The leading phase signal between PHASEA and PHASEB decides the DIRECTION bit status and counting direction.

NOTE: Although the PBEN and PAEN bit are "0", disable, if inserted any signal into PHASE A or PHASE B input port and CAP_A0/A1 or CAP_B0/B1 interrupt are unmask, those interrupts occur.

7.2.2.2 Type 1

If the input pulse is applied into PHASEA, the PCR increases (up counting). If the input pulse is applied into PHASEB, the PCR decreases (down counting).

Figure 7-4 illustrates the counter operation for Type 1.

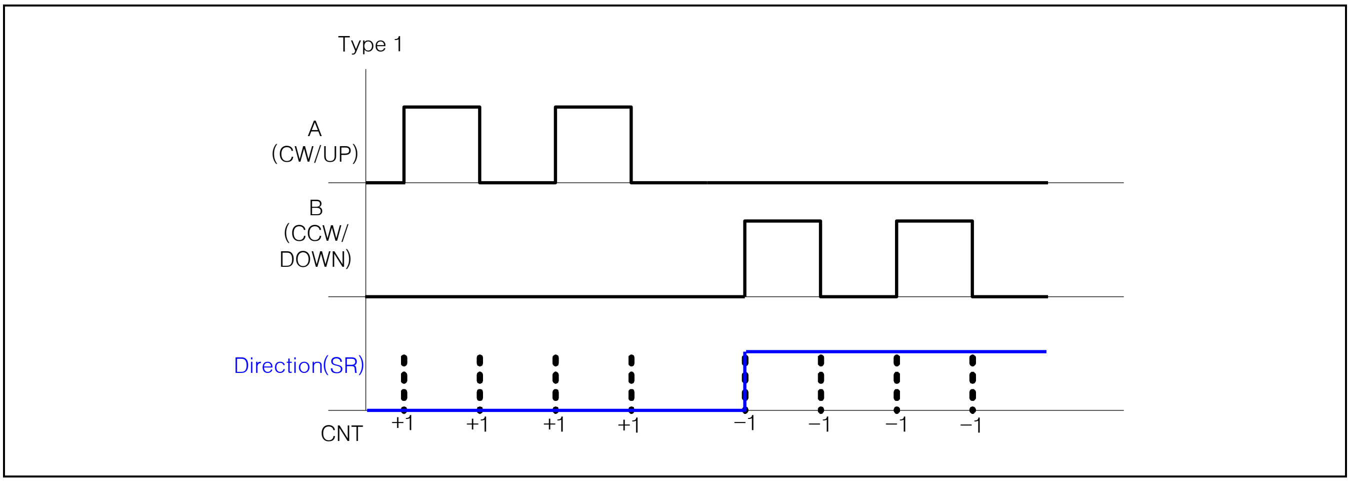


Figure 7-4 Counter Operation (Type 1)

7.2.2.3 Type 2

If the high input is applied into PHASEA, the PCR increases (up counting). If the low input is applied into PHASEB, the PCR decreases according to direction signal (down counting).

Figure 7-5 illustrates the counter operation for Type 2.

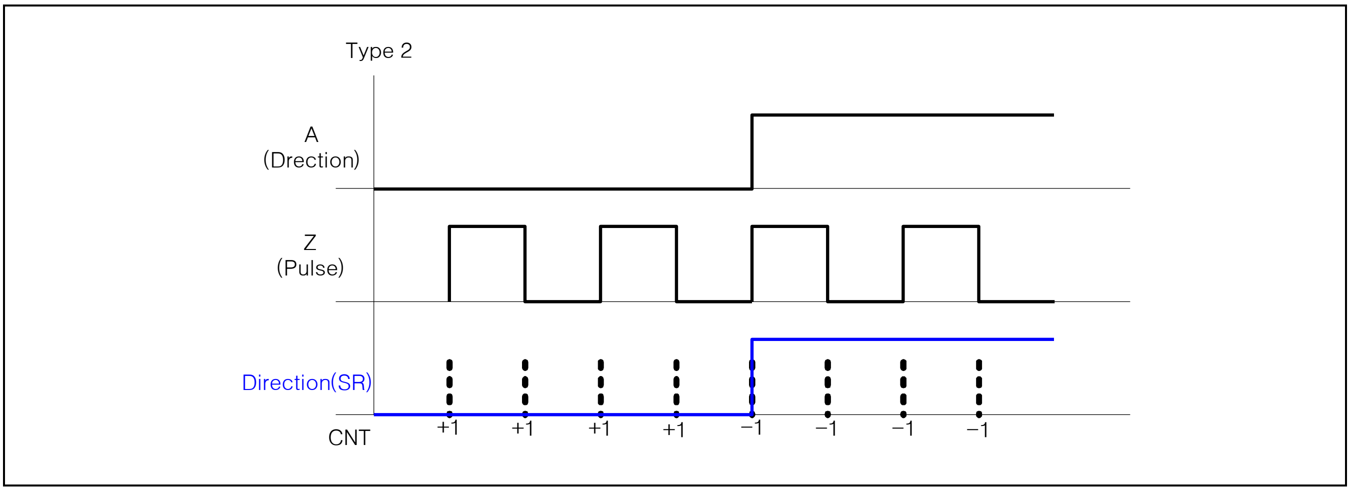


Figure 7-5 Counter Operation (Type 2)

7.3 Register Description

7.3.1 Register Map Summary

- Base Address: 0x400C_0000

Register	Offset	Description	Reset Value
PPD_IDR	0x0000	ID register	0x0001_8703
PPD_CEDR	0x0004	Clock enable/disable register	0x0000_0000
PPD_SRR	0x0008	Software reset register	0x0000_0000
PPD_CR0	0x000C	Control register 0	0x0000_0000
PPD_CR1	0x0010	Control register 1	0x0000_0000
PPD_SR	0x0014	Status register	0x0000_0000
PPD_IMSCR	0x0018	Interrupt mask set and clear register	0x0000_0000
PPD_RISR	0x001C	Raw interrupt status register	0x0000_0000
PPD_MISR	0x0020	Masked interrupt status register	0x0000_0000
PPD_ICR	0x0024	Interrupt clear register	0x0000_0000
PPD_PCR	0x0028	Position counter register	0x0000_0000
PPD_PCRR	0x002C	Position counter reference register	0x0000_0000
PPD_PCTR	0x0030	Position capture timer register	0x0000_0000
PPD_PCTVR	0x0034	Position capture timer value register	0x0000_0000
PPD_SCR	0x0038	Speed counter register	0x0000_0000
PPD_SCRR	0x003C	Speed counter reference register	0x0000_0000
PPD_SCTR	0x0040	Speed capture timer register	0x0000_0000
PPD_SCTVR	0x0044	Speed capture timer value register	0x0000_0000
PPD_PCHR	0x0048	Position counter hold register	0x0000_0000
PPD_PCTHR	0x004C	Position capture timer hold register	0x0000_0000
PPD_PCTVHR	0x0050	Position capture timer value hold register	0x0000_0000
PPD_SCHR	0x0054	Speed counter hold register	0x0000_0000
PPD_SCTHR	0x0058	Speed capture timer hold register	0x0000_0000
PPD_SCTVHR	0x005C	Speed capture timer value hold register	0x0000_0000

NOTE: PPD_PCR and PPD_SCR are 2's complement. The range of PCR and SCR is – 2^{15} to (+ $2^{15\,-}$ 1).

7.3.1.1 PPD_IDR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0000, Reset Value = 0x0001_8703

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	Identification Code Register This field stores the ID code for the corresponding IP.	0x0001_8703

7.3.1.2 PPD_CEDR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	RW	Debug Enable Bit 0 = Disables debug mode bit PPD is not halted during processor debug mode. 1 = Enables debug mode bit PPD is halted during processor debug mode.	0'b
RSVD	[30:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable 0 = Disables PPD Clock 1 = Enables PPD Clock	0'b

7.3.1.3 PPD_SRR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="31">RSVD</td><td>SWRST</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>W</td></tr>
</table>

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Performs PPD Software Reset	0'b

7.3.1.4 PPD_CR0

- Base Address: 0x400C_0000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HOLDTRIG				RSVD	PPDFILTER			RSVD	ESELZ		PHASEZ			ESELB		PHASEB			ESELA		PHASEA			RSVD	SCDCTRL	PPDTYPE		PPDEN	PPDCLKSEL		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Name	Bit	Type	Description	Reset Value
HOLDTRIG	[31:28]	RW	Hold Trigger Source This field determines the trigger source for hold function. 0000 = No Trigger (Disconnect) 0001 = IMC Zero 0010 = IMC Top 0011 = IMC Zero and Top 0100 = TIMER0 Match 0101 = TIMER1 Match 0110 = TIMER2 Match Others = Not used [(2)]	0000'b
RSVD	[27]	R	Reserved	0
PPDFILTER	[26:24]	RW	Filter Clock Selection of Pulse Position Decoder This field determines the filter clock selection of a pulse position decoder. 000 = PPDCLK 001 = PPDCLK/2 010 = PPDCLK/4 011 = PPDCLK/8 100 = PPDCLK/16 101 = PPDCLK/32 110 = PPDCLK/64 111 = PPDCLK/128 NOTE: Recognizes as effective signal only five times in a row in the same level.	000'b
RSVD	[23]	R	Reserved	0
ESELZ	[22:21]	RW	PHASEZ Edge Selection This field determines the edge selection for PHASEZ. 00 = Selects rising edge of PHASEZ input 01 = Selects falling edge of PHASEZ input	00'b

Name	Bit	Type	Description	Reset Value
			10 =Selects both (rising and falling) edge of PHASEZ input 11 = No detection at any edge of PHASEZ	
PHASEZ	[20:18]	RW	PHASEZ Signal Selection Bits 000 = Specifies PHASEZ signal is the external signal, asserted on PHASEZ pin. Do not use the signal asserted from COMP as PHASEZ. 001 = Use the signal asserted from COMP as PHASEZ Others = Not used [2]	000'b
ESELB	[17:16]	RW	PHASEB Edge Selection Bits This field determines the edge selection for PHASEB. 00 = Selects rising edge of PHASEB input 01 = Selects falling edge of PHASEB input 10 =Selects both (rising and falling) edge of PHASEB input 11 = No detection at any edge of PHASEB	00'b
PHASEB	[15:13]	RW	PHASEB Signal Selection Bits 000 = Specifies PHASEB signal is the external signal, asserted on PHASEB pin. Do not use the signal asserted from COMP as PHASEB. 001 = Use the signal asserted from COMP0 as PHASEB. 010 = Use the signal asserted from COMP1 as PHASEB. 011 = Use the signal asserted from COMP2 as PHASEB. 100 = Use the signal asserted from COMP3 as PHASEB. Others = Not used [2]	000'b
ESELA	[12:11]	RW	PHASEA Edge Selection Bits This field determines the edge selection for PHASEA. 00 = Selects rising edge of PHASEA input 01 =Selects falling edge of PHASEA input 10 = Selects both (rising and falling) edge of PHASEA input 11 = No detection at any edge of PHASEA	00'b
PHASEA	[10:8]	RW	PHASEA Signal Selection Bit 000 = Specifies PHASEA signal is the external signal, asserted on PHASEA pin. Do not use the signal asserted from COMP as PHASEA. 001 = Use the signal asserted from COMP0 as PHASEA. 010 = Use the signal asserted from COMP1 as PHASEA. 011 = Use the signal asserted from COMP2 as PHASEA. 100 = Use the signal asserted from COMP3 as PHASEA. Others = Not used [2]	000'b
RSVD	[7]	R	Reserved	0
SCDCTRL	[6]	RW	Speed Counter Direction (Up or Down) Control Bit [1] 0 = Speed counter does not use 'DIRECTION' status which Pulse Position Decoder Type Analyzer detects. Speed counter always increases. 1 = Speed counter uses 'DIRECTION' status which Pulse Position Decoder Type Analyzer detects. Speed counter increases or decreases according to the 'DIRECTION'	0'b

Name	Bit	Type	Description	Reset Value
			information.	
PPDTYPE	[5:4]	RW	Pulse Position Decoder Type Selection Bits The pulse position decoder type analyzer block generates direction and input signal for speed and position counter with this value. 00 = The way to use a pulse position decoder type 0 01 = The way to use a pulse position decoder type 1 10 = The way to use a pulse position decoder type 2 11 = Not used (2) • The feature of Type 1 is: – The difference of phase between phase A and phase B pulse is 90°. • The feature of Type 2 is: – The input of PHASEA is Clockwise (CW) signal. The PCR up -counts using PHASEA pulse. – The input of PHASEB is Counter Clockwise (CCW) signal. The PCR down-counts using the PHASEB pulse. • The feature of Type 3 is: – The input of PHASEA is the direction signal. – The input of PHASEZ is the pulse signal.	00'b
PPDEN	[3]	RW	Pulse Position Decoder Block Enable/Disable Control Bit 0 = Disables pulse position decoder block bit 1 = Enables pulse position decoder block bit	0'b
PPDCLKSEL	[2:0]	RW	Pulse Position Decoder Clock (PPDCLK) Selection This field determines the PPDCLK. 000 = PCLK 001 = PCLK/2 010 = PCLK/4 011 = PCLK/8 100 = PCLK/16 101 = PCLK/32 110 = PCLK/64 111 = PCLK/128	000'b

NOTE:

1. SCDCTRL bit is valid only in Type 0. This bit has no effect in type 1 and 2.
2. If you write a not-used value for control bits, the value will not change. The control bit will have the previous value.

7.3.1.5 PPD_CR1

- Base Address: 0x400C_0000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PZCL	RSVD			SCTPRESCALE				RSVD				SCSYNCHCL	SCCL	SCTCL	SCTEN	RSVD				PCTPRESCALE				RSVD				PCSYNCHCL	PCCL	PCTCL	PCTEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
PZCL	[31]	RW	Position Counter Clear by PHASEZ This field determines PCNT clear enable by Phase Z. 0 = Disables position counter 1 = Enables position counter NOTE: PZCL bit clears position counter only when PPDEN is set to 1.	0'b
RSVD	[30:28]	R	Reserved	0
SCTPRESCALE	[27:24]	RW	Speed Capture Timer Pre-scale Bits This bit sets the pre-scale value for Phase A. SCTCLK = PPDCLK/2^ PRESCALEA	0000'b
RSVD	[23:20]	R	Reserved	0
SCSYNCHCL	[19]	RW	Speed Counter Synchronous Clear Bit 0 =Do not clear a speed capture timer whenever position counter is changed (up or down) 1 = Clear a speed capture timer whenever position counter is changed (up or down)	0'b
SCCL	[18]	RW	Speed Counter Clear This bit clears the content of speed counter register. 0 = No effect 1 = Clears counter register. NOTE: This bit is automatically cleared after clear.	0'b
SCTCL	[17]	RW	Speed Capture Timer Clear Bit 0 = No effect 1 = Clears speed capture timer bit. NOTE: This bit is automatically cleared after clear.	0'b
SCTEN	[16]	RW	Speed Capture Timer Enable This bit enables or disables speed capture timer.	0'b

Name	Bit	Type	Description	Reset Value
			0 = Disables speed capture timer (Stop) 1 = Enables speed capture timer (Start)	
RSVD	[15:12]	R	Reserved	0
PCTPRESCALE	[11:8]	RW	Position Capture Timer Pre-scale Bits This bit sets the pre-scale value for Phase A. PCTCLK = PPDCLK/2^ PRESCALEA	0000'b
RSVD	[7:4]	R	Reserved	0
PCSYNCHCL	[3]	RW	Position Counter Synchronous Clear Bit 0 = Do not clear position capture timer whenever position counter is changed (up or down) 1 = Clear position capture timer whenever position counter is changed (up or down)	0'b
PCCL	[2]	RW	Position Counter Clear Bit This bit clears the content of position counter register. 0 = No effect 1 = Clears the counter register. NOTE: This bit is automatically cleared after clear.	0'b
PCTCL	[1]	RW	Position Capture Timer Clear Bit 0 = No effect 1 = Clears the position capture timer. NOTE: This bit is automatically cleared after clear.	0'b
PCTEN	[0]	RW	Position Capture Timer Enable This bit enables or disables position capture timer. 0 = Disables position capture timer (Stop) 1 = Enables position capture timer (Start)	0'b

NOTE: PCCL, SCCL, PCTCL, and SCTCL are automatically cleared after clears. PZCL bit clears PCR.

7.3.1.6 PPD_SR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																												PBSTAT	PASTAT	GLITCH	DIRECTION
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
PBSTAT	[3]	R	PHASEB Status Bit 0 = Specifies Low level 1 = Specifies High level NOTE: This bit is a read only bit.	0
PASTAT	[2]	R	PHASE A Status Bit 0 = Specifies Low level 1 = Specifies High level NOTE: This bit is a read only bit.	0
GLITCH	[1]	RW	Glitch Detection of PHASEA, PHASEB and PHASEZ This bit notifies the glitch detection in the PHASEA, PHASEB or PHASEZ pin. Read: 0 = Glitch does not occur 1 = Glitch occurs Write: 0 = Clears glitch bit. 1 = No effect NOTE: Detects the Glitch bit after verifying whether five times in a row in the same level recognizes as effective signal.	0
DIRECTION	[0]	R	Direction of Motor Rotation Bit. 0 = Clockwise – Increases the value of PCNT 1 = Counter-clockwise – Decreases the value of PCNT. NOTE: This bit is a read-only bit.	0

7.3.1.7 PPD_IMSCR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD															PHASEZ	RSVD	SCCUNF	SCCOVF	SCTOVF	SCAPT	SCSUNF	SCSOVF	SCMAT	RSVD	PCCUNF	PCCOVF	PCTOVF	PCAPT	PCSUNF	PCSOVF	PCMAT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	R	Reserved	0
PHASEZ	[16]	RW	PHASEZ Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
RSVD	[15]	R	Reserved	0
SCCUNF	[14]	RW	Speed Counter Carry Underflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
SCCOVF	[13]	RW	Speed Counter Carry Overflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
SCTOVF	[12]	RW	Speed Capture Timer Overflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
SCAPT	[11]	RW	Speed Capture Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
SCSUNF	[10]	RW	Speed Counter Sign Underflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
SCSOVF	[9]	RW	Speed Counter Sign Overflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
SCMAT	[8]	RW	Speed Counter Match Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
RSVD	[7]	R	Reserved	0

Name	Bit	Type	Description	Reset Value
PCCUNF	[6]	RW	Position Counter Carry Underflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
PCCOVF	[5]	RW	Position Counter Carry Overflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
PCTOVF	[4]	RW	Position Capture Timer Overflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
PCAPT	[3]	RW	Position Capture Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
PCSUNF	[2]	RW	Position Counter Sign Underflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
PCSOVF	[1]	RW	Position Counter Sign Overflow Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b
PCMAT	[0]	RW	Position Counter Match Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (enables the interrupt)	0'b

NOTE: On a Read, PPD_IMSCR register gives the current value of mask on the relevant interrupt.
A Write of 1 to a particular bit, sets the mask and enables the interrupt to be read. A Write of 0 to a particular bit, clears the corresponding mask.

7.3.1.8 PPD_RISR

- Base Address: 0x400C_0000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD															PHASEZ	RSVD	SCCUNF	SCCOVF	SCTOVF	SCAPT	SCSUNF	SCSOVF	SCMAT	RSVD	PCCUNF	PCCOVF	PCTOVF	PCAPT	PCSUNF	PCSOVF	PCMAT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	R	Reserved	0
PHASEZ	[16]	R	PHASEZ Raw Interrupt State Gives the raw interrupt state (prior to masking) of PHASEZ interrupt.	0'b
RSVD	[15]	R	Reserved	0
SCCUNF	[14]	R	Speed Counter Carry Underflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of SCCUNF interrupt.	0'b
SCCOVF	[13]	R	Speed Counter Carry Overflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of SCCOVF interrupt.	0'b
SCTOVF	[12]	R	Speed Capture Timer Overflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of SCTOVF interrupt.	0'b
SCAPT	[11]	R	Speed Capture Raw Interrupt State Gives the raw interrupt state (prior to masking) of SCAPT interrupt.	0'b
SCSUNF	[10]	R	Speed Counter Sign Underflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of SCSUNF interrupt.	0'b
SCSOVF	[9]	R	Speed Counter Sign Overflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of SCSOVF interrupt.	0'b
SCMAT	[8]	R	Speed Counter Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of SCMAT interrupt.	0'b
RSVD	[7]	R	Reserved	0

Name	Bit	Type	Description	Reset Value
PCCUNF	[6]	R	Position Counter Carry Underflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of PCCUNF interrupt.	0'b
PCCOVF	[5]	R	Position Counter Carry Overflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of PCCOVF interrupt.	0'b
PCTOVF	[4]	R	Position Capture Timer Overflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of PCTOVF interrupt.	0'b
PCAPT	[3]	R	Position Capture Raw Interrupt State Gives the raw interrupt state (prior to masking) of PCAPT interrupt.	0'b
PCSUNF	[2]	R	Position Counter Sign Underflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of PCSUNF interrupt.	0'b
PCSOVF	[1]	R	Position Counter Sign Overflow Raw Interrupt State Gives the raw interrupt state (prior to masking) of PCSOVF interrupt.	0'b
PCMAT	[0]	R	Position Counter Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of PCMAT interrupt.	0'b

NOTE: On a Read, PPD_RISR gives the current raw status value of the corresponding interrupt prior to masking. A Write has no effect.

- Sign overflow bit is set when counter value is changed from 0x7FFF to 0x8000
- Sign underflow bit is set when counter value is changed from 0x8000 to 0x7FFF
- Carry overflow bit is set when counter value is changed from 0xFFFF to 0x0000
- Carry underflow bit is set when counter value is changed from 0x0000 to 0xFFFF

7.3.1.9 PPD_MISR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD															PHASEZ	RSVD	SCCUNF	SCCOVF	SCTOVF	SCAPT	SCSUNF	SCSOVF	SCMAT	RSVD	PCCUNF	PCCOVF	PCTOVF	PCAPT	PCSUNF	PCSOVF	PCMAT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	R	Reserved	0
PHASEZ	[16]	R	PHASEZ Masked Interrupt State Gives the masked interrupt state (prior to masking) of PHASEZ interrupt.	0'b
RSVD	[15]	R	Reserved	0
SCCUNF	[14]	R	Speed Counter Carry Underflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of SCCUNF interrupt.	0'b
SCCOVF	[13]	R	Speed Counter Carry Overflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of SCCOVF interrupt.	0'b
SCTOVF	[12]	R	Speed Capture Timer Overflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of SCTOVF interrupt.	0'b
SCAPT	[11]	R	Speed Capture Masked Interrupt State Gives the masked interrupt state (prior to masking) of SCAPT interrupt.	0'b
SCSUNF	[10]	R	Speed Counter Sign Underflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of SCSUNF interrupt.	0'b
SCSOVF	[9]	R	Speed Counter Sign Overflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of SCSOVF interrupt.	0'b
SCMAT	[8]	R	Speed Counter Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of SCMAT interrupt.	0'b
RSVD	[7]	R	Reserved	0

Name	Bit	Type	Description	Reset Value
PCCUNF	[6]	R	Position Counter Carry Underflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of PCCUNF interrupt.	0'b
PCCOVF	[5]	R	Position Counter Carry Overflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of PCCOVF interrupt.	0'b
PCTOVF	[4]	R	Position Capture Timer Overflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of PCTOVF interrupt.	0'b
PCAPT	[3]	R	Position Capture Masked Interrupt State Gives the masked interrupt state (prior to masking) of PCAPT interrupt.	0'b
PCSUNF	[2]	R	Position Counter Sign Underflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of PCSUNF interrupt.	0'b
PCSOVF	[1]	R	Position Counter Sign Overflow Masked Interrupt State Gives the masked interrupt state (prior to masking) of PCSOVF interrupt.	0'b
PCMAT	[0]	R	Position Counter Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of PCMAT interrupt.	0'b

NOTE: On a Read, PPD_MISR register gives the current masked status value of the corresponding interrupt. A Write has no effect.

7.3.1.10 PPD_ICR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0024, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD															PHASEZ	RSVD	SCCUNF	SCCOVF	SCTOVF	SCAPT	SCSUNF	SCSOVF	SCMAT	RSVD	PCCUNF	PCCOVF	PCTOVF	PCAPT	PCSUNF	PCSOVF	PCMAT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	R	W	W	W	W	W	W	W	R	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	R	Reserved	0
PHASEZ	[16]	W	PHASEZ Interrupt Clear 0 = No effect 1 = Clears PHASEZ interrupt	0'b
RSVD	[15]	R	Reserved	0
SCCUNF	[14]	W	Speed Counter Carry Underflow Interrupt Clear 0 = No effect 1 = Clears SCCUNF interrupt	0'b
SCCOVF	[13]	W	Speed Counter Carry Overflow Interrupt Clear 0 = No effect 1 = Clears SCCOVF interrupt	0'b
SCTOVF	[12]	W	Speed Capture Timer Overflow Interrupt Clear 0 = No effect 1 = Clears SCTOVF interrupt	0'b
SCAPT	[11]	W	Speed Capture Interrupt Clear 0 = No effect 1 = Clears SCAPT interrupt	0'b
SCSUNF	[10]	W	Speed Counter Sign Underflow Interrupt Clear 0 = No effect 1 = Clears SCSUNF interrupt	0'b
SCSOVF	[9]	W	Speed Counter Sign Overflow Interrupt Clear 0 = No effect 1 = Clears SCSOVF interrupt	0'b
SCMAT	[8]	W	Speed Counter Match Interrupt Clear 0 = No effect 1 = Clears SCMAT interrupt	0'b
RSVD	[7]	R	Reserved	0

Name	Bit	Type	Description	Reset Value
PCCUNF	[6]	W	Position Capture Carry Underflow Interrupt Clear 0 = No effect 1 = Clears PCCUNF interrupt	0'b
PCCOVF	[5]	W	Position Counter Carry Overflow Interrupt Clear 0 = No effect 1 = Clears PCCOVF interrupt	0'b
PCTOVF	[4]	W	Position Capture Timer Interrupt Clear 0 = No effect 1 = Clears PCTOVF interrupt	0'b
PCAPT	[3]	W	Position Capture Interrupt Clear 0 = No effect 1 = Clears PCAPT interrupt	0'b
PCSUNF	[2]	W	Position Capture Sign Underflow Interrupt Clear 0 = No effect 1 = Clears PCSUNF interrupt	0'b
PCSOVF	[1]	W	Position Counter Sign Overflow Interrupt Clear 0 = No effect 1 = Clears PCSOVF interrupt	0'b
PCMAT	[0]	W	Position Counter Match Interrupt Clear 0 = No effect 1 = Clears PCMAT interrupt	0'b

A Write of 1, clears the corresponding interrupt. A Write of 0 has no effect.

NOTE:

1. Position Counter Sign/Carry Overflow (PCSOVF, PCCOVF) and Position Counter Sign/Carry Underflow (PCSUNF, PCCUNF) will be cleared automatically by PHASEZ and PCCL (CR1.2).
2. Speed Counter Sign/Carry Overflow (SCSOVF/SCCOVF) and Speed Counter Sign/Carry Underflow (SCSUNF/SCCUNF) will be cleared automatically by SCCL (CR1.18).

7.3.1.11 PPD_PCR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PCV	[15:0]	RW	Position Counter Value This field contains the current position counter value.	0x0000

7.3.1.12 PPD_PCRR

- Base Address: 0x400C_0000
- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PREFDAT	[15:0]	RW	Position Counter Reference Data Value This field determines the reference value for position counter.	0x0000

7.3.1.13 PPD_PCTR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																PCAPTV															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PCAPTV	[15:0]	RW	Position Capture Timer Value This field contains the current value of position timer.	0x0000

7.3.1.14 PPD_PCTVR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0034, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																PCTV															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PCTV	[15:0]	RW	Position Capture Timer Value This field contains the captured value of position capture timer.	0x0000

7.3.1.15 PPD_SCR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0038, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
SCV	[15:0]	RW	Speed Counter Value This field contains the current speed counter value.	0x0000

7.3.1.16 PPD_SCRR

- Base Address: 0x400C_0000
- Address = Base Address + 0x003C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
SREFDAT	[15:0]	RW	Speed Counter Reference Data Value This field determines the reference value for speed counter.	0x0000

7.3.1.17 PPD_SCTR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0040, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
SCAPTV	[15:0]	RW	Speed Capture Timer Value This field contains the current speed capture timer value.	0x0000

7.3.1.18 PPD_SCTVR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0044, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
SCTV	[15:0]	RW	Speed Capture Timer Value This field contains the captured value of speed capture timer.	0x0000

7.3.1.19 PPD_PCHR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0048, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PCHDAT	[15:0]	RW	Position Counter Hold Data This field contains the position counter value which will be copied by holding the event trigger.	0x0000

7.3.1.20 PPD_PCTHR

- Base Address: 0x400C_0000
- Address = Base Address + 0x004C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PCTVHDAT	[15:0]	RW	Position Capture Timer Hold Data This field contains the position capture timer value which will be copied by holding the event trigger.	0x0000

7.3.1.21 PPD_PCTVHR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0050, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																PCTVDAT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PCVTVDAT	[15:0]	RW	Position Capture Timer Value Hold Data Hold data register for position capture timer value.	0x0000

7.3.1.22 PPD_SCHR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0054, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
SCHDAT	[15:0]	RW	Speed Counter Hold Data This field contains the speed counter value copies it by holding the event trigger.	0x0000

7.3.1.23 PPD_SCTHR

- Base Address: 0x400C_0000
- Address = Base Address + 0x0058, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
SCTHDAT	[15:0]	RW	Speed Capture Timer Hold Data This field contains the speed capture timer value copies it by holding the event trigger.	0x0000

7.3.1.24 PPD_SCTVHR

- Base Address: 0x400C_0000
- Address = Base Address + 0x005C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
SCTVHDAT	[15:0]	RW	Speed Capture Timer Hold Data Hold data register for speed capture timer value	0x0000

8 General Purpose I/O (GPIO)

8.1 Overview

General Purpose I/O chapter describes the configuration of General Purpose I/O (GPIO), such as input and output.

S3FN429 has 32 ports as the GPIO. Each port is one of Port 0 group, P0.[31:0]. All I/O lines in the microcontroller are unified in GPIO controller. This GPIO controller module controls all I/O lines. GPIO controller also provides interrupt signals to the interrupt controller.

You can configure each port by software to fulfill various configuration requirements of target system and design. You should define the functionality of the port before starting an application program. If you do not want to use the function for multiplexed pins, you can configure these pins as simple I/O ports.

8.1.1 Features

- Clock Supply Enable/Disable
  - Configuration of GPIO requires a clock supply
  - You can disable the clock supply to optimize the power consumption
- Output (Data Direction) Enable/Disable/Status Monitoring
- Output Data Set/Clear/Data Status Monitoring
- Software Reset (SWRST) Function
  - Set to default configuration
- GPIO Interrupt
  - Mask Set and Clear/Raw Interrupt Status/Masked Interrupt Status/Clear Interrupt

8.1.2 Pin Description

Table 8-1 describes GPIO pin description.

Table 8-1 GPIO Pin Description

Pin Name	Function	I/O Type	Active Level	Comments
P0.[31:0]	General Purpose Input Output	I/O	–	Reset Input Status

NOTE: P0.30 and P0.31 are defined as XOUT and XIN function by fabrication. But you can change the function by using Smart Option program. (Refer to IFC chapter).

8.2 Functional Description

This section describes GPIO Configuration, Input Configuration, Output Configuration, Operation Mode, and Interrupt.

8.2.1 GPIO Configuration

Pins have their multiplexed functions from 0 to 3. IOCONF defines these functions. After IOCONF has defined multiplexed pins as GPIO pins, you should configure the GPIO. Therefore, it is necessary to control IOCONF before using the GPIO.

Each of 32 bits of control registers, such as enable, disable, and status registers is corresponding to individual I/O ports. You can either configure I/O port individually or you can configure the entire port by writing a value to the 32 bits register.

Before configuring I/O ports, it is mandatory to enable the clock supply to the I/O ports. The registers responsible for clock supply are:

- Clock Enable/Disable Register (GPIO_CEDR)
- Software Reset Register (GPIO_SRR)

It is also possible to reset each of GPIO blocks and recall the default values. For example, you can reset PIO0 using the GPIO_SRR.

8.2.2 Input Configuration

When you use the input function of I/O port, the output function of I/O port disables. The default of I/O port is the input function. When the I/O port is an input function, an external condition decides the level status of I/O. You can read the level status of I/O through GPIO_PDSR register.

To configure the ports as input, the registers are:

- Output Disable Register (GPIO_ODR)
- Pin Data Status Register (GPIO_PDSR)

8.2.3 Output Configuration

The data direction of I/O ports should be configured as output. This data can be written to the ports.

To configure the data direction, enable/disable/monitor, and data value the registers used are:

- Output Enable Register (GPIO_OER)
- Write Output Data Register (GPIO_WODR)
- Set Output Data Register (GPIO_SODR)
- Clear Output Data Register (GPIO_CODR)
- Output Data Status Register (GPIO_ODSR)
- Pin Data Status Register (GPIO_PDSR)

GPIO_WODR is to set the data value to I/O ports in accordance with the register value (high or low level). On the contrary, GPIO_SODR and GPIO_CODR set or clear the data value of I/O ports according to the register value. In other words, GPIO_SODR sets the I/O ports to high level when values are written to the register. GPIO_CODR sets the I/O ports to low level when values are written to the register. Either GPIO_ODSR or GPIO_PDSR can read the current data value of each pin.

8.2.4 Operation Mode

GPIO has different configurations and behaviors according to the operation modes.

8.2.4.1 Normal Mode

In a normal mode, the GPIO is powered, operational, and configurable. The GPIO control block is clocked by PCLK.

8.2.4.2 Low Power Modes

In an idle mode, the GPIO is powered on and clocked by PCLK. It is configurable. The PCLK is also allowed to disconnect the clock to reduce the power consumption. After exiting from an idle mode, the configuration and states of I/Os are not changed.

In a stop mode, the clock is disconnected, but the power is still maintained. Therefore, the configuration and states of I/Os are preserved by itself. After exiting from stop mode, the configuration and states of I/Os do not change.

8.2.5 Interrupt

Each GPIO controller block also provides an internal interrupt signal. When a rising edge or falling edge level change occurs, you can program each GPIO to generate an interrupt

The interrupt occurs regardless of the port configuration and data direction. It occurs when an edge transition occurs on a pin while the port is configured for the peripherals or output. The interrupt detects any edge transition.

The GPIO interrupt registers are:

- Interrupt Mask Set and Clear Register (GPIO_IMSCR)
- Raw Interrupt Status Register (GPIO_RISR)
- Masked Interrupt Status Register (GPIO_MISR)
- Interrupt Clear Register (GPIO_ICR)

Corresponding bits in the GPIO_IMSCR enables or disables the interrupt for a pin.

When an edge transition occurs on a pin, the corresponding bit in the GPIO_RISR register is set to "1". If a bit in GPIO_IMSCR register is set to "1", then an interrupt for the pin is enabled.

The PIO interrupt is cleared when a "1" is set to a corresponding bit of GPIO_ICR register.

NOTE: The interrupt registers mentioned above are different types of interrupt from the external interrupt in the Clock Manager (CM).

8.3 Register Description

8.3.1 Register Map Summary

- Base Address: 0x4005_0000

Register	Offset	Description	Reset Value
GPIO_IDR	0x0000	ID Code register	0x0001_0020
GPIO_CEDR	0x0004	Clock enable/disable register	0x0000_0000
GPIO_SRR	0x0008	Software reset register	0x0000_0000
GPIO_IMSCR	0x000C	Interrupt mask set/clear register	0x0000_0000
GPIO_RISR	0x0010	Raw interrupt status register	0x0000_0000
GPIO_MISR	0x0014	Masked interrupt status register	0x0000_0000
GPIO_ICR	0x0018	Interrupt clear register	0x0000_0000
GPIO_OER	0x001C	Output enable register	0x0000_0000
GPIO_ODR	0x0020	Output disable register	0x0000_0000
GPIO_OSR	0x0024	Output status register	0x0000_0000
GPIO_WODR	0x0028	Write output data register	0x0000_0000
GPIO_SODR	0x002C	Set output data register	0x0000_0000
GPIO_CODR	0x0030	Clear output data register	0x0000_0000
GPIO_ODSR	0x0034	Output data status register	0x0000_0000
GPIO_PDSR	0x0038	Pin data status register	0x0000_0000

8.3.1.1 GPIO_IDR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0000, Reset Value = 0x0001_0020

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	Identification Code Register This field stores the ID code for the corresponding IP.	0x0001_0020

8.3.1.2 GPIO_CEDR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable/Disable 0 = Disables GPIO Clock 1 = Enables GPIO Clock GPIO_SRR does not affect CLKEN.	0

8.3.1.3 GPIO_SRR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Performs GPIO Software Reset and auto-clears	0

8.3.1.4 GPIO_IMSCR

- Base Address: 0x4005_0000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
Py	[y]	RW	Port y Interrupt Mask 0 = This interrupt is masked (Disables the interrupt) 1 = This interrupt is not masked (Enables the interrupt) Interrupt occurs when a logic level change is detected on the corresponding pin.	0

NOTE: On a Read, GPIO_IMSCR register gives the current value of the mask on the relevant interrupt. A Write of 1 to the particular bit sets the mask, enabling the interrupt to be read. A Write of 0, clears the corresponding mask.

8.3.1.5 GPIO_RISR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
Py	[y]	R	Port y Raw Interrupt State Gives the raw interrupt state (prior to masking) of the Py interrupt.	0

NOTE: On a Read, GPIO_RISR register gives the current raw status value of the corresponding interrupt prior to masking.

8.3.1.6 GPIO_MISR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
Py	[y]	R	Port y Masked Interrupt State Gives the masked interrupt status (after masking) of the Py interrupt.	0

NOTE: On a Read, GPIO_MISR register gives the current masked status value of the corresponding interrupt. A Write has no effect.

8.3.1.7 GPIO_ICR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
Py	[y]	W	Port y Interrupt Clear 0 = No effect 1 = Clears Py interrupt	0

NOTE: On a Write of 1, the corresponding interrupt clears. A Write of 0 has no effect.

8.3.1.8 GPIO_OER

- Base Address: 0x4005_0000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
Py	[y]	W	Port y Output Enable Bit 0 = No effect 1 = Enables the GPIO output (data direction) on the corresponding pin	0

8.3.1.9 GPIO_ODR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Name	Bit	Type	Description	Reset Value
Py	[y]	W	Port y Output Disable Bit (Input Enable) 0 = No effect 1 = Disables the GPIO output (data direction) on the corresponding pin In other words, GPIO_ODR enables GPIO input on the corresponding pin.	0

8.3.1.10 GPIO_OSR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0024, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
Py	[y]	R	Port y Data Direction (Input or Output) Status 0 = The corresponding GPIO is input on this line. 1 = The corresponding GPIO is output on this line.	0

8.3.1.11 GPIO_WODR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
Py	[y]	W	Port y Output Data Control Bit 0 = Programs GPIO output data on the corresponding pin to 0, Low Level. 1 = Programs GPIO output data on the corresponding pin to 1, High Level. The purpose of GPIO_WODR register is similar to GPIO_SODR and GPIO_CODR. But, the output data (1 and 0) affects it at the same time. This function differs from the GPIO_SODR and GPIO_CODR.	0

8.3.1.12 GPIO_SODR

- Base Address: 0x4005_0000
- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Name	Bit	Type	Description	Reset Value
Py	[y]	W	Port y Output Data Set (1) 0 = No effect 1 = Sets GPIO output data on the corresponding pin You can read the result through GPIO_PDSR register.	0

8.3.1.13 GPIO_CODR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Name	Bit	Type	Description	Reset Value
Py	[y]	W	Port y Output Data Clear (0) 0 = No effect 1 = Clears GPIO output data on the corresponding pin You can read the result through GPIO_PDSR register.	0

8.3.1.14 GPIO_ODSR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0034, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
Py	[y]	R	Port y Output Data Status 0 = GPIO output data for the corresponding pin is programmed to 0, Low Level. 1 = GPIO output data for the corresponding pin is programmed to 1, High Level.	0

8.3.1.15 GPIO_PDSR

- Base Address: 0x4005_0000
- Address = Base Address + 0x0038, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16	P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
Py	[y]	R	Port y Pin Status 0 = The real level of corresponding pin is at logic 0. 1 = The real level of corresponding pin is at logic 1.	0

9 Internal Flash Controller (IFC)

9.1 Overview

S3FN429 has an on-chip program flash ROM, internally. The flash memory size is 32 KB.

9.1.1 Features

The features of the flash memory are:

- Flash memory size = 32 KB
- Program size = word (32-bit)
- Page size = 256 Bytes
- Sector size = 8 KB
- Erase unit = Page or Sector
- Number of sectors = 4 sectors
- Number of pages = 128 pages
- Program/Ease Cycle (Endurance) = 10,000
- Protection supports = Serial Wire Debug (SWD) interface protection, Hardware protection, and Read protection.

Table 9-1 describes flash configuration.

Table 9-1 Flash Configuration

Parameter	Description
Size of flash	32 KB
Size of page	256 Bytes
Number of pages	128 pages
Size of sector	8 KB
Number of sectors	4 sectors
Boot sector	Sector 0 (8 KB)

9.2 Functional Description

Figure 9-1 illustrates the non pre-fetch flash block diagram.

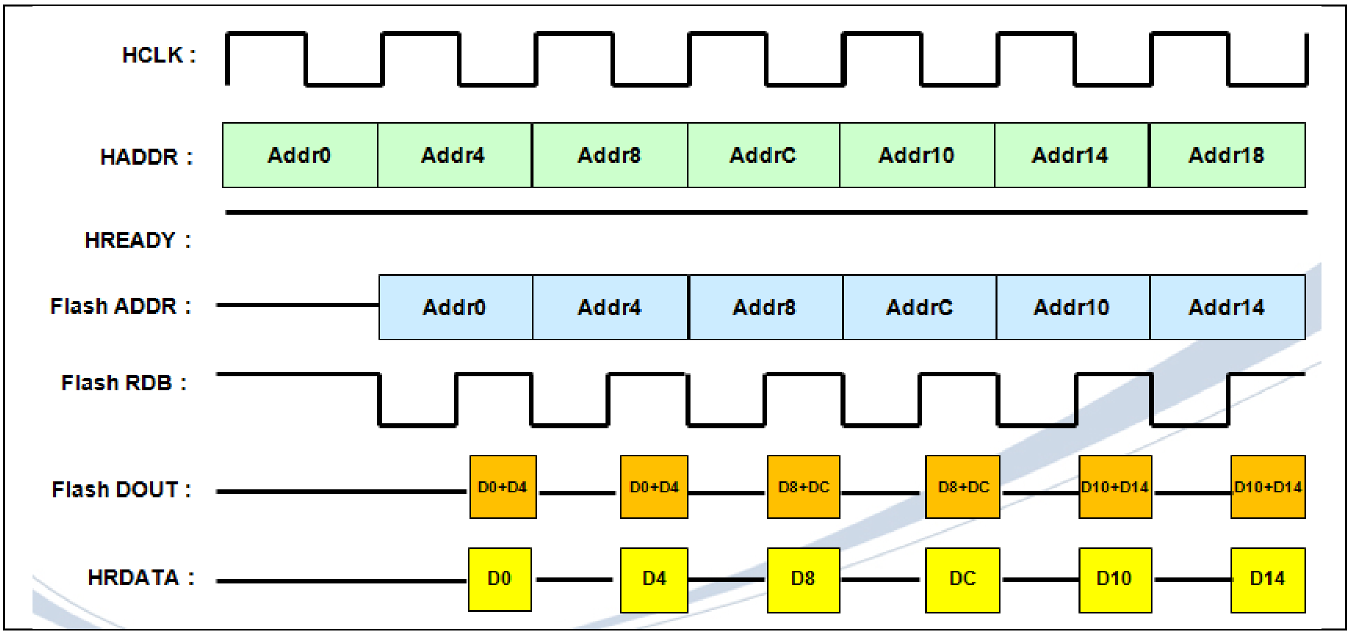


Figure 9-1 Flash Block Diagram (Non Pre-Fetch)

Figure 9-2 illustrates the pre-fetch flash block diagram.

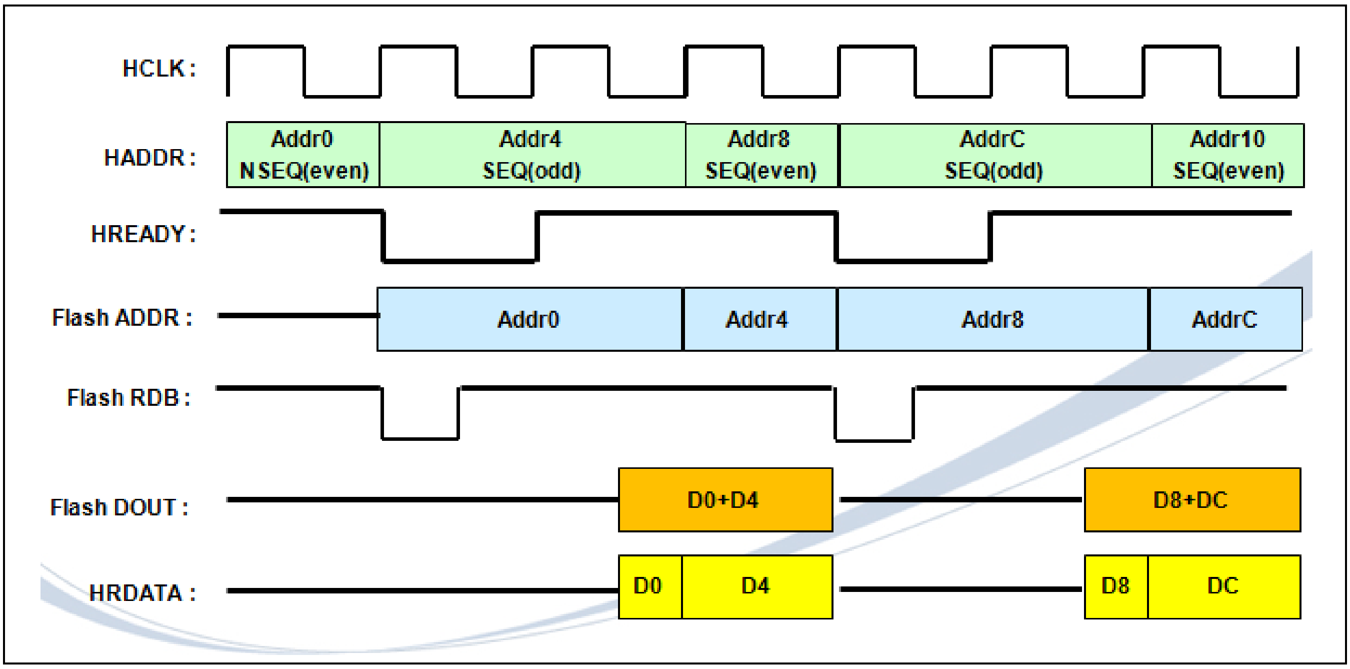


Figure 9-2 Flash Block Diagram (Pre-Fetch)

9.2.1 Organization

This section describes Flash Configuration and Address Alignment in detail.

9.2.1.1 Flash Configuration

The flash ROM consists of four sectors. Each sector consists of 8 KB. Therefore, the total size of flash ROM is 4 (sector number) × 8 KB (each sector size) = 32 KB. One sector is made up of 32 pages. *Figure 9-3* illustrates the physical configuration.

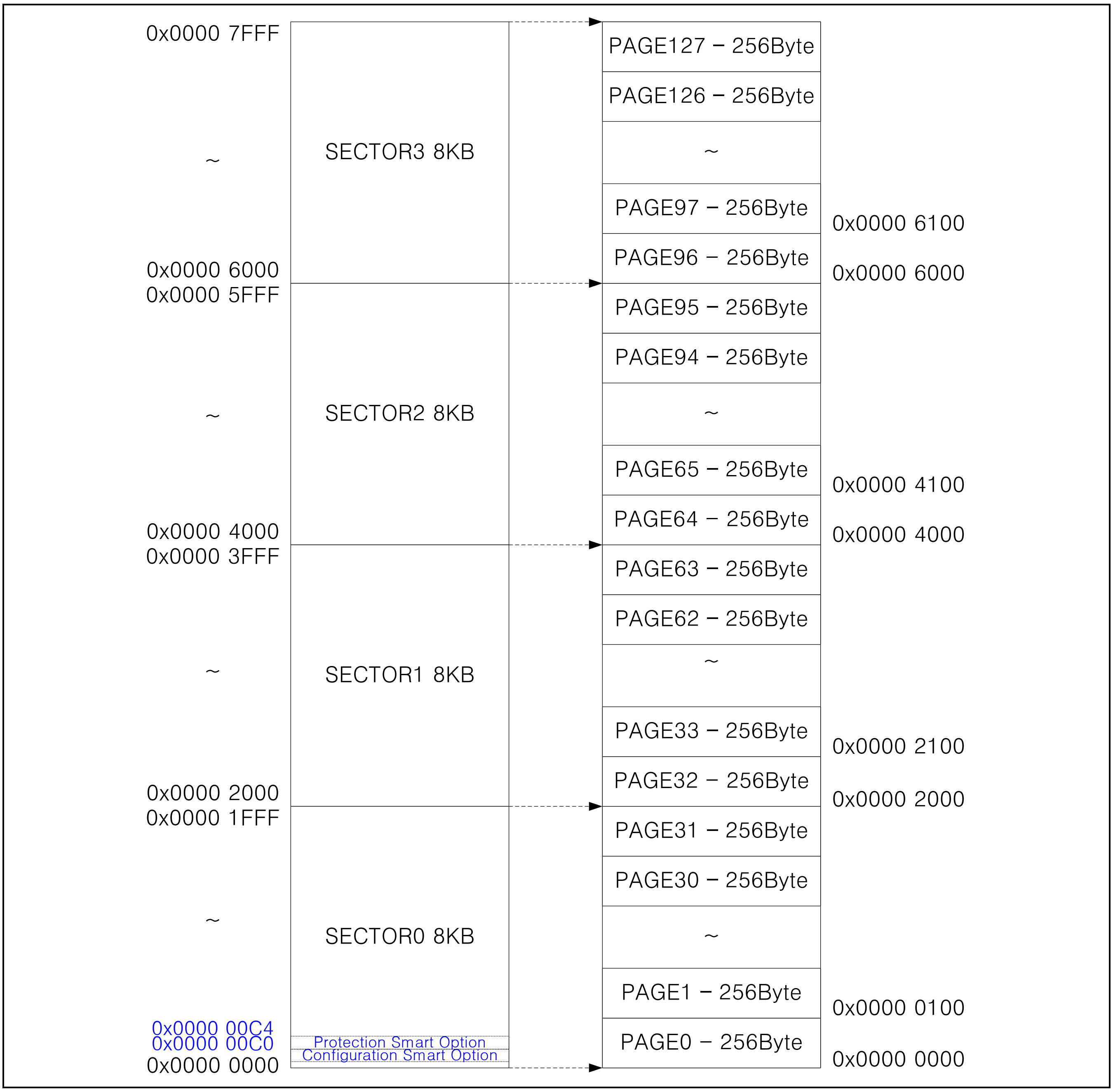


Figure 9-3 Physical Configuration

9.2.1.2 Address Alignment

This sub-section describes the address alignment method to set an address value in IFC_AR Register, according to each operation.

9.2.1.2.1 Program

To write data to flash you should set lower 2 bits to 0. To write data to flash by word-unit (4 bytes) you should set lower 2 bits to 0 during flash programming. You can select address from the range 0x0000 to 0x7FFF in 32 KB. In the tool program, the low 2-bit address is also 00b.

- IFC_AR.ADDR[31:0] = ADDRESS & 0xFFFFFFFC

9.2.1.2.2 Sector Erase

When erasing a sector, you should set lower 13 bits of address to 0, because the size of a sector is 8 KB. You can select a value from 0 to 3 as the SECTOR_ORDER, from 4 sectors.

- IFC_AR.ADDR[31:0] = (SECTOR_ ORDER << 13)
  - Sector 0= 0x0000_0000 to 0x0000_1FFF → ADDR[31:0] = 0x00000000 = 0'b << 13
  - Sector 1= 0x0000_2000 to 0x0000_3FFF → ADDR[31:0] = 0x00002000 = 1'b << 13
  - Sector 2= 0x0000_4000 to 0x0000_5FFF → ADDR[31:0] = 0x00004000 = 2'b << 13
  - Sector 3= 0x0000_6000 to 0x0000_7FFF → ADDR[31:0] = 0x00006000 = 3'b << 13

Table 9-2 describes the base address of each sector.

Table 9-2 Base Address of Each Sector

Sector Number	Base Address
0	0x0000_0000
1	0x0000_2000
2	0x0000_4000
3	0x0000_6000

9.2.1.2.3 Page Erase

When erasing a page, you should set lower 8 bits of address to 0, because the size of a page is 256 B. You can select a value from 0 to 127 as the PAGE_ORDER, from 128 pages.

Table 9-3 describes the base address of each page.

Table 9-3 Base Address of Each Page

Page Number	Page Base Address	Page Number	Page Base Address
0	0x0000_0000	17	0x0000_1100
1	0x0000_0100	18	0x0000_1200
2	0x0000_0200	19	0x0000_1300
3	0x0000_0300	20	0x0000_1400
4	0x0000_0400	21	0x0000_1500
5	0x0000_0500	22	0x0000_1600
6	0x0000_0600	23	0x0000_1700
7	0x0000_0700	24	0x0000_1800
8	0x0000_0800	25	0x0000_1900
9	0x0000_0900	26	0x0000_1A00
10	0x0000_0A00	27	0x0000_1B00
11	0x0000_0B00	28	0x0000_1C00
12	0x0000_0C00	29	0x0000_1D00
13	0x0000_0D00	30	0x0000_1E00
14	0x0000_0E00	31	0x0000_1F00
15	0x0000_0F00	32	0x0000_2000
16	0x0000_1000	…	…

9.2.2 Smart Option

There are two kinds of Smart Options. One is "Protection" Smart Option. Another is "Configuration" Smart Option.

The flash memory addresses for Smart Options

- Configuration Smart Option= 0x0000_00C0
- Protection Smart Option= 0x0000_00C4

9.2.2.1 Configuration Smart Option

The Configuration Smart Option includes four hardware configuration values. If you want to change the configuration, you can do that by Smart Option Erase and Smart Operation Program. The Configuration Smart Option value should be configured at address 0x0000_00C0.

The four kinds of configuration are:

- POCCS Configuration: CPU clock selection at reset time
- XIN Pin Configuration: Selection between XIN and P0.31 (general function pin)
- XOUT Pin Configuration: Selection between XOUT and P0.30 (general function pin)
- BT Divider Configuration: Basic timer divider value selection

Table 9-5 describes the Configuration Smart Option address and control bits.

Table 9-4 Configuration Smart Option Address and Control Bits

IFC_AR	Bit	Description	Reset Value
0x0000_00C0	Bit[0]	Power-On CPU (system) Clock Selection 0 = External Main Clock (EMCLK) 1 = Internal Main Clock (IMCLK)	1
	Bit[2]	XIN Pin Configuration Bit 0 = P0.31 as the general function pin 1 = XIN as the system pin	1
	Bit[3]	XOUT Pin Configuration Bit 0 = P0.30 as the general function pin. 1 = XOUT as the system pin	1
	Bit[15:12]	Basic Timer Divider Value Configuration Bit 0000, 0001, 0010 = Not used 0011 = /1 0100 = /2 0101 = /4 0110 = /8 0111 = /16 1000 = /32 1001 = /64 1010 = /128 1011 = /256 1100 = /512 1101 = /1024 1110 = /2048 1111 = /4096	0xF

9.2.2.1.1 Power-On CPU (system) Clock Configuration Bit

When chip reset occurs, CPU runs with the clock to be defined by this smart option bit. The initial value to be programmed on factory is IMCLK (Internal Main Clock). If it is changed from IMCLK to EMCLK, external clock pin configuration (XIN or XIN/XOUT) should be done together.

9.2.2.1.2 External Main Clock Input and Output Pin Configuration Bits

Pin7 and Pin8 are decided between the system pin (XOUT, XIN) and the function pin (P0.30/-/USARTTX0/EXI22, P0.31/EXI23/USARTRX0/PWM0) by smart option. That means they are selected by hardware configuration at reset time. If they are defined as function pins at reset time, you can use these pines as one function among 4 functions by software (register control) after reset. If they are defined as system pins at reset time, it can be used for the external clock only. The initial value to be programmed on factory is the system pin (XOUT, XIN).

9.2.2.1.3 Basic Timer Divider Configuration Bits

Internal reset generation circuit uses the basic timer to obtain the stabilization time of system. The time comprised by basic timer in reset time is the 256 counts by basic timer. If the basic timer divider value is increased, the reset time will be longer. The default divider value, 4096, is enough to make the chip reset under the stabilization for internal logic. If you need to reduce the reset time, you can do it by changing BT divider value of smart option. Refer to the CM (Clock Manager) chapter about basic timer related to the reset time.

9.2.2.2 Protection Smart Option

In some cases, you might need to protect the data or code programmed in the flash memory. You should protect the data or code programmed in flash memory. The internal flash memory controller supports three kinds of protection mechanism to protect data. You can control it by programming the Protection Smart Option bit. The Protection Smart Option value should be configured at address 0x0000_00C4.

The three kinds of protection mechanism are:

- HARDWARE (HARDLOCK) Protection: Protection for selected regions among four regions or full region.
- READ Protection: Flash Read protection for serial interface.
- SWD Protection: Flash Read protection through SWD interface

Table 9-5 describes the Protection Smart Option address and the protection bits.

Table 9-5 Protection Smart Option Address and Control Bits

IFC_AR	Bit	Description		Reset Value
0x0000_00C4	Bit[8]	0 = Enables SWD Protection 1 = Disables SWD Protection	–	1
	Bit[17]	0 = Enables Hardware Protection 1 = Disables Hardware Protection	(NOTE)	1
	Bit[27]	0 = Enables Read Protection 1 = Disables Read Protection	–	1
	Bit[4]	Hardware Protection Region Selection Bits Each of these bits is mapped to a corresponding region that is composed of four sectors (8 KB). 0 = Enables HARDWARE protection of selected regions. 1 = Disables HARDWARE protection of selected regions.	Sector 0	1
	Bit[5]		Sector 1	1
	Bit[6]		Sector 2	1
	Bit[7]		Sector 3	1
	Others	Should be "1"	–	–

NOTE: You should define the region or regions for protection to enable hardware protection and then perform smart option program. The hardware protection data in Protection Smart Option should include Hardware Protection Bit (Bit [17]) and hardware protection region bits.

9.2.2.2.1 SWD Interface Protection Bit [8]

Use SWD interface protection to enable or disable SWD access. If you do debug through SWD in initial chip development state, then you should disable SWD interface protection. But in final design development state, if you enable the SWD interface protection, then the other users cannot access the flash memory data through SWD interface.

9.2.2.2.2 Hardware (Hard-Lock) Protection Bit[17]

When the Hard-Lock protection bit is enabled, you cannot write or erase the data in selected regions of flash memory. You can protect the partial or all flash memory and unprotect it by Software. Tool can clear the protection using the entire erase.

9.2.2.2.3 Read Protection Bit [27]

If you want your data and code in memory to be inaccessible to others, you can do this by read protection .It prevents flash data from being read serially in the tool program mode .When you enable this function, then reading flash data with serial interface results in zero read-out.

9.2.3 Modes

This section describes User Mode and Tool Mode.

- User Mode
  - Read Operation
  - Page Erase Operation
  - Sector Erase Operation
  - Entire Erase Operation
  - Normal Program Operation
  - Smart Option Erase Operation
  - Smart Option Program Operation

- Tool Mode (Also called as "Serial Program Mode" (SPGM))
  - Flash Program Tool

9.2.3.1 User Mode

You can program or erase normal flash and smart option area in User Program Mode. You can configure User Program Mode by MODE1 and MODE 0. (Refer to "Pin Configuration" chapter). To write any data to normal flash area, you should clear the target address before program. There are three ways (page, sector, and entire erase) to clear the normal flash area. The flash area to be erased becomes all "1".

9.2.3.1.1 Read Operation

You can read and access 8-bit, 16-bit, or 32-bit.

9.2.3.1.2 Page Erase Operation

The size of a page is 256 B. The page erase operation erases one page to include address written into Address Register (IFC_AR).

To perform page erase, the steps are:

1. Write the value 0x5A5A5A5A into the IFC_KEY register.

2. Write the address for target page to the Address Register (IFC_AR)
   The address should be one in range of target page. Simply, use the base address of page.

3. Write the page erase command and start control bit to Control Register (IFC_CR)

4. Check whether the operation of page erase is completed or not.
   You can use one ways between polling and interrupt check. If you monitor the status bit of END, check END bit in Raw Interrupt Status Register (IFC_RISR).

9.2.3.1.3 Sector Erase Operation

A sector has 8 KB size. The sector erase operation erases one sector to include address written into d Address Register (IFC_AR).

To perform sector erase the steps are:

1. Write the value 0x5A5A5A5A into the IFC_KEY register.
2. Write the base address for target sector to the Address Register (IFC_AR)
   The address should be one in range of target sector. Simply, use the base address of sector.
3. Write the sector erase command and start control bit to Control Register (IFC_CR)
4. Check whether the operation of sector erase is completed or not.
   You can use one ways between polling and interrupt check. If you monitor the status bit of END, check END bit in Raw Interrupt Status Register (IFC_RISR).

9.2.3.1.4 Entire Erase Operation

Basically the entire erase initializes all memory area. But if BACEN bit in Mode Register (IFC_MR) is set to "1", the sector 0 is excluded from erase area. BACEN means Boot Area Configuration bit. When you include the specific code like as bootloader, you can use sector0 with Boot Area Configuration.

To perform entire erase the steps are:

1. Write the value 0x5A5A5A5A into the IFC_KEY register.
   If you require entire erase, then there is no need to select address and data.
2. Write the entire erase command and start control bit to Control Register (IFC_CR)
   You can use one ways between polling and interrupt check. If you monitor the status bit of END, check END bit in Raw Interrupt Status Register (IFC_RISR).

9.2.3.1.5 Normal Program Operation

The normal program operation writes one word (data or code) to the target address selected by IFC_AR register. That means the writing size is 4B. The data size that you can program is one word at a time. So the lower 2 bits of target address should be "0".

- IFC_AR[31:0] = ADDRESS and 0xFFFFFFFC

You can perform normal program of flash memory using software in SRAM or flash memory. The flash memory of S3FN429 is full flash, because you can program flash memory by code programmed in flash memory. Before program, target address should be initialized by any erase operation.

To program a normal flash memory the steps are:

1. Write the value 0x5A5A5A5A into the IFC_KEY register.
2. Write the address to be written into the Address Register (IFC_AR)
3. Write the data into the Data Register (IFC_DR)
4. Write the normal program command and start control bit in Control Register (IFC_CR)
5. Check whether the operation of normal program is completed or not.
   You can use one ways between polling and interrupt check. If you monitor the status bit of END, check END bit in Raw Interrupt Status Register (IFC_RISR).

9.2.3.1.6 Smart Option Erase Operation

You can change the Smart Option. You have to program the Smart Option with new option value. To change the smart option value (Configuration Smart Option or Protection Smart Option), you should clear the smart option area before program. Smart Option Erase erases and clears both Configuration Smart Option and Protection Smart Option. So you need to store the smart option value not to be changed and program again.

To perform the smart option erase the steps are:

1. Write the value 0x5A5A5A5A into the IFC_KEY register.
2. Write the smart option erase command and start control bit to Control Register (IFC_CR)
3. Check whether the operation of smart option erase is completed or not.
   You can use one ways between polling and interrupt check. If you monitor the status bit of END, check END bit in Raw Interrupt Status Register (IFC_RISR).

9.2.3.1.7 Smart Option Program Operation

All smart option is cleared by smart option erase. If both Configuration Smart option and Protection Smart Option don't have the initial value (0xFFFFFFFF) as the new value, you should do smart option program twice. When you program the new smart option value, all other bits except the smart option control bits (field) should be "1".

To perform the smart option program the steps are:

1. Write the value 0x5A5A5A5A into the IFC_KEY register.

2. Write the address to be written into the Address Register (IFC_AR)
   In case of Configuration Smart Option, the address is 0x000000C0.
   In case of Protection Smart Option, the address is 0x000000C4.

3. Write the smart option value into the Data Register (IFC_DR).

3. Write the smart option program command and start control bit in Control Register (IFC_CR)

4. Check whether the operation of smart option program is completed or not.
   You can use one ways between polling and interrupt check. If you monitor the status bit of END, check END bit in Raw Interrupt Status Register (IFC_RISR).

9.2.3.1.8 Interrupt

There are four interrupt sources related to the flash operation. *Table 9-6* describes the interrupt sources.

Table 9-6 Description of Interrupt Sources

END	Occurs when the command operation is completed. (CMD[2:0] filed in control register has the selected command.)
ERR0	Occurs when trying to execute other operation (program, erase) during operating "normal program" command (BUSY = 1)
ERR1	Occurs when writing the undefined value (CMD[2:0] = 111b) into CMD field of a control register during operation sequence
ERR2	Occurs when writing or erasing to the protected memory region

Sequence according to the each interrupt is:

- END
  - Enable END interrupt – IMSCR register
  - Configuration for a target operation – KEY, AR, or DR register
  - Start the operation by a command – CR register
  - END interrupt occurs – MISR register
  - Clear an END interrupt – ICR register
  - Finish

- ERR0
  - Enable ERR0 interrupt – IMSCR register
  - Configuration for a target operation – KEY, AR, or DR register
  - Start the operation by a command – CR register
  - ERR0 interrupt occurs – MISR register
  - Clear an ERR0 interrupt – ICR register
  - Wait END (status/interrupt) – RISR/MISR register
  - Clear an END status/interrupt – ICR register
  - Finish

- ERR1 or ERR2
  - Enable ERR1 or ERR2 interrupt – IMSCR register
  - Configuration for a target operation – KEY, AR, or DR register
  - Start the operation by a command – CR register
  - ERR1 or ERR2 interrupt occur – MISR register
  - Clear an ERR1 or ERR2 interrupt – ICR register
  - Finish

9.2.3.2 Tool Mode

Tool program mode is a flash memory program mode that uses an equipment tool such as AS-pro Flash ROM Writer. To program with Tool Program Mode, MODE1 and MODE0 should be set as Tool Mode. (Refer to "Pin Configuration" chapter.)

The microcontroller has several pins used for flash ROM Writer to Read/Write/Erase the flash memory (VDDIO, VSSIO, nRESET, VDDCORE, VSSCORE, F_SDAT, and F_SCLK).

Table 9-7 describes the pins used to Read/Write/Erase the flash ROM in Tool Program Mode.

Table 9-7 Pins Used to Read/Write/Erase the Flash ROM in Tool Program Mode

Signal	Pin Name	I/O	Function
F_SDAT	P0.14	I/O	Serial bi-directional DATA pin (Output when reading, Input when writing). Input and push-pull output port can be assigned.
F_SCLK	P0.13	I	Serial CLOCK input pin.
RESET	nRESET	I	Chip Initialization
VDD	VDDCORE	P	Power pin for flash block
	VDDIO		Power pin for I/O interface
VSS	VSSCORE	P	Power pin for flash block
	VSSIO		Power pin for I/O interface
	VDDCOREOUT	–	Connected to GND through a 0.1 μF capacitor (From internal regulator)
MODE	MODE[1:0]	–	Flash program mode using flash writing tool (serial interface) The value should be 01'b.

9.2.3.2.1 Flash Program Tool

There are several Flash Program Tools like a GW-uni2. If you want to make a dedicated Flash Program Tool (Flash ROM Writer), please contact www.seminix.com for more detail document.

9.2.4 Flow Chart

9.2.4.1 Normal Program

Figure 9-4 illustrates the Normal Program flowchart.

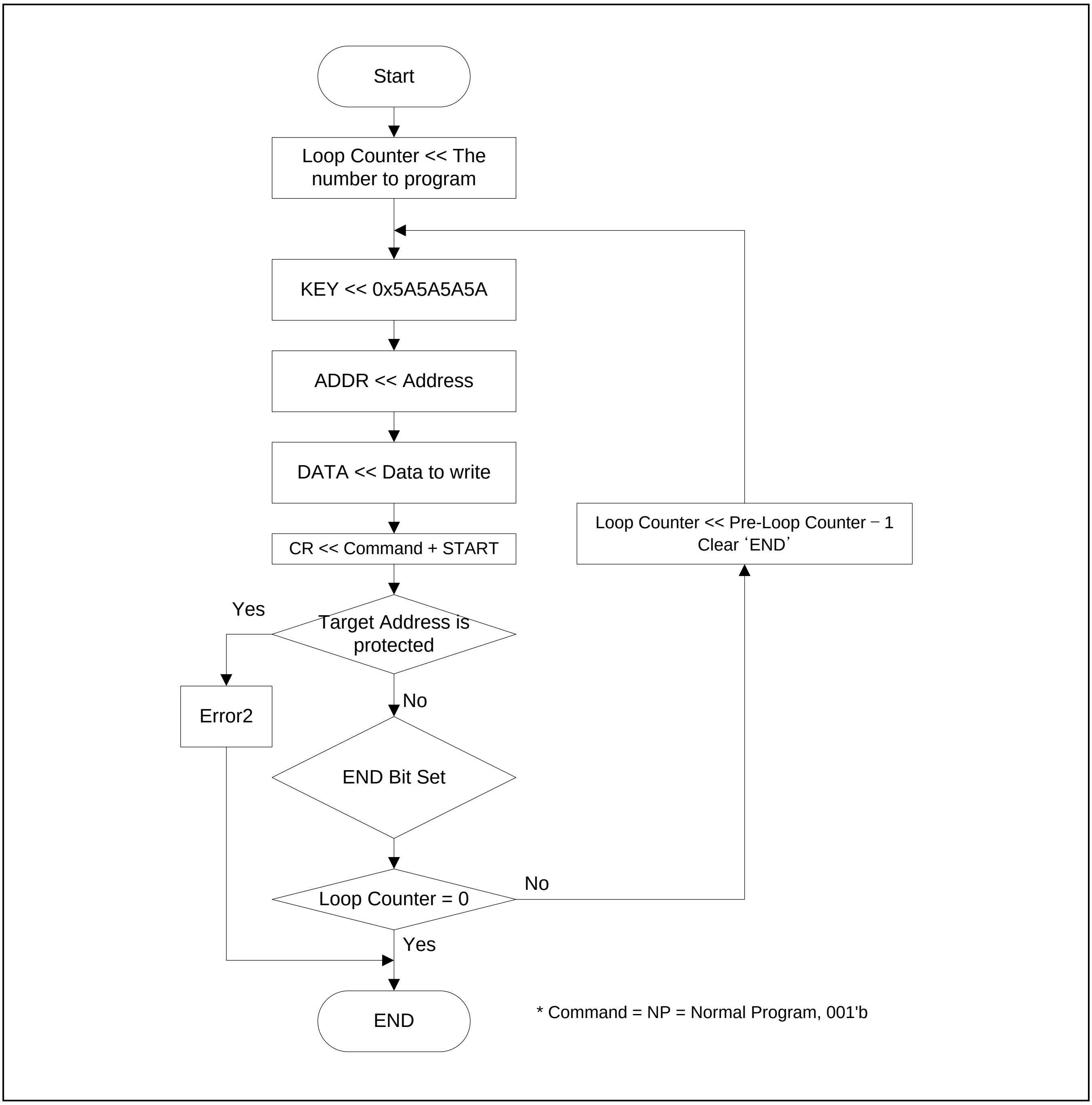

Start
Loop Counter << The number to program
KEY << 0x5A5A5A5A
ADDR << Address
DATA << Data to write
CR << Command + START
Loop Counter << Pre-Loop Counter – 1
Clear 'END'
Yes
Target Address is protected
Error2
No
END Bit Set
No
Loop Counter = 0
Yes
END
* Command = NP = Normal Program, 001'b

Figure 9-4 Normal Program Flowchart

9.2.4.2 Page Erase

Figure 9-5 illustrates the Page Erase flowchart.

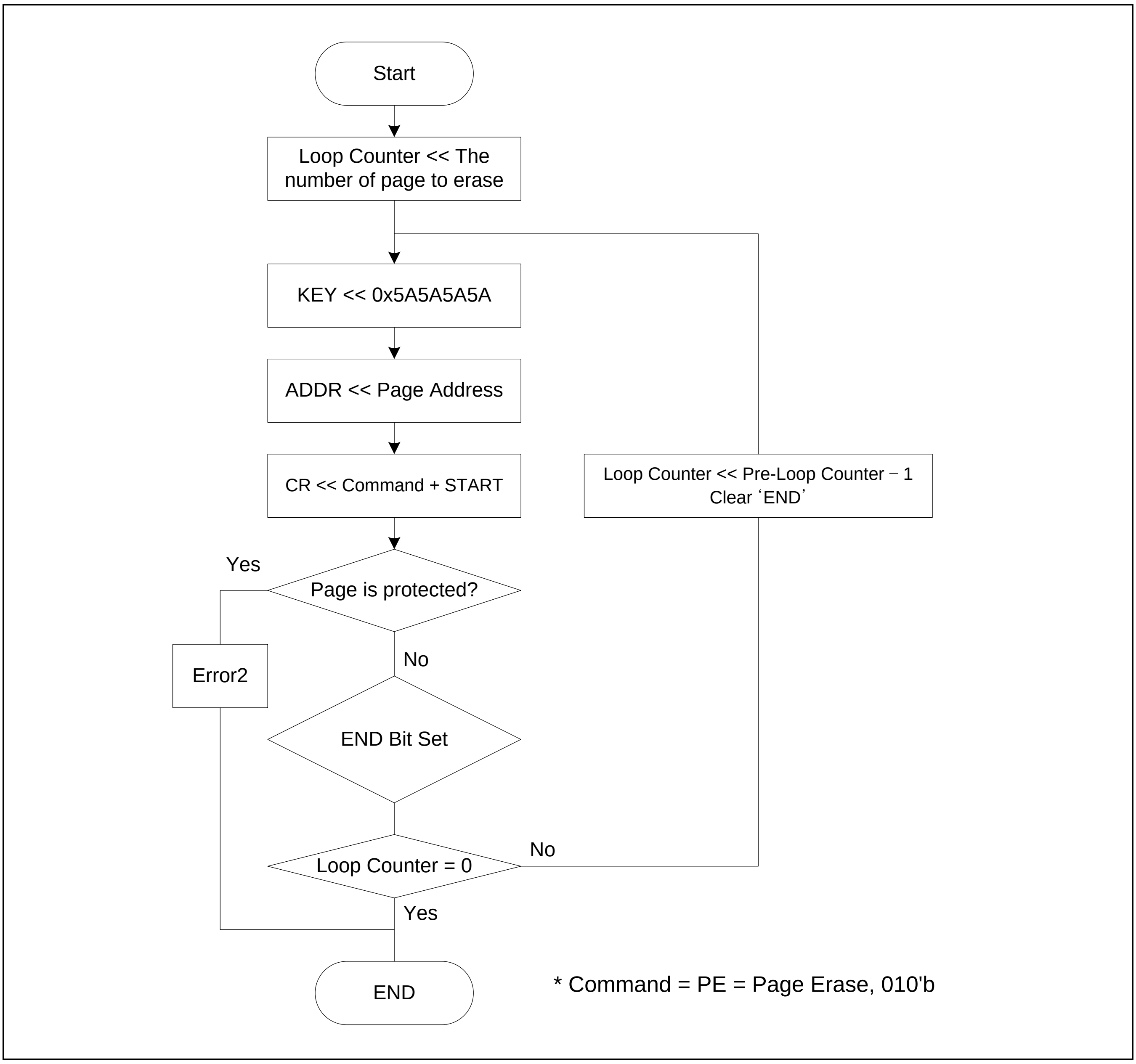


Figure 9-5 Page Erase Flowchart

9.2.4.3 Sector Erase

Figure 9-6 illustrates the Sector Erase flowchart.

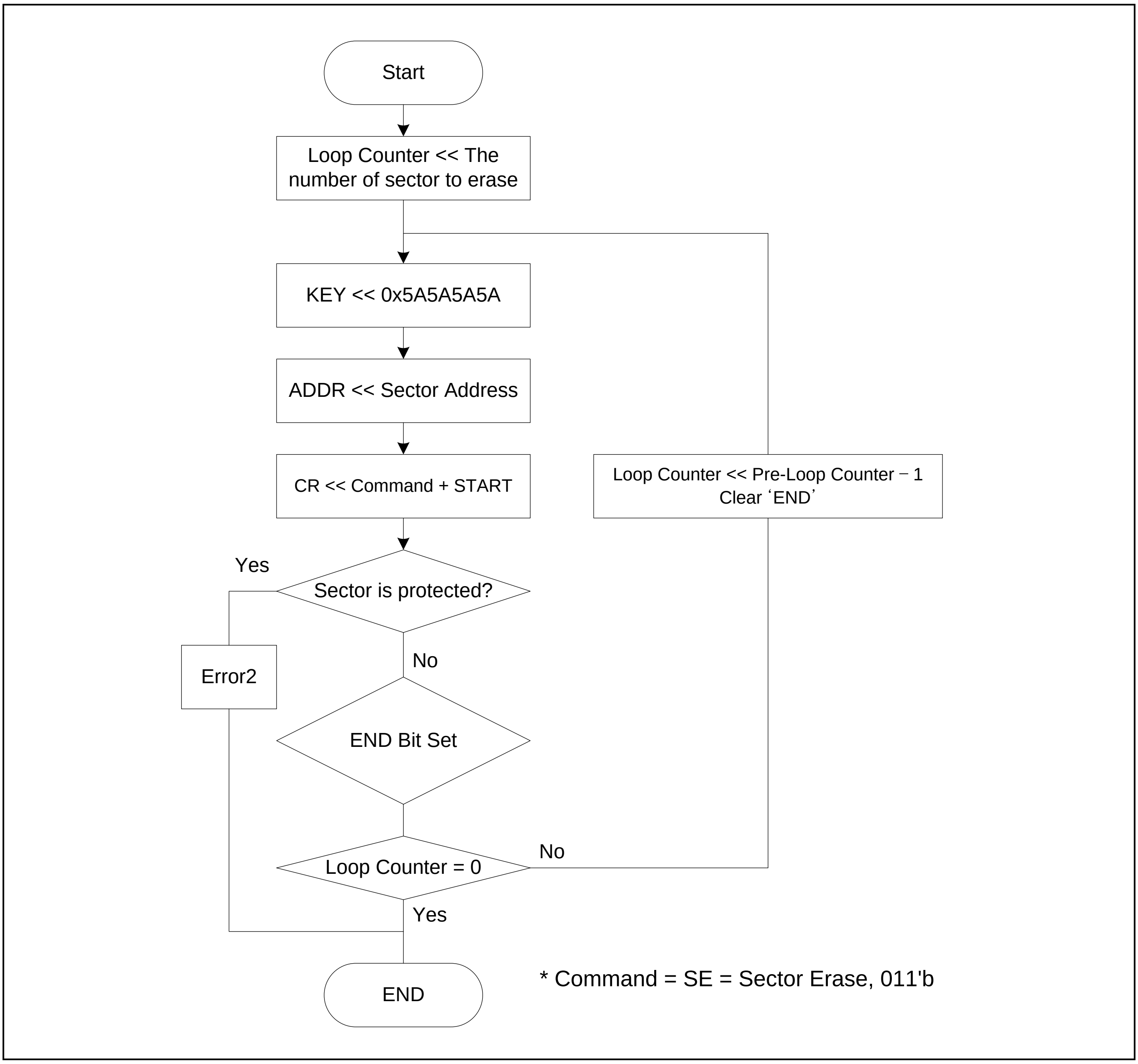


Figure 9-6 Sector Erase Flowchart

9.2.4.4 Entire Erase

Figure 9-7 illustrates the Entire Erase flowchart.

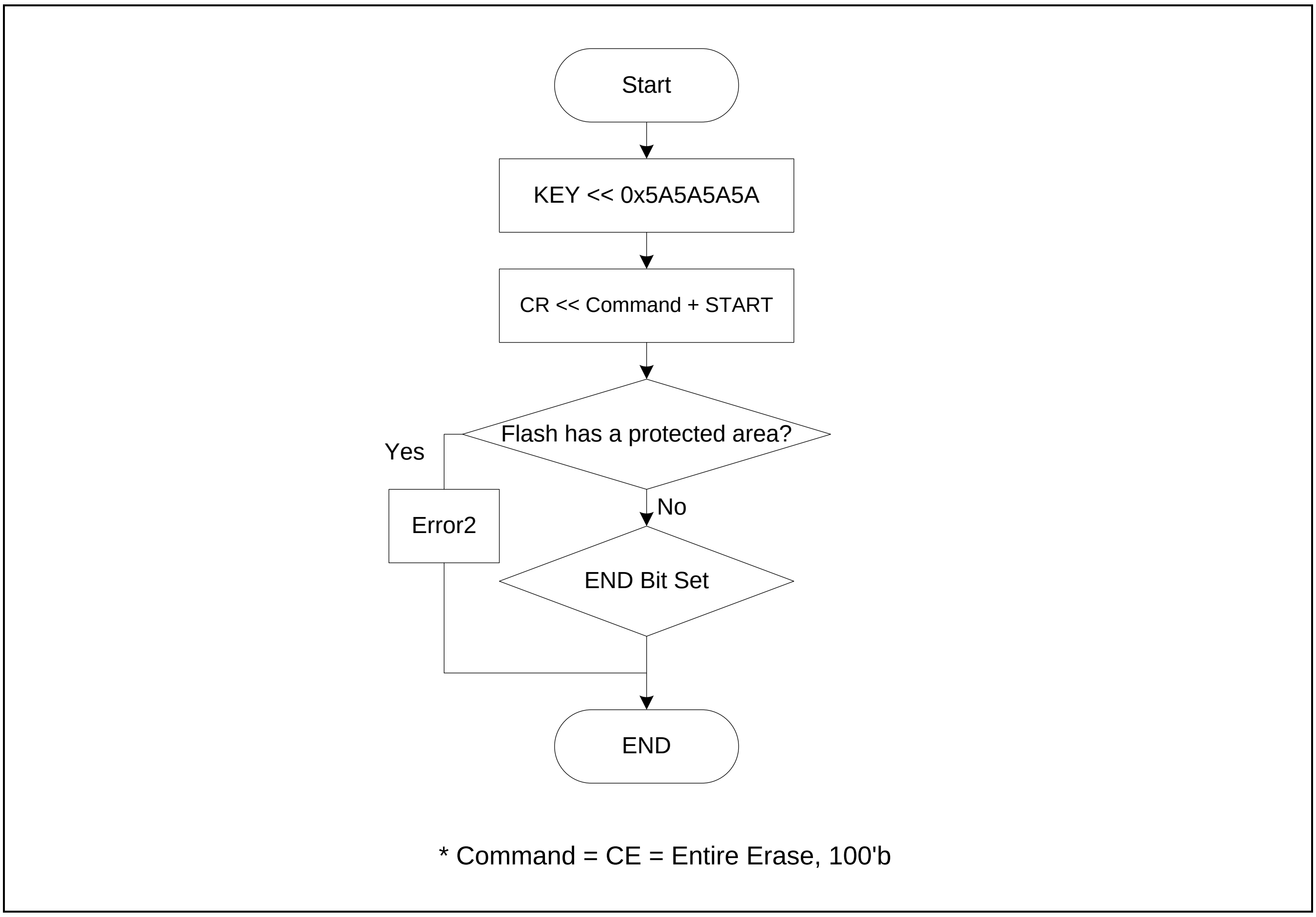
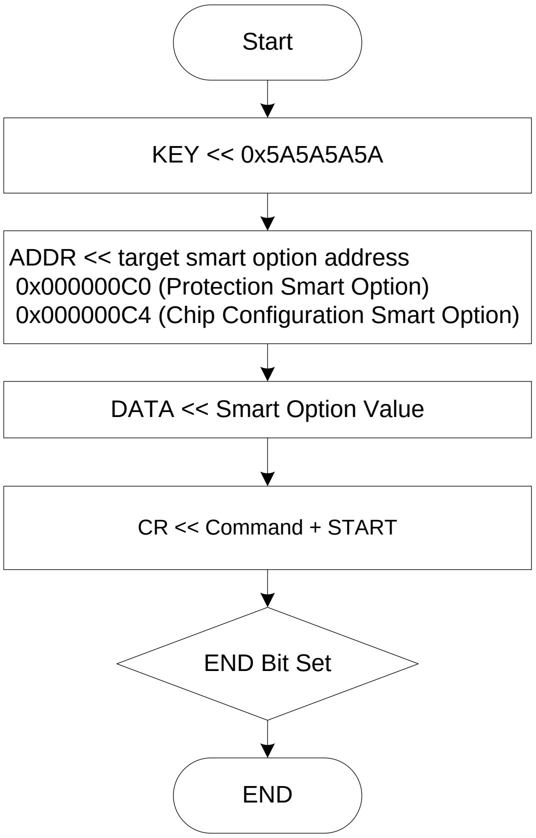


Figure 9-7 Entire Erase Flowchart

9.2.4.5 Smart Option Program

Figure 9-8 illustrates the Smart Option flowchart.


Start
KEY << 0x5A5A5A5A
ADDR << target smart option address
0x000000C0 (Protection Smart Option)
0x000000C4 (Chip Configuration Smart Option)
DATA << Smart Option Value
CR << Command + START
END Bit Set
END

* Command = SOP = Smart Option Program, 101'b

Figure 9-8 Smart Option Program Flowchart

9.2.4.6 Smart Option Erase

Figure 9-9 illustrates the Smart Option Erase flowchart.

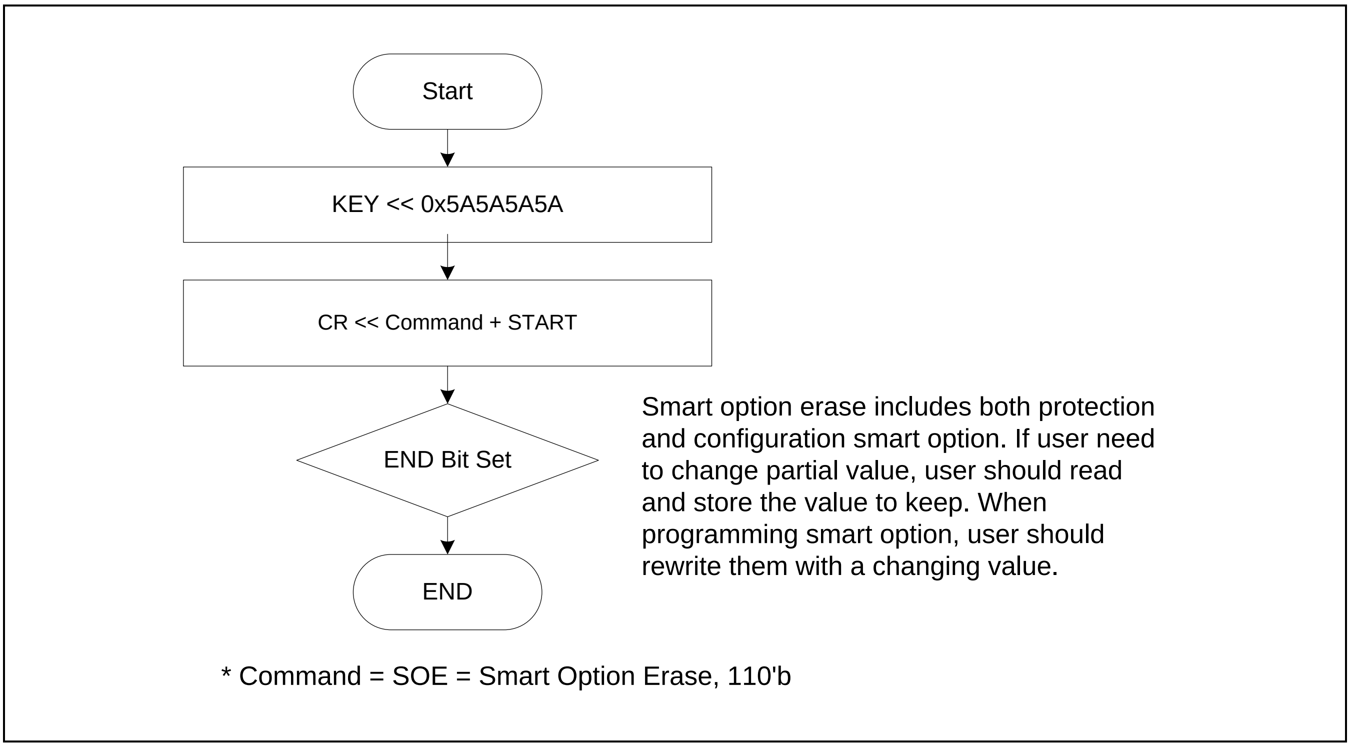

Start
KEY << 0x5A5A5A5A
CR << Command + START
END Bit Set
END
Smart option erase includes both protection and configuration smart option. If user need to change partial value, user should read and store the value to keep. When programming smart option, user should rewrite them with a changing value.
* Command = SOE = Smart Option Erase, 110'b

Figure 9-9 Smart Option Erase Flowchart

9.2.5 Error

9.2.5.1 Error Case 0

When a new Read access or normal program occurs while executing normal program operation, this new access is not taken into account and an interrupt for error event is generated.

Figure 9-10 illustrates the Error Case 0.

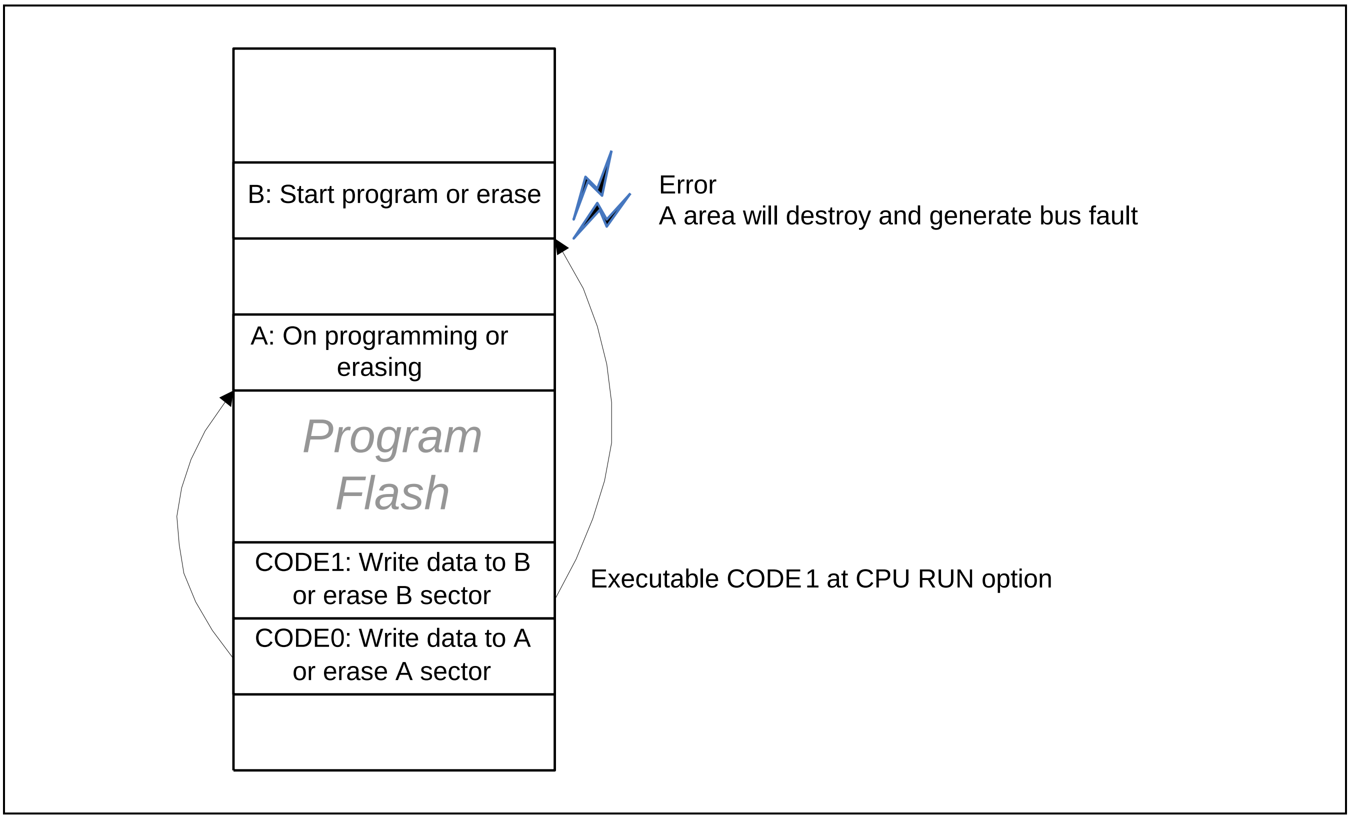

B: Start program or erase
Error
A area will destroy and generate bus fault
A: On programming or erasing
Program Flash
CODE1: Write data to B or erase B sector
Executable CODE 1 at CPU RUN option
CODE0: Write data to A or erase A sector

Figure 9-10 Error0 Condition

If you perform other operation (erase, normal program) while executing normal program operation, that operation is not taken into account and an interrupt for error event is generated.

Figure 9-11 illustrates the Error Case 0.

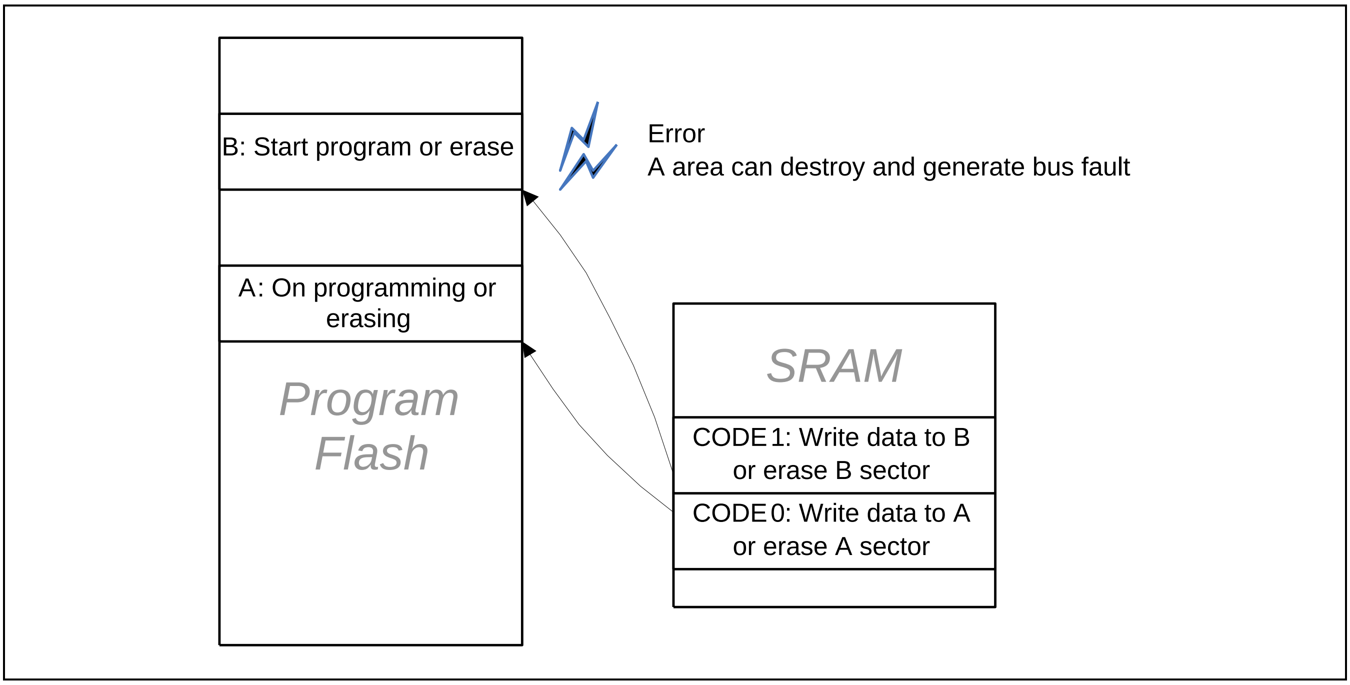

B: Start program or erase
A: On programming or erasing
Program Flash
Error
A area can destroy and generate bus fault
SRAM
CODE 1: Write data to B or erase B sector
CODE 0: Write data to A or erase A sector

Figure 9-11 Error0 Condition

9.2.5.2 Error Case 1

This error occurs when writing the undefined value (CMD[2:0] = 111'b) into CMD field of a control register during operation sequence

Figure 9-12 illustrates the Error Case 1.

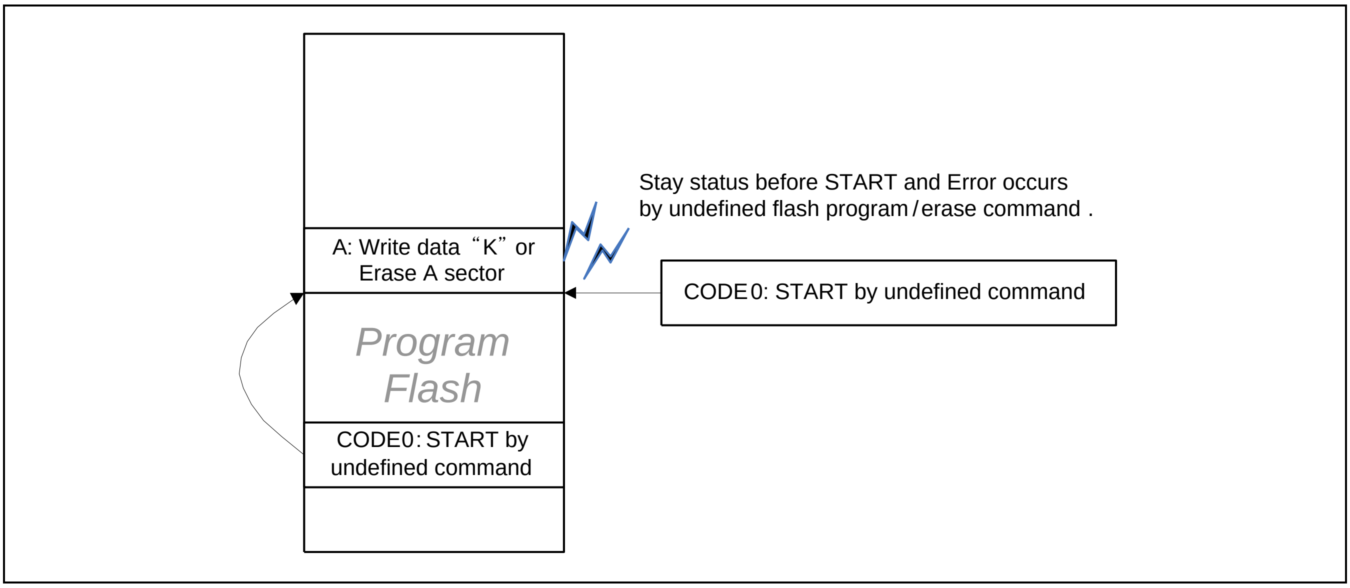

Stay status before START and Error occurs
by undefined flash program / erase command .
A: Write data “K” or
Erase A sector
CODE0: START by undefined command
Program
Flash
CODE0: START by
undefined command

Figure 9-12 Error1 Condition

9.2.5.3 Error Case 2

If the operation is not allowed because of protection while executing normal program operation, that operation is executed and an abort signal is generated on the internal bus along with an interrupt error.

Figure 9-13 illustrates the Error Case 2.

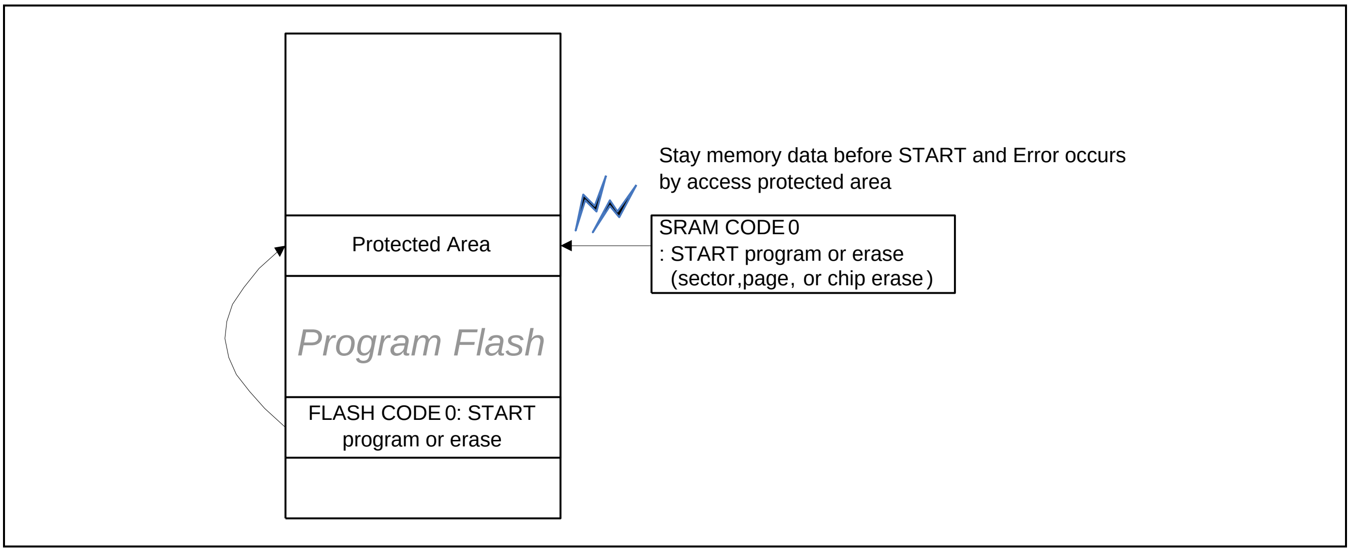


Figure 9-13 Error2 Condition

9.3 Register Description

9.3.1 Register Map Summary

- Base Address: 0x4001_0000

Register	Offset	Description	Reset Value
IFC_IDR	0x0000	ID register	0x0002_0039
IFC_CEDR	0x0004	Clock enable/disable register	0x0000_0000
IFC_SRR	0x0008	Software reset register	0x0000_0000
IFC_CR	0x000C	Control register	0x0000_0000
IFC_MR	0x0010	Mode register	0x0000_0000
IFC_IMSCR	0x0014	Interrupt mask set/clear register	0x0000_0000
IFC_RISR	0x0018	Raw interrupt status register	0x0000_0000
IFC_MISR	0x001C	Masked interrupt status register	0x0000_0000
IFC_ICR	0x0020	Interrupt clear register	0x0000_0000
IFC_SR	0x0024	Status register	0x0000_0000
IFC_AR	0x0028	Address register	0x0000_0000
IFC_DR	0x002C	Data register	0x0000_0000
IFC_KR	0x0030	Key register	0x0000_0000
SO_PSR	0x0034	Smart option protection status register	0xFFFF_FFFF (NOTE)
SO_CSR	0x0038	Smart option configuration status register	0xFFFF_FFFF (NOTE)
IFC_IOTR	0x003C	Internal OSC trimming register	0x00XX_XXXX

NOTE: For 1 and 2 the reset value by fabrication is "0xFFFFFFFF". But you can change the reset value by using Smart Option program.

Caution: IFC_CR, IFC_AR, IFC_DR, and IFC_KR are auto-cleared as command operation finish.

Other Write registers except IFC_IOTR can write when flash is not in running (operation by command).

9.3.1.1 IFC_ID

- Base Address: 0x4001_0000
- Address = Base Address + 0x0000, Reset Value = 0x0002_0039

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores the ID code for the corresponding IP.	0x0002_0039

9.3.1.2 IFC_CEDR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable Bit 0 = Disables Flash Clock. 1 = Enables Flash Clock. Flash software reset does not affect CLKEN bit status. Flash Clock controls can access Special Function Register (SFR) for flash controller.	0

9.3.1.3 IFC_SRR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Performs Flash Controller Software Reset operation. You can initialize all registers except Smart Option, Clock Enable/Disable and internal OSC trimming registers.	0

9.3.1.4 IFC_CR

- Base Address: 0x4001_0000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	R	Reserved	0
CMD	[6:4]	RW	Flash Program/Erase Command Field 000 = No effect 001 = Selects Normal Program Command. 010 = Selects Page Erase Command. 011 = Selects Sector Erase Command. 100 = Selects Entire Erase Command. 101 = Selects Smart Option Program Command. 110 = Selects Smart Option Erase Command. 111 = Prohibited Smart Option Erase operation clears protection and configuration Smart Option. The smart option has the several options in one register. If you change partial option, you should program new value to change and prior value not to change at a time.	0
RSVD	[5:1]	R	Reserved	0
START	[0]	RW	Operation (1) Start Bit This bit clears automatically at the end of command operation. 0 = No effect 1 = Starts When you write "1" while executing the command (normal program, page/sector/entire erase, Smart Option program, or erase), it does not have an effect. But this becomes one of the error cases. If you use the command not to be defined, the flash controller generates the signal for an error. That is defined as an ERR1. You can catch the error occurred with RISR (status) and MISR (interrupt) register.	0

NOTE: The register clears automatically after finishing the operation of command. You can write any value in the register but it will not be affected. You should check END status (interrupt) for next command operation.

1. The operations that support are:

Command	Description
NP	Normal Program
PE	Page Erase
SE	Sector Erase
CE	Entire Erase
SOP	Smart Option Program
SOE	Smart Option Erase

2. The IFC_CR determines the program/erase operation. In user program mode, internal flash controller can support Normal Program, Smart Option Program, Smart Option Erase, Sector Erase, and Entire Erase. Among six operations, select only one operation.

9.3.1.5 IFC_MR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0080

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	R	Reserved	0
FSMODE	[7]	RW	Flash Speed Mode Selection Bit You should change the flash mode before the clock switching occurs from normal speed mode to high speed mode. 0 = Selects normal speed flash mode. Use this when the system clock frequency is less than or equal 20 MHz. 1 = Selects high speed flash mode. Use this when the system clock frequency is greater than 20 MHz.	0
RSVD	[6:1]	R	Reserved	0
BACEN	[0]	RW	Boot Area Configuration Enable/Disable Control Bit You can enable or disable boot area. Enabled boot area is protected. You can not erase it by entire erase operation. To erase this area, you will have to perform Sector Erase or 32 times page erase operation. 0 = Disables boot sector area. You can erase entire area by performing entire erase in user mode. 1 = Enables boot sector area, 8 KB from 0x0000_0000 Flash area except sector 0. You can erase this using entire erase in user mode.	0

9.3.1.6 IFC_IMSCR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																					ERR2	ERR1	ERR0	RSVD							END
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	R	R	R	R	R	R	R	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	R	Reserved	0
ERR2	[10]	RW	ERRn: ERR Interrupt Mask Control Bit Error2 = Write or erase to protected memory region 0 = Interrupt is masked. (Disables an Interrupt) 1 = Interrupt is not masked. (Enables an Interrupt)	0
ERR1	[9]	RW	ERRn: ERR Interrupt Mask Control Bit Error1 = Write undefined value (CMD[2:0] = 111b) into CMD field of a control register during operation flow 0 = Interrupt is masked. (Disables an Interrupt) 1 = Interrupt is not masked. (Enables an Interrupt)	0
ERR0	[8]	RW	ERRn: ERR Interrupt Mask Control Bit Error0 = Tries to execute other operation (program, erase) during "normal user program" command operation. 0 = Interrupt is masked. (Disables an Interrupt) 1 = Interrupt is not masked. (Enables an Interrupt)	0
RSVD	[7:1]	R	Reserved	0
END	[0]	RW	END: END Interrupt Mask Control Bit 0 = Interrupt is masked. (Disables an Interrupt) 1 = Interrupt is not masked. (Enables an Interrupt)	0

NOTE: On a Read, the IFC_IMSCR register gives the current value of the mask on the relevant interrupt.
A Write of 1 to a particular bit, sets the mask and enables the interrupt to be read.
A Write of 0 clears the corresponding mask.

9.3.1.7 IFC_RISR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																					ERR2	ERR1	ERR0	RSVD							END
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	R	Reserved	0
ERR2	[10]	R	ERR Interrupt Raw Status Gives the raw interrupt state (prior to masking) of the ERRn interrupts. Error2 = Writes or erases to the protected memory region.	0
ERR1	[9]	R	ERR Interrupt Raw Status Gives the raw interrupt state (prior to masking) of the ERRn interrupts. Error1 = Writes undefined value (CMD [2:0] = 111b) into CMD field of a control register during operation flow.	0
ERR0	[8]	R	ERR Interrupt Raw Status Gives the raw interrupt state (prior to masking) of the ERRn interrupts. Error0 = Tries to execute other operation (program, erase) while executing "normal program" command operation.	0
RSVD	[7:1]	R	Reserved	0
END	[0]	R	END Interrupt Raw Status Gives the raw interrupt state (prior to masking) of the END interrupt. When operation defined by CMD [2:0] is finished then END bit is set to 1 (a command field in control register).	0

NOTE: On a Read, the IFC_RISR register gives the current raw status value of the corresponding interrupt prior to masking.

Caution: When ERRn occurs in operation, to continue flash operation after clear ERRn bit is recommended. You can clear ERRn interrupt/status by ICR register.

9.3.1.8 IFC_MISR

- Base Address: 0x4001_0000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																					ERR2	ERR1	ERR0	RSVD							END
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	R	Reserved	0
ERR2	[10]	R	ERR Interrupt Status Error2 = Writes or erases to protected memory region 0 = Interrupt does not occur. 1 = Interrupt occurs.	0
ERR1	[9]	R	ERR Interrupt Status Error1 = Writes undefined value (CMD [2:0] = 111b) into CMD field of a control register during operation flow. 0 = Interrupt does not occur. 1 = Interrupt occurs.	0
ERR0	[8]	R	ERR Interrupt Status Error0 = Try to execute other operation (program, erase) while executing "normal program" command operation. 0 = Interrupt does not occur. 1 = Interrupt occurs.	0
RSVD	[7:1]	R	Reserved	0
END	[0]	R	END Interrupt Status Gives the masked interrupt status (after masking) of the END interrupt. 0 = Interrupt does not occur. 1 = Interrupt occurs.	0

NOTE: On a Read, the IFC_MISR register gives current masked status value of corresponding interrupt.

9.3.1.9 IFC_ICR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																					ERR2	ERR1	ERR0	RSVD							END
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	R	Reserved	0
ERR2	[10]	W	ERRn: ERR Interrupt Clear Bit Error 2 = Writes or erases to the protected memory region 0 = No effect 1 = Clears ERR2 interrupt clear bit	0
ERR1	[9]	W	ERRn: ERR Interrupt Clear Bit Error 1 = Writes undefined value (CMD [2:0] = 111b) into the CMD field of a control register during operation flow. 0 = No effect 1 = Clears ERR1 interrupt clear bit.	0
ERR0	[8]	W	ERRn: ERR Interrupt Clear Bit Error 0 = Tries to execute other operation (program, erase) while operating by "normal program" command. 0 = No effect 1 = Clears ERR0 interrupt clear bit.	0
RSVD	[7:1]	R	Reserved	0
END	[0]	W	END: END Interrupt Clear Bit 0 = No effect 1 = Clears End interrupt clear bit.	0

NOTE: When you set to 1, corresponding bit in raw interrupt status and masked interrupt status register is cleared.

Caution: When ERRn occurs in operation, it is impossible to write any value into CR register. Clear it to execute next command. You can clear ERRn interrupt/status by the IFC_ICR register.

9.3.1.10 IFC_SR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0024, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
BUSY	[0]	R	Busy Status Flag 0 = Specifies flag not in operation (Programming or Erasing) 1 = Specifies reported flag, when flash is programming or erasing.	0

NOTE: This bit is changed from 0 to 1 at the real time (A) to start flash IP operation, after START bit with command sets (B) to 1. There will be some delay between (A) and (B). So you should not write new IFC_CR, IFC_KR, IFC_DR, and IFC AR values, before BUSY bit (IFC_SR[0]) sets to 1. Clear is performed automatically, when END bit is set to 1.

9.3.1.11 IFC_AR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
ADDR	[31:0]	RW	Internal Flash Memory Address to Program (Normal/Smart Option) or Erase (Page/Sector) • IFC_AR [31:0] ← You can select address in flash memory range. • IFC_AR [31:0] ← 0x0000_00C0 for Configuration Smart Option. • IFC_AR [31:0] ← 0x0000_00C4 for Protection Smart Option Sector 8 KB Unit= 0x2000 Page 256B Unit= 0x100	0

NOTE: The IFC_AR register is auto-cleared when command operation is completed.

9.3.1.12 IFC_DR

- Base Address: 0x4001_0000
- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
DATA	[31:0]	RW	Data to Write on the Target Address Register (IFC_AR) of the Internal Flash Memory. • DATA [31:0] ← You select specific word data (4B) to be written into the flash memory. • DATA [31:0] ← Trimming data in case of Configuration Smart Option. If you program with target value to change a Configuration Option, other bits except control bits (field) is set to "1". • DATA [31:0] ← Protection option data in case of Protection Smart Option.	0

NOTE: The IFC_DR register is auto-cleared after finishing command operation.

9.3.1.13 IFC_KR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
KEY	[31:0]	W	Key Register Value for Program and Erase Operation You can program any data into the flash memory or erase the data in the program mode. A specific register, key register, is used to prevent flash data from getting destroyed accidentally. You should write 0x5A5A5A5A into key register at first step. If you write other value (not 0x5A5A5A5A) at first step, you cannot execute the command operation. If the sequence for a flash operation has the invalid key value, not 0x5A5A5A5A, it has "No effect". NOTE: The IFC_KR register is cleared automatically after the completion of erase or program.	0

NOTE: The IFC_KR register is auto-cleared after finishing command operation.

9.3.1.14 SO_PSR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0034, Reset Value = 0xFFFF_FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD				nSRP	RSVD									nHWP	RSVD								nSWDP	nHWPA3	nHWPA2	nHWPA1	nHWPA0	RSVD			
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	R	Reserved	0xF
nSRP	[27]	R	Serial Read Protection Status Flag 0 = Enables Serial Read Protection 1 = Disables Serial Read Protection	1
RSVD	[26:18]	R	Reserved	0x1FF
nHWP	[17]	R	Hardware Protection Status Flag 0 = Enables Hardware Protection 1 = Disables Hardware Protection	1
RSVD	[16:7]	R	Reserved	0xFF
nSWDP	[8]	R	SWD Protection Status Flag 0 = Enables SWD Protection 1 = Disables SWD Protection	1
nHWPAx	[7:4]	R	Hardware Protection Area Area is a unit that consists of: 0 = Specifies area to be protected by Hardware Protection Smart Option 1 = Specifies area not to be protected because Hardware Protection of corresponding area is disabled.	0xF
RSVD	[3:0]	R	Reserved	0xF

This register shows the Smart Option value. Programmed smart option affects on hardware after chip reset. After chip reset, the value to read is real current smart option status

The reset value of fabrication is "0xFFFFFFFF" but you can change the value using Smart Option Program.

9.3.1.15 SO_CSR

- Base Address: 0x4001_0000
- Address = Base Address + 0x0038, Reset Value = 0xFFFF_FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																BTDIV				RSVD								XOUT	XIN	RSVD	POCCS
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0xFFFF
BTDIV	[15:12]	R	Basic Timer Divider selection bit in the reset time. 0000, 0001, 0010 = Not used 0011 = 1 0100 = 2 0101 = 4 0110 = 8 0111 = 16 1000 = 32 1001 = 64 1010 = 128 1011 = 256 1100 = 512 1101 = 1024 1110 = 2048 1111 = 4096	0xF
RSVD	[11:4]	R	Reserved	0xFF
XOUT	[3]	R	External Main Clock Input/Output Pin Configuration Bit 0 = Configure IO pin as the corresponding pin. 1 = Configure XOUT as the corresponding pin.	1
XIN	[2]	R	External Main Clock Input/Output Pin Configuration Bit 0 = Configure IO pin as the corresponding pin. 1 = Configure XIN as the corresponding pin.	1
RSVD	[1]	R	Reserved	1
POCCS	[0]	R	Power-On CPU (system) Clock Selection Field When Power-On, the selected clock source by Smart Option is used for operation (CPU Clock). 0 = Selects EMCLK (External Main Clock) 1 = Selects IMCLK (Internal Main Clock)	1

Control bits are related to Clock Manager for chip operation. You can change the reset value of a Clock Manager by programming smart option value. If Program Smart Option and chip reset do not occur, the value that SO_CSR register shows is different from current Smart Option because the programmed Smart Option (new option) is effective after chip reset.

The reset value of fabrication is "0xFFFFFFFF" but you can change the value using Smart Option program. If you program with target value to change a configuration option then other bits except control bits is set to 1.

9.3.1.16 IFC_IOTR

- Base Address: 0x4001_0000
- Address = Base Address + 0x003C, Reset Value = 0x00XX_XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IOTKEY								RSVD																	OSC						
0	0	0	0	0	0	0	0	0	0	X	X	X	X	X	X	0	X	X	X	X	X	X	X	0	X	X	X	X	X	X	X
W	W	W	W	W	W	W	W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
IOTKEY	[31:24]	W	Key for Write Access into the CM_IOTR Register Any Write in CM_IOTR register bits is effective only if the IOTKEY is equal to 0x53.	0
RSVD	[26:7]	RW	Reserved	0
OSC	[6:0]	RW	Internal Oscillator (40 MHz) Trim Value You can use this field to trim values on changed condition (Voltage, Temperature, and so on). NOTE: Reset value depends on trimming value by fabrication.	–

10 Inverter Motor Controller (IMC)

10.1 Overview

You can use the Inverter Motor Controller (IMC) for three phase inverter motor.

10.1.1 Features

The features of IMC are:

- 3-phase Pulse Width Modulation (PWM) signal outputs
  - PWMxU0 and PWMxD0
  - PWMxU1 and PWMxD1
  - PWMxU2 and PWMxD2
- Dead time insertion
- Eight compare registers for IMC start trigger signal generation and interrupt
- High-Z output generation

10.1.2 Pin Description

Table 10-1 describes the pin description of IMC.

Table 10-1 IMC Pin Description

Pin Name	Function	I/O Type	Comments
PWMU[2:0]	PWM Up Side Output for Inverter Motor	O	–
PWMD[2:0]	PWM Down Side Output for Inverter Motor	O	–
PWMOFF	Input Pin for PWM Output Off	I	–

10.2 Functional Description

Functional description section includes:

- Block diagram of IMC
- Operations
- Phase Signal Generation

10.2.1 Block Diagram

Figure 10-1 illustrates the block diagram for IMC.

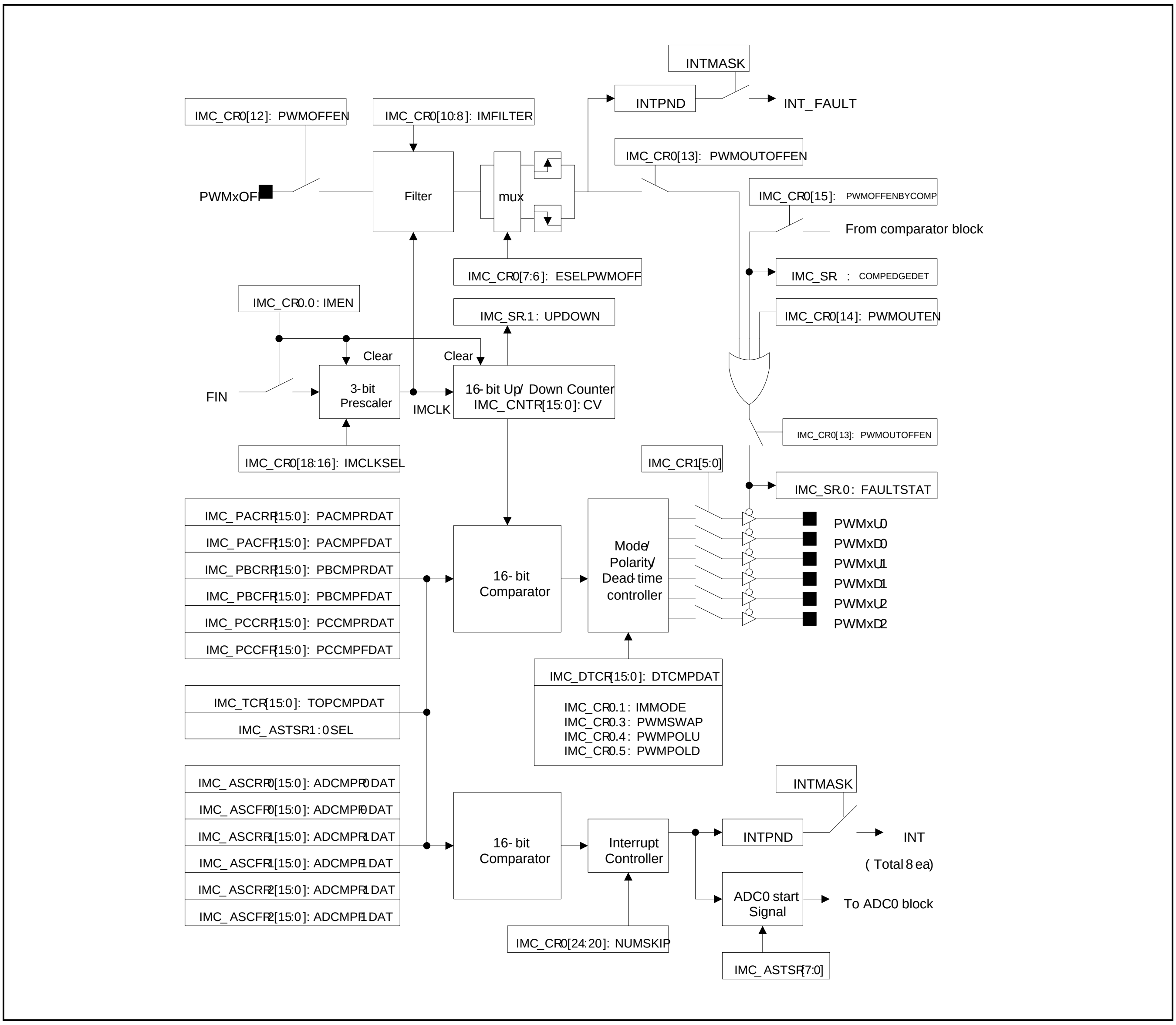


Figure 10-1 IMC Block Diagram

10.2.2 Operation

There are two modes for the operation of IMC:

- Tri-Angular Wave mode
- Saw-Tooth Wave mode

10.2.2.1 Tri-Angular Wave

Some examples of Tri-Angular wave mode are:

- High-Active Switch-ON Inverter Motor Control
  - PWMPOLU is set to the value of "LOW START"
  - PWMPOLD is set to the value of "HIGH START"
  - IMC_TCR = 7, IMC_DTCR = 2, and IMC_ASCRR0 = 6
  - IMC_PACRR = 3 and IMC_PACFR = 4

Figure 10-2 illustrates the Tri-Angular wave signal generation.

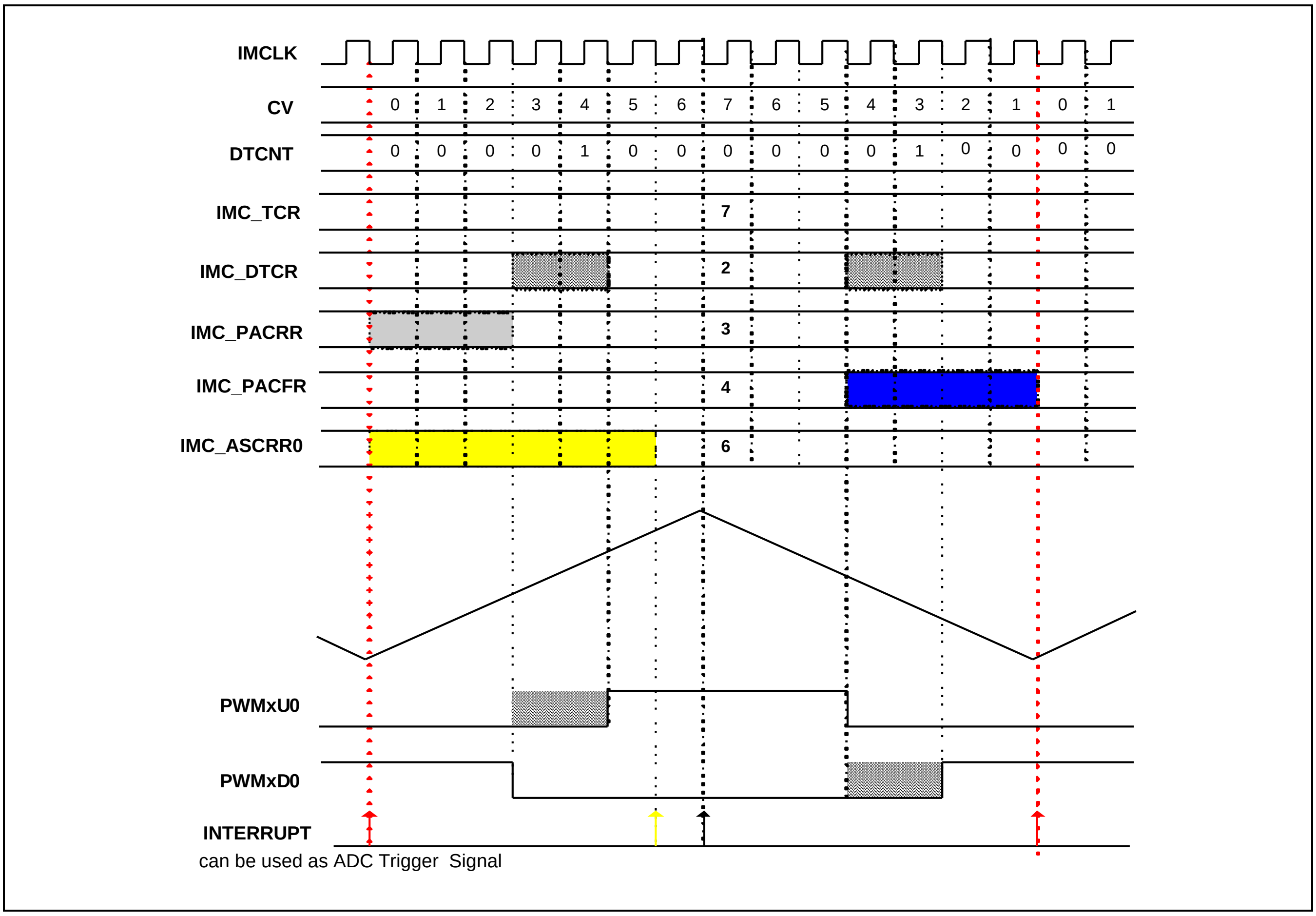


Figure 10-2 Tri-Angular Wave Signal Generation

10.2.2.2 Saw-Tooth Wave

Some examples of Saw-Tooth wave mode are:

- High-Active Switch-ON Inverter Motor Control
  - PWMPOLU is set to the value of "LOW START"
  - PWMPOLD is set to the value of "HIGH START"
    - IMC_TCR = 7, IMC_DTCR = 2, IMC_ASCRR0 = 6, and IMC_PACRR = 3

Figure 10-3 illustrates the Saw-Tooth wave signal generation.

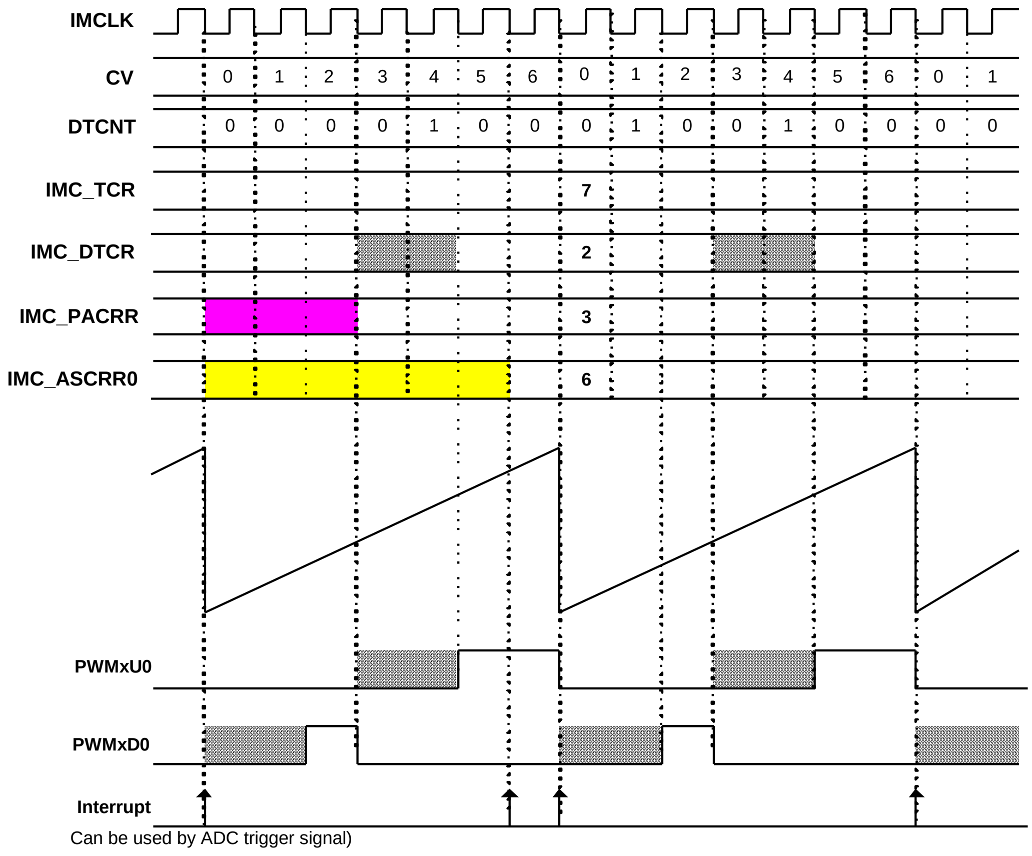


Figure 10-3 Saw-Tooth Wave Signal Generation

10.2.3 Phase Signal Generation

This section describes the phase signal generation process.

10.2.3.1 Tri-Angular Wave (IMMODE = 0)

PWMSWAP = 0, PWMPOLU = 0 (Low Start), and PWMPOLD = 1 (High Start)

You can use these phase signals when switches of UP side and DOWN side are High active in three phase motor application. This implies one pair of switches of UP side and DOWN side do not have the conditions that are High active simultaneously. It inserts the dead time appropriately.

Figure 10-4 illustrates the Tri-Angular wave (No SWAP, a Low Start PWMxUy, and High Start PWMxDy).

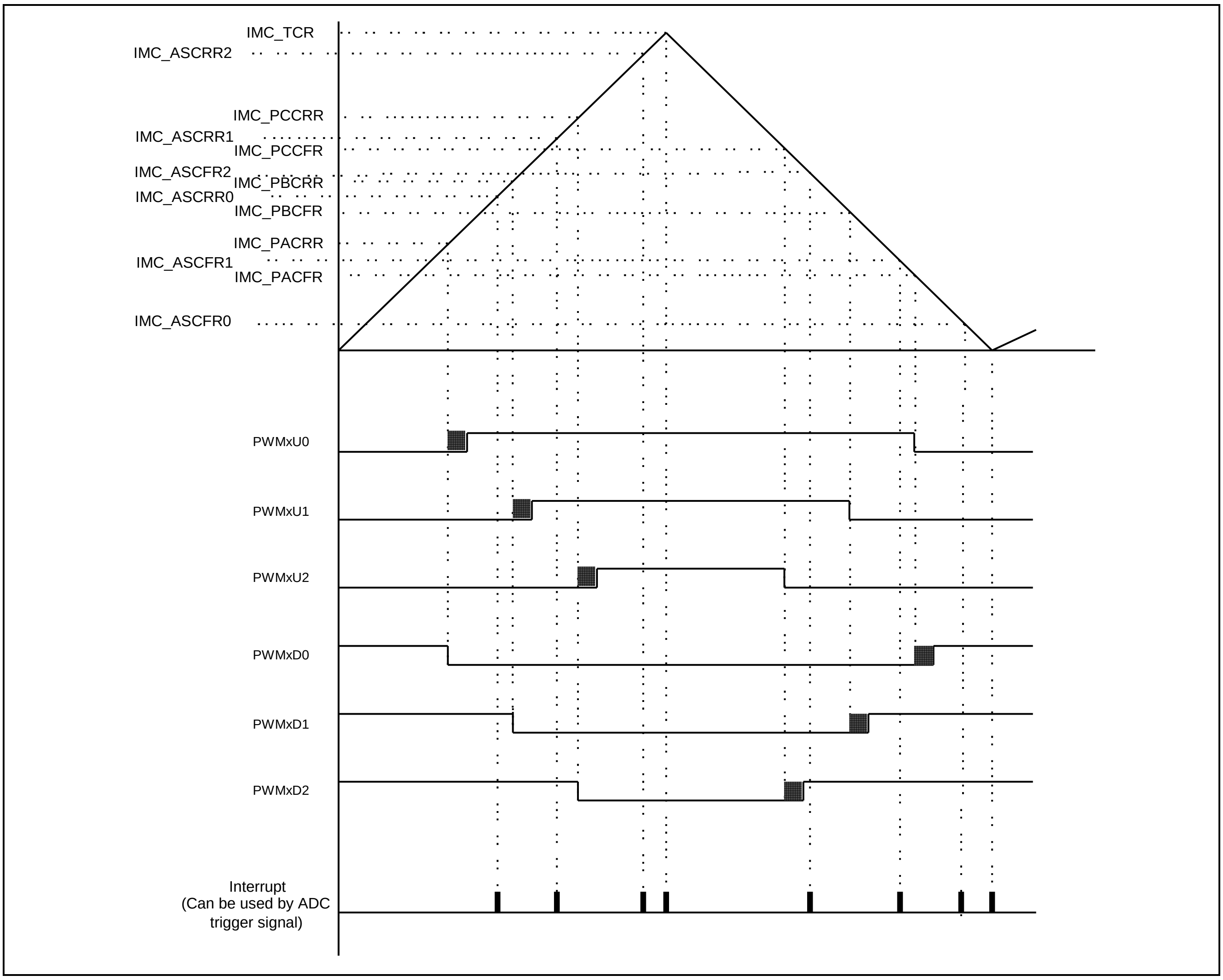


Figure 10-4 Tri-Angular Wave (No SWAP, a Low Start PWMxUy, and High Start PWMxDy)

NOTE: For 100 % duty of upside, you should set the rising/falling compare register to "0".
For 0 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

Figure 10-5 illustrates the signal of PWM (Assumption: Duration of dead time is 2 percent duty.)

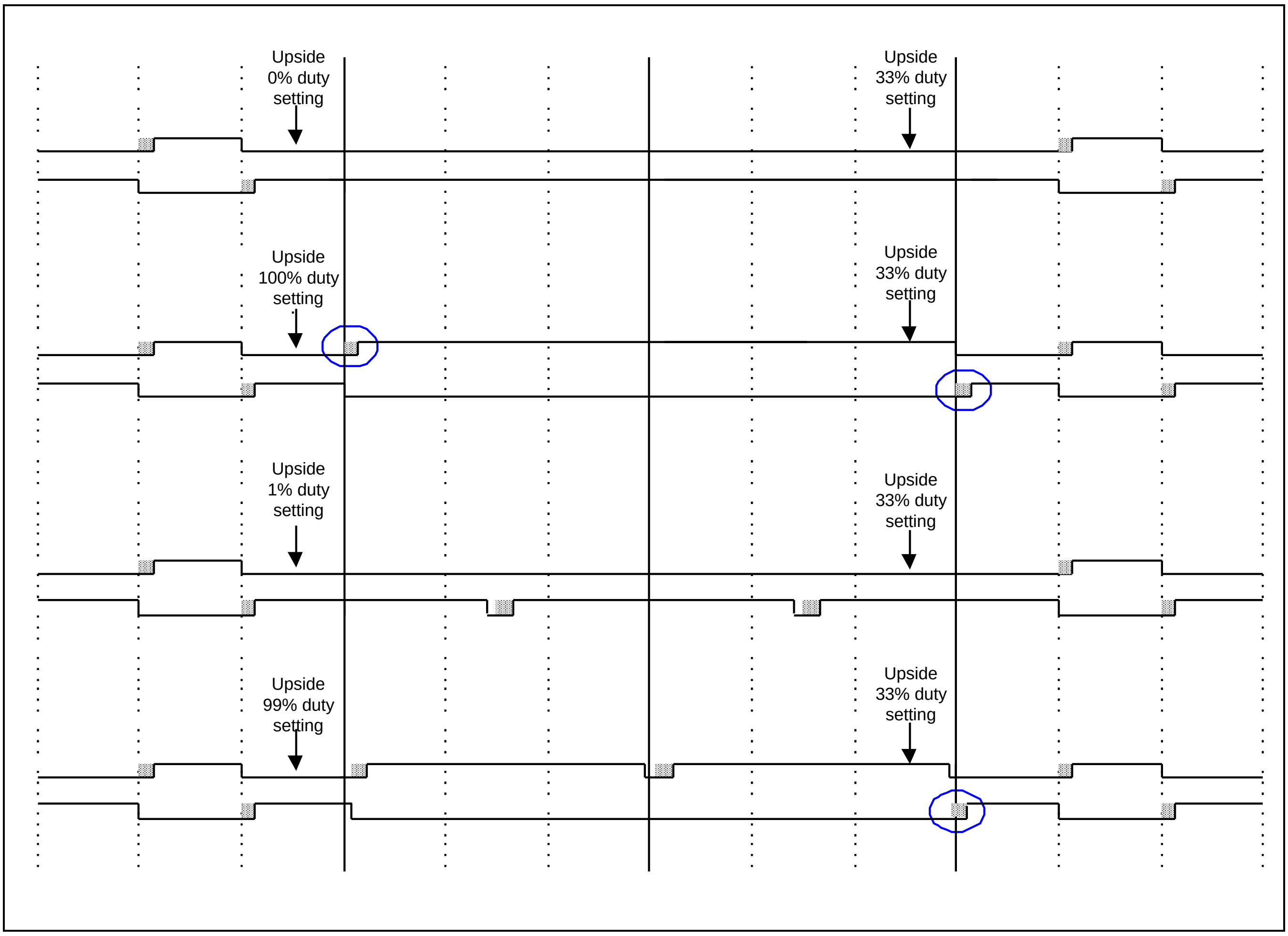


Figure 10-5 Tri-Angular Wave Duty (No SWAP, a Low Start PWMxUy, and High Start PWMxDy)

10.2.3.2 Tri-Angular Wave (IMMODE = 0)

PWMSWAP = 1, PWMPOLU = 0 (Low Start), and PWMPOLD = 1 (High Start)

Figure 10-6 illustrates the Tri-Angular wave (SWAP, a Low Start PWMxUy, and High Start PWMxDy).

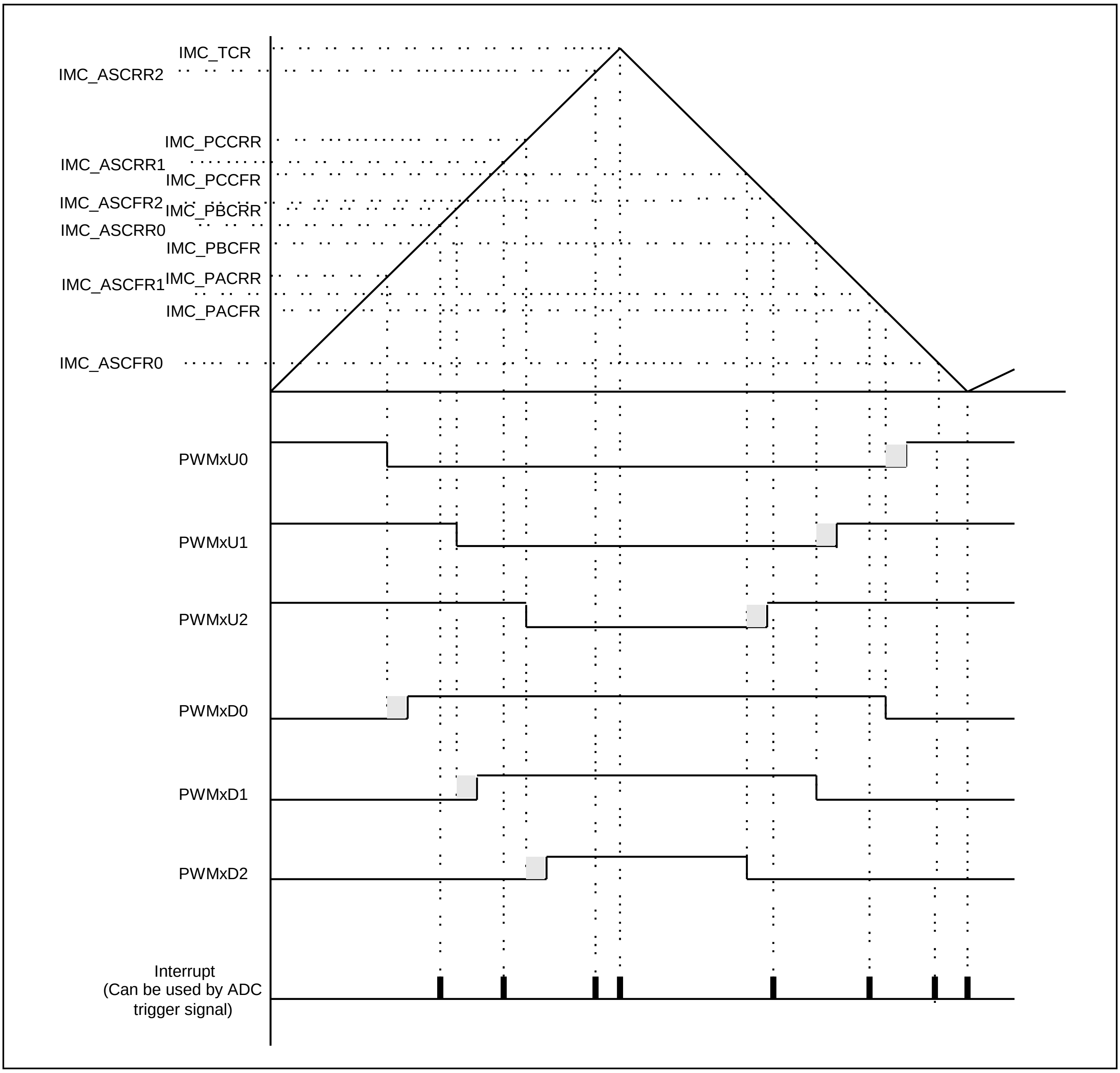


Figure 10-6 Tri-Angular Wave (SWAP, a Low Start PWMxUy, and High Start PWMxDy)

NOTE:

1. Both the switches of upside and down side are high active.
2. For 0 % duty of upside, you should set the rising/falling compare register to "0".
   For 100 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

Figure 10-7 illustrates the signal of PWM (Assumption: Duration of dead time is 2 percent duty.).

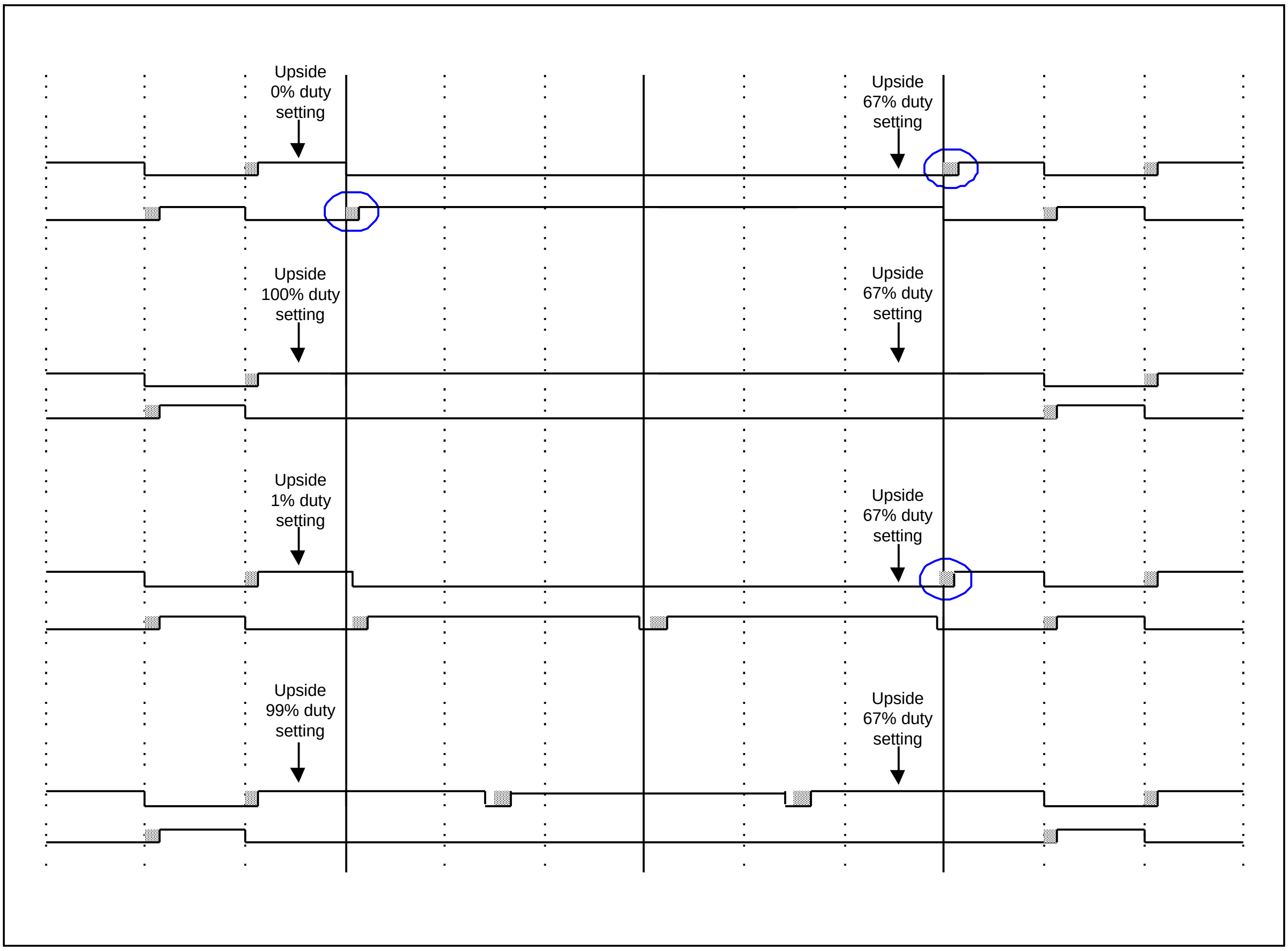


Figure 10-7 Tri-Angular Wave Duty (SWAP, a Low Start PWMxUy, and High Start PWMxDy)

10.2.3.3 Tri-Angular Wave (IMMODE = 0)

PWMSWAP = 0, PWMPOLU = 0 (Low Start), and PWMPOLD = 0 (Low Start)

Figure 10-8 illustrates the Tri-Angular wave (No SWAP, a Low Start PWMxUy, and Low Start PWMxDy).

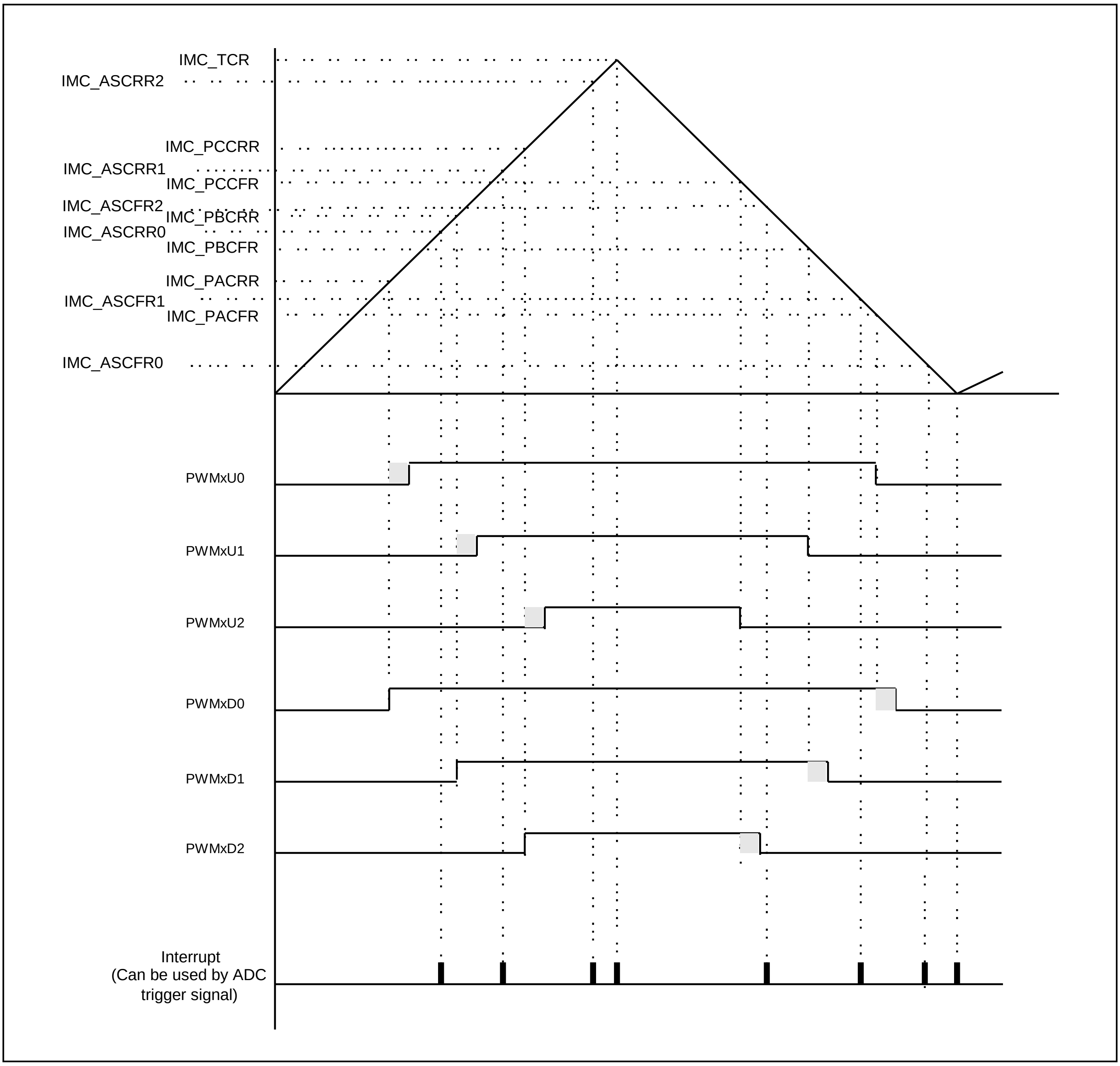


Figure 10-8 Tri-Angular Wave (No SWAP, a Low Start PWMxUy, and Low Start PWMxDy)

NOTE:

1. The switch of upside is high active and the switch of down side is low active.
2. For 100 % duty of upside, you should set the rising/falling compare register to "0".
   For 0 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.4 Tri-Angular Wave (IMMODE = 0)

PWMSWAP = 1, PWMPOLU = 0 (Low Start), and PWMPOLD = 0 (Low Start)

Figure 10-9 illustrates the Tri-Angular wave (SWAP, a Low Start PWMxUy, and Low Start PWMxDy).

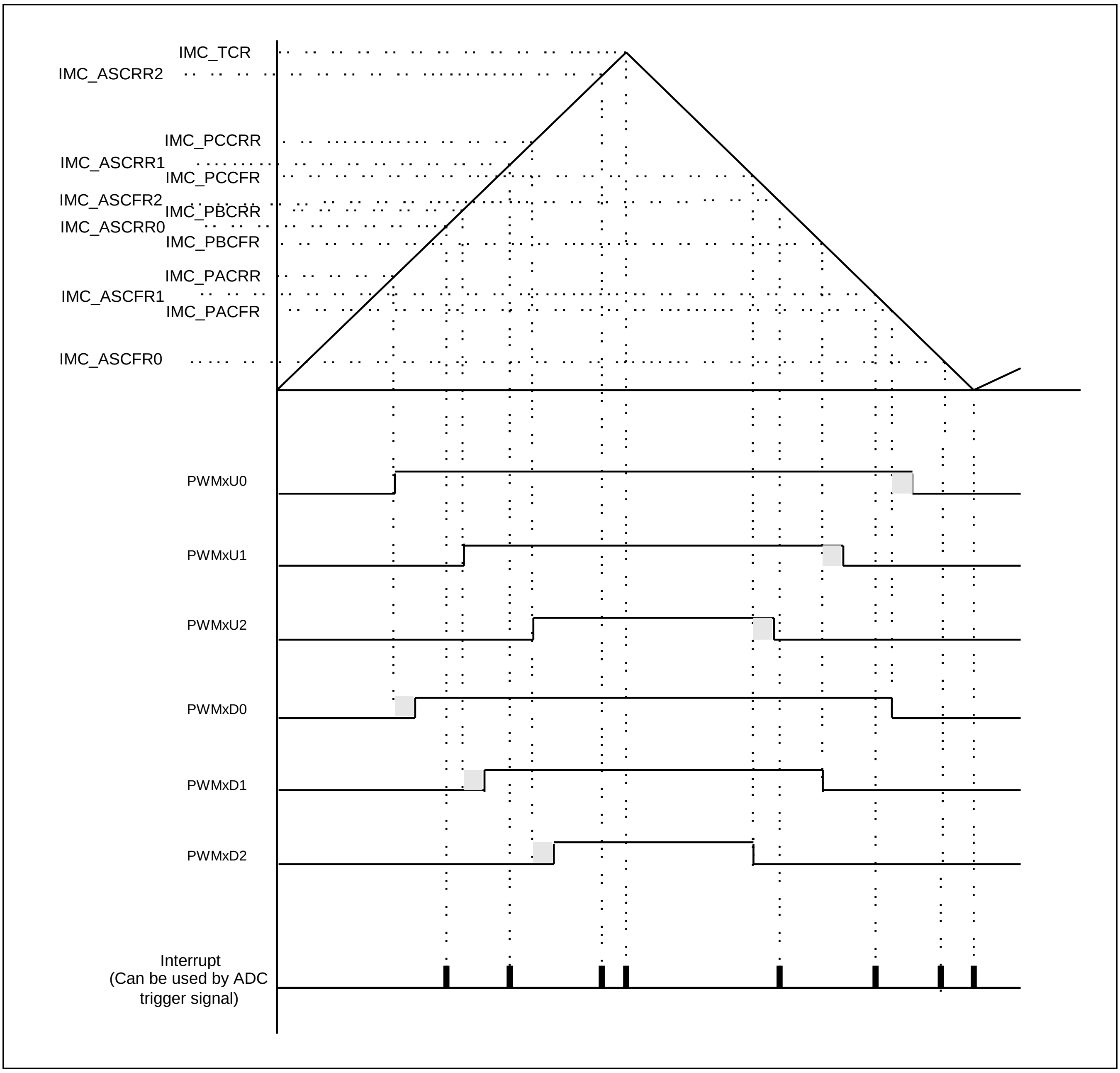


Figure 10-9 Tri-Angular Wave (SWAP, a Low Start PWMxUy, and Low Start PWMxDy)

NOTE:

1. The switch of upside is low active and the switch of down side is high active.
2. For 0 % duty of upside, you should set the rising/falling compare register to "0".
   For 100 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.5 Tri-Angular Wave (IMMODE = 0)

PWMSWAP = 0, PWMPOLU = 1 (High Start), and PWMPOLD = 0 (Low Start)

Figure 10-10 illustrates the Tri-Angular wave (No SWAP, a High Start PWMxUy, and Low Start PWMxDy).

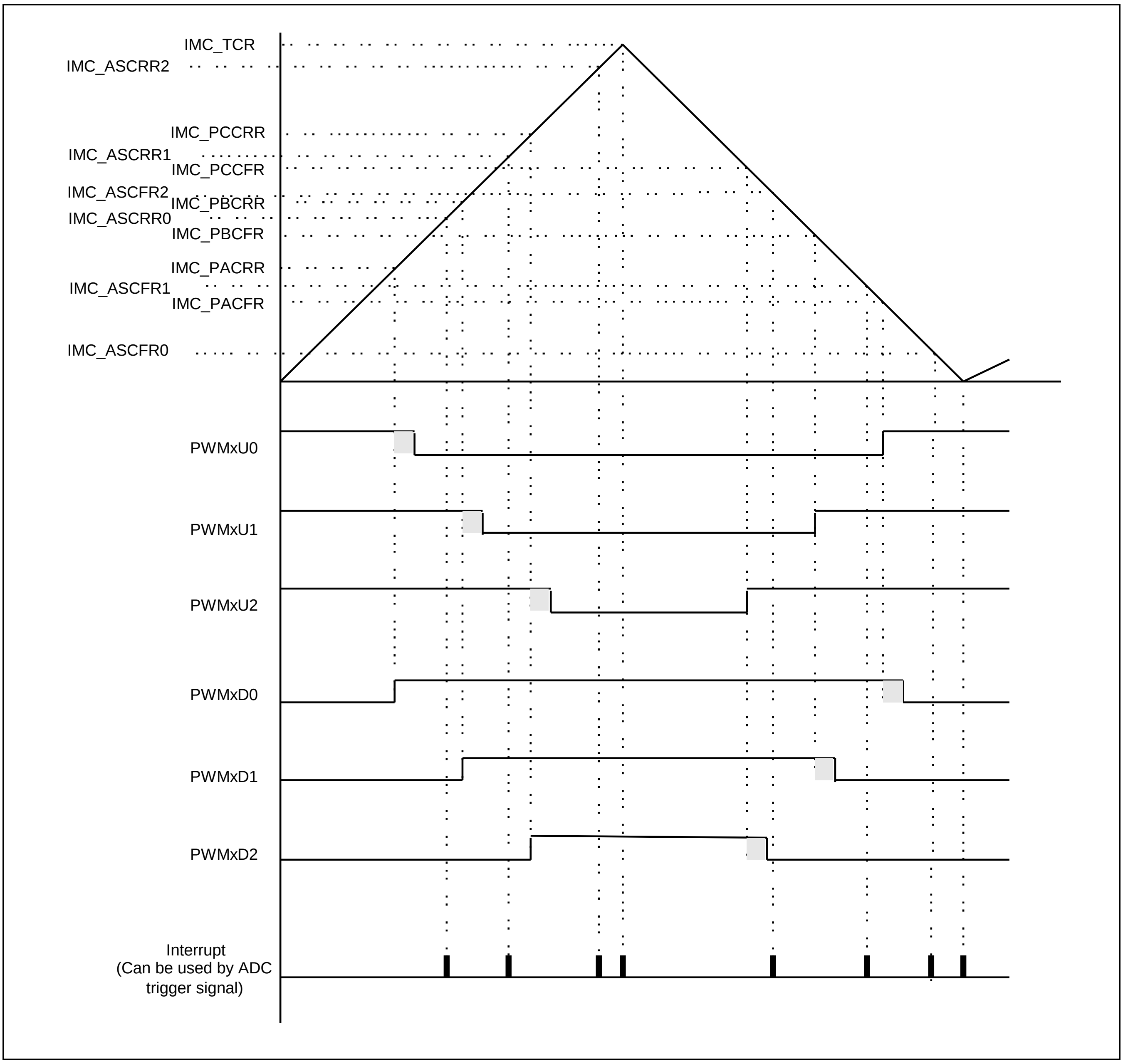


Figure 10-10 Tri-Angular Wave (No SWAP, a High Start PWMxUy, and Low Start PWMxDy)

NOTE:

1. Both the switches of upside and down side are low active.
2. For 100 % duty of upside, you should set the rising/falling compare register to "0".
   For 0 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

Figure 10-11 illustrates the signal of PWM. (Assumption: Duration of dead time is 2 percent duty).

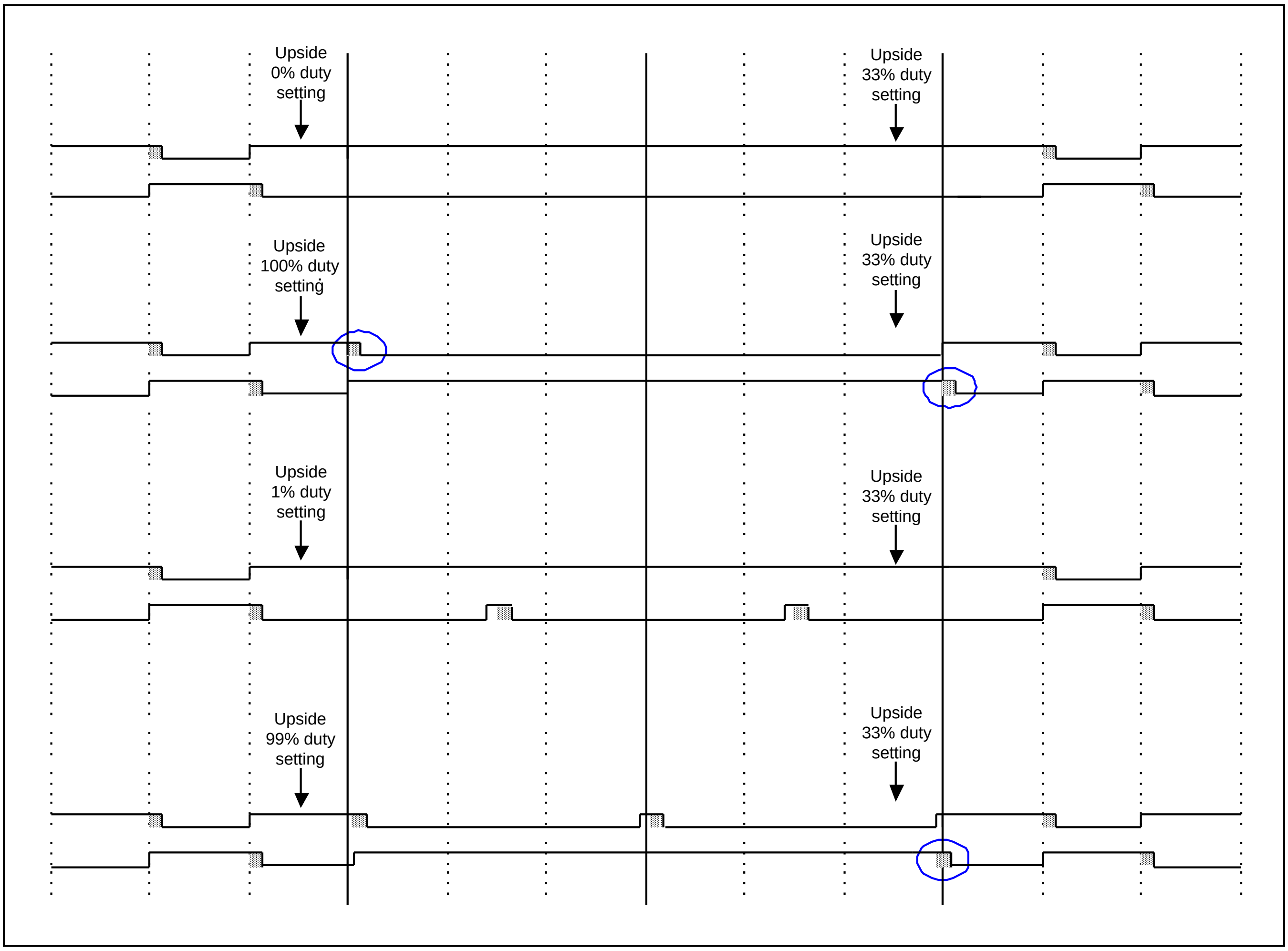


Figure 10-11 Tri-Angular Wave Duty (No SWAP, a High Start PWMxUy, and Low Start PWMxDy)

10.2.3.6 Tri-Angular Wave (IMMODE = 0)

PWMSWAP = 1, PWMPOLU = 1 (High Start), and PWMPOLD = 0 (Low Start)

Figure 10-12 illustrates the Tri-Angular wave (SWAP, a High Start PWMxUy, and Low Start PWMxDy).

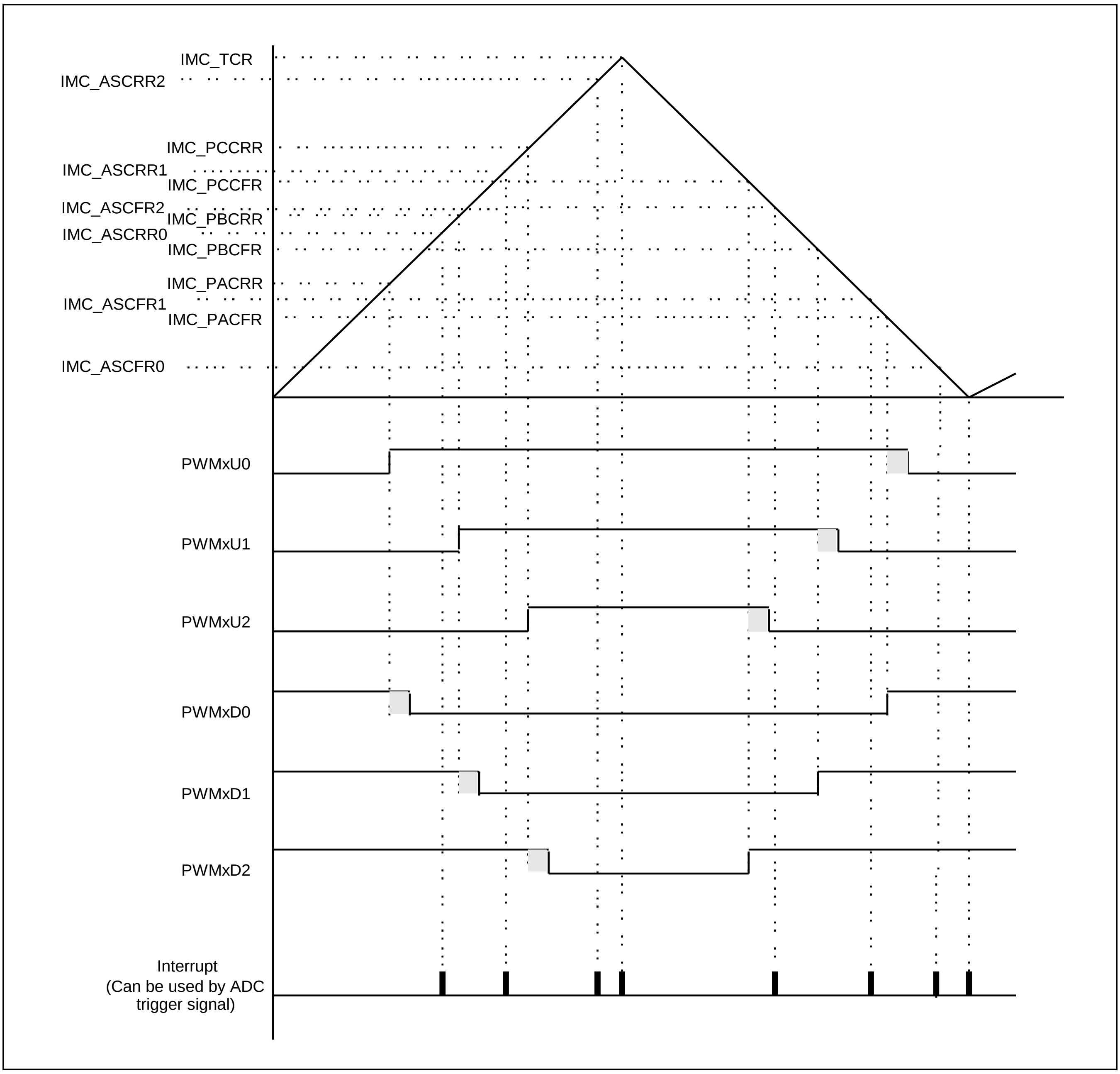


Figure 10-12 Tri-Angular Wave (SWAP, a High Start PWMxUy, and Low Start PWMxDy)

NOTE:

1. Both the switches of upside and down side are low active.
2. For 0 % duty of upside, you should set the rising/falling compare register to "0".
   For 100 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.7 Tri-Angular Wave (IMMODE = 0)

PWMSWAP = 0, PWMPOLU = 1 (High Start), and PWMPOLD = 1 (High Start)

Figure 10-13 illustrates the Tri-Angular wave (No SWAP, a High Start PWMxUy, and High Start PWMxDy).

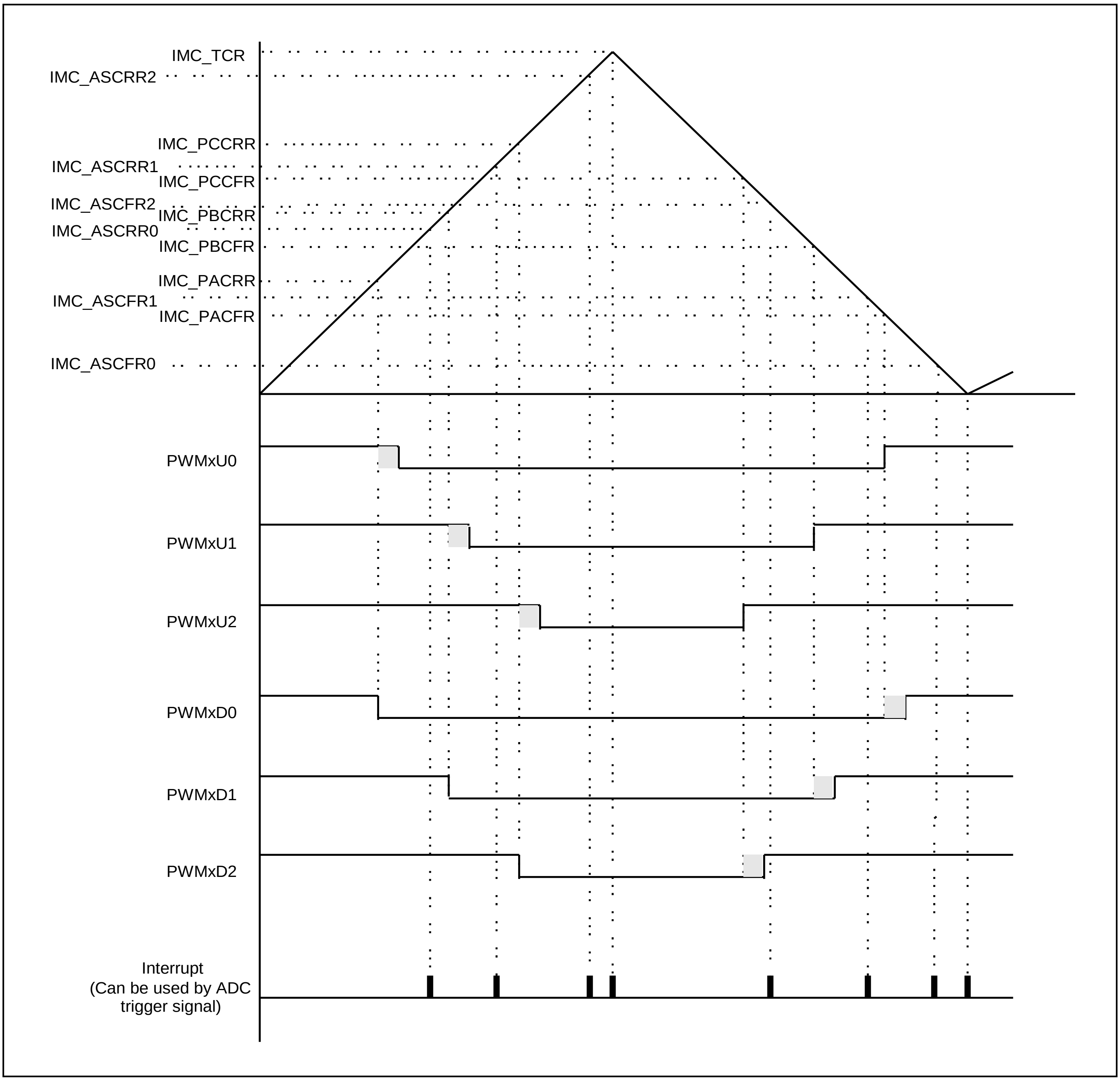


Figure 10-13 Tri-Angular Wave (No SWAP, a High Start PWMxUy, and High Start PWMxDy)

NOTE:

1. The switch of upside is low active and the switch of down side is high active.
2. For 100 % duty of upside, you should set the rising/falling compare register to "0".
   For 0 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.8 Tri-Angular Wave (IMMODE = 0)

PWMSWAP = 1, PWMPOLU = 1 (High Start), and PWMPOLD = 1 (High Start)

Figure 10-14 illustrates the Tri-Angular wave (SWAP, a High Start PWMxUy, and High Start PWMxDy).

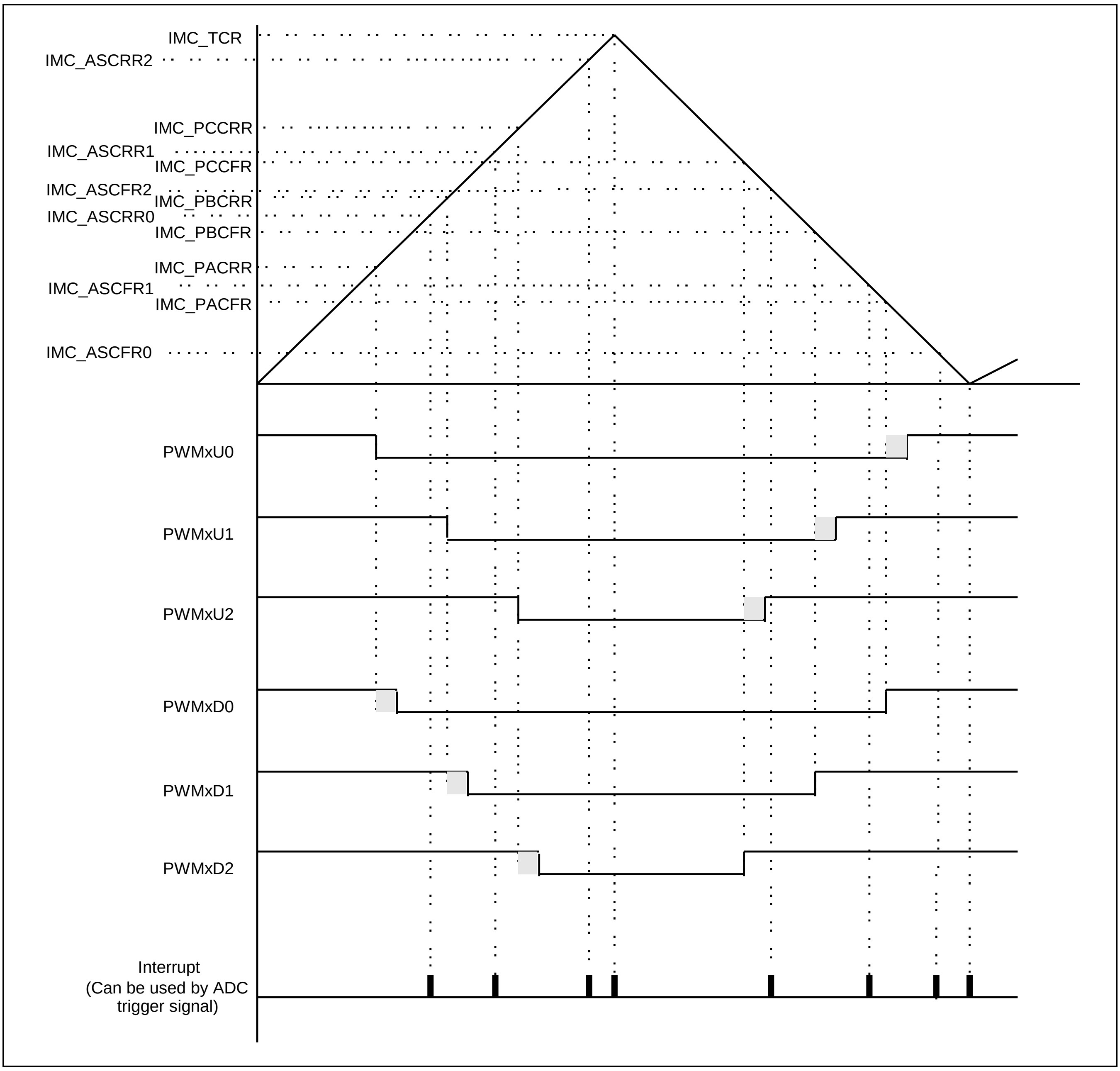


Figure 10-14 Tri-Angular Wave (SWAP, a High Start PWMxUy, and High Start PWMxDy)

NOTE:

1. The switch of upside is high active and the switch of down side is low active.
2. For 0 % duty of upside, you should set the rising/falling compare register to "0".
   For 100 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.9 Saw-Tooth Wave (IMMODE = 1)

PWMSWAP = 0, PWMPOLU = 0 (Low Start), and PWMPOLD = 1 (High Start)

Figure 10-15 illustrates the Saw-Tooth wave (No SWAP, a Low Start PWMxUy, and High Start PWMxDy).

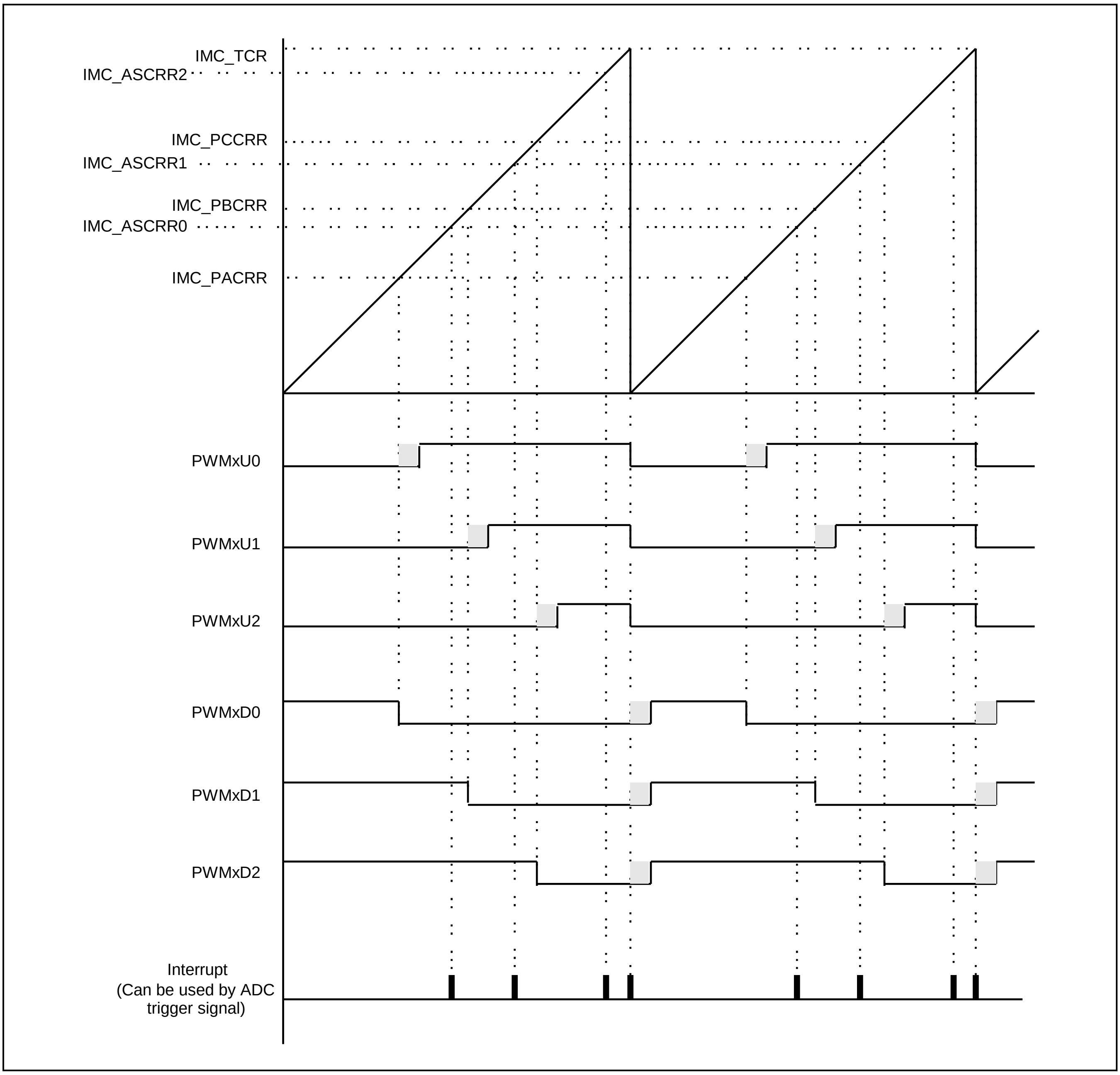


Figure 10-15 Saw-Tooth Wave (No SWAP, a Low Start PWMxUy, and High Start PWMxDy)

NOTE:

1. Both the switches of upside and down side are high active.
2. For 100 % duty of upside, you should set the rising/falling compare register to "0".
   For 0 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

Figure 10-16 illustrates the Saw-Tooth wave duty (No SWAP, a Low Start PWMxUy, and High Start PWMxDy).

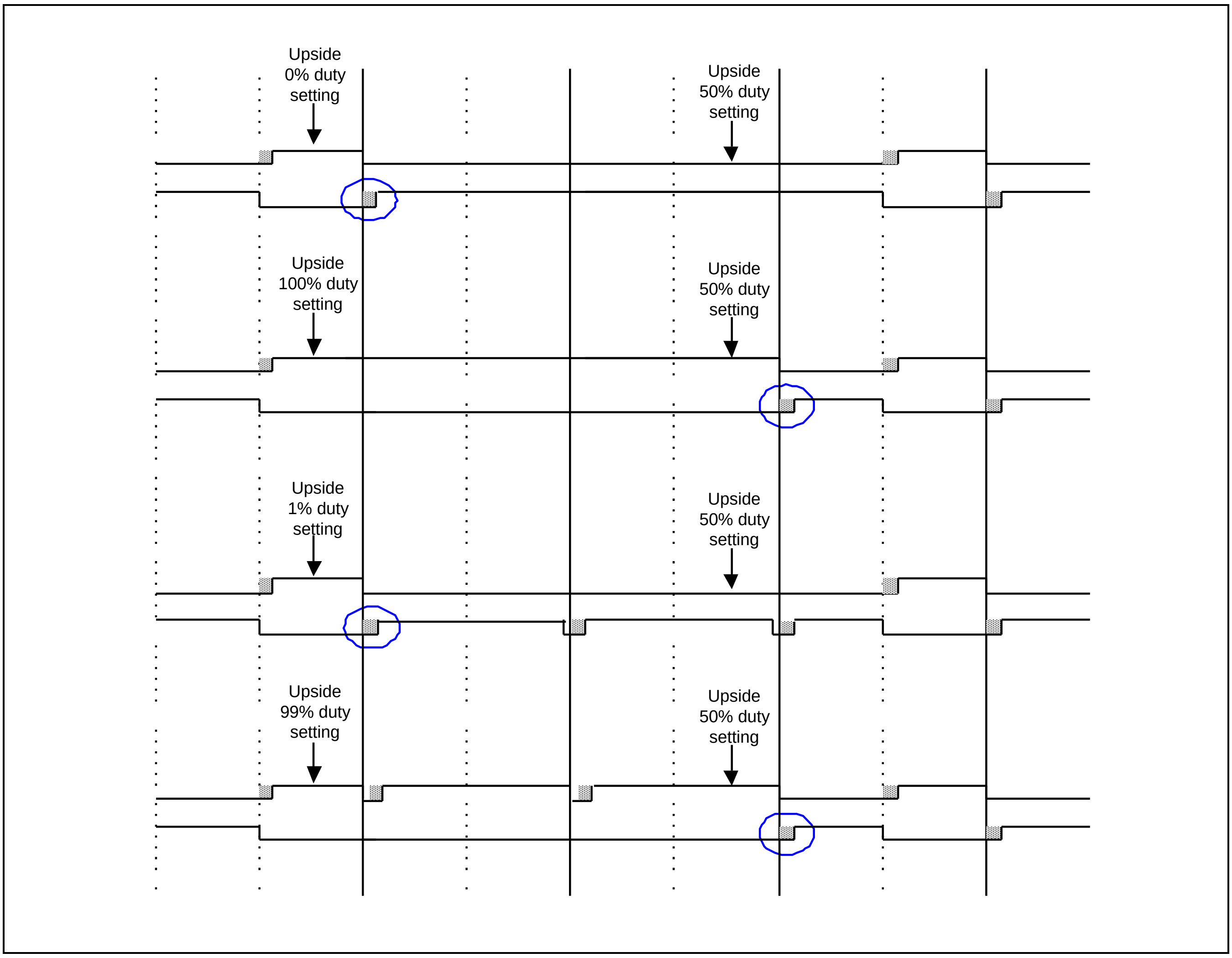


Figure 10-16 Saw-Tooth Wave Duty (No SWAP, a Low Start PWMxUy, and High Start PWMxDy)

10.2.3.10 Saw-Tooth Wave (IMMODE = 1)

PWMSWAP = 1, PWMPOLU = 0 (Low Start), and PWMPOLD = 1 (High Start).

Figure 10-17 illustrates the Saw-Tooth wave (SWAP, a Low Start PWMxUy, and High Start PWMxDy).

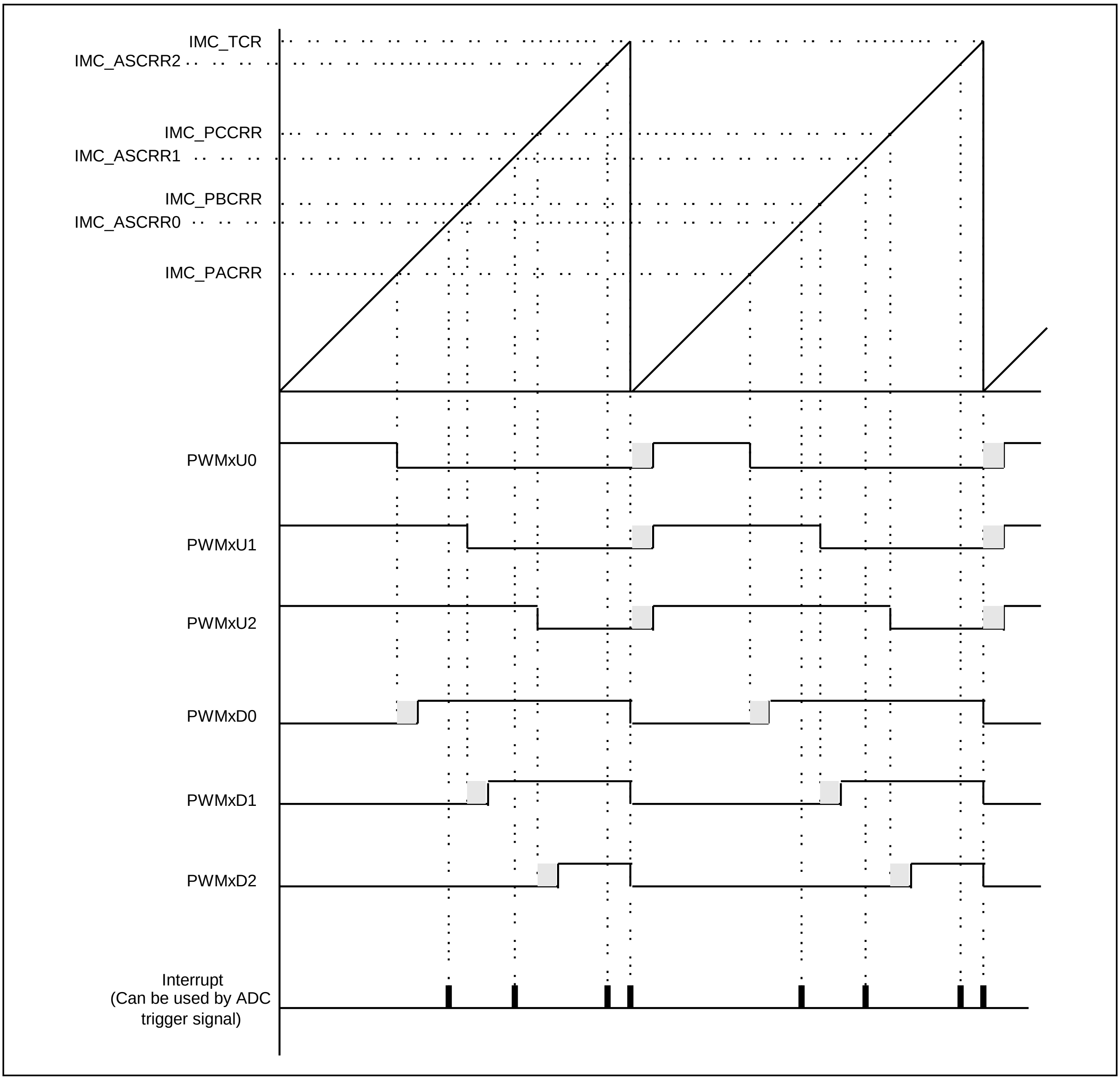


Figure 10-17 Saw-Tooth Wave (SWAP, a Low Start PWMxUy, and High Start PWMxDy)

NOTE:

1. Both the switches of upside and down side are high active.
2. For 0 % duty of upside, you should set the rising/falling compare register to "0".
   For 100 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.11 Saw-Tooth Wave (IMMODE = 1)

PWMSWAP = 0, PWMPOLU = 0 (Low Start), and PWMPOLD = 0 (Low Start)

Figure 10-18 illustrates the Saw-Tooth wave (No SWAP, a Low Start PWMxUy, and Low Start PWMxDy).

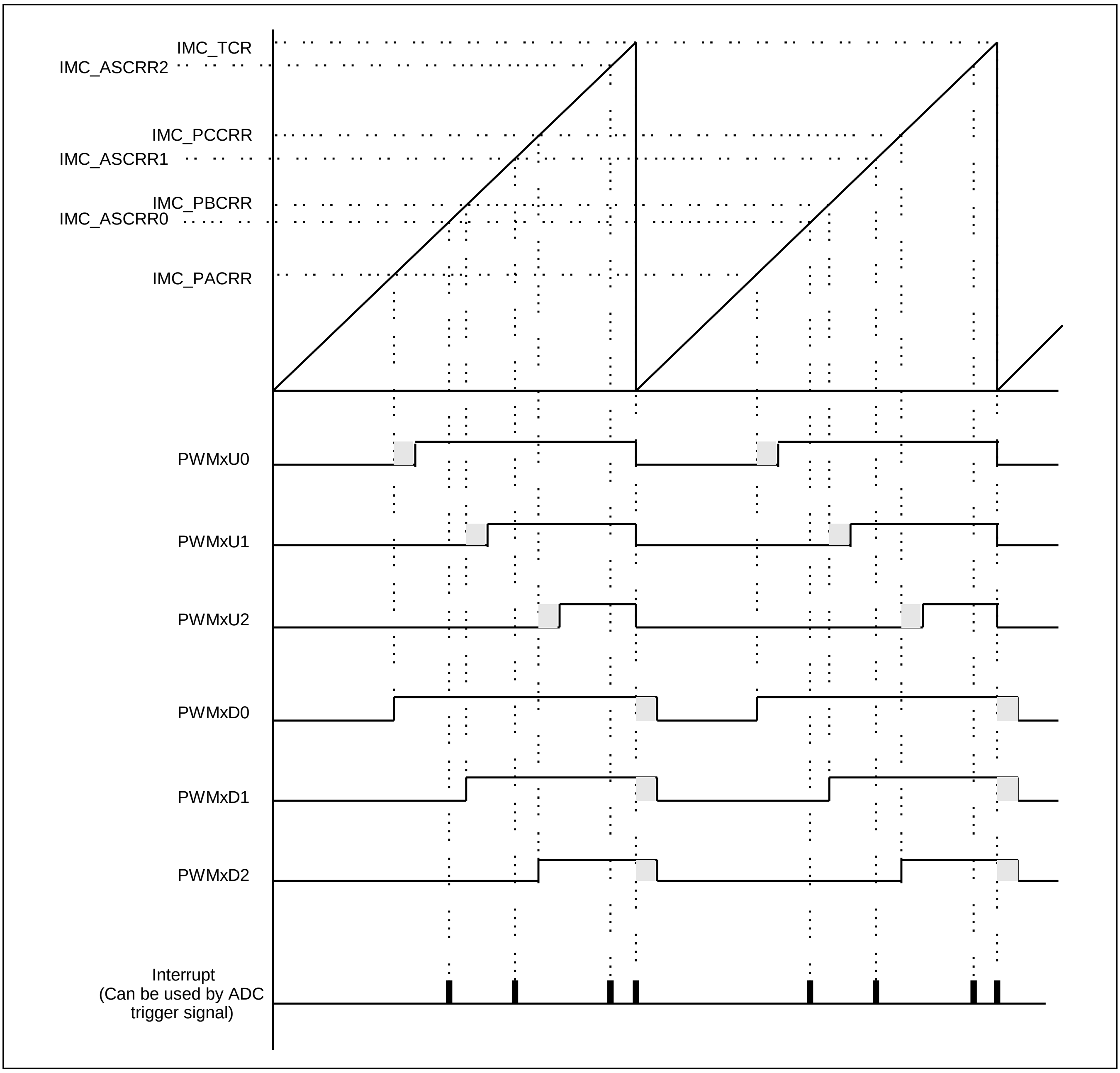


Figure 10-18 Saw-Tooth Wave (No SWAP, a Low Start PWMxUy, and Low Start PWMxDy)

NOTE:

1. The switch of upside is high active and the switch of down side is low active.
2. For 100 % duty of upside, you should set the rising/falling compare register to "0".
   For 0 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.12 Saw-Tooth Wave (IMMODE = 1)

PWMSWAP = 1, PWMPOLU = 0 (Low Start), and PWMPOLD = 0 (Low Start)

Figure 10-19 illustrates the Saw-Tooth wave (SWAP, a Low Start PWMxUy, and Low Start PWMxDy).

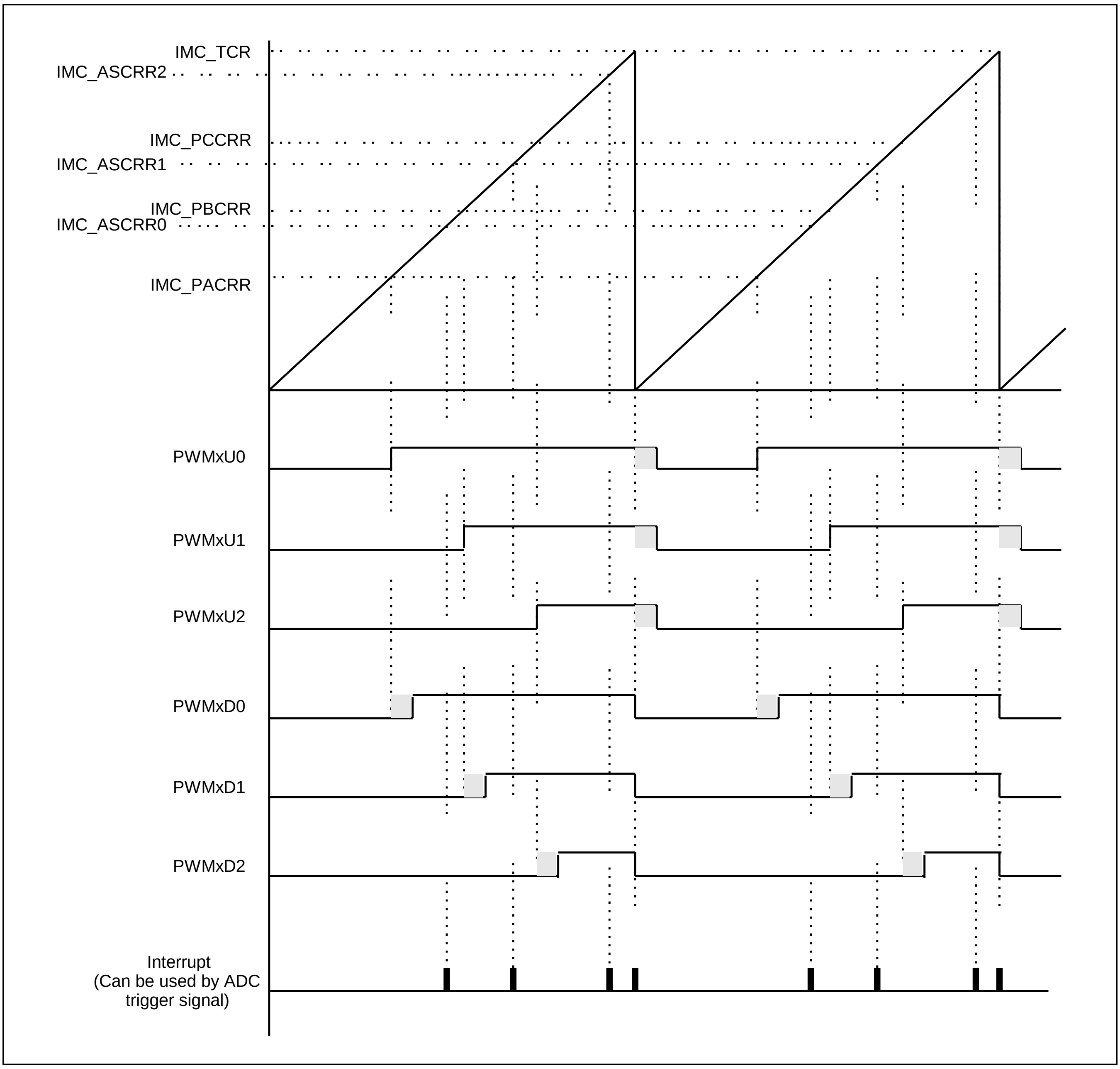


Figure 10-19 Saw-Tooth Wave (SWAP, a Low Start PWMxUy, and Low Start PWMxDy)

NOTE:

1. The switch of upside is low active and the switch of down side is high active.
2. For 0 % duty of upside, you should set the rising/falling compare register to "0".
   For 100 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.13 Saw-Tooth Wave (IMMODE = 1)

PWMSWAP = 0, PWMPOLU = 1 (High Start), and PWMPOLD = 0 (Low Start)

Figure 10-20 illustrates the Saw-Tooth wave (No SWAP, a High Start PWMxUy, and Low Start PWMxDy).

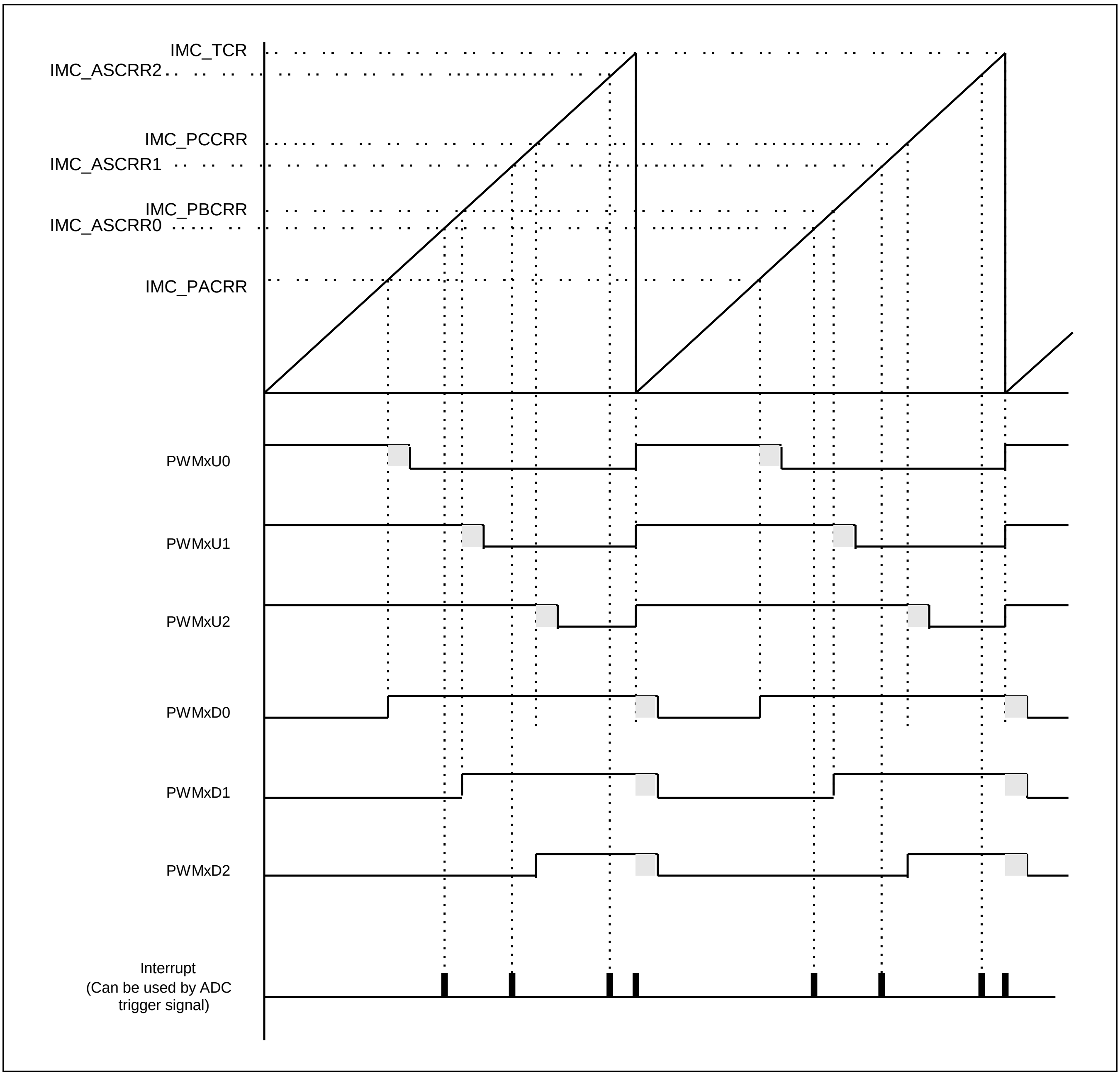


Figure 10-20 Saw-Tooth Save (No SWAP, a High Start PWMxUy, and Low Start PWMxDy)

NOTE:

1. Both the switches of upside and down side are low active.
2. For 100 % duty of upside, you should set the rising/falling compare register to "0".
   For 0 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.14 Saw-Tooth Wave (IMMODE = 1)

PWMSWAP = 1, PWMPOLU = 1 (High Start), and PWMPOLD = 0 (Low Start)

Figure 10-21 illustrates the Saw-Tooth wave (SWAP, a High Start PWMxUy, and Low Start PWMxDy).

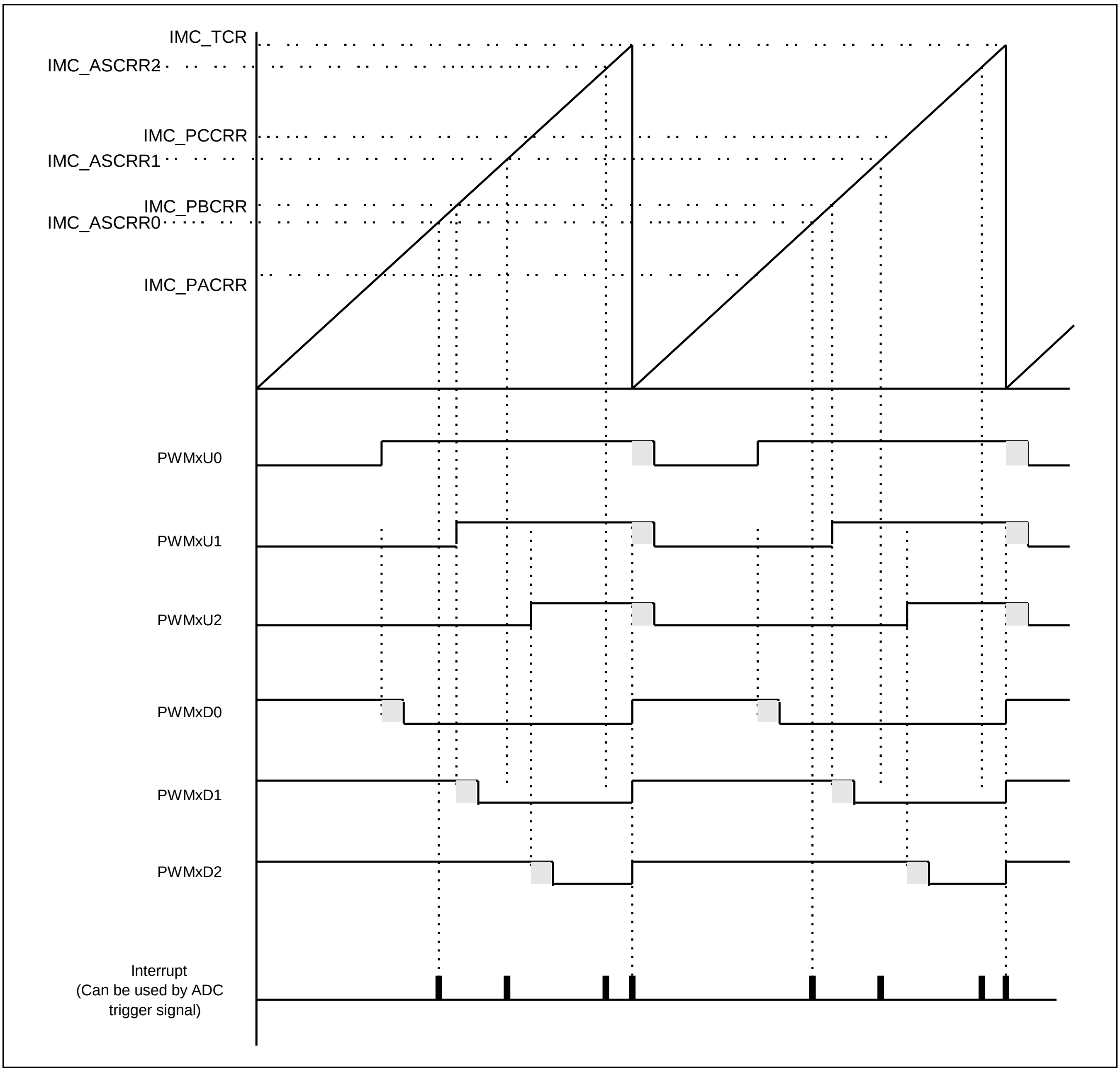


Figure 10-21 Saw-Tooth Wave (SWAP, a High Start PWMxUy, and Low Start PWMxDy)

NOTE:

1. Both the switches of upside and down side are low active.
2. For 0 % duty of upside, you should set the rising/falling compare register to "0".
   For 100 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.15 Saw-Tooth Wave (IMMODE = 1)

PWMSWAP = 0, PWMPOLU = 1 (High Start), and PWMPOLD = 1 (High Start)

Figure 10-22 illustrates the Saw-Tooth wave (No SWAP, a High Start PWMxUy, and High Start PWMxDy).

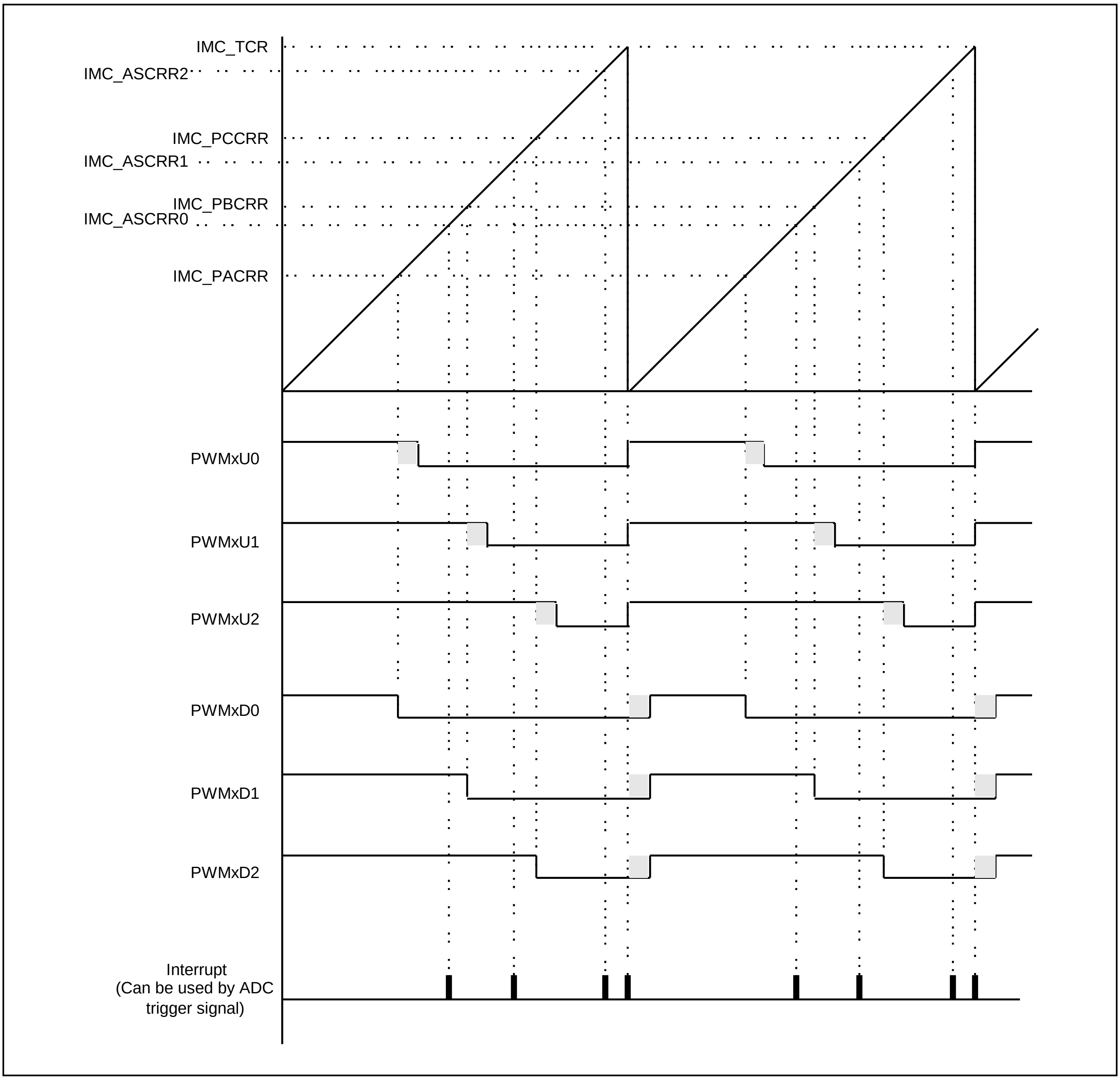


Figure 10-22 Saw-Tooth Wave (No SWAP, a High Start PWMxUy, and High Start PWMxDy)

NOTE:

1. The switch of upside is low active and the switch of down side is high active.
2. For 100 % duty of upside, you should set the rising/falling compare register to "0".
   For 0 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.2.3.16 Saw-Tooth Wave (IMMODE = 1)

PWMSWAP = 1, PWMPOLU = 1 (High Start), and PWMPOLD = 1 (High Start)

Figure 10-23 illustrates the Saw-Tooth wave (SWAP, a High Start PWMxUy, and High Start PWMxDy).

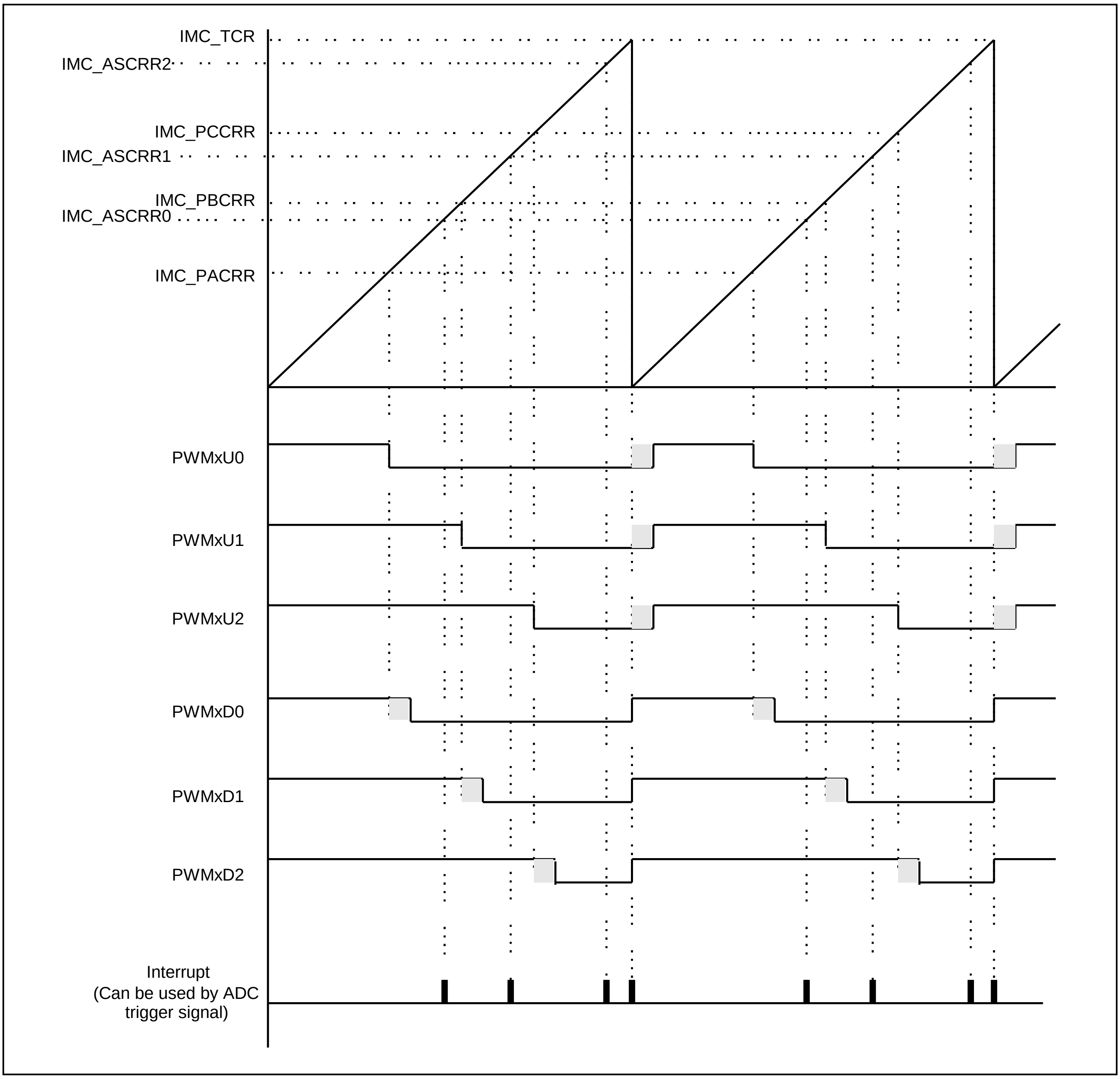


Figure 10-23 Saw-Tooth Wave (SWAP, a High Start PWMxUy, and High Start PWMxDy)

NOTE:

1. The switch of upside is high active and the switch of down side is low active.
2. For 0 % duty of upside, you should set the rising/falling compare register to "0".
   For 100 % duty of upside, you should set the rising compare register to equal to TOPCMP value.

10.3 Register Description

10.3.1 Register Map Summary

- Base Address: 0x400B_0000

Register	Offset	Description	Reset Value
IMC_IDR	0x0000	IMC ID register	0x0001_0012
IMC_CEDR	0x0004	IMC clock enable/disable register	0x0000_0000
IMC_SRR	0x0008	IMC software reset register	0x0000_0000
IMC_CR0	0x000C	IMC control register 0	0x0000_0000
IMC_CR1	0x0010	IMC control register 1	0x0000_0000
IMC_CNTR	0x0014	IMC counter register	0x0000_0000
IMC_SR	0x0018	IMC status register	0x0000_0000
IMC_IMSCR	0x001C	IMC interrupt mask set/clear register	0x0000_0000
IMC_RISR	0x0020	IMC raw interrupt status register	0x0000_0000
IMC_MISR	0x0024	IMC masked interrupt status register	0x0000_0000
IMC_ICR	0x0028	IMC interrupt clear register	0x0000_0000
IMC_TCR	0x002C	16-bit top compare register	0x0000_0000
IMC_DTCR	0x0030	16-bit dead time control register	0x0000_0000
IMC_PACRR	0x0034	IMC 16-bit phase A compare rising register	0x0000_0000
IMC_PBCRR	0x0038	IMC 16-bit phase B compare rising register	0x0000_0000
IMC_PCCRR	0x003C	IMC 16-bit phase C compare rising register	0x0000_0000
IMC_PACFR	0x0040	IMC 16-bit phase A compare falling register	0x0000_0000
IMC_PBCFR	0x0044	IMC 16-bit phase B compare falling register	0x0000_0000
IMC_PCCFR	0x0048	IMC 16-bit phase C compare falling register	0x0000_0000
IMC_ASTSR	0x004C	IMC ADC start trigger selection register	0x0000_0000
IMC_ASCRR0	0x0050	16-bit ADC start compare rising register 0	0x0000_0000
IMC_ASCRR1	0x0054	16-bit ADC start compare rising register 1	0x0000_0000
IMC_ASCRR2	0x0058	16-bit ADC start compare rising register 2	0x0000_0000
IMC_ASCFR0	0x005C	16-bit ADC start compare falling register 0	0x0000_0000
IMC_ASCFR1	0x0060	16-bit ADC start compare falling register 1	0x0000_0000
IMC_ASCFR2	0x0064	16-bit ADC start compare falling register 2	0x0000_0000

10.3.1.1 IMC_IDR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0000, Reset Value = 0x0001_0012

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores the ID code for the corresponding IP.	0x0001_0012

10.3.1.2 IMC_CEDR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	RW	Debug Enable Bit 0 = IMC is not halted during the processor debug mode 1 = IMC is halted during the processor debug mode	0'b
RSVD	[30:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable/Disable Control Bit 0 = Disables IMC Clock 1 = Enables IMC Clock	0'b

10.3.1.3 IMC_SRR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Performs IMC Software Reset operation	0'b

10.3.1.4 IMC_CR0

- Base Address: 0x400B_0000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD	PCCRWM	PBCRWM	PACRWM	SYNCSEL		RSVD	NUMSKIP					RSVD	IMCLKSEL			PWMOUTOFFENBYCOMP	PWMOUTEN	PWMOUTOFFEN	PWMOFFEN	RSVD	IMFILTER			ESELPWMOFF		PWMPOLD	PWMPOLU	PWMSWAP	WMODE	IMMODE	IMEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31]	R	Reserved	0
PCCRWM	[30]	RW	Phase C Compare Register Write Mode Selection Bit 0 = Common Write mode PCCRR value written by the user is copied to the PCCFR field simultaneously. User will use only PCCRR register. 1 = Separate Write modeUser should write both PCCRR and PCCFR register to each proper value.	0'b
PBCRWM	[29]	RW	Phase B Compare Register Write Mode Selection Bit 0 = Common Write mode PBCRR value written by the user is copied to the PBCFR field simultaneously. User will use only PBCRR register. 1 = Separate Write mode User should write both PBCRR and PBCFR register to each proper value.	0'b
PACRWM	[28]	RW	Phase A Compare Register Write Mode Selection Bit 0 = Common Write mode User written PACRR value copies to the PACFR field simultaneously. You can use only PACRR register. 1 = Separate Write mode You should write PACRR and PACFR registers with proper value each by each.	0'b
SYNCSEL	[27:26]	RW	Synchronous Write Selection This field determines the timing of synchronous write. 00 = Synchronous write at counter matches ZERO and	00'b

Name	Bit	Type	Description	Reset Value
			IMC_TCR 01 = Synchronous write at counter matches ZERO 10 = Synchronous write at counter matches TOPCMP 11 = Should not use	
RSVD	[25]	R	Reserved	0
NUMSKIP	[24:20]	RW	Numbers of Skip for Motor Match Interrupt This field determines the number of skips for motor match interrupt and ADC trigger signal. The unit of skip is PWM full cycle. 00000 = No skip 00001 = 1 Time skip 00010 = 2 Times skip 00011 = 3 Times skip 00100 = 4 Times skip … 11100 = 28 Times skip 11101 = 29 Times skip 11110 = 30 Times skip 11111 = 31 Times skip	0000'b
RSVD	[19]	R	Reserved	0
IMCLKSEL	[18:16]	RW	Inverter Clock (IMCLK) Selection This field determines the inverter motor clock selection. Also you can select the clock source (IMCLK, EMCLK, PLLCLK, or PCLK) in a Clock Manager. FIN means the frequency of the clock source that Clock Manager defines. 000 = FIN 001 = FIN/2 010 = FIN/4 011 = FIN/8 100 = FIN/16 101 = FIN/32 110 = FIN/64 111 = FIN/128 NOTE: The clock source of dead time compare register is IMCLK.	000'b
PWMOUTOFFEN BYCOMP	[15]	RW	PWM Output off Enable by Comparator This field determines whether the PWM output signal is enabled/disabled by detecting edge in the comparator block. The PWM output goes to High-Z state if this bit is set to "1". 0 = Enables PWM output signal 1 = Disables PWM output signal	0
PWMOUTEN	[14]	RW	PWM Output Enable Bit This field determines whether the PWM output signal is enabled/disabled. The PWM output goes to High-Z state when this bit is set to "1". You can use this bit	0

Name	Bit	Type	Description	Reset Value
			during debugging. 0 = Enables PWM output signal 1 = Disables PWM output signal	
PWMOUTOFFEN	[13]	RW	PWM Output Disable by PWMxOFF This field determines the PWM output disable by PWMxOFF. 0 = Disables PWM output disable by PWMxOFF 1 = Enables PWM output disable by PWMxOFF NOTE: If this bit is set to "1" and PWMxOFF condition is met, then the PWM output goes to High-Z state.	0
PWMOFFEN	[12]	RW	PWMxOFF Enable Bit This field determines the PWMxOFF enable/disable. 0 = Disables fault detection of PWMxOFF 1 = Enables fault detection of PWMxOFF	0
RSVD	[11]	R	Reserved	0
IMFILTER	[10:8]	RW	Filter Clock Selection of PWMxOFF Pin This field determines the filter clock selection of IMC. 000 = IMCLK 001 = IMCLK/2 010 = IMCLK/4 011 = IMCLK/8 100 = IMCLK/16 101 = IMCLK/32 110 = IMCLK/64 111 = IMCLK/128 NOTE: Only six times same level in a row is recognized as effective signal.	000'b
ESELPWMOFF	[7:6]	RW	PWMxOFF Active Selection This field determines the active selection for PWM0OFF. 00 = Falling Edge 01 = Rising Edge 10 = Low Level 11 = High Level NOTE: You can change these bits only when IMCON.0 is 0.	00'b
PWMPOLD	[5]	RW	Polarity of PWM in the PWMxD0/1/2 This field determines the polarity of PWM signal in the PWMxD0/1/2. 0 = Low Start 1 = High Start	0
PWMPOLU	[4]	RW	Polarity of PWM in the PWMxU0/1/2 This field determines the polarity of PWM signal in the PWMxU0/1/2. 0 = Low Start 1 = High Start	0

Name	Bit	Type	Description	Reset Value
PWMSWAP	[3]	RW	Swapping of PWMxUx and PWMxDx This field determines swapping of PWMxUx and PWMxDx. 0 = No swap 1 = Swap NOTE: You can change this bit only when IMCON.0 is 0.	0
WMODE	[2]	RW	Write Mode of Compare Register This field determines the Write mode of all compare registers. 0 = Immediate write 1 = Synchronous write NOTE: In the synchronous write, if IMCNT is equal to 0 or TOPCMP, compare registers including dead time compare register which are written are updated simultaneously. Synchronous write relates to NUMSKIP. For example, if NUMSKIP is 30, then the synchronous write happens only once in every 30 times. ADC trigger signal is in the same situation.	0
IMMODE	[1]	RW	Inverter Motor Mode Selection Bit 0 = Tri-Angular shape 1 = Saw-Tooth shape You can change this bit only when IMC_CR0.0 is 0. If this bit is set to "1", then the comparison with "0" has no effect (INT_ZEROx will not be occurred.)	0
IMEN	[0]	RW	Inverter Motor Block Enable Bit This field determines the inverter motor block enable/disable. 0 = Disables inverter motor block 1 = Enables inverter motor block NOTE: If this bit is set to "0", then the IMCNT bit is automatically cleared to "0".	0

NOTE: If IMEN is equal to 0, all PWM output (PWMxU/Dx) goes to High-Z state.

Figure 10-24 illustrates the synchronous write at zero and IMC_TCR match.

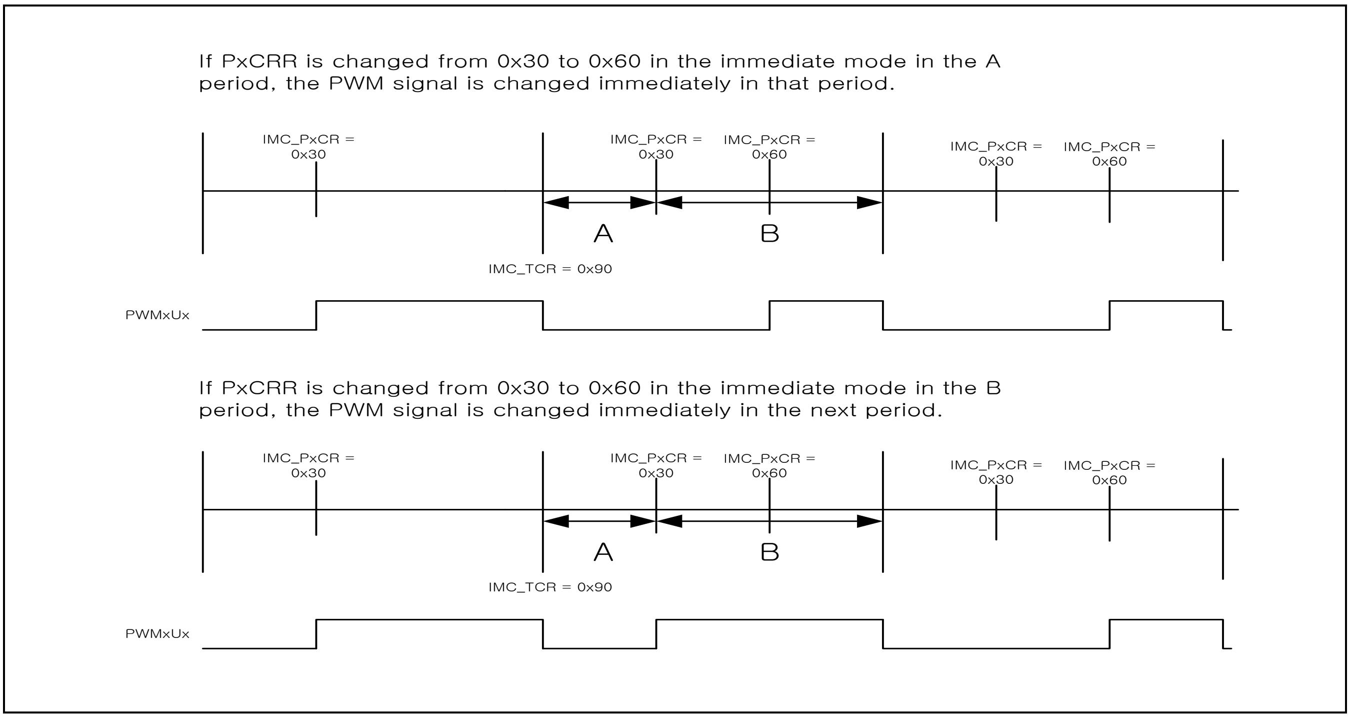

If PxCRR is changed from 0x30 to 0x60 in the immediate mode in the A period, the PWM signal is changed immediately in that period.
IMC_PxCR = 0x30
IMC_PxCR = 0x30
IMC_PxCR = 0x60
IMC_PxCR = 0x30
IMC_PxCR = 0x60
A
B
IMC_TCR = 0x90
PWMxUx
If PxCRR is changed from 0x30 to 0x60 in the immediate mode in the B period, the PWM signal is changed immediately in the next period.
IMC_PxCR = 0x30
IMC_PxCR = 0x30
IMC_PxCR = 0x60
IMC_PxCR = 0x30
IMC_PxCR = 0x60
A
B
IMC_TCR = 0x90
PWMxUx

Figure 10-24 Synchronous Write at Zero and IMC_TCR Match (SYNCSEL = 00'b, NUMSKIP = 00000'b)

NOTE: If the immediate Write Mode (WMODE) is equal to 0 and Saw-Tooth mode (IMMODE) is equal to 1, then the compare register update operation looks like the illustration in *Figure 10-25*.

Figure 10-25 illustrates the synchronous Write at Zero and IMC_TCR Match (SYNCSEL = 00'b, NUMSKIP = 00000'b).

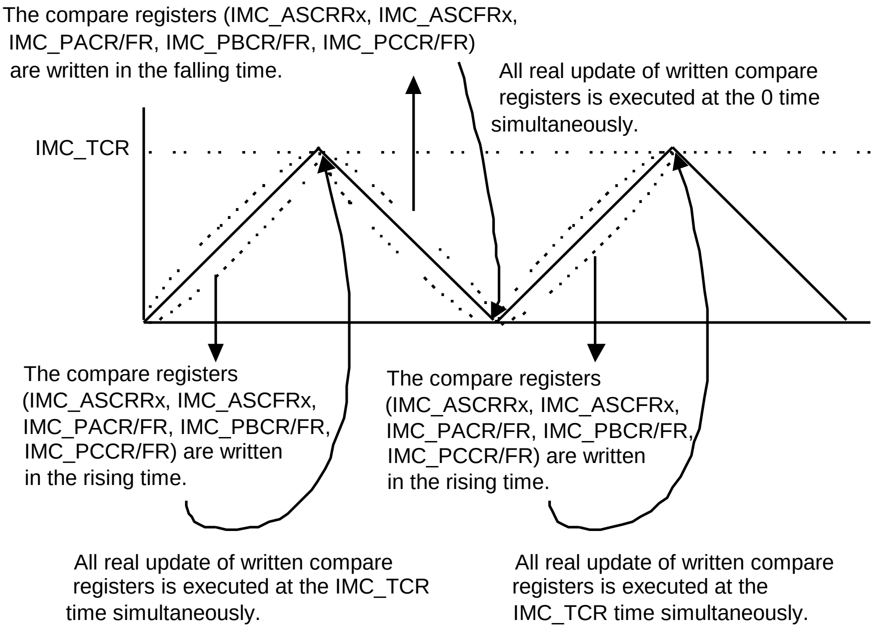

The compare registers (IMC_ASCRRx, IMC_ASCFRx, IMC_PACR/FR, IMC_PBCR/FR, IMC_PCCR/FR) are written in the falling time.
All real update of written compare registers is executed at the 0 time simultaneously.
IMC_TCR
The compare registers (IMC_ASCRRx, IMC_ASCFRx, IMC_PACR/FR, IMC_PBCR/FR, IMC_PCCR/FR) are written in the rising time.
The compare registers (IMC_ASCRRx, IMC_ASCFRx, IMC_PACR/FR, IMC_PBCR/FR, IMC_PCCR/FR) are written in the rising time.
All real update of written compare registers is executed at the IMC_TCR time simultaneously.
All real update of written compare registers is executed at the IMC_TCR time simultaneously.

Figure 10-25 Synchronous Write at Zero and IMC_TCR Match (SYNCSEL = 00'b, NUMSKIP = 00000'b)

NOTE: If WMODE is equal to 1 and NUMSKIP is equal to 0, then the update of compare registers looks like the illustration in Figure 10-26. If NUMSKIP is 0, then the written compare registers are updated every IMC_TCR and 0 time.

Figure 10-26 illustrates the synchronous Write at Zero Match (SYNCSEL = 01'b, NUMSKIP = 00000'b).

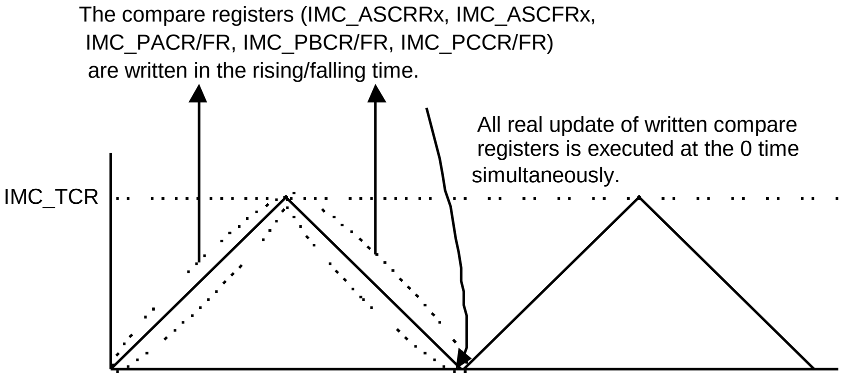

The compare registers (IMC_ASCRRx, IMC_ASCFRx, IMC_PACR/FR, IMC_PBCR/FR, IMC_PCCR/FR) are written in the rising/falling time.
All real update of written compare registers is executed at the 0 time simultaneously.
IMC_TCR

Figure 10-26 Synchronous Write at Zero Match (SYNCSEL = 01'b, NUMSKIP = 00000'b)

Figure 10-27 illustrates the synchronous write at IMC_TCR match.

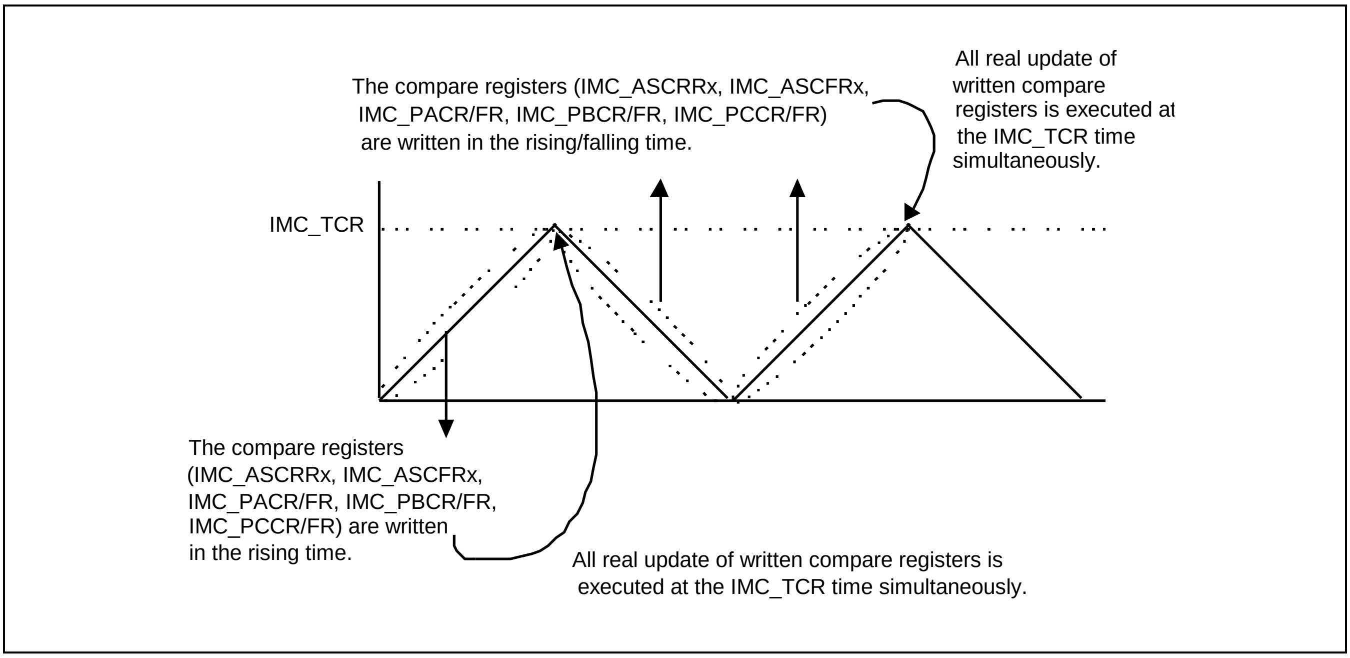

The compare registers (IMC_ASCRRx, IMC_ASCFRx, IMC_PACR/FR, IMC_PBCR/FR, IMC_PCCR/FR) are written in the rising/falling time.
All real update of written compare registers is executed at the IMC_TCR time simultaneously.
IMC_TCR
The compare registers (IMC_ASCRRx, IMC_ASCFRx, IMC_PACR/FR, IMC_PBCR/FR, IMC_PCCR/FR) are written in the rising time.
All real update of written compare registers is executed at the IMC_TCR time simultaneously.

Figure 10-27 Synchronous Write at IMC_TCR Match (SYNCSEL = 10'b, NUMSKIP = 00000'b)

NOTE: If WMODE is equal to 1 and NUMSKIP is equal to 1, then the update of compare registers look like the illustration in Figure 10-28. If NUMSKIP is 1, then the written compare registers are updated once per two IMC_TCR and 0 time. Second and fourth pulse is the skipped pulse.

Figure 10-28 illustrates the synchronous Write at Zero and IMC_TCR Match (SYNCSEL = 00'b, NUMSKIP = 00001'b).

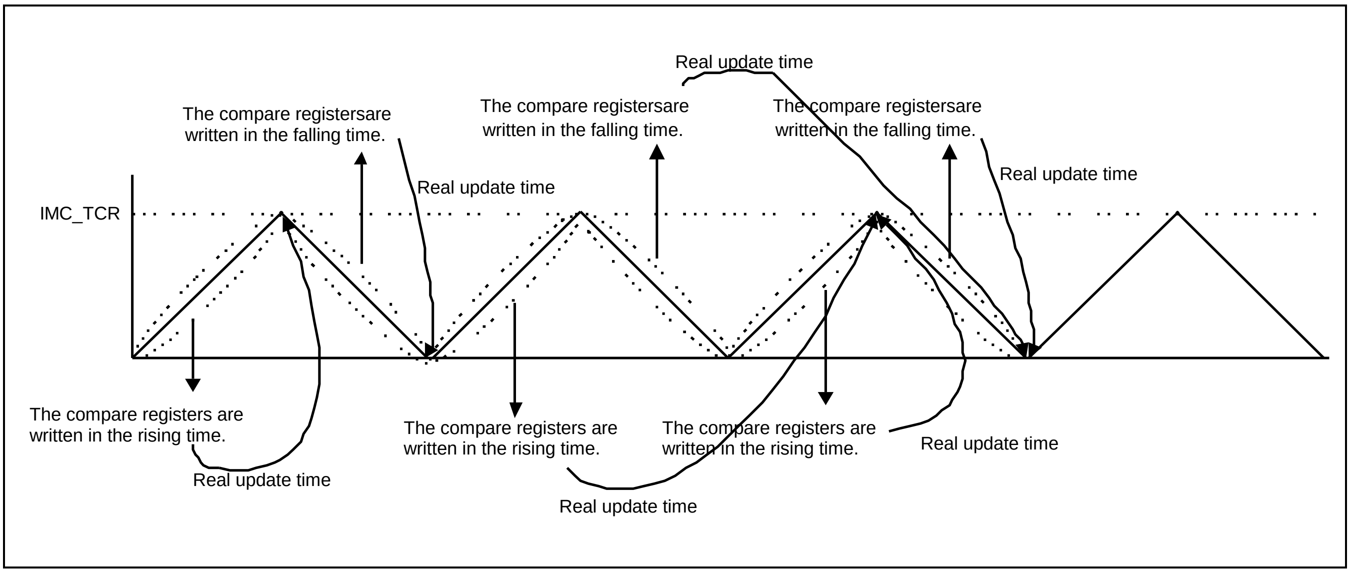

Real update time
The compare registersare written in the falling time.
The compare registersare written in the falling time.
The compare registersare written in the falling time.
Real update time
Real update time
IMC_TCR
The compare registers are written in the rising time.
Real update time
The compare registers are written in the rising time.
The compare registers are written in the rising time.
Real update time
Real update time

Figure 10-28 Synchronous Write at Zero and IMC_TCR Match (SYNCSEL = 00'b, NUMSKIP = 00001'b)

Figure 10-29 illustrates the synchronous write at zero match.

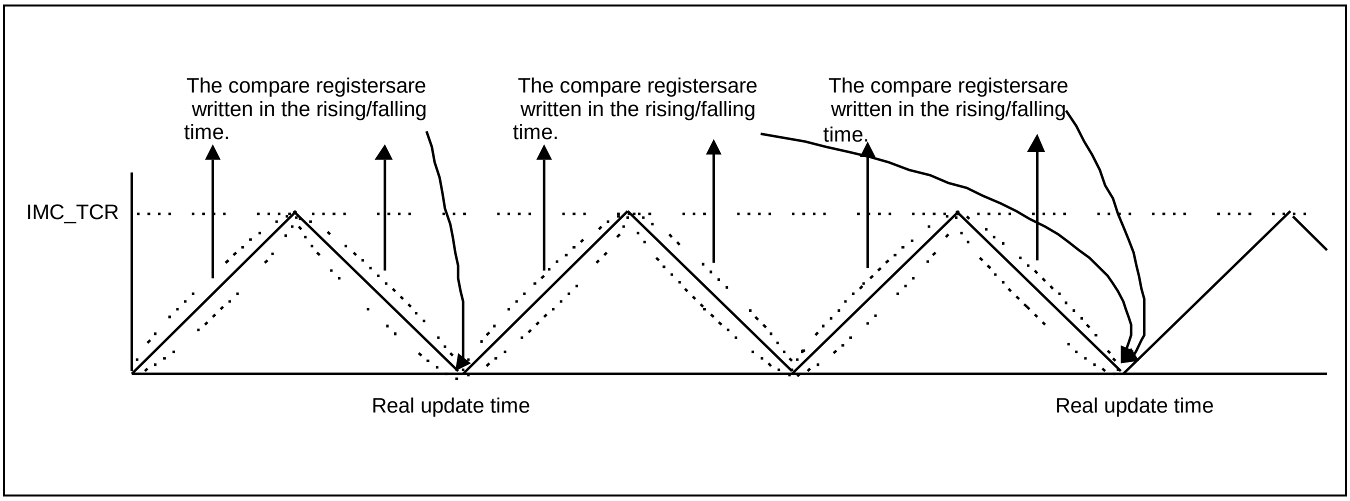

The compare registersare written in the rising/falling time.
The compare registersare written in the rising/falling time.
The compare registersare written in the rising/falling time.
IMC_TCR
Real update time
Real update time

Figure 10-29 Synchronous Write at Zero Match (SYNCSEL = 01'b, NUMSKIP = 00001'b)

Figure 10-30 illustrates the synchronous write at IMC_TCR match.

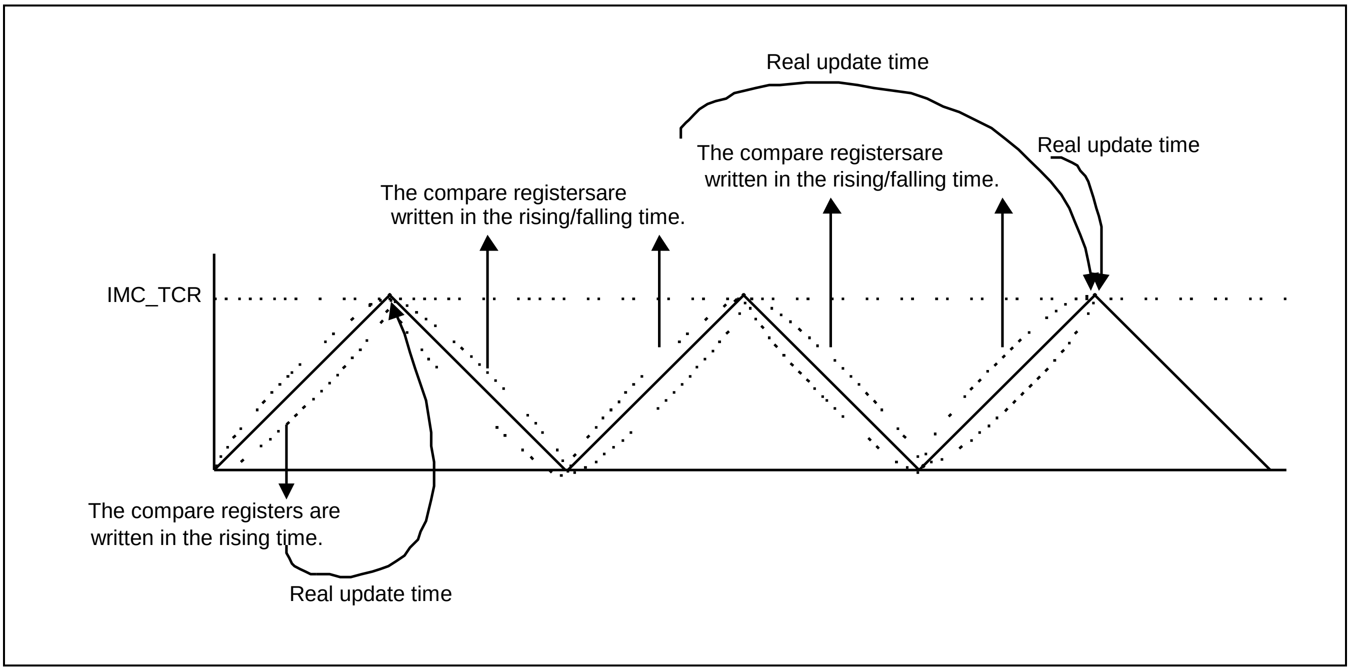

Real update time
The compare registersare written in the rising/falling time.
Real update time
The compare registersare written in the rising/falling time.
IMC_TCR
The compare registers are written in the rising time.
Real update time

Figure 10-30 Synchronous Write at IMC_TCR Match (SYNCSEL = 10'b, NUMSKIP = 00001'b)

NOTE: The value of NUMSKIP affects interrupt also. If IMC_ASCRRx/FRx is set to interrupt source and the value of NUMSKIP is 1, then the interrupt does not occur in the second and fourth pulse.

Figure 10-31 illustrates the skip control of ADC trigger signal interrupt.

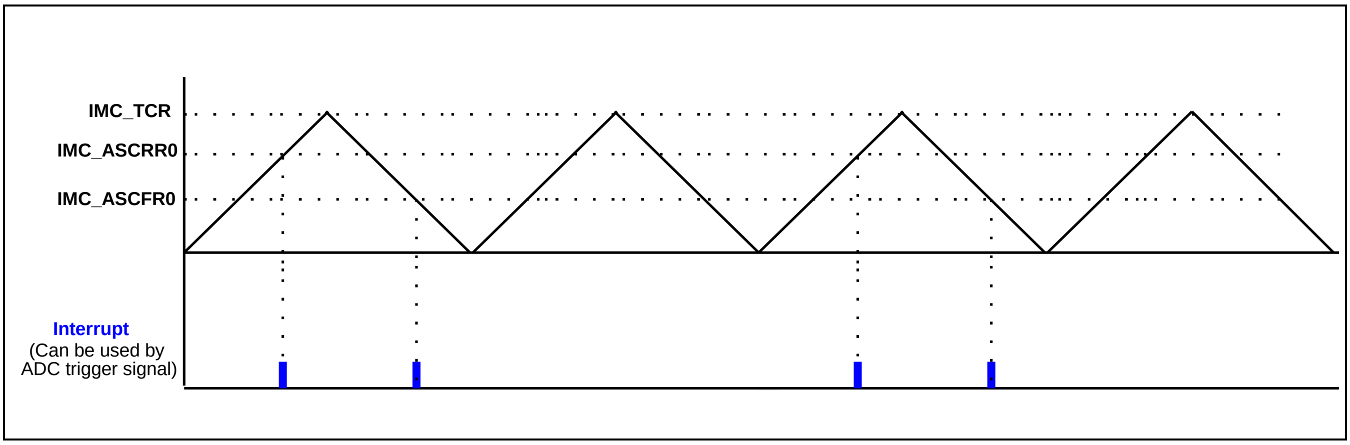


Figure 10-31 Skip Control of ADC Trigger Signal Interrupt

10.3.1.5 IMC_CR1

- Base Address: 0x400B_0000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD										PWMxU0DT	PWMxU1DT	PWMxU2DT	PWMxD0DT	PWMxD1DT	PWMxD2DT	RSVD		PWMxU0 LEVEL	PWMxU1 LEVEL	PWMxU2LEVEL	PWMxD0 LEVEL	PWMxD1 LEVEL	PWMxD2LEVEL	RSVD		PWMxU0EN	PWMxU1EN	PWMxU2EN	PWMxD0EN	PWMxD1EN	PWMxD2EN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R	R	R W	R W	R W	R W	R W	R W	R	R	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:22]	R	Reserved	0
PWMxU0DT	[21]	RW	PWMxU0 Dead Time Insert Bit This bit determines whether dead time is inserted or not before PWM output disable by setting PWMxU0EN. 0 = Does not insert dead time 1 = Inserts dead time	0
PWMxU1DT	[20]	RW	PWMxU1 Dead Time Insert Bit This bit determines whether dead time is inserted or not before PWM output disable by setting PWMxU1EN. 0 = Does not insert dead time 1 = Inserts dead time	0
PWMxU2DT	[19]	RW	PWMxU2 Dead Time Insert Bit This bit determines whether dead time is inserted or not before PWM output disable by setting PWMxU2EN. 0 = Does not insert dead time 1 = Inserts dead time	0
PWMxD0DT	[18]	RW	PWMxD0 Dead Time Insert Bit This bit determines whether dead time is inserted or not before PWM output disable by setting PWMxD0EN. 0 = Does not insert dead time 1 = Inserts dead time	0
PWMxD1DT	[17]	RW	PWMxD1 Dead Time Insert Bit This bit determines whether dead time is inserted or not before PWM output disable by setting PWMxD1EN. 0 = Does not insert dead time 1 = Inserts dead time	0
PWMxD2DT	[16]	RW	PWMxD2 Dead Time Insert Bit This bit determines whether dead time is inserted or not	0

Name	Bit	Type	Description	Reset Value
			before PWM output disable by setting PWMxD2EN. 0 = Does not insert dead time 1 = Inserts dead time	
RSVD	[15:14]	R	Reserved	0
PWMxU0 LEVEL	[13]	RW	PWMxU0 Output Level Selection Bit 0 = Low Level 1 = High Level	0
PWMxU1 LEVEL	[12]	RW	PWMxU1 Output Level Selection Bit 0 = Low Level 1 = High Level	0
PWMxU2LEVEL	[11]	RW	PWMxU2 Output Level Selection Bit 0 = Low Level 1 = High Level	0
PWMxD0 LEVEL	[10]	RW	PWMxD0 Output Level Selection Bit 0 = Low Level 1 = High Level	0
PWMxD1 LEVEL	[9]	RW	PWMxD1 Output Level Selection Bit 0 = Low Level 1 = High Level	0
PWMxD2LEVEL	[8]	RW	PWMxD2 Output Level Selection Bit 0 = Low Level 1 = High Level	0
RSVD	[7:6]	R	Reserved	0
PWMxU0EN	[5]	RW	PWMxU0 PWM Output Enable Bit 0 = Enables PWM signal to PWMxU0 1 = Disables PWM signal to PWMxU0 → IMCON1.13 determines the level of PWMxU0	0
PWMxU1EN	[4]	RW	PWMxU1 PWM Output Enable Bit 0 = Enables PWM signal to PWMxU1 1 = Disables PWM signal to PWMxU1 → IMCON1.12 determines the level of PWMxU1	0
PWMxU2EN	[3]	RW	PWMxU2 PWM Output Enable Bit 0 = Enables PWM signal to PWMxU2 1 = Disables PWM signal to PWMxU2 → IMCON1.11 determines the level of PWMxU2	0
PWMxD0EN	[2]	RW	PWMxD0 PWM Output Enable Bit 0 = Enables PWM signal to PWMxD0 1 = Disables PWM signal to PWMxD0 → IMCON1.10 determines the level of PWMxD2	0
PWMxD1EN	[1]	RW	PWMxD1 PWM Output Enable Bit 0 = Enables PWM signal to PWMxD1 1 = Disables PWM signal to PWMxD1 → IMCON1.9 determines the level of PWMxD1	0

Name	Bit	Type	Description	Reset Value
PWMxD2EN	[0]	RW	PWMxD2 PWM Output Enable Bit 0 = Enables PWM signal to PWMxD2 1 = Disables PWM signal to PWMxD2 → IMCON1.8 determines the level of PWMxD2.	0

NOTE: There is no relationship between level setting in the IMC_CR1 register and SWAP function in the IMC_CR0 register.

10.3.1.6 IMC_CNTR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
CV	[15:0]	R	Count Value This field contains the count value of the current IMC.	0x0000

10.3.1.7 IMC_SR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																													COMPEDGEDET	UPDOWN	FAULTSTAT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	R	Reserved	0
COMPEDGEDET	[2]	RW	Comparator Edge Detect This bit notifies the status of edge detection from comparator block. 0 = Does not detect 1 = Detects NOTE: If it detects the comparator edge, then this bit retains. The software clears this bit.	0
UPDOWN	[1]	R	Status of PWM Counter (Read Only Bit) This bit notifies the status of PWM counter. 0 = Up counting 1 = Down counting NOTE: This bit is always '0' in the Saw-Tooth mode.	0
FAULTSTAT	[0]	RW	Status of PWM Output Signal 0 = Normal operating 1 = High-Z (Inverter motor block is operating but the status of PWM signal is High-Z. This bit can be set by fault detection of PWMxOFF pin or IMC_CR0.14.) NOTE: If you write this bit to 0 and IMC_CR0.14 is 0, then the inverter motor control signal is output to PWM output.	0

10.3.1.8 IMC_IMSCR

- Base Address: 0x400B_0000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																		ADCFM2	ADCRM 2	ADCFM1	ADCRM 1	ADCFM0	ADCRM0	TOP	ZERO	RSVD					FAULT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	R	Reserved	0
ADCFM2	[13]	RW	ADC Compare2 Falling Match Interrupt Mask 0 = This interrupt is masked. (Disables the interrupt) 1 = This interrupt is not masked. (Enables the interrupt)	0
ADCRM2	[12]	RW	ADC Compare2 Rising Match Interrupt Mask 0 = This interrupt is masked. (Disables the interrupt) 1 = This interrupt is not masked. (Enables the interrupt)	0
ADCFM1	[11]	RW	ADC Compare1 Falling Match Interrupt Mask 0 = This interrupt is masked. (Disables the interrupt) 1 = This interrupt is not masked. (Enables the interrupt)	0
ADCRM1	[10]	RW	ADC Compare1 Rising Match Interrupt Mask 0 = This interrupt is masked. (Disables the interrupt) 1 = This interrupt is not masked. (Enables the interrupt)	0
ADCFM0	[9]	RW	ADC Compare0 Falling Match Interrupt Mask 0 = This interrupt is masked. (Disables the interrupt) 1 = This interrupt is not masked. (Enables the interrupt)	0
ADCRM0	[8]	RW	ADC Compare0 Rising Match Interrupt Mask 0 = This interrupt is masked. (Disables the interrupt) 1 = This interrupt is not masked. (Enables the interrupt)	0
TOP	[7]	RW	TOP Match Interrupt Mask 0 = This interrupt is masked. (Disables the interrupt) 1 = This interrupt is not masked. (Enables the interrupt)	0
ZERO	[6]	RW	ZERO Match Interrupt Mask 0 = This interrupt is masked. (Disables the interrupt) 1 = This interrupt is not masked. (Enables the interrupt)	0
RSVD	[5:1]	R	Reserved	0
FAULT	[0]	RW	FAULT Interrupt Mask	0

Name	Bit	Type	Description	Reset Value
			0 = Mask the interrupt (Disables the interrupt) 1 = Unmask the interrupt (Enables the interrupt)	

10.3.1.9 IMC_RISR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																		ADCFM2	ADCRM2	ADCFM1	ADCRM1	ADCFM0	ADCRM0	TOP	ZERO	RSVD					FAULT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	R	Reserved	0
ADCFM2	[13]	R	ADC Compare2 Falling Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of the ADCFM2 interrupt.	0
ADCRM2	[12]	R	ADC Compare2 Rising Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of the ADCRM2 interrupt.	0
ADCFM1	[11]	R	ADC Compare1 Falling Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of the ADCFM1 interrupt.	0
ADCRM1	[10]	R	ADC Compare1 Rising Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of the ADCRM1 interrupt.	0
ADCFM0	[9]	R	ADC Compare0 Falling Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of the ADCFM0 interrupt.	0
ADCRM0	[8]	R	ADC Compare0 Rising Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of the ADCRM0 interrupt.	0
TOP	[7]	R	TOP Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of the TOP interrupt.	0
ZERO	[6]	R	ZERO Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of the ZERO interrupt.	0
RSVD	[5:1]	R	Reserved	0
FAULT	[0]	R	FAULT Raw Interrupt State	0

Name	Bit	Type	Description	Reset Value
			Gives the raw interrupt state (prior to masking) of the FAULT interrupt.	

NOTE:

1. On a Read, the IMC_RISR register gives the current raw status value of the corresponding interrupt prior to masking.
2. A Write has no effect. ZERO is set right after IMC enable.

10.3.1.10 IMC_MISR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0024, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																		ADCFM2	ADCRM2	ADCFM1	ADCRM1	ADCFM0	ADCRM0	TOP	ZERO	RSVD					FAULT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	R	Reserved	0
ADCFM2	[13]	R	ADC Compare2 Falling Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of the ADCFM2 interrupt.	0
ADCRM2	[12]	R	ADC Compare2 Rising Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of the ADCRM2 interrupt.	0
ADCFM1	[11]	R	ADC Compare1 Falling Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of the ADCFM1 interrupt.	0
ADCRM1	[10]	R	ADC Compare1 Rising Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of the ADCRM1 interrupt.	0
ADCFM0	[9]	R	ADC Compare0 Falling Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of the ADCFM0 interrupt.	0
ADCRM0	[8]	R	ADC Compare0 Rising Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of the ADCRM0 interrupt.	0
TOP	[7]	R	TOP Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of the TOP interrupt.	0
ZERO	[6]	R	ZERO Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of the ZERO interrupt.	0
RSVD	[5:1]	R	Reserved	0
FAULT	[0]	R	FAULT Masked Interrupt State	0

Name	Bit	Type	Description	Reset Value
			Gives the masked interrupt state (prior to masking) of the FAULT interrupt.	

NOTE:

1. On a Read, the IMC_MISR register gives the current masked status value of the corresponding interrupt.
2. A Write has no effect.

10.3.1.11 IMC_ICR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																		ADCFM2	ADCRM2	ADCFM1	ADCRM1	ADCFM0	ADCRM0	TOP	ZERO	RSVD					FAULT
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	R	Reserved	0
ADCFM2	[13]	W	ADC Compare2 Falling Match Interrupt Clear 0 = No effect 1 = Clears the ADCFM2 interrupt	0'b
ADCRM2	[12]	W	ADC Compare2 Rising Match Interrupt Clear 0 = No effect 1 = Clears the ADCRM2 interrupt	0'b
ADCFM1	[11]	W	ADC Compare1 Falling Match Interrupt Clear 0 = No effect 1 = Clears the ADCFM1 interrupt	0'b
ADCRM1	[10]	W	ADC Compare1 Rising Match Interrupt Clear 0 = No effect 1 = Clears the ADCRM1 interrupt	0'b
ADCFM0	[9]	W	ADC Compare0 Falling Match Interrupt Clear 0 = No effect 1 = Clears the ADCFM0 interrupt	0'b
ADCRM0	[8]	W	ADC Compare0 Rising Match Interrupt Clear 0 = No effect 1 = Clears the ADCRM0 interrupt	0'b
TOP	[7]	W	TOP Match Interrupt Clear 0 = No effect 1 = Clears the TOP interrupt	0'b
ZERO	[6]	W	ZERO Match Interrupt Clear 0 = No effect 1 = Clears the ZERO interrupt	0'b
RSVD	[5:1]	R	Reserved	0
FAULT	[0]	W	FAULT Interrupt Clear	0'b

Name	Bit	Type	Description	Reset Value
			0 = No effect 1 = Clears the FAULT interrupt	

NOTE: On a Write of 1, clears the corresponding interrupt. A Write of 0 has no effect.

10.3.1.12 IMC_TCR

- Base Address: 0x400B_0000
- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
TOPCMPDAT	[15:0]	RW	Compare Data for TOP This field determines the TOP compare register value.	0x0000

NOTE: The update of IMC_TCR can be executed only when IMC is disabled. (IMC_CR0.0 = 0)

Caution: IMC_PACRR/FR, IMC_PBCRR/FR and IMC_PCCRR/FR should be less than or equal to IMC_TCR. (IMC_PxCRR/FR ← IMC_TCR)

10.3.1.13 IMC_DTCR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
DTCMPDAT	[15:0]	RW	Compare Data for Dead Time This field determines the dead time compare register value.	0x0000

NOTE: If you use ADC compare interrupt, you should set ADCCMPR/Fx from 1 to TOPCMP – 1. (0 < ADCCMPR/Fx < TOPCMP)

10.3.1.14 IMC_PACRR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0034, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																PACMPRDAT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PACMPRDAT	[15:0]	RW	Compare Data for Phase A Rising time This field determines the Phase A compare register value at rising.	0x0000

NOTE: If you use ADC compare interrupt, you should set IMC_PxCRR/F from 1 to IMC_TCR – 1.
(0 < IMC_PxCRR/FR < IMC_TCR)

10.3.1.15 IMC_PBCRR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0038, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PBCMPRDAT	[15:0]	RW	Compare Data for Phase B Rising time This field determines the Phase B compare register value at rising.	0x0000

NOTE: If you use ADC compare interrupt, you should set IMC_PxCRR/F from 1 to IMC_TCR – 1.
(0 < IMC_PxCRR/FR < IMC_TCR)

10.3.1.16 IMC_PCCRR

- Base Address: 0x400B_0000
- Address = Base Address + 0x003C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PCCMPRDAT	[15:0]	RW	Compare Data for Phase C Rising time This field determines the Phase C compare register value at rising.	0x0000

NOTE: If you use ADC compare interrupt, you should set IMC_PxCRR/F from 1 to IMC_TCR – 1.
(0 < IMC_PxCRR/FR < IMC_TCR)

10.3.1.17 IMC_PACFR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0040, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PACMPFDAT	[15:0]	RW	Compare Data for Phase A Falling time This field determines the Phase A compare register value at falling.	0x0000

NOTE: If you use ADC compare interrupt, you should set IMC_PxCRR/F from 1 to IMC_TCR – 1.
(0 < IMC_PxCRR/FR < IMC_TCR)

10.3.1.18 IMC_PBCFR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0044, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PBCMPFDAT	[15:0]	RW	Compare Data for Phase B Falling time This field determines the Phase B compare register value at falling.	0x0000

NOTE: If you use ADC compare interrupt, you should set IMC_PxCRR/F from 1 to IMC_TCR – 1.
(0 < IMC_PxCRR/FR < IMC_TCR)

10.3.1.19 IMC_PCCFR

- Base Address: 0x400B_0000
- Address = Base Address + 0x0048, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PCCMPFDAT	[15:0]	RW	Compare Data for Phase C Falling time This field determines the Phase C compare register value at falling.	0x0000

NOTE: If you use ADC compare interrupt, you should set IMC_PxCRR/F from 1 to IMC_TCR – 1.
(0 < IMC_PxCRR/FR < IMC_TCR)

10.3.1.20 IMC_ASTSR

- Base Address: 0x400B_0000
- Address = Base Address + 0x004C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																								ADCMPF2SEL	ADCMPR2SEL	ADCMPF1SEL	ADCMPR1SEL	ADCMPF0SEL	ADCMPR0SEL	0SEL	TOPCMPSEL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	R	Reserved	0
ADCMPF2SEL	[7]	RW	ADC Start Trigger Signal by ADCCMPF2 Match. This bit determines whether ADCMPF2 match of IMCNT is used for ADC trigger signal or not. 0 = Does not select 1 = Selects	0
ADCMPR2SEL	[6]	RW	ADC Start Trigger Signal by ADCCMPR2 Match. This bit determines whether ADCMPR2 match of IMCNT is used for ADC trigger signal or not. 0 = Does not select 1 = Selects	0
ADCMPF1SEL	[5]	RW	ADC Start Trigger Signal by ADCCMPF1 Match. This bit determines whether ADCMPF1 match of IMCNT is used for ADC trigger signal or not. 0 = Does not select 1 = Selects	0
ADCMPR1SEL	[4]	RW	ADC Start Trigger Signal by ADCCMPR1 Match. This bit determines whether ADCCMPR1 match of IMCNT is used for ADC trigger signal or not. 0 = Does not select 1 = Selects	0
ADCMPF0SEL	[3]	RW	ADC Start Trigger Signal by ADCCMPF0 Match. This bit determines whether ADCCMPF0 match of IMCNT is used for ADC trigger signal or not. 0 = Does not select 1 = Selects	0
ADCMPR0SEL	[2]	RW	ADC Start Trigger Signal by ADCCMPR0 Match. This bit determines whether ADCCMPR0 match of IMCNT	0

Name	Bit	Type	Description	Reset Value
			is used for ADC trigger signal or not. 0 = Does not select 1 = Selects	
0SEL	[1]	RW	ADC Start Trigger Signal by Counter Zero Match. This bit determines whether 0 match of IMCNT is used for ADC trigger signal or not. 0 = Does not select 1 = Selects	0
TOPCMPSEL	[0]	RW	ADC Start Trigger Signal by TOPCMP Match. This bit determines whether TOPCMP match of IMCNT is used for ADC trigger signal or not. 0 = Does not select 1 = Selects	0

NOTE:

1. The ADC conversion should not be overlapped by setting the appropriate value to each compare register.
2. The setting of IMC_ASTSR register bit does not affect interrupt generation.
3. When IMC is in an Saw-Tooth wave mode, the values of ADCCMPF0SEL, ADCCMPF1SEL, and ADC MPF2SEL bit do not have any effect in operation.

10.3.1.21 IMC_ASCRR0

- Base Address: 0x400B_0000
- Address = Base Address + 0x0050, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																ADCMPR0DAT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
ADCMPR0DAT	[15:0]	RW	ADC Compare Data 0 for Rising time This field determines the ADC compare register value at rising.	0x0000

10.3.1.22 IMC_ASCRR1

- Base Address: 0x400B_0000
- Address = Base Address + 0x0054, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																ADCMPR1DAT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
ADCMPR1DAT	[15:0]	RW	ADC Compare Data 1 for Rising time This field determines the ADC compare register value at rising.	0x0000

10.3.1.23 IMC_ASCRR2

- Base Address: 0x400B_0000
- Address = Base Address + 0x0058, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																ADCMPR2DAT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
ADCMPR2DAT	[15:0]	RW	ADC Compare Data 2 for Rising time This field determines the ADC compare register value at rising.	0x0000

10.3.1.24 IMC_ASCFR0

- Base Address: 0x400B_0000
- Address = Base Address + 0x005C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																ADCMPF0DAT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
ADCMPF0DAT	[15:0]	RW	ADC Compare Data 0 for Falling time This field determines the ADC compare register value at falling.	0x0000

10.3.1.25 IMC_ASCFR1

- Base Address: 0x400B_0000
- Address = Base Address + 0x0060, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																ADCMPF1DAT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
ADCMPF1DAT	[15:0]	RW	ADC Compare Data 1 for Falling time This field determines the ADC compare register value at falling.	0x0000

10.3.1.26 IMC_ASCFR2

- Base Address: 0x400B_0000
- Address = Base Address + 0x0064, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																ADCMPF2DAT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
ADCMPF2DAT	[15:0]	RW	ADC Compare Data 2 for Falling time This field determines the ADC compare register value at falling.	0x0000

11 Interrupt Controller (INTC)

11.1 Overview

Interrupt Controller includes NVIC of Cortex-M0 processors. These processors enable the low latency interrupt processing and provide efficient service for late arriving interrupts. There are 38 individual interrupt vector sources including six Cortex-M0 interrupt vector sources.

11.1.1 Features

The features of the interrupt controller are:

- Cortex-M0's NVIC interface
- Six core interrupt vector sources
  - Reset
  - NMI
  - HardFault
  - SVCall
  - PendSW
  - SysTick
- 32 device interrupt vector sources
- Dynamically reconfigurable interrupt priority, exist at four levels
- Low latency exception and interrupt handling
- Implement Cortex-M0 system control registers

NVIC and the processor core interface are closely coupled, which enable low latency interrupt processing and efficient processing of late arriving interrupts. NVIC manages all interrupts including core exceptions. NVIC manages all interrupts including core exceptions. Refer to Vectored Interrupt Controller of ARM Cortex-M0 TM Technical Reference Manual

11.2 Functional Description

Functional description section includes:

- Interrupt vector
- Block diagram

11.2.1 Interrupt Vector

The interrupt vector section includes:

- Core interrupt vector
- Device interrupt vector

11.2.1.1 Core Interrupt Vector

S3FN429 has six core interrupt vectors to be supported by Cortex M0 as below. *Table 11-1* describes the core interrupt vector.

Table 11-1 Core Interrupt Vector

Number	Address	Vector	Description
0	0x0000_0000	Reserved	Starting value of MSP
1	0x0000_0004	Reset	–
2	0x0000_0008	NMI	Non-maskable interrupt
3	0x0000_000C	HardFault	All class of fault
4 to 10	0x0000_0010 to 0x0000_0028	Reserved	–
11	0x0000_002C	SVCall	System service call through SWI instruction
12	0x0000_0030	Reserved	–
13	0x0000_0034	Reserved	–
14	0x0000_0038	PendSV	Pending request for system service
15	0x0000_003C	SysTick	System tick timer interrupt

11.2.1.2 Device Interrupt Vector

Table 11-2 describes the device interrupt vector for S3FN429.

Table 11-2 Device Interrupt Vector

Num.	Address	Vector Name	Block	Interrupt Sources
IRQ0	0x0000_0040	WDT	WDT	Refer to watch-dog timer interrupt register
IRQ1	0x0000_0044	FAULT	IMC	Fault interrupt
IRQ2	0x0000_0048	EDGEDET0	COMP	Comparator0 interrupt
IRQ3	0x0000_004C	EDGEDET1	COMP	Comparator1 interrupt
IRQ4	0x0000_0050	EDGEDET2	COMP	Comparator2 interrupt
IRQ5	0x0000_0054	EDGEDET3	COMP	Comparator3 interrupt
IRQ6	0x0000_0058	ZERO or TOP	IMC	Counter zero match, top compare match interrupt
IRQ7	0x0000_005C	ADCRM0/ADCFM0	IMC	ADC compare match in rising/falling time interrupt 0
IRQ8	0x0000_0060	ADCRM1/ADCFM1	IMC	ADC compare match in rising/falling time interrupt 1
IRQ9	0x0000_0064	ADCRM2/ADCFM2	IMC	ADC compare match in rising/falling time interrupt 2
IRQ10	0x0000_0068	ADC	ADC	Refer to ADC interrupt register
IRQ11	0x0000_006C	PPD_P	PPD	Position counter/capture and phasez interrupt
IRQ12	0x0000_0070	PPD_S	PPD	Speed counter/capture interrupt
IRQ13	0x0000_0074	USART_RXRDY	USART	RXRDY interrupt
IRQ14	0x0000_0078	CM	CM	Refer to CM interrupt register
IRQ15	0x0000_007C	IFC	IFC	Refer to IFC interrupt register
IRQ16	0x0000_0080	EEIA	CM	Refer to external event source interrupt 0
IRQ17	0x0000_0084	EEIB	CM	Refer to external event source interrupt 1
IRQ18	0x0000_0088	EEIC	CM	Refer to external event source interrupt 2
IRQ19	0x0000_008C	EEID	CM	Refer to external event source interrupt 3
IRQ20	0x0000_0090	EEIE	CM	Refer to external event source interrupt 4 and 5
IRQ21	0x0000_0094	EEIF	CM	Refer to external event source interrupt 6 and 7
IRQ22	0x0000_0098	TC0	TC	Refer to timer/counter0 interrupt register
IRQ23	0x0000_009C	TC1	TC	Refer to timer/counter1 interrupt register
IRQ24	0x0000_00A0	TC2	TC	Refer to timer/counter2 interrupt register
IRQ25	0x0000_00A4	PWM0	PWM	Refer to PWM0 interrupt register
IRQ26	0x0000_00A8	PWM1	PWM	Refer to PWM1 interrupt register
IRQ27	0x0000_00AC	PWM2	PWM	Refer to PWM2 interrupt register
IRQ28	0x0000_00B0	PWM3	PWM	Refer to PWM3 interrupt register
IRQ29	0x0000_00B4	SPI0	SPI	Refer to SPI interrupt register
IRQ30	0x0000_00B8	USART0	USART	Refer to USART0 interrupt register
IRQ31	0x0000_00BC	GPIO0	GPIO	Refer to GPIO interrupt register

11.2.2 Block Diagram

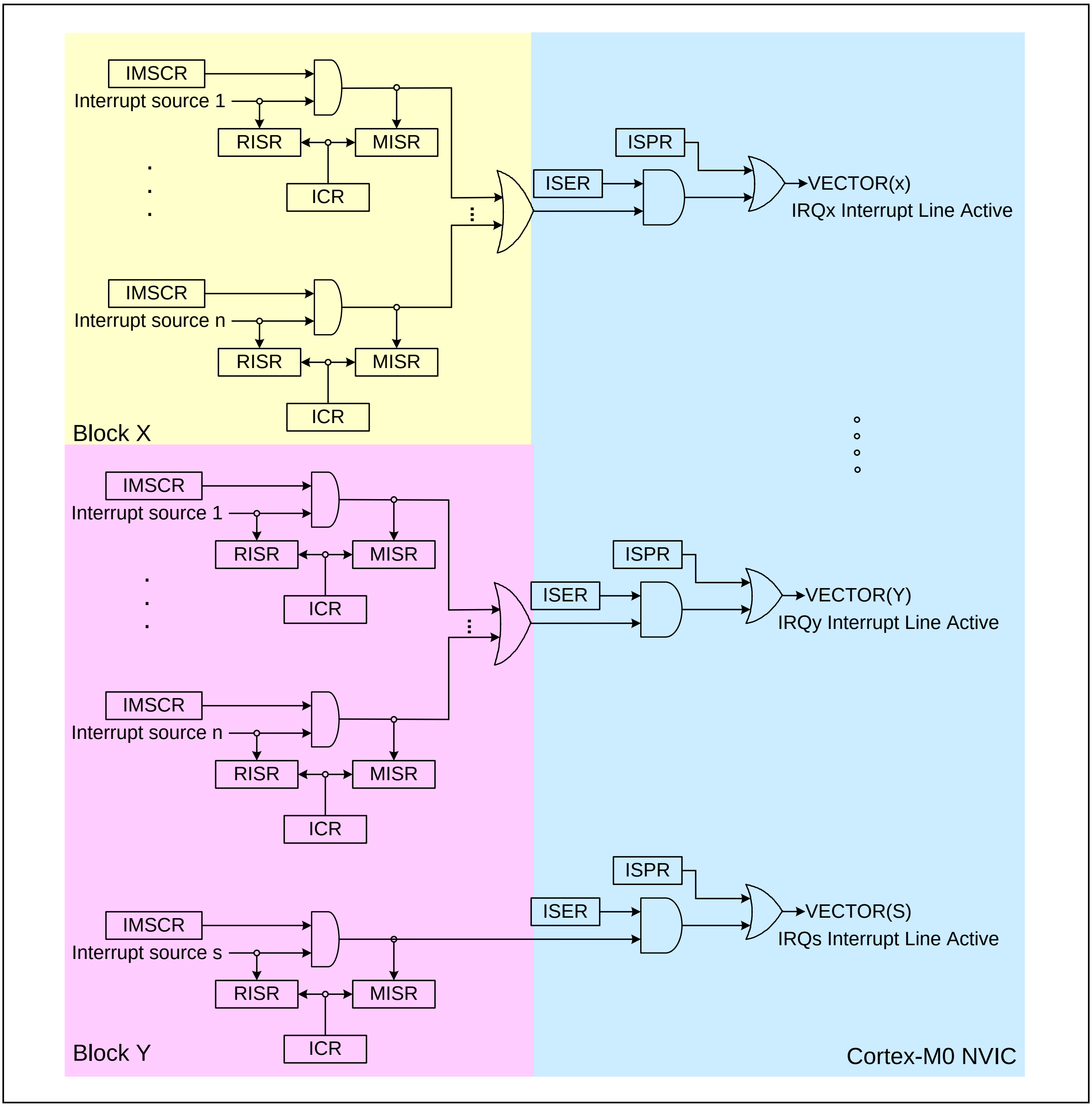


Figure 11-1 Interrupt Block Diagram

Figure 11-1 is the block diagram to show the connection and relationship between device interrupt sources and interrupt vectors. Each vector has one or several interrupt event sources. Each interrupt event source has control and status bits as follows; interrupt mask set/clear, raw interrupt status, masked interrupt status, and interrupt clear bit. That control should be done at each block. The specific interrupt source in some blocks has one's own interrupt vector. For example, USART has 15 interrupt sources including RXRDY. Among them, RXRDY interrupt source is mapped to IRQ13. Other interrupt sources are mapped to IRQ30. For more details about interrupt sources, refer to the corresponding block of manual.

NOTE: Refer to each corresponding block chapter of manual related to registers mentioned Figure 11-1
IMSCR: Interrupt Mask Set/Clear Register (Read/Write)
RISR: Raw Interrupt Status Register (Read only)
MISR: Masked Interrupt Status Register (Read only)
ICR: Interrupt Clear Register (Write only)
ISER: Interrupt Set Enable Register (Read/Write)
ISPR: Interrupt Set Pending Register (Read/Write)

Block X or Block Y is as follows: WDT, IMC, COMP, ADC, PPD, USART, CM, IFC, TC, PWM, SPI, GPIO

11.3 Register Description

11.3.1 Register Map Summary

The tables in this section describe NVIC specific registers in System Control Space (SCS) of Cortex-M0.

- Base Address: 0xE000_0000

Register	Offset	Description	Reset Value
NVIC_ISER	0xE100	IRQ 0 to 31 interrupt set-enable register	0x0000_0000
NVIC_ICER	0xE180	IRQ 0 to 31 interrupt clear-enable register	0x0000_0000
NVIC_ISPR	0xE200	IRQ 0 to 31 interrupt set-pending register	0x0000_0000
NVIC_ICPR	0xE280	IRQ 0 to 31 interrupt clear-pending register	0x0000_0000
NVIC_IPR0	0xE400	IRQ 0 to 3 interrupt priority register	0x0000_0000
NVIC_IPR1	0xE404	IRQ 4 to 7 interrupt priority register	0x0000_0000
NVIC_IPR2	0xE408	IRQ 8 to 11 interrupt priority register	0x0000_0000
NVIC_IPR3	0xE40C	IRQ 12 to 15 interrupt priority register	0x0000_0000
NVIC_IPR4	0xE410	IRQ 16 to 19 interrupt priority register	0x0000_0000
NVIC_IPR5	0xE414	IRQ 20 to 23 interrupt priority register	0x0000_0000
NVIC_IPR6	0xE418	IRQ 24 to 27 interrupt priority register	0x0000_0000
NVIC_IPR7	0xE41C	IRQ 28 to 31 interrupt priority register	0x0000_0000

The system control region includes the status and configuration registers that apply to the NVIC as a part of the general exception model. All other external interrupt specific registers are within the NVIC region of the SCS.

11.3.1.1 NVIC_ISER

- Base Address: 0xE000_0000
- Address = Base Address + 0xE100, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
SETENA	[31:0]	RW	Enables for device interrupt vector #0-31 Each bit represents an interrupt vector from IRQ0 to IRQ3 Writing "1" enables the associated interrupt. Writing "0" has no effect. The register reads back with the current enable state. Bit[0] for IRQ0 Bit[2] for IRQ1 … Bit[x] for IRQx	0x0000_0000

11.3.1.2 NVIC_ICER

- Base Address: 0xE000_0000
- Address = Base Address + 0xE180, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
CLRENA	[31:0]	RW	Disables for device interrupt vector #0-31 Each bit represents an interrupt vector from IRQ0 to IRQ3 Writing "1" disables the associated interrupt Writing "0" has no effect. The register reads back with the current enable state. Bit[0] for IRQ0 Bit[2] for IRQ1 … Bit[x] for IRQx	0x0000_0000

11.3.1.3 NVIC_ISPR

- Base Address: 0xE000_0000
- Address = Base Address + 0xE200, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
SETPEND	[31:0]	RW	Writing "1" to a bit insert the associated interrupt under the software control to a pending state. Each bit represents an interrupt pin number from IRQ0 to IRQ31. Writing "0" to a bit has no effect on the associated interrupt. The register reads back from the pending state. Bit[0] for IRQ0 Bit[2] for IRQ1 … Bit[x] for IRQx	0x0000_0000

11.3.1.4 NVIC_ICPR

- Base Address: 0xE000_0000
- Address = Base Address + 0xE280, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
CLRREND	[31:0]	RW	Writing "1" to a bit removes the associated interrupt from its pending state under the software control. Each bit represents an interrupt pin number from IRQ0 to IRQ31. Writing "0" to a bit has no effect on the associated interrupt. The register reads back from the pending state. Bit[0] for IRQ0 Bit[2] for IRQ1 … Bit[x] for IRQx	0x0000_0000

11.3.1.5 NVIC_IPR0

- Base Address: 0xE000_0000
- Address = Base Address + 0xE400, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_N3		RSVD						PRI_N2		RSVD						PRI_N1		RSVD						PRI_N0		RSVD					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PRI_N3	[31:30]	RW	The priority of the interrupt vector number 3	00
RSVD	[29:24]	RW	Reserved	0
PRI_N2	[23:22]	RW	The priority of the interrupt vector number 2	00
RSVD	[21:16]	RW	Reserved	0
PRI_N1	[15:14]	RW	The priority of the interrupt vector number 1	00
RSVD	[13:8]	RW	Reserved	0
PRI_N0	[7:6]	RW	The priority of the interrupt vector number 0	00
RSVD	[5:0]	RW	Reserved	0

11.3.1.6 NVIC_IPR1

- Base Address: 0xE000_0000
- Address = Base Address + 0xE404, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_N7		RSVD						PRI_N6		RSVD						PRI_N5		RSVD						PRI_N4		RSVD					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
PRI_N7	[31:30]	RW	The priority of the interrupt vector number 7	00
RSVD	[29:24]	RW	Reserved	0
PRI_N6	[23:22]	RW	The priority of the interrupt vector number 6	00
RSVD	[21:16]	RW	Reserved	0
PRI_N5	[15:14]	RW	The priority of the interrupt vector number 5	00
RSVD	[13:8]	RW	Reserved	0
PRI_N4	[7:6]	RW	The priority of the interrupt vector number 4	00
RSVD	[5:0]	RW	Reserved	0

11.3.1.7 NVIC_IPR2

- Base Address: 0xE000_0000
- Address = Base Address + 0xE408, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_N11		RSVD						PRI_N10		RSVD						PRI_N9		RSVD						PRI_N8		RSVD					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PRI_N11	[31:30]	RW	The priority of the interrupt vector number 11	00
RSVD	[29:24]	RW	Reserved	0
PRI_N10	[23:22]	RW	The priority of the interrupt vector number 10	00
RSVD	[21:16]	RW	Reserved	0
PRI_N9	[15:14]	RW	The priority of the interrupt vector number 9	00
RSVD	[13:8]	RW	Reserved	0
PRI_N8	[7:6]	RW	The priority of the interrupt vector number 8	00
RSVD	[5:0]	RW	Reserved	0

11.3.1.8 NVIC_IPR3

- Base Address: 0xE000_0000
- Address = Base Address + 0xE40C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_N15		RSVD						PRI_N14		RSVD						PRI_N13		RSVD						PRI_N12		RSVD					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PRI_N15	[31:30]	RW	The priority of the interrupt vector number 15	00
RSVD	[29:24]	RW	Reserved	0
PRI_N14	[23:22]	RW	The priority of the interrupt vector number 14	00
RSVD	[21:16]	RW	Reserved	0
PRI_N13	[15:14]	RW	The priority of the interrupt vector number 13	00
RSVD	[13:8]	RW	Reserved	0
PRI_N12	[7:6]	RW	The priority of the interrupt vector number 12	00
RSVD	[5:0]	RW	Reserved	0

11.3.1.9 NVIC_IPR4

- Base Address: 0xE000_0000
- Address = Base Address + 0xE410, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_N19		RSVD						PRI_N18		RSVD						PRI_N17		RSVD						PRI_N16		RSVD					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
PRI_N19	[31:30]	RW	The priority of the interrupt vector number 19	00
RSVD	[29:24]	RW	Reserved	0
PRI_N18	[23:22]	RW	The priority of the interrupt vector number 18	00
RSVD	[21:16]	RW	Reserved	0
PRI_N17	[15:14]	RW	The priority of the interrupt vector number 17	00
RSVD	[13:8]	RW	Reserved	0
PRI_N16	[7:6]	RW	The priority of the interrupt vector number 16	00
RSVD	[5:0]	RW	Reserved	0

11.3.1.10 NVIC_IPR5

- Base Address: 0xE000_0000
- Address = Base Address + 0xE414, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_N23		RSVD						PRI_N22		RSVD						PRI_N21		RSVD						PRI_N20		RSVD					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PRI_N23	[31:30]	RW	The priority of the interrupt vector number 23	00
RSVD	[29:24]	RW	Reserved	0
PRI_N22	[23:22]	RW	The priority of the interrupt vector number 22	00
RSVD	[21:16]	RW	Reserved	0
PRI_N21	[15:14]	RW	The priority of the interrupt vector number 21	00
RSVD	[13:8]	RW	Reserved	0
PRI_N20	[7:6]	RW	The priority of the interrupt vector number 20	00
RSVD	[5:0]	RW	Reserved	0

11.3.1.11 NVIC_IPR6

- Base Address: 0xE000_0000
- Address = Base Address + 0xE418, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_N27		RSVD						PRI_N26		RSVD						PRI_N25		RSVD						PRI_N24		RSVD					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PRI_N27	[31:30]	RW	The priority of the interrupt vector number 27	00
RSVD	[29:24]	RW	Reserved	0
PRI_N26	[23:22]	RW	The priority of the interrupt vector number 26	00
RSVD	[21:16]	RW	Reserved	0
PRI_N25	[15:14]	RW	The priority of the interrupt vector number 25	00
RSVD	[13:8]	RW	Reserved	0
PRI_N24	[7:6]	RW	The priority of the interrupt vector number 24	00
RSVD	[5:0]	RW	Reserved	0

11.3.1.12 NVIC_IPR7

- Base Address: 0xE000_0000
- Address = Base Address + 0xE41C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_N31		RSVD						PRI_N30		RSVD						PRI_N29		RSVD						PRI_N28		RSVD					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
PRI_N31	[31:30]	RW	The priority of the interrupt vector number 31	00
RSVD	[29:24]	RW	Reserved	0
PRI_N30	[23:22]	RW	The priority of the interrupt vector number 30	00
RSVD	[21:16]	RW	Reserved	0
PRI_N29	[15:14]	RW	The priority of the interrupt vector number 29	00
RSVD	[13:8]	RW	Reserved	0
PRI_N28	[7:6]	RW	The priority of the interrupt vector number 28	00
RSVD	[5:0]	RW	Reserved	0

12 I/O Configuration

12.1 Overview

The I/O Configuration (IOCONF) chapter describes the configuration of specific function pins mapped to each I/O pin.

12.1.1 Features

The features of IOCONF are:

- Configuration of function pin:
  - Select one among four functions
    - Function 0 (GPIO)
    - Function 1
    - Function 2
    - Function 3
- Configuration of Pull-Up Resistor (PUCR)
- Configuration of Open-Drain Control Resistor (ODCR)

12.2 Functional Description

The Functional Description section describes the general description and peripheral configuration of IOCONF.

12.2.1 General Description

The peripherals have their own dedicated pins multiplexed with General Purpose IO (GPIO) pin and other peripheral pins. User should configure the corresponding pin for the function of the target peripheral. Define the pin as one function pin among maximum four functions. GPIO will be the first function of pin.

IOCONF block controls Enable/Disable for each pins Pull-Up Resistor and Open-Drain Control Resistor.

12.2.2 Peripheral Configuration

Table 12-1 describes the I/O function mode configuration.

Table 12-1 I/O Function Mode Configuration

I/O Group 0	Function Number	F0	F1	F2	F3
Pin Number	**IO0x.y_FSEL[1:0]**	**00'b**	**01'b**	**10'b**	**11'b**
9	IO0.0_FSEL[1:0]	P0.0	EXI0	TPWM0	OP0_N
10	IO0.1_FSEL[1:0]	P0.1	EXI1	TCAP0	OP0_P
11	IO0.2_FSEL[1:0]	P0.2	AIN1	TCLK0	OP0_O
12	IO0.3_FSEL[1:0]	P0.3	AIN2	PWM0	EXI2
13	IO0.4_FSEL[1:0]	P0.4	AIN3	PWM1	EXI3
14	IO0.5_FSEL[1:0]	P0.5	AIN4	PWM2	EXI4
15	IO0.6_FSEL[1:0]	P0.6	AIN5	PWM3	EXI5
16	IO0.7_FSEL[1:0]	P0.7	AIN6	PWM0	TPWM1
17	IO0.8_FSEL[1:0]	P0.8	AIN7	PWM1	TCAP1
18	IO0.9_FSEL[1:0]	P0.9	AIN8	PWM2	TCLK1
19	IO0.10_FSEL[1:0]	P0.10	AIN9	PWM3	EXI6
20	IO0.11_FSEL[1:0]	P0.11	AIN10	USARTCLK0	ADTRG
24	IO0.12_FSEL[1:0]	P0.12	PWMU2	TPWM2	EXI7
25	IO0.13_FSEL[1:0]	P0.13	PWMU1	TCAP2	EXI8
26	IO0.14_FSEL[1:0]	P0.14	PWMU0	TCLK2	EXI9
27	IO0.15_FSEL[1:0]	P0.15	PWMD2	COP0	EXI10
28	IO0.16_FSEL[1:0]	P0.16	PWMD1	COP1	EXI11
29	IO0.17_FSEL[1:0]	P0.17	PWMD0	COP2	EXI12
32	IO0.18_FSEL[1:0]	P0.18	PWMOFF	COP3	EXI13
33	IO0.19_FSEL[1:0]	P0.19	PHASEA	USARTRX0	EXI14
34	IO0.20_FSEL[1:0]	P0.20	PHASEB	USARTTX0	EXI15
35	IO0.21_FSEL[1:0]	P0.21	PHASEZ	USARTCLK0	EXI16
36	IO0.22_FSEL[1:0]	P0.22	COMP0_N	MISO0	EXI17
37	IO0.23_FSEL[1:0]	P0.23	COMP0_P	MOSI0	PWM1

I/O Group 0	Function Number	F0	F1	F2	F3
Pin Number	**IO0x.y_FSEL[1:0]**	**00'b**	**01'b**	**10'b**	**11'b**
38	IO0.24_FSEL[1:0]	P0.24	COMP1_N	SCLK0	PWM2
39	IO0.25_FSEL[1:0]	P0.25	COMP1_P	FSS0	COP4
40	IO0.26_FSEL[1:0]	P0.26	COMP2_N	USARTRX0	EXI18
41	IO0.27_FSEL[1:0]	P0.27	COMP2_P	USARTTX0	EXI19
42	IO0.28_FSEL[1:0]	P0.28	COMP3_N	EXI20	SWDIO
43	IO0.29_FSEL[1:0]	P0.29	COMP3_P	EXI21	SWDCLK
7	IO0.30_FSEL[1:0]	XOUT/P0.30	XOUT/–	XOUT/USARTTX0	XOUT/EXI22
8	IO0.31_FSEL[1:0]	XIN/P0.31	XIN/EXI23	XIN/USARTRX0	XIN/PWM0

12.3 Register Description

12.3.1 Register Map Summary

- Base Address: 0x4005_8000

Register	Offset	Description	Reset Value
IOCONF_MLR0	0x0000	Mode low register	0x0000_0000
IOCONF_MHR0	0x0004	Mode high register	0x0F00_0000
IOCONF_PUCR0	0x0008	Pull-up control register	0x3000_0000
IOCONF_ODCR0	0x000C	Open-drain control register	0x0000_0000

12.3.1.1 IOCONF_MLR0

- Base Address: 0x4005_8000
- Address = Base Address + 0x0000, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IO0_15_FSEL		IO0_14_FSEL		IO0_13_FSE		IO0_12_FSEL		IO0_11_FSEL		IO0_10_FSEL		IO0_9_FSEL		IO0_8_FSEL		IO0_7_FSEL		IO0_6_FSEL		IO0_5_FSEL		IO0_4_FSEL		IO0_3_FSEL		IO0_2_FSE		IO0_1_FSEL		IO0_0_FSEL	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
IO0_15_FSEL	[31:30]	RW	IO0.y Function Selection. 00'b = Function 0 (GPIO) 01'b = Function 1 10'b = Function 2 11'b = Function 3	00'b
IO0_14_FSEL	[29:28]			
IO0_13_FSEL	[27:26]			
IO0_12_FSEL	[25:24]			
IO0_11_FSEL	[23:22]			
IO0_10_FSEL	[21:20]			
IO0_9_FSEL	[19:18]			
IO0_8_FSEL	[17:16]			
IO0_7_FSEL	[15:14]			
IO0_6_FSEL	[13:12]			
IO0_5_FSEL	[11:10]			
IO0_4_FSEL	[9:8]			
IO0_3_FSEL	[7:6]			
IO0_2_FSEL	[5:4]			
IO0_1_FSEL	[3:2]			
IO0_0_FSEL	[1:0]			

NOTE: If you write undefined function value to IOx_xFSEL[1:0] [Fx(Function x)], then IOx_x_FSEL[1:0] will not change and will remain as the valid pre-configuration. *Table 12-1* (Function Mode Configuration) describes the function values for each I/O.

NOTE: If OP-AMP is enabled, P0.2 should be set to OP-AMP output pin.

12.3.1.2 IOCONF_MHR0

- Base Address: 0x4005_8000
- Address = Base Address + 0x0004, Reset Value = 0x0000_F000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IO0_31_FSEL		IO0_30_FSEL		IO0_29_FSEL		IO0_28_FSEL		IO0_27_FSEL		IO0_26_FSEL		IO0_25_FSEL		IO0_24_FSEL		IO0_23_FSEL		IO0_22_FSEL		IO0_21_FSEL		IO0_20_FSEL		IO0_19_FSEL		IO0_18_FSEL		IO0_17_FSEL		IO0_16_FSEL	
0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
IO0_31_FSEL	[31:30]	RW	IO0.y Function Selection. 00'b = Function 0 (GPIO) 01'b = Function 1 10'b = Function 2 11'b = Function 3	00'b
IO0_30_FSEL	[29:28]			00'b
IO0_29_FSEL	[27:26]			11'b
IO0_28_FSEL	[25:24]			11'b
IO0_27_FSEL	[23:22]			00'b
IO0_26_FSEL	[21:20]			00'b
IO0_25_FSEL	[19:18]			00'b
IO0_24_FSEL	[17:16]			00'b
IO0_23_FSEL	[15:14]			00'b
IO0_22_FSEL	[13:12]			00'b
IO0_21_FSEL	[11:10]			00'b
IO0_20_FSEL	[9:8]			00'b
IO0_19_FSEL	[7:6]			00'b
IO0_18_FSEL	[5:4]			00'b
IO0_17_FSEL	[3:2]			00'b
IO0_16_FSEL	[1:0]			00'b

NOTE:

1. If you write undefined function value to IOx_xFSEL[1:0] [Fx(Function x)], then IOx_x_FSEL[1:0] will not change and will remain as the valid pre-configuration. *Table 12-1* (I/O Function Mode Configuration) describes the function values for each I/O *Table 12-1* (I/O Function Mode Configuration) describes the function values for each I/O.
2. The reset value (11'b) of IO0_28 and IO0_29 are for Serial Wire Debug (SWD).

12.3.1.3 IO_PUCR0

- Base Address: 0x4005_8000
- Address = Base Address + 0x0008, Reset Value = 0x3000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IO0_31_PUEN	IO0_30_PUEN	IO0_29_PUEN	IO0_28_PUEN	IO0_27_PUEN	IO0_26_PUEN	IO0_25_PUEN	IO0_24_PUEN	IO0_23_PUEN	IO0_22_PUEN	IO0_21_PUEN	IO0_20_PUEN	IO0_19_PUEN	IO0_18_PUEN	IO0_17_PUEN	IO0_16_PUEN	IO0_15_PUEN	IO0_14_PUEN	IO0_13_PUEN	IO0_12_PUEN	IO0_11_PUEN	IO0_10_PUEN	IO0_9_PUEN	IO0_8_PUEN	IO0_7_PUEN	IO0_6_PUEN	IO0_5_PUEN	IO0_4_PUEN	IO0_3_PUEN	IO0_2_PUEN	IO0_1_PUEN	IO0_0_PUEN
0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
IO0_y_PUEN	[y]	RW	IO0_y Pull-Up Enable/Disable 0 = Disables Pull-Up resistor on pin IO0_y 1 = Enables Pull-Up resistor on pin IO0_y	0

NOTE: SWD pins (Port 28 and 29 of PIO0) multiplexed with GPIO, have the internal Pull-Up resistor enabled by default. When these ports are not used for SWD, we strongly recommend to disable the internal Pull-Up resistors by the software after reset.

12.3.1.4 IO_ODCR0

- Base Address: 0x4005_8000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IO0_31_ODEN	IO0_30_ODEN	IO0_29_ODEN	IO0_28_ODEN	IO0_27_ODEN	IO0_26_ODEN	IO0_25_ODEN	IO0_24_ODEN	IO0_23_ODEN	IO0_22_ODEN	IO0_21_ODEN	IO0_20_ODEN	IO0_19_ODEN	IO0_18_ODEN	IO0_17_ODEN	IO0_16_ODEN	IO0_15_ODEN	IO0_14_ODEN	IO0_13_ODEN	IO0_12_ODEN	IO0_11_ODEN	IO0_10_ODEN	IO0_9_ODEN	IO0_8_ODEN	IO0_7_ODEN	IO0_6_ODEN	IO0_5_ODEN	IO0_4_ODEN	IO0_3_ODEN	IO0_2_ODEN	IO0_1_ODEN	IO0_0_ODEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
IO0_y_ODEN	[y]	RW	IO0_y Open-Drain Enable/Disable 0 = Disables Open-Drain output mode on pin IO0_y (Push-Pull Output mode) 1 = Enables Open-Drain Output mode on pin IO0_y	0

13 Operational Amplifier

13.1 Overview

The Operation Amplifier (OP-AMP) chapter describes OP-AMP that operates separately or with an Analog to Digital Converter (ADC). You can configure OP-AMP gain using control bits.

13.1.1 Features

The features of OP-AMP are:

- Amplification of input signal
- Configurable internal gain from x2.5 to x15 by register setting
- Gain control by external circuit.

13.1.2 Pin Description

Table 13-1 describes the pin description of OP-AMP.

Table 13-1 OP-AMP Pin Description

Pin Name	Function	I/O Type	Comments
OPA_Ox	Operational Amplifier Output	O	–
OPA_Px	Operational Amplifier Positive Input	I	–
OPA_Nx	Operational Amplifier Negative Input	I	–

NOTE: "x" represents the channel of an OP-AMP. For example, OP-AMP0 has channels represented as OPA_O0, OPA_P0, and OPA_N0.

13.1.3 Block Diagram

Figure 13-1 illustrates the block diagram for an OP-AMP.

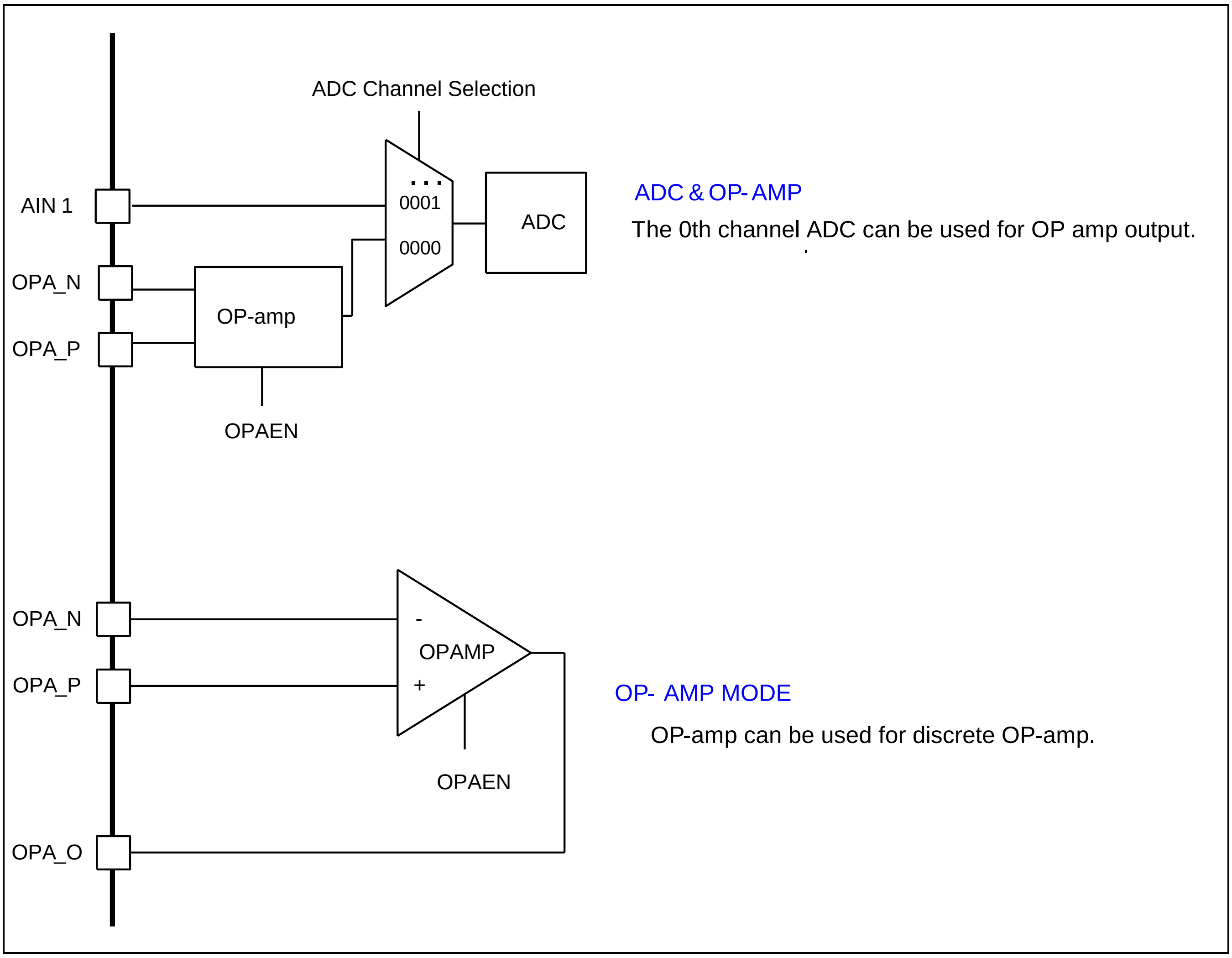


Figure 13-1 OP-AMP Block Diagram

If the ADC has to convert the signal amplified by the OP-AMP with internal gain, then you have to enable the OP-AMP and program it with a gain value before the ADC operation. The ADC should select AIN0 as a conversion input channel.

13.1.4 Gain Generation Circuit

You can select an external gain control circuit with appropriate pins as OPA_N, OPA_P, and OPA_O or an internal gain control. In this case, you should configure OPAG bit that corresponds to each mode.

Table 13-2 describes the gain specification of OP-AMP.

Table 13-2 Gain Configuration Table

OPAGVx				Gain Specification of OP-AMP
0	0	0	0	× 2.500
0	0	0	1	× 2.667
0	0	1	0	× 2.833
0	0	1	1	× 3.000
0	1	0	0	× 3.333
0	1	0	1	× 3.667
0	1	1	0	× 4.000
0	1	1	1	× 4.500
1	0	0	0	× 5.000
1	0	0	1	× 5.667
1	0	1	0	× 6.667
1	0	1	1	× 8.000
1	1	0	0	× 10.00
1	1	0	1	× 12.00
1	1	1	0	× 15.00

13.2 Register Description

13.2.1 Register Map Summary

- Base Address: 0x4004_1000

Register	Offset	Description	Reset Value
OPA_IDR	0x000	OP-AMP ID register	0x0001_001D
OPA_CEDR	0x004	OP-AMP clock enable/disable register	0x0000_0000
OPA_SRR	0x008	OP-AMP software reset register	0x0000_0000
OPA_CR	0x00C	OP-AMP control register	0x0000_0000
OPA_GCR	0x010	OP-AMP gain control register	0x0000_0000

13.2.1.1 OPA_IDR

- Base Address: 0x4004_1000
- Address = Base Address + 0x0000, Reset Value = 0x0001_001D

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores the ID code for the corresponding IP.	0x0001_001D

13.2.1.2 OPA_CEDR

- Base Address: 0x4004_1000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															CLKEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable/ Disable Control Bit 0 = Disables OP-AMP Clock bit 1 = Enables OP-AMP Clock bit OP-AMP software reset does not affect CLKEN bit status.	0

13.2.1.3 OPA_SRR

- Base Address: 0x4004_1000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Performs OP-AMP software reset Software Reset operation	–

13.2.1.4 OPA_CR

- Base Address: 0x4004_1000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	R	Reserved	0
OPAM	[8]	RW	This bit should be set to "0"	0
RSVD	[7:1]	R	Reserved	0
OPA	[0]	RW	OP-AMP Enable Bit 0 = Disables OP-AMP 1 = Enables OP-AMP	0

NOTE: If OP-AMP is enabled, P0.2 should be set to OP-AMP output pin.

13.2.1.5 OPA_GCR

- Base Address: 0x4004_1000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

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

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="5">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:8]</td><td>R</td><td colspan="5">Reserved</td><td>0</td></tr>
<tr><td>GCT</td><td>[7]</td><td>RW</td><td colspan="5">OP-AMP Gain Control Type Selection Bit
0 = Selects external gain control
1 = Selects internal gain control</td><td>0</td></tr>
<tr><td>RSVD</td><td>[6:1]</td><td>R</td><td colspan="5">Reserved</td><td>0</td></tr>
<tr><td rowspan="18">GV</td><td rowspan="18">[3:0]</td><td rowspan="18">RW</td><td colspan="5">Channel x OP-AMP Gain Value Field
This has an effect on an internal gain control.</td><td rowspan="18">0000'b</td></tr>
<tr><td colspan="4"><b>OPAGVx</b></td><td><b>Gain Specification of OP-AMP</b></td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>× 2.500</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>× 2.667</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>× 2.833</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>× 3.000</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>× 3.333</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>× 3.667</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>× 4.000</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>× 4.500</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>× 5.000</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>× 5.667</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>× 6.667</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>× 8.000</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>× 10.00</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>× 12.00</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>× 15.00</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>Not used</td></tr>
</table>

14 Pulse Width Modulation

14.1 Overview

The Pulse Width Modulation (PWM) chapter describes the on chip PWM generator. The PWM is a common technique to control power devices in industrial fields. PWM output channel uses 16-bit up counter to generate rectangular pluses with the programmable period and duty ratio. It has output extensions up to 22-bit resolution.

14.1.1 Features

The features of PWM are:

- The control bit allows the new configuration of:
  - Programmable duty cycle and frequency
  - Programmable active level
  - Duty ratio from 0 to 1 with 16-bit resolution

- The PWM output signal control is:
  - Programmable idle level

- The Enable/Disable interrupt sources are:
  - Pulse start and stop
  - Period start and end
  - Pulse match

- The extension cycle circuit implements the PWM extension function.
- Up to 22-bit resolution including extension function

14.1.2 Pin Description

Table 14-1 describes the function of PWM pin.

Table 14-1 PWM Pin Description

Pin Name	Function	I/O Type	Comments
PWMx	Pulse width modulation output	O	–

NOTE: "x" stands for channel number of PWM. For example, PWM0 or PWM1.

14.2 Functional Description

The functional description section includes:

- Block Diagram
- General Description
- Clock and Operation Frequency
- Period
- PULSE Level
- nPULSE Width
- IDLE Level
- Parameter Relationship
- Extension Bit

14.2.1 Block Diagram

Figure 14-1 illustrates the Pulse Width Modulation (PWM) block diagram.

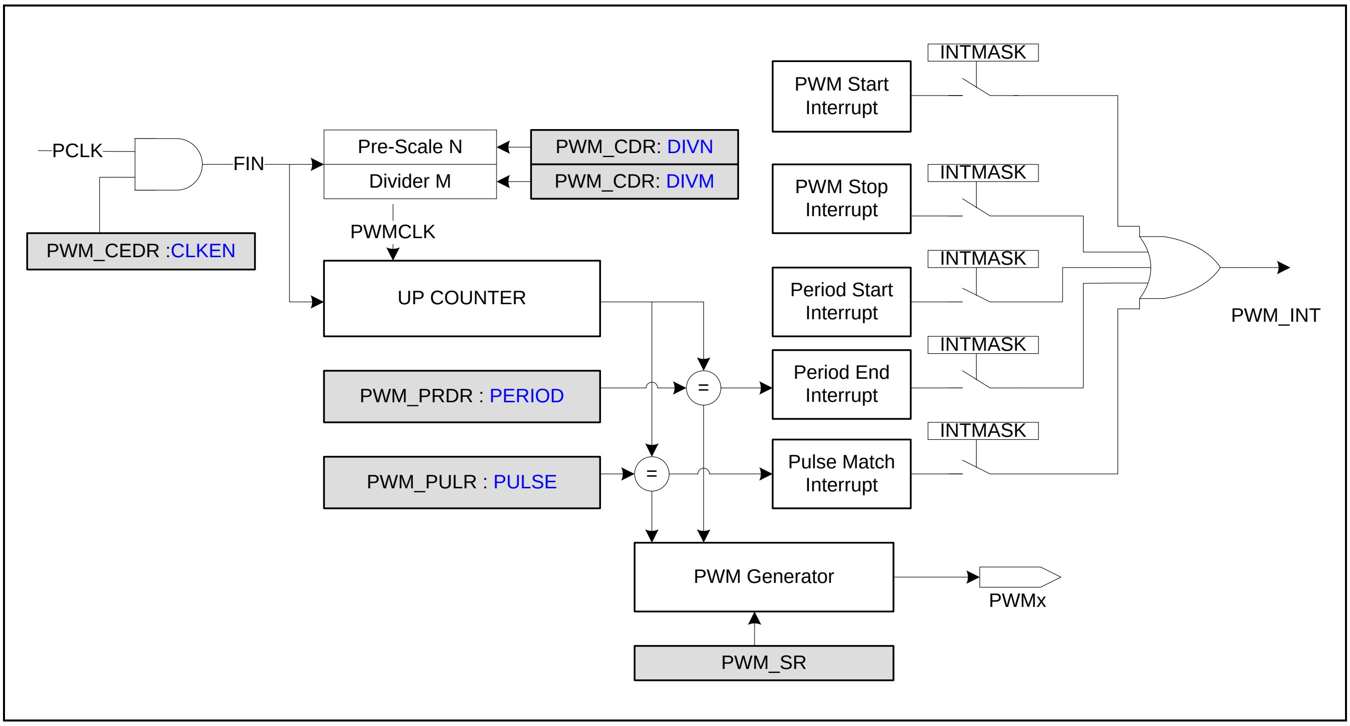


Figure 14-1 Pulse Width Modulation (PWM) Block Diagram

14.2.2 General Description

The PWM generates the periodic waveform. The waveform includes PULSE and nPULSE width.

Figure 14-2 illustrates the diagram of PWM cycle description.

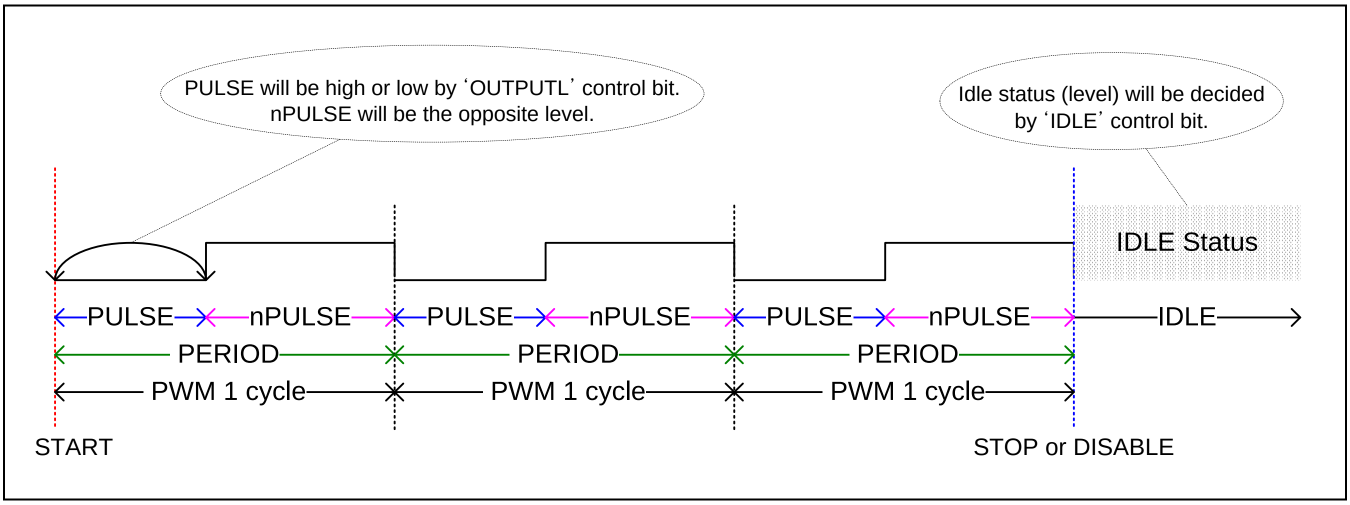

PULSE will be high or low by ʻOUTPUTLʼ control bit.
nPULSE will be the opposite level.
Idle status (level) will be decided by ʻIDLEʼ control bit.
IDLE Status
PULSE
nPULSE
PULSE
nPULSE
PULSE
nPULSE
IDLE
PERIOD
PERIOD
PERIOD
PWM 1 cycle
PWM 1 cycle
PWM 1 cycle
START
STOP or DISABLE

Figure 14-2 PWM Cycle Description

The functions of periodic waveform are:

- PERIOD: The "PERIOD" means the period width. You can define the periodic width (time) of PWM with setting the PERIOD[15:0] field in the PWM_PRDR register.
- PULSE: The "PULSE" means the duty (On time) of PWM. You can define the periodic duty ratio of PWM with setting PULSE[15:0] field in the PWM_PULR register. Also you can select the level of PULSE with setting PWM_CSR/PWM_CCR register.
- nPULSE: (PERIOD-PULSE[15:0]) value calculates the active width (time) for nPulse.
- IDLE: IDLESL bit in PWM_CSR/PWM_CCR register decides the idle level.
- OUTPUTL: Output Level is the level for an active width. It means start level at each period and OUTPUTL bit in the PWM_CSR/PWM_CCR register defines the output level.

The CLKEN bit in the PWM_CEDR (Clock Enable/Disable Register) enables/disables the PWM clock. MCU chip reset or IP software reset, resets the module on the SWRST bit in the PWM_SRR (Software Reset Register).

When you stop or disable PWM, PWM operation should be in IDLE state at next PWM period (cycle).

Caution: The UPDATE bit in the PWM_CSR/PWM_CCR register allows the configuration of new parameters only if the PWM channel disables or associates at the end cycle period

14.2.3 Clock and Operation Frequency

The Clock Source (FIN) is PCLK (Peripheral Clock). The divider value M and pre-scale value N in the PWM_CDR (Clock Divider) register decides the frequency of PWM operation clock.

Defines the PWM counter clock as:

- PWM Counter Clock Frequency, PWMCLK = PCLK/2N/(M + 1)
- $0 \le N < 16$, $0 \le M < 2^{12}$

14.2.4 Period

The "PERIOD" cycle represents periodic PWM output. The PERIOD field in the PWM_PRDR register controls the number of down-counter cycles to fix the period.

- PERIOD[15:0] for the 16-bit PWM period width

If you configure the PERIOD field at 0, then the logical level configured by IDLESL (Idle Level) bit drives the PWM output. The PWM channel automatically disables after setting the UPDATE bit.

14.2.5 PULSE Level

OUTSL bit controls the level of active width to fix the PWM output device. The OUTSL bit affects the PWM output start level.

When the OUTSL bit is set to "0", then the PULSE width level is low and the nPULSE width level is high. On the other hand, when the OUTSL bit is set to "1", then the PULSE width level is high and the nPULSE width becomes low.

14.2.6 nPULSE Width

PWM output starts with PULSE when you enable PWM channel. nPULSE width starts at the end of PULSE width. When you configure a PWM channel to disable in the middle of the PWM PERIOD, it will not disable the PWM output immediately. Disable of the PWM output will be activate at the end of nPULSE. The nPULSE width represents the PWM output nACTIVE state. PULSE field in PWM_PULR register controls the number of up-counter cycles to fix nACTIVE width.

- PULSE[15:0] for the 16-bit PWM PULSE Width

PMATCH bit in the PWM_RISR register indicates the start of nPULSE width and PEND bit indicates the "PERIOD" cycle end. PWM_CSR register bit clears the status of PMATCH and PEND bits. The software has the possibility to enable/disable the PSTA or PEND, or both the interrupts.

If you configure PULSE field at 0, then the PULSE cycle does not exist. The logical level configured by OUTPUTL bit drive PWM output until PWM channel remains enable. For this particular case, consider the configuration set by the UPDATE bit at the end of PERIOD cycle.

14.2.7 IDLE Level

IDLESL bit in the PWM_CSR (Control Set Register)/PWM_CCR (Control Clear Register) controls idle level to fix the PWM output level on the high or low. It is done when PWM stops on the START bit in PWM_CCR (Control Clear Register) or when CLKEN bit in the PWM_CEDR (Clock Enable/Disable Register) disables the clock.

14.2.8 Parameter Relationship

The parameter relationships are:

- DUTY = The duty means the ratio between PERIOD and PULSE.
- DUTY = PULSE[15:0]/PERIOD[15:0]

PERIOD should not be 0. When PULSE[15:0] is 0, the duty is 0 percent. If PULSE[15:0] is greater than PERIOD[15:0], then the duty is 100 percent. This PWM supports 0 percent and 100 percent duty.

Table 14-2 describes the PERIOD and PULSE field relationship in normal mode.

Table 14-2 PERIOD and PULSE Field Relationship in Normal Mode

Duty Ratio	Configuration	Normal Mode
0 %	• PERIOD = M (0x1: 0xFFFF) • PULSE Value = 0	Opposite OUTPUTL bit value (Level)
n %	X > Y • PERIOD = X (0x1: 0xFFFF) • PULSE Value = Y (0x1: 0xFFFF)	Normal PWM Wave
100 %	K ≤ S • PERIOD = K (0x1: 0xFFFF) • PULSE Value = S (0x1: 0xFFFF)	OUTPUTL bit value (Level)

NOTE: In case of an interval mode, if the value of PERIOD is not zero, regardless of PULSE value PWM output will toggle with every PERIOD. In PWM operation, if the PERIOD[15:0] value is 0, it is invalid.

Table 14-3 describes the PWM output.

Table 14-3 PWM Output

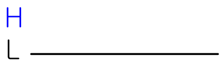
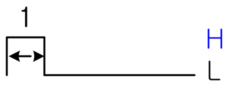
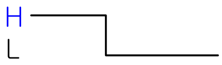
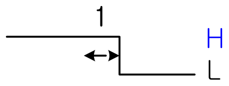
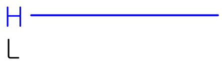
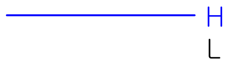
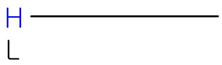
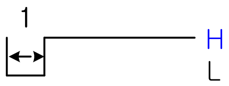
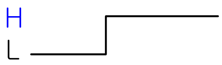
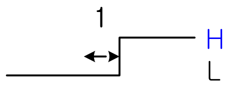
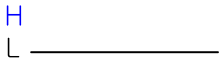
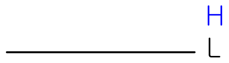
Start	Duty	Normal	Extension
High	0 %		
	N %		
	100 %		
Low	0 %		
	N %		
	100 %		

Figure 14-3 illustrates the PWM basic waveform, OUTSL = 1 diagram.

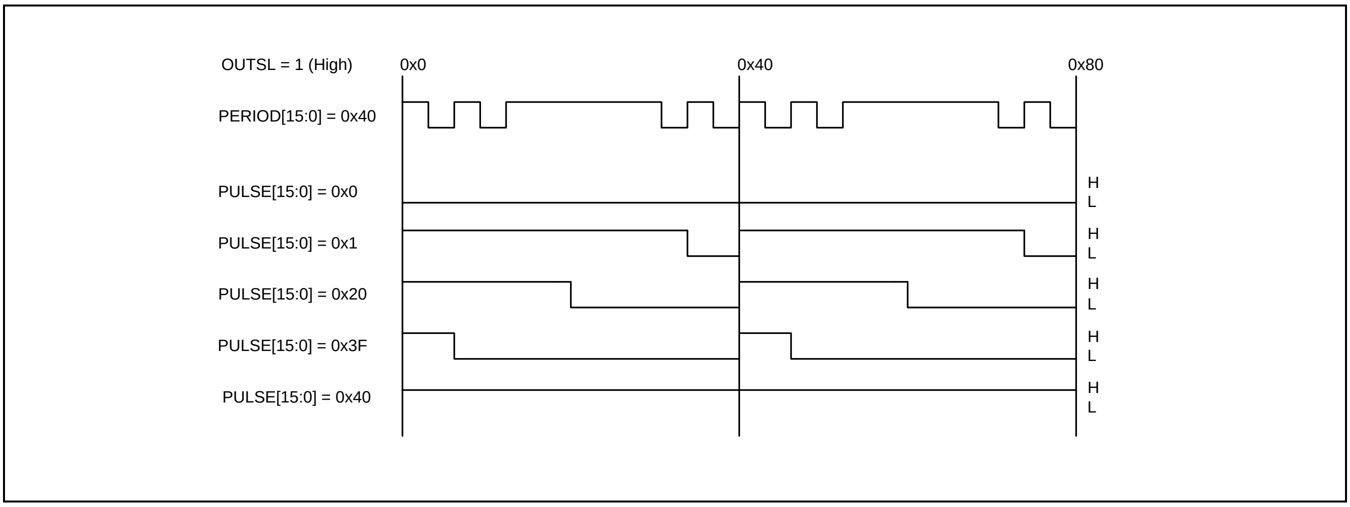


Figure 14-3 PWM Basic Waveform (OUTSL = 1, PWM Period = 0x40, Pulse = 0x0, 0x1, 0x20, 0x3F, 0x40)

Figure 14-4 illustrates the PWM basic waveform, OUTSL = 0 diagram.

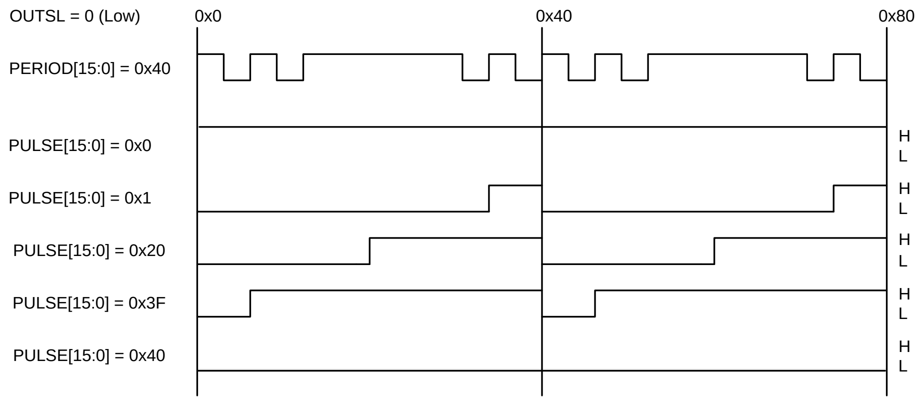


Figure 14-4 PWM Basic Waveform (OUTSL = 0, PWM Period = 0x40, Pulse = 0x0, 0x1, 0x20, 0x3F, 0x40)

14.2.9 Extension Bit

Compare the values of extension counter with extension settings in the extension bits. In every 64 periods some periods are Extension Periods. Extension period with extended Pulse width is a clock longer than normal period. PWMEX bits in TC_SR register determines which period is Extension Period Because PWMEX is 6 bits long the PWM output has approximately up to 22-bit resolution though the counter is 16-bit long. The "stretch" value is an extra clock period at specific intervals or cycles. For example, in 8-bit base and 6-bit extension, value of PWMEX.0 is equals to 1, then 32nd cycle is one pulse longer than the other 63 cycles. If the base duty cycle is 50 percent, then the duty of the 32nd cycle stretches approximately to 51 percent duty

PWMEX Bit	"Stretched" Cycle Number
PWMEX0	32
PWMEX1	16, 48
PWMEX2	8, 24, 40, 56
PWMEX3	4, 12, 20, 28, 36, 44, 52, 60
PWMEX4	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
PWMEX5	1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63

For example, if PWMEX[4:0] bit is set to 1, then all odd-numbered cycles are one pulse longer. If all extension bits, PWMEX[5:0] is set to 1, then all cycles stretches by one pulse except the 64th cycle. Thus, you can obtain high output resolution at high frequencies.

Figure 14-5 illustrates the extended PWM waveform (PWM Period = 0x40, Pulse = 0x3E, PWMEX0).

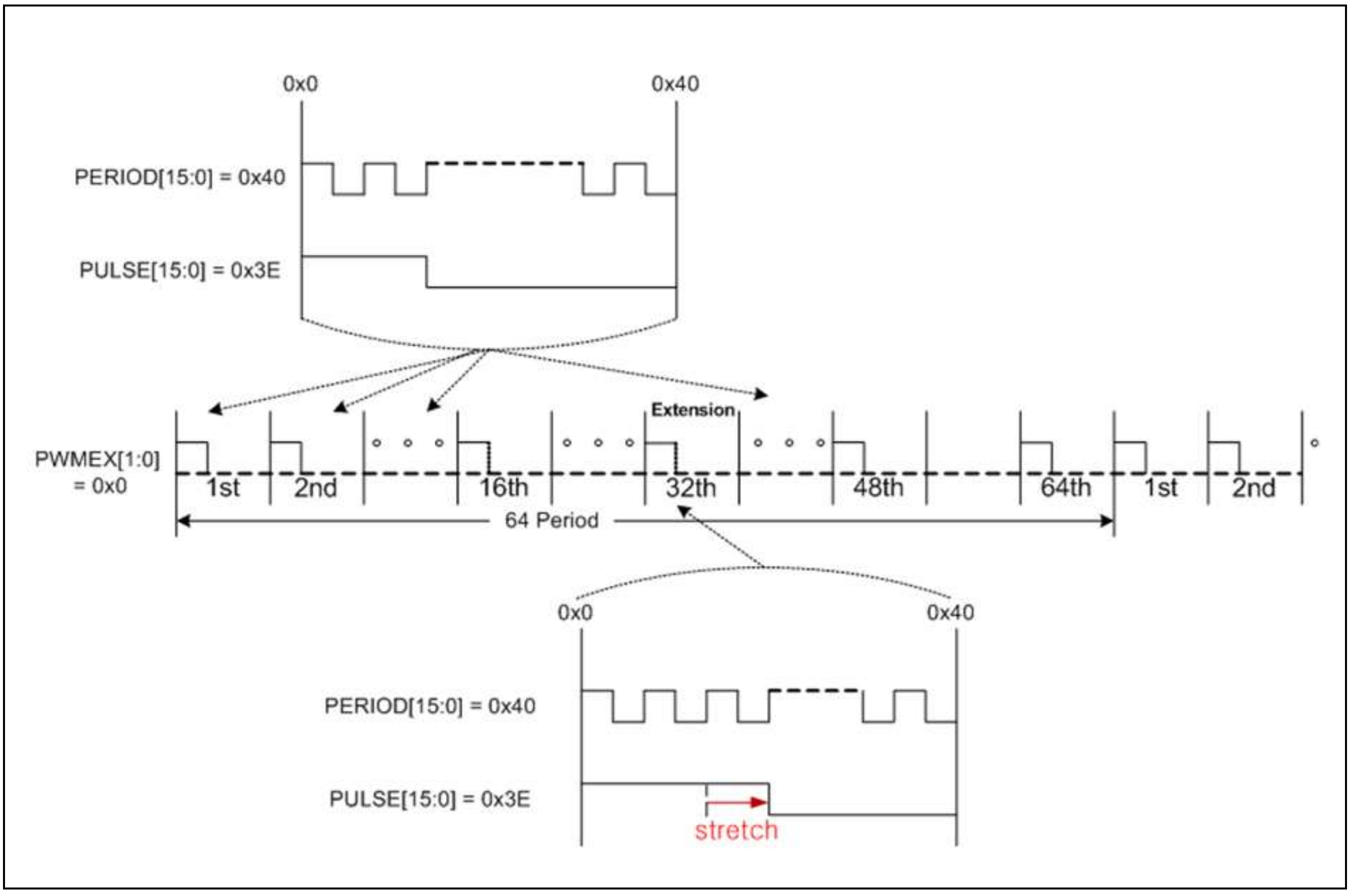


Figure 14-5 Extended PWM Waveform (PWM Period = 0x40, Pulse = 0x3E, PWMEX0)

Figure 14-6 illustrates the extended PWM Waveform (PWM Period = 0x40, Pulse = 0x3E, PWMEX1).

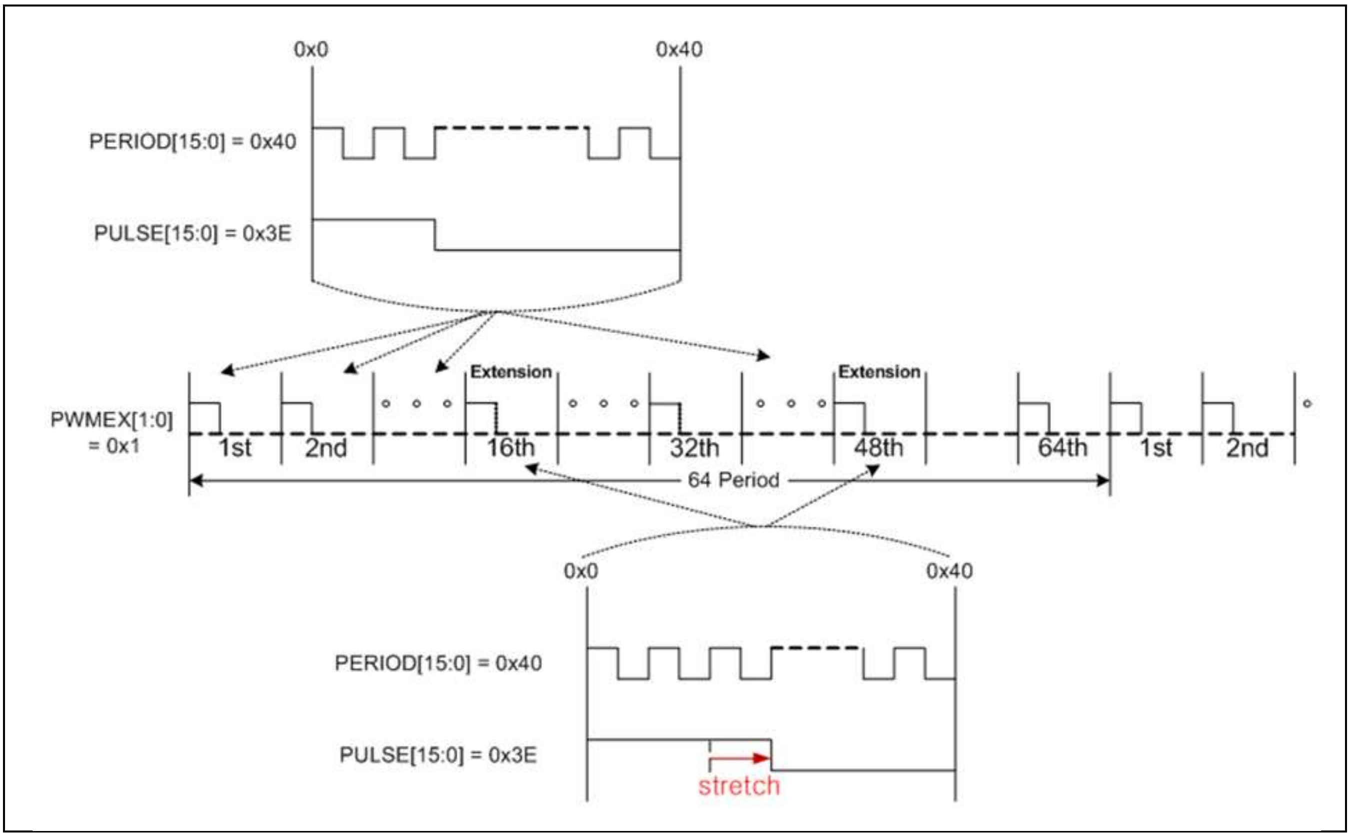


Figure 14-6 Extended PWM Waveform (PWM Period = 0x40, Pulse = 0x3E, PWMEX1)

Figure 14-7 illustrates the extended PWM Waveform (PWM Period = 0x40, Pulse = 0x3E, PWMEX1 and PWMEX0).

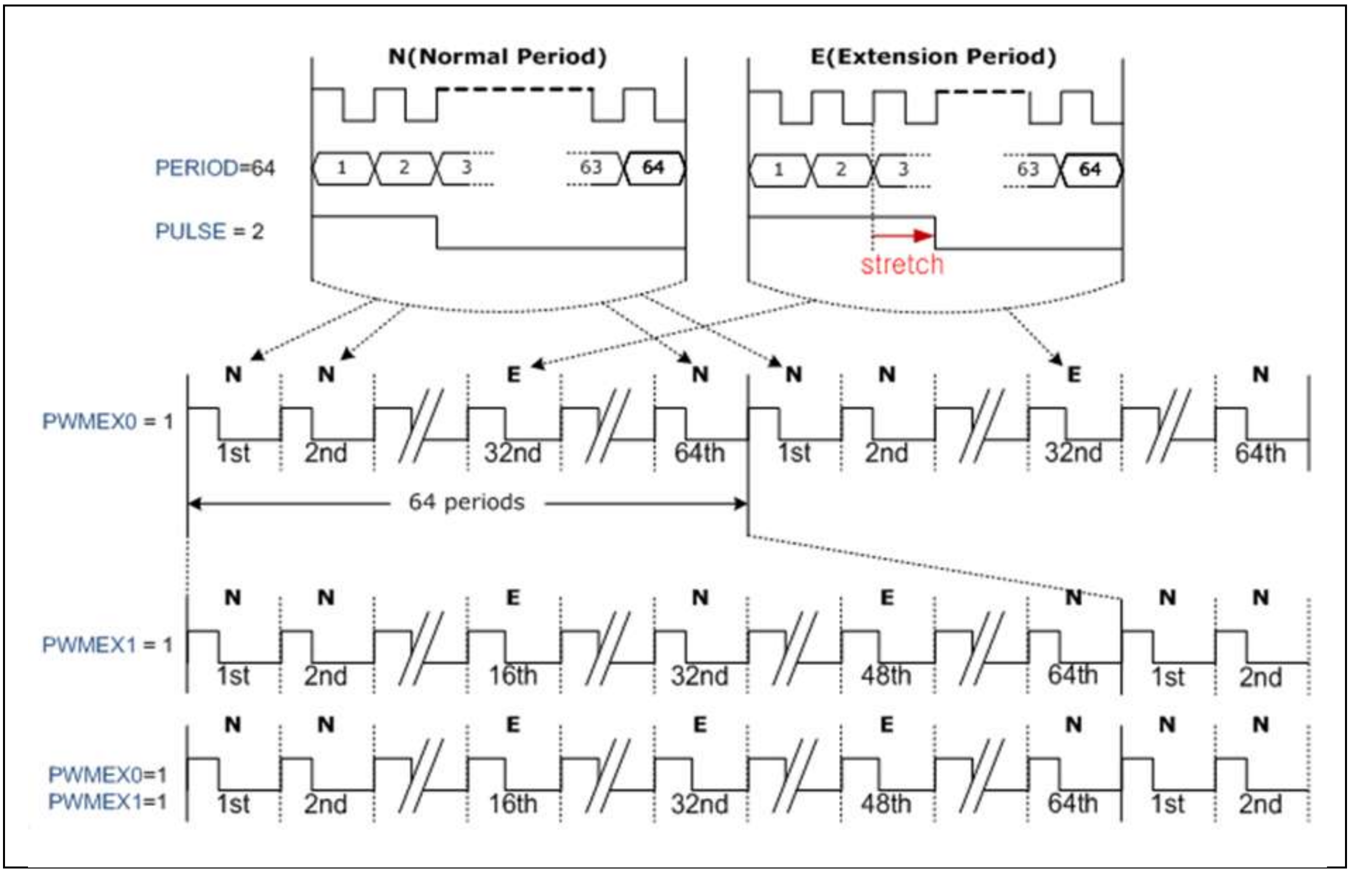


Figure 14-7 Extended PWM Waveform (PWM Period = 0x40, Pulse = 0x3E, PWMEX1 and PWMEX0)

Figure 14-8 illustrates the extended PWM waveform (high start).

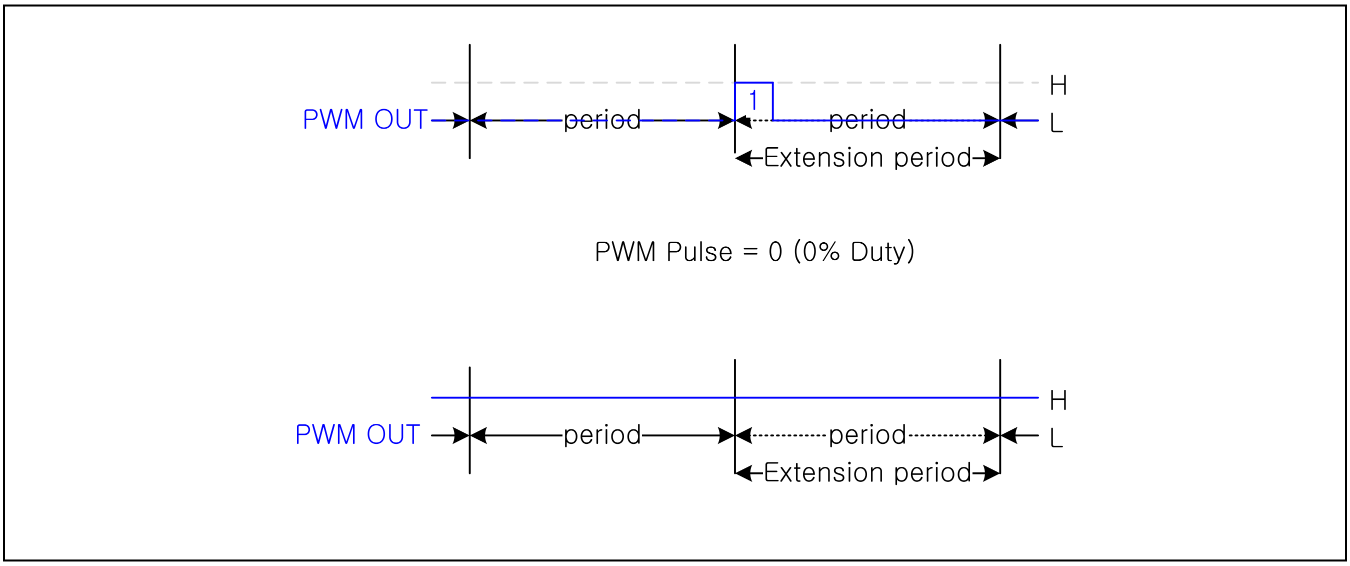


Figure 14-8 Extended PWM Waveform (High Start)

Figure 14-9 illustrates the extended PWM waveform (low start).

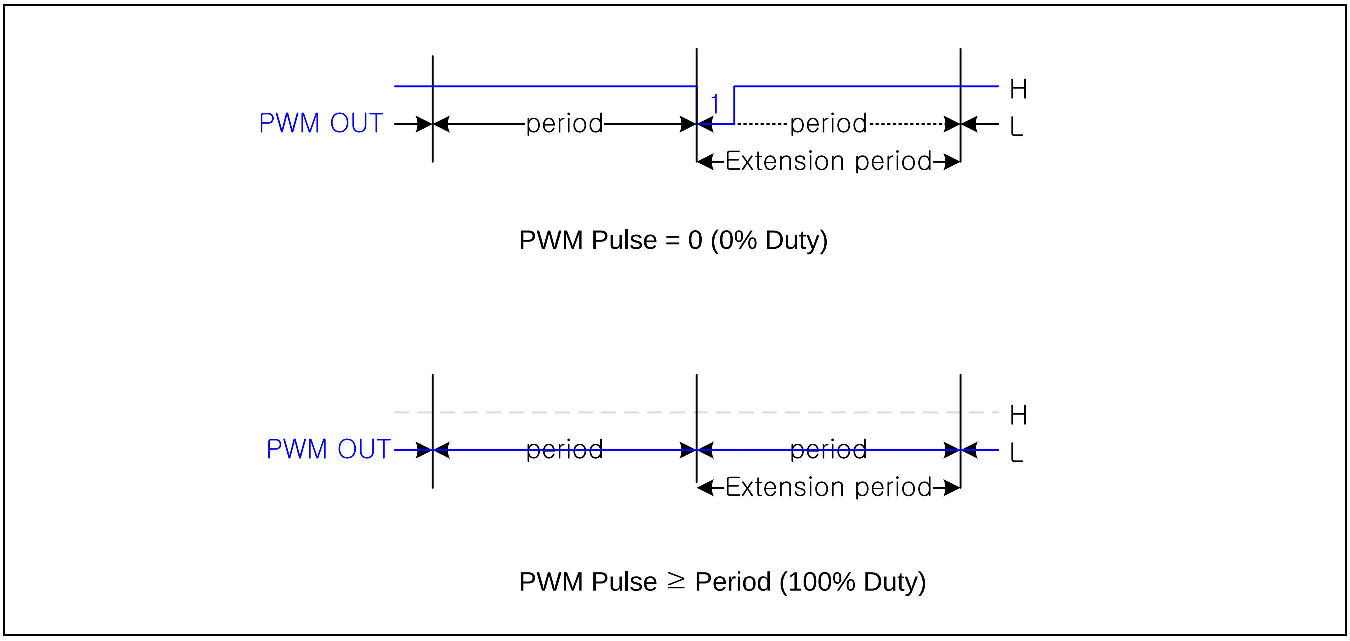


Figure 14-9 Extended PWM Waveform (Low Start)

14.3 Register Description

14.3.1 Register Map Summary

- Base Address: 0x4007_0000, 0x4007_1000, 0x4007_2000, 0x4007_3000

Register	Offset	Description	Reset Value
PWM_IDR	0x0000	ID register	0x0001_0009
PWM_CEDR	0x0004	Clock enable/disable register	0x0000_0000
PWM_SRR	0x0008	Software reset register	0x0000_0000
PWM_CSR	0x000C	Control set register	0x0000_0000
PWM_CCR	0x0010	Control clear register	0x0000_0000
PWM_SR	0x0014	Status register	0x0000_0000
PWM_IMSCR	0x0018	Interrupt mask set/clear register	0x0000_0000
PWM_RISR	0x001C	Raw interrupt status register	0x0000_0000
PWM_MISR	0x0020	Masked interrupt status register	0x0000_0000
PWM_ICR	0x0024	Interrupt clear register	0x0000_0000
PWM_CDR	0x0028	Clock divider register	0x0000_0000
PWM_PRDR	0x002C	Period register	0x0000_0000
PWM_PULR	0x0030	Pulse register	0x0000_0000
PWM_CCDR	0x0034	Current clock divider register	0x0000_0000
PWM_CPRDR	0x0038	Current period register	0x0000_0000
PWM_CPULR	0x003C	Current pulse register	0x0000_0000

14.3.1.1 PWM_ID

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0000, Reset Value = 0x0001_0009

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores the ID code for the corresponding IP.	0x0001_0009

14.3.1.2 PWM_CEDR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	RW	Debug mode enable 0 = Disables debug mode Debugger interface generates Debug Acknowledge that has no influence on PWM function 1 = Enables debug mode When you activate the debugger interface, the Debug Acknowledge freezes the PWM function. However, keep the full Read/Write access to internal register for debug purpose.	0
RSVD	[30:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable/Disable Bit 0 = Disables PWM Clock bit 1 = Enables PWM Clock bit PWM software reset does not affect CLKEN bit status.	0

14.3.1.3 PWM_SRR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Performs PWM Software Reset operation	0

14.3.1.4 PWM_CSR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD		PWMEX5	PWMEX4	PWMEX3	PWMEX2	PWMEX1	PWMEX0	RSVD												PWMIM	KEEP	OUTSL	IDLESL	RSVD						UPDATE	START
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	W	W	W	W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	R	R	R	R	R	R	W	W

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="2">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:30]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td rowspan="9">PWMEX5
PWMEX4
PWMEX3
PWMEX2
PWMEX1
PWMEX0</td><td rowspan="9">[29]
[28]
[27]
[26]
[25]
[24]</td><td rowspan="9">W</td><td colspan="2">PWM Extension Control Bit
0 = No effect
1 = Includes the corresponding stretched cycle number</td><td rowspan="9">0</td></tr>
<tr><td><b>PWMEXy</b></td><td><b>"Stretched" Cycle Number</b></td></tr>
<tr><td>PWMEX0</td><td>32</td></tr>
<tr><td>PWMEX1</td><td>16, 48</td></tr>
<tr><td>PWMEX2</td><td>8, 24, 40, 56</td></tr>
<tr><td>PWMEX3</td><td>4, 12, 20, 28, 36 , 44, 52, 60</td></tr>
<tr><td>PWMEX4</td><td>2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62</td></tr>
<tr><td>PWMEX5</td><td>1, 3, 5, 7, 9,11,13,15,17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63</td></tr>
<tr><td colspan="2">If PWMEX0 and PWMEX3 are set to "1", then the PWM inserts extra 1-cycle at (32) and (4, 12, 20, 28, 36, 44, 52, 60).</td></tr>
<tr><td>RSVD</td><td>[23:12]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td>PWMIM</td><td>[11]</td><td>W</td><td colspan="2">PWM Interval Mode
0 = No effect
1 = Specifies PWM Interval Mode.
PWM phase toggles when a period matches.</td><td>0</td></tr>
<tr><td>KEEP</td><td>[10]</td><td>W</td><td colspan="2">Keep Last Period State
0 = No effect
1 = Keep the PWM output level when PWM stops
When PWM is restarted, PWM output level begins at the kept PWM output level regardless of IDLE level.</td><td>0</td></tr>
</table>

Name	Bit	Type	Description	Reset Value
OUTSL	[9]	W	PWM Output Start Level 0 = No effect 1 = Starts PWM output level from high (Logic-1) for the specified PERIOD.	0
IDLESL	[8]	W	IDLE State Level 0 = No effect 1 = Idle State PWM Output Level is High (Logic-1)	0
RSVD	[7:2]	R	Reserved	0
UPDATE	[1]	W	Update PWM Parameter 0 = No effect 1 = Updates PWM Parameter (Period, Pulse, and Clock Divider)	0
START	[0]	W	Start PWM 0 = No effect 1 = Starts PWM Operation	0

14.3.1.5 PWM_CCR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD		PWMEX5	PWMEX4	PWMEX3	PWMEX2	PWMEX1	PWMEX0	RSVD												PWMIM	KEEP	OUTSL	IDLESL	RSVD							START
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	W	W	W	W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	R	R	R	R	R	R	R	W

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="2">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:30]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td rowspan="9">PWMEX5
PWMEX4
PWMEX3
PWMEX2
PWMEX1
PWMEX0</td><td rowspan="9">[29]
[28]
[27]
[26]
[25]
[24]</td><td rowspan="9">W</td><td colspan="2">PWM Extension Control Bit
0 = No effect
1 = Deletes the corresponding stretched cycle number</td><td rowspan="9">0</td></tr>
<tr><td><b>PWMEXy</b></td><td><b>"Stretched" Cycle Number</b></td></tr>
<tr><td>PWMEX0</td><td>32</td></tr>
<tr><td>PWMEX1</td><td>16, 48</td></tr>
<tr><td>PWMEX2</td><td>8, 24, 40, 56</td></tr>
<tr><td>PWMEX3</td><td>4, 12, 20, 28, 36 , 44, 52, 60</td></tr>
<tr><td>PWMEX4</td><td>2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62</td></tr>
<tr><td>PWMEX5</td><td>1, 3, 5, 7, 9,11,13,15,17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63</td></tr>
<tr><td colspan="2">NOTE: Combination of six control bits can control PWMEXy.

For example:
• 1 Extension Case: (PWMEX0 = 1), (PWMEX1 = 1), (PWMEX2 = 1), (PWMEX3 = 1), (PWMEX4 = 1), (PWMEX5 = 1)
• 2 Extension Case: (PWMEX0 = 1 and PWMEX5 = 1), (PWMEX2 =1, PWMEX3 = 1), and so on.
• 3 Extension Case: (PWMEX1 = 1, PWMEX3 = 1, PWMEX4 = 1), (PWMEX0 = 1, PWMEX2 = 1, PWMEX5 = 1), and so on.</td></tr>
<tr><td>RSVD</td><td>[23:12]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
</table>

Name	Bit	Type	Description	Reset Value
PWMIM	[11]	W	PWM Interval Mode 0 = No effect 1 = Specifies PWM interval mode. PWM phase toggles every period.	0
KEEP	[10]	W	Keep Last Period State 0 = No effect 1 = Keep the PWM output level as last period output level when PWM stops. In this case, Ignore IDLE level.	0
OUTSL	[9]	W	PWM Start Level 0 = No effect 1 = Starts PWM output level from low (Logic-0) for the specified PERIOD.	0
IDLESL	[8]	W	Idle State Level 0 = No effect 1 = Idle State PWM output level is low (Logic-0)	0
RSVD	[7:1]	R	Reserved	0
START	[0]	W	Start PWM 0 = No effect 1 = Stops PWM Operation	0

14.3.1.6 PWM_SR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD		PWMEX5	PWMEX4	PWMEX3	PWMEX2	PWMEX1	PWMEX0	RSVD												PWMIM	KEEP	OUTSL	IDLESL	RSVD							START
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:30]	R	Reserved	0
PWMEX5 PWMEX4 PWMEX3 PWMEX2 PWMEX1 PWMEX0	[29] [28] [27] [26] [25] [24]	R	PWM Extension Bit Control 0 = PWM Extension is not effective to the corresponding bit-field. 1 = PWM Extension is effective to the corresponding bit-field.	0
RSVD	[23:12]	R	Reserved	0
PWMIM	[11]	R	PWM Output Mode Status 0 = PWM Mode 1 = Interval Mode	0
KEEP	[10]	R	Keep Last Period Status 0 = IDLESL determines the PWM output level when PWM stops 1 = Keeps the PWM output level as last period output level when PWM stops. Ignores IDLESL in this case	0
OUTSL	[9]	R	PWM Output Start Level Status 0 = Starts PWM output level from LOW (Logic-0) for the PERIOD that it specifies. 1 = Starts PWM Output Level from High (Logic-1) for the PERIOD that it specifies.	0
IDLESL	[8]	R	IDLE State PWM Output Level Status 0 = Idle state PWM output level is low (Logic-0) 1 = Idle state PWM output level is high (Logic-1)	0
RSVD	[7:1]	R	Reserved	0
START	[0]	R	PWM Start/Stop Status 0 = Stops PWM 1 = Starts PWM	0

14.3.1.7 PWM_IMSCR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																											PMATCH	PEND	PSTART	PWMSTOP	PWMSTART
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	R	Reserved	0
PMATCH	[4]	RW	Pulse Match Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (disables the interrupt))	0
PEND	[3]	RW	Period End Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (disables the interrupt)	0
PSTART	[2]	RW	Period Start Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (disables the interrupt)	0
PWMSTOP	[1]	RW	PWM Stop Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (disables the interrupt)	0
PWMSTART	[0]	RW	PWM Start Interrupt Mask 0 = Mask the interrupt (disables the interrupt) 1 = Unmask the interrupt (disables the interrupt)	0

NOTE: On a Read, PWM_IMSCR register gives current value of mask on the relevant interrupt. A Write of 1 to a particular bit, sets the mask and enables the interrupt to be read. A Write of 0 clears the corresponding mask.

14.3.1.8 PWM_RISR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																											PMATCH	PEND	PSTART	PWMSTOP	PWMSTART
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	R	Reserved	0
PMATCH	[4]	R	Pulse Match Raw Interrupt State Gives the raw interrupt state (prior to masking) of PMATCH interrupt.	0
PEND	[3]	R	Period End Raw Interrupt State Gives the raw interrupt state (prior to masking) of PEND interrupt.	0
PSTART	[2]	R	Period Start Raw Interrupt State Gives the raw interrupt state (prior to masking) of PSTART interrupt.	0
PWMSTOP	[1]	R	PWM Stop Raw Interrupt State Gives the raw interrupt state (prior to masking) of PWMSTOP interrupt.	0
PWMSTART	[0]	R	PWM Start Raw Interrupt State Gives the raw interrupt state (prior to masking) of PWMSTART interrupt.	0

NOTE: On a Read, PWM_RISR register gives current raw status value of the corresponding interrupt prior to masking. A Write has no effect.

14.3.1.9 PWM_MISR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																											PMATCH	PEND	PSTART	PWMSTOP	PWMSTART
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	R	Reserved	0
PMATCH	[4]	R	Pulse Match Masked Interrupt State Gives the masked interrupt state (prior to masking) of PMATCH interrupt	0
PEND	[3]	R	Period End Masked Interrupt State Gives the masked interrupt state (prior to masking) of PEND interrupt	0
PSTART	[2]	R	Period Start Masked Interrupt State Gives the masked interrupt state (prior to masking) of PSTART interrupt	0
PWMSTOP	[1]	R	PWM Stop Masked Interrupt State Gives the masked interrupt state (prior to masking) of PWMSTOP interrupt	0
PWMSTART	[0]	R	PWM Start Masked Interrupt State Gives the masked interrupt state (prior to masking) of PWMSTART interrupt	0

NOTE: On a Read, PWM_MISR register gives current masked status value of the corresponding interrupt.
A Write has no effect.

14.3.1.10 PWM_ICR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0024, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																											PMATCH	PEND	PSTART	PWMSTOP	PWMSTART
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	R	Reserved	0
PMATCH	[4]	W	PWM Counter Value is Matched with Pulse Value Status 0 = No effect 1 = Clear pulse match interrupt	0
PEND	[3]	W	PWM Period End Interrupt clear 0 = No effect 1 = Clears PWM Period Ended interrupt	0
PSTART	[2]	W	PWM Period Start Interrupt clear 0 = No effect 1 = Clears PWM period started interrupt	0
PWMSTOP	[1]	W	PWM Stop Interrupt clear 0 = No effect 1 = Clears PWM stopped interrupt	0
PWMSTART	[0]	W	PWM Start Interrupt clear 0 = No effect 1 = Clears PWM started interrupt	0

NOTE: On a Write of 1, clears the corresponding interrupt. A Write of 0 has no effect.

14.3.1.11 PWM_CDR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:15]	R	Reserved	0
DIVM	[14:4]	RW	Clock divider value PWMCLK/(M + 1), where(0 ≤ M < 2^{12})	0x000
DIVN	[3:0]	RW	Pre-scale value PWMCLK/2^N, where (0 ≤ N < 16)	0x0

14.3.1.12 PWM_PRDR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PERIOD	[15:0]	RW	PWM Period Value Refer to Section *14.2.8 Parameter Relationship*.	0x0000

14.3.1.13 PWM_PULR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																PULSE															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PULSE	[15:0]	RW	PWM Pulse Value Refer to Section *14.2.8 Parameter Relationship*.	0x0000

14.3.1.14 PWM_CCDR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0034, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																	DIVM											DIVN			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:15]	R	Reserved	0
DIVM	[14:4]	R	Current Clock Divider Value PWMCLK/(M + 1), where (0 ≤ M < 2^{12})	0x000
DIVN	[3:0]	R	Current Pre-scale Value PWMCLK/2^{N}, where (0 ≤ N < 16)	0x0

14.3.1.15 PWM_CPRDR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x0038, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PERIOD	[15:0]	R	PWM Current Period Value This field shows operating PWMs current period value.	0x0000

14.3.1.16 PWM_CPULR

- Base Address: 0x4007_0000, 0x4007_1000
- Address = Base Address + 0x003C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PULSE	[15:0]	R	PWM Pulse Value This field shows operating PWMs current pulse value.	0x0000

15 Serial Peripheral Interface (SPI)

15.1 Overview

The PrimeCell Synchronous Serial Port (SSP, PL022) is used for Serial Peripheral Interface. Serial communication is the process of sequentially sending data one bit at a time.

About the ARM PrimeCell SSP (PL022)

The PrimeCell Synchronous Serial Port (SSP) is an Advanced Microcontroller Bus Architecture (AMBA) slave block that connects to the Advanced Peripheral Bus (APB). The PrimeCell SSP is an AMBA compliant System-on-Chip (SoC) peripheral. That is developed, tested, and is licensed by ARM.

15.1.1 Features

This section describes the features of SPI.

The PrimeCell SSP is a master or slave interface that enables synchronous serial communication with slave or master peripherals having:

- A Motorola SPI-compatible interface.

In both the master and slave configurations, the PrimeCell SSP performs:

- A parallel-to-serial conversion on the data written to an internal 16-bit wide, and 8-location deep transmit First-In First-Out (FIFO).
- A serial-to-parallel conversion on the received data, buffering it in a similar 16-bit wide, and 8-location deep receive FIFO.

Interrupts are generated to:

- Request servicing for transmit and receive FIFO
- Inform the system that a receive FIFO over-run has occurred
- Inform the system that the data is present in the receive FIFO after an idle period has expired

15.1.1.1 Features of the PrimeCell SSP

The features of the PrimeCell are:

- Compliant to the AMBA Specification (Revision 2.0) for easy integration into SOC implementation
- Master or slave operation
- Programmable clock bit-rate and pre-scale
- Separate transmit and receive FIFO memory buffers, 16-bit wide, and 8 locations deep
- Programmable data frame size of 4-16 bits
- Independent masking of transmit FIFO, receive FIFO, and receive overrun interrupts
- Internal loopback test mode.

15.1.1.2 Programmable Parameters

The programmable parameters are:

- Master or slave mode
- Enabling of operation
- Frame format
- Communication baud rate
- Clock phase and polarity
- Data widths of 4-16 bits
- Interrupt masking

15.1.1.3 SPI Features

The features of the Motorola SPI-compatible interface are:

- Full-duplex, four-wired synchronous transfer
- Programmable clock polarity and phase

15.1.2 Pin Description

Table 15-1 describes the pin description of the SSP.

Table 15-1 SSP Pin Description

Pin Name	Function	I/O Type	Comments
SCK	SPI serial clock	I/O	–
MOSI	Master out slave in	I/O	–
MISO	Master in slave out	I/O	–
FSS	Frame, slave select (master) Frame input (slave)	I/O	–

15.2 Functional Description

The functional description section includes:

- Block Diagram
- Operation
- Register Description

15.2.1 Block Diagram

Figure 15-1 illustrates the SSP block diagram.

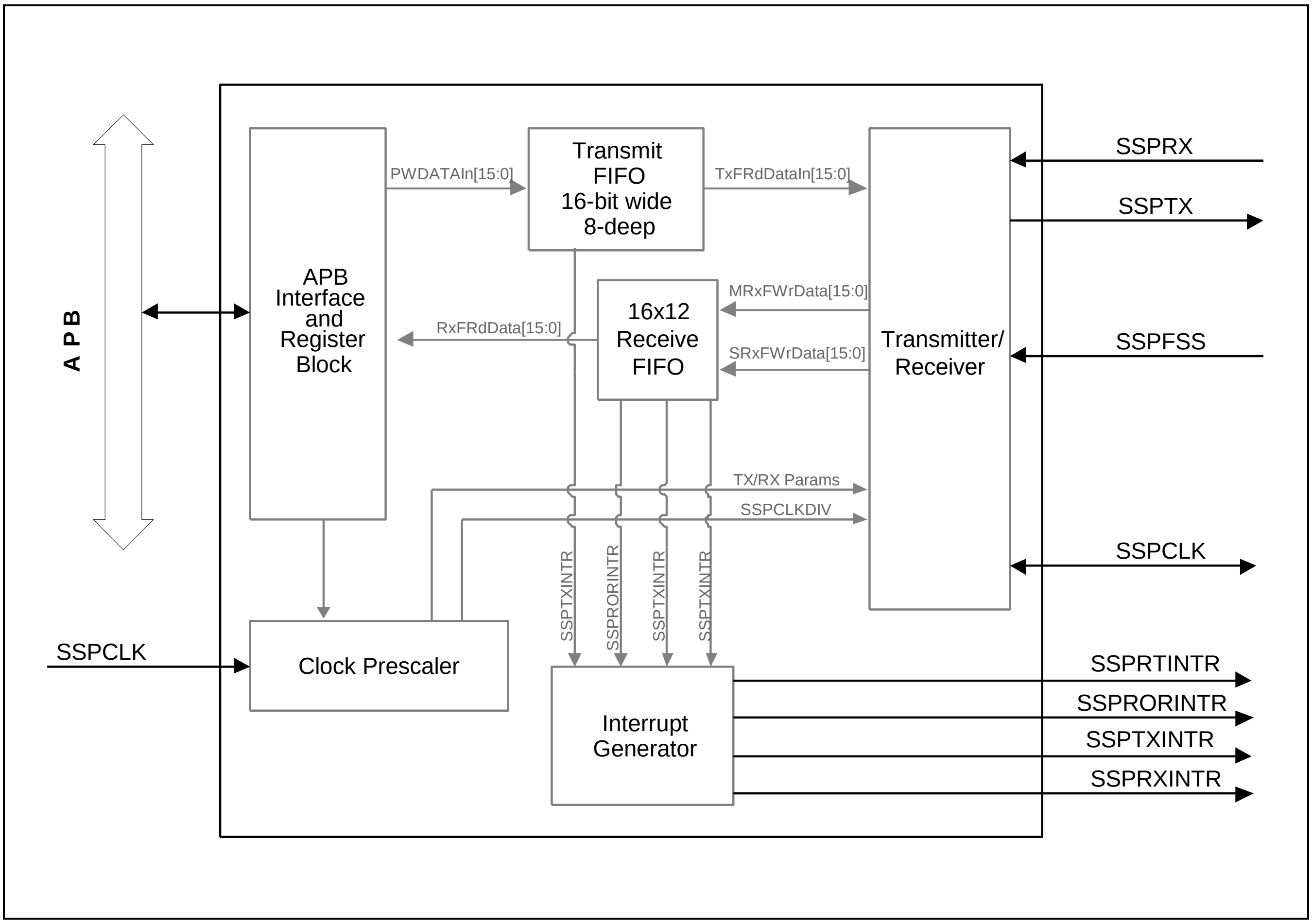


Figure 15-1 SSP Block Diagram

15.2.2 Operation

The operation section describes the functional description of SPI. ARM PrimeCell, PrimeCell SSP, Register Block, Pre-scaler, and FIFO (receive and transmit).

15.2.2.1 Function Description

This section describes the overview and functionality of ARM PrimeCell SSP.

15.2.2.1.1 ARM PrimeCell SSP (PL022) Overview

The PrimeCell SSP is either a master or slave interface for synchronous serial communication with peripheral devices having Motorola SPI. The PrimeCell SSP performs serial-to-parallel conversion on the data received from a peripheral device. The CPU accesses data, control, and status information through the AMBA APB interface. Transmit and receive paths are buffered with internal FIFO memories allowing up to eight 16-bit values to be stored independently in both transmit and the receive modes. The serial data is transmitted on SSPTXD and received on SSPRXD. The PrimeCell SSP includes a programmable bit-rate clock divider and pre-scaler to generate the serial output clock SSPCLK from the input clock FSSPCLK. Bit rates are supported up to 2 MHz and higher, subjected to choice of frequency for SSPCLK. The maximum bit-rate is determined by the peripheral devices. The PrimeCell SSP operating mode, frame format, and size are programmed through the control registers namely, SSPCR0 and SSPCR1.

The four individually generated output interrupts that can be masked are, SSPTXINTR, SSPRXINTR, SSPRORINTR, and SSPRTINTR. These contribute to:

- SSPTXINTR requests servicing of the transmit buffer
- SSPRXINTR requests servicing of the receive buffer
- SSPRORINTR indicates an overrun condition in the receive FIFO
- SSPRTINTR indicates that a timeout period expired while the data was present in the receive FIFO.

Depending on the operating mode selected, the SSPFSSOUT output will operate as an active-low slave-select for the SPI.

15.2.2.1.2 PrimeCell SSP Functional Description

The functional descriptions for PrimeCell SSP are:

- AMBA APB Interface
- Register Block
- Clock Pre-scaler
- Transmit FIFO
- Receive FIFO
- Transmit and Receive Logic
- Interrupt Generation Logic

15.2.2.1.2.1 AMBA APB Interface

The AMBA APB interface generates Read and Write decodes for accessing:

- Status and control registers
- Transmit and receive FIFO memories

The AMBA APB is a local secondary bus that provides a low-power extension to the higher bandwidth AMBA Advanced High-performance Bus (AHB) within the AMBA system hierarchy. The AMBA APB groups narrow-bus peripherals to avoid loading the system bus and provides an interface using memory-mapped registers that can be accessed under programmed control.

15.2.2.1.2.2 Register Block

The register block stores the data written or to be read across the AMBA APB interface.

15.2.2.1.2.3 Clock Pre-Scaler

When configured as a master, an internal pre-scaler comprising of two free-running reloadable serially linked counters, is used to provide the serial output clock (SSPCLKOUT). The clock pre-scaler is programmable through the SSPCPSR register that divides the FSSPCLK by a factor of 2 to 254 in steps of two. By not utilizing the least significant bit of the SSPCPSR register, division by an odd number is not possible ad this ensures a symmetrical (equal mark space ratio) clock is generated.

The output of the pre-scaler is further divided by a factor of 1 to 256, through the programming of the SSPCR0 control register, to give the final master output clock SSPCLKOUT.

15.2.2.1.2.4 Transmit FIFO

The common Transmit FIFO is a 16-bit wide, 8-locations deep, FIFO memory buffer. CPU data written across the AMBA APB interface are stored in the buffer until read out by the transmit logic.

When configured as a master or a slave parallel data is written into the transmit FIFO prior to serial conversion and transmission to the attached slave or master respectively, through the SSPTXD pin.

15.2.2.1.2.5 Receive FIFO

The common Receive FIFO is a 16-bit wide, 8-locations deep, FIFO memory buffer. Received data from the serial interface are stored in the buffer until read out by the CPU across the AMBA APB interface.

When configured as a master or a slave, the serial data received through the SSPRXD pin is registered prior to parallel loading into the attached slave or master Receive FIFO respectively.

15.2.2.1.2.6 Transmit and Receive Logic

When configured as a master, the clock to the attached slaves is derived from a divided down version of FSSPCLK through the pre-scaler operations. The master transmit logic successively reads a value from its Transmit FIFO and performs a parallel to serial conversion on it. Then the serial data stream and frame control signal, synchronized to SSPCLKOUT, are output through the SSPTXD pin to the attached slaves. The master receive logic performs a serial to parallel conversion on the incoming synchronous SSPRXD data stream, extracting and storing values into its Receive FIFO, for subsequent reading through the APB interface.

When configured as a slave, the SSPCLKIN clock is provided by an attached master and used to time its transmission and reception sequences.

The slave transmits logic that is:

- Under the control of the master clock
- Successively reads a value from its Transmit FIFO
- Performs parallel to serial conversion
- Outputs the serial data stream and frame control signal through the slave SSPTXD pin.

The slave receives logic to perform:

- Serial to parallel conversion on the incoming SSPRXD data stream
- Extracting and storing values into its Receive FIFO for subsequent reading through the APB interface

15.2.2.1.2.7 Interrupt Generation Logic

Four individual maskable, active high interrupts are generated by the PrimeCell SSP.

The individual interrupt requests could also be used with a system interrupt controller that provides masking for the outputs of each peripheral. In this way, a global interrupt controller service routine will be able to read the entire set of sources from one wide register in the system interrupt controller. This is attractive where the time to read from the peripheral registers is significant compared to the CPU clock speed in a real-time system.

This peripheral supports both the methods.

The transmit and receive dynamic data-flow interrupts, SSPTXINTR and SSPRXINTR, are separated from the status interrupts so that data can be read or written in response to the FIFO trigger levels.

15.2.2.1.3 PrimeCell SSP Operation

PrimeCell SSP Operation section includes:

- Interface Reset
- Configuring the SSP
- Enabling PrimeCell SSP Operation
- Clock Ratio
- Programming the SSPCR0 Control Register

15.2.2.1.3.1 Interface Reset

The PrimeCell SSP is reset by the global reset signal PRESETn and a block-specific reset signal nSSPRST. An external reset controller should use PRESETn to assert nSSPRST asynchronously and negate it synchronously to SSPCLK. PRESETn should be asserted LOW for a period long enough to reset the slowest block in the on-chip system, and then taken HIGH again. The PrimeCell SSP requires PRESETn to be asserted LOW for at least one period of PCLK.

15.2.2.1.3.2 Configuring the SSP

Following reset, the PrimeCell SSP logic is disabled and must be configured when in this state. Control registers SSPCR0 and SSPCR1 need to be programmed to configure the peripheral as a master or a slave operating under Motorola SPI.

The bit-rate derived from the external SSPCLK requires the programming of the clock pre-scale register SSPCPSR.

15.2.2.1.3.3 Enable PrimeCell SSP Operation

The Transmit FIFO can either be primed by writing up to eight 16-bit values when the PrimeCell SSP is disabled, or allow the transmit FIFO service request to interrupt the CPU. Once enabled, the transmission or reception of data begins on the transmit (SSPTXD) and receive (SSPRXD) pins.

15.2.2.1.3.4 Clock Ratios

The clock ratio is defined as the ratio of the frequencies of PCLK to SSPCLK. The frequency of SSPCLK must be less than or equal to that of PCLK. This ensures that control signals from the SSPCLK domain to the PCLK domain are certain to get synchronized before one frame duration:

- $F_{SSPCLK} \leftarrow F_{PCLK}$

In the slave mode of operation, the SSPCLKIN signal from the external master is double synchronized and then delayed to detect an edge. It takes three SSPCLKs to detect an edge on SSPCLKIN. The SSPTXD pin has less setup time to the falling edge of SSPCLKIN on which the master is sampling the line. The setup and hold times on SSPRXD with reference to SSPCLKIN must be more conservative to ensure that it is at the right value when the actual sampling occurs within the SSPMS. To ensure correct device operation, SSPCLK must be at least 12 times faster than the maximum expected frequency of the SSPCLKIN.

The frequency selected for SSPCLK must accommodate the desired range of bit clock rates. The ratio of minimum SSPCLK frequency to the SSPCLKOUT maximum frequency in the case of slave mode is 12 and for the master mode it is two.

To generate a maximum bit-rate of 1.8432 Mbps in the master mode, the frequency of SSPCLK must be at least 3.6864 MHz. With an SSPCLK frequency of 3.6864 MHz, the SSPCPSR register has to be programmed with a value of two and the SCR[7:0] field in the SSPCR0 register needs to be programmed as zero.

To work with a maximum bit-rate of 1.8432 Mbps in the slave mode, the frequency of SSPCLK must be at least 22.12 MHz. With an SSPCLK frequency of 22.12 MHz, the SSPCPSR register can be programmed with a value of 12 and the SCR[7:0] field in the SSPCR0 register can be programmed as zero. Similarly, the ratio of the SSPCLK maximum frequency to SSPCLKOUT minimum frequency is 254 × 256.

The minimum frequency of SSPCLK is governed by the following equations, both of which have to be satisfied:

- F_{SSPCLK} (Minimum) $\rightarrow 2 \times F_{SSPCLKOUT}$ (Maximum) [for Master Mode]
- F_{SSPCLK} (Minimum) $\rightarrow 12 \times F_{SSPCLKIN}$ (Maximum) [for Slave Mode]

The maximum frequency of SSPCLK is governed by the following equations, both of which have to be satisfied:

- F_{SSPCLK} (Maximum) $\leftarrow 254 \times 256 \times F_{SSPCLKOUT}$ (Minimum) [for Master Mode]
- F_{SSPCLK} (Maximum) $\leftarrow 254 \times 256 \times F_{SSPCLKIN}$ (Minimum) [for Slave Mode]

15.2.2.1.3.5 Programming the SSPCR0 Control Register

The SSPCR0 register is used to:

- Program the Serial Clock Rate (SCR).
- Select one of the three protocols.
- Select the data word size (where applicable).

The SCR value, in conjunction with the SSPCPSR Clock Prescale Divisor Value (CPSDVSR) is used to derive the PrimeCell SSP transmit and receive bit rate from the external SSPCLK. The frame format is programmed through the FRF bits and the data word size through the DSS bits. Bit phase and polarity, applicable to Motorola SPI format only, are programmed through the SPH and SPO bits.

15.2.2.1.3.6 Programming the SSPCR1 Control Register

The SSPCR1 register is used to:

- Select master or slave mode.
- Enable the PrimeCell SSP peripheral

To configure the PrimeCell SSP as a master, clear the SSPCR1 register Master or Slave Selection (MS) bit to 0, which is the default value on reset.

Setting the SSPCR1 register MS bit to 1 configures the PrimeCell SSP as a slave.

When configured as a slave, enabling or disabling of the PrimeCell SSP SSPTXD signal is provided through the SSPCR1 slave mode SSPTXD output disable bit (SOD). This can be used in some multi-slave environments where master might parallel broadcast.

To enable the operation of the PrimeCell SSP, set the Synchronous Serial Port Enable (SSE) bit to 1.

Bit Rate Generation

The serial bit-rate is derived by dividing down the input clock SSPCLK. The clock is first divided by an even pre-scale value CPSDVSR from 2 to 254, which is programmed in SSPCPSR. The clock is further divided by a value from 1 to 256, which is 1 + SCR, where SCR is the value programmed in SSPCR0.

The frequency of the output signal bit clock SSPCLKOUT is defined below:

- $F_{SSPCLKOUT} = F_{SSPCLK}/(CPSDVR \times (1 + SCR))$

For example, if SSPCLK is 3.6864 MHz, and CPSDVSR = 2, then the SSPCLKOUT has a frequency range from 7.2 kHz to 1.8432 MHz

15.2.2.1.3.7 Frame Format

Each data frame is between 4 and 16 bits long depending on the size of the data programmed, and is transmitted starting with the Most Significant Bit (MSB).

For all three formats, the serial clock (SSPCLKOUT) is held inactive while the PrimeCell SSP is idle, and transitions at the programmed frequency only during active transmission or reception of data. The idle state of SSPCLKOUT is utilized to provide a receive timeout indication that occurs when the Receive FIFO still contains data after a timeout period.

For Motorola SPI, the serial frame (SSPFSSOUT) pin is active LOW, and is asserted (pulled down) during the entire transmission of the frame.

15.2.2.1.3.8 Motorola SPI Frame Format

The Motorola SPI interface is a four-wire interface where the SSPFSSOUT signal behaves as a slave-select. The main feature of the Motorola SPI format is that the inactive state and phase of the SSPCLKOUT signal are programmable through the SPO and SPH bits within the SSPSCR0 control register.

- SPO, Clock Polarity

When the SPO clock polarity control bit is LOW, it produces a steady state low value on the SSPCLKOUT pin. if the SPO clock polarity control bit is HIGH, a steady state high value is placed on the SSPCLKOUT pin when data is not being transferred.

- SPH, Clock Phase

The SPH control bit selects the clock edge that captures data and allows it to change state. It has the most impact on the first bit transmitted by either allowing or not allowing a clock transition before the first data capture edge.

When the SPH phase control bit is LOW, data is captured on the first clock edge transition. If the SPH clock phase control bit is HIGH, data is captured on the second clock edge transition.

15.2.2.1.3.9 Motorola SPI Format with SPO = 0, SPH = 0

Figure 15-2 and *Figure 15-3* illustrates the single and continuous transmission signal sequences for Motorola SPI format with SPO = 0 and SPH = 0 respectively.

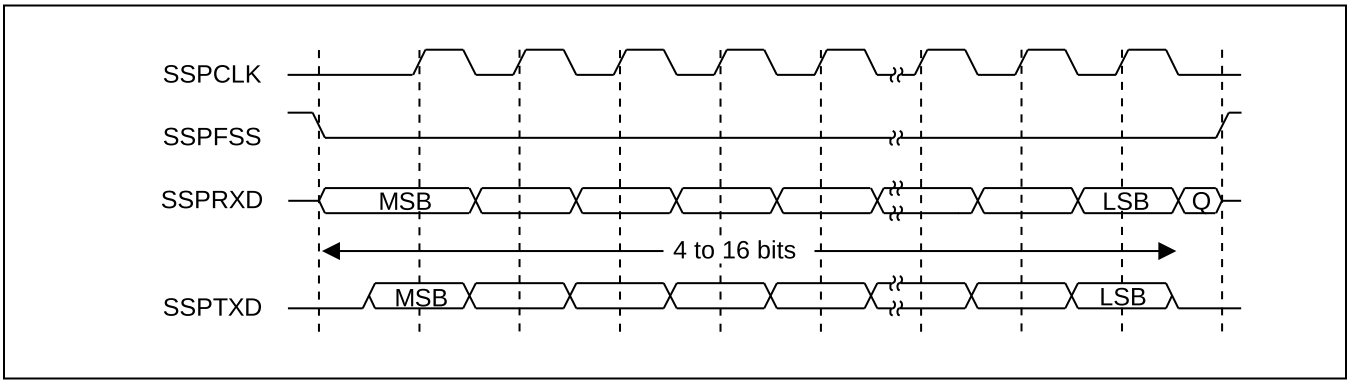


Figure 15-2 Motorola SPI Frame Format (Single Transfer) with SPO = 0 and SPH = 0

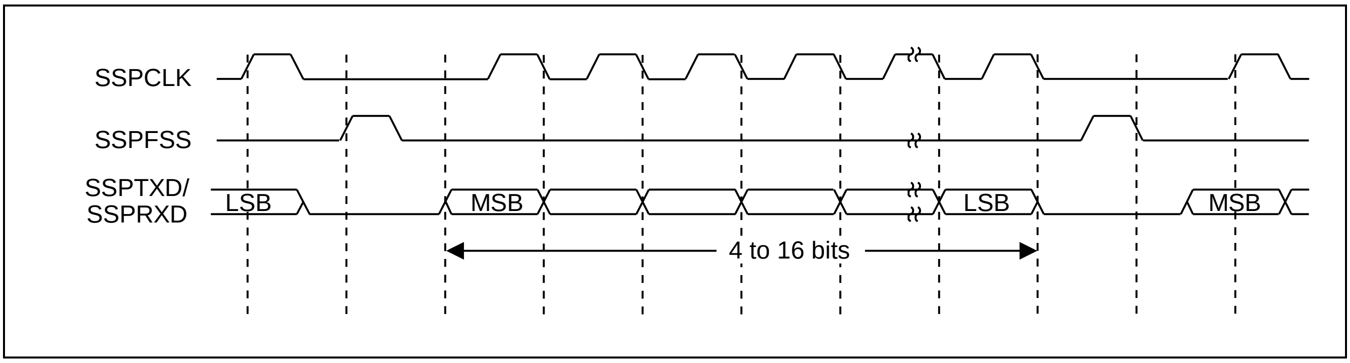


Figure 15-3 Motorola SPI Frame Format (Continuous Transfer) with SPO = 0 and SPH = 0

In this configuration, during idle periods:

- The SSPCLKOUT signal is forced LOW
- SSPFSSOUT is forced HIGH
- The transmit data line SSPTXD is arbitrarily forced LOW
- When the PrimeCell SSP is configured as a master, the SSPCLK pin is enabled
- When the PrimeCell SSP is configured as a slave, SSPCLK pin is disabled

If the PrimeCell SSP is enabled and there is valid data within the Transmit FIFO, the start of transmission is signified by the SSPFSSOUT master signal being driven LOW. This causes the slave data to be enabled onto the SSPRXD input line of the master.

One half SSPCLKOUT period later, valid master data is transferred to the SSPTXD pin. Now that both the master and slave data have been set, the SSPCLKOUT master clock pin goes HIGH after one further half SSPCLKOUT period.

The data is now captured on the rising and propagated on the falling edges of the SSPCLKOUT signal.

In the case of a single word transmission, after all bits of the data-word have been transferred, the SSPFSSOUT line is returned to its idle HIGH state one SSPCLKOUT period after the last bit has been captured.

However, in case of continuous back-to-back transmissions, the SSPFSSOUT signal should be pulsed HIGH between each data word transfer. This is because the slave select pin freezes the data in its serial peripheral register and does not allow it to be altered if the SPH bit is logic zero. Therefore the master device must raise the SSPFSSIN pin of the slave device between each data transfer to enable the serial peripheral data write. On completion of the continuous transfer, the SSPFSSOUT pin is returned to its idle state one SSPCLKOUT period after the last bit has been captured.

15.2.2.1.3.10 Motorola SPI Format with SPO = 0, SPH = 1

Figure 15-4 illustrates the transfer signal sequence for Motorola SPI format with SPO = 0, SPH = 1.

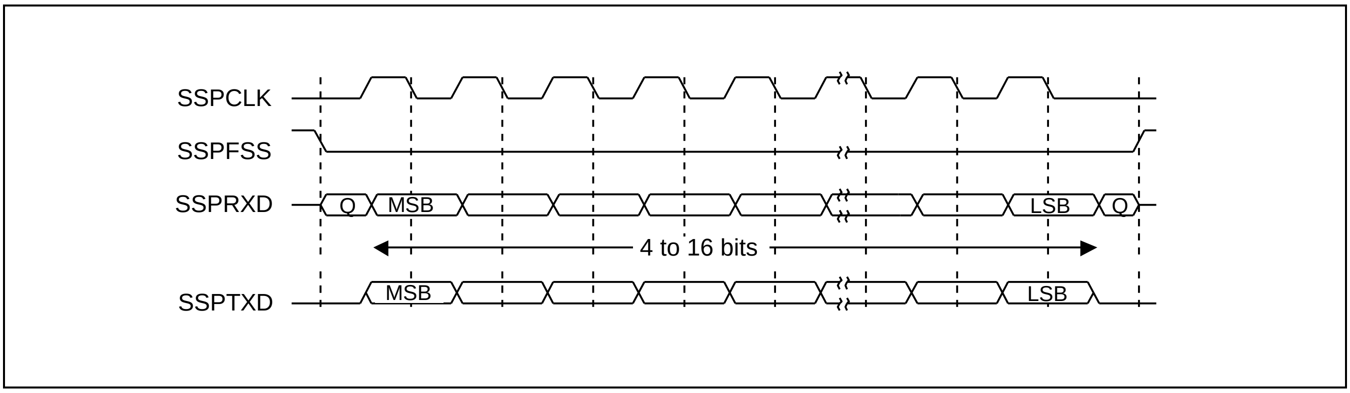


Figure 15-4 Motorola SPI Frame Format with SPO = 0 and SPH = 1

In this configuration, during idle periods:

- The SSPCLKOUT signal is forced LOW
- SSPFSSOUT is forced HIGH
- The transmit data line SSPTXD is arbitrarily forced LOW
- When the PrimeCell SSP is configured as a master, the SSPCLKOUT is enabled
- When the PrimeCell SSP is configured as a slave, the SSPCLKOUT is disabled

If the PrimeCell SSP is enabled and there is valid data within the Transmit FIFO, the start of transmission is signified by the SSPFSSOUT master signal being driven LOW. The master SSPTXD output pas is enabled. After a further one half of the SSPCLKOUT period, both master and slave valid data is enabled onto their respective transmission lines. At the same time the SSPCLKOUT is enabled with a rising edge transition.

Data is then captured on the falling edges and propagated on the rising edges of the SSPCLKOUT signal.

In the case of a single word transfer, after all bits have been transferred, the SSPFSSOUT line is returned to its idle HIGH state one SSPCLKOUT period after the last bit has been captured.

For continuous back-to-back transfers, the SSPFSSOUT pin is held LOW between successive data words and termination is the same as that of the single word transfer.

15.2.2.1.3.11 Motorola SPI Format with SPO = 1, SPH = 0

Figure 15-5 and *Figure 15-6* illustrates the single and continuous transmission signal sequences for Motorola SPI format with SPO = 1, SPH = 0 respectively:

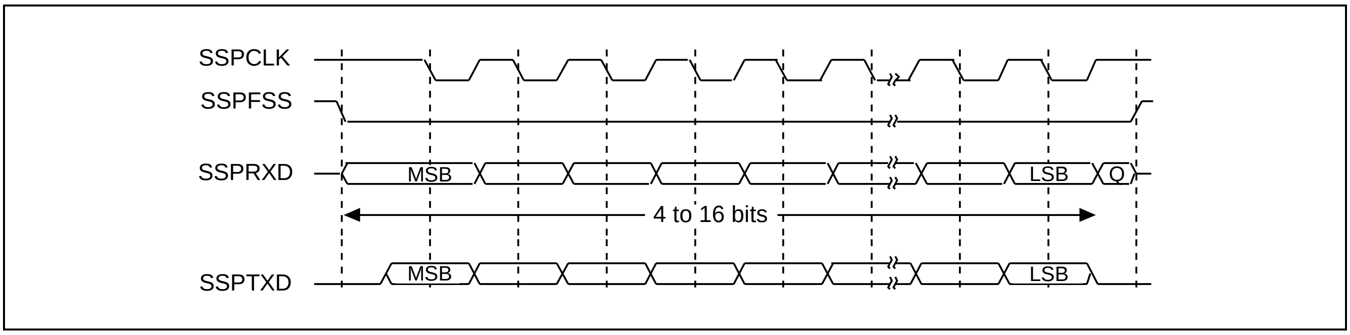


Figure 15-5 Motorola SPI Frame Format (Single Transfer) with SPO = 1 and SPH = 0

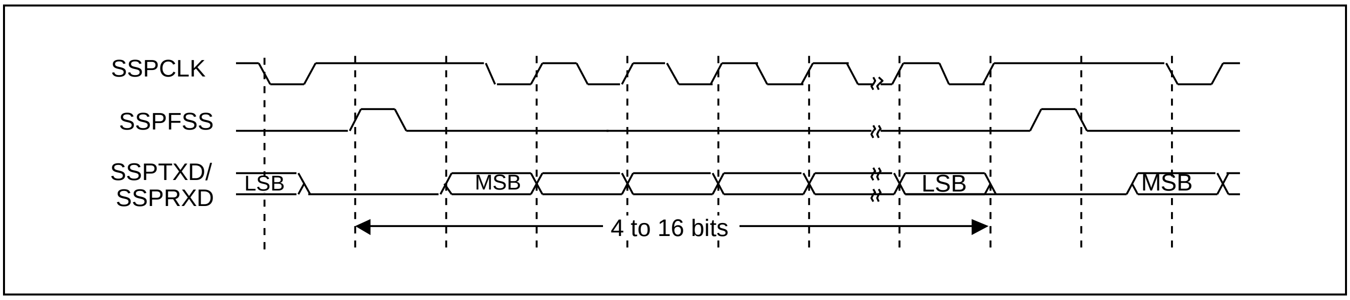


Figure 15-6 Motorola SPI Frame Format (Continuous Transfer) with SPO = 1 and SPH = 0

In this configuration, during the idle periods:

- The SSPCLKOUT signal is forced HIGH
- SSPFSSOUT is forced HIGH
- The transmit data line SSPTXD is arbitrarily forced LOW
- When the PrimeCell SSP is configured as a master, the SSPCLKOUT is enabled
- When the PrimeCell SSP is configured as a slave, the SSPCLKOUT is disabled

If the PrimeCell SSP is enabled and there is valid data within the Transmit FIFO, the start of transmission is signified by the SSPFSSOUT master signal being driven LOW which causes slave data to be immediately transferred onto the SSPRXD line of the master. The master SSPTXD output pin is enabled.

One half period later, valid master data is transferred to the SSPTXD line. Now that both the master and slave data have been set, the SSPCLKOUT master clock pin becomes LOW after one further half SSPCLKOUT period. This means that data is captured on the falling edges and be propagated on the rising edges of the SSPCLKOUT signal.

In the case of a single word transmission, after all bits of the data word are transferred, the SSPFSSOUT line is returned to its idle HIGH state one SSPCLKOUT period after the last bit has been captured.

However, in case of continuous back-to-back transmissions, the SSPFSSOUT signal must be pulsed HIGH between each data word transfer. This is because the slave select pin freezes the data in its serial peripheral register and does not allow it to be altered if the SPH bit is logic zero. Therefore, the master device must raise the SSPFSSIN pin of the slave device between each data transfer to enable the serial peripheral data write. On completion of the continuous transfer, the SSPFSSOUT pin is returned to its idle state one SSPCLKOUT period after the last bit has been captured.

15.2.2.1.3.12 Motorola SPI Format with SPO = 1, SPH = 1

Figure 15-7 illustrates the transfer signal sequence for Motorola SPI format with SPO = 1, SPH = 1 for both single and continuous transfers.

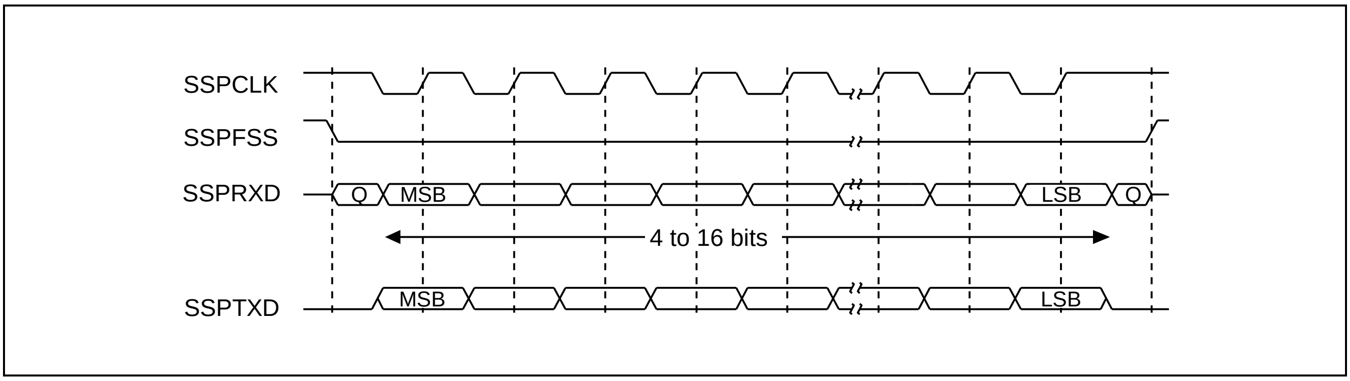


Figure 15-7 Motorola SPI Frame Format with SPO = 1 and SPH = 1

In this configuration, during idle periods:

- The SSPCLK signal is forced HIGH
- SSPFSS is forced HIGH
- The transmit data line SSPTXD is arbitrarily forced LOW
- When the PrimeCell SSP is configured as a master, the SSPCLKOUT is enabled
- When the PrimeCell SSP is configured as a slave, the SSPCLKOUT is disabled

If the PrimeCell SSP is enabled and there is valid data within the Transmit FIFO, the start of transmission is signified by the SSPFSSOUT master signal being driven LOW. The master SSPTXD output pin is enabled.

After a further one half SSPCLKOUT period, both master and slave data are enabled onto their respective transmission lines. At the same time, the SSPCLK is enabled with a falling edge transition. Data is then captured on the rising edges and propagated on the falling edges of the SSPCLKOUT signal.

After all bits have been transferred, in the case of a single word transmission, the SSPFSSOUT line is returned to its idle HIGH state one SSPCLKOUT period after the last bit has been captured.

For continuous back-to-back transfers, the SSPFSSOUT pin is held LOW between successive data words and termination is the same as that of the single word transfer.

15.2.2.1.3.13 Examples of Master and Slave Configurations

PrimeCell SSP (PL022) peripheral can be connected to other synchronous serial peripherals when it is configured as a master or a slave.

NOTE: The SSP (PL022) does not support dynamic switching between master and slave in a system. Each instance is configured and connected either as a master or as a slave.

Figure 15-8 and *Figure 15-9* illustrates PrimeCell SSP master coupled to two slaves and SPI master coupled to two PrimeCell SSP slaves respectively.

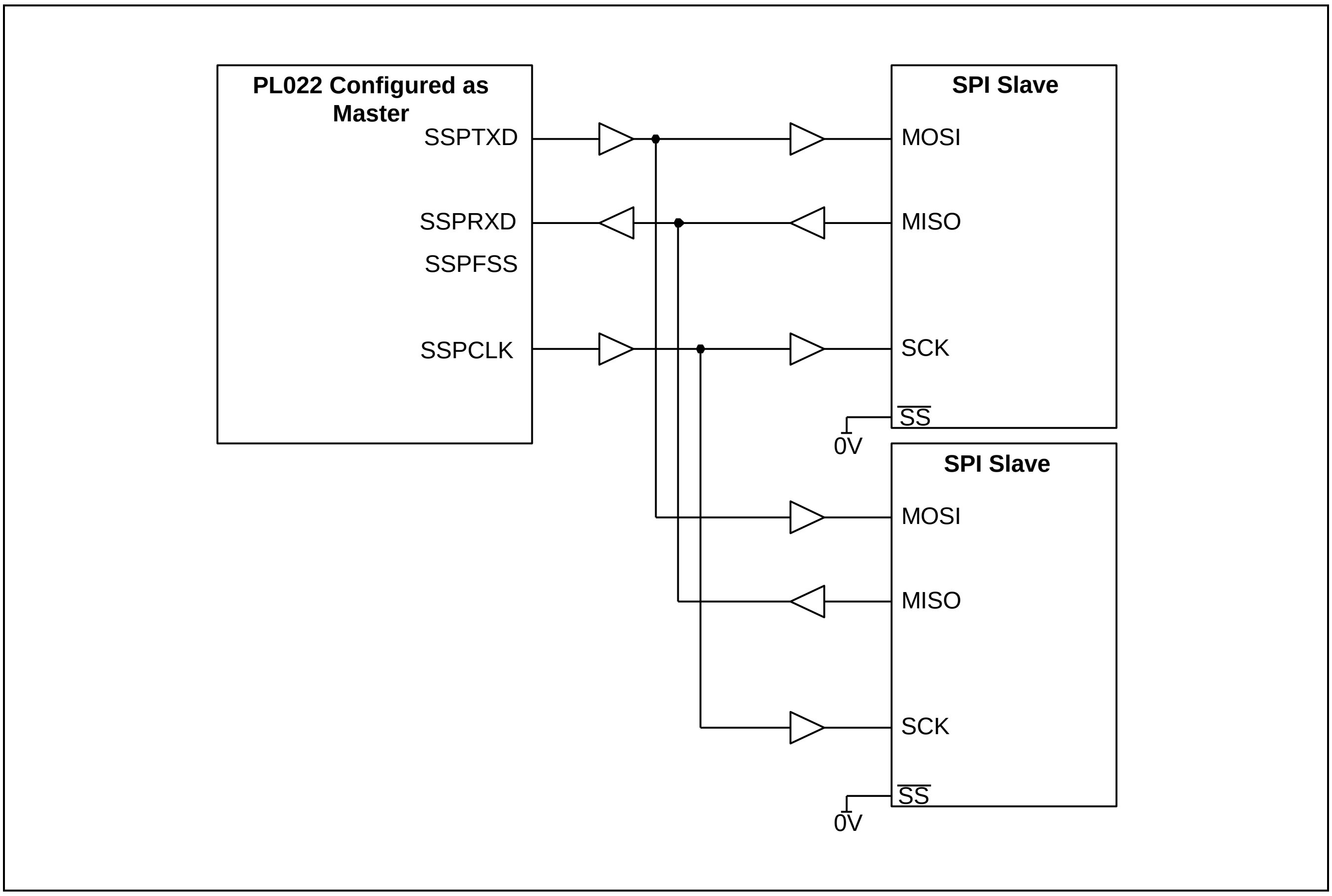


Figure 15-8 PrimeCell SSP Master Coupled to Two Slaves

The PrimeCell SSP (PL022) is configured as master, and is interfaced to two Motorola SPI slaves. Each SPI Slave Select (SS) signal is permanently tied LOW and configures them as slaves. Similar to the above operation, the master can broadcast to two slaves through the master PrimeCell SSP SSPTXD line. In response, only one slave drives its SPI MISO port onto the SSPRXD line of the master.

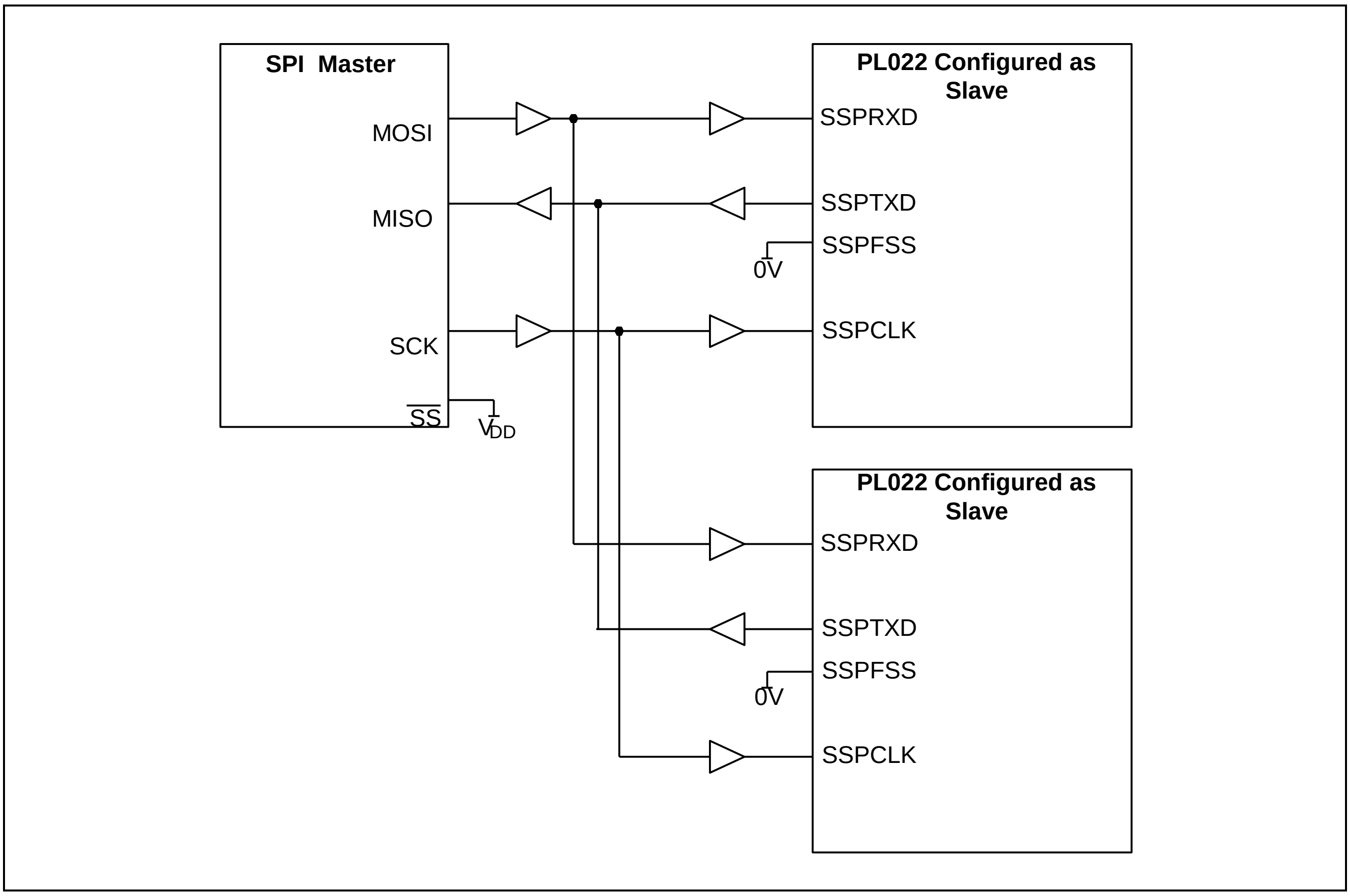


Figure 15-9 SPI Master Coupled to two PrimeCell SSP Slaves

The Motorola SPI is configured as a master and interfaced to two instances of PrimeCell SSP (PL022) configured as slaves. In this case the slave Select Signal (SS) is permanently HIGH and configures it as a master. The master can broadcast to the two slaves through the master SPI MOSI line and in response, only one slave drives its SSPTXD data onto the MISO line of the master.

15.2.2.1.4 Interrupt

There are five interrupts generated by the PrimeCell SSP. Four of these are individual, maskable, active HIGH interrupts:

- SSPRXINTR: PrimeCell SSP Receive FIFO service interrupt request
- SSPTXINTR: PrimeCell SSP Transmit FIFO service interrupt request
- SSPRORINTR: PrimeCell SSP receive overrun interrupt request
- SSPRTINTR: PrimeCell SSP time out interrupt request

The four individual maskable interrupts by setting the appropriate bits in the SSPIMSC register. Setting the appropriate mask bit HIGH enables the interrupt.

Provision of the individual outputs as well as a combined interrupt output, allows use of either a global interrupt service routine, or a modular device drivers to handle interrupts.

The transmit and receive dynamic data flow interrupts SSPTXINTR and SSPRXINTR have been separated from the status interrupts, so that data can be read or written in response to just the FIFO trigger levels.

The status of the individual interrupt sources can be read from SSPRIS and SSPMIS registers.

- SSPRXINTR
  - The receive interrupt is asserted when there are four or more valid entries in the Receive FIFO.

- SSPTXINTR
  - The transmit interrupt is asserted when there are four or less valid entries in the Transmit FIFO. The transmit interrupt SSPTXINTR is not qualified with the PrimeCell SSP enable signal, which allows operation in one of two ways. Data can be written to the Transmit FIFO prior to enabling the PrimeCell SSP and the interrupts. Alternatively, the PrimeCell SSP and interrupts can be enabled so that data can be written to Transmit FIFO by an interrupt service routine.

- SSPRORINTR
  - The receive overrun interrupt SSPORINTR is asserted when the FIFO is already full and an additional data frame is received, causing an overrun of the FIFO. Data is over-written in the receive shift register, but not the FIFO.

- SSPRTINTR
  - The receive timeout interrupt is asserted when the Receive FIFO is not empty and the PrimeCell SSP has remained idle for a fixed 32-bit period. This mechanism ensures that the user is aware that the data is still present in the Receive FIFO and requires servicing. This interrupt is de-asserted if the Receive FIFO becomes empty by subsequent reads, or if new data is received on SSPRXD. It can also be cleared by writing to the RTIC bit in the SSPICR register.

15.3 Register Description

15.3.1 Register Map Summary

- Base Address: 0x4009_0000

Register	Offset	Description	Reset Value
SSP_CR0	0x0000	SSP control register 0	0x0000_0000
SSP_CR1	0x0004	SSP control register 1	0x0000_0010
SSP_DR	0x0008	SSP receive FIFO data register (read) SSP transmit FIFO data register (Write)	0x0000_0000
SSP_SR	0x000C	SSP status register	0x0000_0003
SSP_CPSR	0x0010	SSP clock rescale register	0x0000_0000
SSP_IMSCR	0x0014	SSP interrupt mask set/clear register	0x0000_0000
SSP_RISR	0x0018	SSP raw interrupt status register	0x0000_0008
SSP_MISR	0x001C	SSP masked interrupt status register	0x0000_0000
SSP_ICR	0x0020	SSP interrupt clear register	0x0000_0000

15.3.1.1 SSP_CR0

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x0000, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																SCR								SPH	SPO	FRF		DSS			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
SCR	[15:8]	RW	Serial Clock Rate Field Use the SCR value to generate the transmit and receive bit-rate. The Bit Rate = FPCLK/(CPSDVR × (1 + SCR)) The CPSDVSR is an even value from 2 to 254, you can program this through the SSP_CPSR register and SCR is a value from 0 to 255.	0x00
SPH	[7]	RW	SSPCLKOUT Phase 0 = Captures data on the first clock edge transition 1 = Captures data on the second clock edge transition	0'b
SPO	[6]	RW	SSPCLK Polarity Bit 0 = Captures data on the first clock edge transition 1 = Captures data on the second clock edge transition	0'b
FRF	[5:4]	RW	Frame Format Selection Field Should be set to 00 for Motorola SPI frame format.	00'b
DSS	[3:0]	RW	Data Size Selection Field 0000 to 0010 = Reserved 0011 = 4-bit data 0100 = 5-bit data 0101 = 6-bit data 0110 = 7-bit data 0111 = 8-bit data 1000 = 9-bit data 1001 = 10-bit data 1010 = 11-bit data 1011 = 12-bit data 1100 = 13-bit data 1101 = 14-bit data 1110 = 15-bit data 1111 = 16-bit data	0000'b

15.3.1.2 SSP_CR1

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																									RXIFLSEL			SOD	MS	SSE	LBM
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	R	Reserved	0
RXIFLSEL	[6:4]	RW	Receive Interrupt FIFO Level Selection Field 001 = Trigger points, Receive FIFO becomes ≥ 1/8 010 = Trigger points, Receive FIFO becomes ≥ 1/4 100 = Trigger points, Receive FIFO becomes ≥ 1/2 Others = Reserved	0001'b
SOD	[3]	RW	Slave-mode Output Disable Bit 0 = SSP can drive the SSPTXD output in slave mode. 1 = SSP should not drive the SSPTXD output in slave mode.	0'b
MS	[2]	RW	Master or Slave Mode Selection Bit 0 = Configures device as master 1 = Configures device as slave	0'b
SSE	[1]	RW	Synchronous Serial Port Enable Bit 0 = Disables SSP operation 1 = Enables SSP operation	0'b
LBM	[0]	RW	Loop-Back Mode Bit 0 = Enables normal serial port operation 1 = Output of transmit serial shifter is connected to input of receive serial shifter internally.	0'b

NOTE: SOD bit is relevant only in the slave mode (MS = 1). In multiple-slave systems, it is possible for a PrimeCell SSP master to broadcast a message to all slaves in the system while ensuring that only one slave drives data onto its serial output line. In such systems the RXD lines from multiple slaves could be tied together. To operate in such systems, the SOD bit can be set if the PrimeCell SSP slave is not supposed to drive the SSPTXD line.

15.3.1.3 SSP_DR

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
DATA	[15:0]	RW	Transmit/Receive FIFO Read: Receives FIFO. Write: Transmits FIFO. The user should right-justify the data when the PrimeCell SSP is programmed for a data size that is less than 16-bits. Unused bits at the top are ignored by the transmit logic. The receive logic will automatically right-justify.	0x0000

When SSPDR is read, the entry in the Receive FIFO that is pointed to by the current FIFO read pointer is accessible. As data values are removed by the PrimeCell SSP receive logic from the incoming data frame they are placed into the entry in the Receive FIFO that is pointed to by the current FIFO write pointer.

When SSPDR is written to, the entry in the Transmit FIFO that is pointed to by the write pointer is accessible. The data values are removed from the Transmit FIFO one value at a time by the transmit logic. It is loaded to the transmit serial shifter, serially shifted out onto the SSPTXD pin at the programmed bit rate.

When a data size of less than 16-bit is selected, the user must right-justify the data written to the Transmit FIFO. The transmit logic ignores the unused bits. Received data that are less than 16-bit is automatically right-justified in the receive buffer.

15.3.1.4 SSP_SR

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																											BSY	RFF	RNE	TNF	TFE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	R	Reserved	0
BSY	[4]	R	Prime-Cell SSP Busy Flag Bit 0 = SSP is idle 1 = SSP is currently transmitting And/or receiving a frame or the transmit FIFO is not empty.	0'b
RFF	[3]	R	Receive FIFO Full Status Bit 0 = Receive FIFO is not full. 1 = Receive is full	0'b
RNE	[2]	R	Receive Empty Status Bit 0 = Receive FIFO is empty 1 = Receive FIFO is not empty	0'b
TNF	[1]	R	Transmit FIFO Full Status Bit 0 = Transmit FIFO is full 1 = Transmit FIFO is not full	1'b
TFE	[0]	R	Transmit FIFO Empty Status Bit 0 = Transmit FIFO is not empty 1 = Transmit FIFO is empty	1'b

15.3.1.5 SSP_CPSR

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	R	Reserved	0
CPSDVSR	[7:0]	RW	Clock Pre-scale Divisor Field Should be an even number from 2 to 254, depending on the frequency of FSSPCLK. The least significant bit always returns zero on reads.	0x00

SSPCPSR is the clock pre-scale register and specifies the division factor by which the input F_{PCLK} should be internally divided before any use.

The value programmed into this register should be an even number between 2 to 254. The least significant bit of the programmed number is hard-coded to zero. If an odd number is written to this register, the data can be read back from this register that has zero as its least significant bit.

15.3.1.6 SSP_IMSCR

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																												TXIM	RXIM	RTIM	RORIM
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
TXIM	[3]	RW	Transmit FIFO Interrupt Mask 0 = Tx FIFO half full or less condition interrupt is masked. (Disables the interrupt) 1 = Tx FIFO half full or less condition interrupt is not masked. (Enables the interrupt.)	0'b
RXIM	[2]	RW	Receive FIFO Interrupt Mask 0 = Rx FIFO trigger level (1/2, 1/4, or 1/8) condition interrupt is masked. (Disables the interrupt) 1 = Rx FIFO trigger level (1/2, 1/4, or 1/8) condition interrupt is not masked. (Enables the interrupt).	0'b
RTIM	[1]	RW	Receive Timeout Interrupt Mask 0 = RxFIFO not empty and no read prior to timeout period interrupt is masked. (Disables the interrupt) 1 = RxFIFO not empty and no read prior to timeout period interrupt is not masked. (Enables the interrupt.)	0'b
RORIM	[0]	RW	Receive Overrun Interrupt Mask 0 = RxFIFO written to while full condition interrupt is masked. (Disables the interrupt) 1 = RxFIFO written to while full condition interrupt is not masked. (Enables the interrupt)	0'b

NOTE: On a Read, SSP_IMSCR register gives the current value of the mask on the relevant interrupt. A Write of 1 to a particular bit, sets the mask, thus enabling the interrupt to be read. A Write of 0 clears the corresponding mask.

15.3.1.7 SSP_RISR

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																												TXRIS	RXRIS	RTRIS	RORRIS
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
TXRIS	[3]	R	Transmit FIFO Raw Interrupt State Gives the raw interrupt state (prior to masking) of the SSPTXINTR interrupt.	1'b
RXRIS	[2]	R	Receive FIFO Raw Interrupt State Gives the raw interrupt state (prior to masking) of the SSPRXINTR interrupt.	0'b
RTRIS	[1]	R	Receive Timeout Raw Interrupt State Gives the raw interrupt state (prior to masking) of the SSPRTINTR interrupt.	0'b
RORRIS	[0]	R	Receive Overrun Raw Interrupt State Gives the raw interrupt state (prior to masking) of the SSPRORINTR interrupt.	0'b

NOTE: On a Read, SSP_RISR register gives the current raw status value of the corresponding interrupt prior to masking. A Write has no effect.

15.3.1.8 SSP_MISR

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																												TXMIS	RXMIS	RTMIS	RORMIS
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
TXMIS	[3]	R	Transmit FIFO Masked Interrupt State Gives the transmit FIFO masked interrupt state (after masking) of the SSPTXINTR interrupt.	0'b
RXMIS	[2]	R	Receive FIFO Masked Interrupt State Gives the receive FIFO masked interrupt state (after masking) of the SSPRXINTR interrupt.	0'b
RTMIS	[1]	R	Receive Timeout Masked Interrupt State Gives the receive timeout masked interrupt state (after masking) of the SSPRTINTR interrupt.	0'b
ROMIS	[0]	R	Receive Overrun Masked Interrupt State Gives the receive overrun masked interrupt status (after masking) of the SSPRORINTR interrupt.	0'b

NOTE: On a Read, SSP_MISR register gives the current masked status value of the corresponding interrupt.
A Write has no effect.

15.3.1.9 SSP_ICR

- Base Address: 0x4009_0000, 0x4009_1000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																														RTIC	RORIC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
RTIC	[1]	W	Receive Timeout Interrupt Clear 0 = No effect 1 = Clears the SSPRTINTR interrupt	0'b
RORIC	[0]	W	Receive Overrun Interrupt Clear 0 = No effect 1 = Clears the SSPRORINTR interrupt	0'b

On a Write of 1, it clears the corresponding interrupt. A Write of 0 has no effect.

16 Timer/Counter

16.1 Overview

The Timer/Counter (TC) chapter describes TIMER/COUNTER module that operates in match and overflow, capture, interval or in PWM operation. The TC can also generate PWM signals through the dedicated pin and supports an external clock as its source clock.

16.1.1 Features

The features of TC are:

- Programmable clock source for timer, including an external clock
- Programmable n-bit up counter with up to 16-bit
- One-shot operation or Repeated operation
- Match and Overflow operation
- Capture operation
  - Capture on rising edge, falling edge, or both edges
  - Two capture registers for each edge
- Interval operation
- Pulse Width Modulation (PWM) operation
  - Programmable duty cycle and frequency
  - Programmable active level and idle level
  - Up to 22-bit resolution including extension function
- Debug option
- Analog to Digital Converter (ADC) trigger source

16.1.2 Pin Description

Table 16-1 describes the pin description of TC.

Table 16-1 Pin Description

Pin Name	Function	I/O Type	Comments
TCLKn	External clock input pin	I	–
TCAPn	Capture pin	I	–
TPWMn	PWM output pin	O	–

NOTE: "n" means the channel number of TC. For example, they will be TCLK0, TCAP0, and TPWM0, where "n" is 0.

16.2 Functional Description

The TC supports up to 16-bit counter increased by PCLK or the external clock input through TCLK pin. The TC operates as Overflow mode or Period mode to generate periodical time event. TC can output the PWM signal through TPWM pin. You can capture the counter value by an external trigger signal on TCAP pin.

16.2.1 Block Diagram

Figure 16-1 illustrates the block diagram for TC.

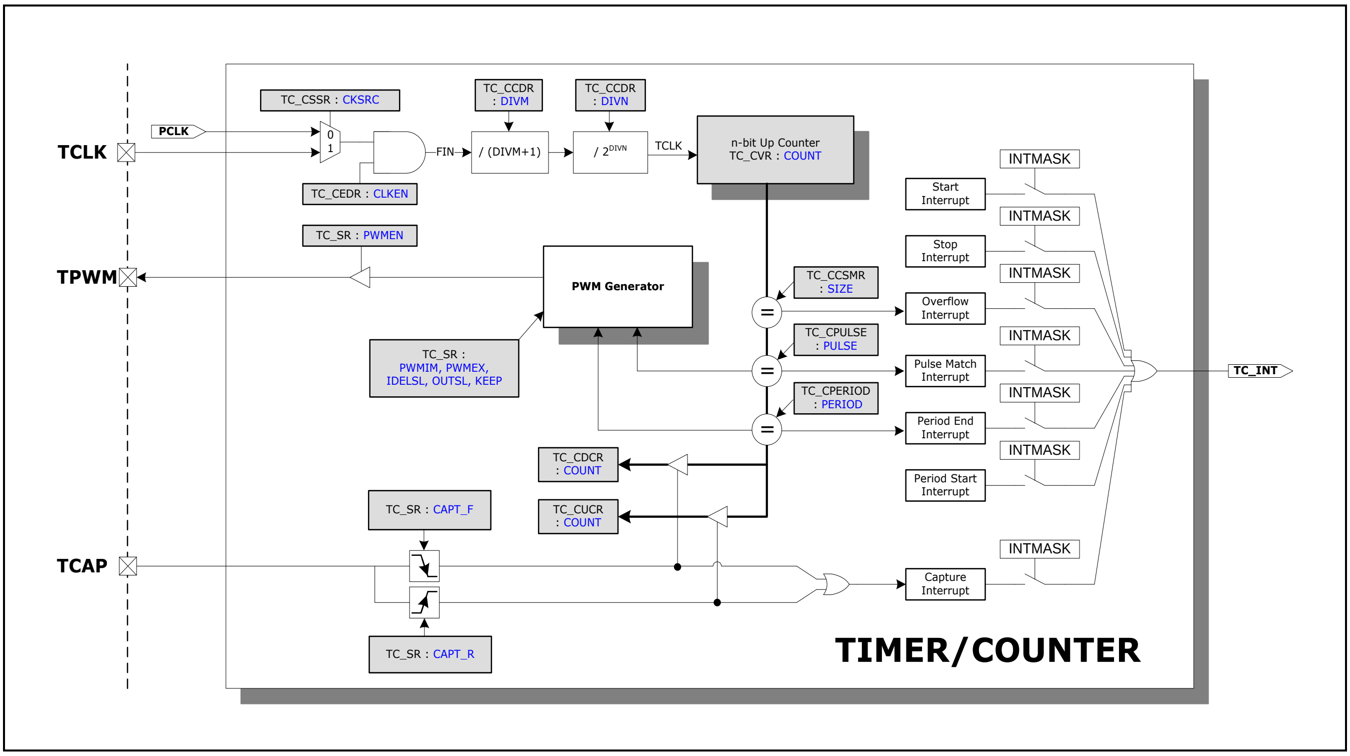


Figure 16-1 TC Block Diagram

NOTE: You can assert the external clock source on the TCLK Pin. The external clock frequency (TCLK) should be equal to or lesser than the internal clock frequency (PCLK).

16.2.2 Counter Size

You can use TC_CSMR (Counter Size Mask Register) and TC_CCSMR (Current Counter Size Mask Register) to determine the timer count size. Both the registers include SIZE[3:0] field. If the value of SIZE[3:0] is n, then the TC will become a (n + 1) bit timer. In other words, this timer can count from 1 to $2^{(n + 1)} – 1$.

When the TC starts or UPDATE bit is set in the TC_CSR (Control Set Register), TC_CSMR is copied to the TC_CCSMR register. SIZE field in TC_CCSMR register shows the information for the counter bit size of a current operating timer. During the operation, you can prepare the new counter size with SIZE field in TC_CSMR register.

16.2.3 Counter Clock

The counter clock increases the counter value. The clock source and the clock divisor determine the frequency of the counter clock.

16.2.3.1 Clock Source

TC can use either the internal clock or the external clock as the counter clock source. TC can use either the internal clock or the external clock as the counter clock source. The internal clock is the PCLK and the external clock is a clock asserted on the TCLK pin To use the external clock supplied by TCLK, you should configure the pin properly before starting the timer. In case of using an external clock source, the external clock frequency (TCLK) should be less than the internal clock frequency (PCLK). You can specify which clock is used as the clock source by setting CLKSRC in TC_CSSR register.

16.2.3.2 Counter Clock

You can determine the frequency of the counter clock (TCCLK) using the frequency of the clock source (FIN) and the internal clock dividers, DIVM[10:0] and DIVN[3:0] in TC_CCDR register.

- TCCLK = $(FIN/2^{DIVN})/(DIVM + 1)$
  You can determine the counter resolution using the TCCLK value:
- Counter Resolution = 1/TCCLK
  Since TC_CCDR is read only, you should modify the DIVM and DIVN in TC_CDR (Clock Divider Register). When the timer starts or UPDATE bit is set in the TC_CSR register, the TC copies the DIVM and DIVN in the TC_CDR register into the TC_CCDR register.

Caution: Do not set DIVM to zero when DIVN is not zero.
For example,
If the counter clock is 4 four times slower than the clock source, then the allowed settings are:
- DIVN = 0 and DIVM = 3
- DIVN = 1 and DIVM = 1
But following is forbidden:
- DIVN = 2 and DIVM = 0

16.2.4 Debug Option

When the debugger halts the CPU, DBGEN bit in the TC_CEDR register determines whether the TC freezes the counter or not. You can set DBGEN bit to "1" in order to easily check and verify the states of the TC for debug purpose under the development.

16.2.5 ADC Trigger Source

You can use TC as a trigger source of ADC conversion when ADTRIG bit is set in the TC_SR register. You can start the ADC conversion by using Pulse Match event.

16.2.6 Overflow Mode

When OVFM bit is set in the TC_SR register, the timer operates as Overflow Mode. The TC increases the counter value in this mode until it reaches $2^{(SIZE + 1)} – 1$, where "SIZE" is specified in TC_CCSMR (Current Counter Size Mask Register). After that, if REPEAT bit is set to clear in TC_SR register, then the counter value will be cleared to "0" and generates Stop interrupt and Overflow interrupt. You should set PERIOD to greater than "0" in the TC_PRDR register before starting the timer even though PERIOD is not used.

16.2.6.1 Match and Overflow Operation

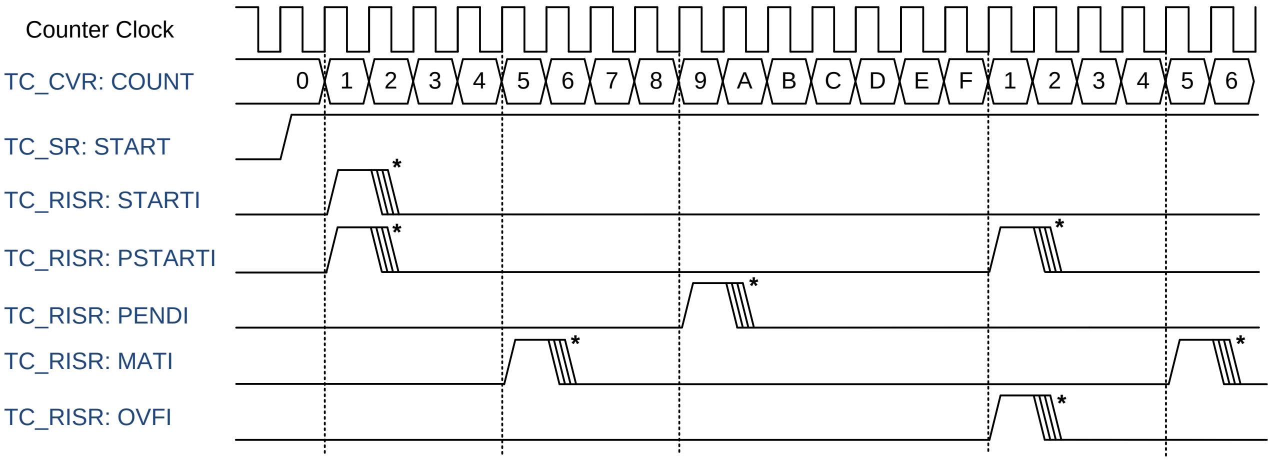
In the match and overflow operation, the TC generates:

- Start interrupt and Period Start interrupt when the timer starts.
- Pulse Match interrupt when the counter value is identical to PULSE bits in TC_CPULR register.
- Period End interrupt when the counter value is identical to PERIOD bits in TC_CPRDR register.
- Overflow interrupt when the counter overflows.

If REPEAT bit in TC_SR register is set on overflow, then the counter value will be changed to "1" and the timer restarts automatically.

Figure 16-2 illustrates the match and overflow operation timing.

Conditions :
TC_SR: OVFM = 1, REPEAT = 1, TC_CCSMR: SIZE = 3, TC_CPULR: PULSE = 4, TC_CPRDR: PERIOD = 8



* : Interrupts are cleared by writing a 1 to corresponding bits in TC_ICR

Figure 16-2 Match and Overflow Operation Timing

To stop the timer, you should clear the START bit in TC_SR register by writing 1 to START bit in TC_CCR register. The timer stops according to the STOPHOLD bit and STOPCLEAR bit in TC_SR register.

Figure 16-3 illustrates the counter values according to START, STOPCLEAR and STOPHOLD.

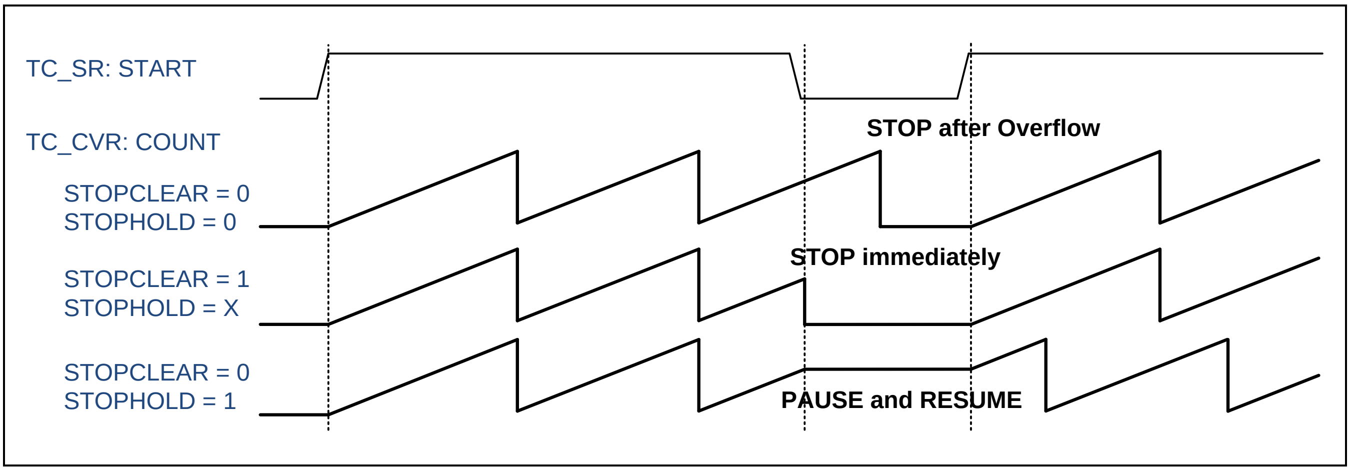


Figure 16-3 Counter Values According to START, STOPCLEAR and STOPHOLD

To stop the timer right after the counter overflows, you should clear both STOPHOLD bit and STOPCLEAR bit in TC_SR register before clearing START bit. The counter value is cleared to "0".

To stop the timer immediately, you should set the STOPCLEAR bit in TC_SR register before clearing START bit. The timer is stopped immediately and the counter value is cleared to "0".

To pause the timer immediately, you should set the STOPHOLD bit but you should clear the STOPCLEAR bit in TC_SR register before clearing START bit. The timer is stopped immediately but the timer holds the counter value. Therefore when the timer is resumed by setting START bit again, the counter value will be increased continuously from the last value it has kept.

16.2.6.2 Capture Operation

The TC can perform the capturing operation wherein the counter value is transferred into the capture registers, namely, the TC_CUCR (Capture Up Counter Register) and TC_CDCR (Capture Down Counter Register), in synchronization with an external trigger signal. After the completing the capture operation, you can determine the time difference between the external events. The external triggering signal for the capturing operation is a pre-defined valid edge on the capture input pin (TCAP). If CAPT_F bit is set in TC_SR register, then the counter value in process is copied into TC_CDCR register when a falling edge signal is detected on TCAP pin. If CAPT_R bit is set in TC_SR register, then the counter value in process is copied into TC_CUCR register when a rising edge signal is detected on TCAP pin. You should keep the external trigger signal on TCAP pin at least three times longer than PCLK to distinguish from glitch signals. When either CAPT_R or CAPT_F is set, the capture function always operates regardless of whether the timer is running or not.

Figure 16-4 illustrates the capture operation timing.

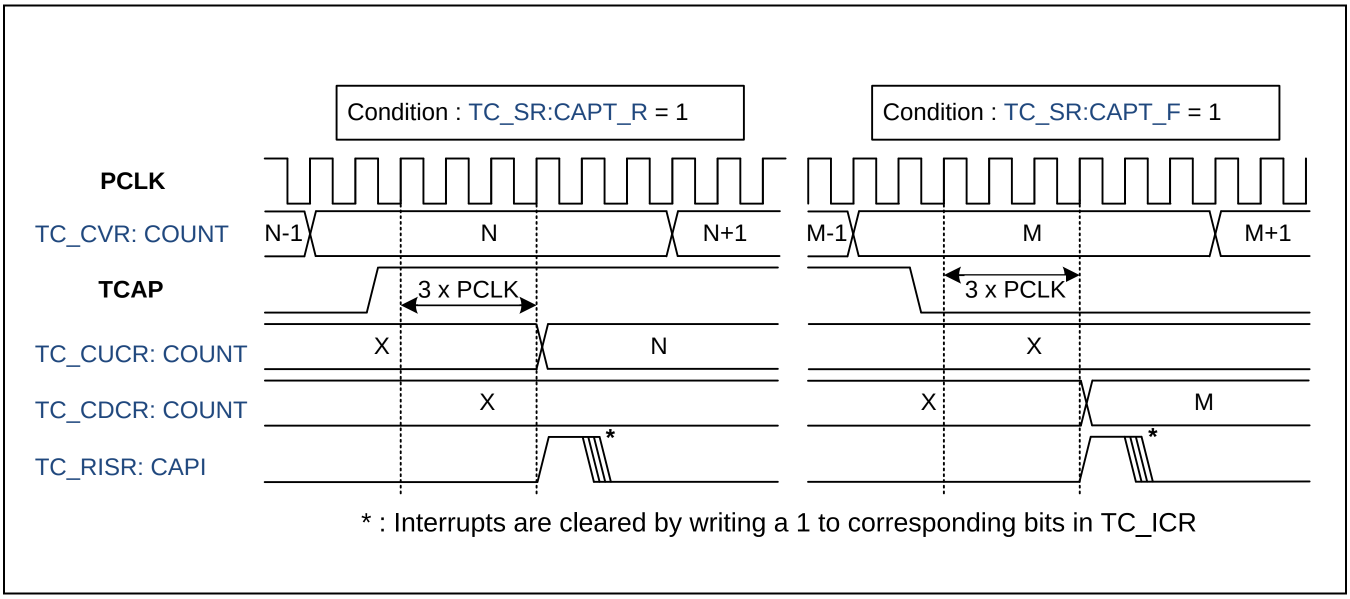

Condition : TC_SR:CAPT_R = 1
Condition : TC_SR:CAPT_F = 1
PCLK
TC_CVR: COUNT
N-1
N
N+1
M-1
M
M+1
TCAP
3 x PCLK
3 x PCLK
TC_CUCR: COUNT
X
N
X
TC_CDCR: COUNT
X
X
M
TC_RISR: CAPI
*
*
* : Interrupts are cleared by writing a 1 to corresponding bits in TC_ICR

Figure 16-4 Capture Operation Timing

Caution: The TC supports the capture operation only when the clock source is PCLK.

16.2.7 Period Mode

The TC operates in Period Mode when OVFM bit in TC_SR register is clear. In this mode, the counter value increases from 1 to PERIOD bits in TC_CPRDR register. When the counter value reaches PERIOD value, Period End event is generated. If REPEAT bit in TC_SR register is set, then the counter value restarts from 1. If REPEAT bit is clear, then the timer stops and the counter value is cleared to "0".

During operation, TC_CPRDR and TC_CPULR represent the current configured values. When the timer starts or UPDATE bit in TC_CSR register is set, the TC changes the values of PERIOD in TC_CPRDR and PULSE in TC_CPULR to new values specified by TC_PRDR and TC_PULR register respectively. You should set PERIOD to greater than "1" before starting the timer.

Figure 16-5 illustrates the period mode timing.

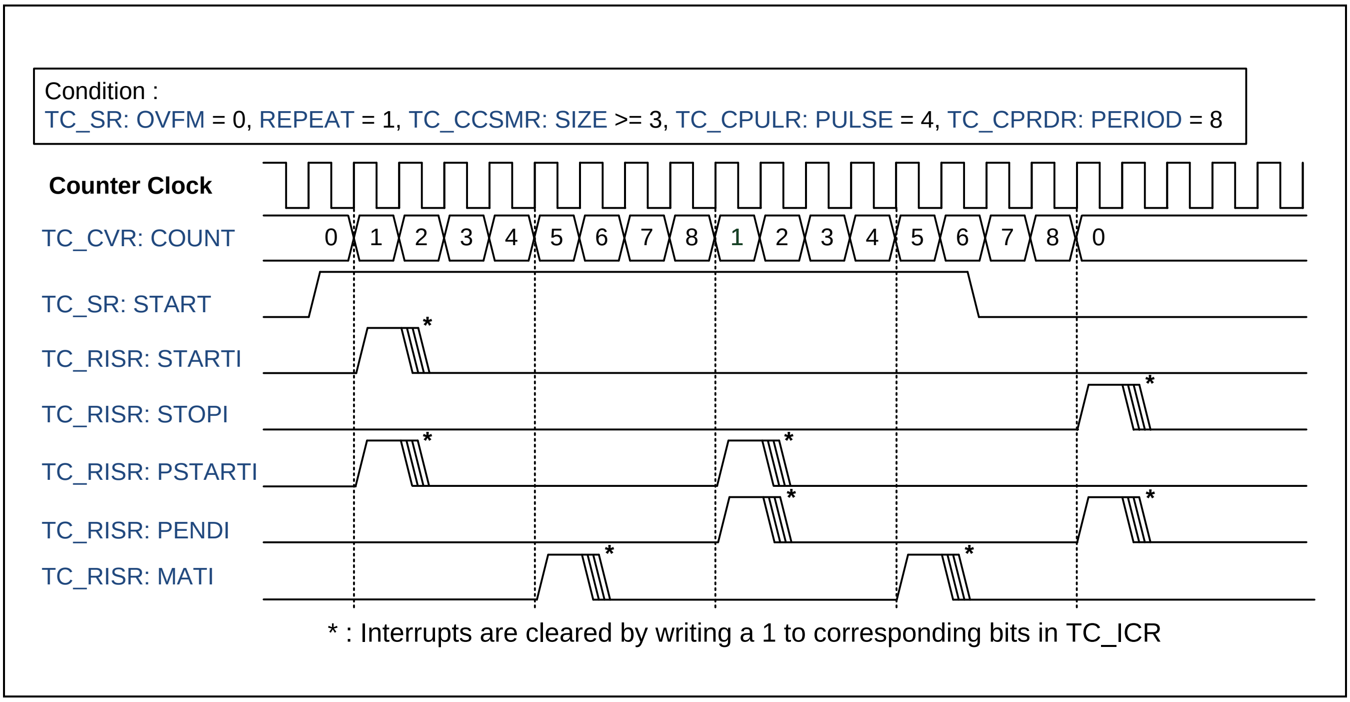

Condition :
TC_SR: OVFM = 0, REPEAT = 1, TC_CCSMR: SIZE >= 3, TC_CPULR: PULSE = 4, TC_CPRDR: PERIOD = 8
Counter Clock
TC_CVR: COUNT
0
1
2
3
4
5
6
7
8
1
2
3
4
5
6
7
8
0
TC_SR: START
TC_RISR: STARTI
TC_RISR: STOPI
TC_RISR: PSTARTI
TC_RISR: PENDI
TC_RISR: MATI
*
* : Interrupts are cleared by writing a 1 to corresponding bits in TC_ICR

Figure 16-5 Period Mode Timing

The TC can also generate two types of output signals according to PWMIM bit in TC_SR register. The two output signals are Interval signal and PWM signal. You should set the PWMEN bit in TC_SR register in order to output the signal generated through the TPWM pin. You should also configure the pin as the relevant alternative function.

16.2.7.1 Interval Operation

In an interval mode operation, the Pulse Width Modulated Output Signal (TPWM) level toggles for all instances whenever the time period end is detected. For example, if you write a value of 0x08 to the TC_PRDR register, then the timer increments until it reaches a count of 0x08. At this instance, the timer period end interrupt is generated. The time period of TPWM is 2 × TCCLK × PERIOD.

Figure 16-6 illustrates the interval operation.

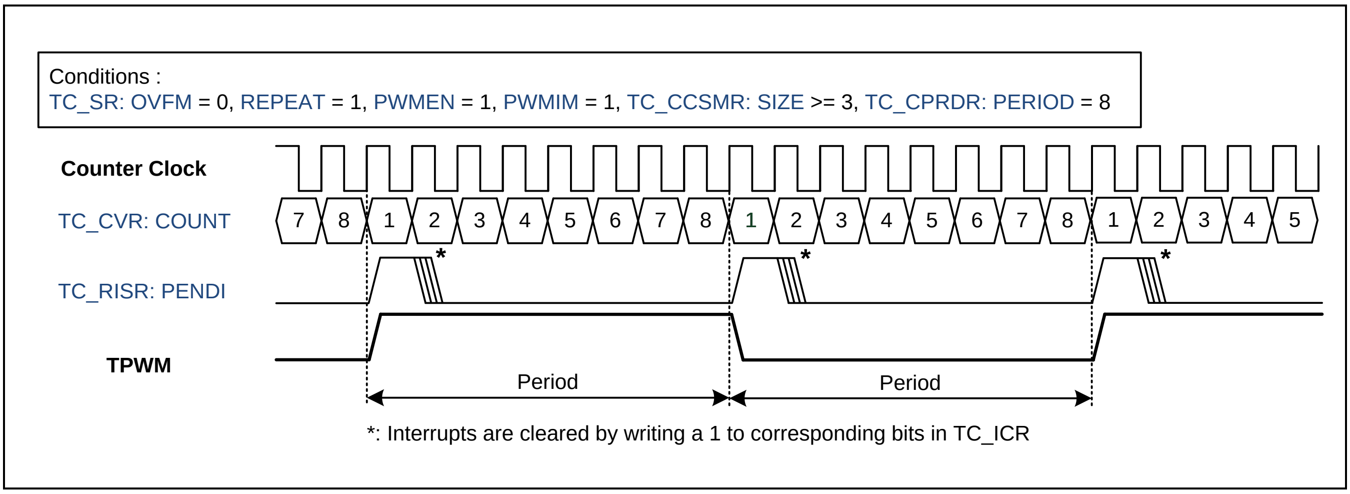

Conditions :
TC_SR: OVFM = 0, REPEAT = 1, PWMEN = 1, PWMIM = 1, TC_CCSMR: SIZE >= 3, TC_CPRDR: PERIOD = 8
Counter Clock
TC_CVR: COUNT
7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5
TC_RISR: PENDI
*
*
*
TPWM
Period
Period
*: Interrupts are cleared by writing a 1 to corresponding bits in TC_ICR

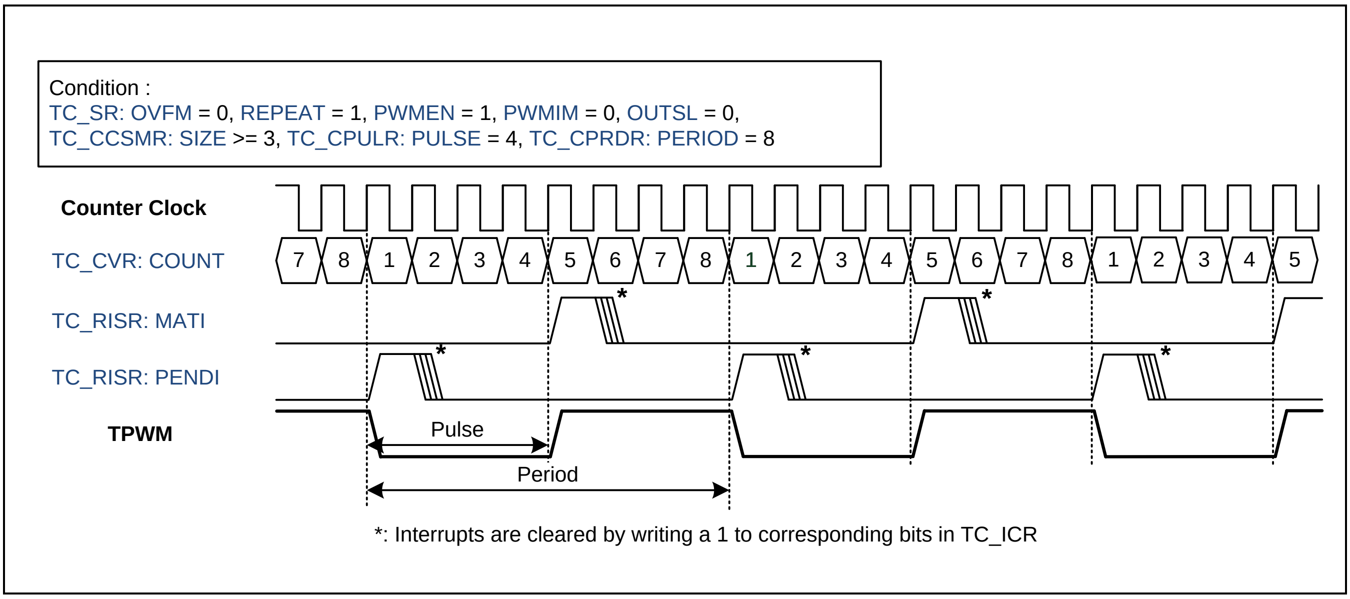
Figure 16-6 Interval Operation

16.2.7.2 PWM Operation

You can use the TC for generating the Pulse Width Modulation (PWM) signal. In this operation, you should specify the TC_CPULR register by writing a relevant value to PULSE bits in TC_PULR register. The TC copies the PULSE bits in TC_PULR register into TC_CPULR register when the timer starts or UPDATE bit in TC_CSR register is set. PULSE in TC_CPULR register should be less than or equal to PERIOD in TC_CPRDR register.

The OUTSL bit in TC_SR register determines the TPWM output signal when the timer starts. When OUTSL is clear, the output signal will be LOW and When OUTSL is set, the output signal will be HIGH. When the counter value is equal to TC_CPULR, the pulse match signal is generated and TPWM output is toggled to the opposite level of OUTSL bit. After that, the counter value will be increased until PERIOD value specified in TC_CPRDR and the period end signal is generated. At this time, if REPEAT bit is set in TC_SR register, then the counter value restarts from 1. If REPEAT is clear, then the timer stops and the counter value is cleared to "0".

Figure 16-7 illustrates the PWM operation.



Condition :
TC_SR: OVFM = 0, REPEAT = 1, PWMEN = 1, PWMIM = 0, OUTSL = 0,
TC_CCSMR: SIZE >= 3, TC_CPULR: PULSE = 4, TC_CPRDR: PERIOD = 8

*: Interrupts are cleared by writing a 1 to corresponding bits in TC_ICR

Figure 16-7 PWM Operation

16.2.7.2.1 Extension Bit

Some Periods can be Extension Periods in every 64 Periods. Extension Period has an extended pulse width that is a counter clock longer than the normal Period. PWMEX bit in TC_SR register determines which Period is Extension Period. Since PWMEX bit is 6-bit long, the PWM output has approximately up to 22-bit resolution though the counter is 16-bit long.

Table 16-2 describes the PWM extension bits.

Table 16-2 PWM Extension Bits

PWMEX Bit	Extension Period
PWMEX0	32
PWMEX1	16, 48
PWMEX2	8, 24, 40, 56
PWMEX3	4, 12, 20, 28, 36, 44, 52, 60
PWMEX4	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
PWMEX5	1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63

Figure 16-8 illustrates the PWM output signals according to PWMEX bits.

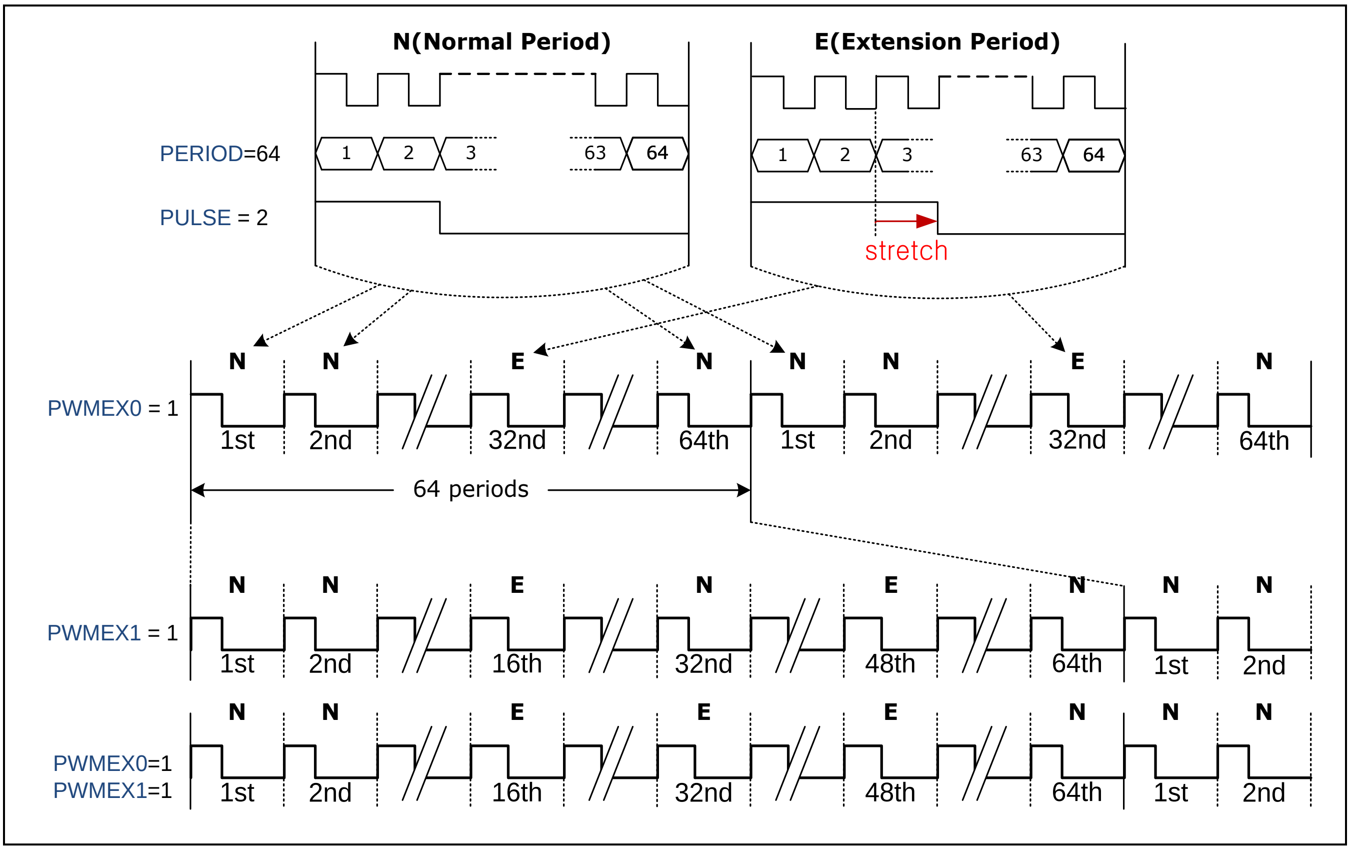


Figure 16-8 PWM Extension Waveform

The equation to calculate the duty of PWM signal is:

$$\mathrm{Duty}(\%) = \left(\frac{(\mathrm{PULSE}\times(64-\mathrm{E})) + (\mathrm{PULSE}+1)\times\mathrm{E}}{\mathrm{PERIOD}\ \times 64}\right)\times 100 = \left(\frac{\mathrm{PULSE}}{\mathrm{PERIOD}} + \frac{\mathrm{E}}{\mathrm{PERIOD}\times 64}\right)\times 100$$

$$\text{, where 'E' is number of Extension Periods in every 64 periods.}$$

For example, the PWM output has normally 50 percent duty when PERIOD is 100 and PULSE is 50. If PWMEX0 is only set in this case, then the pulse cycle of the 32nd period in 64 periods has 51 counter clocks. Therefore, the PWM output has 50.015625 percent duty because 1/64 is 0.015625. If PWMEX5 is only set, then the 32 periods in every 64 periods are Extension Periods and the duty would be 50.5 percent.

16.2.7.2.2 PWM Waveform

Figure 16-9 and *Figure 16-10* illustrates PWM waveforms according to the relationship between PERIOD and PULSE in the PWM operation. In Normal Period, when PULSE is 0, the duty is 0 percent and when PULSE is equal to PERIOD, the duty is 100 percent.

The OUTSL bit in TC_SR register determines the output level on TPWM pin when the timer is running. The TPWM signal starts from LOW when OUTSL bit is clear.

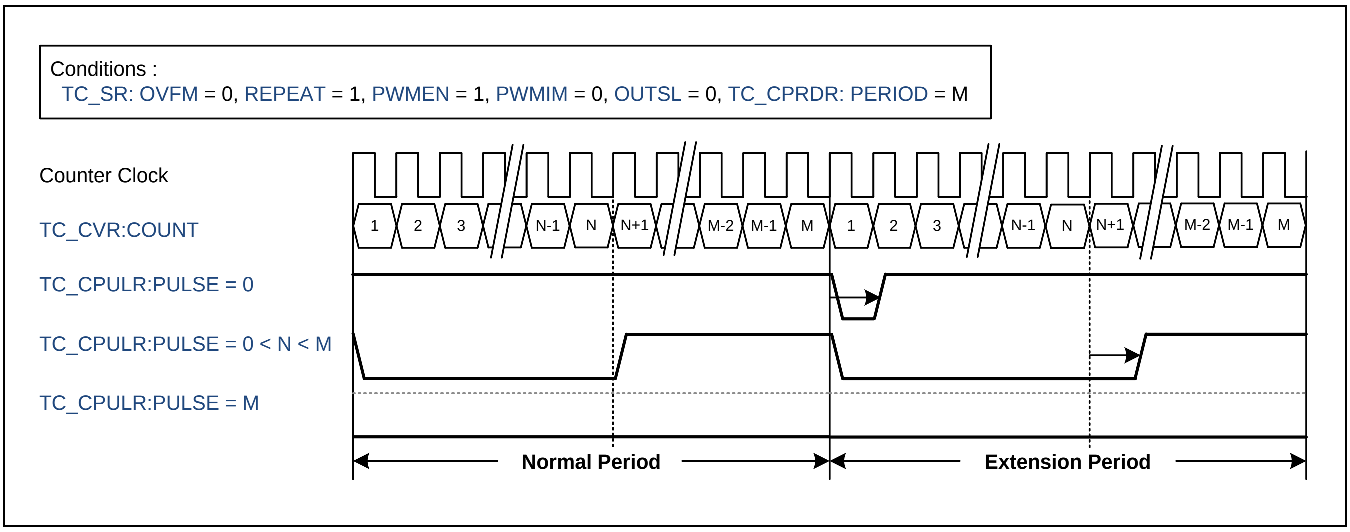

Conditions :
TC_SR: OVFM = 0, REPEAT = 1, PWMEN = 1, PWMIM = 0, OUTSL = 0, TC_CPRDR: PERIOD = M
Counter Clock
TC_CVR:COUNT
TC_CPULR:PULSE = 0
TC_CPULR:PULSE = 0 < N < M
TC_CPULR:PULSE = M
1 2 3 N-1 N N+1 M-2 M-1 M 1 2 3 N-1 N N+1 M-2 M-1 M
Normal Period
Extension Period

Figure 16-9 PWM Waveform with OUTSL = 0

If OUTSL bit is set, then the TPWM signal starts from HIGH.

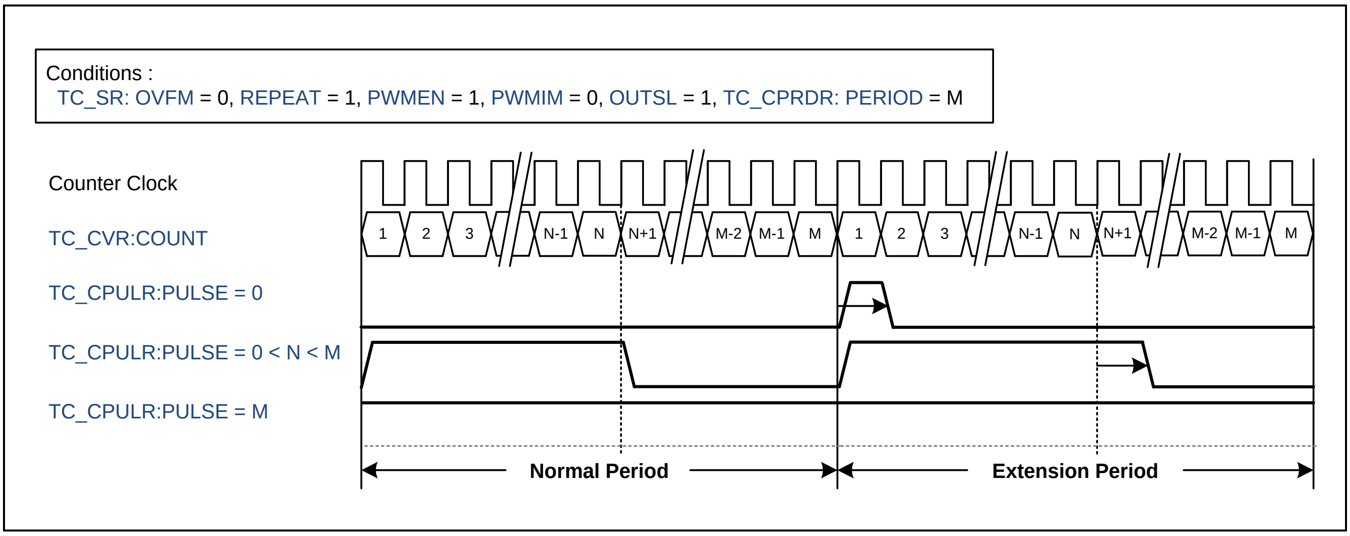

Conditions :
TC_SR: OVFM = 0, REPEAT = 1, PWMEN = 1, PWMIM = 0, OUTSL = 1, TC_CPRDR: PERIOD = M
Counter Clock
TC_CVR:COUNT
TC_CPULR:PULSE = 0
TC_CPULR:PULSE = 0 < N < M
TC_CPULR:PULSE = M
1 2 3 N-1 N N+1 M-2 M-1 M 1 2 3 N-1 N N+1 M-2 M-1 M
Normal Period
Extension Period

Figure 16-10 PWM Waveform with OUTSL = 1

16.2.7.2.3 PWM Output Polarity

When the timer stops, the PWM output polarity on TPWM pin will have different states according to STOPCLEAR, STOPHOLD, KEEP, IDLESL, and OUTSL bits in TC_SR register.

Table 16-3 describes the differences of output polarity according to control bits.

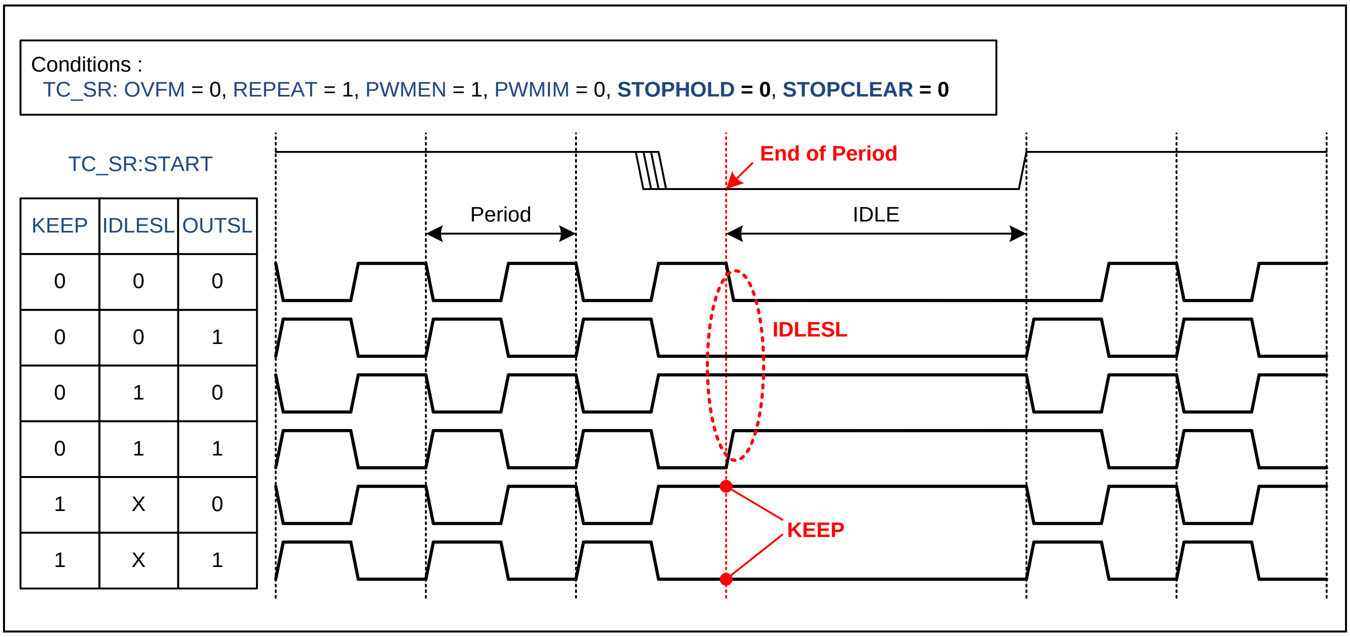
Table 16-3 PWM Output Polarity According to Control Bits

STOPCLEAR	STOPHOLD	KEEP	IDLESL	OUTSL	TPWM	Note
0	0	0	0	X	L	The timer stops at the end of period and TPWM is IDLESL.
0	0	0	1	X	H	
0	0	1	X	0	H	The timer stops at the end of period and TPWM is the opposite of OUTSL.
0	0	1	X	1	L	
0	1	X	X	X	L/H	The timer pauses immediately and TPWM keeps the last level.
1	X	X	0	X	L	The timer stops immediately and TPWM is IDLESL.
1	X	X	1	X	H	

NOTE: Priorities among control bits are STOPCLEAR > STOPHOLD > KEEP > IDLESL.

When both STOPHOLD and STOPCLEAR are clear, you can clear the START bit to make the timer stop after completing the current PERIOD cycle. In this state, if KEEP bit in TC_SR register is set, then the TC keeps the output level on TPWM pin same as the opposite level of OUTSL. If KEEP bit is set to clear, then the IDLESL bit in TC_SR register determines the output level on TPWM pin

Figure 16-11 illustrates the PWM waveform under IDLE state.



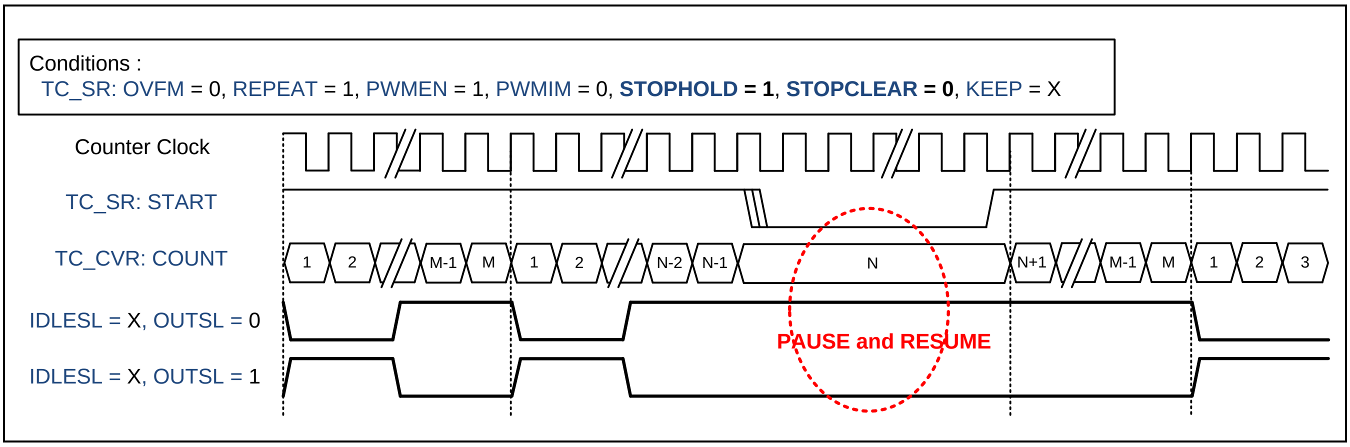
Conditions :
TC_SR: OVFM = 0, REPEAT = 1, PWMEN = 1, PWMIM = 0, **STOPHOLD = 0**, **STOPCLEAR = 0**

KEEP	IDLESL	OUTSL
0	0	0
0	0	1
0	1	0
0	1	1
1	X	0
1	X	1

Figure 16-11 PWM Waveform Under IDLE State

When clearing START bit, if STOPHOLD bit is set but STOPCLEAR bit is clear in TC_SR register, then the TC immediately stops to increase the counter value. As a result, TC keeps the counter value and the output level on TPWM pin. You can set the START bit again to restart the TC and to increase the counter value from the last value.

Figure 16-12 illustrates the PWM waveform with STOPHOLD = 1 and STOPCLEAR = 0.



Conditions :
TC_SR: OVFM = 0, REPEAT = 1, PWMEN = 1, PWMIM = 0, **STOPHOLD = 1**, **STOPCLEAR = 0**, KEEP = X

Figure 16-12 PWM Waveform with STOPHOLD = 1, STOPCLEAR = 0

If STOPCLEAR bit in TC_SR register is set when clearing START bit, then the TC stops immediately. As a result, the output level on TPWM pin changes to the level specified by IDLESL bit in TC_SR register.

Figure 16-13 illustrates the PWM waveform with STOPCLEAR = 1.

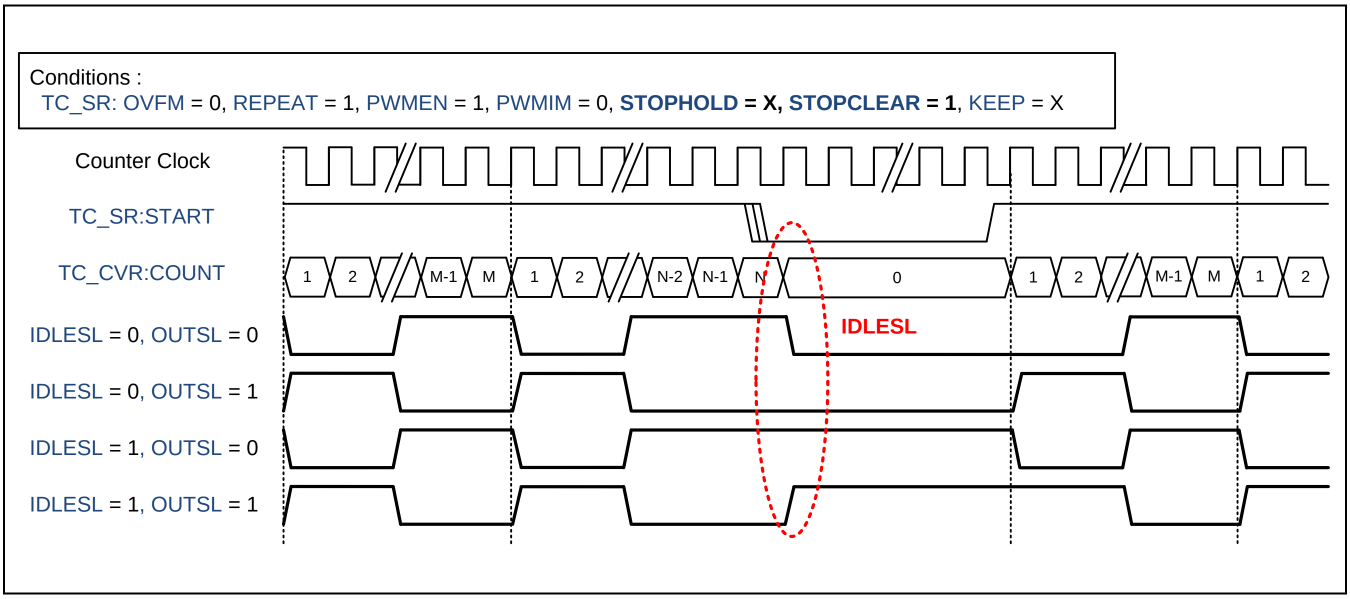


Figure 16-13 PWM Waveform with STOPCLEAR = 1

When the TC is reset, the initial level of output signal is LOW. You can immediately change the output level to the level specified by the IDLESL bit, by changing the IDLESL bit when STOPCLEAR bit is set and the timer is not running.

16.2.8 Interrupt

The section describes interrupts supported by the TC.

16.2.8.1 Types of interrupt

The TC can generate seven types of interrupt. The seven interrupts are:

- Start Interrupt (STARTI)
  Start interrupt is generated when the timer starts.
- Stop Interrupt (STOPI)
  Stop interrupt is generated when the timer stops.
- Period Start Interrupt (PSTARTI)
  Period Start interrupt is generated when the period starts.
- Period End Interrupt (PENDI)
  Period End interrupt is generated when the period ends.
- Pulse Match Interrupt (MATI)
  Pulse Match interrupt is generated when the counter value is identical to PULSE.
- Overflow Interrupt (OVFI)
  Overflow interrupt is generated when the counter overflows.
- Capture Interrupt (CAPTI)
  Capture interrupt is generated when the external capture signal is triggered.

16.2.8.2 Interruption Handling

The mechanism to carry out the interrupt handling is:

- Interrupt Service Routine (ISR) Entry and call C function.
- Read from the TC_MISR register and verify the source of the interrupt.
- Clear the corresponding interrupt by writing "1" to relevant bit in the TC_ICR register.
- Interrupt service.
- Exit ISR.

16.3 Register Description

16.3.1 Register Map Summary

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000

Register	Offset	Description	Reset Value
TC_IDR	0x0000	ID register	0x0011_000A
TC_CSSR	0x0004	Clock source selection register	0x0000_0000
TC_CEDR	0x0008	Clock enable/disable register	0x0000_0000
TC_SRR	0x000C	Software reset register	0x0000_0000
TC_CSR	0x0010	Control set register	0x0000_0000
TC_CCR	0x0014	Control clear register	0x0000_0000
TC_SR	0x0018	Status register	0x0000_0000
TC_IMSCR	0x001C	Interrupt mask set/clear register	0x0000_0000
TC_RISR	0x0020	Raw interrupt status register	0x0000_0000
TC_MISR	0x0024	Masked interrupt status register	0x0000_0000
TC_ICR	0x0028	Interrupt clear register	0x0000_0000
TC_CDR	0x002C	Clock divider register	0x0000_0000
TC_CSMR	0x0030	Counter size mask register	0x0000_000F
TC_PRDR	0x0034	Period register	0x0000_0000
TC_PULR	0x0038	Pulse register	0x0000_0000
TC_CCDR	0x003C	Current clock divider register	0x0000_0000
TC_CCSMR	0x0040	Current counter size mask register	0x0000_000F
TC_CPRDR	0x0044	Current period register	0x0000_0000
TC_CPULR	0x0048	Current pulse register	0x0000_0000
TC_CUCR	0x004C	Capture up count register	0x0000_0000
TC_CDCR	0x0050	Capture down count register	0x0000_0000
TC_CVR	0x0054	Counter value register	0x0000_0000

16.3.1.1 TC_IDR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0000, Reset Value = 0x0011_000A

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores the ID code for the corresponding IP.	0x0011_000A

16.3.1.2 TC_CSSR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															CLKSRC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
CLKSRC	[0]	RW	Clock Source Selection Field 0 = Counter Clock Source is PCLK 1 = Counter Clock Source is external clock which is provided through TCLK pin.	0

Caution: The frequency of external clock (TCLK) should be lesser than the internal clock (PCLK).

After enabling the TC clock, you cannot change the clock source (CLKSRC). Therefore, before changing the clock source (CLKSRC), you should clear the CLKEN bit of the TC_CSSR register.

16.3.1.3 TC_CEDR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	RW	Debug Mode Enable/Disable Control Bit 0 = Disables Debug Mode 1 = Enables Debug Mode If DBGEN is set, then the counter is frozen when the CPU is halted in debug mode.	0
RSVD	[30:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable/Disable Control Bit 0 = Disables Counter Clock 1 = Enables Counter Clock SWRST does not affect CLKEN bit status.	0

Caution: You should set the CLKEN bit before writing to other registers. When the CLKEN bit is clear, you cannot change values in registers. Regardless of CLKEN, Reading from registers and writing to DBGEN and SWRST are always available.

16.3.1.4 TC_SRR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = Performs Software reset	0

16.3.1.5 TC_CSR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD		PWMEX5	PWMEX4	PWMEX3	PWMEX2	PWMEX1	PWMEX0	RSVD					CAPT_R	CAPT_F	RSVD	ADTRIG	OVFM	REPEAT	PWMEN	PWMIM	KEEP	OUTSL	IDLESL	RSVD				STOPCLEAR	STOPHOLD	UPDATE	START
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	W	W	W	W	W	W	R	R	R	R	R	W	W	R	W	W	W	W	W	W	W	W	R	R	R	R	W	W	W	W

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="2">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:30]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td rowspan="8">PWMEX[5:0]</td><td rowspan="8">[29:24]</td><td rowspan="8">W</td><td colspan="2">PWM Output Extension
0 = No effect
1 = Enables corresponding extension bits</td><td rowspan="8">0</td></tr>
<tr><th>PWMEX</th><th>"Stretched" Cycle Number</th></tr>
<tr><td>PWMEX0</td><td>32</td></tr>
<tr><td>PWMEX1</td><td>16, 48</td></tr>
<tr><td>PWMEX2</td><td>8, 24, 40, 56</td></tr>
<tr><td>PWMEX3</td><td>4, 12, 20, 28, 36 , 44, 52, 60</td></tr>
<tr><td>PWMEX4</td><td>2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62</td></tr>
<tr><td>PWMEX5</td><td>1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63</td></tr>
<tr><td>RSVD</td><td>[23:19]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td>CAPT_R</td><td>[18]</td><td>W</td><td colspan="2">Capture by Rising Edge Trigger
0 = No effect
1 = Enables rising edge capture. When the TC detects rising edge of external input signal on TCAP pin, it stores the current counter value into the capture up register.</td><td>0</td></tr>
<tr><td>CAPT_F</td><td>[17]</td><td>W</td><td colspan="2">Capture by Falling Edge Trigger
0 = No effect
1 = Enables falling edge capture. When the TC detects falling edge of external input signal on TCAP pin, it stores the current counter value into the capture down register.</td><td>0</td></tr>
<tr><td>RSVD</td><td>[16]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td>ADTRIG</td><td>[15]</td><td>W</td><td colspan="2">ADC Trigger</td><td>0</td></tr>
</table>

Name	Bit	Type	Description	Reset Value
			0 = No effect 1 = Enables ADC trigger signal-out	
OVFM	[14]	W	Overflow Mode 0 = No effect 1 = Enables Overflow mode. The counter value increases until it overflows	0
REPEAT	[13]	W	Repeat Mode 0 = No effect 1 = Enables Repeat mode	0
PWMEN	[12]	W	PWM Enable 0 = No effect 1 = Enables the signal output	0
PWMIM	[11]	W	Interval Mode 0 = No effect 1 = Enables Interval mode to toggle PWM output at the end of period	0
KEEP	[10]	W	Keep Stop Level 0 = No effect 1 = Enables Keep State Mode	0
OUTSL	[9]	W	Output Start Level 0 = No effect 1 = The output signal level will be HIGH when starting	0
IDLESL	[8]	W	IDLE State Level 0 = No effect 1 = The output signal level will be HIGH in Idle state	0
RSVD	[7]	R	Reserved	0
STOPCLEAR	[3]	W	Stop Count Clear 0 = No effect 1 = Enables Stop Clear mode	0
STOPHOLD	[2]	W	Stop Count Hold 0 = No effect 1 = Enables Stop Hold mode	0
UPDATE	[1]	W	Update Parameters 0 = No effect 1 = Updates TC_CCDR, TC_CCSMR, TC_CPRDR, and TC_CPULR registers immediately to new values specified by TC_CDR, TC_CSMR, TC_PRDR, and TC_PULR respectively.	0
START	[0]	W	Start the TC 0 = No effect 1 = Starts the counter	0

Caution: If you set the UPDATE bit when the timer is running, then the TC_CCDR, TC_CCSMR, TC_CPRDR, and TC_CPULR registers immediately change to new values in TC_CDR, TC_CSMR, TC_PRDR, and TC_PULR registers respectively. But values in them will take effect only after Overflow event in Overflow mode or Period End event in Period mode.

16.3.1.6 TC_CCR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD		PWMEX5	PWMEX4	PWMEX3	PWMEX2	PWMEX1	PWMEX0	RSVD					CAPT_R	CAPT_F	RSVD	ADTRIG	OVFM	REPEAT	PWMEN	PWMIM	KEEP	OUTSL	IDLESL	RSVD				STOPCLEAR	STOPHOLD	RSVD	START
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	W	W	W	W	W	W	R	R	R	R	R	W	W	R	W	W	W	W	W	W	W	W	R	R	R	R	W	W	W	W

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="2">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:30]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td rowspan="8">PWMEX[5:0]</td><td rowspan="8">[29:24]</td><td rowspan="8">W</td><td colspan="2">PWM output extension
0 = No effect
1 = Disables corresponding extension bits</td><td rowspan="8">0</td></tr>
<tr><th>PWMEX</th><th>"Stretched" Cycle Number</th></tr>
<tr><td>PWMEX0</td><td>32</td></tr>
<tr><td>PWMEX1</td><td>16, 48</td></tr>
<tr><td>PWMEX2</td><td>8, 24, 40, 56</td></tr>
<tr><td>PWMEX3</td><td>4, 12, 20, 28, 36 , 44, 52, 60</td></tr>
<tr><td>PWMEX4</td><td>2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62</td></tr>
<tr><td>PWMEX5</td><td>1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63</td></tr>
<tr><td>RSVD</td><td>[23:19]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td>CAPT_R</td><td>[18]</td><td>W</td><td colspan="2">Capture by Rising Edge Trigger
0 = No effect
1 = Disables rising edge capture</td><td>0</td></tr>
<tr><td>CAPT_F</td><td>[17]</td><td>W</td><td colspan="2">Capture by Falling Edge Trigger
0 = No effect
1 = Disables falling edge capture</td><td>0</td></tr>
<tr><td>RSVD</td><td>[16]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td>ADTRIG</td><td>[15]</td><td>W</td><td colspan="2">ADC Trigger
0 = No effect
1 = Disables ADC trigger signal-out</td><td>0</td></tr>
<tr><td>OVFM</td><td>[14]</td><td>W</td><td colspan="2">Overflow Mode</td><td>0</td></tr>
</table>

Name	Bit	Type	Description	Reset Value
			0 = No effect 1 = Disables Overflow mode. The counter value increases until the end of period.	
REPEAT	[13]	W	Repeat Mode 0 = No effect 1 = Disables Repeat mode	0
PWMEN	[12]	W	PWM Enable 0 = No effect 1 = Disables the signal output	0
PWMIM	[11]	W	Interval Mode 0 = No effect 1 = Disables Interval mode. The type of output signal is PWM operation.	0
KEEP	[10]	W	Keep Stop Level 0 = No effect 1 = Disables Keep State mode	0
OUTSL	[9]	W	Output Start Level 0 = No effect. 1 = The output signal level will be LOW when starting	0
IDLESL	[8]	W	IDLE State Level 0 = No effect 1 = The output signal level will be LOW in Idle state	0
RSVD	[7:4]	R	Reserved	0
STOPCLEAR	[3]	W	Stop Count Clear 0 = No effect 1 = Disables Stop Clear mode	0
STOPHOLD	[2]	W	Stop Count Hold 0 = No effect 1 = Disables Stop Hold mode	0
RSVD	[1]	R	Reserved	0
START	[0]	W	Stop the TC 0 = No effect 1 = Stops the counter	0

16.3.1.7 TC_SR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD		PWMEX5	PWMEX4	PWMEX3	PWMEX2	PWMEX1	PWMEX0	RSVD					CAPT_R	CAPT_F	RSVD	ADTIRG	OVFM	REPEAT	PWMEN	PWMIM	KEEP	OUTSL	IDLESL	RSVD				STOPCLEAR	STOPHOLD	RSVD	START
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="2">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:30]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td rowspan="8">PWMEX[5:0]</td><td rowspan="8">[29:24]</td><td rowspan="8">R</td><td colspan="2">PWM output extension status
0 = Corresponding extension bits are disabled
1 = Corresponding extension bits are enabled</td><td rowspan="8">0</td></tr>
<tr><td><b>PWMEX</b></td><td><b>"Stretched" Cycle Number</b></td></tr>
<tr><td>PWMEX0</td><td>32</td></tr>
<tr><td>PWMEX1</td><td>16, 48</td></tr>
<tr><td>PWMEX2</td><td>8, 24, 40, 56</td></tr>
<tr><td>PWMEX3</td><td>4, 12, 20, 28, 36 , 44, 52, 60</td></tr>
<tr><td>PWMEX4</td><td>2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62</td></tr>
<tr><td>PWMEX5</td><td>1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63</td></tr>
<tr><td>RSVD</td><td>[23:19]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td>CAPT_R</td><td>[18]</td><td>R</td><td colspan="2">Capture by Rising Edge Trigger
0 = External rising edge capture is disabled.
1 = External rising edge capture is enabled. When the TC detects rising edge of external input signal on TCAP pin, it stores the current counter value into the capture up register.</td><td>0</td></tr>
<tr><td>CAPT_F</td><td>[17]</td><td>R</td><td colspan="2">Capture by Falling Edge Trigger
0 = External falling edge capture is disabled.
1 = External falling edge capture is enabled. When the TC detects falling edge of external input signal on TCAP pin, it stores the current counter value into the capture down register.</td><td>0</td></tr>
<tr><td>RSVD</td><td>[16]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
</table>

Name	Bit	Type	Description	Reset Value
ADTRIG	[15]	R	ADC Trigger 0 = ADC Trigger Signal-Out is disabled 1 = ADC Trigger Signal-Out is enabled If ADC trigger selection field is "TC" and this bit is set, then the ADC conversion starts by Pulse Match event.	0
OVFM	[14]	R	Overflow Mode 0 = Period mode is enabled. The counter value is increased until the end of period 1 = Overflow mode is enabled. The counter value is increased until it overflows	0
REPEAT	[13]	R	Repeat Mode 0 = Repeat mode is disabled 1 = Repeat mode is enabled The counter automatically restarts when it overflows in Overflow mode or the end of period in Period mode.	0
PWMEN	[12]	R	PWM Enable 0 = The signal output is disabled 1 = The signal output is enabled	0
PWMIM	[11]	R	Interval Mode 0 = The type of output signal is PWM operation 1 = The type of output signal is Interval operation	0
KEEP	[10]	R	Keep Stop Level 0 = Keep state mode is disabled 1 = Keep state mode is enabled When the counter stops, the TC keeps the output signal level as the last level regardless of IDLESL.	0
OUTSL	[9]	R	Output Start Level 0 = The output signal level is LOW when starting 1 = The output signal level is HIGH when starting	0
IDLESL	[8]	R	IDLE State Level 0 = The output signal level is LOW in Idle state 1 = The output signal level is HIGH in Idle state	0
RSVD	[7:4]	R	Reserved	0
STOPCLEAR	[3]	R	Stop Count Clear 0 = Stop Clear mode is disabled 1 = Stop Clear mode is enabled If you clear START bit when STOPCLEAR is set, then the counter stops and clears to zero. In this state, IDLESL determines the output signal level.	0
STOPHOLD	[2]	R	Stop Count Hold 0 = Stop Hold mode is disabled 1 = Stop Hold mode is enabled If you clear START bit when STOPHOLD is set and STOPCLEAR is clear, then the counter stops but keeps	0

Name	Bit	Type	Description	Reset Value
			the current counter value and the output signal level. Later, the counter resumes when START is set again.	
RSVD	[1]	R	Reserved	0
START	[0]	R	Start/Stop the TC 0 = The counter is stopped 1 = The counter is started	0

16.3.1.8 TC_IMSCR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																									CAPTI	OVFI	MATI	PENDI	PSTARTI	STOPI	STARTI
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	R	Reserved	0
CAPTI	[6]	RW	Capture Interrupt Mask 0 = Mask the interrupt (Disables this interrupt) 1 = Unmask the interrupt (Enables this interrupt)	0
OVFI	[5]	RW	Overflow Interrupt Mask 0 = Mask the interrupt (Disables this interrupt) 1 = Unmask the interrupt (Enables this interrupt)	0
MATI	[4]	RW	Pulse Match Interrupt Mask 0 = Mask the interrupt (Disables this interrupt) 1 = Unmask the interrupt (Enables this interrupt)	0
PENDI	[3]	RW	Period End Interrupt Mask 0 = Mask the interrupt (Disables this interrupt) 1 = Unmask the interrupt (Enables this interrupt)	0
PSTARTI	[2]	RW	Period Start Interrupt Mask 0 = Mask the interrupt (Disables this interrupt) 1 = Unmask the interrupt (Enables this interrupt)	0
STOPI	[1]	RW	Stop Interrupt Mask 0 = Mask the interrupt (Disables this interrupt) 1 = Unmask the interrupt (Enables this interrupt)	0
STARTI	[0]	RW	Start Interrupt Mask 0 = Mask the interrupt (Disables this interrupt) 1 = Unmask the interrupt (Enables this interrupt)	0

NOTE: On a Read, TC_IMSCR register gives the current value of the mask on the relevant interrupt.
A Write of 1 to a particular bit sets the mask and a Write of 0 clears the corresponding mask.

16.3.1.9 TC_RISR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																									CAPTI	OVFI	MATI	PENDI	PSTARTI	STOPI	STARTI
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	R	Reserved	0
CAPTI	[6]	R	Capture Interrupt Gives the raw interrupt state (prior to masking) of the Capture interrupt.	0
OVFI	[5]	R	Overflow Interrupt Gives the raw interrupt state (prior to masking) of the Overflow interrupt.	0
MATI	[4]	R	Pulse Match Interrupt Gives the raw interrupt state (prior to masking) of the Pulse Match interrupt.	0
PENDI	[3]	R	Period End Interrupt Gives the raw interrupt state (prior to masking) of the Period End interrupt.	0
PSTARTI	[2]	R	Period Start Interrupt Gives the raw interrupt state (prior to masking) of the Period Start interrupt.	0
STOPI	[1]	R	Stop Interrupt Gives the raw interrupt state (prior to masking) of the Stop interrupt.	0
STARTI	[0]	R	Start Interrupt Gives the raw interrupt state (prior to masking) of the Start interrupt.	0

NOTE:

1. TC_IMSCR register does not affect the TC_RISR register.
2. On a Read, TC_RISR register gives the current raw status value of the corresponding interrupt prior to masking. A Write has no effect.

16.3.1.10 TC_MISR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0024, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																									CAPTI	OVFI	MATI	PENDI	PSTARTI	STOPI	STARTI
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	R	Reserved	0
CAPTI	[6]	R	Capture Interrupt Gives the masked interrupt status (after masking) of the Capture interrupt.	0
OVFI	[5]	R	Overflow Interrupt Gives the masked interrupt status (after masking) of the Overflow interrupt.	0
MATI	[4]	R	Pulse Match Interrupt Gives the masked interrupt status (after masking) of the Pulse Match interrupt.	0
PENDI	[3]	R	Period End Interrupt Gives the masked interrupt status (after masking) of the Period End interrupt.	0
PSTARTI	[2]	R	Period Start Interrupt Gives the masked interrupt status (after masking) of the Period Start interrupt.	0
STOPI	[1]	R	Stop Interrupt Gives the masked interrupt status (after masking) of the Stop interrupt.	0
STARTI	[0]	R	Start Interrupt Gives the masked interrupt status (after masking) of the Start interrupt.	0

NOTE:

1. TC_IMSCR register affects TC_MISR register.
   TC_MISR = TC_IMSCR AND TC_RISR
2. On a Read, TC_MISR register gives the current masked status value of the corresponding interrupt. A Write has no effect.
   0 = Each interrupt does not occur
   1 = Each interrupt occurs

16.3.1.11 TC_ICR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																									CAPTI	OVFI	MATI	PENDI	PSTARTI	STOPI	STARTI
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	R	Reserved	0
CAPTI	[6]	W	0 = No effect 1 = Clears the Capture interrupt	0
OVFI	[5]	W	0 = No effect 1 = Clears the Overflow interrupt	0
MATI	[4]	W	0 = No effect 1 = Clears the Pulse Match interrupt	0
PENDI	[3]	W	0 = No effect 1 = Clears the Period End interrupt	0
PSTARTI	[2]	W	0 = No effect 1 = Clears the Period Start interrupt	0
STOPI	[1]	W	0 = No effect 1 = Clears the Stop interrupt	0
STARTI	[0]	W	0 = No effect 1 = Clears the Start interrupt	0

NOTE: On a Write of 1, the corresponding interrupt is cleared. A Write of 0 has no effect.

16.3.1.12 TC_CDR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:15]	R	Reserved	0
DIVM	[14:4]	RW	Specifies DIVM.	0
DIVN	[3:0]	RW	Specifies DIVN.	0

NOTE:

1. The equation to define Counter Clock is:
   Timer Counter Clock = ((Clock Source)/(DIVM + 1))/(2^DIVN)
2. Writing into the TC_CDR register completes when UPDATE = 1 or START = 1 condition of TC_CSR register.

Caution: Do not set DIVM to zero when DIVN is not zero.
For example,
If the counter clock is four times slower than the clock source, then the allowed settings are:
- DIVN = 0 and DIVM = 3
- DIVN = 1 and DIVM = 1

But following is forbidden:
- DIVN = 2 and DIVM = 0

16.3.1.13 TC_CSMR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0030, Reset Value = 0x0000_000F

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
SIZE	[3:0]	RW	Specifies the counter size. For example: • If SIZE is 0x07, then the TC acts as 8-bit TC. • If SIZE is 0x09, then the TC acts as 10-bit TC. • If SIZE is 0x0f, then the TC acts as 16-bit TC. Writing into the TC_CSMR register completes when UPDATE = 1 or START = 1 condition of TC_CSR register.	0xF

16.3.1.14 TC_PRDR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0034, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PERIOD	[15:0]	RW	Specifies PERIOD value. Writing into the TC_PRDR register completes when UPDATE = 1 or START = 1 condition of TC_CSR register.	0

Caution: You should set PERIOD to any value greater than "0" in Overflow mode or greater than "1" in Period mode before starting the timer.

16.3.1.15 TC_PULR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0038, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PULSE	[15:0]	RW	Specifies PULSE value. Writing into the TC_PULR register completes when UPDATE = 1 or START = 1 condition of TC_CSR register.	0

16.3.1.16 TC_CCDR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x003C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																	DIVM											DIVN			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:15]	R	Reserved	0
DIVM	[14:4]	R	Indicates current DIVM.	0
DIVN	[3:0]	R	Indicates current DIVN.	0

NOTE: The equation to define the Counter Clock is:
Counter Clock = ((Clock Source)/(DIVM + 1))/2^DIVN

16.3.1.17 TC_CCSMR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0040, Reset Value = 0x0000_000F

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	R	Reserved	0
SIZE	[3:0]	R	Indicates current Counter Size. The counter can count from 1 to $2^{(SIZE + 1)} - 1$.	0xF

16.3.1.18 TC_CPRDR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0044, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PERIOD	[15:0]	R	Indicates current PERIOD value.	0

16.3.1.19 TC_CPULR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0048, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
PULSE	[15:0]	R	Indicates current PULSE value.	0

16.3.1.20 TC_CUCR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x004C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
COUNT	[15:0]	R	Indicates the counter value captured when the last rising edge is detected.	0

Caution: The TC supports the capture function only when the clock source is PCLK.

16.3.1.21 TC_CDCR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0050, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
COUNT	[15:0]	R	Indicates the counter value captured when the last falling edge is detected.	0

Caution: The TC supports the capture function only when the clock source is PCLK.

16.3.1.22 TC_CVR

- Base Address: 0x4006_0000, 0x4006_1000, 0x4006_2000
- Address = Base Address + 0x0054, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
COUNT	[15:0]	R	Indicates the current counter value.	0

Caution: The TC_CVR register is available only when the clock source is PCLK.

17 Universal Synchronous/Asynchronous Receiver/Transmitter

17.1 Overview

The Universal Synchronous/Asynchronous Receiver/Transmitter (USART) controller is used to communicate between micro-controllers. The USART at the transmitting end takes bytes of data and transmits the individual bits sequentially starting from the Least Significant Bit (LSB). The USART at the receiving end re-assembles the bits into the originally transmitted byte.

Serial transmission is commonly used with modems and for non-networked communication between computers, terminals, and other devices.

The two primary forms of serial transmissions are:

- Synchronous serial transmission
- Asynchronous serial transmission

The Asynchronous mode uses two lines for data transfer. They are:

- Rx for the reception
- Tx for the transmission of the bits

The Synchronous mode uses an additional clock signal to strobe the input and output data.

17.1.1 Feature

The features of USART are:

- Programmable BaudRate generator
- Parity, framing, and overrun error detection
- Idle flag for J1587 protocol
- Line break generation and detection
- Automatic echo, local loopback, and remote loopback channel modes
- Multi-drop mode: address detection and generation
- Interrupt generation
- Character length of 5 to 9 bits
- Configurable start bit of data transmission
- Smart Card protocol: error signaling and re-transmission
- Asynchronous mode maximum BaudRate: PCLK/16
- Synchronous mode maximum BaudRate when providing USARTCLK clock: PCLK/2
- Synchronous mode maximum BaudRate when receiving USARTCLK clock: PCLK/4

17.1.2 Pin Description

Table 17-1 describes the USART pin description.

Table 17-1 USART Pin Description

Pin Name	Function	I/O Type	Active Level	Comments
USARTTX	USART Transmit Data Line	O	–	–
USARTRX	USART Reception Data Line	I	–	–
USARTCLK	USART Transmission Clock	Bi-direction	–	–

17.2 Functional Description

This section contains functional description of USART.

The functional description includes:

- Block diagram
- BaudRate generator
- General description

17.2.1 Block Diagram

Figure 17-1 illustrates the block diagram of USART.

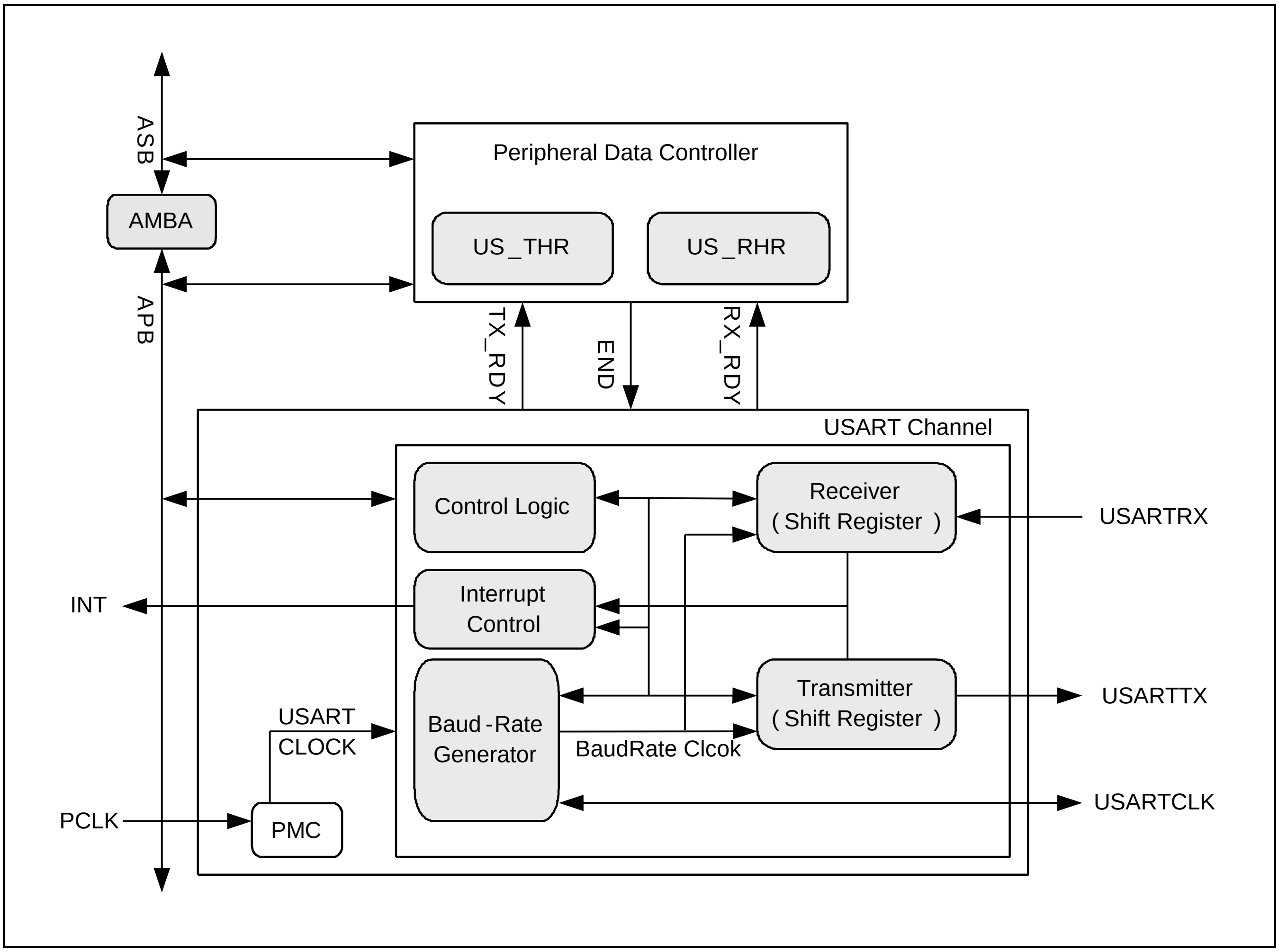


Figure 17-1 USART Block Diagram

17.2.2 BaudRate Generator

Figure 17-2 illustrates the BaudRate generator.

17.2.3 General Description

The BaudRate generator provides a periodic clock for the receiver and the transmitter. The BaudRate generator can select the internal or external clock sources. The external clock source is asserted on the USARTCLK pin. The internal clock sources are either PCLK or PCLK/8 clock sources. See Receiver and Transmitter in *Figure 17-1*.

NOTE: The duration of the external clock source period should be longer than the period of PCLK. The external clock source frequency (USARTCLK) should be less than 40 percent of the PCLK frequency.

17.3 Asynchronous Mode

When the USART is programmed to operate in asynchronous mode (SYNC = 0 in the Mode Register US_MR), then the selected clock is divided by 16 times the value (CD) written in US_BRGR (BaudRate Generator Register). If the CD[15:4] field in the US_BRGR is set to 0, then the BaudRate clock is disabled.

- BaudRate = Selected Clock/(16 × CD) where the selected clock is PCLK, PCLK/8, or USARTCLK.

17.3.1 Synchronous Mode

When the USART is programmed to operate in synchronous mode (SYNC = 1 in the Mode Register US_MR) and the selected clock is internal (CLKS[1] = 0 in the Mode Register US_MR), then the BaudRate clock is the internal selected clock divided by the value written in US_BRGR.

If the CD[15:4] field in the US_BRGR is set to 0, then the BaudRate clock is disabled.

- BaudRate = Selected Clock/CD where the selected clock is PCLK, PCLK/8, or USARTCLK.

In synchronous mode, when the external clock (CLKS[1] = 1 in the Mode Register US_MR) is selected, the signal directly provides the clock on the USARTCLK pin. No division is active. The value written in US_BRGR has no effect.

17.3.1.1 Block Diagram

Figure 17-2 illustrates the block diagram of USART BaudRate generator.

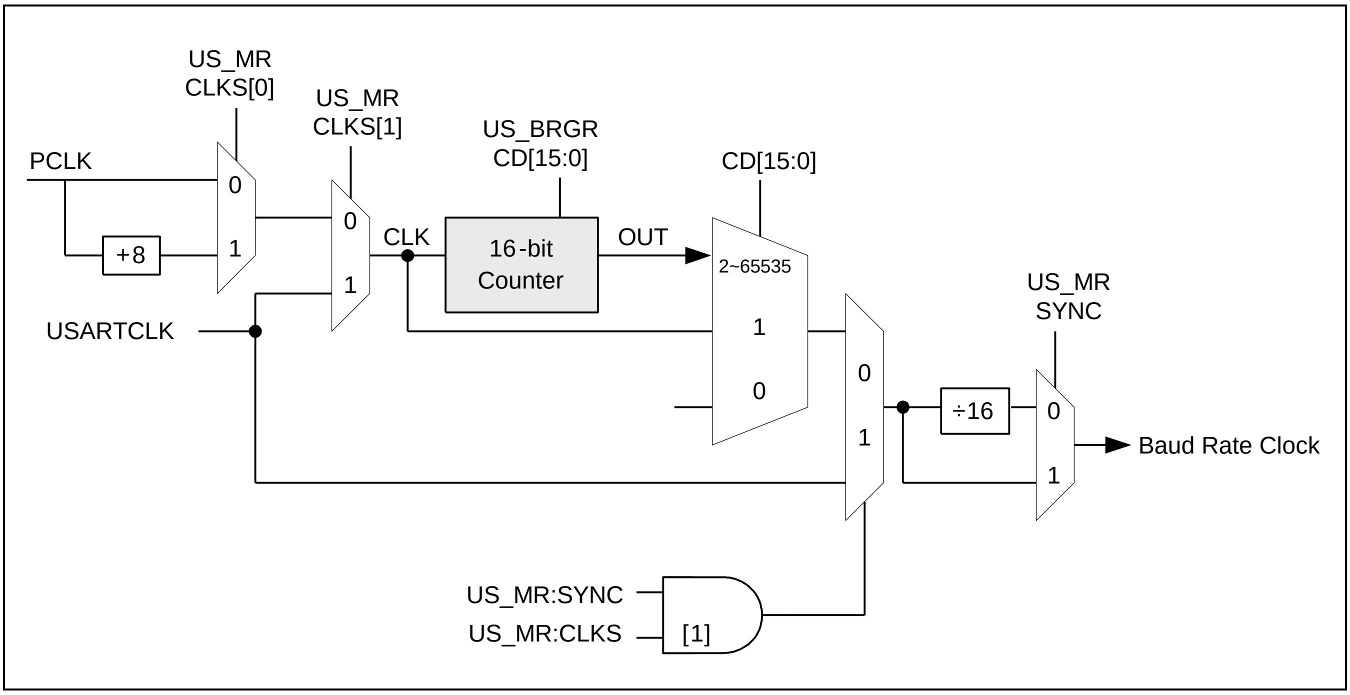


Figure 17-2 USART BaudRate Generator Block Diagram

17.3.1.2 BaudRate Configuration Example

The *Table 17-2* describes different registers configuration of the US_BRGR register for different core frequency. For each case, the calculated error shows the difference between the real BaudRate and the expected BaudRate.

In *Table 17-2*, CLKS[1:0] = 00 (PCLK selected as USART clock USARTCLK) and SYNC = 0 (asynchronous mode) in the US_MR register.

Table 17-2 describes the asynchronous mode (SYNC = 0).

Table 17-2 Asynchronous Mode (SYNC = 0)

PCLK (MHz)	US_BRGR CD[15:0]	BaudRate	% Error
40	2083	1200	– 0.02 %
	1042	2400	0.03 %
	521	4800	0.03 %
	260	9600	– 0.16 %
	174	14400	0.22 %
	130	19200	– 0.16 %
	65	38400	– 0.16 %
37.5	1953	1200	– 0.01 %
	977	2400	0.04 %
	488	4800	– 0.06 %
	244	9600	– 0.06 %
	163	14400	0.15 %
	122	19200	– 0.06 %
	61	38400	– 0.06 %
36	1875	1200	0.00 %
	938	2400	0.05 %
	469	4800	0.05 %
	234	9600	– 0.16 %
	156	14400	– 0.16 %
	117	19200	– 0.16 %
	39	57600	– 0.16 %
30	1563	1200	0.03 %
	781	2400	– 0.03 %
	391	4800	0.10 %
	195	9600	– 0.16 %
	130	14400	– 0.16 %
	98	19200	0.35 %
	49	38400	0.35 %
20	1042	1200	0.03 %

PCLK (MHz)	US_BRGR CD[15:0]	BaudRate	% Error
	521	2400	0.03 %
	260	4800	– 0.16 %
	130	9600	– 0.16 %
	87	14400	0.22 %
	65	19200	– 0.16 %
18.75	977	1200	0.04 %
	488	2400	– 0.06 %
	244	4800	– 0.06 %
	122	9600	– 0.06 %
	81	14400	– 0.47 %
	61	19200	– 0.06 %
18	938	1200	0.05 %
	469	2400	0.05 %
	234	4800	– 0.16 %
	117	9600	– 0.16 %
	78	14400	– 0.16 %
16	833	1200	– 0.04 %
	417	2400	0.08 %
	208	4800	– 0.16 %
	104	9600	– 0.16 %
	52	19200	– 0.16 %
	26	38400	– 0.16 %
15	781	1200	– 0.03 %
	391	2400	0.10 %
	195	4800	– 0.16 %
	98	9600	0.3 %
	65	14400	– 0.16 %
	49	19200	0.35 %
10	521	1200	0.03 %
	260	2400	– 0.16 %
	130	4800	– 0.16 %
	65	9600	– 0.16 %
9.375	488	1200	– 0.06 %
	244	2400	– 0.06 %
	122	4800	– 0.06 %
	61	9600	– 0.06 %
8	417	1200	0.08 %

PCLK (MHz)	US_BRGR CD[15:0]	BaudRate	% Error
	208	2400	– 0.16 %
	104	4800	– 0.16 %
	52	9600	– 0.16 %
	26	19200	– 0.16 %
	13	38400	– 0.16 %
5	260	1200	– 0.16 %
	130	2400	– 0.16 %
	65	4800	– 0.16 %
4.6875	244	1200	– 0.06 %
	122	2400	– 0.06 %
	61	4800	– 0.06 %
4	208	1200	– 0.16 %
	104	2400	– 0.16 %
	52	4800	– 0.16 %
	26	9600	– 0.16 %
	13	19200	– 0.16 %
2.5	130	1200	– 0.16 %
	65	2400	– 0.16 %
2	104	1200	– 0.16 %
	52	2400	– 0.16 %
	26	4800	– 0.16 %
	13	9600	– 0.16 %
1.25	65	1200	– 0.16 %
1	52	1200	– 0.16 %
	26	2400	– 0.16 %
	13	4800	– 0.16 %
0.5	26	1200	– 0.16 %
	13	2400	– 0.16 %
0.25	13	1200	– 0.16 %

Table 17-3 describes the synchronous mode (SYNC = 1).

Table 17-3 Synchronous Mode (SYNC = 1)

PCLK (MHz)	US_BRGR CD[15:0]	BaudRate	% Error
40	2083 × 16	1200	– 0.02 %
	1042 × 16	2400	0.03 %
	521 × 16	4800	0.03 %
	260 × 16	9600	– 0.16 %
	174 × 16	14400	0.22 %
	130 × 16	19200	– 0.16 %
	65 × 16	38400	– 0.16 %
37.5	1953 × 16	1200	– 0.01 %
	977 × 16	2400	0.04 %
	488 × 16	4800	– 0.06 %
	244 × 16	9600	– 0.06 %
	163 × 16	14400	0.15 %
	122 × 16	19200	– 0.06 %
	61 × 16	38400	– 0.06 %
36	1875 × 16	1200	0.00 %
	938 × 16	2400	0.05 %
	469 × 16	4800	0.05 %
	234 × 16	9600	– 0.16 %
	156 × 16	14400	– 0.16 %
	117 × 16	19200	– 0.16 %
	39 × 16	57600	– 0.16 %
30	1563 × 16	1200	0.03 %
	781 × 16	2400	– 0.03 %
	391 × 16	4800	0.10 %
	195 × 16	9600	– 0.16 %
	130 × 16	14400	– 0.16 %
	98 × 16	19200	0.35 %
	49 × 16	38400	0.35 %
20	1042 × 16	1200	0.03 %
	521 × 16	2400	0.03 %
	260 × 16	4800	– 0.16 %
	130 × 16	9600	– 0.16 %
	87 × 16	14400	0 .22 %
	65 × 16	19200	– 0.16 %

PCLK (MHz)	US_BRGR CD[15:0]	BaudRate	% Error
18.75	977 × 16	1200	0.04 %
	488 × 16	2400	– 0.06 %
	244 × 16	4800	– 0.06 %
	122 × 16	9600	– 0.06 %
	81 × 16	14400	– 0.47 %
	61 × 16	19200	– 0.06 %
18	938 × 16	1200	0.05 %
	469 × 16	2400	0.05 %
	234 × 16	4800	– 0.16 %
	117 × 16	9600	– 0.16 %
	78 × 16	14400	– 0.16 %
16	833 × 16	1200	– 0.04 %
	417 × 16	2400	0.08 %
	208 × 16	4800	– 0.16 %
	104 × 16	9600	– 0.16 %
	52 × 16	19200	– 0.16 %
	26 × 16	38400	– 0.16 %
15	781 × 16	1200	– 0.03 %
	391 × 16	2400	0.10 %
	195 × 16	4800	– 0.16 %
	98 × 16	9600	0.35 %
	65 × 16	14400	– 0.16 %
	49 × 16	19200	0.35 %
10	521 × 16	1200	0.03 %
	260 × 16	2400	– 0.16 %
	130 × 16	4800	– 0.16 %
	65 × 16	9600	– 0.16 %
9.375	488 × 16	1200	– 0.06 %
	244 × 16	2400	– 0.06 %
	122 × 16	4800	– 0.06 %
	61 × 16	9600	– 0.06 %
8	417 × 16	1200	0.08 %
	208 × 16	2400	– 0.16 %
	104 × 16	4800	– 0.16 %
	52 × 16	9600	– 0.16 %
	26 × 16	19200	– 0.16 %

PCLK (MHz)	US_BRGR CD[15:0]	BaudRate	% Error
	13 × 16	38400	– 0.16 %
5	260 × 16	1200	– 0.16 %
	130 × 16	2400	– 0.16 %
	65 × 16	4800	– 0.16 %
4.6875	244 × 16	1200	– 0.06 %
	122 × 16	2400	– 0.06 %
	61 × 16	4800	– 0.06 %
4	208 × 16	1200	– 0.16 %
	104 × 16	2400	– 0.16 %
	52 × 16	4800	– 0.16 %
	26 × 16	9600	– 0.16 %
	13 × 16	19200	– 0.16 %
2.5	130 × 16	1200	– 0.16 %
	65 × 16	2400	– 0.16 %
2	104 × 16	1200	– 0.16 %
	52 × 16	2400	– 0.16 %
	26 × 16	4800	– 0.16 %
	13 × 16	9600	– 0.16 %
1.25	65 × 16	1200	– 0.16 %
1	52 × 16	1200	– 0.16 %
	26 × 16	2400	– 0.16 %
	13 × 16	4800	– 0.16 %
0.5	26 × 16	1200	– 0.16 %
	13 × 16	2400	– 0.16 %
0.25	13 × 16	1200	– 0.16 %

17.4 Receiver

The receiver section includes:

- Asynchronous Receiver
- Synchronous Receiver

17.4.1 Asynchronous Receiver

The USART is configured for asynchronous operation when SYNC = 0 (8-bit of US_MR). In asynchronous mode, the USART detects the start of a received character by sampling the RXD signal until it detects a valid start bit. USART interprets a low level signal on RXD as a valid start bit if it detects the signal for more than seven cycles of the sampling clock, which is 16 times the BaudRate. Therefore, it detects the signal that is longer than 7/16 of the bit period as a valid start bit. It ignores a signal that is 7/16 of a bit period or shorter and the receiver continues to wait for a valid start bit.

When the receiver detects a valid start bit, it samples the RXD at the theoretical mid-point of each bit. It assumes that each bit lasts for 16 cycles of the sampling clock (one bit period). Therefore, the sampling point is eight cycles (0.5-bit periods) after the start of the bit. The first sampling point is 24 cycles (1.5-bit periods) after detecting the falling edge of the start bit. The receiver samples 16 cycles (1-bit period) for each subsequent bit after the previous bit.

Figure 17-3 illustrates the asynchronous mode, start bit detection.

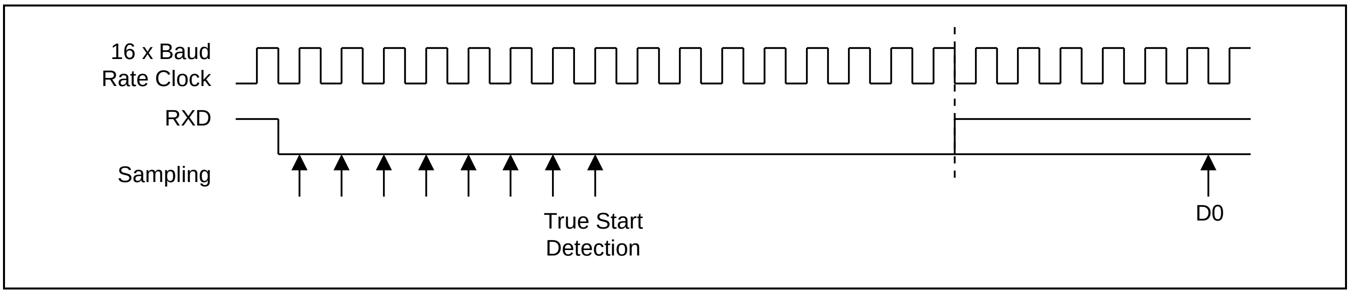


Figure 17-3 Asynchronous Mode, Start Bit Detection

Figure 17-4 illustrates the asynchronous mode, character reception.

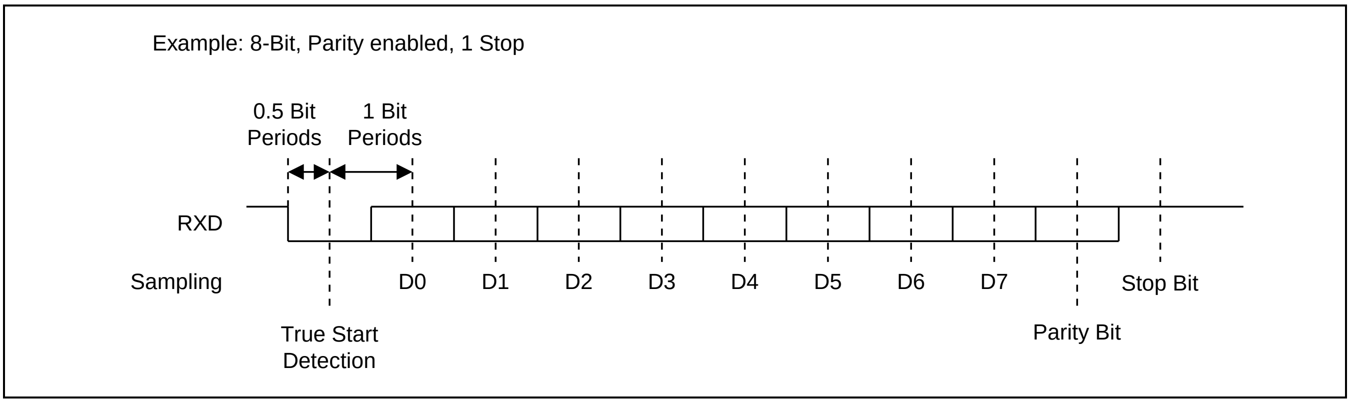


Figure 17-4 Asynchronous Mode, Character Reception

17.4.2 Synchronous Receiver

When the receiver is configured for synchronous operation (SYNC = 1), it samples the RXD signal on each rising edge of USARTCLK. If a low level is detected, then it is considered as a start. After sampling data bits, parity bit, and stop bit, the receiver waits for the next start bit.

Figure 17-5 illustrates the Synchronous mode, character reception diagram.

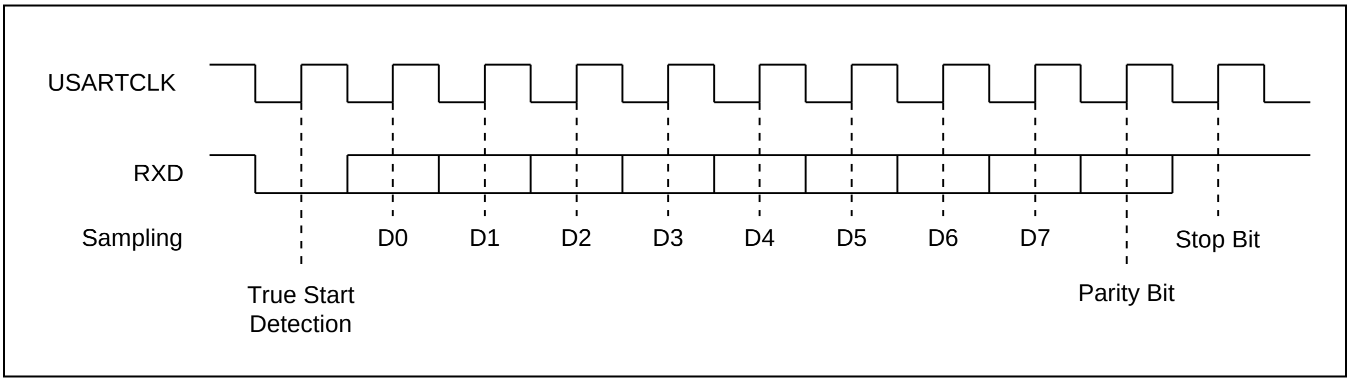


Figure 17-5 Synchronous Mode, Character Reception

17.4.2.1 Receiver Ready

When a complete character is received, it is transferred to the US_RHR and then the RXRDY status bit in US_SR is set.

17.4.2.2 Overrun Error

If the US_RHR register has not been read since the last transfer and another character is transferred to the US_RHR register, then the OVRE status bit in US_SR register is set.

17.4.2.3 Parity Error

Each time a character is received, the receiver calculates the parity of the received data bits in accordance with the field PAR[2:0] in US_MR (USART Mode Register). Then the US_MR register compares the result with the received parity bit. If the result is different, then the parity error bit PARE is set in US_SR.

17.4.2.4 Framing Error

If the receiver receives a character with a stop bit at a low level and with at least one data bit at a high level, then it generates a framing error. This sets FRAME in US_SR.

17.4.2.5 IDLE Flag

The idle flag turns low when USART receives a start bit and the flag turns high at the end of a J1587 protocol frame (after 10 stop bits). It generates an interrupt at the rising edge of the flag.

Figure 17-6 illustrates the IDLE flag diagram.

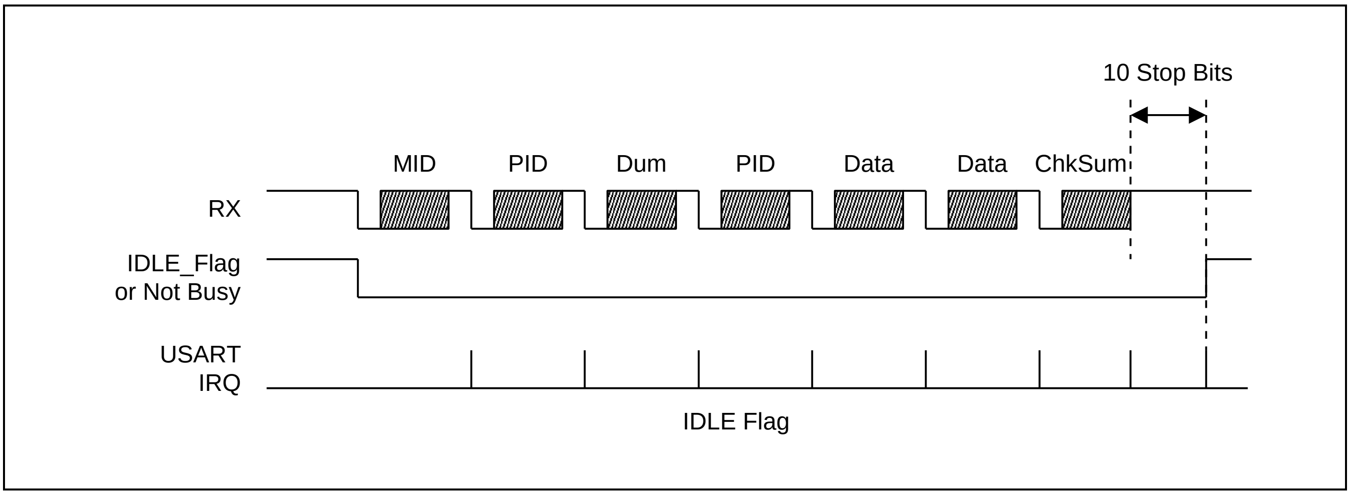


Figure 17-6 IDLE Flag

17.4.2.6 Time-Out

The Time-Out is used to detect an idle condition on the RXD line. The maximum delay for the USART to wait for a new character while the RXD line is inactive (high level) is programmed in TO[15:0] of US_RTOR (Receiver Time-out) register. When this register is set to 0, no time-out is detected.

Otherwise, the receiver waits for a first character and then initializes a counter that decrements at each bit period and reloads at each byte reception. When the counter reaches to 0, the TIMEOUT bit in US_SR is set to 1. You can start or restart waiting for a first character by setting the STTTO (Start Time-Out) bit in US_CR register.

To start a time-out, the mandatory conditions are:

- US_RTOR should not be equal to 0
- Start the time-out by setting the STTTO bit in the US_CR register to 1
- Receive one character

Calculation of Time-Out Duration is:

- Duration = Value × Bit period in asynchronous mode
- Duration = Value × 16 × Bit period in synchronous mode

17.5 Transmitter

The transmitter section includes:

- General Description
- Time-Guard
- Multi-Drop Mode

17.5.1 General Description

The transmitter has the same behavior in both synchronous and asynchronous operating modes. The start bit, data bit, parity bit, and stop bits are serially shifted keeping the least significant bit first.

The number of data bits is selected in the CHRL[1:0] field in US_MR.

The parity bit is set according to the PAR[2:0] field in US_MR register. If the parity type is even, then the parity bit depends on the one bit sum of all data bits. For odd parity, the parity bit is the inverted sum of all data bits.

The number of stop bits is selected in the NBSTOP[1:0] field in US_MR.

When a character is written to US_THR (Transmit Holding), it is transferred to the Shift Register as soon as the register is empty.

When the transfer occurs, the TXRDY bit in US_SR is set to 1 until a new character is written to US_THR. If the Transmit Shift Register and the US_THR are empty, then the TXEMPTY bit in US_SR is set to 1 (after the last stop bit of the last transfer).

Figure 17-7 illustrates the synchronous and asynchronous modes of character transmission.

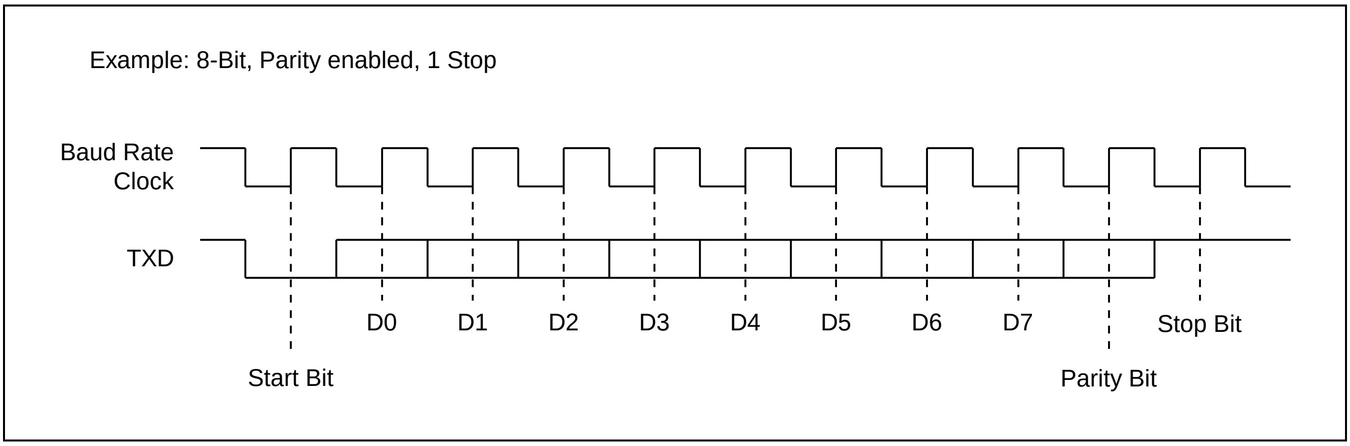


Figure 17-7 Synchronous and Asynchronous Modes, Character Transmission

17.5.2 Time-Guard

The time-guard allows the transmitter to insert an idle state on the USARTTX line between two characters. The duration of the idle state is programmed in US_TTGR (Transmitter Time-Guard). When the US_TTGR register is set to zero, then no time-guard is generated. Otherwise, the transmitter holds a high level on USARTTX after each transmitted byte during the number of bit periods programmed in US_TTGR register.

17.5.3 Multi-Drop Mode

When a PAR field in the US_MR register is equal to 11xB, the USART is configured to run in multi-drop mode for automatic address/data detection. The PARE (Parity Error) bit in the US_SR register is used to identify a data byte (parity bit is detected low) or an address byte (parity bit is detected high).Therefore, in the multi-drop mode, the PARE bit in the US_SR register is set when the data is identified as an address byte. The PARE status bit in the US_SR register is cleared by setting the PARE bit in the US_CSR (Clear Status Register) to 1. If the parity bit is detected low, the data is identified as an address byte and PARE bit in the US_SR register is not set.

The transmitter sends an address byte (parity bit set) when a Send Address Command (SENDA) is written to US_CR.

In this case, the byte written to US_THR immediately after setting SENDA bit in the US_CR is transmitted as an address. After this transmission any byte transmitted will have the parity bit cleared.

17.6 Break

The break section includes:

- Transmit Break
- Receive Break
- Interrupts
- Test Modes
- Smart Card Protocol
- Character Transmission to Smart Card
- Character Reception from Smart Card
- USART Configuration in Smart Card Mode

17.6.1 Transmit Break

The transmitter generates a break condition on the USARTTX line when the STTBRK (Start Break) command is set to 1 in US_CR (Control Register). In this case, the characters present in the Transmit Shift Register will complete before the line is held low.

To remove this break condition on the USARTTX line, the STPBRK command in US_CR should be set. The USART generates a minimum break-duration of one character length.

The USARTTX line then returns to high level (idle state) for at least 12-bit periods to ensure that the end of break is correctly detected. Then the transmitter resumes normal operation.

17.6.2 Receive Break

The receiver detects the break condition when all data, parity, and stop bits are low. When the low stop bit is detected, the receiver asserts the RXBRK (Break Received) bit in the US_SR register.

17.6.3 Interrupts

Most of the status bit in US_SR has a corresponding bit in US_IMSCR (Interrupt Mask Set/Clear Register), US_RISR (Raw Interrupt Status Register), US_MISR (Masked Interrupt Status Register), and US_ICR (Interrupt Clear Register). These bits controls the generation of interrupts by asserting the USART interrupt line connected to the NVIC.

17.6.4 Test Modes

You can program the USART to operate in three modes using the field CHMODE[1:0] in the US_MR register.

The three test modes are:

- Automatic Echo Mode
- Local Loop-back Mode
- Remote Loop-back Mode

Automatic echo mode: It allows bit by bit retransmission. When the USARTRX line receives a bit, it is sent to the USARTTX line. Programming the transmitter has no effect.

Local loop-back mode: It receives the transmitted characters. This mode does not use the USARTTX and the USARTRX pins. The output of the transmitter connects internally to the input of the receiver. The USARTRX pin level has no effect and the USARTTX pin is held high, as in idle state.

Remote loop-back mode: It directly connects the USARTRX pin to the USARTTX pin. Disable The transmitter and the receiver are disabled and have no effect. This mode allows bit by bit retransmission.

17.6.5 Smart Card Protocol

The USARTs are ISO7816-3 compliant and allow character repetition or error signaling on parity errors.

When the SMCARDPT bit is set on the US_MR register, the available functions are:

- The USART smart card protocol requires that the transmitter and the receiver are enabled.
- If PIO block allows it, then the USARTTX can be configured as an open drain output and connected to the USARTRX pin with an additional external pull-up resistor resulting in the smart card COMMS line.

17.6.6 Character Transmission to Smart Card

The USART is able to detect that the smart card has not received correctly the last transmitted byte by checking the parity error signal generated by the smart card.

When the smart card generates the parity error signal, the last transmitted byte is re-transmitted multiple times as determined in the SENDTIME[2:0] bits of the US_MR register until the error signal is no longer generated by the smart card.

When the USART detects the error signal, the FRAME error flag is set in the US_SR register.

The USART checks error signal at (t0 + 11 bits) where t0 is the falling edge of the start bit (that is between the 2 stop bits).

In the *Figure 17-8*, smart card detects the parity error. The smart card generates the error signal on the COMMS line. The USART detects the error signal and re-transmits the last character.

Figure 17-8 illustrates the smart card transmission error diagram.

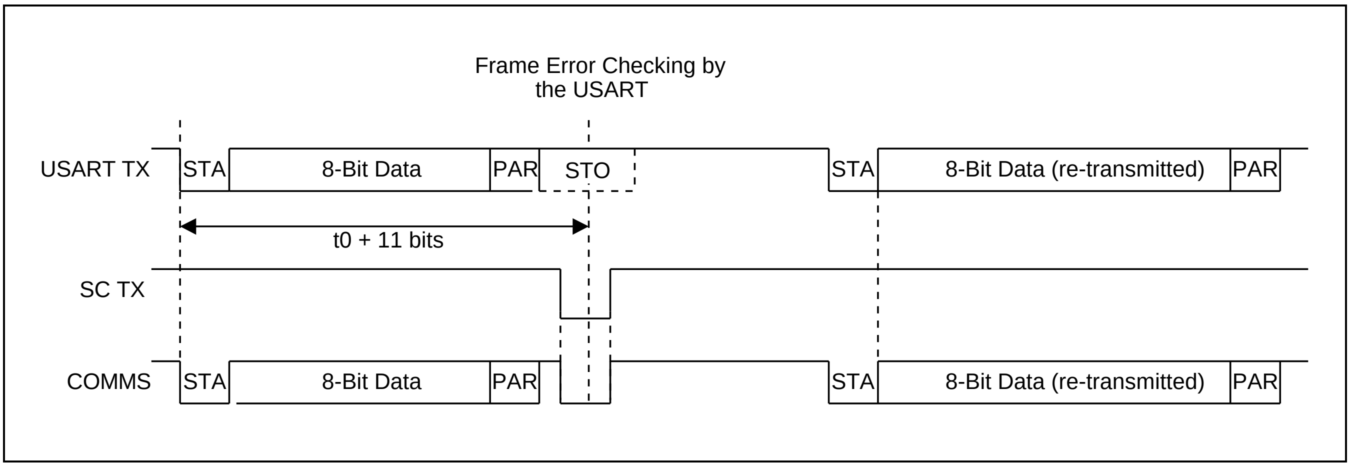


Figure 17-8 Smart Card Transmission Error

If a Direct Memory Access (DMA) transfer is used to send data byte to the smart card, then the DMA counter will not be decremented and the DMA memory pointer is not be incremented until the smart card receives a correct byte or the maximum repetition time is reached.

17.6.7 Character Reception from Smart Card

The USART is able to generate the error signal (See ISO7816-3 protocol) when it receives the last byte that has a parity error.

When USART detects the parity error, the USART transmission line is driven low for 1.0625-bit period. This period starts at t0 + 10.625 + [0:0.0625] (that is, during the 2 stop bits) to indicate the smart card that a bad reception has occurred on the USART. With T = 0 protocol type smart cards, the smart card should re-send the last character. In this scenario, the USART signals only a parity error by setting the PARE bit in the US_SR register.

Figure 17-9 illustrates the error signaling on reception diagram.

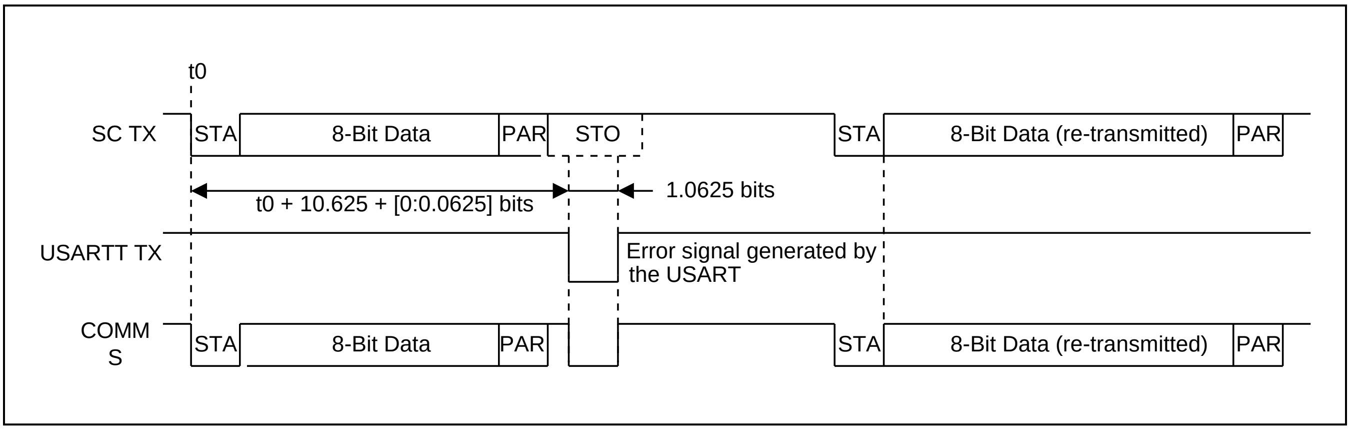

t0
SC TX
STA
8-Bit Data
PAR
STO
STA
8-Bit Data (re-transmitted)
PAR
t0 + 10.625 + [0:0.0625] bits
1.0625 bits
USARTT TX
Error signal generated by the USART
COMM S
STA
8-Bit Data
PAR
STA
8-Bit Data (re-transmitted)
PAR

Figure 17-9 Error Signaling on Reception

If a DMA transfer is used to receive data byte from the smart card, then the DMA counter will be decremented and the DMA memory pointer will be incremented even when a parity error is detected. You should reconfigure the DMA memory pointer (decrement by 1) and counter (increment by 1) to receive all the remaining bytes.

17.6.8 USART Configuration in Smart Card Mode

The USART should be set in normal mode and the number of stop bits should be programmed at two to work in smart card mode. Refer to the MODE register for more information.

17.7 Programming Examples

Interruption Handling:

- ISR (Interrupt Service Routine) Entry and call "C" function
- Read the US_MISR register and verify the source of the interrupt
- Clear the corresponding interrupt at peripheral level by writing in the US_ICR register
- Interrupt treatment: informs the background software that header or message is transmitted
- ISR Exit

17.8 Register Description

17.8.1 Register Map Summary

- Base Address: 0x4008_0000

Register	Offset	Description	Reset Value
US_IDR	0x0000	ID register	0x0011_001B
US_CEDR	0x0004	Clock enable/disable register	0x0000_0000
US_SRR	0x0008	Software reset register	0x0000_0000
US_CR	0x000C	Control register	0x0000_0000
US_MR	0x0010	Mode register	0x0000_0000
US_IMSCR	0x0014	Interrupt mask set/ clear register	0x0000_0000
US_RISR	0x0018	Raw interrupt status register	0x0000_0000
US_MISR	0x001C	Masked interrupt status register	0x0000_0000
US_ICR	0x0020	Interrupt clear register	0x0000_0000
US_SR	0x0024	Status register	0x0000_0800
US_RHR	0x0028	Receiver holding register	0x0000_0000
US_THR	0x002C	Transmitter holding register	0x0000_0000
US_BRGR	0x0030	BaudRate generator register	0x0000_0000
US_RTOR	0x0034	Receiver time-out register	0x0000_0000
US_TTGR	0x0038	Transmitter time-guard register	0x0000_0000

17.8.1.1 US_IDR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0000, Reset Value = 0x0011_001B

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores the ID code for the corresponding IP.	0x0011_001B

17.8.1.2 US_CEDR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBGEN	RSVD																														CLKEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	RW	Debug Mode Enable 0 = Disables debug mode 1 = Enables debug mode Read 0 = The debug acknowledge generated by the debug interface (dbgack_sclk input signal) has no influence on the USART function. 1 = The debug acknowledge freezes the USART function when the debug interface is activated (high level on input pin). However, Read/Write access to internal register is kept for the debug purpose.	0
RSVD	[30:1]	R	Reserved	0
CLKEN	[0]	RW	Clock Enable/Disable Control Bit 0 = Disables the USART Clock 1 = Enables the USART Clock	0

17.8.1.3 US_SRR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																															SWRST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	R	Reserved	0
SWRST	[0]	W	Software Reset 0 = No effect 1 = USART software reset	0

17.8.1.4 US_CR

- Base Address: 0x4008_0000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																			SENDA	STTTO	STPBRK	STTBRK	RSVD	TXDIS	TXEN	RXDIS	RXEN	RSTTX	RSTRX	RSVD	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W	W	W	W	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[30:13]	R	Reserved	0
SENDA	[12]	W	Send Address 0 = No effect 1 = In Multi-drop mode only, the next character written to the US_THR register is sent with the address bit set.	0
STTTO	[11]	W	Start Time out 0 = No effect 1 = Starts waiting for a character before clocking the time-out counter value	0
STPBRK	[10]	W	Stop Break 0 = No effect 1 = If a break is being transmitted, then it stops after a minimum of one character length and a high level is transmitted during 12-bit periods.	0
STTBRK	[9]	W	Start Break 0 = No effect 1 = If break is not being transmitted, then the transmission of a break starts transmission of a break after the characters present in the Transmit Shift Register is transmitted	0
RSVD	[8]	R	Reserved	0
TXDIS	[7]	W	Transmitter Disabled 0 = No effect 1 = Disables the transmitter	0
TXEN	[6]	W	Transmitter Enable 0 = No effect 1 = Enables the transmitter if TXDIS is 0.	0
RXDIS	[5]	W	Receiver Disable	0

Name	Bit	Type	Description	Reset Value
			0 = No effect 1 = Disables the receiver	
RXEN	[4]	W	Receiver Enable 0 = No effect 1 = Enables the receiver if RXDIS is 0	0
RSTTX	[3]	W	Reset Transmitter 0 = No effect 1 = Resets the transmitter logic.	0
RSTRX	[2]	W	Reset Receiver 0 = No effect 1 = Resets the receiver logic.	0
RSVD	[1:0]	R	Reserved	0

17.8.1.5 US_MR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD											DSB	RSVD	CLKO	MODE9	SMCARDPT	CHMODE		NBSTOP		PAR			SYNC	CHRL		CLKS		SENDTIME			RSVD
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="3">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:21]</td><td>R</td><td colspan="3">Reserved</td><td>0</td></tr>
<tr><td>DSB</td><td>[20]</td><td>RW</td><td colspan="3">Data Start Bit Selection
0 = Data transmission starts from LSB and ends at MSB.
1 = Data transmission starts from MSB and ends at LSB.</td><td>0'b</td></tr>
<tr><td>RSVD</td><td>[19]</td><td>R</td><td colspan="3">Reserved</td><td>0</td></tr>
<tr><td>CLKO</td><td>[18]</td><td>RW</td><td colspan="3">Clock Output Select
0 = The USART does not drive the USARTCLK pin.
1 = The USART drives the USARTCLK pin if CLKS[1] is 0.</td><td>0'b</td></tr>
<tr><td>MODE9</td><td>[17]</td><td>RW</td><td colspan="3">9-bit Character Length
0 = The CHRL field defines the character length
1 = 9-bit character length</td><td>0'b</td></tr>
<tr><td>SMCARDPT</td><td>[16]</td><td>RW</td><td colspan="3">Smart Card Protocol
0 = Disables smart card protocol on USART
1 = Enables smart card protocol on USART</td><td>0'b</td></tr>
<tr><td rowspan="5">CHMODE</td><td rowspan="5">[15:14]</td><td rowspan="5">RW</td><td colspan="3">Channel Mode
• Channel mode field</td><td rowspan="5">00'b</td></tr>
<tr><th colspan="2">CHMODE[1:0]</th><th>Mode Description</th></tr>
<tr><td>0</td><td>0</td><td>Normal Mode
The USART channel operates as an Rx/Tx USART.</td></tr>
<tr><td>0</td><td>1</td><td>Automatic Echo
Receiver data input connects to the USARTTX pin.</td></tr>
<tr><td>1</td><td>0</td><td>Local Loop-back
Transmitter output signal connects to the receiver input signal.</td></tr>
</table>

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="5">Description</th><th>Reset Value</th></tr>
<tr><td></td><td></td><td></td><td>1</td><td>1</td><td colspan="3">Remote Loop-back
USARTRX pin connects internally to the USARTTX pin.</td><td></td></tr>
<tr><td rowspan="6">NBSTOP</td><td rowspan="6">[13:12]</td><td rowspan="6">RW</td><td colspan="5">Number of Stop Bits
The interpretation of the number of stop bits depends on SYNC.
• NBSTOP configuration field</td><td rowspan="6">00'b</td></tr>
<tr><td colspan="2"><b>NBSTOP[1:0]</b></td><td colspan="2"><b>Asynchronous (SYNC = 0)</b></td><td><b>Synchronous (SYNC = 1)</b></td></tr>
<tr><td>0</td><td>0</td><td colspan="2">1 stop bit</td><td>1 stop bit</td></tr>
<tr><td>0</td><td>1</td><td colspan="2">1.5 stop bits</td><td>Reserved</td></tr>
<tr><td>1</td><td>0</td><td colspan="2">2 stop bits</td><td>2 stop bits</td></tr>
<tr><td>1</td><td>1</td><td colspan="2">Reserved</td><td>Reserved</td></tr>
<tr><td rowspan="9">PAR</td><td rowspan="9">[11:9]</td><td rowspan="9">RW</td><td colspan="5">Parity Type
• Parity type field</td><td rowspan="9">000'b</td></tr>
<tr><td colspan="2"><b>PAR[2:0]</b></td><td colspan="3"><b>Parity Type</b></td></tr>
<tr><td>0</td><td>0</td><td>0</td><td colspan="2">Even Parity</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td colspan="2">Odd Parity</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td colspan="2">Parity forced to 0 (Space)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td colspan="2">Parity forced to 1 (Mark)</td></tr>
<tr><td>1</td><td>0</td><td>X</td><td colspan="2">No parity</td></tr>
<tr><td>1</td><td>1</td><td>X</td><td colspan="2">Multi-drop mode</td></tr>
<tr><td colspan="5">NOTE: For LIN, PAR[2:0] should be set to "10X".</td></tr>
<tr><td>SYNC</td><td>[8]</td><td>RW</td><td colspan="5">Synchronous Mode Select
0 = USART operates in Asynchronous Mode.
1 = USART operates in Synchronous Mode.</td><td>0</td></tr>
<tr><td rowspan="6">CHRL</td><td rowspan="6">[7:6]</td><td rowspan="6">RW</td><td colspan="5">Character Length
Start, stop, and parity bits are added to the character length.
• Character length field</td><td rowspan="6">00'b</td></tr>
<tr><td colspan="2"><b>CHRL[1:0]</b></td><td colspan="3"><b>Character Length</b></td></tr>
<tr><td>0</td><td>0</td><td colspan="3">5-bits</td></tr>
<tr><td>0</td><td>1</td><td colspan="3">6-bits</td></tr>
<tr><td>1</td><td>0</td><td colspan="3">7-bits</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">8-bits</td></tr>
<tr><td rowspan="3">CLKS</td><td rowspan="3">[5:4]</td><td rowspan="3">RW</td><td colspan="5">Clock selection (BaudRate generator input clock)
• CLKS clock selection field</td><td rowspan="3">00'b</td></tr>
<tr><td colspan="2"><b>CLKS[1:0]</b></td><td colspan="3"><b>Selected Clock</b></td></tr>
<tr><td>0</td><td>0</td><td colspan="3">PCLK</td></tr>
</table>

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="4">Description</th><th>Reset Value</th></tr>
<tr><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td>0</td><td>1</td><td colspan="2">PCLK/8</td><td rowspan="2"></td></tr>
<tr><td>1</td><td>X</td><td colspan="2">External clock (USARTCLK)</td></tr>
<tr><td rowspan="6">SENDTIME</td><td rowspan="6">[3:1]</td><td rowspan="6">RW</td><td colspan="4">Number of re-transmission
Indicates the maximum number of repetitions a character has to be transmitted by the USART when configured as smart card protocol.
• SENDTIME configuration field</td><td rowspan="6">000'b</td></tr>
<tr><td colspan="3"><b>SENDTIME[2:0]</b></td><td><b>Time Number</b></td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td colspan="4">–</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>7</td></tr>
<tr><td>RSVD</td><td>[1:0]</td><td>R</td><td colspan="4">Reserved</td><td>0</td></tr>
</table>

17.8.1.6 US_IMSCR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																					IDLE	TXEMPTY	TIMEOUT	PARE	FRAME	OVRE	RSVD		RXBRK	TXRDY	RXRDY
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	R	Reserved	0
IDLE	[10]	RW	IDLE Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt) 1 = This interrupt is not masked. (Enables an interrupt)	0
TXEMPTY	[9]	RW	Transmitter Empty Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt) 1 = This interrupt is not masked. (Enables an interrupt)	0
TIMEOUT	[8]	RW	Time-Out Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt) 1 = This interrupt is not masked. (Enables an interrupt)	0
PARE	[7]	RW	Parity Error Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt) 1 = This interrupt is not masked. (Enables an interrupt)	0
FRAME	[6]	RW	Framing Error Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt) 1 = This interrupt is not masked. (Enables an interrupt)	0
OVRE	[5]	RW	Overrun Error Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt) 1 = This interrupt is not masked. (Enables an interrupt)	0
RSVD	[4:3]	R	Reserved	0
RXBRK	[2]	RW	Receiver Break Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt) 1 = This interrupt is not masked. (Enables an interrupt)	0
TXRDY	[1]	RW	Transmitter Ready Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt) 1 = This interrupt is not masked. (Enables an interrupt)	0
RXRDY	[0]	RW	Receiver Ready Interrupt Mask 0 = This interrupt is masked. (Disables an interrupt)	0

Name	Bit	Type	Description	Reset Value
			1 = This interrupt is not masked. (Enables an interrupt)	

NOTE: On a Read, the US_IMSCR register gives the current value of the mask on the relevant interrupt. A Write of 1 to a particular bit, sets the mask, enabling the interrupt to be read. A Write of 0 clears the corresponding mask.

17.8.1.7 US_RISR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																					IDLE	TXEMPTY	TIMEOUT	PARE	FRAME	OVRE	RSVD		RXBRK	TXRDY	RXRDY
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	R	Reserved	0
IDLE	[10]	R	Idle Raw Interrupt State Gives the raw interrupt state (prior to masking) of the idle interrupt	0
TXEMPTY	[9]	R	Transmitter Empty Raw Interrupt State Gives the raw interrupt state (prior to masking) of the TXEMPTY interrupt	0
TIMEOUT	[8]	R	Time-Out Raw Interrupt State Gives the raw interrupt state (prior to masking) of the TIME-OUT interrupt	0
PARE	[7]	R	Parity Error Raw Interrupt State Gives the raw interrupt state (prior to masking) of the PARE interrupt	0
FRAME	[6]	R	Framing Error Raw Interrupt State Gives the raw interrupt state (prior to masking) of the FRAME interrupt	0
OVRE	[5]	R	Overrun Error Raw Interrupt State Gives the raw interrupt state (prior to masking) of the OVRE interrupt	0
RSVD	[4:3]	R	Reserved	0
RXBRK	[2]	R	Receiver Break Raw Interrupt State Gives the raw interrupt state (prior to masking) of the RXBRK interrupt	0
TXRDY	[1]	R	Transmitter Ready Raw Interrupt State Gives the raw interrupt state (prior to masking) of the TXRDY interrupt	0
RXRDY	[0]	R	Receiver Ready Raw Interrupt State Gives the raw interrupt state (prior to masking) of the RXRDY interrupt	0

NOTE: On a Read, US_RISR register gives the current raw status value of the corresponding interrupt prior to masking. A Write has no effect.

17.8.1.8 US_MISR

- Base Address: 0x4008_0000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																					IDLE	TXEMPTY	TIMEOUT	PARE	FRAME	OVRE	RSVD		RXBRK	TXRDY	RXRDY
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	R	Reserved	0
IDLE	[10]	R	Idle Masked Interrupt State Gives the masked interrupt state (prior to masking) of the idle interrupt.	0
TXEMPTY	[9]	R	Transmitter Empty Masked Interrupt State Gives the masked interrupt state (prior to masking) of the TXEMPTY interrupt	0
TIMEOUT	[8]	R	Time-Out Masked Interrupt State Gives the masked interrupt state (prior to masking) of the TIMEOUT interrupt	0
PARE	[7]	R	Parity Error Masked Interrupt State Gives the masked interrupt state (prior to masking) of the PARE interrupt	0
FRAME	[6]	R	Framing Error Masked Interrupt State Gives the masked interrupt state (prior to masking) of the FRAME interrupt	0
OVRE	[5]	R	Overrun Error Masked Interrupt State Gives the masked interrupt state (prior to masking) of the OVRE interrupt	0
RSVD	[4:3]	R	Reserved	0
RXBRK	[2]	R	Receiver Break Masked Interrupt State Gives the masked interrupt state (prior to masking) of the RXBRK interrupt	0
TXRDY	[1]	R	Transmitter Ready Masked Interrupt State Gives the masked interrupt state (prior to masking) of the TXRDY interrupt	0
RXRDY	[0]	R	Receiver Ready Masked Interrupt State Gives the masked interrupt state (prior to masking) of the	0

Name	Bit	Type	Description	Reset Value
			RXRDY interrupt	

NOTE: On a Read, the US_MISR register provides the current masked status value of the corresponding interrupt. A Write has no effect.

17.8.1.9 US_ICR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																					IDLE	RSVD	TIMEOUT	PARE	FRAME	OVRE	RSVD		RXBRK	RSVD	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	R	W	W	W	W	R	R	W	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	R	Reserved	0
IDLE	[10]	W	Idle Masked Interrupt State 0 = No effect 1 = Clears the idle interrupt	0
RSVD	[9]	R	Reserved	0
TIMEOUT	[8]	W	Time-Out Masked Interrupt State 0 = No effect 1 = Clears the TIMEOUT interrupt	0
PARE	[7]	W	Parity Error Masked Interrupt State 0 = No effect 1 = Clears the PARE interrupt	0
FRAME	[6]	W	Framing Error Masked Interrupt State 0 = No effect 1 = Clears the FRAME interrupt	0
OVRE	[5]	W	Overrun Error Masked Interrupt State 0 = No effect 1 = Clears the OVRE interrupt	0
RSVD	[4:3]	R	Reserved	0
RXBRK	[2]	W	Receiver Break Masked Interrupt State 0 = No effect 1 = Clears the RXBRK interrupt	0
RSVD	[1:0]	R	Reserved	0

NOTE: On a Write of 1, the corresponding interrupt is cleared. A Write of 0 has no effect.

17.8.1.10 US_SR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0024, Reset Value = 0x0000_0800

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																				IDLEFLAG	IDLE	TXEMPTY	TIMEOUT	PARE	FRAME	OVRE	RSVD		RXBRK	TXRDY	RXRDY
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:12]	R	Reserved	0
IDLEFLAG	[11]	R	Idle Flag 0 = A frame is being received by the USART. 1 = No frame is being received by the USART This bit indicates a frame transmission in J1587 protocol. It is turned low when a reception starts and turned high when a reception is followed by at least 10 stop bits (10-bits at high level).	1'b
IDLE	[10]	R	Idle Interrupt 0 = No end of J1587 protocol frame 1 = An end of J1587 protocol frame occurs.	0'b
TXEMPTY	[9]	R	Transmitter Empty 0 = There are characters in either US_THR or the Transmit Shift Register. 1 = There are no characters in both US_THR and the Transmit Shift Register Equals to zero when the USART is disabled or after reset. Transmitter Enable command in US_CR sets this bit to 1.	0'b
TIMEOUT	[8]	R	Time-Out 0 = There is no time-out since the last "Start Time out" command or the Time-Out Register is 0. 1 = There is a time-out since the last "Start Time- Out" command.	0'b
PARE	[7]	R	Parity Error 0 = No parity bit is detected false (or a parity bit high in multi-drop mode) since the last "Reset Status Bits" command. 1 = At least one parity bit is detected false (or a parity bit high in multi-drop mode) since the last "Reset Status Bits"	0'b

Name	Bit	Type	Description	Reset Value
			command.	
FRAME	[6]	R	Framing Error 0 = No stop bit is detected low since the last "Reset Status Bits" command. 1 = At least one stop bit is detected low since the last "Reset Status Bits" command.	0'b
OVRE	[5]	R	Overrun Error 0 = No byte is transferred from the Receive Shift Register to the US_RHR when RxRDY was asserted since the last "Reset Status Bits" command. 1 = At least one byte is transferred from the Receive Shift Register to the US_RHR when RxRDY was asserted since the last "Reset Status Bits" command.	0'b
RSVD	[4:3]	R	Reserved	0
RXBRK	[2]	R	Receiver Break 0 = "No Break Received" is detected since the last "Reset Status Bits" command in the Control Register. 1 = "Break Received" is detected since the last "Reset Status Bits" command in the Control Register.	0'b
TXRDY	[1]	R	Transmitter Ready 0 = A character is in the US_THR waiting to be transferred to the Transmit Shift Register or the transmitter is disabled. 1 = There is no character in the US_THR. Equals to zero when the USART is disabled or at reset. Transmitter Enable command (in US_CR) sets this bit to 1.	0'b
RXRDY	[0]	R	Receiver Ready 0 = No complete character has been received since the last read of the US_RHR or the receiver is disabled. 1 = At least one complete character is received and the US_RHR has not been read yet.	0'b

17.8.1.11 US_RHR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	R	Reserved	0
RXCHR	[8:0]	R	Received Character Last character received if RXRDY is set. The register maintains the value until it receives a new byte. When number of data bits is less than 9-bit, the bits are right aligned.	0x000

Caution: On a Read, the US_RHR register clears the RXRDY bit.
During the debug mode, you should use the ghost registers to avoid clearing RXRDY bit.

17.8.1.12 US_THR

- Base Address: 0x4008_0000
- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	R	Reserved	0
TXCHR	[8:0]	W	Character to be Transmitted If TXRDY is at a logical 1, then this is the next character to be transmitted (after a current one, if already a character is present in the transmit shift register). If TXRDY is at a logical 0, then the current character in the US_THR register is overwritten. When number of data bits is less than 9-bit, then the bits are right aligned.	0x000

17.8.1.13 US_BRGR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

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

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="2">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:16]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td rowspan="5">CD</td><td rowspan="5">[15:4]</td><td rowspan="5">RW</td><td colspan="2">Clock Divider
This register has no effect if the synchronous mode is selected with an external clock.
• ASYNC Mode:
CD = FLOOR[Fclk/(BaudRate × 16), 1]
• SYNC Mode:
CD = FLOOR[Fclk/BaudRate, 1]</td><td rowspan="5">0x0000</td></tr>
<tr><td><b>CD[15:4]</b></td><td><b>Action</b></td></tr>
<tr><td>0</td><td>Disables clock</td></tr>
<tr><td>1</td><td>Bypasses clock divider</td></tr>
<tr><td>2 to 65535</td><td>• BaudRate (Asynchronous Mode)
= Fclk/(16 × (CD + FRACTION/16))
• BaudRate (Synchronous Mode)
= Fclk/(CD + FRACTION/16)</td></tr>
<tr><td rowspan="3">FRACTION</td><td rowspan="3">[3:0]</td><td rowspan="3">RW</td><td colspan="2">Fractional Correction Value
This register has no effect if the synchronous mode is selected with an external clock.
• ASYNC Mode:
FRACTION = ROUND[((Fclk/(BaudRate × 16)) – CD) × 16, 0]
• SYNC Mode:
FRACTION = ROUND[((Fclk/BaudRate) – CD) × 16, 0]</td><td rowspan="3">0x0000</td></tr>
<tr><td><b>FRACTION[3:0]</b></td><td><b>Action</b></td></tr>
<tr><td>0 to 15</td><td>• BaudRate (Asynchronous Mode)
= Fclk/(16 × (CD + FRACTION/16))
• BaudRate (Synchronous Mode)
= Fclk/(CD + FRACTION/16)</td></tr>
</table>

NOTE: FCLK is the USART input clock and BaudRate is the desired communication speed.

Caution: In the synchronous mode, the programmed value should be even to ensure a 50:50 mark/space ratio.
You should enable the clock (that is CD is different to zero) after configuring the BaudRate clock using the US_MR register.
You should not use CD = 1 when selecting the internal clock (PCLK) that is USCLKS[1:0] = 0).

17.8.1.14 US_RTOR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0034, Reset Value = 0x0000_0000

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

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="2">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:16]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td rowspan="5">TO</td><td rowspan="5">[15:0]</td><td rowspan="5">RW</td><td colspan="2">Time-Out Value
When a value is written to this register, a time-out start command is automatically performed.
• Time-Out configuration field</td><td rowspan="5">0x0000</td></tr>
<tr><th>TO[15:0]</th><th>Action</th></tr>
<tr><td>0</td><td>Disables the receiver time-out function</td></tr>
<tr><td>1 to 65565</td><td>The time-out counter is loaded with TO[15:0] when the time-out start command is given or when each new data character is received (after reception has started).</td></tr>
<tr><td colspan="2">• In asynchronous mode:
Time-out duration = TO[15:0] × Bit period
• In synchronous mode:
Time-out duration = TO[15:0] × 16 × Bit period</td></tr>
</table>

Caution: When the receiver is disabled by setting the RXDIS bit in the US_CR register, the time-out is stopped. If the receiver is re-enabled by setting the RXEN bit in the US_CR register, then the time-out restarts from where it was stopped (it is not reset).

17.8.1.15 US_TTGR

- Base Address: 0x4008_0000
- Address = Base Address + 0x0038, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																								TG							
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R W	R W	R W	R W	R W	R W	R W	R W

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="2">Description</th><th>Reset Value</th></tr>
<tr><td>RSVD</td><td>[31:8]</td><td>R</td><td colspan="2">Reserved</td><td>0</td></tr>
<tr><td rowspan="5">TG</td><td rowspan="5">[7:0]</td><td rowspan="5">RW</td><td colspan="2">Time Guard Value
Time Guard configuration field</td><td rowspan="5">0x00</td></tr>
<tr><td><b>TO[15:0]</b></td><td><b>Action</b></td></tr>
<tr><td>0</td><td>Disables the transmitter time guard function</td></tr>
<tr><td>1 to 255</td><td>USARTTX is inactive high after the transmission of each character for the time guard duration.</td></tr>
<tr><td colspan="2">Time guard duration = TG[7:0] × bit period</td></tr>
</table>

17.9 4 to 40 MHz Asynchronous Mode

Table 17-4 Asynchronous Mode (SYNC = 0)

SYSCLK	PDIV	PCLK	US_BRGR CD[15:0]	Baud Rate	Result	% Error
4 MHz	1	4	208	1200	1201.92	– 0.16 %
			104	2400	2403.85	– 0.16 %
			52	4800	4807.69	– 0.16 %
			26	9600	9615.38	– 0.16 %
			13	19200	19230.77	– 0.16 %
	2	2	104	1200	1201.92	– 0.16 %
			52	2400	2403.85	– 0.16 %
			26	4800	4807.69	– 0.16 %
			13	9600	9615.38	– 0.16 %
	4	1	52	1200	1201.92	– 0.16 %
			26	2400	2403.85	– 0.16 %
			13	4800	4807.69	– 0.16 %
	8	0.5	26	1200	1201.92	– 0.16 %
			13	2400	2403.85	– 0.16 %
	16	0.25	13	1200	1201.92	– 0.16 %
8 MHz	1	8	417	1200	1199.04	0.08 %
			208	2400	2403.85	– 0.16 %
			104	4800	4807.69	– 0.16 %
			52	9600	9615.38	– 0.16 %
			26	19200	19230.77	– 0.16 %
			13	38400	38461.54	– 0.16 %
	2	4	208	1200	1201.92	– 0.16 %
			104	2400	2403.85	– 0.16 %
			52	4800	4807.69	– 0.16 %
			26	9600	9615.38	– 0.16 %
			13	19200	19230.77	– 0.16 %
	4	2	104	1200	1201.92	– 0.16 %
			52	2400	2403.85	– 0.16 %
			26	4800	4807.69	– 0.16 %
			13	9600	9615.38	– 0.16 %
	8	1	52	1200	1201.92	– 0.16 %
			26	2400	2403.85	– 0.16 %
			13	4800	4807.69	– 0.16 %
	16	0.5	26	1200	1201.92	– 0.16 %

SYSCLK	PDIV	PCLK	US_BRGR CD[15:0]	Baud Rate	Result	% Error
			13	2400	2403.85	– 0.16 %
16 MHz	1	16	833	1200	1200.48	– 0.04 %
			417	2400	2398.08	0.08 %
			208	4800	4807.69	– 0.16 %
			104	9600	9615.38	– 0.16 %
			52	19200	19230.77	– 0.16 %
			26	38400	38461.54	– 0.16 %
	2	8	417	1200	1199.04	0.08 %
			208	2400	2403.85	– 0.16 %
			104	4800	4807.69	– 0.16 %
			52	9600	9615.38	– 0.16 %
			26	19200	19230.77	– 0.16 %
			13	38400	38461.54	– 0.16 %
	4	4	208	1200	1201.92	– 0.16 %
			104	2400	2403.85	– 0.16 %
			52	4800	4807.69	– 0.16 %
			26	9600	9615.38	– 0.16 %
			13	19200	19230.77	– 0.16 %
	8	2	104	1200	1201.92	– 0.16 %
			52	2400	2403.85	– 0.16 %
			26	4800	4807.69	– 0.16 %
			13	9600	9615.38	– 0.16 %
	16	1	52	1200	1201.92	– 0.16 %
			26	2400	2403.85	– 0.16 %
			13	4800	4807.69	– 0.16 %
20 MHz	1	20	1042	1200	1199.62	0.03 %
			521	2400	2399.23	0.03 %
			260	4800	4807.69	– 0.16 %
			130	9600	9615.38	– 0.16 %
			87	14400	14367.82	0.22 %
			65	19200	19230.77	– 0.16 %
	2	10	521	1200	1199.62	0.03 %
			260	2400	2403.85	– 0.16 %
			130	4800	4807.69	– 0.16 %
			65	9600	9615.38	– 0.16 %
	4	5	260	1200	1201.92	– 0.16 %
			130	2400	2403.85	– 0.16 %

SYSCLK	PDIV	PCLK	US_BRGR CD[15:0]	Baud Rate	Result	% Error
			65	4800	4807.69	– 0.16 %
	8	2.5	130	1200	1201.92	– 0.16 %
			65	2400	2403.85	– 0.16 %
	16	1.25	65	1200	1201.92	– 0.16 %
40 MHz	1	40	2083	1200	1200.19	– 0.02 %
			1042	2400	2399.23	0.03 %
			521	4800	4798.46	0.03 %
			260	9600	9615.38	– 0.16 %
			174	14400	14367.82	0.22 %
			130	19200	19230.77	– 0.16 %
			65	38400	38461.54	– 0.16 %
	2	20	1042	1200	1199.62	0.03 %
			521	2400	2399.23	0.03 %
			260	4800	4807.69	– 0.16 %
			130	9600	9615.38	– 0.16 %
			87	14400	14367.82	0.22 %
			65	19200	19230.77	– 0.16 %
	4	10	521	1200	1199.62	0.03 %
			260	2400	2403.85	– 0.16 %
			130	4800	4807.69	– 0.16 %
			65	9600	9615.38	– 0.16 %
	8	5	260	1200	1201.92	– 0.16 %
			130	2400	2403.85	– 0.16 %
			65	4800	4807.69	– 0.16 %
	16	2.5	130	1200	1201.92	– 0.16 %
			65	2400	2403.85	– 0.16 %

18 Watchdog Timer

18.1 Overview

Use Watchdog Timer to prevent the system from locking-up. For example, you can stop the software program running in an infinite loop by setting interrupts. If the software does not write to the Watchdog during the programmed time, it can either generate an interrupt (WDTOVF) or an internal reset.

18.1.1 Feature

The Watchdog Timer has a programmable 16-bit down counter.

The software can decide what to do when the WDT counter reaches "0" (overflows).

- If the RSTEN bit is set in the WDT_OMR register, then it generates an internal reset.
- If the WDTOVF bit is set in the WDT_IMSCR register, then it generates an interrupt on the Interrupt Controller.

The Input Frequency Clock (FIN) is a clock source (EMCLK, IMCLK, or PLLCLK) from the clock manager. It supplies the Watchdog counter through the WDTPDIV programmable divider.

It is possible to set a programmable pending window where you can restart the Watchdog counter only within this window + 1. This protection is set with the RSTALW bit. If this protection is not there, then you can restart the Watchdog counter whenever it is required. When it reaches the pending window, the WDTPEND bit is set before the PENDING bit.

The WDTPDIV[2:0] divider divides the supplied clock (FIN) and provides to the down-counter input WDTCLK.

To prevent corruption of the Watchdog, control access key protects all write accesses.

Write the correct bit pattern to the control access key bit simultaneously with the control bits (write access) to update the contents of the mode and control registers.

18.2 Functional Description

The functional description section describes the function of Watchdog Timer.

18.2.1 Block Diagram

Figure 18-1 illustrates the Watchdog Timer block diagram.

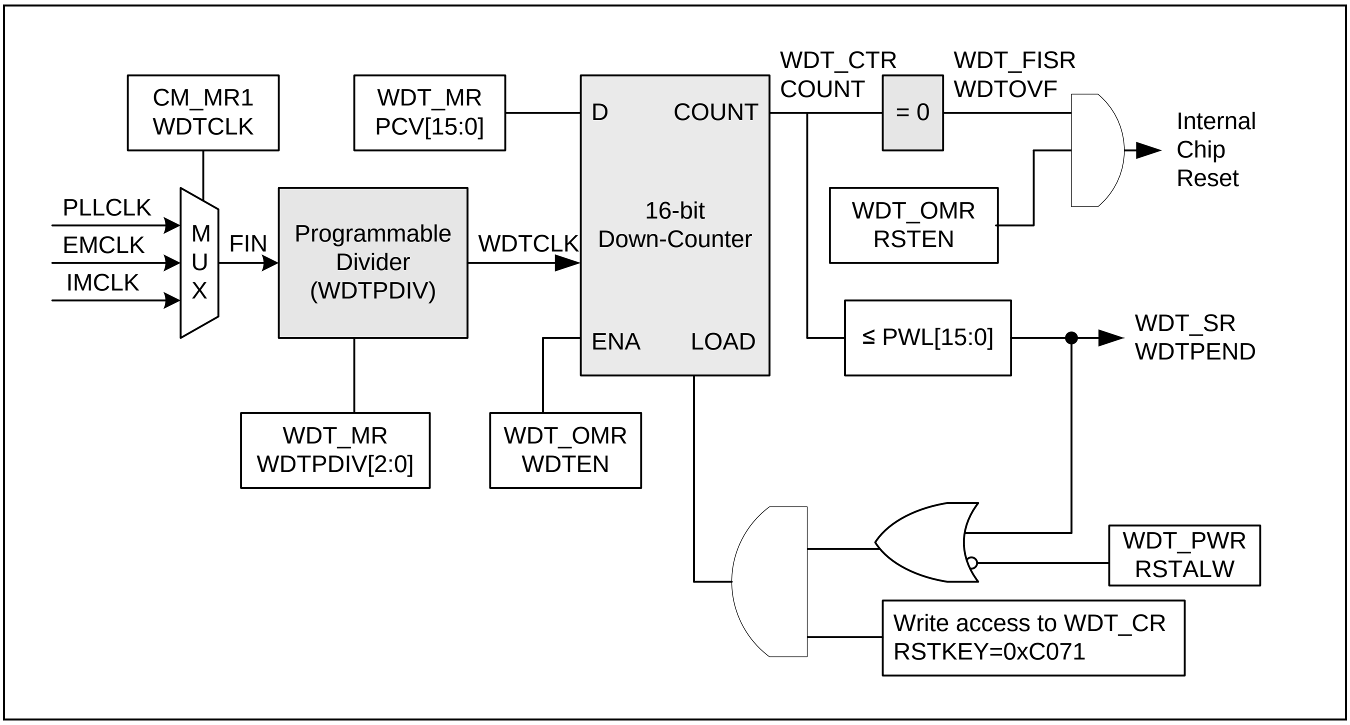


Figure 18-1 Watchdog Timer Block Diagram

18.2.2 Watchdog Timer Functionality

The general description of the Watchdog Timer functionality is described in this section.

18.2.2.1 General Description

The Watchdog Timer contains a programmable length down-counter. The down-counter input clock is a subdivision of the FIN (EMCLK, IMCLK, or PLLCLK) from the clock manager.

NOTE: The count value for overflow should be greater than (3 PCLK + 5 FIN).

- WDTCLK = FIN/WDTPDIV[2:0]

WDTPDIV[2:0]			**WDTCLK**
0	0	0	FIN/2
0	0	1	FIN/4
0	1	0	FIN/8
0	1	1	FIN/16
1	0	0	FIN/64
1	0	1	FIN/128
1	1	0	FIN/256
1	1	1	FIN/512

The count length determines the time-out period. Loading PCV field of WDT_MR register controls the time-out period.

The time-out period (in seconds) is:

- (PCV[15:0] + 1)/WDTCLK freq.

When the counter reaches the value programmed in the pending windows PWL[15:0] of WDT_PWR register, the Watchdog generates a Watchdog pending interrupt.

The pending interrupt occurs after:

- {(PCV[15:0]) – (PWL[15:0])}/WDTCLK freq.

If PWL[15:0] is greater than PCV[15:0] (the previous time is negative), then do not use the WDT pending interrupt.

To prevent an internal chip reset (if RSTEN bit is set in the WDT_OMR) or interrupt (if bit WDTOVF is set in the WDT_IMSCR), the software should be able to reset the counter before it reaches 0 by writing the correct key in the WDT_CR register (0xC071). The time difference (in seconds) between the WDT pending interrupt and the WDT overflow is:

- (PWL[15:0])/WDTCLK freq.

When the counter reaches 0, it triggers the programmed action (internal chip reset or overflow interrupt).

If WDT reset is not programmed, after reaching 0, it is reset to the programmed value and continues to down count. This down count will continue till it disables the WDT. Use this to generate periodic interrupts

18.2.3 Watchdog Timer Events

Watchdog timer events consist of:

- Internal Chip Reset Pulse Generation
- Internal Interrupt Request

18.2.3.1 Internal Chip Reset Pulse Generation

If the RSTEN bit is set in the WDT_OMR register, then it generates an internal system (chip) reset pulse when the overflow occurs. After a reset, the clock selected by the Watchdog Timer is FIN/512.

18.2.3.2 Internal Interrupt Request

The Watchdog generates an Internal Interrupt Request when the overflow occurs. You can enable or disable the state by using the WDT module (WDT_IMSCR register) configuration.

How to use the Watchdog Timer:

Use the windows to generate an interrupt and reload the Watchdog counter within the windows only.
If there is a bug and the interruption is not called, then a reset occurs when the Watchdog counter reaches 0.

18.2.3.3 Configuration

The four types of configurations are:

- Configuration of WDT_MR: Choice of the clock to decrease counter, preload value from where counter starts to decrease.
- Configuration of WDT_ PWR: This is the upper limit of the window from where it generates an interrupt when reached. The bit restarts the counter only within this window.
- Configuration of WDT_IMSCR: It enables Interrupt at the peripheral level when the window is reached upper limit (bit WDTPEND) or when the counter overflows (bit WDTOVF if watchdog reset is not enabled). Then you should configure the interrupt mask.
- Configuration of WDT_OMR: It enables the Watchdog (starts decrementing the counter) and enables the Watchdog reset in a situation when the counter overflows.

18.2.3.4 Interrupt Handing

The procedure for interrupt handling is:

1. IRQ Entry and call C function.
2. Read WDT_MISR and verify the source of the interrupt.
3. Clear the corresponding interrupt at peripheral level by writing in the WDT_ICR.
4. Interrupt treatment: If this is a windows pending interrupt, restart the Watchdog by writing in WDT_CR.
5. Exit IRQ.

18.3 Register Description

18.3.1 Register Map Summary

- Base Address: 0x4003_0000

Register	Offset	Description	Reset Value
WDT_IDR	0x0000	Watchdog ID register	0x0001_0000
WDT_CR	0x0004	Watchdog control register	0x0000_0000
WDT_MR	0x0008	Watchdog mode register	0x00FF_FF07
WDT_OMR	0x000C	Watchdog overflow mode register	0x0000_0003
WDT_SR	0x0010	Watchdog status register	0x8000_0100
WDT_IMSCR	0x0014	Watchdog interrupt mask set /clear register	0x0000_0000
WDT_RISR	0x0018	Watchdog raw interrupt status register	0x0000_0001
WDT_MISR	0x001C	Watchdog masked interrupt status register	0x0000_0000
WDT_ICR	0x0020	Watchdog interrupt clear register	0x0000_0000
WDT_PWR	0x0024	Watchdog pending window register	0x00FF_FF00
WDT_CTR	0x0028	Watchdog counter test register	0x0000_FFFF

18.3.1.1 WDT_IDR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0000, Reset Value = 0x0001_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	R	Reserved	0
IDCODE	[25:0]	R	ID Code Register This field stores the ID code for the corresponding IP.	0x0001_0000

18.3.1.2 WDT_CR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBGEN	RSVD															RSTKEY															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	W	Debug Enable/Disable Control Bit 0 = Disables debug mode. The WDT counter keeps running when a debug is requested. 1 = Enables debug mode. The WDT counter stops running when a debug is requested.	0
RSVD	[7:3]	R	Reserved	0
RSTKEY	[15:0]	W	Restart Key Field 0xC071 = Restarts the Watchdog counter if its value is equal or less than the length of a pending window or if the pending window is disabled. Other values = No effect	0

NOTE: A restart command (write the restart key in WDT_CR) will be effective, if Watchdog counter becomes greater than (2WDTCLK + 1/2 FIN) periods.

18.3.1.3 WDT_MR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0008, Reset Value = 0x00FF_FF07

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKEY								PCV																RSVD					WDTPDIV		
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	1	1	1
W	W	W	W	W	W	W	W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R W	R	R	R	R	R	R W	R W	R W

<table>
<tr><th>Name</th><th>Bit</th><th>Type</th><th colspan="4">Description</th><th>Reset Value</th></tr>
<tr><td>CKEY</td><td>[31:24]</td><td>W</td><td colspan="4">Clock Access Key Field
This is used only when writing in WDT_MR. CKEY is read as 0.
Allows Write access in WDT_MR only if CKEY[7:0] = 0x37.</td><td>0</td></tr>
<tr><td>PCV</td><td>[23:8]</td><td>RW</td><td colspan="4">Preload Counter Value
Preloads the counter when watchdog counter restarts
Time to generate overflow = PCV[15:0]/WDTCLK frequency.</td><td>0xFFFF</td></tr>
<tr><td>RSVD</td><td>[7:3]</td><td>R</td><td colspan="4">Reserved</td><td>0</td></tr>
<tr><td rowspan="11">WDTPDIV</td><td rowspan="11">[2:0]</td><td rowspan="11">RW</td><td colspan="4">WDT Clock Divider Field
This field determines the clock frequency of Watchdog Timer.</td><td rowspan="11">111'b</td></tr>
<tr><td colspan="3"><b>WDPDIV[2:0]</b></td><td><b>WDCLK</b></td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>FIN/2</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>FIN/4</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>FIN/8</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>FIN/16</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>FIN/64</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>FIN/128</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>FIN/256</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>FIN/512</td></tr>
<tr><td colspan="4">NOTE: FIN is an input clock asserted into a Watchdog Timer supplied by clock manager.</td></tr>
</table>

18.3.1.4 WDT_OMR

- Base Address: 0x4003_0000
- Address = Base Address + 0x000C, Reset Value = 0x0000_0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																OKEY												RSVD	LOCKRSTEN	RSTEN	WDTEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W	W	W	W	W	R	R W	R W	R W

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
OKEY	[15:4]	W	Overflow Access Key Field Used only when writing WDT_OMR. OKEY is read as 0. 0x234 = Allows Write access in WDT_OMR. Other values = Does not allow Write access in WDT_OMR.	0
RSVD	[3]	R	Reserved	0
LOCKRSTEN	[2]	RW	CPU Lock-up Reset Enable/Disable Control Bit 0 = Disables the generation of an internal reset on CPU Lock-up 1 = Enables the generation of an internal reset on CPU Lock-up Cortex-M0 supports the flag of CPU Lock-up status and reset signal. Use this in case of a malfunction.	0
RSTEN	[1]	RW	System (Chip) Reset Enable/Disable Control Bit. 0 = Disables the generation of a system (chip) reset by the Watchdog. 1 = Watchdog generates the system (chip) reset on overflow.	1
WDTEN	[0]	RW	Watchdog Enable/Disable Control Bit 0 = Disables Watchdog Timer 1 = Enables Watchdog Timer	1

18.3.1.5 WDT_SR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0010, Reset Value = 0x8000_0100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBGEN	RSVD																					CLEAR_STATUS	PENDING	RSVD							
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
DBGEN	[31]	R	DBGEN Status 0 = Disables debug mode 1 = Enables debug mode	1
RSVD	[30:10]	R	Reserved	0
CLEAR_STATUS	[9]	R	Clear Status 0 = Finishes Watchdog Counter Reset 1 = Watchdog Counter Reset operation starts and does not end.	0
PENDING	[8]	R	Watchdog Pending status 0 = Watchdog counter is more than the pending window length. 1 = Watchdog counter is equal or less than pending window length.	1
RSVD	[7:0]	R	Reserved	0

18.3.1.6 WDT_IMSCR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
WDTOVF	[1]	RW	Watchdog Timer Overflow Interrupt Mask. 0 = Disables the interrupt (This interrupt is masked) 1 = Enables the interrupt (This interrupt is not masked)	0
WDTPEND	[0]	RW	Watchdog Timer Pending Interrupt Mask 0 = Disables the interrupt (This interrupt is masked) 1 = Enables the interrupt (This interrupt is not masked)	0

NOTE: WDT_IMSCR register in the interrupt mask will set or clear register. It is a Read/Write register.
On a Read, this register gives the current value of the mask on the relevant interrupt. A Write of '1' to the particular bit, sets the mask, enabling the interrupt to be read. A Write of '0' clears the corresponding mask.

18.3.1.7 WDT_RISR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0018, Reset Value = 0x0000_0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																														WDTOVF	WDTPEND
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
WDTOVF	[1]	R	Watchdog Overflow interrupt raw state Gives the raw interrupt state (prior to masking) of the WDTOVF interrupt	0
WDTPEND	[0]	R	Watchdog Pending interrupt raw state Gives the raw interrupt state (prior to masking) of the WDTPEND interrupt	1

NOTE: On a Read, WDT_RISR register gives the current raw status value of the corresponding interrupt prior to masking.

18.3.1.8 WDT_MISR

- Base Address: 0x4003_0000
- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																														WDTOVF	WDTPEND
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
WDTOVF	[1]	R	Watchdog Overflow interrupt mask Gives the masked interrupt status (after masking) of the WDTOVF interrupt 0 = The WDTOVF interrupt does not occur 1 = The WDTOVF interrupt occurs	0
WDTPEND	[0]	R	Watchdog Pending masked interrupt Status Gives the masked interrupt status (after masking) of the WDTPEND interrupt 0 = The WDTPEND interrupt does not occur 1 = The WDTPEND interrupt occurs	0

NOTE: On a Read, WDT_MISR register gives the current masked status value of the corresponding interrupt.

18.3.1.9 WDT_ICR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD																														WDTOVF	WDTPEND
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	R	Reserved	0
WDTOVF	[1]	W	Watchdog Overflow Clear 0 = No effect 1 = Clears Watchdog overflow interrupt	0
WDTPEND	[0]	W	Watchdog Pending Clear 0 = No effect 1 = Clears Watchdog pending interrupt	0

18.3.1.10 WDT_PWR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0024, Reset Value = 0x00FF_FF00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PWKEY								PWL																RSVD							RSTALW
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
W	W	W	W	W	W	W	W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Name	Bit	Type	Description	Reset Value
PWKEY	[31:24]	W	Pending window access key This is used only when writing in WDT_PWR. PWKEY is read as 0. Allows Write access in WDT_PWR only if PWKEY[7:0] = 0x91.	0
PWL	[23:8]	RW	Pending Window Length Length of the window. The time difference between preload value and watchdog timer pending is: (PCV[15:0] – PWL[15:0])/WDTCLK freq.	0xFFFF
RSVD	[7:1]	R	Reserved	0
RSTALW	[0]	RW	Restart allowed This bit does not disable the bit WDTPEND in the interrupt register. 0 = Restart allowed every time 1 = Restart allowed only within (pending window +1)	0

18.3.1.11 WDT_CTR

- Base Address: 0x4003_0000
- Address = Base Address + 0x0028, Reset Value = 0x0000_FFFF

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

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	R	Reserved	0
COUNT	[15:0]	R	Watchdog Counter Value of Watchdog timer counter.	0xFFFF

19 Electrical Data

19.1 Overview

The Electrical Data chapter describes all the electrical parameters for S3FN429.

19.2 Absolute Maximum Ratings

Every electrical device will have a rating on its electrical data at which the device will safely operate. The maximum stress that you can apply on the device in terms of voltage and current, above which the device will not operate as desired, is called the Absolute Maximum Rating. Samsung does not define the functional operations of the device at the absolute maximum rating conditions. The document does not explain the functional operations of the device at these conditions. Exposing the device to operate at maximum rating conditions for extended periods will affect the reliability of the device.

Table 19-1 describes the Absolute Maximum Ratings.

Table 19-1 Absolute Maximum Ratings

Parameter	Symbol	Conditions	Rating	Unit
DC Supply Voltage for V_{DDCORE}	V_{DDCORE}	–	– 0.3 to 6.0	V
DC Supply Voltage for I/O	V_{DDIO}	–	– 0.3 to 6.0	
DC Supply Voltage for AV_{DD}	AV_{DD}	–	– 0.3 to 6.0	
DC Supply Voltage for AV_{REF}	AV_{REF}	–	– 0.3 to 6.0	
Digital I/O Input Voltage	V_{IN}	–	– 0.3 to V_{DD_IO} + 0.3	
Analog I/O Input Voltage	AV_{IN}	–	– 0.3 to AV_{DD} + 0.3	
Output Voltage	V_O	All Output Pins	– 0.3 to V_{DD_IO} + 0.3	V
Latch Up Current	I_{LATCH}	–	± 100	mA
Operating Temperature	T_A	–	– 40 to 105	°C
Storage Temperature	T_{STG}	–	– 65 to 150	

NOTE: Do not operate the device above the Absolute Maximum Ratings. Samsung cannot guarantee that the device will operate as desired.

19.3 Recommended Operating Conditions

This semiconductor device will require recommended operating conditions to ensure normal operation. The warranty on electrical characteristics of S3FN429 is valid as long as you operate it according to the recommended operating conditions. If you operate the device beyond the recommended electrical parameter, then it might adversely affect reliability of the device and can lead to a device failure.

Table 19-2 describes the recommended operating conditions at which the device will operate as desired.

Table 19-2 Recommended Operating Conditions

Parameter	Symbol	Conditions	Rating	Unit
DC supply voltage for $V_{DD\ Core}$	V_{DDCORE}	–	2.5 to 5.5	V
DC supply voltage for I/O	V_{DDIO}	–	2.5 to 5.5	V
DC supply voltage for AV_{DD}	AV_{DD}	–	2.5 to 5.5	V
DC supply voltage for AV_{REF}	AV_{REF}	–	2.5 to AV_{DD}	V
Operating temperature	T_A	–	– 40 to 105	°C

19.4 I/O Characteristics

Table 19-3 describes the I/O characteristics for semiconducting device.

Table 19-3 I/O Characteristics

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Operating voltage	V_{DD}	X_{IN} = 1 to 12 MHz, PLLCLK = 40 MHz	2.5	–	5.5	V
Input high voltage	V_{IH1}	All input pins except V_{IH2}	$0.8V_{DD}$	–	V_{DD}	
	V_{IH2}	XIN, MODE[1:0], nRESET	V_{DD} – 0.3	–	V_{DD}	
Input low voltage	V_{IL1}	All input pins except V_{IL2}	–	–	$0.2V_{DD}$	
	V_{IL2}	XIN, MODE[1:0], nRESET	–	–	0.3	
Output high voltage	V_{OH1}	I_{OH} = – 1.6 mA, V_{DD} = 5.0 V	V_{DD} – 0.4	–	–	
	V_{OH2}	I_{OL} = 20 mA, 6 IMC pads, V_{DD} = 5.0 V (PWMU[2:0], PWMD[2:0])	V_{DD} – 1.0	–	–	
Output low voltage	V_{OL1}	I_{OL} = 1.6 mA , V_{DD} = 5.0 V	–	–	0.4	
	V_{OL2}	I_{OL} = 20 mA, 6 IMC pads, V_{DD} = 5.0 V (PWMU[2:0], PWMD[2:0])	–	–	1.0	
Input high leakage current	I_{LIH1}	All input pins except I_{LIH2}, V_{IN} = V_{DD}	–	–	1	µA
	I_{LIH2}	XIN, V_{IN} = V_{DD}	–	–	2	
Input low leakage current	I_{LIL1}	All input pins except I_{LIL2}, V_{IN} = 0	–	–	– 1	
	I_{LIL2}	XIN, V_{IN} = 0	–	–	– 2	
Output high leakage current	I_{LOH}	V_{OUT} = V_{DD}, All output pins	–	–	1	
Output low leakage current	I_{LOL}	V_{OUT} = 0 V, All output pins	–	–	– 1	
I/O pull-up resistor	R_{PU}	All pull-up controllable GPIO Port, V_{DD} = 5 V and V_{IN} = 0 V	10	55	100	kΩ
Feedback resistor	R_{FD}	XIN Pin, V_{IN} = V_{DD} and V_{DD} = 5 V	500	1000	1500	

NOTE: All pins are of Schmitt-trigger type.

19.5 I/O AC Electrical Characteristics

Table 19-5 describes the reset input characteristics for the semiconducting device.

Table 19-4 Reset Input Characteristics

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input maximum operating frequency	IOF_{IN}	All I/O	–	–	30	MHz
Output maximum operating frequency	IOF_{OUT0}	All I/O except IOF_{OUT1}	–	–	8	MHz
Output maximum operating frequency	IOF_{OUT1}	P0.[18:12]	–	–	20	MHz

19.6 Reset Input Characteristics

Table 19-5 describes the reset input characteristics for the semiconducting device.

Table 19-5 Reset Input Characteristics

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Pull-up resistor	R_{NRST}	V_{DD} = 5 V, V_{IN} = 0 V	100	250	400	kΩ
Input low width	T_{NRST}	–	0.8	1.2	2	μs
nRESET schmitt trigger hysteresis	V_{hyst}	Falling and rising edge	–	250	–	mV

NOTE: The noise filter for reset input signal has the distribution like as from 0.8 us to 2 us.
If the reset input signal width is smaller than minimum value (0.8 us), it is always recognized as an invalid signal (non-reset).
If the reset input signal width is greater than maximum value (2 us), it is always recognized as a valid signal (reset).

Figure 19-1 illustrates the input timing diagram for the nRESET pin.

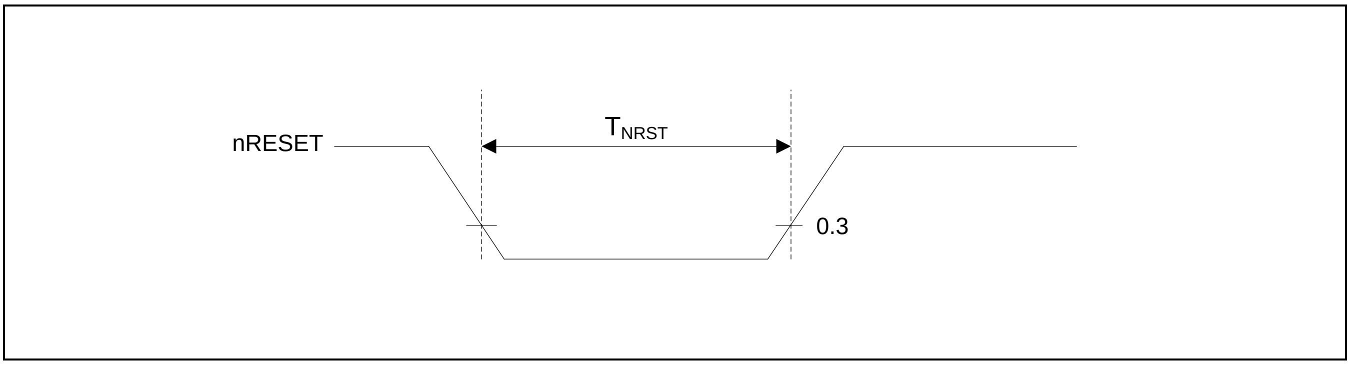


Figure 19-1 Input Timing for nRESET

19.7 External Interrupt Input Characteristics

Table 19-6 describes the external interrupt input characteristics.

Table 19-6 External interrupt Input Characteristics

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Interrupt input high width	t_{INTH}	V_{DD} = 5.0 V	200	300	400	ns
Interrupt input low width	t_{INTL}	V_{DD} = 5.0 V	200	300	400	

NOTE: The noise filter for external signal has the distribution like as from 200 ns to 400 ns.
If the external signal width is smaller than minimum value (200 ns), it is always recognized as an invalid signal.
If the external signal width is greater than maximum value (400 ns), it is always recognized as a valid signal.

Figure 19-2 illustrates the input timing diagram for an external interrupt.

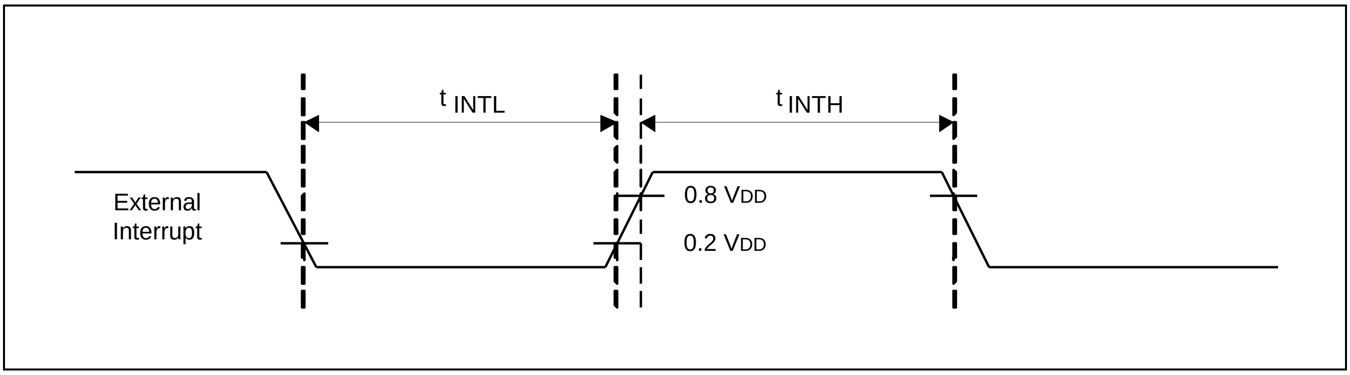


Figure 19-2 Input Timing for External Interrupt

19.8 Oscillator Characteristics

The two kinds of oscillators are:

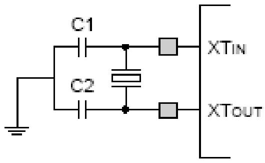
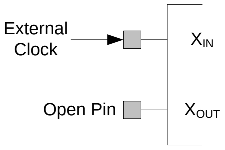
- External Main Oscillator
- Internal Main Oscillator

19.8.1 External Main Oscillator Characteristics

Table 19-7 describes the external main oscillator characteristics.

Table 19-7 External Main Oscillator Characteristics

(T_A = – 40 to 105 °C, $V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD}$ = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Oscillator frequency	f_{EMOSC}	–	1	–	12	MHz
Feedback resistor	R_{FD}	XIN Pin, $V_{IN} = V_{DD}$ and V_{DD} =5 V	500	1000	1500	kΩ
Stabilization time	T_{STA}	–	–	–	10	ms
Crystal/resonator/ ceramic	–	Main oscillation frequency  	1	–	12	MHz
External clock	–	 	1	–	12	

Caution: If you don't the external main-oscillator when pin7 and pin8 are defined XOUT and XIN function, you should tie XIN to ground. XOUT should be opened.

19.8.2 Internal Main Oscillator Characteristics

Table 19-8 describes the internal main oscillator characteristics.

Table 19-8 Internal Main Oscillator Characteristics

(T_A = – 40 to 105 °C, $V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD}$ = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Oscillator frequency	f_{IMOSC}	–	–	40	–	MHz
Output clock duty ratio	T_{OD}	–	40	–	60	%
Accuracy	T_{ACC}	V_{DD} = 5.0 V, T_A = 25 °C	–	–	± 1.5	
		V_{DD} = 5.0 V, T_A = – 40 to 105 °C	–	–	± 3	
		V_{DD} = 2.5 to 5.5 V, T_A = – 40 to 105 °C	–	–	± 5	
Stabilization time	T_{STA}	Internal oscillator stabilization occurs when V_{DD} is equal to the minimum oscillator voltage range.	–	–	10	clock

19.9 Current Consumption

1. The supply current does not include the current drawn through the internal pull-up resistor and the external output current loads.

2. Stop current has two conditions with LVD. When you power LVD up, the band-gap for LVD reference is alive, and then stop current (I_{DD52}) will be increased. To minimize stop current (I_{DD51}), you can enter stop mode after LVD-OFF.

Table 19-9 describes the current consumption details.

Table 19-9 Current Consumption at 5.5 V

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD} = 5.5 V)

Parameter	Symbol	Condition	Mode	Min.	Typ.	Max.	Unit
Supply current	I_{DD11}	Disable PLLCLK Enable EMCLK and IMCLK Enable all peripherals Run CPU at EMCLK = 8 MHz	Normal operating	–	10	18	mA
	I_{DD12}	Disable EMCLK, PLLCLK Enable IMCLK Enable all peripherals Run CPU at IMCLK = 40 MHz	Normal operating	–	15	30	
	I_{DD2}	Enable EMCLK, IMCLK, and PLLCLK Run CPU at PLLCLK = 40 MHz	High-speed operating	–	17	34	
	I_{DD31}	CPU stops in I_{DD11} condition.	Normal idle	–	8	14	
	I_{DD32}	CPU stops in I_{DD12} condition.	Normal idle	–	11	22	
	I_{DD4}	CPU stops in I_{DD2} condition.	High-speed idle	–	12	24	
	I_{DD51}	Disabled EMCLK, PLLCLK, and IMCLK All peripherals stop. LVD OFF	Stop	–	1.2	28	µA
	I_{DD52}	Disabled EMCLK, PLLCLK, and IMCLK All peripherals stop. LVD ON	Stop	–	25	55	

NOTE: Before entering into the Stop Mode, each clock source will have an enabled or disabled status that is set by the user configuration. Stop-current implies that the microcontroller is in the stop-mode. In the stop-mode all clocks are dead automatically.

19.10 PLL Characteristics

Table 19-10 describes the parameters belonging to the PLL characteristics.

Table 19-10 PLL Characteristics

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input frequency	F_{IN}	–	1	–	12	MHz
Output frequency	F_{OUT}	–	12	–	40	
Output clock duty ratio	T_{OD}	–	40	50	60	%
Locking time	T_{LT}	–	–	–	200	µs

19.11 LVD Characteristics

Table 19-11 describes the LVD characteristic parameters.

Table 19-11 LVD Characteristics

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = V_{DDIO} = AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
LVD detect voltage at V_{DD} falling	V_{LVD0}	Reset default	2.3	2.4	2.5	V
	V_{LVD1}	–	2.5	2.6	2.7	
	V_{LVD2}	–	2.6	2.8	3.0	
	V_{LVD3}	–	3.6	3.8	4.0	
	V_{LVD4}	–	4.1	4.3	4.5	
Hysteresis voltage of V_{LVD} (slew rate of LVD)	V_{LVD}	–	80	150	300	mV

NOTE:

1. You can select the level for reset and interrupt voltage using LVDRL[2:0] and LVDIL[2:0] in the CM_MR0 register.
2. LVD hysteresis is only characterization data, and not tested in the mass production. The minimum value of hysteresis is the simulation data.

19.12 12-bit ADC Electrical Characteristics

Table 19-12 describes the electrical characteristic parameters of a 12-bit ADC.

Table 19-12 ADC Characteristics

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = 2.5 to 5.5 V, AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Resolution	–	–	–	12	–	Bit
Supply voltage	V_{ADC}	–	2.5	5	5.5	V
Reference voltage	V_{AVREF}	–	2.5	–	AV_{DD}	
Input voltage range	V_{AIN}	–	0	–	V_{AVREF}	
Clock frequency	F_{ADC}	50 % duty cycle	–	–	5	MHz
Maximum conversion time	t_{ADC}	Max. $F_{ADC,}$ AV_{DD} = AV_{REF}	–	–	1	μs
Differential nonlinearity	DNL	AV_{DD} = AV_{REF} = 2.5 to 5.5 V AV_{SS} = 0.0 V	–	–	± 1.5	LSB
Integral nonlinearity	INL	AV_{DD} = AV_{REF} = 2.5 to 5.5 V AV_{SS} = 0.0 V	–	–	± 3.5	LSB
Offset error (unadjusted) (NOTE)	TOPOFF	AV_{REF} = 4.0 V to 5.5 V	–	–	37.5	mV
		AV_{REF} = 2.5 V to 4.0 V	–	–	56.25	
	BOTOFF	AV_{REF} = 4.0 V to 5.5 V	–	–	37.5	mV
		AV_{REF} = 2.5 V to 4.0 V	–	–	56.25	

NOTE: The values are the characteristic data and not tested during mass production.

19.13 Comparator Electrical Characteristics

Table 19-13 describes the electrical characteristic parameters of a comparator.

Table 19-13 Comparator Electrical Characteristics

(T_A = –40 to 105 °C, V_{DD} = V_{DDCORE} = 2.5 to 5.5 V, AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input offset voltage	V_{IO}	–	–	± 10	± 20	mV
Input common mode voltage range	V_{cm}	–	GND	–	VDD – 0.1	V

NOTE: The parameters are characterized and not tested.

19.14 OP-AMP Electrical Characteristics

Table 19-14 describes the electrical characteristic parameters of an OP-AMP.

Table 19-14 OP-AMP Electrical Characteristics

(T_A = – 40 to 105 °C, V_{DD} = V_{DDCORE} = 2.5 to 5.5 V, AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input offset voltage	$\|V_{IO}\|$	–	–	± 1	–	mV
Input voltage range	V_I	Gain = 2.500	$0.040AV_{DD0}$	–	$0.360AV_{DD0}$	V
		Gain = 2.667	$0.037AV_{DD0}$	–	$0.337AV_{DD0}$	
		Gain = 2.833	$0.035AV_{DD0}$	–	$0.318AV_{DD0}$	
		Gain = 3.000	$0.033AV_{DD0}$	–	$0.300AV_{DD0}$	
		Gain = 3.333	$0.030AV_{DD0}$	–	$0.270AV_{DD0}$	
		Gain = 3.667	$0.027AV_{DD0}$	–	$0.245AV_{DD0}$	
		Gain = 4.000	$0.025AV_{DD0}$	–	$0.225AV_{DD0}$	
		Gain = 4.500	$0.022AV_{DD0}$	–	$0.200AV_{DD0}$	
		Gain = 5.000	$0.020AV_{DD0}$	–	$0.180AV_{DD0}$	
		Gain = 5.667	$0.018AV_{DD0}$		$0.159AV_{DD0}$	
		Gain = 6.667	$0.015AV_{DD0}$		$0.135AV_{DD0}$	
		Gain = 8.000	$0.013AV_{DD0}$		$0.113AV_{DD0}$	
		Gain = 10.00	$0.010AV_{DD0}$	–	$0.090AV_{DD0}$	
		Gain = 12.00	$0.010AV_{DD0}$	–	$0.075AV_{DD0}$	
		Gain = 15.00	$0.010AV_{DD0}$	–	$0.060AV_{DD0}$	
Slew rate	S_R	–	–	15	–	V/µs
Gain error	G_E	Gain = 2.5 to 4.5 at AV_{DD} = 4.5 to 5.5 V	–	± 2.0	± 3.0	%
		Gain = 5 to 6.667 at AV_{DD} = 4.5 to 5.5 V	–	± 2.0	± 6.0	
		Gain = 8, 10, 12, and 15 at AV_{DD} = 4.5 to 5.5 V	–	± 4.0	± 12.0	

NOTE:

1. These characteristic data are only for IP itself.
2. S_R (Slew Rate) parameter is characterized and not tested.

19.15 Flash Memory Characteristics

Table 19-15 describes the flash memory characteristic parameters.

Table 19-15 Flash Memory Characteristics

(T_A = – 40 to 105°C, V_{DD} = V_{DDCORE} = AV_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Total size	Fsize	–	–	32	–	KB
Program size	Fwsize	–	–	4	–	Byte
Page size	Fpsize	–	–	256	–	
Sector size	Fssize	–	–	8	–	KB
Programming time for one word	Ftw	–	20	25	30	µs
Page erase time	Ftpera	–	4	8	12	ms
Sector erase time	Ftsera	–	10	20	28	
Chip erase time	Ftcera	–	32	50	70	
Data access time	Ft	–	–	–	25	MHz
Endurance: number of writing/erasing	Fnwe	–	10,000	–	–	times
Data retention	Ftdr	–	10	–	–	years

19.16 SPI (SSP) Timing Characteristics

Table 19-16 SPI Timing Characteristics

Parameter	Symbol	Min.	Typ.	Max.	Unit
SPI MOSI master output delay time	tSPIMOD	–	–	1	ns
SPI MOSI slave input setup time	tSPISIS	$1/4 \times T_{SPICLKIN} - 2$	–	–	ns
SPI MOSI slave input hold time	tSPISIH	$1/4 \times T_{SPICLKIN} + 2$	–	–	ns
SPI MISO slave output delay time	tSPISOD	–	–	200	ns
SPI MISO master input setup time	tSPIMIS	$1/4 \times T_{SPICLKOUT} - 2$	–	–	ns
SPI MISO master input hold time	tSPIMIH	$1/4 \times T_{SPICLKOUT} + 2$	–	–	ns
SPI nSS master output delay time	tSPICSSD	–	–	$T_{SPICLK} - 0.3$	ns
SPI nSS slave input setup time	tSPICSSS	–	–	$T_{SPICLK} - 0.3$	ns

NOTE:

1. Clock cycle time ($T_{SPICLK} = 1/F_{SPICLK}$):
   $F_{SPICLK} = F_{SCLK}$
   F_{SPICLK} (Min.) $\geq 2 \times F_{SPICLKOUT}$ (Max.) [for master mode]
   F_{SPICLK} (Min.) $\geq 12 \times F_{SPICLKIN}$ (Max.) [for slave mode]
   F_{SPICLK} (Max.) $\leq 254 \times 256$ x $F_{SPICLKOUT}$ (Min.) [for master mode]
   F_{SPICLK} (Max.) $\leq 254 \times 256$ x $F_{SPICLKIN}$ (Min.) [for slave mode]
   $F_{SPICLKOUT} \leq 12$ Mbps [for master mode]
   $F_{SPICLKIN} \leq 3.3$ Mbps [for slave mode]
2. Clock rise/fall time ($T_{SPI\_R\_F}$): Max. = 12 ns with CL = 30 pF.
3. The SPI timing characteristics is not tested during mass production.

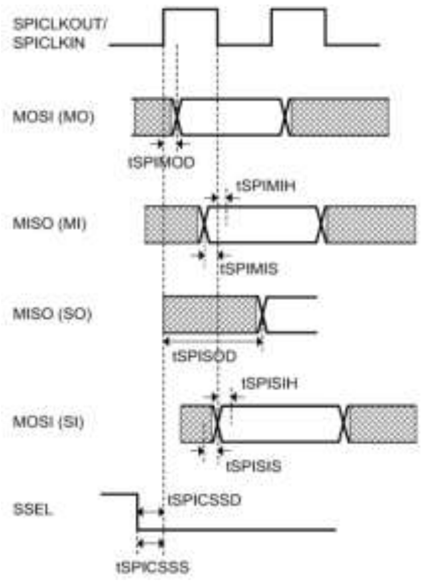


Figure 19-3 SPI Interface Transmit/Receive Timing

19.17 ESD Characteristics

Table 19-17 ESD Characteristics

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Electrostatic discharge	V_{ESD}	HBM	2000	–	–	V
		MM	200	–	–	
		CDM	500	–	–	

20 Package Specification

20.1 Overview

The Package Specification chapter describes the package information available in a 44-QFP-1010 package type.

Table 20-1 describes the package specification information.

Table 20-1 Package Specification Information

Package Number	44-QFP-1010
Package Width × Package Length	10.0 × 10.0 mm
Lead Pitch	0.80 mm

Figure 20-1 illustrates the package outline.

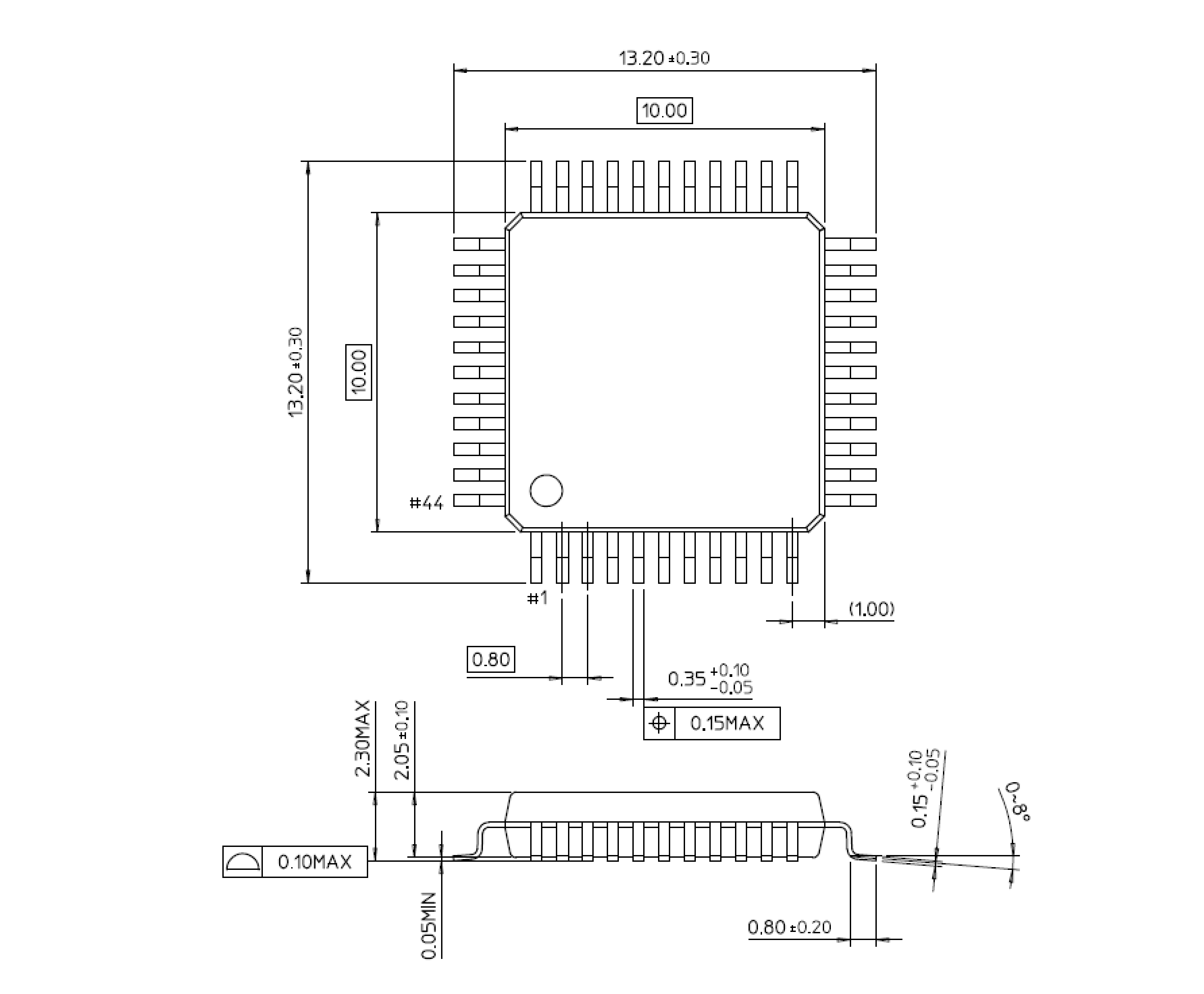


Figure 20-1 44-QFP-1010 Package Dimension