Datasheet

MCU Embedded and Configurable Single-phase Motor Controller FU5821

Fortior Technology (Shenzhen) Co., Ltd

Contents

Explanation of Symbols

- The symbol "[ ]" following a register indicates a bit in the register. For example, ABCD[XY] indicates the XY bit in ABCD register
- [m:n] indicates a range of bits. For example, [3:0] means the bits from bit3 to bit0.
- Pm.n indicates the n^th port of the Portm. For example, P0.0 indicates the 0^th port of Port0.
- Register read and write symbols:
  - R: Read only
  - W: Write only
  - R/W: Read/write
  - W0: Only 0 can be written
  - W1: Only 1 can be written
- The symbol "-" indicates an invalid or uncertainty value.
- The RMW instruction cannot be used for registers with different read and written representations.
- Q (number) format is to store floating-point numbers using fixed-point numbers. MSB is the sign bit, followed by integer bits and fraction bits, where lower Q bits are assigned to the fractional part and the remaining bits are assigned to the integer part. For example, for Q12, bit15 is the sign bit, bit14 ~ bit12 represent the integer part and bit11 ~ bit0 represent the fraction part. The Q12 format has a decimal range -8 ~ 7.9998 (corresponding to 0x8000 ~ 0x7FFF).

Abbreviations

ADC: Analog Digital Convertor

BEMF: Back Electromotive Force

BLDC: Brushless Direct Current

CRC: Cyclic Redundancy Check

DAC: Digital Analog Convertor

DMA: Direct Memory Access

FG: Frequency Generator

FICE: Fortior Interactive Connectivity Establishment

FOC: Field Oriented Control

FOSC: Fast Oscillator

GPIO: General Purpose Input Output

I^{2}C: Inter Integrated Circuit

IC: Integrated Circuit

IRAM: Internal RAM

IDE: Integrated Development Environment

LDO: Low Dropout Regulator

LIN: Local Interconnect Network

LPF: Low Pass Filter

LSB: Least Significant Bit

LVD: Low Voltage Detection

MDU: Multiplication Division Unit

ME: Motor Engine

MSB: Most Significant Bit

MOSFET: Metal Oxide Semiconductor Field Effect Transistor

NC: Not Connected

PGA: Programmable Gain Amplifier

PI/PID: Proportional Integral/Proportional Integral Derivative

PLL: Phase Locked Loop

PWM: Pulse Width Modulation

QEP: Quadrature Encoder Pulse

RAM: Random Access Memory

RMW: Read Modified Write

ROM: Read Only Memory

RSD: Rotating State Detection

RTC: Real Time Clock

SCL: Serial Clock Line

SDA: Serial Data Line

SFR: Special Function Register

SMO: Sliding Mode Observer

SOSC: Slow Oscillator

SPI: Serial Peripheral Interface

SVPWM: Space Vector PWM

TSD: Temperature Sensor Detect

UART: Universal Asynchronous Receiver/Transmitter

WDT: Watch Dog Timer

XRAM: External RAM

XSFR: External SFR

ZCP: Zero Crossing Point

1 System Introduction

1.1 Features

- Power supply: 5V ~ 28V
- Dual core: 8051 core and ME core
- An instruction cycle mostly takes 1 or 2 system clock cycle(s)
- 6kB Flash ROM with CRC, self-program and code protection
- 256 bytes IRAM and 256 bytes XRAM
- ME: Single-phase motor engine
- 15 interrupt sources with 4 configurable priority levels
- GPIO:
  - FU5821T: 8
  - FU5821Q: 12
- Timer:
  - Timer1: Timer supporting square-wave drive timing control, automatic commutation, soft switching, cycle-by-cycle current limiting and motor lock detection
  - Timer2/Timer3/Timer4: Timers supporting PWM output, measurement of duty cycle and period of input PWM wave. Timer3 supports up to 48MHz input.
  - Systick Timer
  - RTC
- Communication interface:
  - 1 I^{2}C
  - 1 UART, supporting reverse input mode, reverse output mode and single-wire mode
- Analogue peripherals:
  - 10-bit ADC, operating with 2μs conversion time and VDD5 selectable as reference voltage
  - Number of ADC channels:
    - FU5821T: 6
    - FU5821Q: 10
    - Internal VCC voltage sampling channel
  - 3 comparator groups (5 comparators)
  - DAC:
    - Single-channel 6-bit DAC for current limiting protection
    - Single-channel 4-bit DAC for over-current protection
- Built-in 2P2N pre-driver
- Automatic phase commutation, cycle-by-cycle current limiting and Hall/BEMF-based position

detection for BLDC motor control

- Built-in oscillator:
  - 24MHz high-speed oscillator
  - 32.8kHz low-speed oscillator
- WDT
- LVD
- TSD
- Two-wire FICE protocol based in-circuit emulation

1.2 Applications

The chip can be used for the drive of Hall-based single-phase BLDC motors.

1.3 Overview

The high-performance motor drive chip incorporates ME core and 8051 core. ME core integrates Smart Engine to complete calculations for high-speed motor control. 8051 core is used for parameter configuration and routine processing. Most of 8051 core instruction cycle takes 1 or 2 clock cycle(s). The dual cores work in parallel to achieve high-performance motor control. The chip integrates high-speed comparators, pre-driver, ADC, CRC, I^{2}C, UART, Timers, PWM and high-speed LDO, which are essential for square wave control of Hall-based single-phase BLDC motors.

Package type of FU5821 include TSSOP16 (FU5821T) and QFN24 (FU5821Q)

1.4 Functional Block Diagram

1.4.1 FU5821T

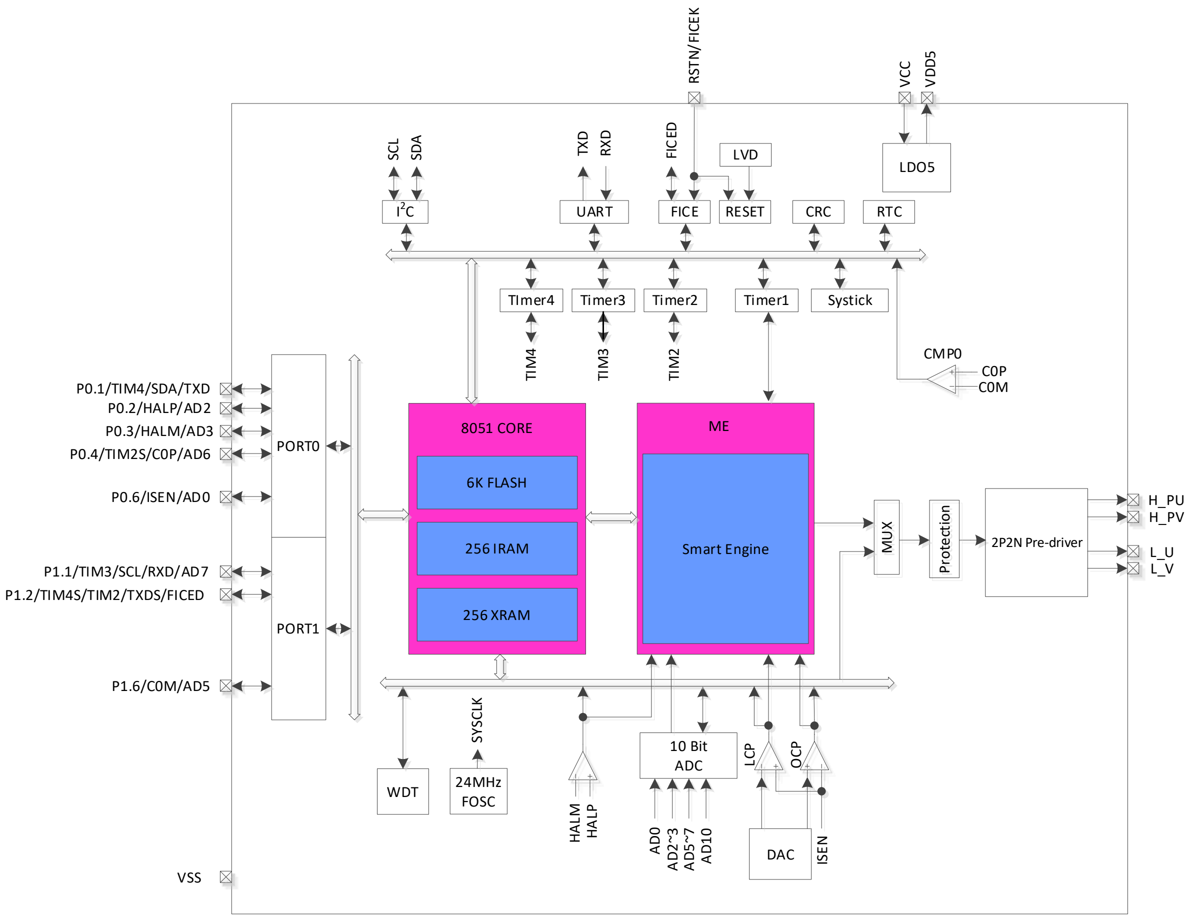


Figure 1-1 Functional Block Diagram of FU5821T

1.4.2 FU5821Q

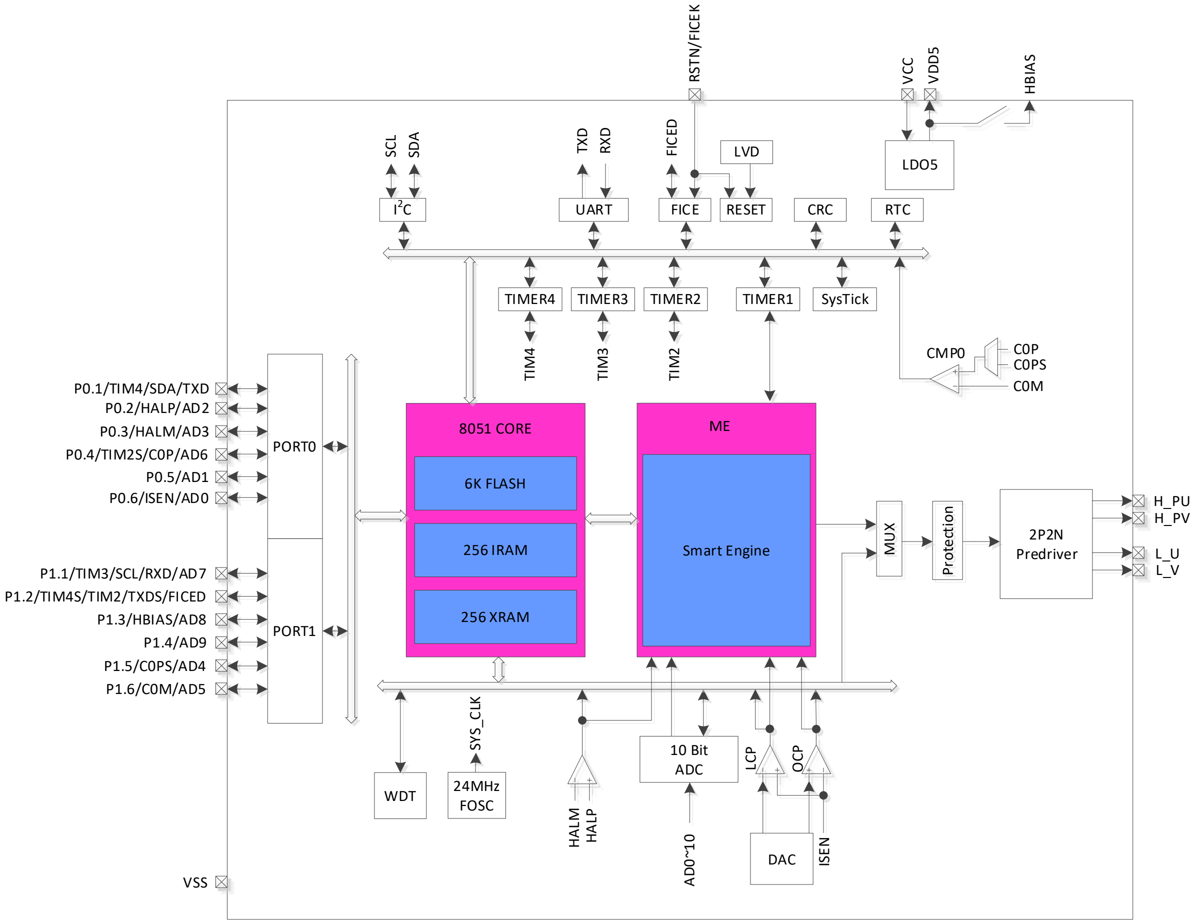


Figure 1-2 Functional Block Diagram of FU5821Q

1.5 Memory Organization

The internal storage space is divided into Program Memory and Data Memory, which are independently addressed.

1.5.1 Program Memory

The chip implements this program memory as Flash memory with a block from addresses 0x0000 to 0x17FF to store the control program.

The first sector (0x0000 ~ 0x007F) is the interrupt vector address area, which is used to store the start address of each interrupt subroutine. The last sector (0x1780 ~ 0x17FF) contains internal control bits of the chip.

1.5.2 Data Memory

The data space is divided into External Data Memory and Internal Data Memory.

The External Data Memory is addressed in the range of 0x0000 ~ 0xFFFF, which can be accessed only with MOVX instructions. It comprises XRAM (0x0000 ~ 0x00FF), extend control register space (0x4020 ~ 0x40FF) and ADC conversion results memory area (0x0100 ~ 0x0115).

The Internal Data Memory is addressed from 0x00 to 0xFF. Locations 0x00 ~ 0x1F are addressable as 4 banks of general purpose registers, each bank consisting of 8 registers, adding up to 32 registers. Locations 0x20 ~ 0x7F are used for low-order RAM memory, supporting direct and indirect addressing. Locations 0x20 ~ 0x2F are 16-bit addressable. When locations 0x80 ~ 0xFF are accessed by indirect addressing, it points to RAM. When locations 0x80 ~ 0xFF are accessed by direct addressing, it points to SFR.

1.5.3 SFR

Table 1-1 SFR Address Mapping

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0xF8	DRV_OUT				P0_OE	P1_OE		
0xF0	B							
0xE8		DRV_DTR						
0xE0	ACC	DRV_CR	DRV_DRL	DRV_DRH	DRV_ARRL	DRV_ARRH	DRV__CNTRL	DRV__CNTRH
0xD8	IP3	CMP_CR2	CMP_CR3	LVSR	TIM1_CR9	TIM1_CR10	TIM1_DBRB	DRV_SR
0xD0	PSW	TIM1_IER	TIM1__BCNTRL	TIM1__BCNTRH	TIM1_SR	CMP_CR0	CMP_CR1	CMP_SR
0xC8	IP2	RST_SR	TIM1__PWMDRL	TIM1__PWMDRH	TIM1__FPWMDDRL	TIM1__FPWM,DDRH	TIM1__RPWMDDRL	TIM1__RPWMDDRH
0xC0	IP1	TIM1_CR8	TIM1__BCORL	TIM1__BCORH	TIM1__SCNTRL	TIM1__SCNTRH	TIM1__SARRL	TIM1__SARRH
0xB8	IP0	TIM1_CR7	TIM1_DBR2	TIM1_DBR3	TIM1__RCNTRL	TIM1__RCNTRH	TIM1__RARRL	TIM1__RARRH
0xB0		TIM1_CR0	TIM1_CR1	TIM1_CR2	TIM1_CR3	TIM1_CR4	TIM1_CR5	TIM1_CR6
0xA8	IE	TIM2_CR1	TIM2__CNTRL	TIM2__CNTRH	TIM2__DRL	TIM2__DRH	TIM2__ARRL	TIM2__ARRH
0xA0		TIM2_CR0	TIM3__CNTRL	TIM3__CNTRH	TIM3__DRL	TIM3__DRH	TIM3__ARRL	TIM3__ARRH
0x98	UT_CR	UT_DR	UT_BAUDL	UT_BAUDH	TIM3_CR0	TIM3_CR1	TIM4_CR0	TIM4_CR1
0x90	P1		TIM4__CNTRL	TIM4__CNTRH	TIM4__DRL	TIM4__DRH	TIM4__ARRL	TIM4__ARRH
0x88	TCON							
0x80	P0	SP	DPL	DPH	FLA_KEY	FLA_CR		PCON

Notes:

- Registers containing the symbol "__" are 16-bit snapshot registers. Snapshot registers are the dynamic registers which shall be read using variables. The value will be incorrect when the register is read directly.
- 8-bit MCU shall read a 16-bit register twice to get the value, the 8 high bits and the 8 low bits respectively. The result will be incorrect when 8 low bits of the register change after MCU has read the 8 high bits. Therefore, when 8 high bits of the snapshot register are read by MCU, the corresponding 8 low bits are stored and read.
- Snapshot register must be read as a whole, the 8 high bits first and then the 8 low bits.

1.5.4 XSFR

Table 1-2 XSFR Address Mapping

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0x4038	ADC_SCYC	ADC_CR	LCP_DR	OCP_DR	PH_SEL		CAL_CR0	CAL_CR1
0x4030	P1_AN	P0_AN		P0_PU	P1_PU		ADC_MASKH	ADC_MASKL
0x4028	I2C_CR	I2C_ID	I2C_DR	I2C_SR	RTC0TMH	RTC0TML	RTC0STA	TSD_CR
0x4020		CRC_DIN	CRC_CR	CRC_DR	CRC_BEG	CRC_CNT	WDT_CR	WDT_REL
0x4018								
0x4010								
0x4008								
0x4000								
0x0078	DBG_DAT0H	DBG_DAT0L	DBG_DAT1H	DBG_DAT1L	DBG_DAT2H	DBG_DAT2L	DBG_DAT3H	DBG_DAT3L
0x0080	ME_TABLE							
0x0088								
0x0090								
0x0098								
0x00a0								
0x00a8								
0x00b0								
0x00b8								
0x00c0								
0x00c8								
0x00d0								
0x00d8								
0x00e0								
0x00e8								
0x00f0								
0x00f8								
0x0100	ADC0_DRH	ADC0_DRL	ADC1_DRH	ADC1_DRL	ADC2_DRH	ADC2_DRL	ADC3_DRH	ADC3_DRL
0x0108	ADC4_DRH	ADC4_DRL	ADC5_DRH	ADC5_DRL	ADC6_DRH	ADC6_DRL	ADC7_DRH	ADC7_DRL
0x0110	ADC8_DRH	ADC8_DRL	ADC9_DRH	ADC9_DRL	ADC10_DRH	ADC10_DRL		

Notes:

- Registers containing the symbol "__" are 16-bit snapshot registers. Snapshot registers are the dynamic registers which shall be read using variables. The value will be incorrect when the register is read directly.
- 8-bit MCU shall read a 16-bit register twice to get the value, the 8 high bits and the 8 low bits respectively. The result will be incorrect when 8 low bits of the register change after MCU has read the 8 high bits. Therefore, when 8 high bits of the snapshot register are read by MCU, the corresponding 8 low bits are stored and read.
- Snapshot register must be read as a whole, the 8 high bits first and then the 8 low bits.

2 Pin Definitions

The IO types are defined as follows:

- DI = Digital Input
- DO = Digital Output
- DB = Digital Bidirectional
- AI = Analogue Input
- AO = Analogue Output
- AB = Analogue Bidirectional
- P = Power Supply

2.1 FU5821T TSSOP16 Pins

Table 2-1 FU5821T TSSOP16 Pins

Pin	FU5821T TSSOP16	IO Type	Description
P0.1/ TIM4/ SDA/ TXD	1	DB/ DB/ DB/ DO	GPIO, configurable as INT0 input, with collector open drain output Timer4 input/output I^{2}C SDA, with collector open drain output UART TXD, with collector open drain output
H_PU	2	DO	2P2N pre-driver U-phase high side output, with built-in 50kΩ pull-up resistor
H_PV	3	DO	2P2N pre-driver V-phase high side output, with built-in 50kΩ pull-up resistor
L_U	4	DO	2P2N pre-driver U-phase low side output, with built-in 25kΩ pull-down resistor
L_V	5	DO	2P2N pre-driver V-phase low side output, with built-in 25kΩ pull-down resistor
VCC	6	P	Power supply, with an external filter capacitor of 2.2μF or larger
VSS	7	P	Ground
VDD5	8	P	Mid-voltage power input or 5V LDO power output, with a 1~4.7μF external capacitor
RSTN/ FICEK	9	DI/ DI	Input of external reset FICE SCL
P1.2/ TIM4S/ TIM2/ TXDS/ FICED	10	DB/ DB/ DB/ DO/ DB	GPIO, configurable as INT0/INT1 input Timer4 input/output after function switching Timer2 input/output UART TXD after function switching FICE SDA
P1.6/ C0M/ AD5	11	DB/ AI/ AI	GPIO, configurable as INT1 input Negative input of CMP0 Input of ADC channel 5
P0.4/ TIM2S/ C0P/ AD6	12	DB/ DB/ AI/ AI	GPIO, configurable as INT0/INT1 input Timer2 input/output after function switching Positive input of CMP0 Input of ADC channel 6
P0.6/ ISEN/ AD0	13	DI/ AI/ AI	Input port P0.6 Bus current detection input Input of ADC channel 0
P0.2/ HALP/ AD2	14	DB/ AI/ AI	GPIO for digital input of Hall-IC Positive input of Hall-sensor Input of ADC channel 2

Pin	FU5821T TSSOP16	IO Type	Description
P0.3/ HALM/ AD3	15	DB/ AI/ AI	GPIO Negative input of Hall-sensor Input of ADC channel 3
P1.1/ TIM3/ SCL/ RXD/ AD7	16	DB/ DB/ DB/ DB/ AI	GPIO, configurable as INT0/INT1 input, with configurable pull-up or pull-down resistor Timer3 input/output I^{2}C SCL, with collector open drain output UART RXD input in two-wire mode or TXD output/RXD input in single-wire mode Input of ADC channel 7

2.2 FU5821T TSSOP16 Pinout Diagram

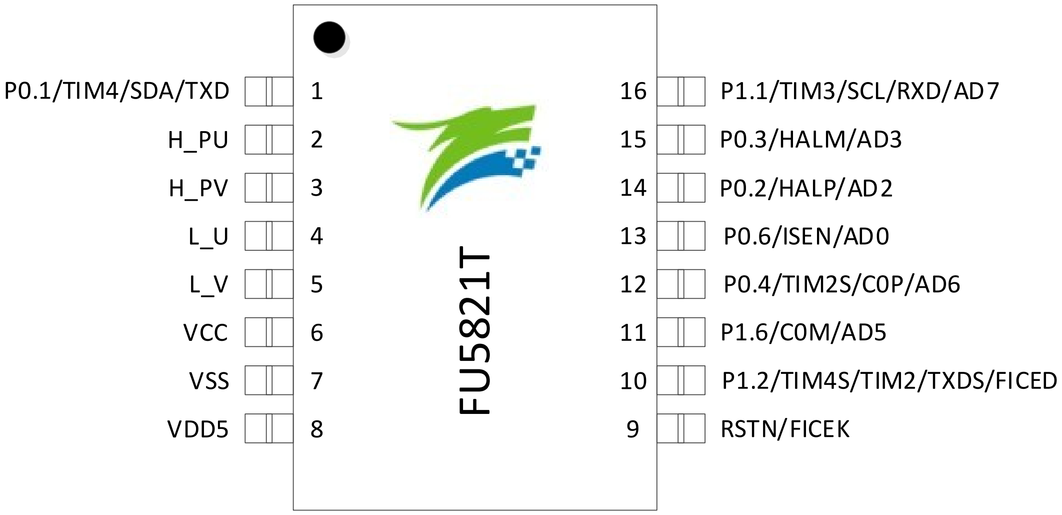


Figure 2-1 FU5821T TSSOP16 Pinout Diagram

2.3 FU5821Q QFN24 Pins

Table 2-2 FU5821Q QFN24 Pins

Pin	FU5821Q QFN24	IO Type	Description
P0.3/ HALM/ AD3	1	DB/ AI/ AI	GPIO P0.3, configurable as INT0 input, Negative input of Hall-sensor Input of ADC channel 3
P1.1/ TIM3/ SCL/ RXD/ AD7	2	DI/ DI/ DB/ DB/ AI	GPIO P1.1, configurable as INT0/INT1 input, with configurable pull-up or pull-down resistor Timer3 input capture mode I^{2}C SCL, with collector open drain output UART RXD input in two-wire mode or TXD output/RXD input in single-wire mode Input of ADC channel 7
P0.1/ TIM4/ SDA/ TXD	3	DB/ DO/ DB/ DO	GPIO P0.1, configurable as INT0 input, with collector open drain output Timer4 output, with collector open drain output I^{2}C SDA, with collector open drain output UART TXD output
NC	4		Not connected
H_PU	5	DO	2P2N pre-driver U-phase high side output, with built-in pull-up resistor
H_PV	6	DO	2P2N pre-driver V-phase high side output, with built-in pull-up resistor
NC	7		Not connected
L_U	8	DO	2P2N pre-driver U-phase low side output, with built-in pull-down resistor
L_V	9	DO	2P2N pre-driver V-phase low side output, with built-in pull-down resistor
NC	10		Not connected
VCC	11	P	Power supply, with an external filter capacitor of 4.7μF or larger
VSS	12	P	Ground
VDD5	13	P	5V LDO power output, with a 1~4.7μF/10V external capacitor
RSTN/ FICEK	14	DI/ DI	Input of external reset, with built-in pull-up resistor FICE SCL
P1.2/ TIM4S/ TIM2/ TXDS/ FICED	15	DB/ DO/ DB/ DO/ DB	GPIO P1.2, configurable as INT0/INT1 input Timer4 input/output after function switching Timer2 input capture mode or PWM output UART TXD after function switching FICE SDA
P1.3/ HBIAS/ AD8	16	DB/ DO/ AI	GPIO P1.3, configurable as INT1 input Hall bias power supply, internally connected to VDD5 via a switch Input of ADC channel 8
P1.4/ AD9	17	DB/ AI	GPIO P1.4, configurable as INT0/INT1 input Input of ADC channel 9
P1.5/ C0PS/ AD4	18	DB/ AI/ AI	GPIO P1.5, configurable as INT0/INT1 input CMP0 positive input after function switching Input of ADC channel 4
P1.6/ C0M/ AD5	19	DB/ AI/ AI	GPIO P1.6, configurable as INT1 input Negative input of CMP0 Input of ADC channel 5
P0.4/ TIM2S/ C0P/ AD6	20	DB/ DB/ AI/ AI	GPIO P0.4, configurable as INT0/INT1 input Timer2 input capture mode or PWM output after function switching Positive input of CMP0 Input of ADC channel 6
P0.5/ AD1	21	DB/ AI	GPIO P0.5, configurable as INT0/INT1 input Input of ADC channel 1

Pin	FU5821Q QFN24	IO Type	Description
P0.6/ ISEN/ AD0	22	DI/ AI/ AI	Input port P0.6 Bus current detection input Input of ADC channel 0
NC	23		Not connected
P0.2/ HALP/ AD2	24	DB/ AI/ AI	GPIO P0.2 for digital input of Hall-IC Positive input of Hall-sensor Input of ADC channel 2

2.4 FU5821Q QFN24 Pinout Diagram

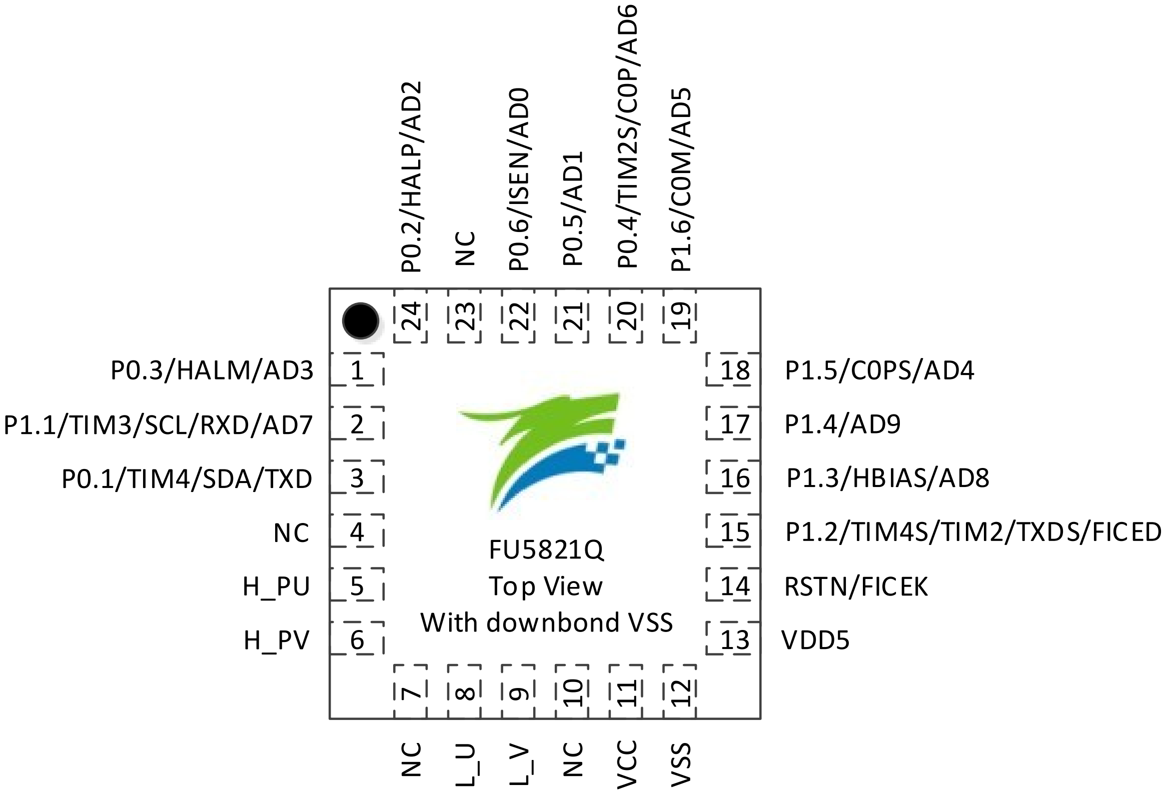


Figure 2-2 FU5821Q QFN24 Pinout Diagram

3 Package Information

3.1 FU5821T TSSOP16_4.4x5.0

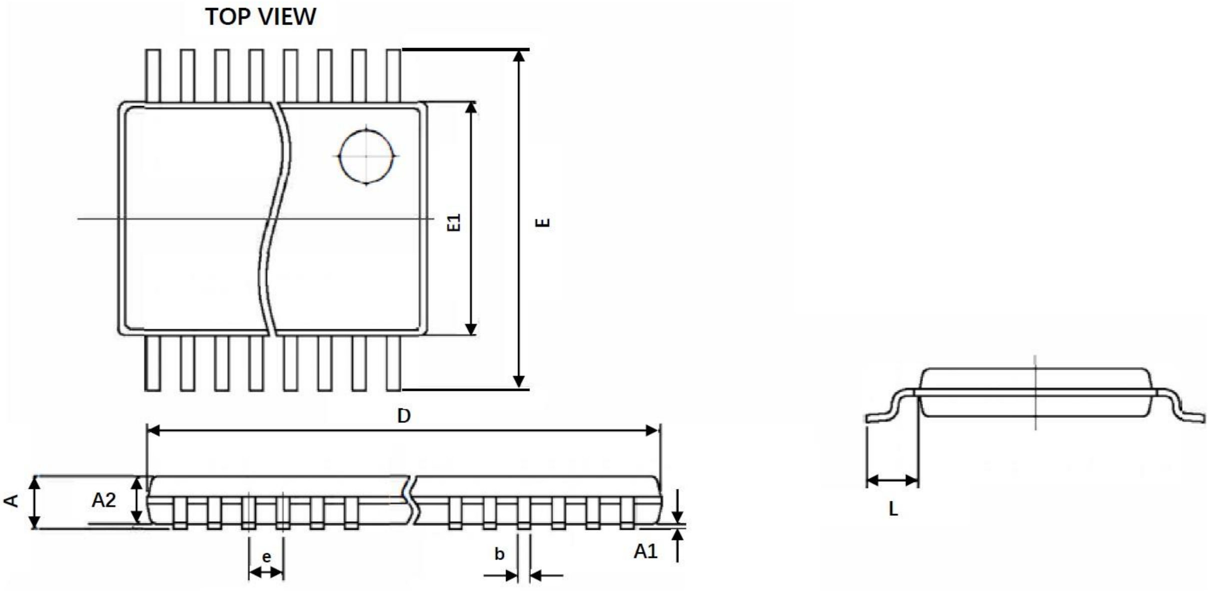


Figure 3-1 Package Drawings of FU5821T TSSOP16_4.4x5.0

Table 3-1 Package Dimensions of FU5821T TSSOP16_4.4x5.0

Symbol	Dimensions In Millimeter	
	Min.	Max.
A	-	1.20
A1	0.05	0.15
A2	0.80	1.05
E	6.20	6.60
E1	4.30	4.50
D	4.90	5.10
L	-	1.00
e	0.65	
b	0.19	0.30

3.2 FU5821Q QFN24_4x4

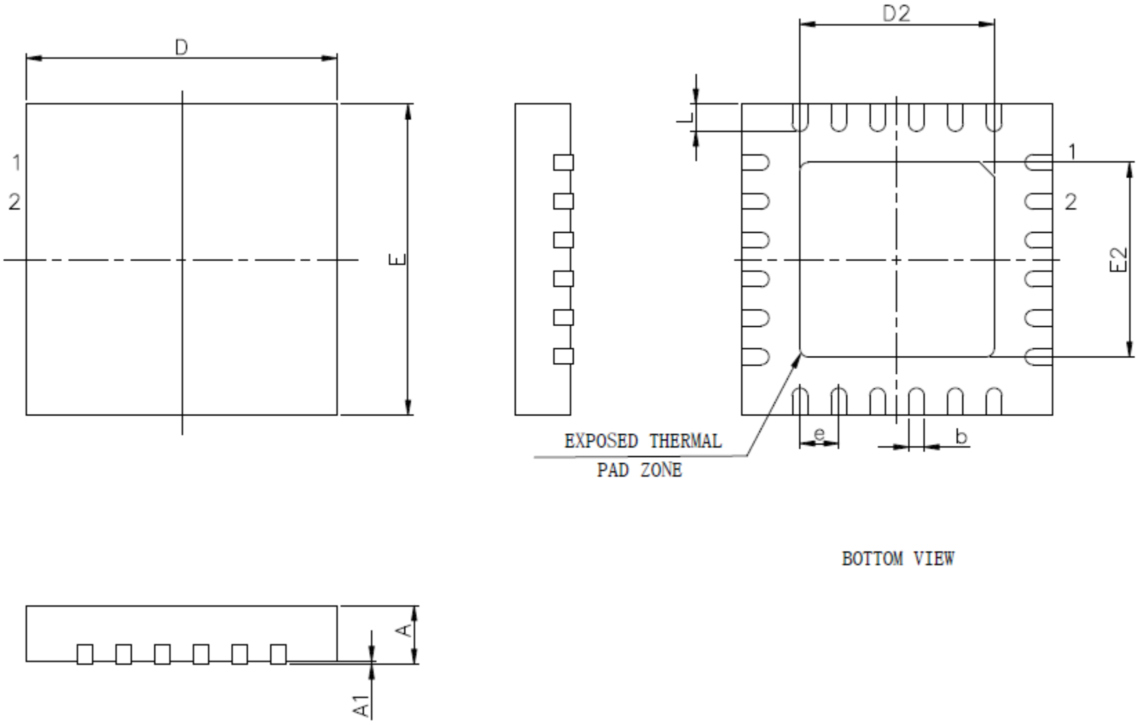


Figure 3-2 Package Drawings of FU5821Q QFN24_4x4

Table 3-2 Package Dimensions of FU5821Q QFN24_4x4

Symbol	Dimensions In Millimeter		
	Min.	Nom.	Max.
A	0.70	-	0.90
A1	0	-	0.05
D	3.90	4.00	4.10
E	3.90	4.00	4.10
D2	2.20	-	2.60
E2	2.20	-	2.60
L	0.25	-	0.50
e	-	0.50	-
b	0.18	-	0.30

4 Ordering Information

Table 4-1 Model Selections

Model	MIPS (Peak)	FLASH (kByte)	XRAM (Byte)	Clock Circuit				Drive Interface			Drive Type			I^2C/UART	DMA	GPIO	Timer	Analogue Peripheral								Lead-free	Package
																		ADC			DAC		VREF	Operational Amplifier	Comparator		
				Internal Fast Clock	External Fast Clock	Internal Slow Clock	External Slow Clock	N+N Pre-driver	P+N Pre-driver	PWM	Soft Switch	Table Look-up	EMC Mode					Number	Channel	Bit	Number	Bit					
FU5821Q	24	6	256	√	—	√	—	—	√	—	√	√	√	√	—	12	5	1	10	10	2	6/4	—	—	5	√	QFN24 (4x4 mm)
FU5821T	24	6	256	√	—	√	—	—	√	—	√	√	√	√	—	8	5	1	6	10	2	6/4	—	—	5	√	TSSOP20 (4.4x5.0mm)

5 Electrical Characteristics

5.1 Absolute Maximum Ratings

Table 5-1 Absolute Maximum Ratings

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Operating Ambient Temperature T_A	VCC ≤ 28V	–40	—	85	°C
	VCC < 15V	–40	—	105	°C
Operating Junction Temperature T_J		–40	—	150	°C
Storage Temperature		–55	—	150	°C
VCC to VSS Voltage		–0.3	—	40	V
VDD5 to VSS Voltage		–0.3	—	6.5	V
RSTN/GPIO to VSS Voltage		–0.3	—	VDD5 + 0.3	V

Note: Stress values greater than the "Absolute Maximum Ratings" listed in Table 5-1 may cause irremediable damages to the device. These are stress ratings only, and it is NOT recommended to use your device in conditions that go beyond these stress ratings. Exposure to "Absolute Maximum Ratings" for extended periods may affect device reliability.

5.2 Global Electrical Characteristics

Table 5-2 Global Electrical Characteristics

(T_A = 25°C and VCC = 5V ~ 28V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
VCC Operating Voltage[1]	High-voltage single-power mode	5	—	28	V
VDD5 Operating Voltage	VCC pin is connected with VDD5 pin.[2]	—	—	5.5	V
System Clock		—	24	—	MHz
I_{VCC} Operating Current[3]		—	12	—	mA
I_{VCC} Standby Current[3]		—	1.5	—	mA
I_{VCC} Sleep-mode Current		—	100	—	μA

Notes:

[1] VCC voltage rise rate ranges from 0.5V/μs to 0.1V/s depending on samples batches;

[2] VDD5 must be in the range of 5~5.5V during Flash write or erase.

[3] Characteristics may vary with different configurations.

5.3 GPIO Electrical Characteristics

Table 5-3 GPIO Electrical Characteristics

(T_A = 25°C and VCC = 12V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Output Rise Time	50pF Load, from 10% to 90%, T_A = 25°C	—	15	—	ns
Output Fall Time	50pF Load, from 90% to 10%, T_A = 25°C	—	13	—	ns
V_{OH} Output High Voltage	I_{OH} = 4mA	VDD - 0.7	—	—	V
V_{OL} Output Low Voltage	I_{OL} = 8mA	—	—	0.7	V
V_{IH} Input High Voltage[1]		0.7*VDD5	—	—	V
V_{IL} Input Low Voltage		—	—	0.2*VDD5	V
Pull-up Resistor[2]		—	35	—	kΩ
Pull-up Resistor[3]		—	6	—	kΩ
Pull-down Resistor[4]		—	15	—	kΩ

Notes:

[1] When VDD5 = 5V, minimum value of V_{IH} is 0.6*VDD5.

[2] GPIOs except P0[3:1].

[3] P0[3:2]

[4] P1[1]

5.4 Pre-driver IO Electrical Characteristics

Table 5-4 Pre-driver IO Electrical Characteristics

(T_A = 25°C and VCC = 15V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
High-side Output Source Current		—	150	—	mA
High-side Output Sink Current		—	90	—	mA
Low-side Output Source Current		—	150	—	mA
Low-side Output Sink Current		—	180	—	mA
High-side Output Rise Time	1nF load, from 10% to 90%	—	85	—	ns
High-side Output Fall Time	1nF load, from 90% to 10%	—	140	—	ns
Low-side Output Rise Time	1nF load, from 10% to 90%	—	85	—	ns
Low-side Output Fall Time	1nF load, from 90% to 10%	—	55	—	ns
V_{OH_HS} High-side Output High Level	VCC=5~ 28V	—	VCC	—	V
V_{OL_HS} High-side Output Low Level	VCC=13~ 28V	—	VCC-11	—	V
	VCC=5~ 12V	—	1.2	—	V
V_{OH_LS} Low-side Output High Level	VCC=13~ 28V	—	11	—	V
	VCC=5~ 12V	—	VCC-1.2	—	V
V_{OL_LS} Low-side Output Low Level	VCC=5~ 28V	—	0	—	V

5.5 OSC Electrical Characteristics

Table 5-5 OSC Electrical Characteristics

(T_A = –40°C ~ 85°C and VCC = 5V ~ 28V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Internal Fast Clock Frequency		23.5	24	24.5	MHz
WDT Clock Frequency		29	32.8	37	kHz

Note: SYSCLK refers to system clock rate, and T to system clock cycle. Unless otherwise specified, the system clock rate of chip is 24MHz and T = 1/SYSCLK.

5.6 Reset Electrical Characteristics

Table 5-6 Reset Electrical Characteristics

(T_A = 25°C and VCC = 5 ~ 28V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Minimum Time for RSTN Released to Low		50	—	—	μs

5.7 LDO Electrical Characteristics

Table 5-7 LDO Electrical Characteristics

(T_A = 25°C and VCC = 5V ~ 28V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
VDD5 Voltage	VCC = 7 ~ 28V	4.7	5	5.3	V

5.8 Package Thermal Characteristics

Table 5-8 FU5821T TSSOP16 Package Thermal Resistance

Parameter	Test Conditions	Value	Unit
Junction-to-ambient Temperature Thermal Resistance θ_{JA} [1]	JEDEC standard, 2S2P PCB	139	°C/W
Junction-to-case Temperature Thermal Resistance θ_{JC}[1]	JEDEC standard, 1S0P PCB	43	°C/W

Table 5-9 FU5821Q QFN24 Package Thermal Resistance

Parameter	Test Conditions	Value	Unit
Junction-to-ambient Temperature Thermal Resistance θ_{JA}[1]	JEDEC standard, 2S2P PCB	50	°C/W
Junction-to-case Temperature Thermal Resistance θ_{JC}[1]	JEDEC standard, 2S2P PCB	25	°C/W

Note:

[1] The actual measurements may vary depending on the conditions.

6 Reset Control

6.1 Reset Sources (RST_SR)

The chip includes a reset circuitry with 7 reset sources:

- Power on reset (RSTPOW)
- External reset (RSTEXT)
- Low voltage detector reset (RSTLVD)
- Watchdog timer reset (RSTWDT)
- Flash error detector reset (RSTFED)
- Debug reset (RSTDBG)
- Soft reset (SOFTR)

Reset flag bits can be searched from register RST_SR. Following the last reset, the affected reset flag is set to “1” and all other reset flags are cleared to “0”. To clear a reset flag, you can set RST_SR[RSTCLR] to “1” so that RST_SR[7:3] & RST_SR[0] are cleared. After reset, MCU restarts the program from address 0.

6.2 Reset Enable

See the corresponding control registers.

6.3 External Reset and Power-on Reset

The chip resets when RSTN pin remains low for 50µs.

The chip resets when the chip is powered on and VDD5 settles above the reset voltage threshold V_{RST}.

6.4 Low Voltage Detector Reset

The chip’s internal circuitry monitors VDD. When VDD drops to a level below V_{RST}, the internal monitor circuitry sends a LVD reset signal to reset the chip.

Configuring corresponding register enables VDD monitor circuitry and sets V_{RST}.

6.5 Watchdog Timer Reset

After the watchdog timer (WDT) is enabled, the software periodically writes 1 to WDT_CR[WDTRF] which initializes watchdog up counter. When watchdog up counter reaches its maximum value, WDT generates an output pulse to reset the chip, which ensures the software runs normally.

6.6 Flash Error Detector Reset

The Flash memory can be programmed by software using the MOVX instruction for read/write/erase operations. A Flash error detector reset (RSTFED) occurs if a Flash erase is attempted targeting the last sector (0x1780 ~ 0x17FF) or a Flash write is attempted targeting the last byte (0x17FF). RSTFED is always enabled and cannot be disabled.

6.7 Debug Reset

Click **Reset** button of IDE to send a Debug reset signal when the chip enters the debug state.

6.8 Soft Reset

The chip resets immediately when RST_SR[SOFTR] is set to “1”. After reset, the flag RST_SR[SOFTR] is set to “1”.

6.9 Reset Register

6.9.1 RST_SR (0xC9)

Bit	7	6	5	4	3	2	1	0
Name	RSTPOW/ RSTCLR	RSTEXT	RSTLVD	RSV	RSTWDT	RSTFED	RSTDBG	SOFTR
Type	R/W1	R	R	—	R	R	R	R/W1
Reset	—	—	—	—	—	—	—	—

Bit	Name	Description
[7]	RSTPOW/ RSTCLR	Power-on Reset Flag Read: 0: Last reset was not a power on reset. 1: Last reset was a power on reset. Write: 0: No effect 1: RST_SR[7:3] & RST_SR[0] are cleared to “0”.
[6]	RSTEXT	External RST Pin Reset flag 0: Last reset was not an RST pin reset. 1: Last reset was an RST reset.
[5]	RSTLVD	Low Voltage Detector (LVD) Reset Flag 0: Last reset was not an LVD reset 1: Last reset was an LVD reset
[4]	RSV	Reserved
[3]	RSTWDT	WDT Reset Flag 0: Last reset was not a WDT reset 1: Last reset was a WDT reset
[2]	RSTFED	Flash Error Detector Reset Flag 0: Last reset was not a Flash error detector reset 1: Last reset was a Flash error detector reset
[1]	RSTDBG	Debug Reset Flag 0: Last reset was not a debug reset 1: Last reset was a debug reset
[0]	SOFTR	Soft Reset Flag Read: 0: Last reset was not a soft reset 1: Last reset was a soft reset Write: 0: No effect 1: A soft reset is generated

7 Interrupt

7.1 Interrupt Introduction

The chip supports 15 interrupt sources. Each interrupt source can be individually programmed in IP0 ~ IP3 registers with one of four priority levels. Interrupt flags are located in an SFR or XSFR. The corresponding interrupt flag is set by the hardware to "1" when the internal circuitry or an external signal meets the interrupt conditions. If IE[EA] = 1 and both the associated interrupt EA and IF bits are set to "1", an interrupt request is sent to CPU. If no other interrupt service routine (ISR) of greater priority is currently being serviced, the system enters interrupt state to service the requesting ISR.

Each interrupt source except the Reset Interrupt can be assigned a priority level. A low priority interrupt can be preempted by a high priority interrupt. The low priority interrupt will not be serviced until the ISR for the high priority interrupt completes. An interrupt will not be preempted by another of the same priority level. Each interrupt source can be individually configured to one of four priority levels in the Interrupt Priority (IP) register. Priority level assigned ascends from 0 to 3 and is defaulted to 0. If two interrupt requests are generated at the same time, the interrupt with the higher priority is serviced first. If two interrupt sources have the same priority, a fixed priority order is used to arbitrate. See Table 7-1 for the interrupt sources and default priority orders, where the lower the mark the higher the priority level.

7.2 Interrupt Enable

IE[EA] is the global interrupt enable bit. When EA = 0, the system does not respond to any interrupt request.

Each interrupt can be individually enabled or disabled by configuring the corresponding interrupt enable bit in an SFR or XSFR. When the enable bit of the global interrupt or an interrupt is cleared, the interrupt flag that is set to "1" is held in a pending state. The MCU will immediately enter the interrupt subroutine once the enable bit is set to "1". Therefore, make sure to clear corresponding interrupt flag bit before enabling the interrupt.

7.3 External Interrupt

The external interrupt has 2 interrupt sources: INT0 and INT1.

The digital input signals from P0.1, P0.3 ~ P0.5, P1.1 ~ 1.2 and P1.4 ~ 1.5 can be used to trigger INT0. The interrupt source is selected through LVSR[EXT0CFG] bit. These interrupt sources share one interrupt entry, one interrupt flag bit TCON[IF0] and one interrupt enable bit IE[EX0]. TCON[IT0] bit selects the interrupt edge. IP0[PX0] bit configures the priority level.

The digital input signals from P1.1 ~ 1.6 and P0.4 ~ 0.5 can be used to trigger INT1. The interrupt source is selected through LVSR[EXT1CFG]. These interrupt sources share one interrupt entry, one interrupt flag bit TCON[IF1] and one interrupt enable bit IE[EX1]. TCON[IT1] bit selects the interrupt edge.

7.4 Interrupt Summary

Table 7-1 Interrupt Summary

Interrupt Source	Priority Order	Vector Address	Interrupt Flag	Cleared by Software?	Interrupt Enable Bit	Priority Control
Reset	Highest	0x0000	None	No	Always Enable	Highest
LVW Interrupt/ TSD Interrupt	0	0x0003	LVSR[0]/ TCON[5]	Yes	CCFG1[6]/ IE[1]	IP0[1:0]
INT0	1	0x000B	TCON[2]	Yes	IE[0]	IP0[3:2]
INT1	2	0x0013	TCON[7]	Yes	IE[2]	IP0[5:4]
DRV Compare Match Interrupt	3	0x001B	DRV_SR[4]	Yes	DRV_SR[1:0]	IP0[7:6]
Timer2 Interrupt	4	0x0023	TIM2_CR1[7:5]	Yes	TIM2_CR1[4:3] TIM2_CR0[3]	IP1[1:0]
Timer1 Interrupt	5	0x002B	TIM1_SR[6:0]	Yes	TIM_IER[5:0]	IP1[3:2]
ADC Interrupt	6	0x0033	ADC_CR[0]	Yes	ADC_CR[1]	IP1[5:4]
LOCP Interrupt	7	0x003B	CMP_SR[6:5]	Yes	CMP_CR1[5:4] CMP_CR2[5:4]	IP1[7:6]
RTC Interrupt	8	0x0043	RTC_STA[6]	Yes	IE[6]	IP2[1:0]
Timer3 Interrupt	9	0x004B	TIM3_CR1[7:5]	Yes	TIM3_CR1[4:3] TIM3_CR0[3]	IP2[3:2]
Systick Interrupt	10	0x0053	DRV_SR[7]	Yes	DRV_SR[6]	IP2[5:4]
Timer4 Interrupt	11	0x005B	TIM4_CR1[7:5]	Yes	TIM4_CR1[4:3] TIM4_CR0[3]	IP2[7:6]
CMP0 Interrupt	12	0x0063	CMP_SR[4]	Yes	CMP_CR0[5:4]	IP3[1:0]
I2C Interrupt	13	0x006B	I2C_SR[0]	Yes	I2C_CR[0]	IP3[3:2]
UART Interrupt	14	0x0073	UT_CR[1:0]	Yes	IE[4]	IP3[5:4]

Notes:

- UT_CR[RI] and UT_CR[TI] bits can be cleared to “0” or set by software to “1”; and MCU generates an interrupt request when these flags are set to “1”. Other flags can be cleared to “0” by software only, and setting them to “1” has no effect.
- For registers containing serval interrupt flags, you can write a “1” to the active interrupt flags in order to prevent clearing a interrupt flag to “0”. For example, use code DRV_SR = (DRV_SR & 0x7F) | 0x10 to clear DRV_SR[SYSTIF].

7.5 Interrupt Registers

7.5.1 IE (0xA8)

Bit	7	6	5	4	3	2	1	0
Name	EA	RTCIE	RSV	ES0	RSV	EX1	TSDIE	EX0
Type	R/W	R/W	—	R/W	—	R/W	R/W	R/W
Reset	0	0	—	0	—	0	0	0

Bit	Name	Description
[7]	EA	All Interrupts Enable 0: Disable 1: Enable
[6]	RTCIE	RTC Interrupt Enable 0: Disable 1: Enable
[5]	RSV	Reserved
[4]	ES0	UART Interrupt Enable 0: Disable 1: Enable
[3]	RSV	Reserved
[2]	EX1	INT1 Interrupt Enable 0: Disable 1: Enable
[1]	TSDIE	TSD Interrupt Enable 0: Disable 1: Enable
[0]	EX0	INT0 Interrupt Enable 0: Disable 1: Enable

7.5.2 IP0 (0xB8)

Bit	7	6	5	4	3	2	1	0
Name	PDRV		PX1		PX0		PLVW_TSD	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	PDRV	DRV Comparison Match Interrupt Priority Control
[5:4]	PX1	INT1 Interrupt Priority Control
[3:2]	PX0	INT0 Interrupt Priority Control
[1:0]	PLVW_TSD	LVW/TSD Interrupt Priority Control

Note: Priority level assigned ascends from 0 to 3, totaling 4 levels

7.5.3 IP1 (0xC0)

Bit	7	6	5	4	3	2	1	0
Name	PLOCP		PADC		PTIM1		PTIM2	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	PLOCP	LOCP Interrupt Priority Control
[5:4]	PADC	ADC Interrupt Priority Control
[3:2]	PTIM1	Timer1 Interrupt Priority Control
[1:0]	PTIM2	Timer2 Interrupt Priority Control

Note: Priority level assigned ascends from 0 to 3, totaling 4 levels

7.5.4 IP2 (0xC8)

Bit	7	6	5	4	3	2	1	0
Name	PTIM4		PSTIC		PTIM3		PRTC	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	PTIM4	Timer4 Interrupt Priority Control
[5:4]	PSTIC	Systick Interrupt Priority Control
[3:2]	PTIM3	Timer3 Interrupt Priority Control
[1:0]	PRTC	RTC Interrupt Priority Control

Note: Priority level assigned ascends from 0 to 3, totaling 4 levels

7.5.5 IP3 (0xD8)

Bit	7	6	5	4	3	2	1	0
Name	RSV		PUART		PI2C		PCMP0	
Type	—	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5:4]	PUART	UART Interrupt Priority Control
[3:2]	PI2C	I^2C Interrupt Priority Control
[1:0]	PCMP0	CMP0 Interrupt Priority Control

Note: Priority level assigned ascends from 0 to 3, totaling 4 levels

7.5.6 TCON (0x88)

Bit	7	6	5	4	3	2	1	0
Name	IF1	TSDF	TSDIF	IT1		IF0	IT0	
Type	R/W0	R	R/W0	R/W	R/W	R/W0	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	IF1	INT1 Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	TSDF	Over-temperature State Flag 0: The current temperature does not exceed the threshold 1: The current temperature exceeds the threshold Note: This flag works with TSD Interrupt Flag TCON[TSDIF].
[5]	TSDIF	TSD Interrupt Flag This bit is set to “1” by hardware when an over-temperature event occurs. Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect Note: This flag works with Over-temperature State Flag TCON[TSDF].
[4:3]	IT1	INT1 Trigger Level Selection 00: Interrupt on rising edge 01: Interrupt on falling edge 1X: Interrupt on level change (rising or falling)
[2]	IF0	INT0 Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[1:0]	IT0	INT0 Trigger Level Selection 00: Interrupt on rising edge 01: Interrupt on falling edge 1X: Interrupt on level change (rising or falling)

8 I^{2}C

8.1 I^{2}C Introduction

The I^{2}C module provides an industry standard two-wire serial interface and is a simple bi-directional synchronous serial bus for communication between MCU and external I^{2}C devices, as shown in Figure 8-1. The bus consists of two serial lines: SDA (serial data line) and SCL (serial clock line). P0.1 serves as SDA port and P1.1 as SCL port. After I^{2}C is enabled, P0.1/P1.1 automatically shift into open-drain outputs.

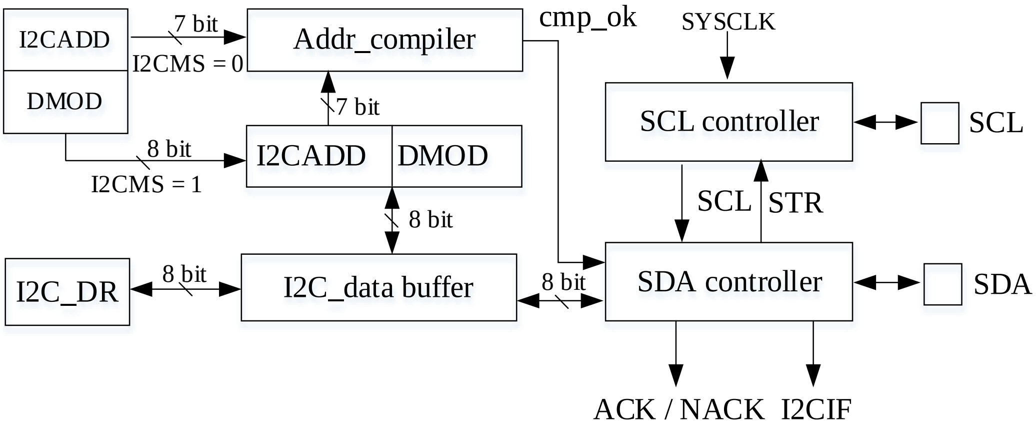


Figure 8-1 I^{2}C Block Diagram

Features:

- Support standard mode (up to 100kHz), fast mode (up to 400kHz) and fast plus mode (up to 1MHz)
- Support master mode and slave mode
- Support 7-bit address mode and general call address mode

Both SDA and SCL lines are high when the bus is idle, which is the only basis for detecting whether the bus is idle or not. Only one master device and at least one slave device are active on the bus during the transmission. When the bus is occupied, other devices must wait for the bus idle to start an I^{2}C communication. The master starts the bus to transfer data. Clock signal is sent to all devices via SCL and the slave address and read/write mode are sent via SDA. When a device on the bus matches the address, it acts as a slave. The relationships between masters and slaves or data transfer direction on the bus are not constant. The process for the master to send data to the slave is shown in Figure 8-2. The master first addresses the slave device and waits for the slave response. And then it sends data to the slave. Finally, the master terminates the data transmission. The process for the master to receive data from the slave is shown in Figure 8-3. The master first addresses the slave and waits for the slave response. And then, it receives the data from the slave. Finally, the master terminates the data transmission. In this case, the master generates the timing clock and stops the data transmission.

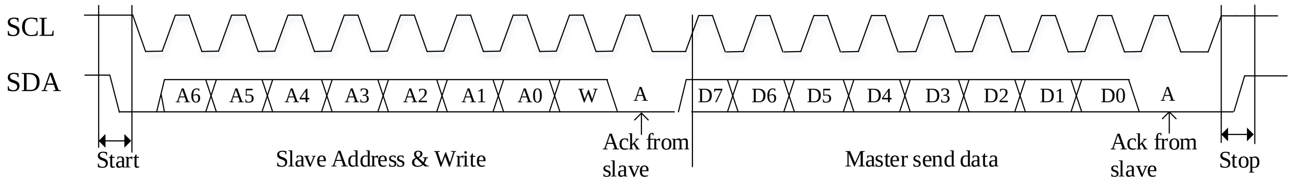


Figure 8-2 Master Sends Data to Slave

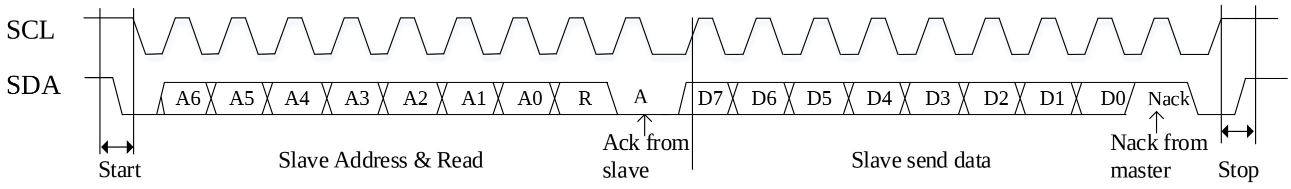


Figure 8-3 Master Receives Data from Slave

8.2 SPI Instructions

8.2.1 Master Mode

1. Set I2C_CR[I2CMS] to “1” to select master mode;
2. Configure I2C_CR [I2CSPD] to set the clock rate of SCL;
3. Configure I2C_ID[I2CADD] to set the slave address;
4. Configure I2C_SR[DMOD] to set the read/write direction;
5. Set I2C_CR[I2CEN] to “1”to enable I^{2}C;
6. Set I2C_SR[I2CSTA] to “1” to send START and address. After ACK/NACK is received, I2C_SR[STR] is set to “1” by hardware and SCL is pulled LOW by the master
7. Data Transmission: Write the data to I2C_DR register. The master starts to send data after I2C_SR[STR] is reset and SCL is released. After the data is transmitted and ACK/NACK is received, I2C_SR[STR] is to “1” by hardware and SCL is pulled to LOW by the master;
8. Data Reception: The master starts to receive data after I2C_SR[STR] is reset and SCL is released. When the data is received, I2C_SR[STR] is set to “1” by hardware and SCL is pulled LOW by the master. Configure ACK/NACK via I2C_SR[NACK], and then clear I2C_SR[STR] to release SCL to send ACK/NACK signal. After the data is received, I2C_SR[STR] is set to “1” by hardware and SCL is pulled LOW by the master
9. Stop Communication: Set I2C_SR[I2CSTP] to “1” when I2C_SR[STR] is “1”. Stop signal is sent after I2C_SR[STR] is cleared.

8.2.2 Slave Mode

1. Set I2C_CR[I2CMS] to “0” to select slave mode;
2. Configure I2C_ID[I2CADD] to set the slave address or set I2C_ID[GC] to “1” to enable general call mode;
3. Set I2C_CR[I2CEN] to “1” to enable I^{2}C;

4. After START signal and the correct address are received, I2C_SR[I2CSTA] and I2C_SR[STR] are set to “1” by hardware and SCL is pulled LOW by the slave. ACK/NACK is configured via I2C_SR[NACK] and the slave determines whether to receive or send the data via I2C_SR[DMOD].
5. Data Transmission: Write the data to I2C_DR register and clear I2C_SR[STR] to release SCL. The data is sent after ACK/NACK is transmitted. After the data is sent and ACK/NACK is received from the master, I2C_SR[STR] is set by hardware to “1” and SCL is pulled LOW by the slave
6. Data Reception: Clear I2C_SR[STR] to release SCL to receive data. After the data is received, I2C_SR[STR] is set to “1” by hardware and SCL is pulled LOW by the slave. ACK/NACK is configured via I2C_SR[NACK] to reset I2C_SR[STR] to release SCL for ACK/NACK transmission. If new data is received, I2C_SR[STR] is set by hardware to “1” and SCL is pulled LOW by the slave.
7. RESTART: If the slave is processing a service when receiving START signal, it halts the current routine and waits for receiving address.

8.2.3 I^{2}C Interrupt Sources

The interrupt sources of I^{2}C include:

- I2C_SR[STR] = 1 generates an interrupt. This interrupt source is valid in both master and slave modes.
- I2C_SR[I2CSTP] = 1 generates an interrupt. This interrupt source is only valid in slave mode.

8.3 I^{2}C Registers

8.3.1 I2C_CR (0x4028)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	I2CEN	I2CMS	RSV			I2CSPD		I2CIE
Type	R/W	R/W	—	—	—	R/W	R/W	R/W
Reset	0	0	—	—	—	0	0	0

Bit	**Name**	**Description**
[7]	I2CEN	I^{2}C Enable Enable the corresponding GPIO and switch to I^{2}C mode, serving as collector open-drain output. The pull-up setting decides whether to pull the I^{2}C HIGH. 0: Disable 1: Enable
[6]	I2CMS	Master/Slave Mode Selection 0: Slave mode 1: Master mode
[5:3]	RSV	Reserved
[2:1]	I2CSPD	I^{2}C transfer rate settings, valid only in master mode 00: 100kHz 01: 400kHz 10: 1MHz 11: Reserved
[0]	I2CIE	I^{2}C Interrupt Enable 0: Disable 1: Enable

8.3.2 I2C_ID (0x4029)

Bit	7	6	5	4	3	2	1	0
Name	I2CADD							GC
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	1	0	1	0	1	0

Bit	Name	Description
[7:1]	I2CADD	Slave address
[0]	GC	General call mode, valid only in slave mode 0: General call is disabled 1: General call is enabled, i.e., the receiving device also reads an ACK at address 0x00

8.3.3 I2C_DR (0x402A)

Bit	7	6	5	4	3	2	1	0
Name	I2C_DR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	I2C_DR	I^{2}C Data Register Read: Data to be sent or received Write: Data to be sent

8.3.4 I2C_SR (0x402B)

Bit	7	6	5	4	3	2	1	0
Name	I2CBSY	DMOD	RSV	I2CSTA	I2CSTP	STR	NACK	I2CIF
Type	R	R/W	—	R/W	R/W	R/W0	R/W	R
Reset	0	0	—	0	0	0	0	0

Bit	Name	Description
[7]	I2CBSY	I^{2}C Busy Flag I2C_SR[I2CBSY] is cleared to “0” by hardware when I2C_CR[I2CEN] is set to 0. Master mode: This bit is set to “1” by hardware after START is sent, and cleared to “0” by hardware after STOP is sent. Slave Mode: This bit is set to “1” by hardware after START is received and address matches, and cleared to “0” by hardware after STOP is received.
[6]	DMOD	I^{2}C R/W Flag 0: Write mode (master sends data, slave receives data) 1: Read mode (master receives data, slave sends data) Note: Read only in slave mode
[5]	RSV	Reserved

<table>
<tr><td>[4]</td><td>I2CSTA</td><td>Master Mode:
When this bit is configured with “1” by the software, START and address bytes are transmitted after both SCL and SDA are HIGH confirmed by the hardware. This bit is cleared to “0” by hardware automatically when the transmission is completed, and I2C_SR[I2CSTA] writing is forbidden during data transmission. After the data is sent or received, I2C_SR[I2CSTA] is set to “1” to transmit RESTART.
0: Not START and address
1: Send START or RESTART and address

Slave Mode:
This bit is set to “1” after hardware receives START and address matches, and cleared to “0” by software.
Table 8-1 Relationship between I2C_SR[I2CSTA/I2CSTP] and the I^{2}C Data Type
<table>
<tr><th>I2CSTA</th><th>I2CSTP</th><th>I^{2}C Data Type</th></tr>
<tr><td>0</td><td>0</td><td>Data byte</td></tr>
<tr><td>0</td><td>1</td><td>STOP</td></tr>
<tr><td>1</td><td>0</td><td>START + address</td></tr>
<tr><td>1</td><td>1</td><td>STOP received first, then START + address bytes</td></tr>
</table>
Note: When I2C_CR[I2CEN] is 0, I2C_SR[I2CSTA] is automatically cleared to “0”.</td></tr>
<tr><td>[3]</td><td>I2CSTP</td><td>Master Mode:
This bit cannot be written to “1” by software unless I2C_SR[I2CBSY] = 1. STOP is transmitted after I2C_SR[STR] is cleared to release SCL. After the transmission, this bit is cleared to “0” automatically by hardware. If I2CSTA and I2CSTP are written to “1” at the same time and I2C_SR[I2CBSY] is “1”, STOP is first send via I^{2}C, then START and address bytes. After START and address bytes are transmitted, I2C_SR[STR] is set to “1” by hardware. I2C_SR[I2CSTP] writing is forbidden during data transmission.
0: STOP is not transmitted.
1: STOP is transmitted.

Slave Mode:
This bit is set to 1 by hardware after STOP is received, and cleared to “0” by software.
Refer to Table 8-1 for status flags.

Note: When I2C_CR[I2CEN] is 0, I2C_SR[I2CSTP] is automatically cleared to “0” by hardware.</td></tr>
<tr><td>[2]</td><td>STR</td><td>I^{2}C Bus Pending Flag
Master Mode:
After START and address or DATA byte are transmitted, I2C_SR[STR] are set to “1” by hardware and SCL is pulled LOW. SCL is released after I2C_SR[STR] is cleared by software. When I2C_SR[I2CSTA] and I2C_SR[I2CSTP] are both “1”, I2C_SR[STR] is set to “1” only after hardware sends STOP and START & address bytes.

Slave mode:
After DATA byte is received or START receives and address matches, I2C_SR[STR] is set to “1” and SCL is pulled LOW. SCL is released after I2C_SR[STR] is cleared by software.

Note: This bit is set to “1” by hardware and cleared to “0” by software. When I2C_CR[I2CEN] = 0, I2C_SR[STR] is automatically cleared to “0”.</td></tr>
<tr><td>[1]</td><td>NACK</td><td>This bit refers to the feedback from a receiver to a sender after a byte is transferred via I^{2}C. It is automatically cleared to “0” when I2C_SR[I2CEN] = 0.
0: ACK, indicating that the receiver can continue to receive data
1: NACK, indicating that the receiver attempts to stop data transmission
When the device is in read mode, I2C_SR[NACK] is configured to send ACK/NACK after the 8th bit of data is received.
0: Bit9 sends ACK
1: Bit9 sends NACK</td></tr>
</table>

		When the device is in write mode, I2C_SR[NACK] is read to receive ACK/NACK after the 8th bit of data is sent. 0: Bit9 receives ACK 1: Bit9 receives NACK
[0]	I2CIF	I^{2}C Interrupt Flag 0: No interrupt pending 1: Interrupt pending In master and slave mode, an interrupt generates when I2C_SR[STR] = 1 In slave mode, an interrupt generates when I2C_SR[I2CSTP] = 1

9 UART

9.1 Introduction

UART is a full-duplex or half-duplex serial data exchange interface as shown in Figure 9-1. The baud rate and UART inverted signals for data transmitting or receiving (mode 1 and mode 3 only) are configurable. Figure 9-2 depicts the UART timing.

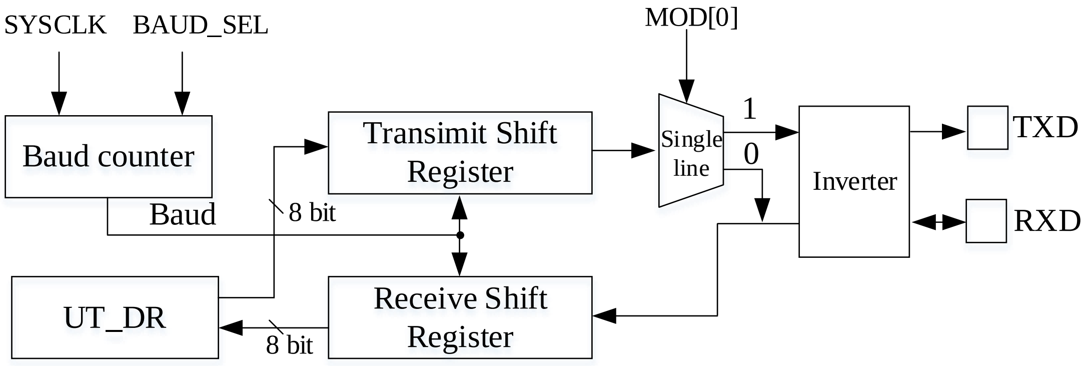


Figure 9-1 UART Block Diagram

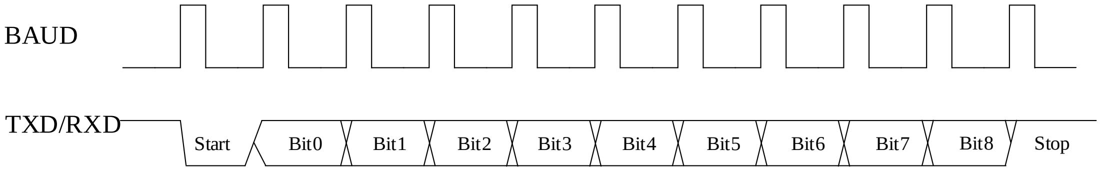


Figure 9-2 UART Communication Timing Diagram

9.2 UART Instructions

The relevant registers shall be enabled before using the UART feature. See section 18.3.7 PH_SEL (0x403C) [7:6] for details.

9.2.1 UART Mode0

Mode0 works in single-wire half-duplex mode. RXD pin is configured as both an output (Transmit Data Bus) and an input (Receive Data Bus). It uses a total of 10 bits (1-bit start, 8-bit data and 1-bit stop) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Data Transmission: Write the data to UT_DR and clear UT_CR[TI]. RXD outputs 10-bit data. UT_CR[TI] is set to “1” after the transmission is completed.

Data Reception: Set UT_CR[REN] to “1” to receive the data and clear UT_CR[RI]. The data is received via RXD. After the data is received, UT_CR[RI] is set to “1” and UT_DR is read to obtain the data.

9.2.2 UART Mode1

Mode1 works in full/half duplex mode. TXD pin is configured as an output (Transmit Data Bus), and RXD as an input (Receive Data Bus). It uses a total of 10 bits (1-bit start, 8-bit data and 1-bit stop) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Data Transmission: Write the data to UT_DR and clear UT_CR[TI]. TXD outputs 10-bit data. UT_CR[TI] is set to “1” after the transmission is completed.

Data Reception: Set UT_CR[REN] to “1” to receive the data and clear UT_CR[RI]. The data is received via RXD. After the data is received, UT_CR[RI] is set to “1” and UT_DR is read to obtain the data.

9.2.3 UART Mode2

Mode2 works in single-wire half-duplex mode. RXD pin is configured as both an output (Transmit Data Bus) and an input (Receive Data Bus). It uses a total of 11 bits (1-bit start, 9-bit data and 1-bit stop) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Data Transmission: Write the first 8 low bits of the data to UT_DR and the 9th bit to UT_CR[TB8], and clear UT_CR[TI]. TXD outputs 11-bit data. UT_CR[TI] is set to “1” after the transmission is completed.

Data Reception: Set UT_CR[REN] to “1” to receive the data and clear UT_CR[RI]. The data is received via RXD. After the data is received, UT_CR[RI] is set to “1”. UT_CR[RB8] stores the 9th bit of the data, and UT_DR stores the first 8 low bits.

9.2.4 UART Mode3

Mode3 works in full/half duplex mode. TXD pin is configured as an output (Transmit Data Bus), and RXD as an input (Receive Data Bus). It uses a total of 11 bits (1-bit start, 9-bit data, 1-bit stop) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Data Transmission: Write the first 8 low bits of the data to UT_DR and the 9th bit to UT_CR[TB8], and clear UT_CR[TI]. TXD outputs 11-bit data. UT_CR[TI] is set to “1” after the transmission is completed.

Data Reception: Set UT_CR[REN] to “1” to receive the data and clear UT_CR[RI]. The data is received via RXD. After the data is received, UT_CR[RI] is set to “1”. UT_CR[RB8] stores the 9th bit of the data, and UT_DR stores the first 8 low bits.

9.2.5 UART Interrupt Sources

UART interrupt sources include:

- After the data is sent via UART, UT_CR[TI] is set to “1” by hardware.
- After the data and STOP are received via UART, UT_CR[RI] is set to “1” by hardware.

9.3 UART Registers

9.3.1 UT_CR (0x98)

Bit	7	6	5	4	3	2	1	0
Name	MOD		SM2	REN	TB8	RB8	TI	RI
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	MOD	Mode Selection 00: Mode 0 01: Mode 1 10: Mode 2 11: Mode 3
[5]	SM2	Communication Mode 0: Single-device communication 1: Multi-device communication
[4]	REN	Serial Input Enable 0: Disable 1: Enable
[3]	TB8	Bit9 of the sent data in mode2 and mode3
[2]	RB8	Bit9 of the received data in mode2 and mode3
[1]	TI	Data Sending Completed Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: The interrupt is generated
[0]	RI	Data Reception Completed Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: The interrupt is generated

9.3.2 UT_DR (0x99)

Bit	7	6	5	4	3	2	1	0
Name	UT_DR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	UT_DR	Send/Receive Data Read: Data received Write: Data to be sent Note: The UART data buffer consists of two independent buffers, i.e., a receive buffer and a transmit buffer, which can send and receive data at the same time. The transmit buffer can be written only but not read, while the receive buffer can be read only but not written. Both buffers share a same address.

9.3.3 UT_BAUD (0x9A, 0x9B)

UT_BAUDH(0x9B)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	BAUD_SEL	RSV	UART_RX_INV	UART_TX_INV	UT_BAUD[11:8]			
Type	R/W	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	—	0	0	0	0	0	0
UT_BAUDL(0x9A)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UT_BAUD[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	1	1	0	1	1

Bit	**Name**	**Description**
[15]	BAUD_SEL	Frequency Multiplier Enable 0: Disable 1: Enable
[14]	RSV	Reserved
[13]	UART_RX_INV	RXD Inverse Enable 0: Disable 1: Enable Note: This feature is invalid in single-wire mode
[12]	UART_TX_INV	TXD Inverse Enable 0: Disable 1: Enable Note: This feature is invalid in single-wire mode
[11:0]	UT_BAUD	Baud Rate Setting Baud rate = SYSCLK(16/(1 + UT_BAUD[BAUD_SEL]))/(UT_BAUD[BAUD] + 1) For example, baud rate = 9600, UT_BAUD[BAUD_SEL] = 0, UT_BAUD[BAUD] = (24M/16/9600/(1 + 0)) - 1 = 155 (0x9B)

10 Timer1

10.1 Timer1 Instructions

Timer1 consists of a 16-bit up-counting base timer, a 16-bit up-counting reload timer and a 16-bit up-counting process timer. Timer1 can be used for phase commutation, soft switching and lead angle control of single-phase motor. Timer1 features as following:

- The 16-bit up-counting base timer is used to record the time between two position detected events or two writing sequence events (i.e., 180-degree commutation)
- The 16-bit up-counting reload timer is used to record the time from position detected event to reload timer overflow event (i.e., control of lead or lag angle)
- The 16-bit up-counting process timer is used to control the time of downslope, block and upslope of soft switching and the time of motor lock detection
- The 3-bit programmable frequency prescaler divides the system clock. The divided clock is used as the clock source of the three timers.
- Filtering and sampling of Hall signal
- Soft switching is used for phase commutation to reduce the spiky current and voltage overshoot
- Motor lock detection
- Braking control

10.1.1 Timer1 Counting Units

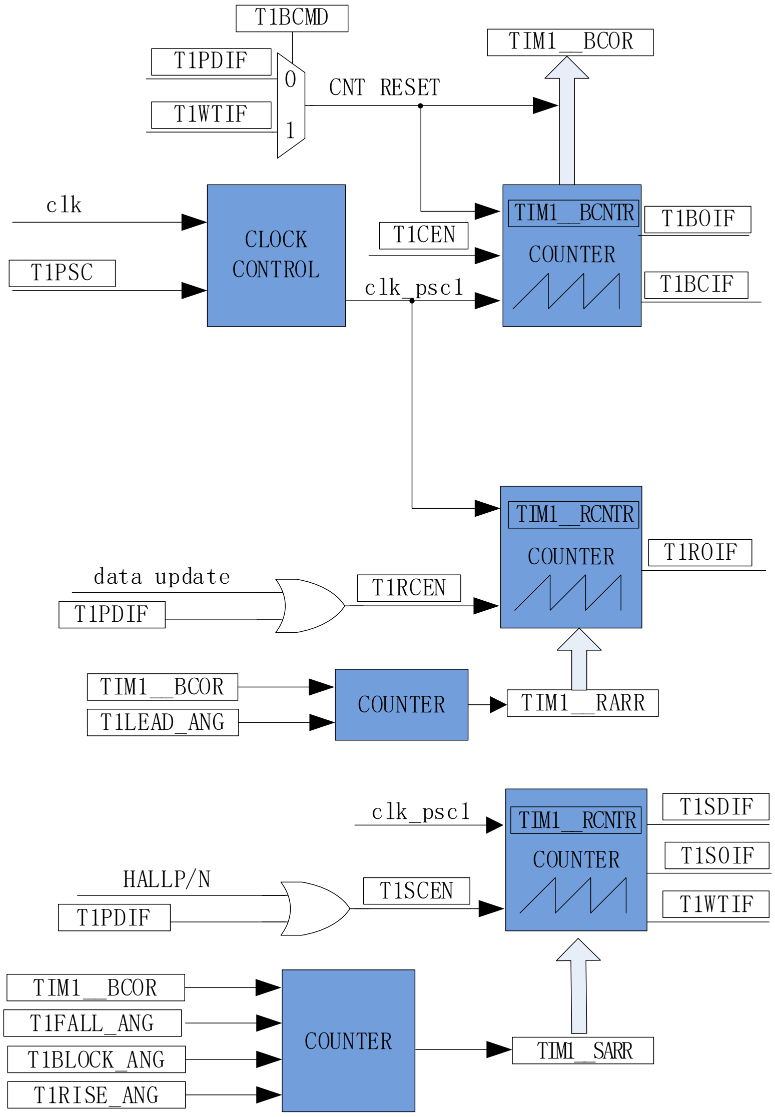


Figure 10-1 Timer1 Counting Units

Timer1 consists of a frequency prescaler, a 16-bit up-counting base timer, a 16-bit up-counting reload timer and a 16-bit up-counting process timer.

10.1.1.1 Prescaler

Frequency prescaler is used to divide the system clock and generate clock source of base timer, reload timer and process timer. It offers 7 division coefficients and can be selected by TIM_CR10[T1PSC]. Since this register has no buffer, the clock rate is immediately updated after the division coefficient is written. Therefore, the division coefficient shall be configured when the base timer, reload timer and process timer are not working. The clock rate clk_psc1 = SYSCLK/(2^TIM_CR10[T1PSC]).

Table 10-1 Mapping between Clock Rate and TIM_CR10[T1PSC]

TIM_CR10[T1PSC]	Coefficient	clk_psc1(Hz)	TIM_CR10[T1PSC]	Coefficient	clk_psc1(Hz)
000	1	24M	100	16	1.5M

TIM_CR10[T1PSC]	Coefficient	clk_psc1(Hz)	TIM_CR10[T1PSC]	Coefficient	clk_psc1(Hz)
001	2	12M	101	32	750k
010	4	6M	110	64	375k
011	8	3M	111	64	375k

10.1.1.2 Base Timer

Base timer is a 16-bit up counter, which consists of register TIM1__BCNTR, TIM1__BCOR and internal register TIM1_BCCR. It is used to periodically sample Hall signals.

Base timer starts when TIM1_CR0[T1CEN] = 1. After its count value reaches 0xFFFF, an overflow event occurs and base timer overflow flag TIM1_SR[T1BOIF] is set to "1". If the base timer overflow interrupt is enabled, an overflow interrupt is generated.

When the base timer receives the reset signal, a capture event occurs and base timer capture flag TIM1_SR[T1BCIF] is set to "1". If the base timer capture interrupt is enabled, an capture interrupt is generated. Meanwhile, the value held by TIM1__BCNTR is sent to TIM1_BCCR and TIM1__BCNTR restarts from 0.

TIM1_CR0[T1BCMD] controls reset signal of the base timer, with Hall active edge reset mode or manual reset mode being selected. In manual reset mode, the reset signal is determined by the parameter held by TIM1_CR8. In Hall active edge reset mode, the reset signal can be configured as falling edge, rising edge or falling/rising edges.

TIM1__BCOR is the average value of TIM1_BCCR register, and represents half cycle length of the Hall signal. The average selection is set by TIM1_CR0[T1CFLT].

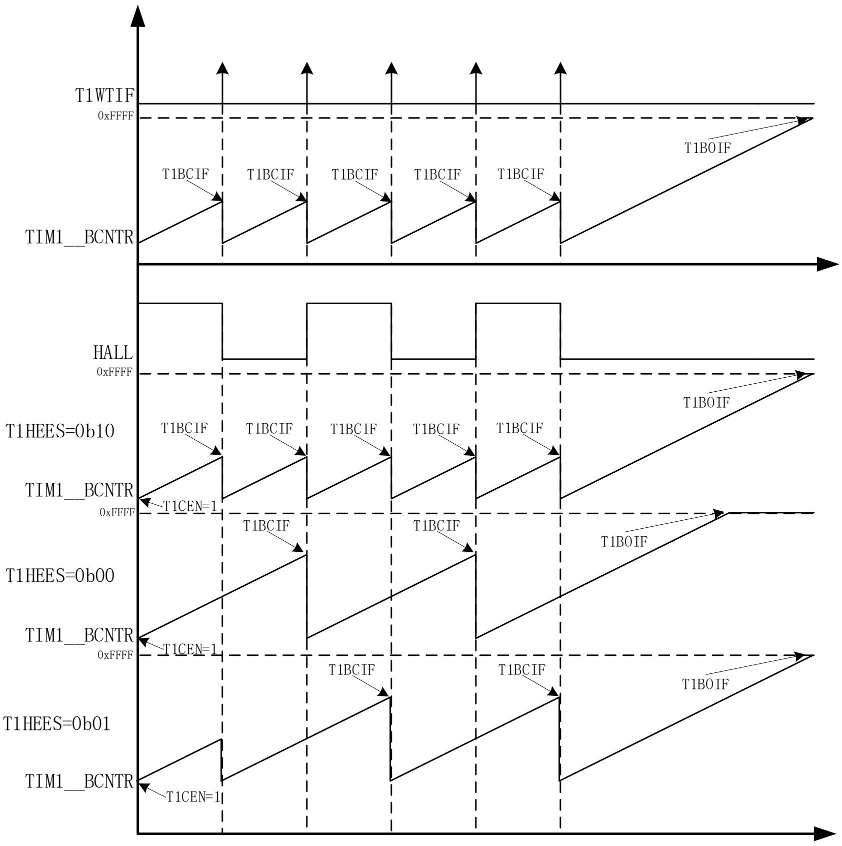


Figure 10-2 Waveform of Basic Timer (in Hall Active Edge Reset Mode)

10.1.1.3 Reload timer

Reload timer is a 16-bit up counter that is implemented by TIM1__RCNTR and TIM1__RARR to control the commutation angle. The lead/lag angle is controlled by TIM1_CR4[T1LEAD_ANG] and Timer1 automatically updates TIM1__RARR based on this input angle.

Reload timer overflows when TIM1__RCNTR increases to TIM1__RARR. In this case, the timer stops and TIM1_SR[T1ROIF] (overflow interrupt flag of the reload counter) is set to “1”. If the reload timer overflow interrupt is enabled, an overflow interrupt is generated.

TIM1_CR0[T1RCMM] controls reload mode of the reload timer, with automatic mode or manual mode being selected. In manual mode, after the reload timer stops, you have to manually update TIM1__RARR register and enable the reload timer. In automatic mode, TIM1__RARR is automatically updated according to soft switching, and the time when reload timer restarts is determined by Timer1.

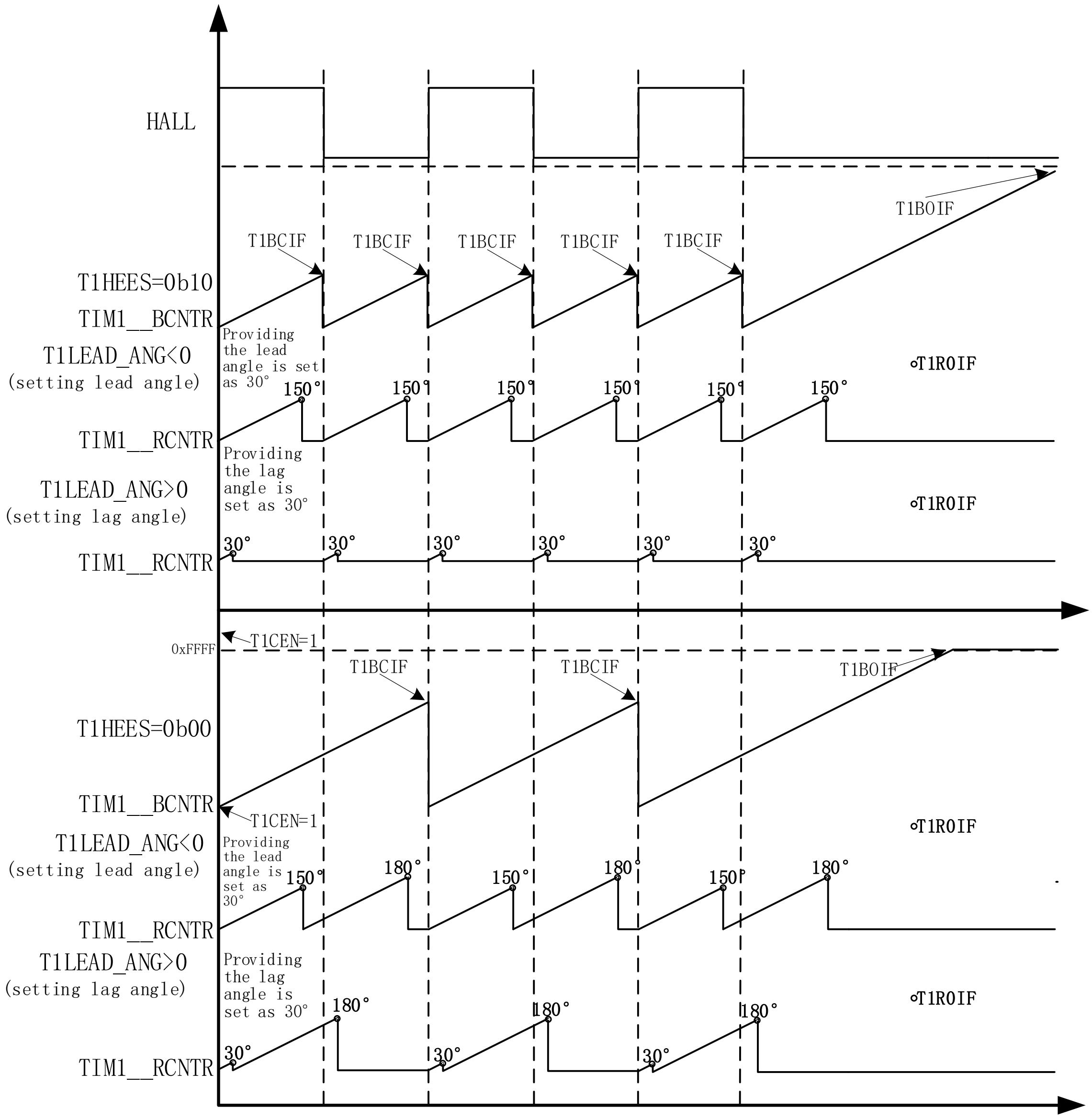


Figure 10-3 Waveform of Reload Timer

10.1.1.4 Process Timer

Process timer is a 16-bit up counter that is controlled by TIM1__SCNTR and TIM1__SARR. It is used to implement motor lock detection and soft switching for phase commutation. The upslope angle, block angle, downslope angle during soft switching and the angle for motor lock detection are configured by TIM1_CR1[T1FANG], TIM1_CR2[T1BANG], TIM1_CR3[T1RANG] and TIM1_CR9[T1SDSEL], respectively. TIM1__SARR is automatically updated based on these settings and motor control state.

Process timer overflows when TIM1__SCNTR increases to TIM1__SARR. In this case, the timer stops and TIM1_SR[T1SOIF] (overflow interrupt flag of the process timer) is set to “1”. If the process timer overflow interrupt is enabled, an overflow interrupt is generated. During downslope, the process timer stops when an overflow event occurs.

TIM1_CR0[T1SCMM] controls reload mode of the process timer, with automatic mode or manual mode being selected. In manual mode, after the process timer stops, you have to manually update TIM1__RARR register and enable the process timer. In automatic mode, TIM1__RARR is automatically updated after the process timer stops, and the process timer restarts when the reload timer overflow flag TIM1_SR[T1ROIF] is set to “1”.

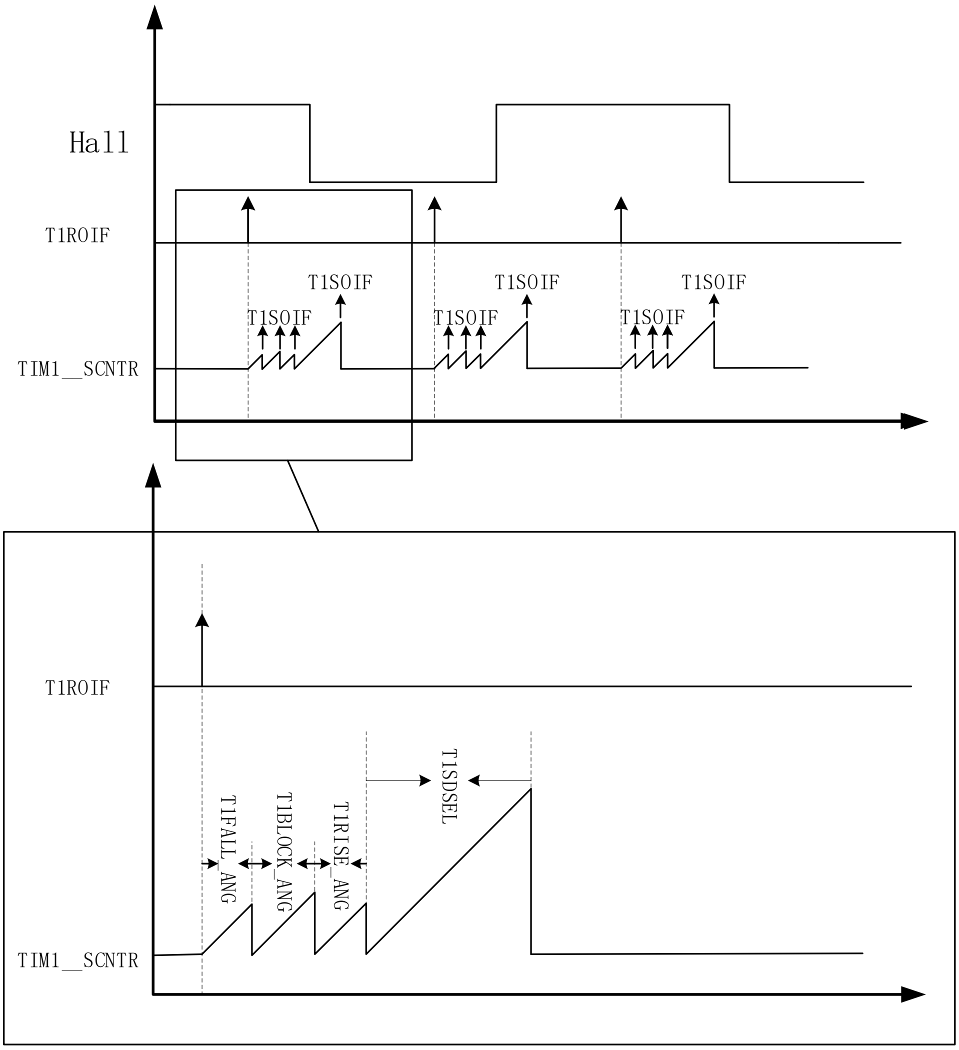


Figure 10-4 Waveform of Process Timer

10.1.2 Filtering

The Hall signal input by comparator or GPIO can be filtered. The filtered pulse width of input noise can be selected as 16/32/64/256 system clocks configured by TIM1_CR4[T1HALLINM]. When the filtering out pulse width is selected as 16 system clocks, the timing diagram of filter module is shown in Figure 10-5.

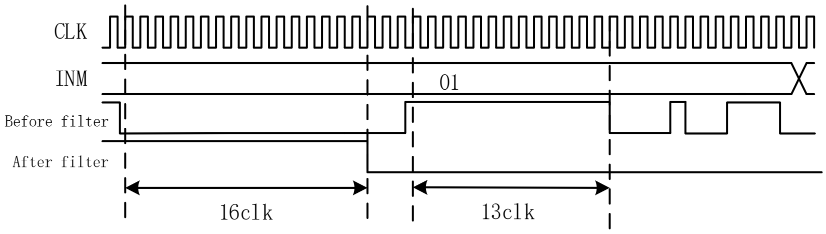


Figure 10-5 Timing Diagram of Filtering Module

10.1.3 Timer1 Interrupt Sources

Timer1 has 7 interrupt sources:

- Base timer overflow interrupt
- Base timer capture interrupt
- Reload timer overflow interrupt
- Process timer overflow interrupt
- Writing sequence interrupt
- Position detected interrupt
- Motor lock detected interrupt

These interrupts can be enabled by setting the corresponding enable bits in TIM1_IRE.

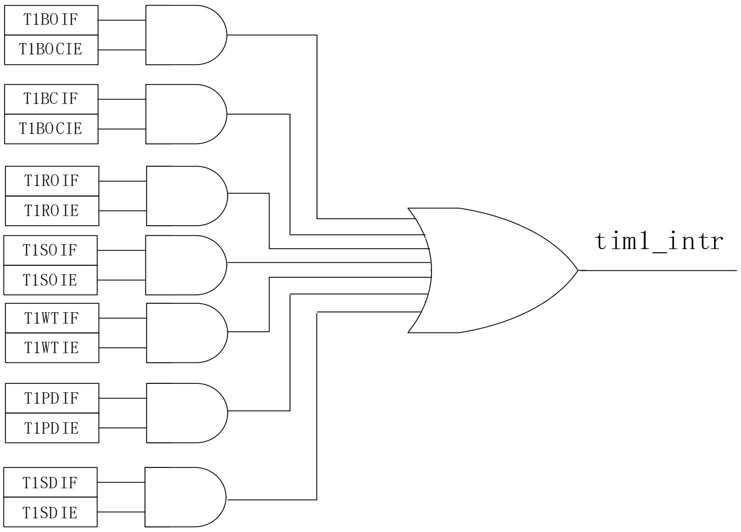


Figure 10-6 Timer1 Interrupt Sources

10.1.4 Soft Switching

Soft switching is used for commutation control of the single-phase motor to reduce the spiky current and voltage overshoot. In soft switching, Timer1 implements downslope process, block process and upslope process sequentially. In downslope process, the output of Driver is reduced from reference and motor starts to release the energy in stator. Then block process starts and motor phase is commutated. Driver output is disabled and motor continues to release the energy in stator. In upslope process, the output of Driver is increased gradually and reaches the reference. The downslope process and block process reduce the voltage overshoot and the upslope process reduce the spiky current effectively.

When TIM1_CR8[T1CST] = 2/3, TIM1_CR8[T1CST] indicates the motor operating sector (each 180 degree is a sector). In block process, Timer1 disables the Driver output and controls the polarity of Driver output corresponding to TIM1_DBRB[7:4]/TIM1_DBRB[3:0] configured by TIM1_CR8[T1CST]. If the controller is not in block process, Timer1 selects the corresponding TIM1_DBR2/3 to control the output of Driver based on the value of TIM1_CR8[T1CST]. TIM1__PWMDR controls the duty cycle of PWM. In downslope/upslope process, TIM1__RPWMDDR/TIM1__FPWMDDR controls the increment/decrement of PWM duty cycle. DRV_DR decides maximum value of the duty cycle, and is loaded to TIM1_PWMDR by hardware when the controller enters motor lock detection state. Timer1 enters automatic commutation after 2/3 is written into TIM1_CR8[T1CST]. When a phase commutation is implemented, TIM1_CR8[T1CST] is changed from 2 to 3 or 3 to 2.

As shown in Figure 10-7, at the beginning, TIM1_CR8[T1CST] is 2 and TIM1_CR9[T1BSCS] is 0. TIM1_DBR2 controls the output of Driver. When the reload timer overflows, the process timer restarts from “0” and enters downslope process. TIM1__PWMDR is decreased by one in every PWM cycle with the step configured by TIM1__FPWMDDR. The minimum value of TIM1__PWMDR is 0. When the process timer overflows and restarts from “0”, a writing sequence event is generated and the writing sequence interrupt flag TIM1_SR[T1WTIF] is set to “1”. TIM1_CR8[T1CST] is updated to 3 by hardware and the process timer enters the block process. In this state, TIM1__PWMDR is fixed at 0 and PWM duty cycle stays at 0, where PWM output is disabled and TIM1_DBRB[7:4] controls output of the Driver. When the process timer overflows and restarts from “0”, and enters upslope process, TIM1_DBR3 controls output of the Driver, and TIM1__PWMDR is decreased by one in every PWM cycle with the step configured by TIM1__RPWMDDR. The maximum value of TIM1__PWMDR is DRV_DR. When the process timer overflows and restarts from “0”, and enters motor lock detection state, TIM1__PWMDR is fixed to DRV_DR.

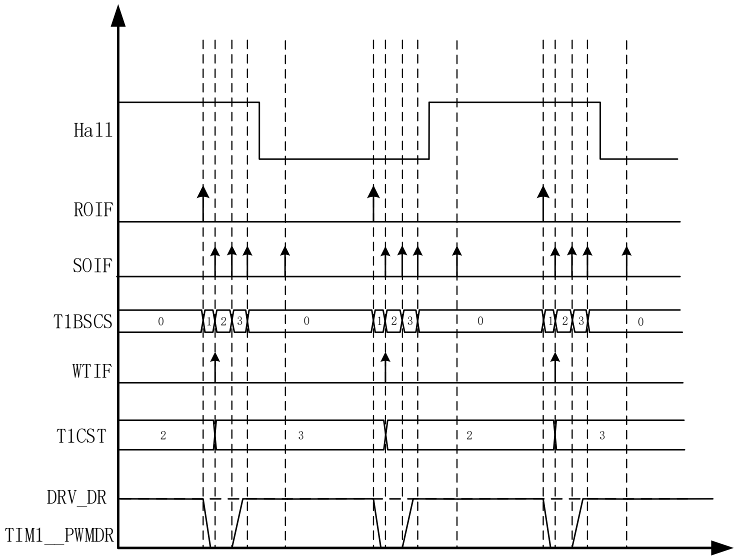


Figure 10-7 Soft Switching and Motor Lock Detection

10.1.5 Motor Lock Detection

The motor lock detection is to detect whether Hall signals are converted into the expected electrical levels by the driver. If not, the rotor position is still in a sector of 180 degree and motor lock event occurs. During motor rotation, the level of Hall signals shall change based on TIM1_CR8[T1CST]. If not, the lock detected interrupt flag TIM1_SR[T1SDIF] is set to “1”. If TIM1_CR8[T1CST] = 2, Hall signals shall be switched to high level, and if TIM1_CR8[T1CST] = 3, Hall signals shall be switched to low level. The time for motor lock detection is configured by TIM1_CR9[T1SDSEL].

As shown in Figure 10-7, during motor lock detection, the process timer stops after it get overflowed. If TIM1_CR8[T1CST] = 2 and Hall signals stay LOW, or TIM1_CR8[T1CST] = 3 and Hall signals stay HIGH, the lock detected interrupt flag TIM1_SR[T1SDIF] is set to “1”.

10.1.6 Braking

A write of “1” by software to TIM1_CR8[T1CST] enters braking state. In this state, the high sides of U and V phases are turned off while the low sides are turned on.

10.2 Timer1 Registers

10.2.1 TIM1_CR0 (0xB1)

Bit	7	6	5	4	3	2	1	0
Name	RSV		T1RCMM	T1SCMM	T1CFLT		T1BCMD	T1CEN
Type	—	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	1	1	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	T1RCMM	Reload Timer Manual Mode Enable 0: Automatic mode is enabled 1: Manual mode is enabled
[4]	T1SCMM	Process Timer Manual Mode Enable 0: Automatic mode is enabled 1: Manual mode is enabled
[3:2]	T1CFLT	TIM1_BCOR Average Selection 00: TIM1_BCOR is equal to TIM1_BCCR 01: TIM1_BCOR is the average value of last 2 TIM1_BCCR 10: TIM1_BCOR is the average value of last 4 TIM1_BCCR 11: TIM1_BCOR is the average value of last 8 TIM1_BCCR
[1]	T1BCMD	Base Timer Reset Mode Selection 0: Hall active edge reset mode 1: Manual reset mode
[0]	T1CEN	Base Timer Enable 0: Disable 1: Enable

10.2.2 TIM1_CR1 (0xB2)

Bit	7	6	5	4	3	2	1	0
Name	T1RCENWE	T1RCEN	T1FANG					
Type	W1	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T1RCENWE	TIM1_CR1[T1RCEN] Write Enable 0: No effect 1: When TIM1_CR1 is updated, TIM1_CR1[T1RCENWE] and TIM1_CR1[T1RCEN] shall be configured simultaneously to enable or disable TIM1_CR1[T1RCEN]. A write of “0xC0” to TIM1_CR1 enables TIM1_CR1[T1RCEN] and “0x80” disables TIM1_CR1[T1RCEN].
[6]	T1RCEN	Reload timer enable When TIM1_CR1 is updated, TIM1_CR1[T1RCENWE] and TIM1_CR1[T1RCEN] should be configured simultaneously to enable or disable TIM1_CR1[T1RCEN]. A write of “0xC0” to TIM1_CR1 enables TIM1_CR1[T1RCEN] and “0x80” disables TIM1_CR1[T1RCEN] 0: Disable 1: Enable
[5:0]	T1FANG	Downslope Angle Setting The value of 0 ~ 63 is mapped to 0° ~ 90°

10.2.3 TIM1_CR2 (0xB3)

Bit	7	6	5	4	3	2	1	0
Name	RSV		T1BANG					
Type	—	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	0	0	0	0	0	0

Bit	Name	Description

[7:6]	RSV	Reserved
[5:0]	T1BANG	Block Angle Setting The value of 0 ~ 63 is mapped to 0° ~ 90°

10.2.4 TIM1_CR3 (0xB4)

Bit	7	6	5	4	3	2	1	0
Name	T1SCENWE	T1SCEN	T1RANG					
Type	W1	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T1SCENWE	T1SCEN Write Enable 0: No effect 1: When TIM1_CR3 is updated, TIM1_CR3[T1SCENWE] and TIM1_CR3[T1SCEN] shall be configured simultaneously to enable or disable TIM1_CR3[T1SCEN]. A write of "0xC0" to TIM1_CR3 enables TIM1_CR3[T1SCEN] and 0x80 disables TIM1_CR3[T1SCEN].
[6]	T1SCEN	Process Timer Enable When TIM1_CR3 is updated, TIM1_CR3[T1SCENWE] and TIM1_CR3[T1SCEN] shall be configured simultaneously to enable or disable TIM1_CR3[T1SCEN]. A write of "0xC0" to TIM1_CR3 enables TIM1_CR3[T1SCEN] and "0x80" disables TIM1_CR3[T1SCEN]. 0: Disable 1: Enable
[5:0]	T1RANG	Upslope Angle Setting The value of 0 ~ 63 is mapped to 0° ~ 90°

10.2.5 TIM1_CR4 (0xB5)

Bit	7	6	5	4	3	2	1	0
Name	T1LEAD_ANG							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	T1LEAD_ANG	Lead/Lag Angle of Phase Commutation The value of 0 ~ 128 is mapped to lag angle 0° ~ 90° The value of 0~ –128 is mapped to lead angle 0° ~ 90°

10.2.6 TIM1_CR5 (0xB6)

Bit	7	6	5	4	3	2	1	0
Name	RSV	T1HREN	T1HALLINM		RSV		T1HEES	
Type	—	R/W	R/W	R/W	—	—	R/W	R/W
Reset	—	0	0	0	—	—	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6]	T1HREN	Hall Reverse Enable 0: Normal Mode 1: Hall Reverse Enable
[5:4]	T1HALLINM	Hall Input Signal Filtering Selection 00: 16 system clocks 01: 32 system clocks 10: 64 system clocks 11: 256 system clocks
[3:2]	RSV	Reserved
[1:0]	T1HEES	Hall Active Edge Selection (phase commutation signal source is selected as Hall, i.e., TIM1_CR1[T1BCMD] = 0) 00: Rising edge

		01: Falling edge 10: Rising edge/falling edge 11: Reserved

10.2.7 TIM1_CR8 (0xC1)

Bit	7	6	5	4	3	2	1	0
Name	T1HALLIN	T1BEMFIN	RSV				T1CST	
Type	R	R	—	—	—	—	R/W	R/W
Reset	0	0	—	—	—	—	0	0

Bit	Name	Description
[7]	T1HALLIN	Hall Signal Filtering Result 0: The filtering result is LOW 1: The filtering result is HIGH
[6]	T1BEMFIN	BEMF Filtering Result of CMP0 0: The CMP0 filtering result of BEMF is LOW 1: The CMP0 filtering result of BEMF is HIGH
[5:2]	RSV	Reserved
[1:0]	T1CST	Timer1 Output State Selection 00: Idle state 01: Braking state 10: Output state 2 11: Output state 3 Note: In automatic mode, the process timer is automatically switched between output mode 2 and output mode 3.

10.2.8 TIM1_CR9 (0xDC)

Bit	7	6	5	4	3	2	1	0
Name	T1BSCS		T1SDSEL		RSV			
Type	R	R	R/W	R/W	—	—	—	—
Reset	0	0	0	0	—	—	—	—

Bit	Name	Description
[7:6]	T1BSCS	Process Timer State Flag 00: Process timer is in motor lock detection state 01: Process timer is in downslope process of soft switching 10: Process timer is in block process of soft switching 11: Process timer is in upslope process of soft switching
[5:4]	T1SDSEL	Motor Lock Detection Angle Setting (count after upslope process is completed) If the corresponding Hall level is not detected in the time sector configured by TIM1_CR9[T1SDSEL], a lock detected interrupt is generated. 00: 0° 01: 45° 10: 90° 11: 135°
[3:0]	RSV	Reserved

10.2.9 TIM1_CR10 (0xDD)

Bit	7	6	5	4	3	2	1	0
Name	RSV					T1PSC		
Type	—	—	—	—	—	R/W	R/W	R/W
Reset	—	—	—	—	—	0	0	0

Bit	Name	Description
[7:3]	RSV	Reserved
[2:0]	T1PSC	Timer Clock Source Frequency Selection This bit is used to divide the system clock as the clock source for base timer, reload timer

		and process timer. The clock source frequency of the three timers: 000: 24MHz 001: 12MHz 010: 6MHz 011: 3MHz 100: 1.5MHz 101: 750kHz 110: 375kHz 111: 375kHz

10.2.10 TIM1_IRE (0xD1)

Bit	7	6	5	4	3	2	1	0
Name	RSV		T1BOCIE	T1SDIE	T1ROIE	T1WTIE	T1PDIE	T1SOIE
Type	—	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	T1BOCIE	Base Timer Overflow & Capture Interrupt Enable 0: Disable 1: Enable
[4]	T1SDIE	Motor Lock Detected Interrupt Enable 0: Disable 1: Enable
[3]	T1ROIE	Reload Timer Overflow Interrupt Enable 0: Disable 1: Enable
[2]	T1WTIE	Writing Sequence Interrupt Enable 0: Disable 1: Enable
[1]	T1PDIE	Position Detected Interrupt Enable 0: Disable 1: Enable
[0]	T1SOIE	Process Timer Overflow Interrupt Enable 0: Disable 1: Enable

10.2.11 TIM1_SR (0xD4)

Bit	7	6	5	4	3	2	1	0
Name	RSV	T1BOIF	T1BCIF	T1SDIF	T1ROIF	T1WTIF	T1PDIF	T1SOIF
Type	—	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0
Reset	—	0	0	0	0	0	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6]	T1BOIF	Base Timer Overflow Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	T1BCIF	Base Timer Capture Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	T1SDIF	Motor Lock Detected Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending

		Write: 0: This bit is cleared to “0” 1: No effect
[3]	T1ROIF	Reload Timer Overflow Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[2]	T1WTIF	Writing Sequence Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[1]	T1PDIF	Position Detected Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[0]	T1SOIF	Process Timer Overflow Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect

10.2.12 TIM1_DBR2 (0xBA)

Bit	7	6	5	4	3	2	1	0
Name	T1VHP	T1VLP	T1UHP	T1ULP	T1VHE	T1VLE	T1UHE	T1ULE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T1VHP	It’s valid for output state 2 (except in the block process of soft switching) High-side Output Polarity of V-phase 0: Active High 1: Active Low
[6]	T1VLP	It’s valid for output state 2 (except in the block process of soft switching) Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[5]	T1UHP	It’s valid for output state 2 (except in the block process of soft switching) High-side Output Polarity of U-phase 0: Active High 1: Active Low
[4]	T1ULP	It’s valid for output state 2 (except in the block process of soft switching) Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[3]	T1VHE	It’s valid for output state 2 (except in the block process of soft switching) High-side Output Enable of V-phase 0: Disable 1: Enable

[2]	T1VLE	It’s valid for output state 2 (except in the block process of soft switching) Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	T1UHE	It’s valid for output state 2 (except in the block process of soft switching) High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	It’s valid for output state 2 (except in the block process of soft switching) Low-side Output Enable of U-phase 0: Disable 1: Enable

10.2.13 TIM1_DBR3 (0xBB)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1VHP	T1VLP	T1UHP	T1ULP	T1VHE	T1VLE	T1UHE	T1ULE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_DBR6L(0x4075)								

Bit	**Name**	**Description**
[7]	T1VHP	It’s valid for output state 3 (except in the block process of soft switching) High-side Output Polarity of V-phase 0: Active High 1: Active Low
[6]	T1VLP	It’s valid for output state 3 (except in the block process of soft switching) Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[5]	T1UHP	It’s valid for output state 3 (except in the block process of soft switching) High-side Output Polarity of U-phase 0: Active High 1: Active Low
[4]	T1ULP	It’s valid for output state 3 (except in the block process of soft switching) Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[3]	T1VHE	It’s valid for output state 3 (except in the block process of soft switching) High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	T1VLE	It’s valid for output state 3 (except in the block process of soft switching) Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	T1UHE	It’s valid for output state 3 (except in the block process of soft switching) High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	It’s valid for output state 3 (except in the block process of soft switching) Low-side Output Enable of U-phase 0: Disable 1: Enable

10.2.14 TIM1_DBRB (0xDE)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1C3VHP	T1C3VLP	T1C3UHP	T1C3ULP	T1C2VHP	T1C2VLP	T1C2UHP	T1C2ULP
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T1C3VHP	It's valid for output state 3 in the block process of soft switching High-side Output Polarity of V-phase 0: Active High 1: Active Low
[6]	T1C3VLP	It's valid for output state 3 in the block process of soft switching Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[5]	T1C3UHP	It's valid for output state 3 in the block process of soft switching High-side Output Polarity of U-phase 0: Active High 1: Active Low
[4]	T1C3ULP	It's valid for output state 3 in the block process of soft switching Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[3]	T1C2VHP	It's valid for output state 2 in the block process of soft switching High-side Output Polarity of V-phase 0: Active High 1: Active Low
[2]	T1C2VLP	It's valid for output state 2 in the block process of soft switching Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[1]	T1C2UHP	It's valid for output state 2 in the block process of soft switching High-side Output Polarity of U-phase 0: Active High 1: Active Low
[0]	T1C2ULP	It's valid for output state 2 in the block process of soft switching Low-side Output Polarity of U-phase 0: Active High 1: Active Low

10.2.15 TIM1__BCNTR (0xD2, 0xD3)

TIM1__BCNTRH(0xD3)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__BCNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__BCNTRL(0xD2)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__BCNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1__BCNTR	Count value of base timer, which is used to record the time between two adjacent Hall active edges

10.2.16 TIM1__BCOR (0xC2, 0xC3)

TIM1__BCORH(0xC3)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__BCOR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0
TIM1__BCOR(0xC2)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__BCORL							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:0]	TIM1__BCOR	This bit is used to capture filtered count values held in the Base Timer. TIM1_BCCR holds the filtering value, or the base of Hall signal width.

10.2.17 TIM1__RCNTR (0xBC, 0xBD)

TIM1__RCNTRH(0xBD)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__RCNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__RCNTRL(0xBC)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__RCNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:0]	TIM1__RCNTR	Count value of reload timer

10.2.18 TIM1__RARR (0xBE, xBF)

TIM1__RARRH(0xBF)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__RARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__RARRL(0xBE)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__RARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:0]	TIM1__RARR	Overflow value of reload timer

10.2.19 TIM1__SCNTR (0xC4, 0xC5)

TIM1__BSCNTRH(0xC5)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__SCNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__SCNTRL(0xC4)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__SCNTR[7:0]							

Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1__SCNTR	Count value of process timer

10.2.20 TIM1__SARR (0xC6, 0xC7)

TIM1__SARRH(0xC7)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__SARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1
TIM1__SARRL(0xC6)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__SARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	Name	Description
[15:0]	TIM1__SARR	Overflow value of process timer

10.2.21 TIM1__FPWMDDR (0xCC, 0xCD)

TIM1__FPWMDDRH(0xCD)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__FPWMDDR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__FPWMDDRL(0xCC)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__FPWMDDR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1__FPWMDDR	Decrement of duty cycle in every PWM cycle (TIM1__PWMDR), which is used to control downslope speed Decrement of duty cycle = TIM1__FPWMDDR/16/DRV_ARR*100%

10.2.22 TIM1__PWMDR (0xCA, 0xCB)

TIM1__PWMDRH(0xCB)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__PWMDR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__PWMDRL(0xCA)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__PWMDR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1__PWMDR	PWM duty cycle Duty cycle = TIM1__PWMDR/16/DRV_ARR*100%

10.2.23 TIM1__RPWMDDR (0xCE, 0xCF)

TIM1__RPWMDDRH(0xCF)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__RPWMDDR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__RPWMDDRL(0xCE)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__RPWMDDR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1__RPWMDDR	Increment of duty cycle in every PWM (TIM1__PWMDR) Increment of duty cycle = TIM1__FPWMDDR/16/DRV_ARR*100%

11 Timer2

11.1 Timer2 Instructions

Timer2 has two working modes:

- Output mode: PWM generation
- Input capture mode: Detect the duration of high and low level of input PWM, which is used to calculate PWM duty cycle

Timer2 features:

- 2-bit programmable prescaler divides the system clock as the clock source of base timer
- 16-bit up-counting base timer. Counting clock source serves as the output of prescaler
- Input signal filtering
- Input signal edge detection
- Output PWM signal and single compare output
- ADC sampling
- Interrupt event

11.1.1 Prescaler

Frequency prescaler is used to divide the system clock and generate clock source for base timer. It offers 4 division coefficients and can be selected by TIM2_CR0[T2PSC]. Since this register has no buffer, the clock rate is immediately updated after the division coefficient is written is written. Therefore, the division coefficient shall be configured when the base timer, reload timer and process timer are not working. The clock rate clk_psc2 = SYSCLK/(4^TIM2_CR0[T2PSC]). The clock rate corresponding to different TIM2_CR0[T2PSC] value as shown in Table 11-1.

Table 11-1 Mapping between Clock Rate and TIM2_CR0[T2PSC]

TIM2_CR0[T2PSC]	Coefficient	clk_psc2(Hz)	TIM2_CR0[T2PSC]	Coefficient	clk_psc2(Hz)
00	1	24M	10	16	1.5M
01	4	6M	11	64	375k

11.1.2 Reading, Writing and Counting of TIM2__CNTR

When TIM2_CR1[T2CEN] = 1, TIM2__CNTR starts to count. The write operation to TIM2__CNTR directly changes the value of the register, so the basic timer shall be disabled before the write operation. When reading TIM2__CNTR, the software reads the high-order bytes first and the hardware synchronously caches the low-order bytes. When reading the low byte, the software reads the cached data.

11.1.3 Output Mode

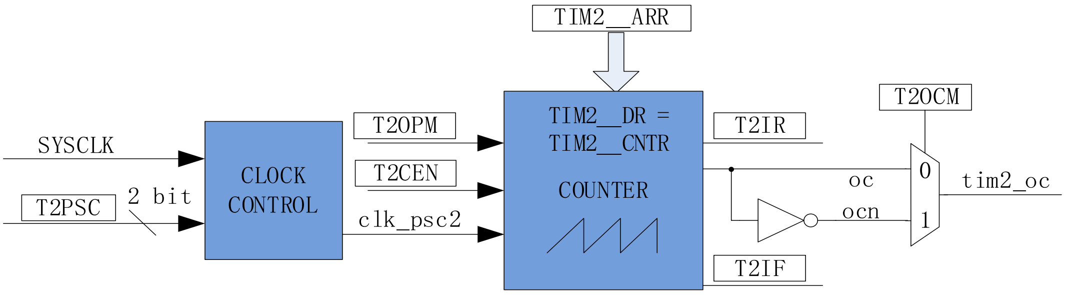


Figure 11-1 Schematic Block Diagram of Output Mode

In output mode, the base timer generates output signals according to TIM2_CR0[T2OCM], and the comparison results between TIM2__CNTR and registers TIM2__DR, TIM2__ARR. Meanwhile, corresponding interrupts are generated.

11.1.3.1 High/Low Level Output Mode

When TIM2_CR0[T2OCM] = 0, the output signal is always low if TIM2__DR > TIM2__ARR. When TIM2_CR0[T2OCM] = 1, the output signal is always high if TIM2__DR > TIM2__ARR.

11.1.3.2 PWM Generation

In PWM generation mode, TIM2__ARR determines PWM cycle, TIM2__DR determines duty cycle, and duty cycle = TIM2__DR/TIM2__ARR*100%. If TIM2_CR0[T2OCM] = 0, the low level is output when TIM2__CNTR < TIM2__DR, and the high level is output when TIM2__CNTR ≥ TIM2__DR. If TIM2_CR0[T2OCM] = 1, the high level is output when TIM2__CNTR < TIM2__DR, and the low level is output when TIM2__CNTR ≥ TIM2__DR. When TIM2__CNTR > TIM2__ARR, the output signal is reversed.

11.1.3.3 ADC Sampling

ADC sampling is triggered when TIM2_CR0[ADC_TRIG_EN] = 1 and Timer2 overflows. The sampling channel and sampling period are determined by ADC module.

11.1.3.4 Interrupt Events

- When TIM2__CNTR = TIM2__DR, a compare match event is generated and the interrupt flag bit TIM2_CR1[T2IR] is set to "1". The timer continues to count
- When TIM2__CNTR = TIM2__ARR, an overflow event is generated and the interrupt flag bit TIM2_CR1[T2IF] is set to "1". The timer is cleared to "0" and TIM2_CR0[T2OPM] decides whether it restarts. If TIM2_CR0[T2OPM] = 1, the timer stops, and if TIM2_CR0[T2OPM] = 0, it restarts.

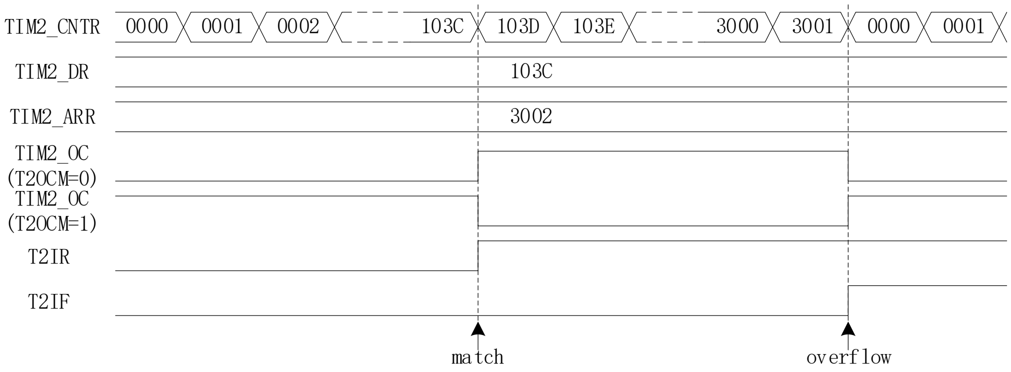


Figure 11-2 Output Mode Waveform

11.1.4 Input Signal Filtering and Edge Detection

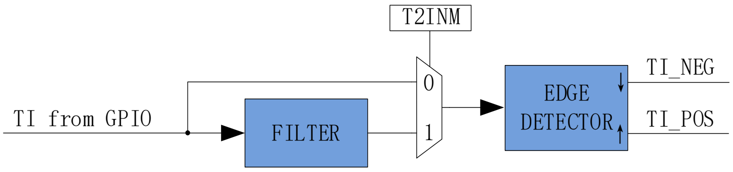


Figure 11-3 Block Diagram of Input Signal Filtering and Edge Detection

The input signal of Timer2 comes from GPIO. The filter of input signal is optional.

TIM2_CR1[T2INM] is configured to disable the filtering circuit or filter out the input noise below 4/8/16 system clock cycles. The filtered signal is 4/8/16 system clock cycles delayed than the signal before filtering.

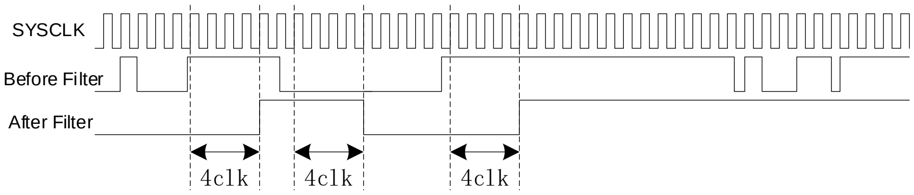


Figure 11-4 Timing Diagram of Filter Module

The edge detection module detects the filtered input signal and records the rising edge and falling edge for input capture mode.

11.1.5 Input Capture Mode

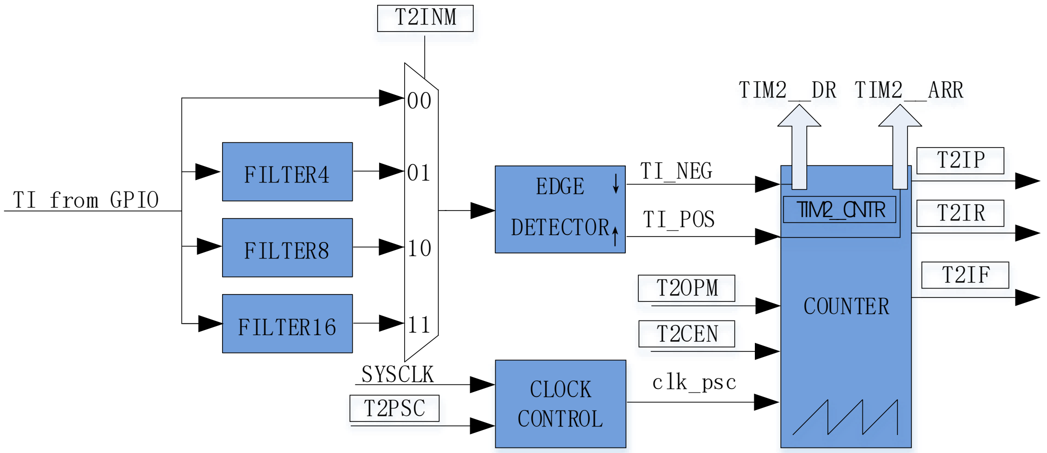


Figure 11-5 Schematic Block Diagram of Input Capture Mode

The input capture mode detects pulse width and period of the input PWM signals. When TIM2_CR0[T2OCM] = 0, the time between two adjacent rising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). When TIM2_CR0[T2OCM] = 1, the time between two adjacent falling edges forms one cycle, and the time from falling edge to rising edge forms the pulse width (LOW). The pulse width and the period obtained by TIM2__CNTR are stored in TIM2__DR and TIM2__ARR respectively.

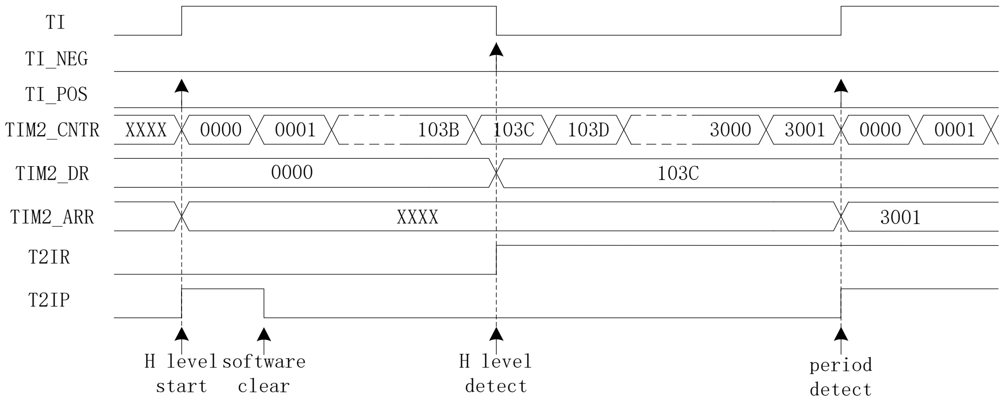


Figure 11-6 Timing Diagram of Input Capture Mode (TIM2_CR0[T2OCM] = 0)

For example, when TIM2_CR0[T2OCM] = 0, TIM2_CR1[T2CEN] is set to “1” to enable the timer. When the first rising edge of the input is detected, TIM2__CNTR is cleared to “0” and restarts. When the falling edge of the input is detected, the value of TIM2__CNTR is stored in TIM2__DR. Meanwhile, the interrupt flag bit TIM2_CR1[T2IR] is set to “1”, and TIM2__CNTR continues to count. When the second rising edge is detected, the value of TIM2__CNTR is stored into TIM2__ARR. The interrupt flag bit TIM2_CR1[T2IP] is set to “1” and TIM2__CNTR is cleared to “0”. TIM2_CR0[T2OPM] determines whether the timer restarts. If TIM2_CR0[T2OPM] = 1, the timer stops; and if TIM2_CR0[T2OPM] = 0, it restarts.

An overflow event occurs if Timer2 does not detect the second rising edge of the input and TIM2__CNTR reaches 0xFFFF. In this case, the interrupt flag bit TIM2_CR1[T2IF] is set to “1”, and TIM2__CNTR is cleared to “0”. TIM2_CR0[T2OPM] determines whether the timer restarts. If TIM2_CR0[T2OPM] = 1, the timer stops; and if TIM2_CR0[T2OPM] = 0, it restarts.

11.2 Timer2 Registers

11.2.1 TIM2_CR0 (0xA1)

Bit	7	6	5	4	3	2	1	0
Name	RSV	T2PSC		T2OCM	T2IRE	ADC_RTIG_EN	T2OPM	T2MOD
Type	—	R/W	R/W	R/W	R	—	R/W	R/W
Reset	—	0	0	0	0	—	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:5]	T2PSC	Base Timer Clock Source Frequency Division Selection This bit divides system clock as the clock source for base timer The clock source frequency after frequency division is as follows: 00: 24MHz 01: 6MHz 10: 1.5MHz 11: 375kHz
[4]	T2OCM	Output mode: Output mode selection 0: Output 0 when TIM2__CNTR < TIM2__DR; and output 1 when TIM2__CNTR ≥ TIM2__DR. 1: Output 1 when TIM2__CNTR < TIM2__DR; and output 0 when TIM2__CNTR ≥ TIM2__DR. Input capture mode: Active edge selection 0: the time between two adjacent rising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). 1: the time between two adjacent falling edges forms one cycle, and the time from falling edge to rising edge forms the pulse width (LOW).
[3]	T2IRE	Output mode: Compare match interrupt enable Input capture mode: Pulse width detected interrupt enable 0: Disable 1: Enable
[2]	ADC_RTIG_EN	ADC Sampling Enable 0: Disable 1: Enable
[1]	T2OPM	Single Mode The base timer stops when the following events occur. Output mode: The base timer overflows Input capture mode: PWM cycle detected or the base timer overflows 0: The base timer continues 1: The base timer stops (TIM2_CR1[T2CEN] is cleared to “0”)
[0]	T2MOD	Operating Mode Selection 0: Input capture mode 1: Output mode

11.2.2 TIM2_CR1 (0xA9)

Bit	7	6	5	4	3	2	1	0
Name	T2IR	T2IP	T2IF	T2IPE	T2IFE	T2INM		T2CEN
Type	R/W0	R/W0	R/W0	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description

[7]	T2IR	Output mode: Compare match interrupt flag Input capture mode: Pulse width detected interrupt flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	T2IP	Output mode: No effect Input capture mode: PWM cycle detected interrupt flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	T2IF	Output mode: Base timer overflow interrupt flag. This bit is set to “1” when TIM2__CNTR matches with the comparison value TIM2__ARR. Input capture mode: Base timer overflow interrupt flag. This bit is set to “1” when TIM2__CNTR reaches 0xFFFF and the timer does not detect the input of a PWM cycle. Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	T2IPE	Output mode: No effect Input capture mode: PWM cycle detected interrupt enable 0: Disable 1: Enable
[3]	T2IFE	Output mode: Base timer overflow interrupt enable Input capture mode: Base timer overflow interrupt enable 0: Disable 1: Enable
[2:1]	T2INM	Input Signal Filtering Pulse Width Selection When pulse width of the input signal is less than the set time, it is filtered as noise 00: No filtering 01: 4 system clock cycles 10: 8 system clock cycles 11: 16 system clock cycles
[0]	T2CEN	Base Timer Enable 0: Disable 1: Enable

11.2.3 TIM2__CNTR (0xAA, 0xAB)

TIM2__CNTRH(0xAB)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM2__CNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM2__CNTRL(0xAA)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM2__CNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:0]	TIM2__CNTR	Count value of the base timer

11.2.4 TIM2__DR (0xAC, 0xAD)

TIM2__DRH(0xAD)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM2__DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM2__DRL(0xAC)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM2__DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
Bit	**Name**	**Description**						
[15:0]	TIM2__DR	Output mode: Compare match values (written by software) Input capture mode: Count value of the detected input pulse width (written by hardware)						

11.2.5 TIM2__ARR (0xAE, 0xAF)

TIM2__ARRH(0xAF)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM2__ARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM2__ARRL(0xAE)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM2__ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
Bit	**Name**	**Description**						
[15:0]	TIM2__ARR	Output mode: PWM cycle (written by software) Input capture mode: Count value of a detected PWM cycle (written by hardware)						

12 Timer3

12.1 Timer3 Instructions

Timer3 support two working modes:

- Output mode: Generate PWM
- Input capture mode: Detect the duration of high and low level of input PWM, which can be used to calculate PWM duty cycle

Timer3 features:

- 2-bit programmable prescaler divides the system clock as the clock source for the base timer (clock source of Timer3 can be doubled to 48MHz in input capture mode)
- 16-bit up-counting base timer; the output of the prescaler serves as the counting clock source
- Input signal filtering
- Input signal edge detection
- Output PWM signal, single compare output
- Interrupt event

12.1.1 Prescaler

Frequency prescaler is used to divide the system clock and generate clock source for the basic timer. It offers 4 division coefficients and is selected by TIM3_CR0[T3PSC]. Since this register has no buffer, the clock source frequency is updated immediately after the division coefficient is written. Therefore, the frequency division coefficients shall be configured when the base timer is not working. The clock rate clk_psc3 = SYSCLK/(4^TIM3_CR0[T3PSC]). The clock rate corresponding to different TIM3_CR0[T3PSC] value as shown in Table 12-1.

Table 12-1 Mapping between Clock Rate and TIM3_CR0[T3PSC]

TIM3_CR0[T3PSC]	Coefficient	clk_psc3(Hz)	TIM3_CR0[T3PSC]	Coefficient	clk_psc3(Hz)
00	1	48M	10	16	1.5M
01	4	6M	11	64	375k

12.1.2 Reading, Writing and Counting of TIM3__CNTR

When TIM3_CR1[T3CEN] = 1, TIM2__CNTR starts to count. The write operation to TIM3__CNTR directly changes the value of the register, so the timer shall be disabled before performing the write operation. When reading TIM3__CNTR, the software reads the high-order bytes first and the hardware caches the low-order bytes simultaneously. When reading low-order bytes, the software reads the cached data.

12.1.3 Output Mode

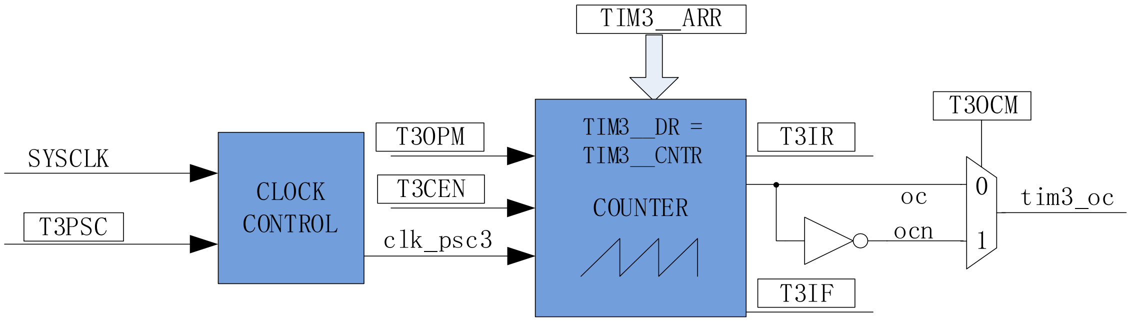


Figure 12-1 Schematic Block Diagram of Output Mode

In output mode, the base timer generates output signals according to TIM3_CR0[T3OCM], and the comparison results between TIM3__CNTR and registers TIM3__DR, TIM3__ARR. Meanwhile, corresponding interrupts are generated.

12.1.3.1 High/Low Level Output Mode

When TIM3_CR0[T3OCM] = 0, the output signal is always low if TIM3__DR > TIM3__ARR. When TIM3_CR0[T3OCM] = 1, the output signal is always high if TIM3__DR > TIM3__ARR.

12.1.3.2 PWM Generation

In PWM generation mode, TIM3__ARR determines PWM cycle, TIM3__DR determines duty cycle and duty cycle = TIM3__DR/TIM3__ARR*100%. If TIM3_CR0[T3_OCM] = 0, the low level is output when TIM3__CNTR < TIM3__DR, and the high level is output when TIM3__CNTR ≥ TIM3__DR. If TIM3_CR0[T3_OCM] = 1, the high level is output when TIM3__CNTR < TIM3__DR, and the low level is output when TIM3__CNTR ≥ TIM3__DR. When TIM2__CNTR > TIM2__ARR, the output signal is reversed.

12.1.3.3 Interrupt Events

- When TIM3__CNTR = TIM3__DR, a compare match event is generated and the interrupt flag bit TIM3_CR1[T3IR] is set to “1”. The timers continues to count
- When TIM3__CNTR = TIM3__ARR, an overflow event is generated and the interrupt flag bit TIM3_CR1[T3IF] is set to “1”. The timer is cleared to “0” and TIM3_CR0[T3OPM] decides whether it restarts. If TIM3_CR0[T3OPM] = 1, the timer stops, and if TIM3_CR0[T3OPM] = 0, it restarts.

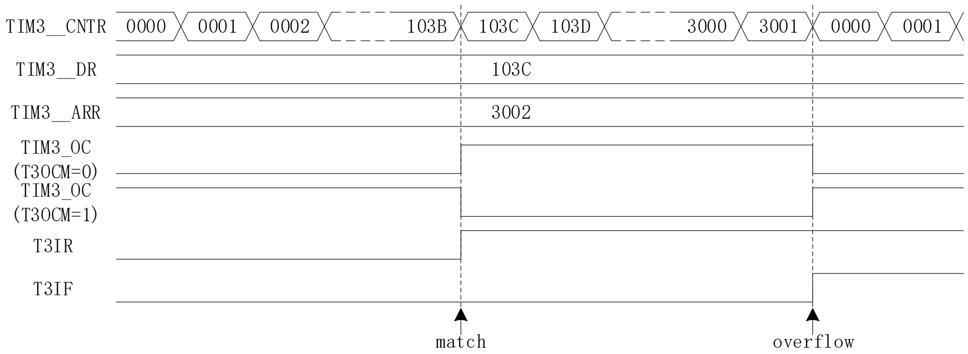


Figure 12-2 Output Mode Waveform

12.1.4 Input Signal Filtering and Edge Detection

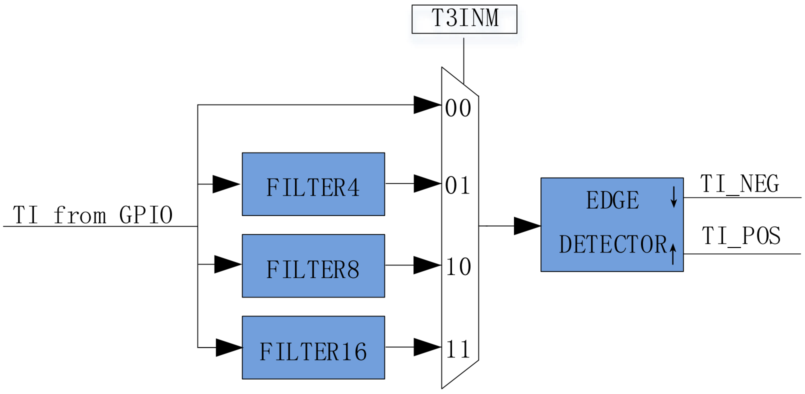


Figure 12-3 Block Diagram of Input Signal Filtering and Edge Detection

The input signal of Timer3 comes from GPIO. The filter of input signal is optional. TIM3_CR1[T3INM] is configured to disable the filtering circuit or filter out the input noise below 4/8/16 system clock cycles. The filtered signal is 4/8/16 system clock cycles delayed than the signal before filtering.

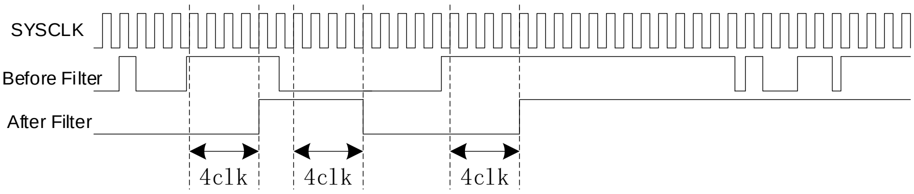


Figure 12-4 Timing Diagram of Filter Module

The edge detection module detects the filtered input signal and records the rising edge and falling edge for input capture mode.

12.1.5 Input Capture Mode

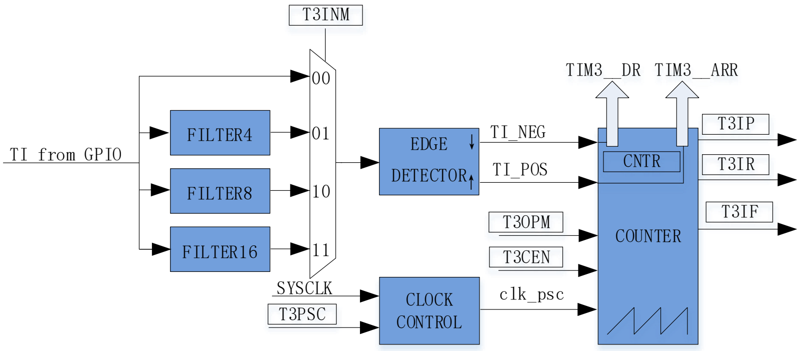


Figure 12-5 Schematic Block Diagram of Input Capture Mode

The input capture mode detects pulse width and period of the input PWM signals. When TIM3_CR0[T3OCM] = 0, the time between two adjacent rising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). When TIM3_CR0[T3OCM] = 1, the time between two adjacent falling edges forms one cycle, and the time from falling edge to rising edge forms the pulse width (LOW). The pulse width and the period obtained by TIM3__CNTR are stored in TIM3__DR and TIM3__ARR respectively.

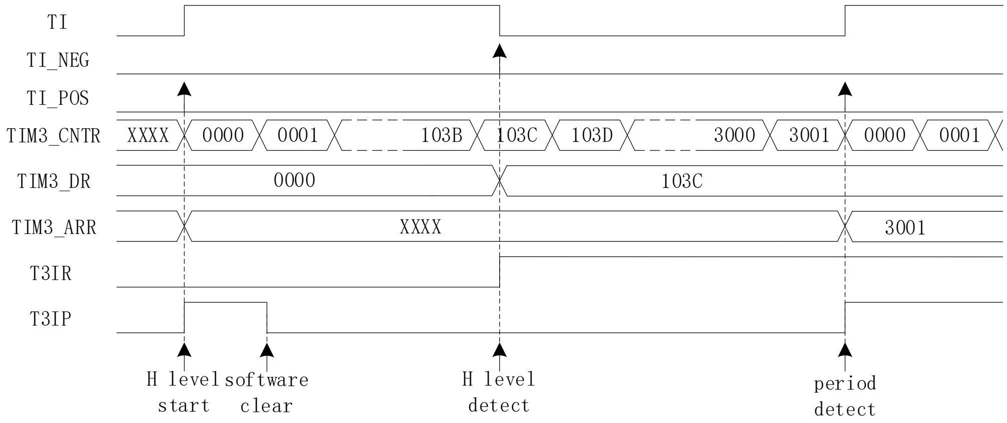


Figure 12-6 Timing Diagram of Input Capture Mode (TIM3_CR0[T3OCM] = 0)

For example, when TIM3_CR0[T3OCM] = 0, TIM3_CR1[T3CEN] = 1 is set to “1” to enable the timer. When the first rising edge of the input is detected, TIM3__CNTR is cleared to “0” and restarts. When the falling edge of the input is detected, the value of TIM3__CNTR is stored in TIM3__DR. Meanwhile, the interrupt flag bit TIM3_CR1[T3IR] is set to “1”, and TIM3__CNTR continues to count. When the second rising edge is detected, the value of TIM3__CNTR is stored into TIM3__ARR. The interrupt flag bit TIM3_CR1[T3IP] is set to “1” and TIM3__CNTR is cleared to “0”. TIM3_CR0[T3OPM] determines whether the timer restarts. If TIM3_CR0[T3OPM] = 1, the timer stops; and if TIM3_CR0[T3OPM] = 0, it restarts.

An overflow event occurs if Timer3 does not detect the second rising edge of the input and TIM3__CNTR reaches 0xFFFF. In this case, the interrupt flag bit TIM3_CR1[T3IF] is set to “1”, and TIM3__CNTR is cleared to “0”. TIM3_CR0[T3OPM] determines whether the timer restarts. If TIM3_CR0[T3OPM] = 1, the timer stops;

and if TIM3_CR0[T3OPM] = 0, it restarts.

12.2 Timer3 Registers

12.2.1 TIM3_CR0 (0x9C)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RSV	T3PSC		T3OCM	T3IRE	RSV	T3OPM	T3MOD
Type	—	R/W	R/W	R/W	R/W	—	R/W	R/W
Reset	—	0	0	0	0	—	0	0

Bit	**Name**	**Description**
[7]	RSV	Reserved
[6:5]	T3PSC	Base Timer Clock Source Frequency Division Selection This bit divides system clock as the clock source for base timer The clock source frequency after frequency division is as follows: 00: 24MHz 01: 6MHz 10: 1.5MHz 11: 375kHz Note: In the input capture mode of Timer3, 00 corresponds to 48MHz
[4]	T3OCM	Output mode: Output mode selection 0: Output 0 when TIM3__CNTR < TIM3__DR; and output 1 when TIM3__CNTR ≥ TIM3__DR 1: Output 1 when TIM3__CNTR < TIM3__DR; and output 0 when TIM3__CNTR ≥ TIM3__DR Input capture mode: Active edge selection 0: The time between two adjacent raising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). 1: The time between two adjacent falling edges forms one cycle, and the time from falling edge to raising edge forms the pulse width (LOW).
[3]	T3IRE	Output mode: Compare match interrupt enable Input capture mode: Pulse width detected interrupt enable 0: Disable 1: Enable
[2]	RSV	Reserved
[1]	T3OPM	Single Mode The base timer stops when the following events occur. Output mode: The base timer overflows Input capture mode: PWM cycle detected or the base timer overflows 0: The base timer continues 1: The base timer stops (TIM3_CR1[T3CEN] is cleared to “0”)
[0]	T3MOD	Operating Mode Selection 0: Input capture mode 1: Output mode

12.2.2 TIM3_CR1 (0x9D)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T3IR	T3IP	T3IF	T3IPE	T3IFE	T3INM		T3CEN
Type	R/W0	R/W0	R/W0	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[7]	T3IR	Output mode: Compare match interrupt flag Input capture mode: Pulse width detected interrupt flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to "0" 1: No effect
[6]	T3IP	Output mode: No effect Input capture mode: PWM cycle detected interrupt flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to "0" 1: No effect
[5]	T3IF	Output mode: Base timer overflow interrupt flag. This bit is set to "1" when TIM3__CNTR =TIM3__ARR. Input capture mode: Base timer overflow interrupt flag. This bit is set to "1" when TIM3__CNTR reaches 0xFFFF and the timer does not detect the input of a PWM cycle. Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to "0" 1: No effect
[4]	T3IPE	Output mode: No effect Input capture mode: PWM cycle detected interrupt enable 0: Disable 1: Enable
[3]	T3IFE	Output mode: Base timer overflow interrupt enable Input capture mode: Base timer overflow interrupt enable 0: Disable 1: Enable
[2:1]	T3INM	Input Signal Filtering Pulse Width Selection When pulse width of the input signal is less than the set time, it is filtered as noise 00: No filtering 01: 4 system clock cycles 10: 8 system clock cycles 11: 16 system clock cycles
[0]	T3CEN	Base Timer Enable 0: Disable 1: Enable

12.2.3 TIM3__CNTR (0xA2, 0xA3)

TIM3__CNTRH(0xA3)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM3__CNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM3__CNTRL(0xA2)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM3__CNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
Bit	**Name**	**Description**						
[15:0]	TIM3__CNTR	Count value of the base timer						

12.2.4 TIM3__DR (0xA4, 0xA5)

TIM3__DRH(0xA5)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM3__DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM3__DRL(0xA4)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM3__DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
Bit	**Name**	**Description**						
[15:0]	TIM3__DR	Output mode: Compare match values (written by software) Input capture mode: Count value of the detected input pulse width (written by hardware)						

12.2.5 TIM3__ARR (0xA6, 0xA7)

TIM3__ARRH(0xA7)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM3__ARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM3__ARRL(0xA6)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM3__ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
Bit	**Name**	**Description**						
[15:0]	TIM3__ARR	Output mode: PWM cycle (written by software) Input capture mode: Count value of a detected PWM cycle (written by hardware)						

13 Timer4

13.1 Timer4 Instructions

Timer4 support two working modes:

- Output mode: Generate PWM
- Input capture mode: Detect the duration of high and low level of input PWM, which can be used to calculate PWM duty cycle

Timer4 features:

- 2-bit programmable prescaler divides the system clock as the clock source for the base timer
- 16-bit up-counting base timer; the output of the prescaler serves as the counting clock source
- Input filter module
- Input signal edge detection
- Output PWM signal, single compare output
- Interrupt event

13.1.1 Prescaler

Frequency prescaler is used to divide the system clock and generate clock source for the base timer. It offers 4 division coefficients and is selected by TIM4_CR0[T4PSC]. Since this register has no buffer, the clock source frequency is updated immediately after the division coefficient is written. Therefore, the frequency division coefficients shall be configured when the base timer is not working. The clock rate clk_psc4 = SYSCLK/(4^TIM4_CR0[T4PSC]). The clock rate corresponding to different TIM4_CR0[T4PSC] value as shown in Table 13-1.

Table 13-1 Mapping between Clock Rate and TIM4_CR0[T4PSC]

TIM4_CR0[T4PSC]	Coefficient	clk_psc4(Hz)	TIM4_CR0[T4PSC]	Coefficient	clk_psc4(Hz)
00	1	24M	10	16	1.5M
01	4	6M	11	64	375k

13.1.2 Reading, Writing and Counting of TIM4__CNTR

When TIM4_CR1[T4CEN] = 1, TIM4__CNTR starts to count. The write operation to TIM4__CNTR directly changes the value of the register, so the timer shall be disabled before performing the write operation. When reading TIM4__CNTR, the software reads the high-order bytes first and the hardware caches the low-order bytes simultaneously. When reading low-order bytes, the software reads the cached data.

13.1.3 Output Mode

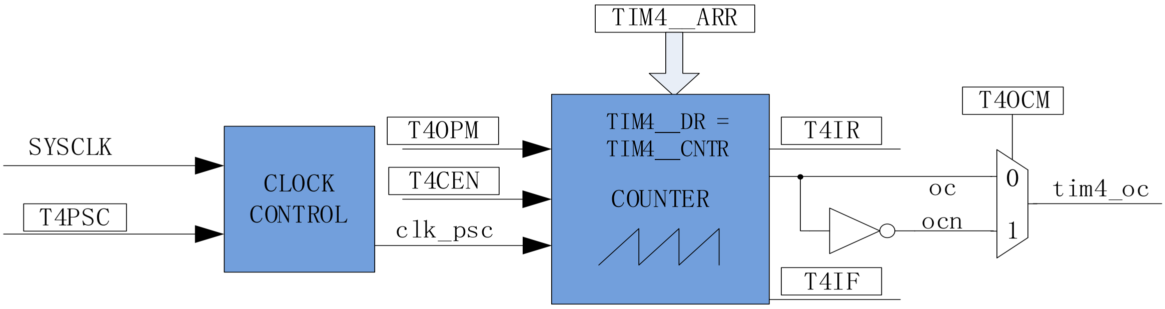


Figure 13-1 Schematic Block Diagram of Output Mode

In output mode, the base timer generates output signals according to TIM4_CR0[T4OCM], and the comparison results between TIM4__CNTR and registers TIM4__DR, TIM4__ARR. Meanwhile, corresponding interrupts are generated.

13.1.3.1 High/Low Level Output Mode

When TIM4_CR0[T4OCM] = 0, the output signal is always low if TIM4__DR > TIM4__ARR. When TIM4_CR0[T4OCM] = 1, the output signal is always high if TIM4__DR > TIM4__ARR.

13.1.3.2 PWM Generation

In PWM output mode, TIM4__ARR determines PWM period, TIM4__DR determines duty cycle, and duty cycle = TIM4__DR/TIM4__ARR*100%. TIM4_CR0[T4OCM] = 0, the low level is output when TIM4__CNTR < TIM4__DR, and the high level is output when TIM4__CNTR ≥ TIM4__DR. TIM4_CR0[T4OCM] = 1, the high level is output when TIM4__CNTR < TIM4__DR, and the low level is output when TIM4__CNTR ≥ TIM4__DR. When TIM4__CNTR is increased to TIM4__ARR, the output signal is reversed.

13.1.3.3 Interrupt Event

- When TIM4__CNTR = TIM4__DR, a compare match event is generated and the interrupt flag bit TIM4_CR1[T4IR] is set to “1”. The timers continues to count
- When TIM4__CNTR = TIM4__ARR, an overflow event is generated and the interrupt flag bit TIM4_CR1[T4IF] is set to “1”. The timer is cleared to “0” and TIM4_CR0[T4OPM] decides whether it restarts. If TIM4_CR0[T4OPM] = 1, the timer stops, and if TIM4_CR0[T4OPM] = 0, it restarts.

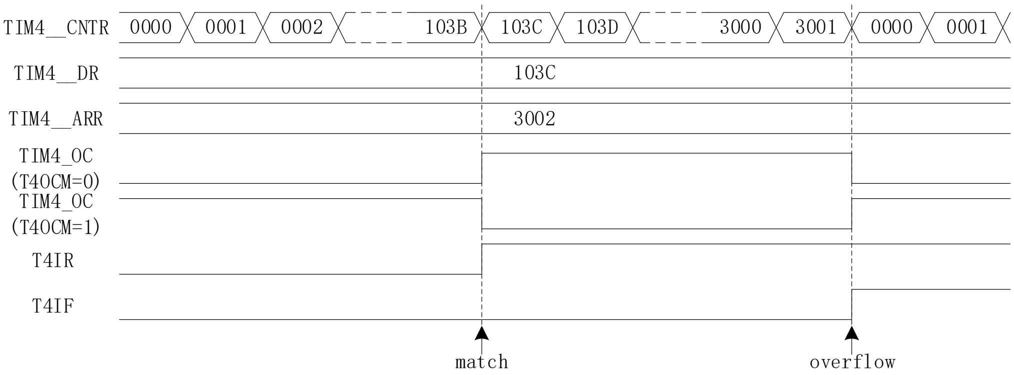


Figure 13-2 Output Mode Waveform

13.1.4 Input Signal Filtering and Edge Detection

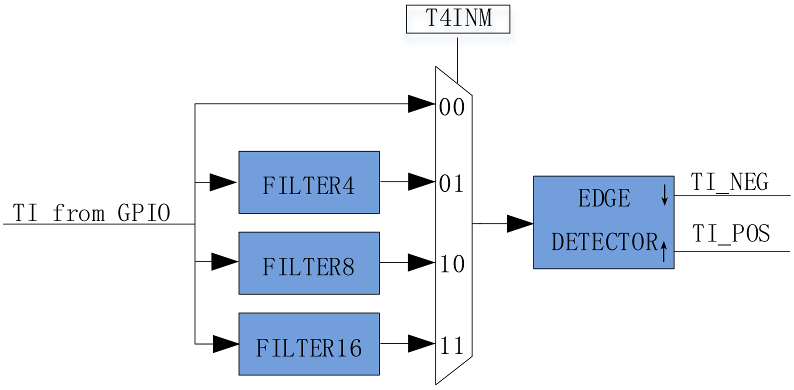


Figure 13-3 Block Diagram of Input Signal Filtering and Edge Detection

The input signal of Timer4 comes from GPIO. The filter of input signal is optional.

The edge detection module detects the filtered input signal and records the rising edge and falling edge for input capture mode.

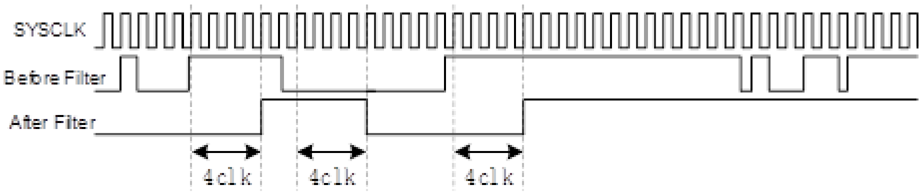


Figure 13-4 Timing Diagram of Filter Module

TIM4_CR1[T4INM] is configured to disable the filtering circuit or filter out the input noise below 4/8/16 system clock cycles. The filtered signal is 4/8/16 system clock cycles delayed than the signal before filtering.

13.1.5 Input Capture Mode

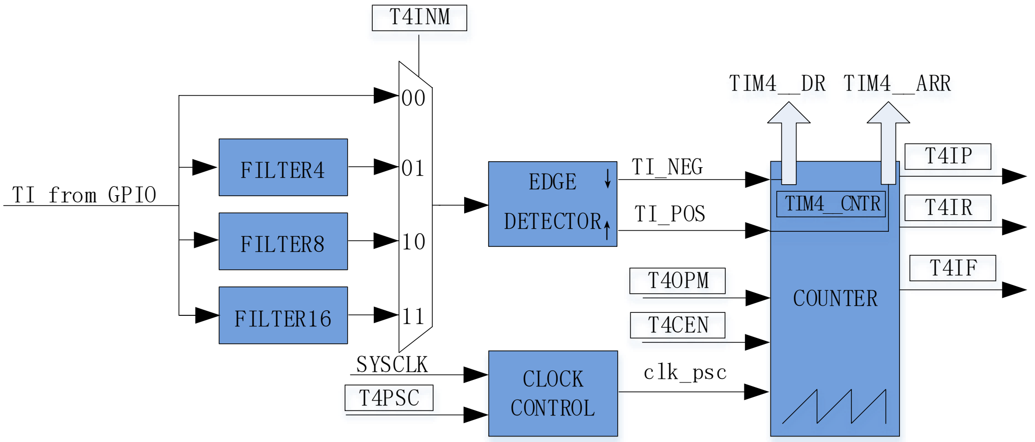


Figure 13-5 Schematic Block Diagram of Input Capture Mode

The input capture mode detects pulse width and period of the input PWM signals. When TIM4_CR0[T4OCM] = 0, the time between two adjacent rising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). When TIM4_CR0[T4OCM] = 1, the time between two adjacent falling edges forms one cycle, and the time from falling edge to rising edge forms the pulse width (LOW). The pulse width and the period obtained by TIM4__CNTR are stored in TIM4__DR and TIM4__ARR respectively.

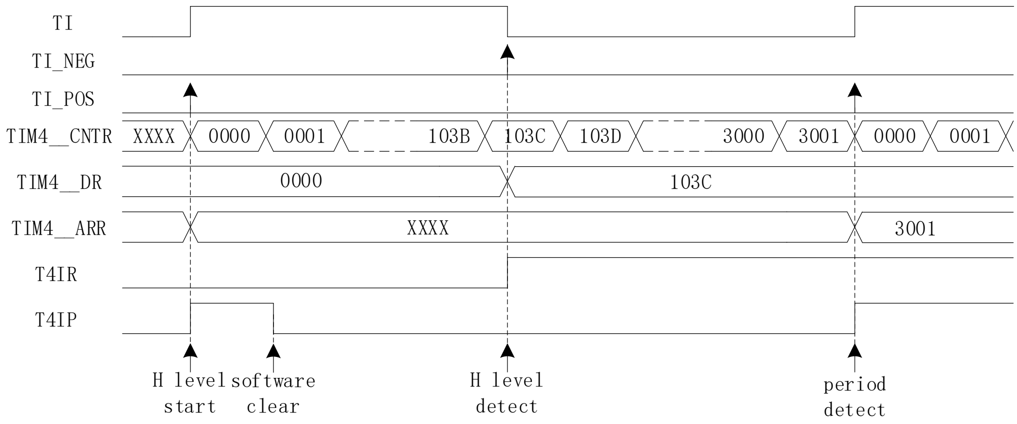


Figure 13-6 Timing Diagram of Input Capture Mode TIM4_CR0[T4OCM] = 0)

For example, when TIM4_CR0[T4OCM] = 0, TIM4_CR1[T4CEN] = 1 is set to “1” to enable the timer. When the first rising edge of the input is detected, TIM4__CNTR is cleared to “0” and restarts. When the falling edge of the input is detected, the value of TIM4__CNTR is stored in TIM4__DR. Meanwhile, the interrupt flag bit TIM4_CR1[T4IR] is set to “1”, and TIM4__CNTR continues to count. When the second rising edge is detected, the value of TIM4__CNTR is stored into TIM4__ARR. The interrupt flag bit TIM4_CR1[T4IP] is set to “1” and TIM4__CNTR is cleared to “0”. TIM4_CR0[T4OPM] determines whether the timer restarts. If TIM4_CR0[T4OPM] = 1, the timer stops; and if TIM4_CR0[T4OPM] = 0, it restarts.

An overflow event occurs if Timer3 does not detect the second rising edge of the input and TIM4__CNTR reaches 0xFFFF. In this case, the interrupt flag bit TIM4_CR1[T4IF] is set to "1", and TIM4__CNTR is cleared to "0". TIM4_CR0[T4OPM] determines whether the timer restarts. If TIM4_CR0[T4OPM] = 1, the timer stops; and if TIM4_CR0[T4OPM] = 0, it restarts.

13.2 Timer4 Register

13.2.1 TIM4_CR0 (0x9E)

Bit	7	6	5	4	3	2	1	0
Name	RSV	T4PSC		T4OCM	T4IRE	RSV	T4OPM	T4MOD
Type	—	R/W	R/W	R/W	R/W	—	R/W	R/W
Reset	—	0	0	0	0	—	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:5]	T4PSC	Base Timer Clock Source Frequency Division Selection This bit divides system clock as the clock source for base timer The clock source frequency after frequency division is as follows: 00: 24MHz 01: 6MHz 10: 1.5MHz 11: 375kHz
[4]	T4OCM	Output mode: Output mode selection 0: Output 0 when TIM4__CNTR < TIM4__DR; and output 1 when TIM4__CNTR ≥ TIM4__DR 1: Output 1 when TIM4__CNTR < TIM4__DR; and output 0 when TIM4__CNTR ≥ TIM4__DR Input capture mode: Active edge selection 0: The time between two adjacent raising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). 1: The time between two adjacent falling edges forms one cycle, and the time from falling edge to raising edge forms the pulse width (LOW).
[3]	T4IRE	Output mode: Compare match interrupt enable Input capture mode: Pulse width detected interrupt enable 0: Disable 1: Enable
[2]	RSV	Reserved
[1]	T4OPM	Single Mode The base timer stops when the following events occur. Output mode: The base timer overflows Input capture mode: PWM cycle detected or the base timer overflows 0: The base timer continues 1: The base timer stops (TIM4_CR1[T4CEN] is cleared to "0")
[0]	T4MOD	Operating Mode Selection 0: Input capture mode 1: Output mode

13.2.2 TIM4_CR1 (0x9F)

Bit	7	6	5	4	3	2	1	0
Name	T4IR	T4IP	T4IF	T4IPE	T4IFE	T4INM		T4CEN
Type	R/W0	R/W0	R/W0	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T4IR	Output mode: Compare match interrupt flag Input capture mode: Pulse width detected interrupt flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	T4IP	Output mode: No effect Input capture mode: PWM cycle detected interrupt flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	T4IF	Output mode: Base timer overflow interrupt flag. This bit is set to “1” when TIM4__CNTR =TIM4__ARR. Input capture mode: Base timer overflow interrupt flag. This bit is set to “1” when TIM4__CNTR reaches 0xFFFF and the timer does not detect the input of a PWM cycle. Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	T4IPE	Output mode: No effect Input capture mode: PWM cycle detected interrupt enable 0: Disable 1: Enable
[3]	T4IFE	Output mode: Base timer overflow interrupt enable Input capture mode: Base timer overflow interrupt enable 0: Disable 1: Enable
[2:1]	T4INM	Input Signal Filtering Pulse Width Selection When pulse width of the input signal is less than the set time, it is filtered as noise 00: No filtering 01: 4 system clock cycles 10: 8 system clock cycles 11: 16 system clock cycles
[0]	T4CEN	Base Timer Enable 0: Disable 1: Enable

13.2.3 TIM4__CNTR (0x92, 0x93)

TIM4__CNTRH(0x93)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM4__CNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM4__CNTRL(0x92)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM4__CNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
Bit	**Name**	**Description**						
[15:0]	TIM4__CNTR	Count value of the base timer						

13.2.4 TIM4__DR (0x94, 0x95)

TIM4__DRH(0x95)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM4__DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM4__DRL(0x94)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM4__DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
Bit	**Name**	**Description**						
[15:0]	TIM4__DR	Output mode: Compare match values (written by software) Input capture mode: Count value of the detected input pulse width (written by hardware)						

13.2.5 TIM4__ARR (0x96, 0x97)

TIM4__ARRH(0x97)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM4__ARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM4__ARRL(0x96)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM4__ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
Bit	**Name**	**Description**						
[15:0]	TIM4__ARR	Output mode: PWM cycle (written by software) Input capture mode: Count value of a detected PWM cycle (written by hardware)						

14 Systick

14.1 Systick Instructions

The chip can generate Systick interrupts with a fixed frequency at 1ms, and the interrupt cycle is controlled by SYST_ARR. Systick interrupt is enabled when DRV_SR[SYSTIE] is set to “1”, and the interrupts are accessed by P10.

14.2 Systick Register

14.2.1 DRV_SR (0xDF)

Bit	7	6	5	4	3	2	1	0
Name	SYSTIF	SYSTIE	RSV	DCIF	RSV		DCIM	
Type	R/W0	R/W	—	R/W0	—	—	R/W	R/W
Reset	0	0	—	0	—	—	0	0

Bit	Name	Description
[7]	SYSTIF	Systick Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	SYSTIE	Systick Interrupt Enable 0: Disable 1: Enable
[5]	RSV	Reserved
[4]	DCIF	Driver Compare Match Interrupt Flag When the Driver counter value is equal to DRV_COMR, the system decides whether to generate an interrupt according to the counting direction set by DRV_SR[DCIM] Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[3:2]	RSV	Reserved
[1:0]	DCIM	Compare Match Interrupt Mode Selection When the count value is equal to DRV_COMR, whether to generate an interrupt request is determined by DRV_SR[DCIM]. 00: No interrupt is generated. 01: Interrupt is generated when the counter counts up. 10: Interrupt is generated when the counter counts down. 11: Interrupt is generated when the counter counts up/down.

15 Driver

15.1 Driver Instructions

15.1.1 Driver Introduction

The chip has built-in pre-driver output.

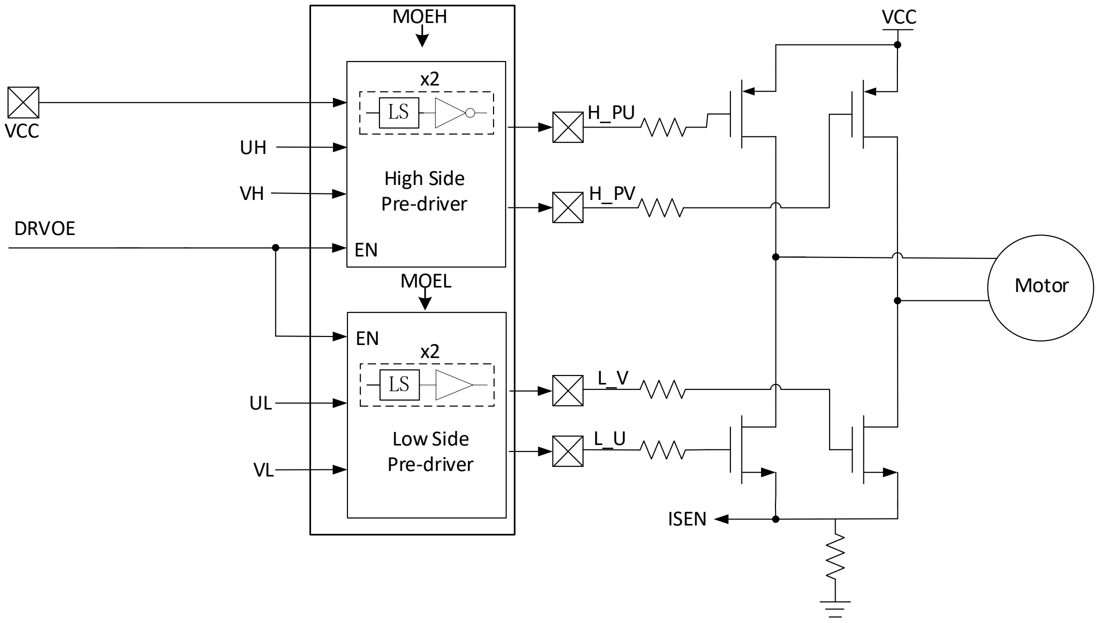


Figure 15-1 Block Diagram of Pre-driver Module

2P2N Pre-driver module is shown in Figure 15-1. UH/VH and UL/VL are the input signals of the pre-driver, and H_PU/H_PV and L_U/L_V are the output signals. H_PU/H_PV are reversely related to UH/VH, and DRV_CR[DRVOE] is the enable bit.

Pre-driver module is enabled when DRV_CR[DRVOE] is set to 1. UH/VH are reversely and sent to H_PU/H_PV for driving gate of PMOS. UL/VL are sent to L_U/L_V for driving gate of NMOS. PMOS and NMOS output voltages to drive motors.

15.1.2 Output Control Module

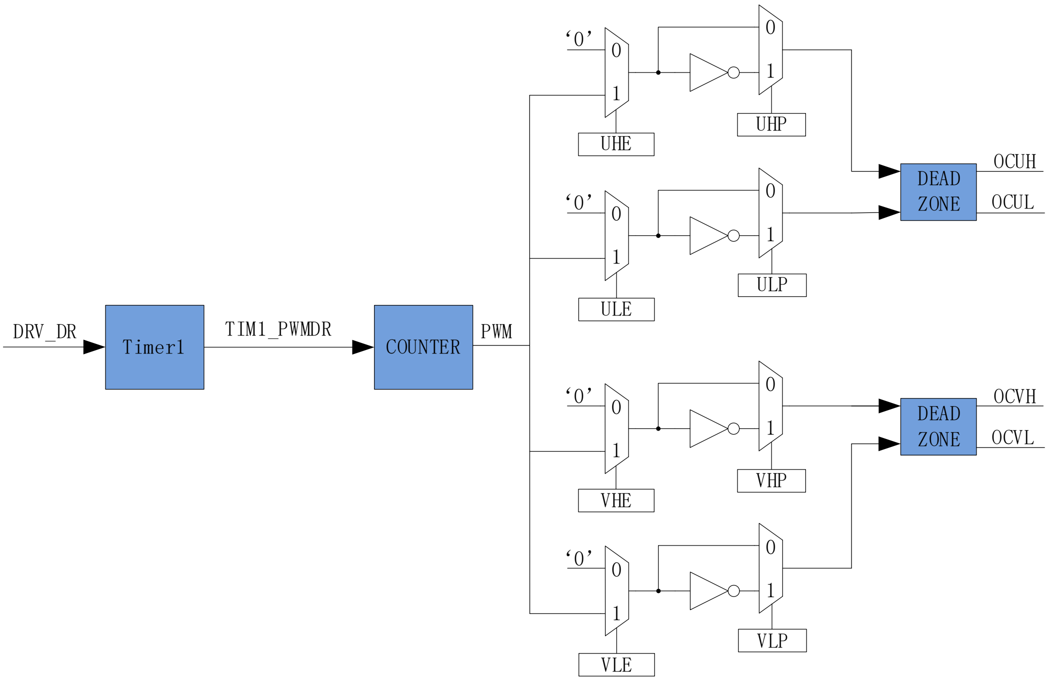


Figure 15-2 Block Diagram of Output Control Module Pre-stage

TIM1__PWMDR is obtained from DRV_DR processed by Timer1, and then it is sent to the timer for comparison to generate OCxREF, serving as the reference for the output PWM signals. If XHE/XLE is set to "1", PWM is output; while if XHP/XLP is set to "1", inverse PWM is output. Take U-phase high side as an example:

0 is output when UHE = 0 and UHP = 0;

PWM is output when UHE = 1 and UHP = 0;

1 is output when UHE = 0 and UHP = 1;

Inverse PWM is output when UHE = 1 and UHP = 1.

If PWM is output on U-phase high side, complementary output is implemented on U-phase low side and V-phase low side is normally open, then UHE = 1, UHP = 0; ULE = 1, ULP = 1; VLE = 0, VLP = 1.

15.1.2.1 Count and Compare Module

TIM1__PWMDR is sent to the timer for comparison to generate OCxREF which is used as the reference for the output PWM signals. If DRV__CNTR is lower than the comparison value, a high level is output. If it higher than the comparison value, a low level is output.

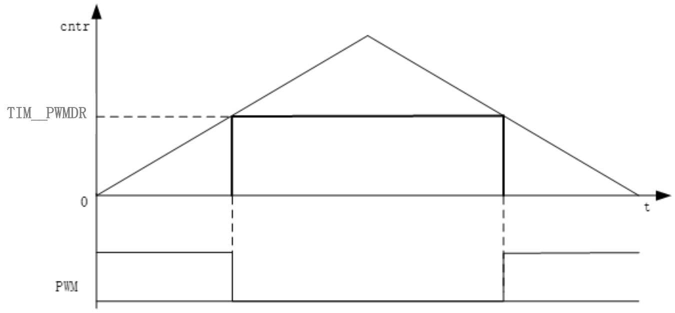


Figure 15-3 PWM Generation

PWM duty cycle = TIM1__PWMDR/16/DRV_ARR*100% (For example, if DRV_ARR = 750 and TIM1__PWMDR = 6000, then PWM duty cycle = 50%).

15.1.2.2 Deadtime Module

OCxREF has hardware deadtime insertion. Each channel has an 8-bit deadtime generator, and three channels have the same dead time, which is set by DRV_DTR. In the rising edge of OCxREF, the delay time for OCxL to generate high-level output is the one set by DRV_DTR. In the falling edge of OCxREF, the delay time for OCxH to generate high-level output is the one set by DRV_DTR. If the delay time is greater than the output pulse width, pulse width for the corresponding channel is not delayed, and pulse width for the opposite channel is not generated.

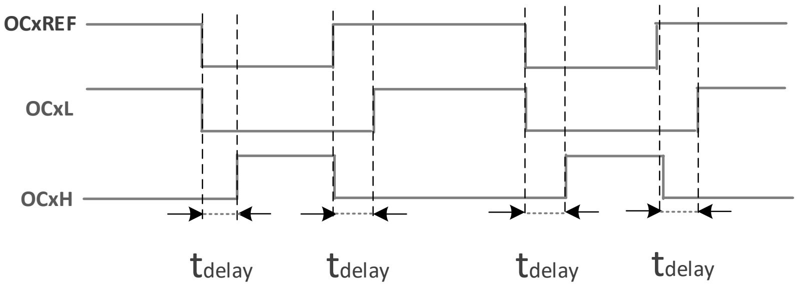


Figure 15-4 Complementary Output With Deadtime Insertion

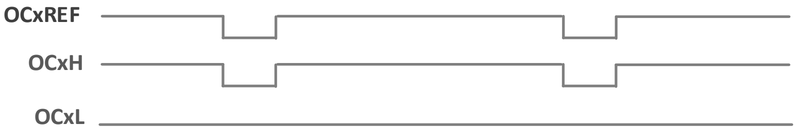


Figure 15-5 Deadtime Bigger Than Negative Level

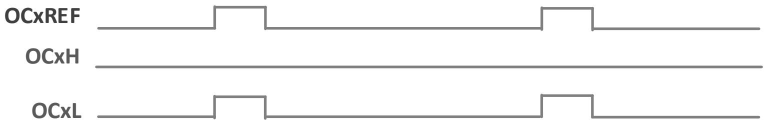


Figure 15-6 Deadtime Bigger Than Positive Level

15.1.2.3 MOEH/MOEL

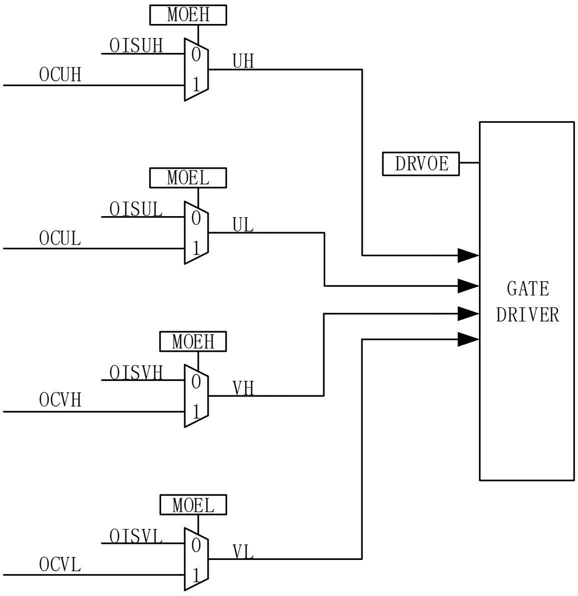


Figure 15-7 Post-stage Block Diagram of Output Control Module

After DRV_OUT[MOEH]/DRV_OUT[MOEL] is enabled, comparison value of the timer is output to drive the motor. If DRV_OUT[MOEH]/DRV_OUT[MOEL] is disabled, the idle level set by the software is output to stop the motor.

15.1.2.4 Driver Interrupt

15.1.2.4.1 Compare Match Interrupt

DRV_SR[DCIM] is configured to control overflow, underflow or down count. When DRV__CNTR = TIM1__PWMDR, a compare match interrupt is generated and the interrupt flag DRV_SR[DCIF] is set to “1” by hardware.

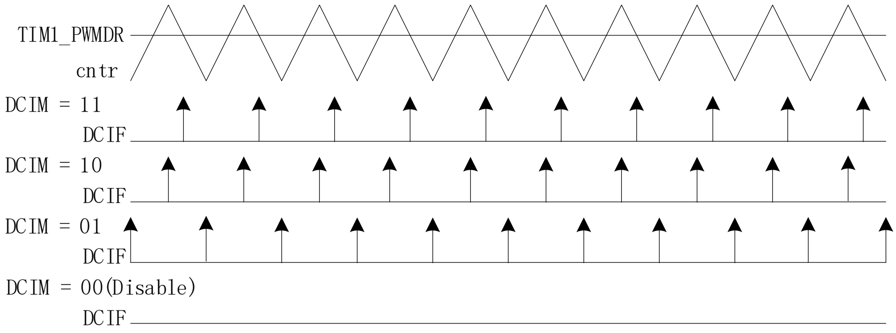


Figure 15-8 Compare Match Interrupt of Driver

15.2 Driver Register

15.2.1 DRV_CR (0xE1)

Bit	7	6	5	4	3	2	1	0
Name	DRVEN	RSV	BTIMEN	HREVEN	LREVEN	RSV		DRVOE
Type	R/W	—	R/W	R/W	R/W	—	—	R/W
Reset	0	—	0	0	0	—	—	0

Bit	Name	Description
[7]	DRVEN	Counter Enable 0: Disable 1: Enable
[6]	RSV	Reserved
[5]	BTIMEN	Base Timer Enable 0: Disable 1: Enable
[4]	HREVEN	High-side Reverse Enable: 0: Disable 1: Enable
[3]	LREVEN	Low-side Reverse enable: 0: Disable 1: Enable
[2:1]	RSV	Reserved
[0]	DRVOE	Driver Enable 0: Disable 1: Enable

15.2.2 DRV_SR (0xDF)

Bit	7	6	5	4	3	2	1	0
Name	SYSTIF	SYSTIE	RSV	DCIF	RSV		DCIM	
Type	R/W0	R/W	—	R/W0	—	—	R/W	R/W
Reset	0	0	—	0	—	—	0	0

Bit	Name	Description
[7]	SYSTIF	Systick Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect

[6]	SYSTIE	Systick Interrupt Enable 0: Disable 1: Enable
[5]	RSV	Reserved
[4]	DCIF	Driver Compare Match Interrupt Flag When the driver counter value is equal to DRV_COMR, the system decides whether to generate an interrupt according to DRV_SR[DCIM] Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to "0" 1: No effect
[3:2]	RSV	Reserved
[1:0]	DCIM	Compare Match Interrupt Mode Selection When the count value is equal to DRV_COMR, the system decides whether to generate an interrupt request according to DRV_SR[DCIM]. 00: No interrupt is generated. 01: Interrupt is generated when the counter counts up. 10: Interrupt is generated when the counter counts down. 11: Interrupt is generated when the counter counts up/down.

15.2.3 DRV_OUT (0xF8)

Bit	7	6	5	4	3	2	1	0
Name	MOEH	MOEL	RSV		OISVH	OISVL	OISUH	OISUL
Type	R/W	R/W	—	—	R/W	R/W	R/W	R/W
Reset	0	0	—	—	0	0	0	0

Bit	Name	Description
[7]	MOEH	High Side Output Enable This bit is used to select the output source of U-phase and V-phase high side signals. It can be set to "1" and cleared to "0"by software. When bus current protection is generated (see section 20.1.2), it is automatically cleared to "0" by the hardware to turn off the output. 0: Disable. Idle level set by the software is output. 1: Enable. Comparison value of the timer is output
[6]	MOEL	Low Side Output Enable This bit is used to select the output source of U-phase and V-phase low side signals. It can be set to "1" and cleared to "0"by software. When bus current protection is generated (see section 20.1.2), it is automatically cleared to "0" by the hardware to turn off the output. 0: Disable. Idle level set by the software is output. 1: Enable. Comparison value of the timer is output
[5:4]	RSV	Reserved
[3]	OISVH	Output Idle Level of VH See descriptions on DRV_OUT[OISUH].
[2]	OISVL	Output Idle Level of VL See descriptions on DRV_OUT[OISUL].
[1]	OISUH	Output Idle Level of UH This bit sets the output idle level of UH. When DRV_OUT[MOEH] = 0, it outputs idle level and disables MOS. 0: Low level 1: High level
[0]	OISUL	Output Idle Level of UL This bit sets the output idle level of UL. When DRV_OUT[MOEL] = 0, it outputs idle level and disables MOS. 0: Low level 1: High level

15.2.4 DRV_ARR (0xE4, 0xE5)

DRV_ARRH(0xE5)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				DRV_ARR[11:8]			
Type	—	—	—	—	R/W	R/W	R/W	R/W
Reset	—	—	—	—	0	0	0	0
DRV_ARRL(0x E4)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	DRV_ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:12]	RSV	Reserved
[11:0]	DRV_ARR	Counter reload value, which determines the PWM frequency (central alignment mode) Driver counter counts from 0 to DRV_ARR/2 - 1 and an overflow event occurs. Then, it counts down to 0 Calculation formula: $f_{carrier}$ = 48MHz/DRV_ARR The value of DRV_ARR is calculated at 48MHz, and the range is [0,4095] Note: LSB is always 0 and a write of "1" has no effect.

15.2.5 DRV_DR (0xE2, 0xE3)

DRV_DRH(0xE3)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				DRV_DR[11:8]			
Type	—	—	—	—	R/W	R/W	R/W	R/W
Reset	—	—	—	—	0	0	0	0
DRV_DRL(0xE2)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	DRV_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:12]	RSV	Reserved
[11:0]	DRV_DR	PWM Duty Cycle Setting Duty Cycle = DRV_DR/DRV_ARR*100% DRV_DR is also used for compare match when DRV_SR[DCIM] = 11. The value of DRV_DR is calculated at 48MHz, and the range is [0,4095) Note: When this register is used as a comparison source, PWM is referenced to high side of the bridge and a deadtime is inserted in the complementary output of the low side of bridge.

15.2.6 DRV_DTR (0xE9)

Bit	7	6	5	4	3	2	1	0
Name	DRV_DTR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	DRV_DTR	Deadtime Setting Deadtime = (DTR + 1)*T For example, when DRV_DTR = 11, Deadtime = 12*41.67ns = 500ns Note: When DRV_DTR = 0, no deadtime is inserted.

15.2.7 DRV__CNTR (0xE6, 0xE7)

DRV__CNTRH(0xE7)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				DRV__CNTR[11:8]			
Type	—	—	—	—	R/W	R/W	R/W	R/W
Reset	—	—	—	—	0	0	0	0
DRV__CNTRL(0xE6)								
Bit	7	6	5	4	3	2	1	0
Name	DRV__CNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:0]	DRV__CNTR	Counter Value The clock rate for the calculation of DRV_ CNTR is 12MHz Duty cycle of driver = DRV__CNTR*4/DRV_ARR*100% Range: [0,4095] Note: DRV__CNTR can only be written when DRV_CR[DRVEN] = 1

16 WDT

The watchdog timer (WDT) is a timer that works on the internal slow clock to monitor the master program operation and prevent the MCU running out. Watchdog works as follows: After watchdog operates, WDT starts counting. When WDT overflows, watchdog sends a signal to reset the MCU and the program restarts running from address 0. During the operation of master program, WDT has to be initialized at regular intervals to prevent WDT overflowing.

After being enabled, WDT starts counting from 0. When it reaches 0xFFFC, watchdog outputs a signal that is 4 internal slow clock cycles wide to reset the MCU, and the program starts running from address 0. WDT has to be initialized at regular intervals during operation, and the WDT rolls over to WDT_ARR setting and restarts counting.

16.1 WDT Notes

- When MCU enters standby or sleep mode, WDT stops counting, but the count values are retained.
- WDT is automatically disabled during emulation.
- RST_SR[RSTWDT] is set to “1” when MCU is reset by WDT counter overflow.

16.2 WDT Operations

1. Set CCFG1[WDT_EN] = 1 to start the WDT which then starts counting from 0;
2. Set WDT_ARR (this operation can also be performed before starting WDT);
3. Set WDT_CR[WDTRF] = 1 in the running of program. The WDT rolls over to WDT_ARR setting.

16.3 WDT Registers

16.3.1 WDT_CR (0x4026)

Bit	7	6	5	4	3	2	1	0
Name	RSV							WDTRF
Type	—	—	—	—	—	—	—	R/W
Reset	—	—	—	—	—	—	—	0

Bit	Name	Description
[7:1]	RSV	Reserved
[0]	WDTRF	WDT Initialization 0: No effect 1: WDT rolls over to the WDT_ARR setting and restarts counting.

16.3.2 WDT_ARR (0x4027)

Bit	7	6	5	4	3	2	1	0
Name	WDT_ARR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	WDT_ARR	WDT Reload Timer Set 8 high-order bits of the initialized value of the watchdog.

16.3.3 CCFG1 (0x401E)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RSV	LVWIE	WDT_EN	RSV				
Type	—	R/W	R/W	—	—	—	—	—
Reset	—	0	0	—	—	—	—	—

Bit	**Name**	**Description**
[7]	RSV	Reserved
[6]	LVWIE	LVW Interrupt Enable 0: Disable 1: Enable
[5]	WDT_EN	WDT Enable 0: Disable 1: Enable
[4:0]	RSV	Reserved

17 RTC and Clock Calibration

17.1 Functional Block Diagram of RTC

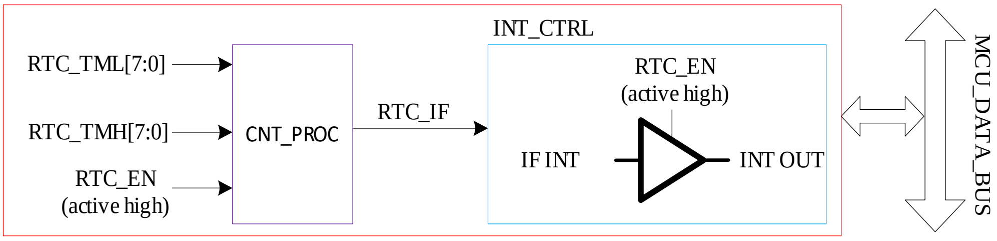


Figure 17-1 RTC Functional Block Diagram

17.2 RTC Instructions

A write to RTC_TM sets the RTC reload value. RTC is enabled when RTC_STA[RTC_EN] is set to “1”.

17.3 RTC Registers

17.3.1 RTC_TM (0x402C, 0x402D)

RTC_TMH(0x402C)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RTC_TM[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1
RTC_TML(0x402D)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RTC_TM[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	Name	Description
[15:0]	RTC_TM	RTC Register Read: Instantaneous value of counter Write: RTC up-counts at a rate of 32768Hz from 0 to the written value and becomes overflowed. Then, an interrupt request is generated, causing the timer to be cleared and restart counting.

17.3.2 RTC_STA (0x402E)

Bit	7	6	5	4	3	2	1	0
Name	RTC_EN	RTC_IF	ISOSCSEL	ISOSCEN	RSV			
Type	R/W	R/W0	R/W	R/W	—	—	—	—
Reset	0	0	0	0	—	—	—	—

Bit	Name	Description
[7]	RTC_EN	RTC Enable 0: Disable 1: Enable
[6]	RTC_IF	RTC Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0”

		1: No effect
[5]	ISOSCSEL	Internal Slow Clock Source Selection 0: Internal slow clock 1: External slow clock
[4]	ISOSCEN	Internal Slow Clock Enable 0: Disable 1: Enable
[3:0]	RSV	Reserved

17.4 Clock Calibration

17.4.1 Introduction

Clock calibration is a feature that uses the slow clock to calibrate the internal fast clock. The slow clock source is selected by RTC_STA[ISOSCSEL], which can be internal slow clock or external slow clock. Calibration principle: A 13-bit counter is used to count the length of 4 slow clock cycles with the fast clock as the clock source.

Calibration operations: Set CAL_CR0[CAL_STA] = 1 to start the calibration. Read CAL_CR0[CAL_BUSY] in software to check whether the calibration is finished. When the calibration is completed (CAL_CR0[CAL_BUSY] = 0), the readout of CAL_CR0[CAL_ARR] is the value of the length of counting 8 slow clock cycles.

17.4.2 Clock Calibration Registers

17.4.2.1 CAL_CR0 (0x403E) CAL_CR1 (0x403F)

CAL_CR0(0x403E)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	CAL_STA/ CAL_BUSY	RSV		CAL_ARR[12:8]				
Type	R/W1	—	—	R/W	R/W	R/W	R/W	R/W
Reset	1	—	—	0	0	0	0	0
CAL_CR1(0x403F)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	CAL_ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15]	CAL_STA	Clock Calibration Enable Read: 0: Calibration is completed. 1: Calibration is in progress. Write: 0: No effect. 1: Clock calibration starts.
[14:13]	RSV	Reserved
[12:0]	CAL_ARR	Calibrate Count Values The value of fast clock to count 8 slow clock cycles Note: When this value is 0, it indicates that there is no corresponding slow clock input; and when this value is 0x1FFF, it indicates a count overflows (slow clock is too slow or fast clock is too fast)

18 IO

18.1 IO Introduction

FU5821T has up to 8 GPIO pins P0.1 ~ P0.4, P0.6, P1.1 ~ P1.2 and P1.6. FU5821Q has up to 12 GPIO pins P0.1 ~ P0.6 and P1.1 ~ P1.6.

18.2 IO Instructions

Each GPIO port pin has relevant configuration registers to meet different application requirements. For example, P0.0 is mapped to register P0, and P1.0 to register P1. P0_OE and P1_OE registers are configured for digital input and output.

- The enable bits of pull-up resistors and pull-down resistors are configured to “1”. See 18.3.5 P0_PU (0x4033) ~ 18.3.6 P1_PU (0x4034) for port pins and registers.
- See 5.3 GPIO Electrical Characteristics for the values of pull-up resistors and pull-down resistors.
- The relevant bits of P0_AN and P1_AN registers are configured to “1” to enable analog mode. See 18.3.3 P0_AN (0x4031) ~ 18.3.4 P1_AN (0x4030) for port pins and registers. After the port pins are configured to analog mode, all their digital features are disabled and the port state is 0 by reading relevant bits in P0 and P1 registers.
- Pull-up resistors of P0.2 ~ P0.6, P1.1 and P1.4 ~ P1.6 are automatically disabled when the port is configured as analog mode.
- P1.1 supports anti-backflow.
- The output source of U/V-phase are the OCUH/OCVH and OCUL/OCVL signals generated by Timer1, which is configured through DRV_OUT[MOEH]/DRV_OUT[MOEL]. If it is set to “0”, idle level set by the software (DRV_OUT[OISUH]/DRV_OUT[OISVH] and DRV_OUT[OISUL]/DRV_OUT[OISVL]) is output. If it is set to “1”, PWM signal OCUH/OCVH and OCUL/OCVL is output.
- DRV_OUT[MOEH]/DRV_OUT[MOEL] can be cleared to “0” or set to “1” by software. When overcurrent protection occurs, the bit is automatically cleared to “0”.

18.3 IO registers

18.3.1 P0_OE (0xFC)

Bit	7	6	5	4	3	2	1	0
Name	RSV		P0_OE					RSV
Type	—	—	R/W	R/W	R/W	R/W	R/W	—
Reset	—	—	0	0	0	0	0	—

Bit	Name	Description
[7:6]	RSV	Reserved
[5:1]	P0_OE	P0.1 ~ P0.5 Digital Input/Output Selection 0: Input 1: Output
[0]	RSV	Reserved

18.3.2 P1_OE (0xFD)

Bit	7	6	5	4	3	2	1	0
Name	RSV	P1_OE						RSV
Type	—	R/W	R/W	R/W	R/W	R/W	R/W	—
Reset	—	0	0	0	0	0	0	—

Bit	Name	Description
[7]	RSV	Reserved
[6:1]	P1_OE	P1.1 ~ P1.6 Digital Input/Output Selection 0: Input 1: Output
[0]	RSV	Reserved

18.3.3 P0_AN (0x4031)

Bit	7	6	5	4	3	2	1	0
Name	RSV	P0_AN					RSV	
Type	—	R/W	R/W	R/W	R/W	R/W	—	—
Reset	—	0	0	0	0	0	—	—

Bit	Name	Description
[7]	RSV	Reserved
[6:2]	P0_AN	P0.2 ~ P0.6 Analog Mode Enable 0: Disable 1: Enable
[1:0]	RSV	Reserved

18.3.4 P1_AN (0x4030)

Bit	7	6	5	4	3	2	1	0
Name	RSV	P1_AN			RSV		P11_AN	HBMOD
Type	—	R/W	R/W	R/W	—	—	R/W	R/W
Reset	—	0	0	0	—	—	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:4]	P1_AN	P1.4 ~ P1.6 Analog Mode Enable 0: Disable 1: Enable

<table>
<tr><td>[3:2]</td><td>RSV</td><td>Reserved</td></tr>
<tr><td>[1]</td><td>P11_AN</td><td>P1.1 Analog Mode Enable
0: Disable
1: Enable</td></tr>
<tr><td>[0]</td><td>HBMOD</td><td>P1.3 mode configuration, which determines the functional mode of P1.3 in combination with P1_OE[3], as shown in Table 18-1.
Table 18-1 P1.3 Mode Configuration
<table>
<tr><th>HBMOD</th><th>P1_OE[3]</th><th>P1.3 mode</th></tr>
<tr><td>0</td><td>0</td><td>Digital Input</td></tr>
<tr><td>0</td><td>1</td><td>Digital Output</td></tr>
<tr><td>1</td><td>0</td><td>Analog Mode</td></tr>
<tr><td>1</td><td>1</td><td>Digital enhanced drive output mode. The maximum output current of high level output can be up to 20mA for Hall bias power supply. The drive mode of low level output is the same as that of the digital output mode.</td></tr>
</table>
</td></tr>
</table>

18.3.5 P0_PU (0x4033)

Bit	7	6	5	4	3	2	1	0
Name	RSV	P0_PU					RSV	
Type	—	R/W	R/W	R/W	R/W	R/W	—	—
Reset	—	0	0	0	0	0	—	—

Bit	Name	Description
[7]	RSV	Reserved
[6:2]	P0_PU	P0.2 ~ P0.6 Pull-up Resistor Enable 0: Disable 1: Enable
[1:0]	RSV	Reserved

18.3.6 P1_PU (0x4034)

Bit	7	6	5	4	3	2	1	0
Name	LVMOD	P1_PU						P11_PL
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	LVMOD	P1.1 1.8V Input Mode Enable 0: Disable 1: Enable
[6:1]	P1_PU	P1.1 ~ P1.6 Pull-up Resistor Enable 0: Disable 1: Enable
[0]	P11_PL	P1.1 Pull-down Resistor Enable 0: Disable 1: Enable

18.3.7 PH_SEL (0x403C)

Bit	7	6	5	4	3	2	1	0
Name	UARTSSEL	UARTEN	T4CT	T4SEL	T3SEL	T2SCT	T2SEL	RSV
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	—
Reset	0	0	0	0	0	0	0	—

Bit	Name	Description
[7]	UARTSSEL	Port Multiplexed as UART UART Function Switching 0: No function switching, with RXD serving as P1.1 and TXD as P0.1 1: Function switching, with RXD serving as P1.1 and TXDS as P1.2
[6]	UARTEN	Port multiplexed as RXD, TXD and UART enable 0: Disable 1: P1.1 and P0.1 are multiplexed as RXD and TXD, or P1.1 and P1.2 are multiplexed as RXDS and TXDS. UART is enabled.
[5]	T4CT	Timer4 Function Switching 0: No function switching, with P0.1 serving as input and output of Timer4 1: Function switching, with P1.2 serving as input and output of Timer4
[4]	T4SEL	Port multiplexed as Timer4/Timer4s 0: Disable 1: P0.1 or P1.2(function switching PH_SEL[T4CT] = 1) multiplexed as input and output of Timer4
[3]	T3SEL	Port multiplexed as Timer3 0: Disable 1: P1.1 multiplexed as input and output of Timer3
[2]	T2CT	Timer2 Function Switching 0: No function switching, with P1.2 serving as input and output of Timer2 1: Function switching, with P0.4 serving as input and output of Timer2
[1]	T2SEL	Port multiplexed as Timer2 or Timer2S 0: Disable 1: P1.2 or P0.4 (function switching PH_SEL[T2CT] = 1) multiplexed as input and output of Timer2
[0]	RSV	Reserved

18.3.8 P0 (0x80)

Port output registers P0/1 support read/write access. The RMW commands are used to access the value of the registers (see Table 18-2 for RMW commands), and other commands are used to access the PORT pin.

Bit	7	6	5	4	3	2	1	0
Name	RSV	GP06	GP05	GP04	GP03	GP02	GP01	RSV
Type	—	R/W	R/W	R/W	R/W	R/W	R/W	—
Reset	—	0	0	0	0	0	0	—

Bit	Name	Description
[7]	RSV	Reserved
[6]	GP06	Port GP06
[5]	GP05	Port GP05
[4]	GP04	Port GP04
[3]	GP03	Port GP03
[2]	GP02	Port GP02
[1]	GP01	Port GP01
[0]	RSV	Reserved

18.3.9 P1 (0x90)

Bit	7	6	5	4	3	2	1	0
Name	RSV	GP16	GP15	GP14	GP13	GP12	GP11	RSV
Type	—	R/W	R/W	R/W	R/W	R/W	R/W	—
Reset	—	0	0	0	0	0	0	—

Bit	Name	Description
[7]	RSV	Reserved

[6]	GP16	Port GP16
[5]	GP15	Port GP15
[4]	GP14	Port GP14
[3]	GP13	Port GP13
[2]	GP12	Port GP12
[1]	GP11	Port GP11
[0]	RSV	Reserved

Table 18-2 RMW Commands

Command	Description
ANL	Logic AND
ORL	Logic OR
XRL	Logic exclusive OR
JBC	ump if bit is set and clear
CPL	Complement bit
INC,DEC	Increment, decrement byte
DJNZ	Decrement and jump if not zero
MOV Px,y, C	Move carry bit to bit y of port x.y
CLR Px,y	Clear bit y of port x.y
SETB Px,y	Set bit y of port x.y

19 ADC

19.1 ADC Introduction

The ADC module is a 10-bit successive approximation register ADC with 11 channels (FU5821T has six external channels), where channel 0 ~ 9 are external pin channels and channel 10 is internal channel. VCC pin is divided by a built-in resistor with a ratio of 1/10 and the result is sampled by channel 10. The sampling mode supports sequential sampling (that is, from ADC Channel 0 to ADC channel 10 in sequence) and triggered sampling (including Timer2 triggered sampling mode). The sampling results are stored in ADCx_DR (x = 0 ~ 10) in right-aligned or left-second-highest-bit-aligned format.

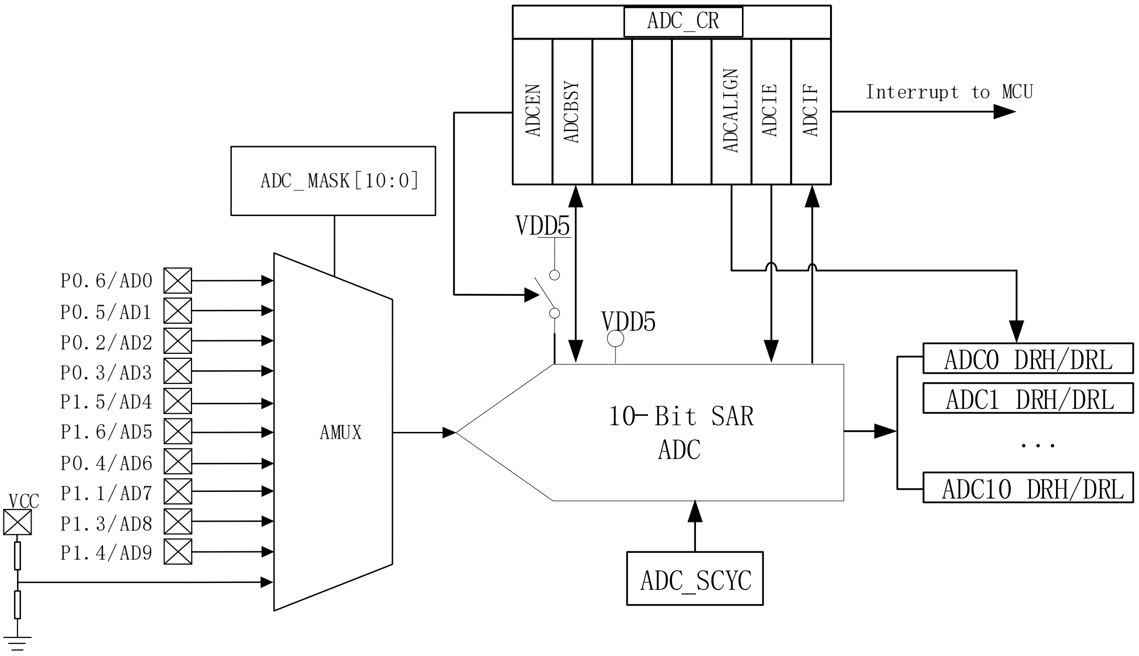


Figure 19-1 ADC Functional Block Diagram

19.2 ADC Operations

Set ADC_CR[ADCBSY] to1 as sequential sampling Mode.

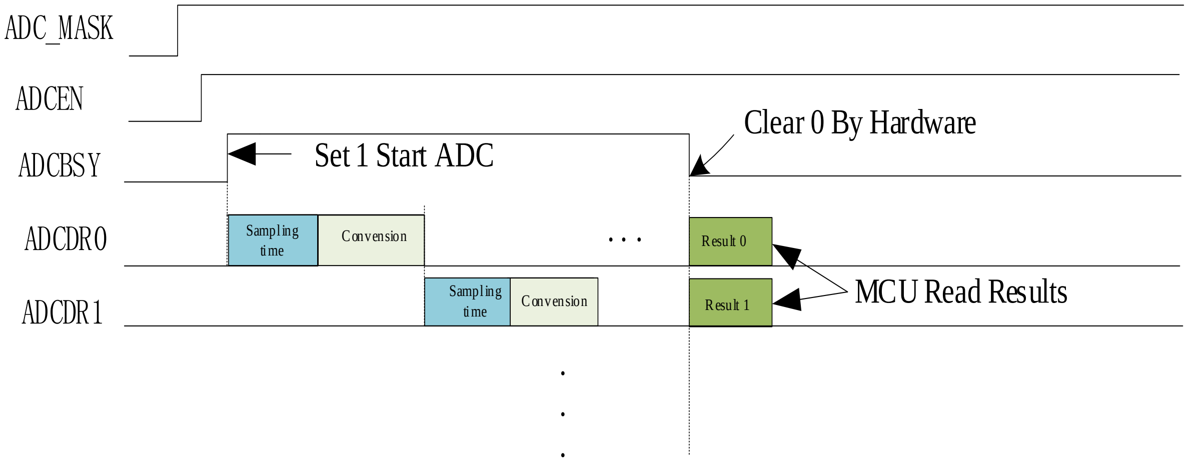


Figure 19-2 Timing Diagram of ADC Sequential Sampling Mode

ADC operations:

1. Configure ADC_MASK to enable the corresponding channel required to sample;
2. Set the sampling period of each channel (minimum value is 3);
3. Set ADC_CR[ADCEN] to “1” to enable ADC;
4. Set ADC_CR[ADCBSY] to “1” to start ADC;
5. Read ADC results When ADC_CR[ADCBSY] = 0.

19.2.1 Output Data Format

Registers ADCx_DRH and ADCx_DRL contain the high-order bytes and the low-order bytes of ADC sampling results. Data can be right-aligned or left-second-high-aligned by configuring ADC_CR[ADCALIGN]. When input voltage ranges from 0 to VDD5, the relation between the input voltage and result data is shown in Table 19-1. The bits, which are not used in ADCx_DRH and ADCx_DRL, are set to “0”.

Table 19-1 Relation between the Input Voltage and Result Data

Input Voltage	Right-aligned	Left-second-highest-aligned
0	0x0000	0x0000
VDD5/2	0x0200	0x4000
VDD5	0x03FF	0x7FE0

19.3 ADC Register

19.3.1 ADC_CR (0x4039)

Bit	7	6	5	4	3	2	1	0
Name	ADCEN	ADCBSY	RSV			ADCALIGN	ADCIE	ADCIF
Type	R/W	R/W1	—	—	—	R/W	R/W	R/W
Reset	0	0	—	—	—	0	0	0

Bit	Name	Description
[7]	ADCEN	ADC Enable 0: Disable 1: Enable
[6]	ADCBSY	ADC Start & ADC Busy Flag Read: 0: ADC idle 1: ADC busy Write: 0: No effect 1: ADC conversion starts Note: When ADC_MASK = 0, writing 1 to this bit has no effect
[5:3]	RSV	Reserved
[2]	ADCALIGN	0: ADC output is right-aligned, and ADC result is ADCx_DR[9:0] 1: ADC output is left-second-highest-bit-aligned, and ADC result is ADCx_DR[14:5]
[1]	ADCIE	ADC Interrupt Enable 0: Disable 1: Enable
[0]	ADCIF	ADC Interrupt Flag This bit is set to “1” by hardware when ADC conversion is completed Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect

19.3.2 ADC_MASK (0x4036, 0x4037)

ADC_MASKH(0x4036)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV					CH10EN	CH9EN	CH8EN
Type	—	—	—	—	—	R/W	R/W	R/W
Reset	—	—	—	—	—	0	0	0
ADC_MASKL(0x4037)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	CH7EN	CH6EN	CH5EN	CH4EN	CH3EN	CH2EN	CH1EN	CH0EN
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:11]	RSV	Reserved
[10]	CH10EN	ADC channel 10 enable
[9]	CH9EN	ADC channel 9 enable
[8]	CH8EN	ADC channel 8 enable
[7]	CH7EN	ADC channel 7 enable

[6]	CH6EN	ADC channel 6 enable
[5]	CH5EN	ADC channel 5 enable
[4]	CH4EN	ADC channel 4 enable
[3]	CH3EN	ADC channel 3 enable
[2]	CH2EN	ADC channel 2 enable
[1]	CH1EN	ADC channel 1 enable
[0]	CH0EN	ADC channel 0 enable

19.3.3 ADC_SCYC (0x4038)

ADC_SCYC(0x4038)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RSV		ADC_SCYC[5:4]		ADC_SCYC[3:2]		ADC_SCYC[1:0]	
Type	—	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	0	1	0	1	0	1

Bit	**Name**	**Description**
[7:6]	RSV	Reserved
[5:4]	ADC_SCYC [5:4]	ADC sampling cycle for ADC channels 8 ~ 10 00: 0.6μs 01: 1.3μs (default value) 10: 2μs 11: 2.6μs
[3:2]	ADC_SCYC [3:2]	ADC sampling cycle for ADC channels 4 ~ 7 00: 0.6μs 01: 1.3μs (default value) 10: 2μs 11: 2.6μs
[1:0]	ADC_SCYC [1:0]	ADC sampling cycle for ADC channels 0 ~ 3 00: 0.6μs 01: 1.3μs (default value) 10: 2μs 11: 2.6μs

19.3.4 ADC0_DR (0x0100, 0x0101)

ADC0_DRH(0x0100)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC0_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC0_DRL(0x0101)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC0_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:0]	ADC0_DR	ADC channel 0 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.5 ADC1_DR (0x0102, 0x0103)

ADC1_DRH(0x0102)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC1_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC1_DRL(0x0103)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC1_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC1_DR	ADC channel 1 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.6 ADC2_DR (0x0104, 0x0105)

ADC2_DRH(0x0104)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC2_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC2_DRL(0x0105)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC2_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC2_DR	ADC channel 2 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.7 ADC3_DR (0x0106, 0x0107)

ADC3_DRH(0x0106)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC3_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC3_DRL(0x0107)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC3_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC3_DR	ADC channel 3 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.8 ADC4_DR (0x0108, 0x0109)

ADC4_DRH(0x0108)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC4_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC4_DRL(0x0109)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC4_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC4_DR	ADC channel 4 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.9 ADC5_DR (0x010A, 0x10B)

ADC5_DRH(0x010A)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC5_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC5_DRL(0x010B)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC5_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC5_DR	ADC channel 5 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.10 ADC6_DR (0x010C, 0x010D)

ADC6_DRH(0x010C)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC6_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC6_DRL(0x010D)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC6_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC6_DR	ADC channel 6 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.11 ADC7_DR (0x010E, 0x010F)

ADC7_DRH(0x010E)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC7_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC7_DRL(0x010F)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC7_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:0]	ADC7_DR	ADC channel 7 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.12 ADC8_DR (0x0110, 0x0111)

ADC8_DRH(0x0110)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC8_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC8_DRL(0x0111)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC8_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:0]	ADC8_DR	ADC channel 8 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.13 ADC9_DR (0x0112, 0x0113)

ADC9_DRH(0x0112)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC9_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC9_DRL(0x0113)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC9_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:0]	ADC9_DR	ADC channel 9 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

19.3.14 ADC10_DR (0x0114, 0x0115)

ADC10_DRH(0x0114)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	ADC10_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC10_DRL(0x0115)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC10_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC10_DR	ADC channel 10 conversion result upon completion of ADC conversion in the Sequential Sampling Mode Data is aligned according to ADC_CR[ADCALIGN].

20 Comparator Instructions

20.1.1 Comparator CMP0

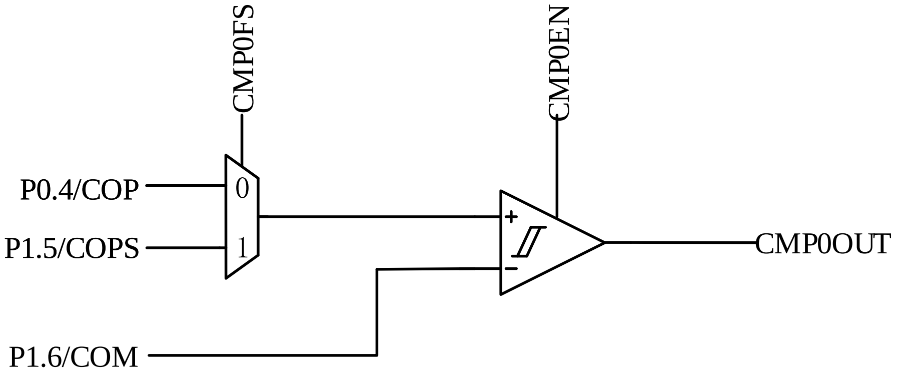


Figure 20-1 I CMP0 Input and Output Pins

CMP0 configurations are as follows:

1. Configure CMP_CR0[CMP0FS] and select P0.4 or P1.5 as the forward input port;
2. Configure CMP_CR0[CMP0EN] = 1 to enable CMP0.

20.1.2 OCP Comparator

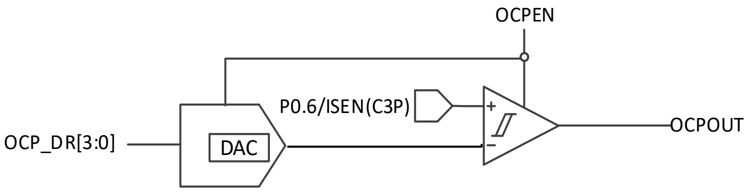


Figure 20-2 Input and Output Signal of OCP

When an OCP signal is generated, OCP comparator automatically turns off the output to protect the chip and motor. The source of OCP interrupt is selected by configuring CMP_CR2[OCP_SRC]. When CMP_CR2[OCP_SRC] = 0, the source of OCP interrupt is OCP Comparator and bus sampling voltage is input from P0.6/ISEN pin to the positive end of OCP comparator. OCP_DR register controls the OCP threshold, generates DAC voltage, and input it to the negative end of OCP comparator. Configuring CMP_CR2[OCP_EN] to “1” enables OCP feature, and OCPOUT outputs the overcurrent comparison result (i.e., OCP signal).

Configuring CMP_CR2[LOCP_DIV] enables the filtering of interrupt signals for OCP, and programming CMP_CR2[LOCP_DIV] = 00/01/10/11 selects filter width of 4/8/16/32 clock cycles. When the filtering feature is enabled, the filtered signal is delayed by 4/8/16/32 clock cycles compared to the signal before filtering.

Trigger mode of OCP interrupt and OCP event is configured by CMP_CR2[OCP_IM]. CMP_CR2[OCP_MOEH_EN] and CMP_CR2[OCP_MOEL_EN] controls high-side/low-side output after an OCP event. If CMP_CR2[OCP_MOEH_EN] is set to “1”, high-side output is turned off, and if

CMP_CR2[OCP_MOEL_EN] is set to “1”, low-side output is turned off. The two bits can be configured to “1” at the same time.

20.1.3 LCP Comparator

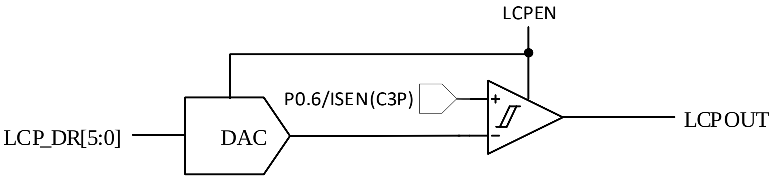


Figure 20-3 Input and Output Signal of LCP

The cycle-by-cycle current limiting feature can temporarily turn off the drive output according to current-limiting protection signal, and then automatically restores the output to protect the chip and the motor without stopping the motor.

The bus sampling voltage is input from P0.6/ISEN pin to the positive end of LCP comparator. LCP_DR register generates DAC voltage and inputs it to the negative end of LCP comparator. LCP comparator is enabled when CMP_CR1[LCP_EN] is set to “1” and CMP_SR[LCPOUT] outputs the current limiting comparison result (i.e., current-limiting protection signal).

Configuring CMP_CR2[LOCP_DIV] enables the filtering of interrupt signals for LCP, and programming CMP_CR2[LOCP_DIV] = 00/01/10/11 selects filter width of 4/8/16/32 clock cycles. When the filtering feature is enabled, the filtered signal is delayed by 4/8/16/32 clock cycles compared to the signal before filtering.

Trigger mode of LCP interrupt and LCP event is configured by CMP_CR1[LCP_IM]. CMP_CR1[LCP_MOEH_EN] and CMP_CR1[LCP_MOEL_EN] controls high-side/low-side output after an LCP event. If CMP_CR1[OCP_MOEH_EN] is set to “1”, DRV_OUT[MOEH] is cleared to “0” and high-side output is turned off, and if CMP_CR1[OCP_MOEL_EN] is set to “1”, DRV_OUT[MOEL] is cleared to “0” and low-side output is turned off. The two bits can be set to “1” at the same time. Configuring CMP_CR1[LCP_MOE_MD] = 00/01/10/11 selects one carrier cycle/half carrier cycle/9µs/4µs to automatically set DRV_OUT[MOEH] and DRV_OUT[MOEL] to “1” to recover the output.

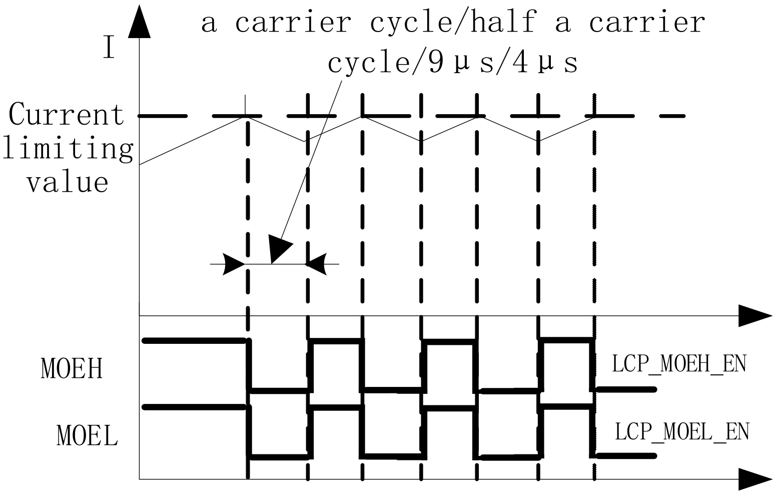


Figure 20-4 Cycle-by-Cycle Current Limiting Waveform

20.1.4 Comparator HALL_COMP

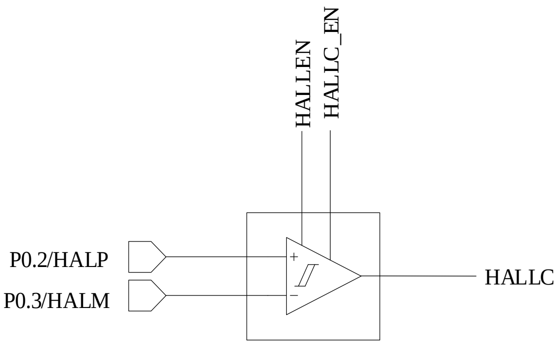


Figure 20-5 HALL_COMP Module

HALL_COMP is enabled when CMP_CR3[HALL_EN] and CMP_CR3[HALLC_EN] are set to “1”. CMP_CR3[HALL_HYS] controls comparator hysteresis.

20.2 Comparator Registers

20.2.1 CMP_CR0 (0xD5)

Bit	7	6	5	4	3	2	1	0
Name	RSV		CMP0IM		CMP0HYS		CMP0FS	CMP0EN
Type	—	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5:4]	CMP0IM	CMP0 Interrupt Mode 00: No interrupt is generated. 01: An interrupt is generated upon rising edge. 10: An interrupt is generated upon falling edge. 11: An interrupt is generated upon rising/falling edge.
[3:2]	CMP0HYS	CMP0 Hysteresis Voltage Selection: 00: No hysteresis 01: 5mV

		10: 10mV 11: 15mV
[1]	CMP0FS	CMP0 Function Switching 0: No function switching, with P0.4 serving as the positive input of CMP0 1: Function switching, with P1.5 serving as the positive input of CMP0
[0]	CMP0EN	CMP0 Enable 0: Disable 1: Enable

20.2.2 CMP_CR1 (0xD6)

Bit	7	6	5	4	3	2	1	0
Name	LCP_MOE_MD		LCP_IM		LCP_MOEH_EN	LCP_MOEL_EN	RSV	LCP_EN
Type	R/W	R/W	R/W	R/W	R/W	R/W	—	R/W
Reset	0	0	0	0	0	0	—	0

Bit	Name	Description
[7:6]	LCP_MOE_MD	Time to recover output after LCP 00: 1 carrier cycle 01: Half carrier cycle 10: 9μs 11: 4μs
[5:4]	LCP_IM	Trigger Mode of LCP Event and LCP Interrupt 00: No LCP event and LCP interrupt are triggered 01: LCP event and LCP interrupt are triggered upon rising edge 10: LCP event and LCP interrupt triggered upon falling edge 11: Only LCP event is triggered on rising edge
[3]	LCP_MOEH_EN	High-side output is turned off when LCP occurs 0: Disable 1: Enable
[2]	LCP_MOEL_EN	Low-side output is turned off enable when LCP occurs 0: Disable 1: Enable
[1]	RSV	Reserved
[0]	LCP_EN	LCP Enable 0: Disable 1: Enable

20.2.3 CMP_CR2 (0xD9)

Bit	7	6	5	4	3	2	1	0
Name	LOCP_DIV		OCP_IM		OCP_MOEH_EN	OCP_MOEL_EN	OCP_SRC	OCP_EN
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	LOCP_DIV	LCP/OCP Signal Filtering Setting 00: 160ns 01: 320ns 10: 640ns 11: 1000ns
[5:4]	OCP_IM	Trigger Mode of OCP Event and OCP Interrupt 00: No OCP event and OCP interrupt are triggered. 01: OCP event and OCP interrupt are triggered upon rising edge 10: OCP event and OCP interrupt are triggered upon falling edge 11: OCP event and OCP interrupt are triggered on both rising edge and falling edge.
[3]	OCP_MOEH_EN	High-side output is turned off when OCP occurs. 0: Disable 1: Enable

[2]	OCP_MOEL_EN	Low-side output is turned off when OCP occurs. 0: Disable 1: Enable
[1]	OCP_SRC	OCP Signal Source Selection 0: OCP 1: External interrupt INT0
[0]	OCP_EN	OCP Enable 0: Disable 1: Enable

20.2.4 CMP_CR3 (0xDA)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	HALL_HYS		HCK_SEL	HALL_EN	HALLC_EN	RSV		
Type	R/W	R/W	R/W	R/W	R/W	—	—	—
Reset	0	0	0	0	0	—	—	—

Bit	**Name**	**Description**
[7:6]	HALL_HYS	HALL_COMP Hysteresis Voltage Selection 00: No hysteresis 01: 10mV (default) 10: 20mV 11: 30mV
[5]	HCK_SEL	HALL_COMP Frequency Selection 0: 1.5MHz (default) 1: 0.75MHz
[4]	HALL_EN	HALL Comparator Mode Enable 0: Disable 1: Enable
[3]	HALLC_EN	HALLC Comparator Mode Enable 0: Disable 1: Enable
[2:0]	RSV	Reserved

20.2.5 CMP_SR (0xD7)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RSV	OCPIF	LCPIF	CMP0IF	RSV	OCPOUT	LCPOUT	CMP0OUT
Type	—	R/W0	R/W0	R/W0	—	R	R	R
Reset	—	0	0	0	—	0	0	0

Bit	**Name**	**Description**
[7]	RSV	Reserved
[6]	OCPIF	OCP Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	LCPIF	LCP Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect

[4]	CMP0IF	CMP0 Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[3]	RSV	Reserved
[2]	OCPOUT	OCP compare result
[1]	LCPOUT	LCP compare result
[0]	CMP0OUT	CMP0 compare result

20.2.6 LCP_DR (0x403A)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RSV		LCP_DR[5:0]					
Type	—	—	—	R/W	R/W	R/W	R/W	R/W
Reset	—	—	—	0	0	0	0	0

Bit	**Name**	**Description**
[7:6]	RSV	Reserved
[5:0]	LCP_DR	LCP Reference Voltage LCP reference voltage = 58mV+4.8mV*LCP_DR

20.2.7 OCP_DR (0x403B)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RSV				OCP_DR[3:0]			
Type	—	—	—	—	R/W	R/W	R/W	R/W
Reset	—	—	—	—	0	0	0	0

Bit	**Name**	**Description**
[7:4]	RSV	Reserved
[3:0]	OCP_DR	OCP Reference Voltage Setting 0000: 78mV 0001: 97mV 0010: 116mV 0011: 135mV 0100: 154mV 0101: 173mV 0110: 192mV 0111: 211mV 1000: 240mV 1001: 269mV 0010: 297mV 1011: 326mV 1100: 354mV 1101: 383mV 1110: 412mV 1111: 440mV

20.2.8 TSD_CR (0x402F)

Bit	7	6	5	4	3	2	1	0
Name	TSDEN	RSV			TSDADJ			
Type	R/W	—	—	—	R/W	R/W	R/W	R/W
Reset	0	—	—	—	1	1	1	1

Bit	Name	Description
[7]	TSDEN	Temperature Sensor Detect Enable 0: Disable 1: Enable
[6:4]	RSV	Reserved
[3:0]	TSDADJ	Over-temperature Value (Junction Temperature of the chip) 0000: 71℃ 0001: 75℃ 0010: 80℃ 0011: 84℃ 0100: 89℃ 0101: 94℃ 0110: 99℃ 0111: 105℃ 1000: 111℃ 1001: 116℃ 1010: 123℃ 1011: 131℃ 1100: 136℃ 1101: 142℃ 1110: 148℃ 1111: 155℃

21 Low Voltage Detector (LVD)

21.1 LVD Introduction

The low voltage detector is used to monitor VCC voltage and responds differently depending on the threshold, including low voltage warning, low voltage reset and under-voltage protection.

21.1.1 Low Voltage Warning

The low voltage warning feature is enabled automatically and low voltage warning threshold (7/8/9/10V) is configured by LVW Config. When LVW interrupt is enabled, it is triggered if VCC voltage is lower than the low voltage warning threshold. LVW Config and LVW Interrupt Enable are configured by Register Config in the IDE.

21.1.2 Low Voltage Reset

The low voltage reset feature is enabled automatically and low voltage reset threshold (2.8/3.0/3.5/3.8V) is configured by LVR Config. Low voltage reset is triggered when VCC voltage is lower than the low voltage reset threshold. LVR Config is configured by Register Config in the IDE.

21.1.3 Under-voltage Protection

Configuring UnderVolProtect En enables or disables the under-voltage protection feature. The under-voltage protection works when VCC voltage is lower than 3.9 ± 0.4V, and is released when VCC voltage is higher than 4.9 ± 0.4V. When under-voltage protection works, DRV works normally but cannot output the driving signal. UnderVolProtect En is configured by Register Config in the IDE.

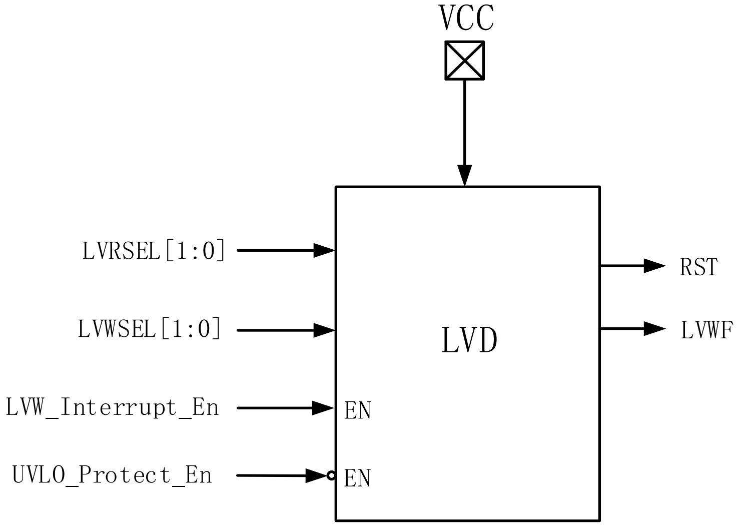


Figure 21-1 Low Voltage Detection Module

21.1.4 LVD Registers

21.1.4.1 LVSR (0xDB)

Bit	7	6	5	4	3	2	1	0
Name	EXT1CFG			EXT0CFG			LVWF	LVWIF
Type	R/W	R/W	R/W	R/W	R/W	R	R	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:5]	EXT1CFG	INT1 Pin Selection 000: P1.1 001: P1.2 010: P1.3 011: P1.4 100: P1.5 101: P1.6 110: P0.4 111: P0.5
[4:2]	EXT0CFG	INT0 Pin Selection 000: P0.1 001: P0.3 010: P0.4 011: P0.5 100: P1.1 101: P1.2 110: P1.4 111: P1.5
[1]	LVWF	VCC Low Voltage Flag This bit indicates whether the chip is in the low voltage state. 0: The chip is not in the low voltage warning state. 1: The chip is in the low voltage warning state.
[0]	LVWIF	VCC Low Voltage Interrupt Flag Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect Note: This bit is not set to “1” by hardware when LVD interrupt is disabled.

22 Flash

22.1 Flash Introduction

The chip provides 6k bytes of Flash space. It supports chip erase/write and sector erase/write.

Main features:

- 48 sectors in total, each with a size of 128 bytes
- Last sector (address range: 0x1780 to 0x17FF) cannot be erased at any time
- 120ms~150ms for chip erase and sector erase

22.2 Flash Instructions

A Flash operation is as follows:

1. Select the Flash operation by setting FLA_CR;
2. Unlock Flash operation by FLA_KEY;
3. Perform the Flash operation by movx@dptr, a;
4. Write any value to FLA_CR to lock the Flash operation.

Notes:

- All interrupts must be disabled before the operation to ensure the security of Flash operations and avoid mis-operation of Flash using MOVX instruction during interrupt processing.
- Operating the last sector triggers the Flash protection reset.
- During Flash operations, the MCU core stops working but other peripherals works normally.

22.3 Flash Registers

22.3.1 FLA_CR (0x85)

Bit	7	6	5	4	3	2	1	0
Name	RSV			FLAERR	RSV	FLAPRE	FLAERS	FLAEN
Type	—	—	—	R	—	R/W	R/W	R/W
Reset	—	—	—	0	—	0	0	0

Bit	Name	Description
[7:5]	RSV	Reserved
[4]	FLAERR	Programming Error Flag 0: Programming or pre-programming succeeds. 1: Programming or pre-programming fails.
[3]	RSV	Reserved
[2]	FLAPRE	Sector Pre-programming Enable (The sector must be pre-programmed before erasing) 0: Disable 1: Enable Note: FLA_CR[FLAPRE] works only when FLA_CR[FLAEN] is set to "1".
[1]	FLAERS	Sector Erase Enable 0: Disable 1: Enable Note: FLA_CR[FLAERS] works only when FLA_CR[FLAEN] is set to "1".
[0]	FLAEN	Programming Enable 0: Disable 1: Enable

22.3.2 FLA_KEY (0x84)

Bit	7	6	5	4	3	2	1	0
Name	FLA_KEY							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FLA_KEY	Write: Write "0x5A" and "0x1F" in sequence to unlock the Flash operation; Write any value to FLA_CR bit to lock the Flash operation.

Bit	7	6	5	4	3	2	1	0
Name	RSV						FLAKSTA	
Type	—	—	—	—	—	—	R	R
Reset	—	—	—	—	—	—	0	0

Bit	Name	Description
[7:2]	RSV	Reserved
[1:0]	FLAKSTA	Read: Flash release status 00: Locked 01: Write of 0x5A is done, waiting for 0x1F 10: Frozen 11: Released

23 CRC

23.1 Functional Block Diagram of CRC

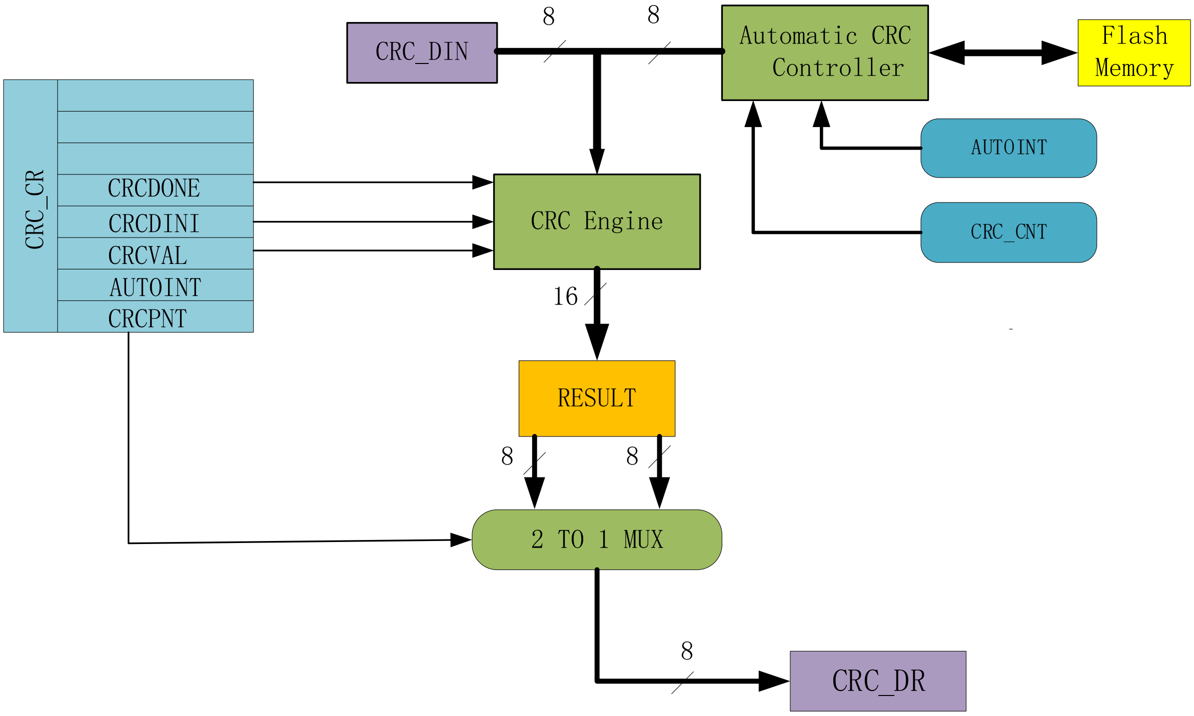


Figure 23-1 CRC Functional Block Diagram

CRC module outputs the result of CRC calculation for any 8-bit data based on a fixed polynomial. As shown in Figure 23-1, CRC receives the 8-bit data from CRC_DIN and sends the 16-bit result to the internal register after the calculation is completed. The result can be indirectly accessed through CRC_CR[CRCPNT] and CRC_DR.

Table 23-1 CRC Criteria and Polynomials

No.	CRC Criteria	Polynomials	Hexadecimal Representation
1	CRC12	x^12+x^11+x^3+x^2+x+1	0x80F
2	CRC16	x^16+x^15+x^2+1	0x 8005
3	CRC16/ CCITT-FALSE	x^16+x^12+x^5+1	0x 1021
4	CRC32	x^32+x^26+x^23+x^22+x^16+x^12 +x^11+x^10+x^8+x^9+x^5+x^4+x+1	0x04C11DB7

23.2 CRC16 Polynomial

The chip uses CRC16/CCITT-FALSE polynomial: x^16+x^12+x^5+1.

23.3 CRC16 Basic Logic Diagram

Figure 23-2 presents the schematics of CRC16. The chip implementation is based on parallel algorithm. For each input byte, MCU calculates the results within 1 system clock cycle.

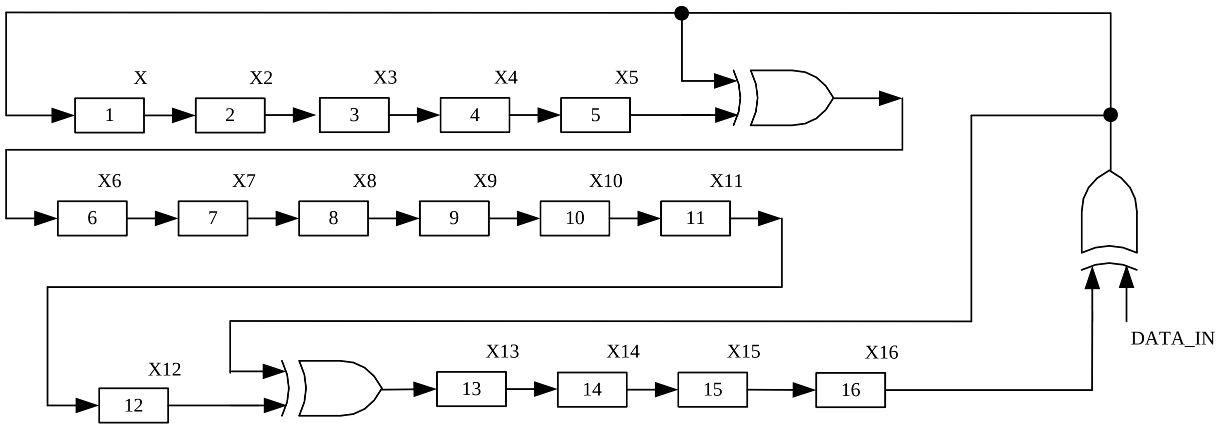


Figure 23-2 CRC16 Schematic Diagram

23.4 CRC Operations

23.4.1 CRC Calculation of Single Byte

CRC of a single byte is calculated as follows:

1. Initialize CRC_DR with two options: Configure CRC_CR[CRCVAL] and set CRC_CR[CRCDINI] to “1”, with an initial value of 0x0000 or 0xFFFF. Or configure CRC_CR[CRCPNT] and CRC_DR, where any initial value can be set.
2. Write data to CRC_DIN, and the CRC calculation is completed in the next clock cycle;
3. Read CRC results: Configure CRC_CR[CRCPNT] = 1, and read off CRC_DR in software to gets the high bytes. Configure CRC_CR[CRCPNT] = 0, and read off CRC_DR to get the low bytes.

23.4.2 CRC Calculation of ROM Sector

CRC of a continuous area of data in the ROM is calculated as follows:

1. Initialize CRC_DR, in the same way as that of single-byte CRC calculation;
2. Configure CRC_BEG to define starting sector of the ROM to be calculated;
3. Configure CRC_CNT to set the offset from the starting sector to the ending sector;
4. Write “1” to CRC_CR[AUTOINT] and keep other bits unchanged. The calculation starts automatically;
5. Read the CRC results.

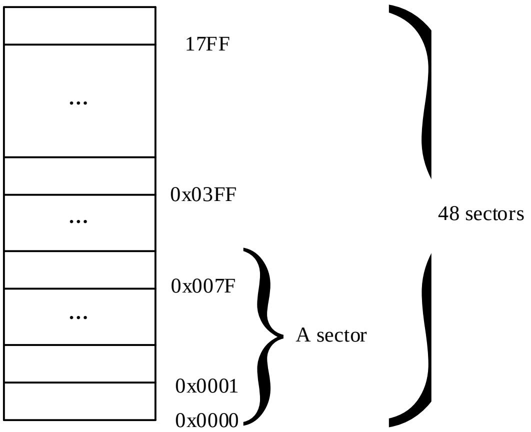


Figure 23-3 ROM Sectors

As shown in Figure 23-3, ROM has 6k bytes and is divided into 48 sectors, numbered from sector0 to sector47. Each sector contains 128 bytes. For CRC calculation of sector, the value of CRC_BEG (the starting sector) can be any value between 0x00 ~ 0x2F, including 0x00 and 0x2F. The CRC_CNT(total number of sectors to be calculated) can be 0x00 ~ 0x2F, including 0x00 and 0x2F.

As CRC_BEG increases, CRC_CNT decreases accordingly. For example, if CRC_BEG is 0x2F, CRC_CNT can be 0x00 only, i.e., the CRC value of the data in the last sector is calculated. In this case, if CRC_CNT is large, CRC controller will automatically limits the number of sectors to be calculated. Finally, CRC module only calculates CRC value of the last sector.

23.5 CRC Registers

23.5.1 CRC_CR (0x4022)

Bit	7	6	5	4	3	2	1	0
Name	RSV			CRCDONE	CRCDINI	CRCVAL	AUTOINT	CRCPNT
Type	—	—	—	R	W1	R/W	W1	R/W
Reset	—	—	—	1	0	0	0	0

Bit	Name	Description
[7:5]	RSV	Reserved
[4]	CRCDONE	CRC Sector Calculation Completion Flag During the calculation, this bit is automatically set to “0” by hardware and the software program stops. In other cases, this bit is automatically set to “1” by the hardware, so the software always returns “1” when reading this bit.
[3]	CRCDINI	CRC Result Initialization Trigger 0: No effect 1: CRC result initialization is triggered.
[2]	CRCVAL	CRC Result Initialization Selection 0: CRC result is initialized to 0x0000. 1: CRC result is initialized to 0xFFFF.
[1]	AUTOINT	CRC Sector Calculation Launch 0: No effect 1: Launch CRC batch calculation See section CRC Calculation of ROM Sector.
[0]	CRCPNT	CRC Result Pointer 0: Read CRC_DR to access 8 low-order bits of the 16-bit CRC result 1: Read CRC_DR to access 8 high-order bits of the 16-bit CRC result

Note: CRC_CR[AUTOINT] is set to “0” to perform single-byte CRC checksum.

23.5.2 CRC_DIN (0x4021)

Bit	7	6	5	4	3	2	1	0
Name	CRC_DIN							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	CRC_DIN	CRC Input Data Each time a data frame is written to this register, CRC module automatically calculates a new CRC result based on the existing CRC result, and overwrites the original one. Note: It is a virtual register, so the written data is not saved. 0x00 is returned when the address is accessed.

23.5.3 CRC_DR (0x4023)

Bit	7	6	5	4	3	2	1	0
Name	CRC_DR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	CRC_DR	CRC Result Output Each time this register is read or written, the configuration of CRC_CR[CRCPNT] determines whether to access the high or low 8 bits of the CRC result.

23.5.4 CRC_BEG (0x4024)

Bit	7	6	5	4	3	2	1	0
Name	RSV	CRC_BEG						
Type	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	0	0	0	0	0	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:0]	CRC_BEG	First ROM Sector Pending Auto CRC Calculation Example: If CRC_BEG is set to “1”, CRC calculation starts from location 1*128 = 128, or rather from the first byte of sector 2.

23.5.5 CRC_CNT (0x4025)

Bit	7	6	5	4	3	2	1	0
Name	RSV	CRC_CNT						
Type	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	0	0	0	0	0	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:0]	CRC_CNT	Offset of Sector Pending Automatic CRC Calculation This bit defines the offset of ROM sector for CRC calculation and determines the last sector pending CRC calculation.

24 Sleep Mode

24.1 Introduction

The chip operates in three modes: Normal, Standby and Sleep. These modes are selected by setting PCON[IDLE] and PCON[STOP].

The operating states of the module under different power modes are summarized in Table 24-1.

Table 24-1 Power Modes

Power Mode	Description	Wake-up Source	Power Consumption Performance
Normal	All modules work at full speed except for peripherals that are disabled	NA	High power consumption with best performance
Standby	CPU clock stops and the other functional modules are enable or disabled, depending on their control bit setting. Watchdog Timer stops.	Any interrupt, Reset/Debug on external interrupt	Low power consumption with flexible performance
Sleep	Flash Deep Sleep. The analog fast clock circuit is disconnected and software shall be operated to check if ADC, FOC and driver modules are disabled before the chip enters the Sleep mode. Watchdog Timer is disabled.	External Interrupt, Reset/Debug on external interrupt	Extremely low power consumption with flexible performance

Note: It is recommended to insert 3 null statements in the sleep mode.

```
PCON = 0x02;
_nop_();
_nop_();
_nop_();
```

24.2 Sleep Mode Register

24.2.1 PCON (0x87)

Bit	7	6	5	4	3	2	1	0
Name	RSV		GF3	GF2	GF1	RSV	STOP	IDLE
Type	—	—	R/W	R/W	R/W	—	R/W	R/W
Reset	—	—	0	0	0	—	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	GF3	General-purpose flag bit 3
[4]	GF2	General-purpose flag bit 2
[3]	GF1	General-purpose flag bit 1
[2]	RSV	Reserved
[1]	STOP	A write of “1” makes the chip enter the sleep mode. The bit is automatically cleared to “0” by hardware after wakeup.
[0]	IDLE	A write of “1” makes the chip enter the standby mode. The bit is automatically cleared to “0” by hardware after wakeup.

Power consumption mode PCON[STOP:IDLE]:

00: Normal

01: Standby

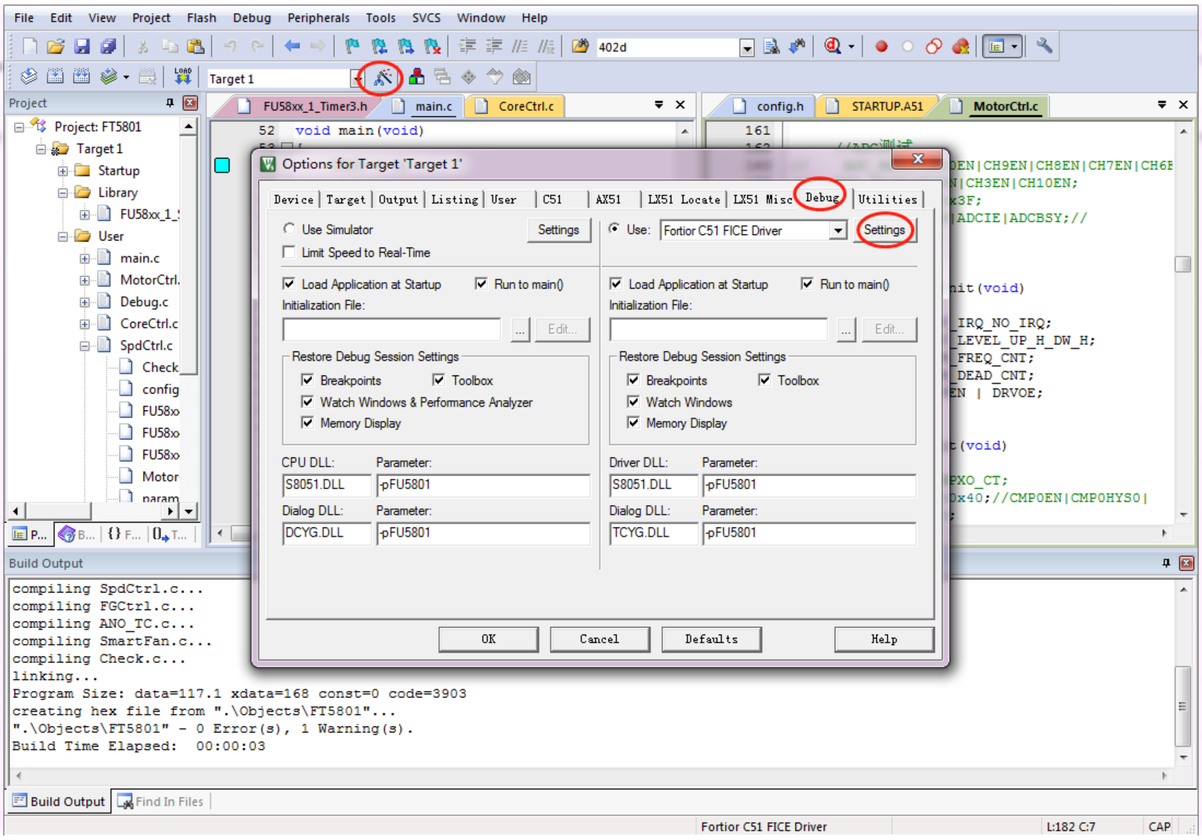
1X: Sleep

25 Code Protection

25.1 Introduction

The chip supports full Flash space encryption to protect your software intellectual property and avoid unauthorized access. When Flash memory is encrypted, the data inside cannot be read, and data consistency can be evaluated by CRC check module only.

25.2 Operating Instructions



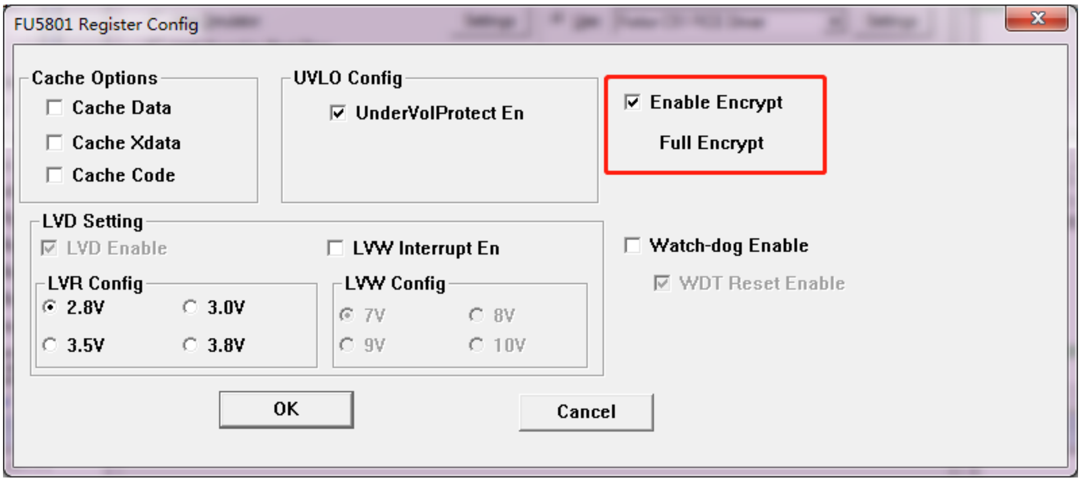
Options for Target 'Target 1' — Debug

○ Use Simulator
□ Limit Speed to Real-Time
☑ Load Application at Startup ☑ Run to main()
Restore Debug Session Settings: ☑ Breakpoints ☑ Toolbox ☑ Watch Windows & Performance Analyzer ☑ Memory Display
CPU DLL: S8051.DLL Parameter: -pFU5801
Dialog DLL: DCYG.DLL Parameter: -pFU5801

● Use: Fortior C51 FICE Driver
☑ Load Application at Startup ☑ Run to main()
Restore Debug Session Settings: ☑ Breakpoints ☑ Toolbox ☑ Watch Windows ☑ Memory Display
Driver DLL: S8051.DLL Parameter: -pFU5801
Dialog DLL: TCYG.DLL Parameter: -pFU5801

```
compiling SpdCtrl.c...
compiling FGCtrl.c...
compiling ANO_TC.c...
compiling SmartFan.c...
compiling Check.c...
linking...
Program Size: data=117.1 xdata=168 const=0 code=3903
creating hex file from ".\Objects\FT5801"...
".\Objects\FT5801" - 0 Error(s), 1 Warning(s).
Build Time Elapsed:  00:00:03
```

Figure 25-1 Code Protection Configurations



Cache Options: □ Cache Data □ Cache Xdata □ Cache Code

UVLO Config: ☑ UnderVolProtect En

☑ Enable Encrypt
Full Encrypt

LVD Setting: ☑ LVD Enable □ LVW Interrupt En

LVR Config: ● 2.8V ○ 3.0V ○ 3.5V ○ 3.8V

LVW Config: ● 7V ○ 8V ○ 9V ○ 10V

□ Watch-dog Enable
☑ WDT Reset Enable

Figure 25-2 Full Code Protection Mode

Operation steps are as follows:

1. Start 8051 IDE, enter Target Options and select Debug tab. As shown in Figure 25-1, click Settings to proceed with the setting;
2. Select the options as shown in Figure 25-2, and click OK. Then compile the project and download it. Get the BIN file and program it to Flash.

26 Revision History

Rev.	Description	Date	Prepared By
V1.0	1. First release, corresponding to internal version V0.07; 2. Added descriptions on features, registers, etc.	2021/03/22	Kun Li
V1.1	3. Added descriptions on VCC slope; 4. Modified VCC capacitor value from 10μF to 2.2μF; 5. Added VCC limit voltage 24V (<1min).	2021/04/09	Kun Li
V1.2	1. Modified some mistakes based on feedback from Taiwan customers; 2. Deleted descriptions on table look-up mode; 3. Deleted descriptions on HALLPN.	2021/11/17	Kun Li
V1.3	Updated all chapters	2022/05/27	Michelle Jiang
V1.4	1. Added descriptions on FU5821Q in sections 1.1 Features, 1.4 Functional Block Diagram, 2 Pin Definitions, 3 Package Information, 4 Ordering Information, 5.8 Package Thermal Characteristics, 18.1 IO Introduction; 2. Updated Functional Block Diagram of FU5821T; 3. Deleted VCC limit voltage in section 5.2 Global Electrical Characteristics; 4. Added “If there are any differences between the Chinese and the English contents, please take the Chinese version as the standard.” in Copyright Notice.	2023/06/02	Eric Deng
V1.5	1. Modified “6K” of FLASH (kByte) as “6” in section 4 Ordering Information; 2. Added High-side Output High Level, High-side Output Low Level, Low-side Output High Level and Low-side Output Low Level in section 5.4 Pre-driver IO Electrical Characteristics; 3. Updated descriptions in section 18 IO; 4. Modified “CRC16/CCITT” in Table 23-1 CRC Criteria and Polynomials and section 23.2 CRC 16 Polynomial as “CRC16/CCITT-FALSE”; 5. Modify I2C in descriptions as “I^{2}C”; 6. Modified the document No. from “MCU-DS-18” to “MCU-DS-19”; 7. Proofread the overall document; 8. Corrected some grammatical mistakes and wrong sentences; 9. Standardized the document format.	2023/07/31	Eric Deng
V1.6	1. Corrected the package of FU5821Q QFN20 as QFN24; 2. Updated IO Type and description of RXD pin; 3. Updated section 3 Package Information; 4. Modified the number of DAC in section 4 Ordering Information from 4 to 2, and TSSOP20 (4.96x6.40mm) as TSSOP16 (4.4x5.0mm); 5. Added the bit “T1HREN” in 10.2.6 TIM1_CR5 (0xB6); 6. Added “The relevant registers shall be enabled before using the UART feature. See section 18.3.7 PH_SEL (0x403C) for details.” in section 9.2 URT Instructions; 7. Corrected “1100: 3548mV” for OCP Reference Voltage Setting in section 20.2.7 OCP_DR (0x403B) as “1100: 354mV”.	2023/08/08	Eric Deng

Copyright Notice

Right to make changes — Fortior Technology (Shenzhen) Co., Ltd. reserves the right to make changes in the products - including circuits, standard cells, and/or software - described or contained herein in order to improve design and/or performance. The information contained in this manual is provided for the general use by our customers. Our customers shall ensure that they take appropriate action so that their use of our products does not infringe upon any patents. It is the policy of Fortior Technology (Shenzhen) Co., Ltd. to respect the valid patent rights of third parties and not to infringe upon or assist others to infringe upon such rights.

If there are any differences between the Chinese and the English contents, please take the Chinese version as the standard.

Fortior Technology (Shenzhen) Co., Ltd.

Room 203, 2/F, Building No.11, Keji Central Road 2,

Software Park, High-Tech Industrial Park, Shenzhen, P.R. China 518057

Tel: 0755-26867710

Fax: 0755-26867715

URL: http://www.fortiortech.com