Datasheet

Three-phase Motor Control MCU
FU6813_63

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.
- The symbol "x" in a register name indicates similar registers. For example, TIMx_CR0 indicates TIM3_CR0 and TIM4_CR0.
- [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. 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 uncertainty value or invalid 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 to Digital Convertor
BEMF	Back Electromotive Force
BLDC	Brushless Direct Current
CRC	Cyclic Redundancy Check
DAC	Digital to 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
IC	Integrated Circuit
I^2C	Inter Integrated Circuit
IRAM	Internal RAM
IDE	Integrated Development Environment
LDO	Low Dropout Regulator
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
MR Sensor	Magnetoresistive Sensor
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
UART	Universal Asynchronous Receiver/Transmitter
WDT	Watch Dog Timer
XRAM	External RAM
XSFR	External SFR

1 System Introduction

1.1 Features

- Power supply of FU6813L/FU6813P:
  - High-voltage single-power supply mode (VCC_MODE = 0): VCC = 5V ~ 24V
  - Low-voltage single-power supply mode (VCC_MODE = 1): VCC = VDD5 = 3V ~ 5.5V
  - Dual-power supply mode (VCC_MODE = 1), VCC ≥ VDD5: VCC = 5V ~ 36V, VDD5 = 5V
- Power supply of FU6813N:
  - High-voltage single-power supply mode: VCC = 5V ~ 24V
  - Low-voltage single-power supply mode: VCC = VDD5 = 3V ~5.5V
- Power supply of FU6863Q:
  - Mode 1: VCC_MODE = 0 and VCC = 5V ~24V
  - Mode 2: VCC_MODE = 1, VCC = 5V ~ 24V and VDD5 = 3V ~ 5.5V
- Dual core: 8051 core and ME core
- An instruction cycle mostly takes 1 or 2 system clock cycle(s)
- 32KB Flash ROM with CRC, self-program and code protection
- 256 bytes IRAM and 1.5K bytes XRAM
- ME: Core integrating LPF module, PI regulator, BLDC module and FOC module
- 1T 16x16 multiplier, 16T 32/16 divider
- 15 interrupt sources with 4 configurable priority levels
- Number of GPIOs:

  FU6813L: 34

  FU6813N: 20

  FU6813P: 35

  FU6863Q: 32
- Timers:
  - 2*Programmable timers with capture feature
  - 1*QEP decoding programmable timer
  - 1*General-purpose timer
  - 1*RTC
- 1*SPI
- 1*I^2C
- 2*UARTs
- Dual-channel DMA: supporting data transmission via I^2C/SPI/UART
- Analog Peripherals:

- 12-bit ADC, operating with 1μs conversion time and internal VREF or external VREF selectable as reference voltage
- Number of ADC channels:
  - FU6813L: 14
  - FU6813N: 9
  - FU6813P: 14
  - FU6863Q: 14
- Internal VREF. 3V, 4V, 4.5V and VDD5 can be selected as the internal reference
- Internal VHALF, with 1/2 VREF as the internal reference
- 4*Standalone operational amplifiers (FU6813N: 2*standalone operational amplifier)
- 4-channel analog comparator
- DAC: Single-channel 9-bit, single-channel 8-bit, single-channel 6-bit

- Driver type:
  - FU6813: PWM output
  - FU6863: 6N Pre-driver
- Automatic commutation, cycle-by-cycle current limiting and Hall/BEMF-based position sensing for BLDC motor control
- FOC module supports single/dual/triple-shunt current sampling (For FU6813N, FOC module supports single-shunt current sampling)
- FOC module supports overmodulation
- PFC
- System clock
  - Built-in 24MHz ± 2% high-speed clock
  - 32.8kHz low-speed clock
  - 32768Hz crystal clock
- WDT
- Two-wire FICE protocol based in-circuit emulation

1.2 Applications

The chip can be used for the drive of sensorless or sensored BLDC/PMSM motors, single-phase/three-phase induction motors and servo motors.

Applications: Range hoods, air conditioner indoor units, ceiling fans, pedestal fans, vacuum cleaners, hair dryers, industrial fans, water pumps, compressors, electric vehicles, power tools, drones, etc.

1.3 Overview

The high-performance motor drive chip incorporates ME core and 8051 core. ME core integrates FOC, MDU, LPF, PI SVPWM/SPWM and PFC modules that allow for automatic calculation of FOC or square-wave control by the hardware for sensored/sensorless BLDC motors. 8051 core is used for parameter configuration and routine processing. Most of 8051 core instruction cycle takes 1T or 2T clock cycle(s). The dual cores work in parallel to achieve high-performance motor control. The chip integrates high-speed operational amplifiers, comparators, high-speed ADC, multiplier/divider, CRC, SPI, I^2C, UART, Timers, PWM modules and built-in high-voltage LDO, which are suitable for FOC or square-wave based BLDC/PMSM motors.

The above are the general descriptions on the product family. The features vary by models. For details, see section 2 Pin Definitions and 4 Ordering Information.

For concise description and easy differentiation, if it is specified that a feature is applied to a specific model, the feature is exclusive to this model. For example, FU6813 refers to all models of FU6813. Otherwise, the feature is a common feature of the product family.

FU6813 includes FU6813L(LQFP48), FU6813N(QFN32) and FU6813P(LQFP52).

FU6863 includes FU6863Q(QFN56).

1.4 Functional Block Diagram

1.4.1 FU6813L

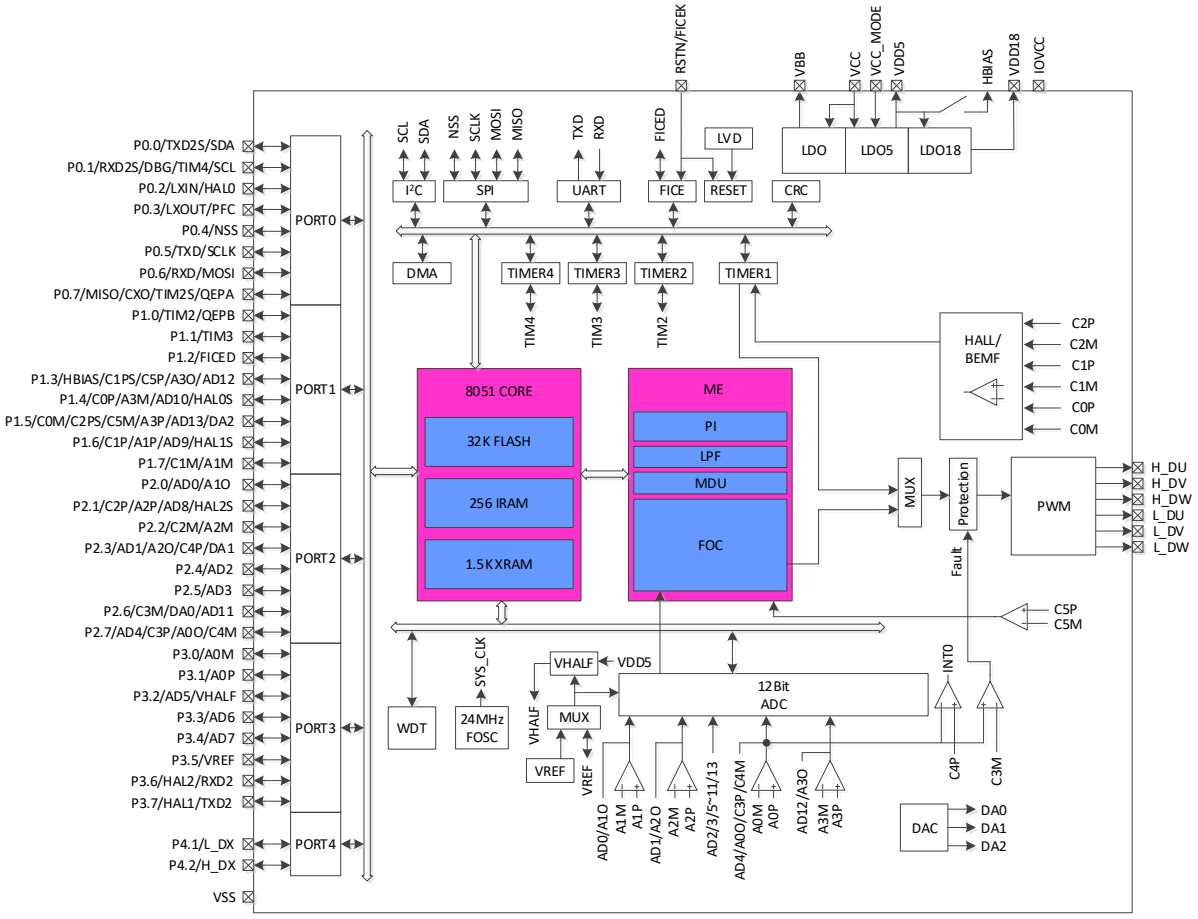


Figure 1-1 Functional Block Diagram of FU6813L

1.4.2 FU6813N

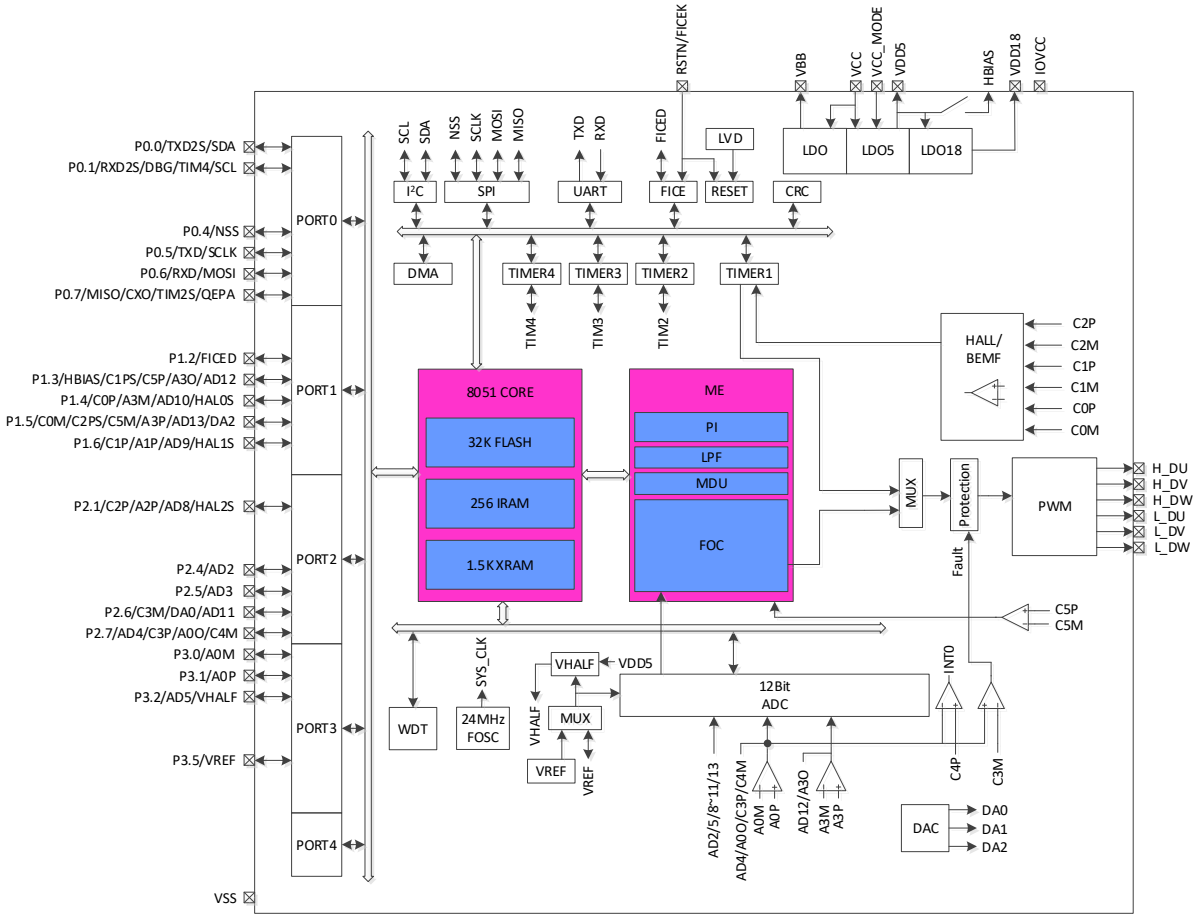


Figure 1-2 Functional Block Diagram of FU6813N

1.4.3 FU6813P

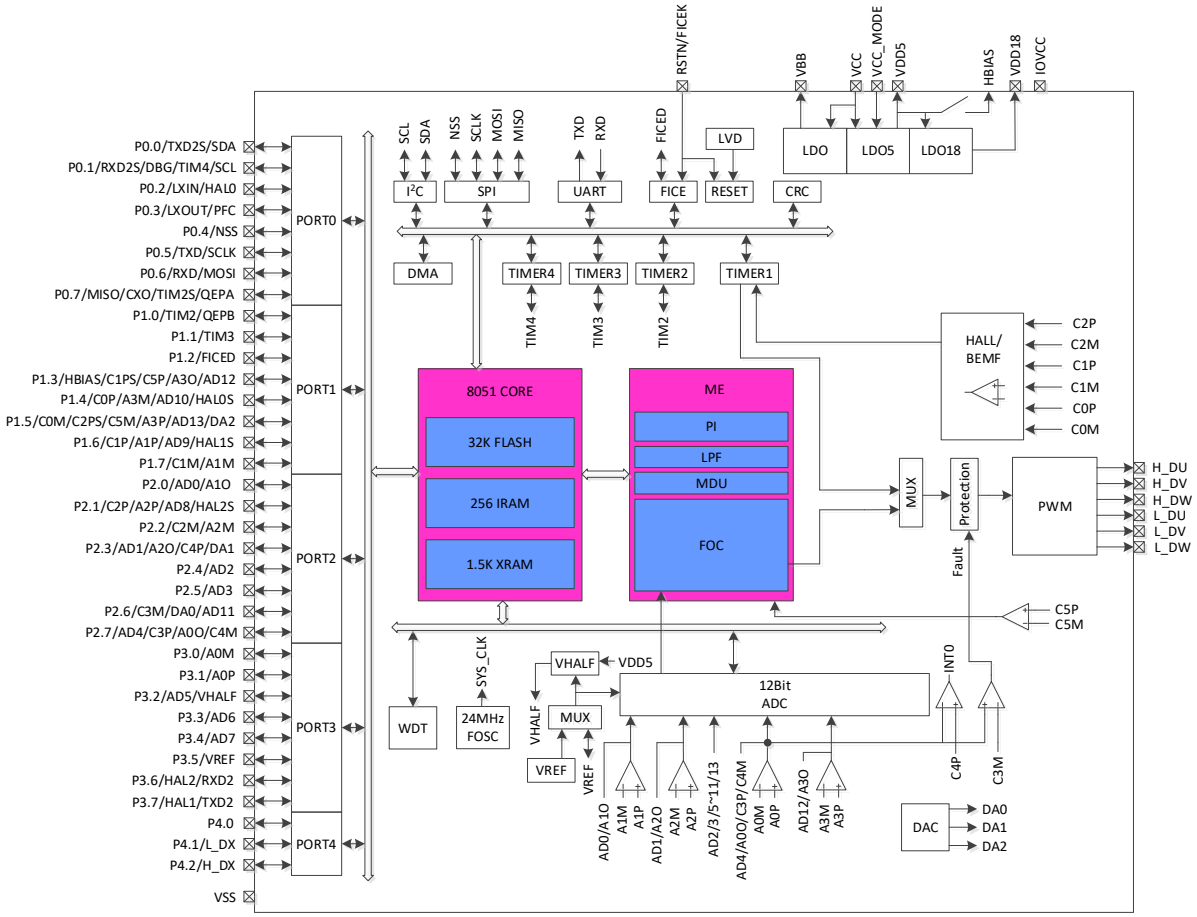


Figure 1-3 Functional Block Diagram of FU6813P

1.4.4 FU6863Q

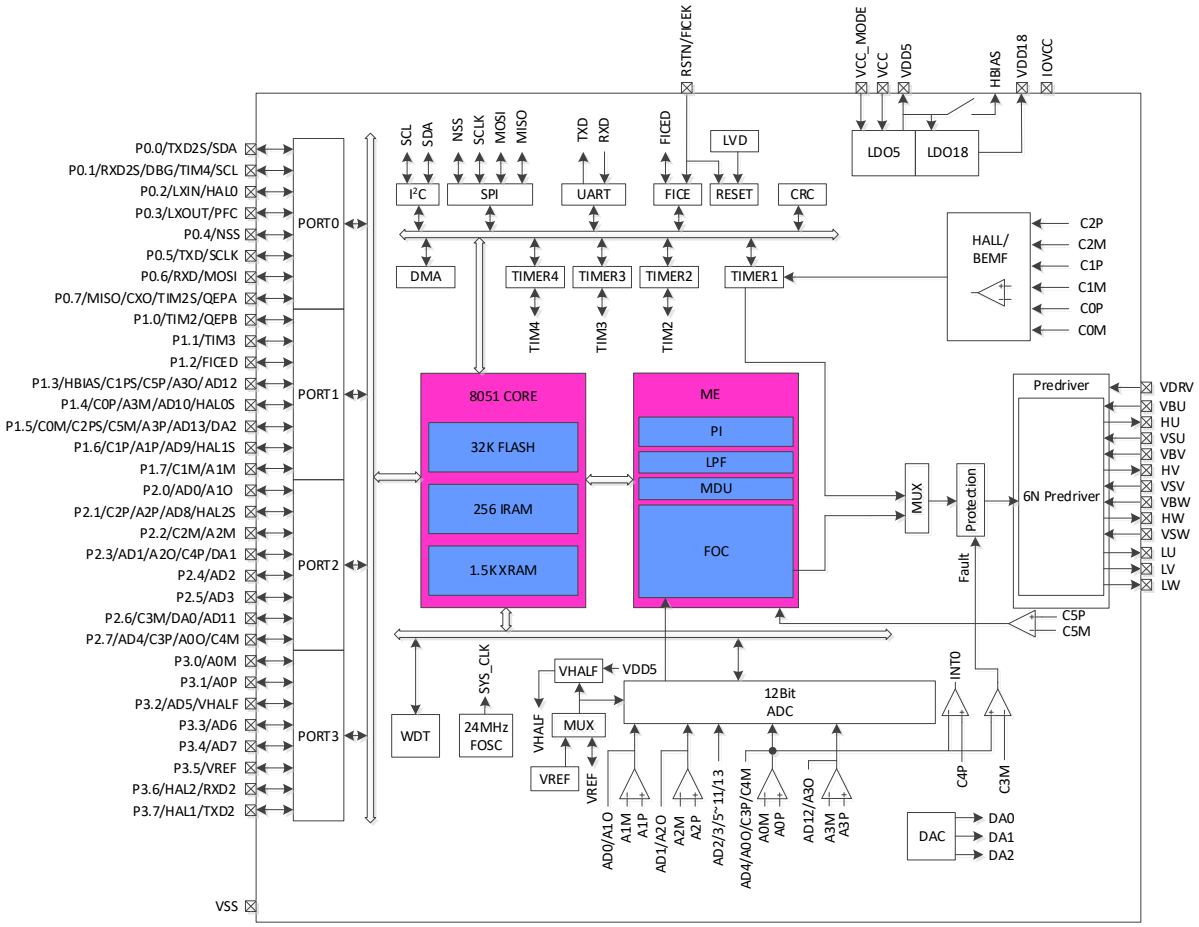


Figure 1-4 Functional Block Diagram of FU6863Q

1.5 Memory Organization

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

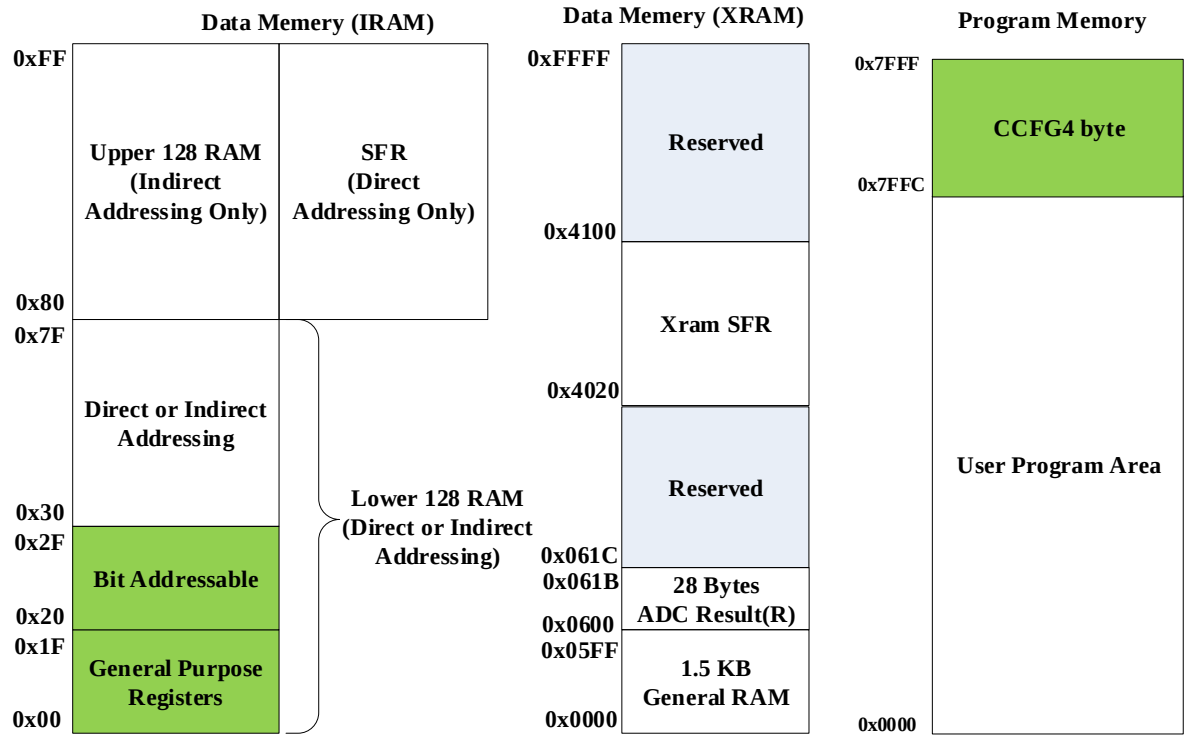


Figure 1-5 Memory Organization

1.5.1 Program Memory

The chip implements this program memory as Flash memory with a block from addresses 0x0000 to 0x7FFF to store control programs. CPU starts from 0x0000 after reset.

1.5.2 Data Memory

The data memory is divided into External Data Memory and Internal Data Memory&SFRs.

The External Data Memory is addressed from 0x0000 to 0x02FF, which can be accessed with MOVX instructions only.

The Internal Data Memory is shown in Figure 1-5. Locations 0x00~0x1F are addressable as 4 banks of general-purpose registers, each bank consisting of 8 registers. Locations 0x20~0x2F are 16-bit addressable, and locations 0x30~0x7F support direct and indirect addressing. 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 SFRs. Stack space is located in the Internal Data Memory.

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	PI_CR			P0_OE	P1_OE	P2_OE	P3_OE
0xF0	B		PI_KIL	PI_KIH	PI_UKMAXL	PI_UKMAXH	PI_UKMINL	PI_UKMINH
0xE8	P4	P4_OE	PI_EKL	PI_EKH	PI_UKL	PI_UKH	PI_KPL	PI_KPH
0xE0	ACC		PI_EK1L	PI_EK1H	PI_UKSL	PI_UKSH		
0xD8	IP3	EVT_FILT	CMP_CR2	LVSR	CMP_CR3			
0xD0	PSW	P1_IE	P1_IF	P2_IE	P2_IF	CMP_CR0	CMP_CR1	CMP_SR
0xC8	IP2	RST_SR	MDU_MD	MDU_D				
0xC0	IP1	MDU_CR	MDU_CL	MDU_CH	MDU_BL	MDU_BH	MDU_AL	MDU_AH
0xB8	IP0							
0xB0	P3							
0xA8	IE	TIM2_CR1	TIM2__CNTRL	TIM2__CNTRH	TIM2__DRL	TIM2__DRH	TIM2__ARRL	TIM2__ARRH
0xA0	P2	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	UT2_DR	UT2_CR					
0x80	P0	SP	DPL	DPH	FLA_KEY	FLA_CR		PCON

Notes:

- Registers with 4 low-order bits as 0 or 8 support addressing access.
- Registers containing the symbol “__” shall be read using variables. If it is read directly, the value will be incorrect.

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)
0x40f8	IAC_KPH	IAC_KPL	IAC_KIH	IAC_KIL	IAC_UKMAXH	IAC_UKMAXL	IAC_UKMINH/ PFC_TRGDLY+ OUTARRH	IAC_UKMINL/ OUTARRL
0x40f0	UDC_UKMAXH	UDC_UKMAXL	UDC_UKMINH/ PFC_KM	UDC_UKMINL/ PFC_CR1	IAC__REFH	IAC__REFL	IAC__UKH	IAC__UKL
0x40e8	UDC_REFH	UDC_REFL	UDC__UKH	UDC__UKL	UDC_KPH	UDC_KPL	UDC_KIH	UDC_KIL
	UDC_EKH	UDC_EKL						
0x40e0	PFC_CR0	PFC_ADCCH	PFC_CSOH	PFC_CSOL	PFC_ARRH/ PFC__UAVGH	PFC_ARRL/ PFC__UAVGL	PFC__DRH	PFC__DRL
0x40d8	FOC__POWH	FOC__POWL	FOC__IAMAXH	FOC__IAMAXL	FOC__IBMAXH	FOC__IBMAXL	FOC__ICMAXH	FOC__ICMAXL
0x40d0	FOC__EALPH	FOC__EALPL	FOC__EBETH	FOC__EBETL	FOC__EOMEH	FOC__EOMEL	FOC__UQEXH	FOC__UQEXL
0x40c8	FOC__IBH	FOC__IBL	FOC__IAH	FOC__IAL	FOC__THETAH	FOC__THETAL	FOC__ETHETAH	FOC__ETHETAL
0x40c0	FOC__IBETH	FOC__IBETL	FOC__VBETH/ FOC_UDCPSH	FOC__VBETL/ FOC_UDCPSL	FOC__VALPH/ FOC_UQCPSH	FOC__VALPL/ FOC_UQCPSL	FOC__ICH	FOC__ICL
0x40b8	FOC__UDH	FOC__UDL	FOC__UQH	FOC__UQL	FOC__IDH	FOC__IDL	FOC__IQH	FOC__IQL
0x40b0	FOC_DMAXH	FOC_DMAXL	FOC_DMINH	FOC_DMINL	FOC_QMAXH	FOC_QMAXL	FOC_QMINH	FOC_QMINL
0x40a8	FOC__RTHESTEPH	FOC__RTHESTEPL	FOC_RTHEACCH	FOC_RTHEACCL	FOC_RTHECNT	FOC_THECOR/ CMP_SAMR	FOC_THECOMPH	FOC_THECOMPL
0x40a0	FOC_CR0	FOC_CR1	FOC_TSMIN	FOC_TGLI	FOC_TBLO	FOC_TRGDLY	FOC_CSOH	FOC_CSOL
0x4098	FOC__UDCFLTH/ TIM1__ITRIPH	FOC__UDCFLTL/ TIM1__ITRIPL	PFC__UACH	PFC__UACL	PFC__IACH	PFC__IACL	PFC_CR2	
0x4090	FOC_IDREFH/ TIM1__URESH	FOC_IDREFL/ TIM1__URESL	FOC_IQREFH/ TIM1__UIGNH	FOC_IQREFL/ TIM1__UIGNL	FOC_DQKPH/ TIM1__KFH	FOC_DQKPL/ TIM1__KFL	FOC_DQKIH/ TIM1__KRH	FOC_DQKIL/ TIM1__KRL
0x4088	FOC_EK3H/ TIM1__RARRH	FOC_EK3L/ TIM1__RARRL	FOC_EK4H/ TIM1__RCNTRH	FOC_EK4L/ TIM1__RCNTRL	FOC_EK1H/ TIM1__UCOPH	FOC_EK1L/ TIM1__UCOPL	FOC_EK2H/ TIM1__UFLPH	FOC_EK2L/ TIM1__UFLPL
0x4080	FOC_FBASEH/ TIM1_DBR7H	FOC_FBASEL/ TIM1_DBR7L	FOC_EFREQACCH/ TIM1__BCNTRH	FOC_EFREQACCL/ TIM1__BCNTRL	FOC_EFREQMINH/ TIM1__BCCRH	FOC_EFRQMINL/ TIM1__BCCRL	FOC_EFREQHOLDH/ TIM1__BARRH	FOC_EFREQHOLDL/ TIM1__BARRL
0x4078	FOC_KSLIDEH/ TIM1_DBR3H	FOC_KSLIDEL/ TIM1_DBR3L	FOC_EKLPFMINH/ TIM1_DBR4H	FOC_EKLPFMINL/ TIM1_DBR4L	FOC_EBMFKH/ TIM1_DBR5H	FOC_EBMFKL/ TIM1_DBR5L	FOC_OMEKLPFH/ TIM1_DBR6H	FOC_OMEKLPFL/ TIM1_DBR6L
0x4070	TIM1_BCORH	TIM1_BCORL	TIM1_CR5		FOC_EKPH/ TIM1_DBR1H	FOC_EKPL/ TIM1_DBR1L	FOC_EKIH/ TIM1_DBR2H	FOC_EKIL/ TIM1_DBR2L
0x4068	TIM1_CR0	TIM1_CR1	TIM1_CR2	TIM1_CR3	TIM1_CR4	TIM1_ IER	TIM1_SR	
0x4060	DRV_DTR	DRV_SR	DRV_CR		SYST_ARRH	SYST_ARRL		
0x4058	DRV_DRH	DRV_DRL	DRV_COMRH	DRV_COMRL	DRV_CMRH	DRV_CMRL	DRV_ARRH	DRV_ARRL

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0x4050	P1_AN	P2_AN	P3_AN	P0_PU	P1_PU	P2_PU	P3_PU	P4_PU
0x4048				DAC_DR	PH_SEL		AMP_CR	VREF_VHALF_CR
0x4040	DMA1_CR1M	DMA1_CR1L	UT2_BAUDL	UT2_BAUDH	CAL_CR0	CAL_CR1		
0x4038	ADC_SCYC	ADC_CR	DMA0_CR0	DMA1_CR0	DMA0_CR1H	DMA0_CR1M	DMA0_CR1L	DMA1_CR1H
0x4030	SPI_CR0	SPI_CR1	SPI_CLK	SPI_DR		DAC_CR	ADC_MASK_SYSCH	ADC_MASK_SYSCL
0x4028	I2C_CR	I2C_ID	I2C_DR	I2C_SR	RTC0TMH	RTC0TML	RTC0STA	
0x4020		CRC_DIN	CRC_CR	CRC_DR	CRC_BEG	CRC_CNT	WDT_CR	WDT_REL
0x4018								
0x0618	AD12_DRH	AD12_DRL	AD13_DRH	AD13_DRL			--	--
0x0610	AD8_DRH	AD8_DRL	AD9_DRH	AD9_DRL	AD10_DRH	AD10_DRL	AD11_DRH	AD11_DRL
0x0608	AD4_DRH	AD4_DRL	AD5_DRH	AD5_DRL	AD6_DRH	AD6_DRL	AD7_DRH	AD7_DRL
0x0600	AD0_DRH	AD0_DRL	AD1_DRH	AD1_DRL	AD2_DRH	AD2_DRL	AD3_DRH	AD3_DRL

Notes:

- Registers containing the symbol "__" shall be read using variables. If it is read directly, the value will be incorrect.
- The SFR is mapped partly to SFR sector of the Internal Data Memory, and partly to External Data Memory (also known as XSFR).

2 Pin Definitions

The IO types are defined as follows:

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

2.1 FU6813L LQFP48 Pins

Table 2-1 FU6813L LQFP48 Pin Descriptions

Pin	FU6813L LQFP48	IO Type	Description
P2.2/ C2M/ A2M	1	DB/ AI/ AI	GPIO, configurable as INT1 input CMP2 negative input AMP2 negative input
P2.3/ AD1/ A2O/ C4P/ DA1	2	DB/ AI/ AO/ AI/ DO	GPIO, configurable as INT1 input Input of ADC channel 1 AMP2 output CMP4 positive input DAC1 output, without Buffer output
P2.4/ AD2	3	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 2 or bus voltage signal
P2.5/ AD3	4	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 3
P2.6/ C3M/ DA0/ AD11	5	DB/ AI/ AO/ AI	GPIO, configurable as INT1 input CMP3 negative input DAC0 output, without Buffer output Input of ADC channel 11
P2.7/ AD4/ C3P/ A0O/ C4M	6	DB/ AI/ AI/ AO/ AI	GPIO, configurable as INT1 input Input of ADC channel 4 for bus current sampling CMP3 positive input AMP0 output CMP4 negative input
P3.0/ A0M	7	DB/ AI	GPIO AMP0 negative input
P3.1/ A0P	8	DB/ AI	GPIO AMP0 positive input
P3.2/ AD5/ VHALF	9	DB/ AI/ AO	GPIO Input of ADC channel 5 1/2 VREF voltage output, with a 1μF external capacitor
P3.3/ AD6	10	DB/ AI	GPIO Input of ADC channel 6
P3.4/ AD7	11	DB/ AI	GPIO Input of ADC channel 7
P3.5/ VREF	12	DB/ AI	GPIO ADC external VREF input or internal VREF output, with a 1μF~4.7μF external capacitor
VSS	13	P	Ground

Pin	FU6813L LQFP48	IO Type	Description
IOVCC	14	P	GPIO power supply, ranging from 3V to 5.5V, with a 1μF ~ 10μF capacitor connected to the ground. IOVCC ≤ VDD5. IOVCC supplies P3.7~6, P0.x, P1.1~0, P4.2~1, H_DU, H_DV, H_DW, L_DU, L_DV and L_DW only, and VDD5 supplies other GPIOs.
P3.6/ HAL2/ RXD2	15	DB/ DI/ DI	GPIO Hall2 logic level input UART2 RXD input
P3.7/ HAL1/ TXD2	16	DB/ DI/ DO	GPIO Hall1 logic level input UART2 TXD output
P0.0/ TXD2S/ SDA	17	DO/ DB/ DO	GPIO, configurable as INT0 input UART2 TXD output after function switching I^2C SDA, configured as collector open-drain output with a 4.7k pull-up resistor
P0.1/ RXD2S/ DBG/ TIM4/ SCL	18	DB/ DI/ DO/ DB/ DB	GPIO UART2 RXD input after function switching Debug port Timer4 input capture mode I^2C SCL, configured as collector open-drain output with a 4.7k pull-up resistor
P0.2/ LXIN/ HAL0	19	DB/ AI/ DI	GPIO 32768Hz crystal clock input Hall0 logic level input
P0.3/ LXOUT/ PFC	20	DB/ AO/ DO	GPIO 32768Hz crystal clock output PFC output
P0.4/ NSS	21	DB/ DB	GPIO SPI NSS
P0.5/ TXD/ SCLK	22	DB/ DO/ DB	GPIO UART1 TXD output SPI SCLK
P0.6/ RXD/ MOSI	23	DB/ DI/ DB	GPIO UART1 RXD input SPI MOSI, master output or slave input
P0.7/ MISO/ CXO/ TIM2S/ QEPA	24	DB/ DB/ DO/ DB/ DI	GPIO SPI MISO, master input or slave output Test pin for comparator output Timer2 input capture mode or PWM output after function switching QEP encode A input
P1.0/ TIM2/ QEPB	25	DB/ DB/ DI	GPIO, configurable as INT1 input Timer2 input capture mode or PWM output before function switching QEP encode B input
P1.1/ TIM3	26	DB/ DB	GPIO, configurable as INT1 input Timer3 input capture mode
P4.1/ L_DX	27	DB/ DO	GPIO Low-side X-phase PWM output
P4.2/ H_DX	28	DB/ DO	GPIO High-side X-phase PWM output
L_DU	29	DO	Low-side U-phase PWM output
L_DV	30	DO	Low-side V-phase PWM output
L_DW	31	DO	Low-side W-phase PWM output
H_DU	32	DO	High-side U-phase PWM output
H_DV	33	DO	High-side V-phase PWM output
H_DW	34	DO	High-side W-phase PWM output

Pin	FU6813L LQFP48	IO Type	Description
VCC	35	P	Power input. The voltage range is determined by VCC_MODE, with an external filter capacitor of 10μF or above. ■ High-voltage single-power supply mode: When VCC_MODE = 0, external power supply 5V~24V is connected to VCC pin, and internal LDO supplies VDD5 voltage. ■ Low-voltage single-power supply mode: When VCC_MODE = 1, external power supply 3V~5.5V is connected to VDD5 pin, and VDD5 pin is shorted to VCC pin. ■ Dual-power supply mode: When VCC_MODE = 1, external power supply 1 (5V~36V) is connected to VCC pin, and external power supply 2 (5V) is connected to VDD5 pin.
VSS	36	P	Ground
VDD5	37	P	Mid-voltage power input or 5V LDO power output is determined by VCC_MODE. See descriptions on VCC pin for power connection. It is connected with a 1μF~4.7μF external capacitor. ■ If VCC_MODE = 0, internal LDO outputs 5V power supply. ■ If VCC_MODE = 1, 3V~5.5V external power is supplied.
VCC_MODE	38	DI	Power supply mode control. See descriptions on VCC pin for details.
RSTN/ FICEK	39	DI/ DI	Input of external reset; Built-in pull-up resistor FICE SCL
VDD18	40	P	1.8V LDO output with an external 1μF ~ 4.7μF capacitor
P1.2/ FICED	41	DB/ DB	GPIO, configurable as INT1 input FICE SDA
P1.3/ HBIAS/ C1PS/ C5P/ A3O/ AD12	42	DB/ DO/ AI/ AI/ DO/ AI	GPIO Hall bias power supply, internally connected to VDD5 via a switch CMP1 positive input after function switching CMP5 positive input AMP3 output Input of ADC channel 12
P1.4/ C0P/ A3M/ AD10/ HAL0S	43	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP0 positive input AMP3 negative input Input of ADC channel 10 Hall0 logic level input after function switching
P1.5/ C0M/ C2PS/ C5M/ A3P/ AD13/ DA2	44	DB/ AI/ AI/ AI/ AI/ AI/ AO	GPIO, configurable as INT1 input CMP0 negative input CMP2 positive input after function switching CMP5 negative input AMP3 positive input Input of ADC channel 13 DAC2 output, without buffer output
P1.6/ C1P/ A1P/ AD9/ HAL1S	45	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP1 positive input AMP1 positive input Input of ADC channel 9 Hall1 logic level input after function switching
P1.7/ C1M/ A1M	46	DB/ AI/ AI	GPIO, configurable as INT1 input CMP1 negative input AMP1 negative input
P2.0/ AD0/ A1O	47	DB/ AI/ AO	GPIO, configurable as INT1 input Input of ADC channel 0 AMP1 output

Pin	FU6813L LQFP48	IO Type	Description
P2.1/ C2P/ A2P/ AD8/ HAL2S	48	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP2 positive input AMP2 positive input Input of ADC channel 8 Hall2 logic level input after function switching

2.2 FU6813L LQFP48 Pinout Diagram

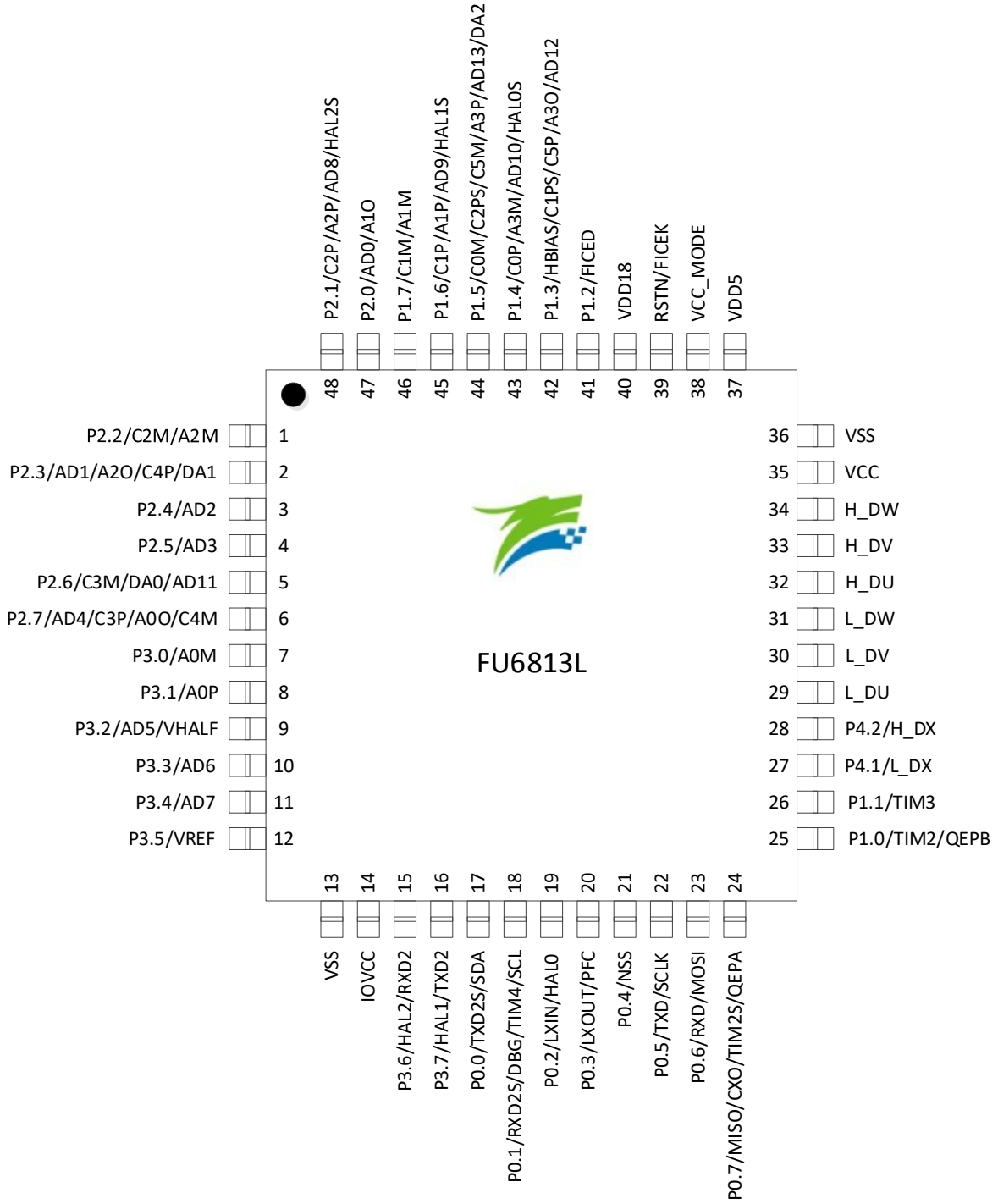


Figure 2-1 FU6813L LQFP48 Pinout Diagram

2.3 FU6813N QFN32 Pins

Table 2-2 FU6813N QFN32 Pin Descriptions

Pin	FU6813N QFN32	IO Type	Description
P2.1/ C2P/ A2P/ AD8/ HAL2S	1	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP2 positive input AMP2 positive input Input of ADC channel 8 Hall2 logic level input after function switching
P2.4/ AD2	2	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 2 or bus voltage signal
P2.6/ C3M/ DA0/ AD11	3	DB/ AI/ AO/ AI	GPIO, configurable as INT1 input CMP3 negative input DAC0 output, without Buffer output Input of ADC channel 11
P2.7/ AD4/ C3P/ A0O/ C4M	4	DB/ AI/ AI/ AO/ AI	GPIO, configurable as INT1 input Input of ADC channel 4 for bus current sampling CMP3 positive input AMP0 output CMP4 negative input
P3.0/ A0M	5	DB/ AI	GPIO AMP0 negative input
P3.1/ A0P	6	DB/ AI	GPIO AMP0 positive input
P3.2/ AD5/ VHALF	7	DB/ AI/ AO	GPIO Input of ADC channel 5 1/2 VREF voltage output, with a 1μF external capacitor
P3.5/ VREF	8	DB/ AI	GPIO ADC external VREF input or internal VREF output, with a 1μF~4.7μF external capacitor
IOVCC	9	P	GPIO power supply, ranging from 3V to 5.5V, with a 1μF ~ 10μF capacitor connected to the ground. IOVCC ≤ VDD5. IOVCC supplies P0.x, P1.1, H_DU, H_DV, H_DW, L_DU, L_DV and L_DW only, and VDD5 supplies other GPIOs.
P0.0/ TXD2S/ SDA	10	DO/ DB/ DO	GPIO, configurable as INT0 input UART2 TXD output after function switching I^2C SDA, configured as collector open-drain output with a 4.7k pull-up resistor
P0.1/ RXD2S/ DBG/ TIM4/ SCL	11	DB/ DI/ DO/ DB/ DB	GPIO UART2 RXD input after function switching Debug port Timer4 input capture mode I^2C SCL, configured as collector open-drain output with a 4.7k pull-up resistor
P0.4/ NSS	12	DB/ DB	GPIO SPI NSS
P0.5/ TXD/ SCLK	13	DB/ DO/ DB	GPIO UART1 TXD output SPI SCLK
P0.6/ RXD/ MOSI	14	DB/ DI/ DB	GPIO UART1 RXD input SPI MOSI, master output or slave input
P0.7/ MISO/ CXO/ TIM2S/ QEPA	15	DB/ DB/ DO/ DB/ DI	GPIO SPI MISO, master input or slave output Test pin for comparator output Timer2 input capture mode or PWM output after function switching QEP encode A input

Pin	FU6813N QFN32	IO Type	Description
P1.1/ TIM3	16	DB/ DB	GPIO, configurable as INT1 input Timer3 input capture mode
L_DU	17	DO	Low-side U-phase PWM output
L_DV	18	DO	Low-side V-phase PWM output
L_DW	19	DO	Low-side W-phase PWM output
H_DU	20	DO	High-side U-phase PWM output
H_DV	21	DO	High-side V-phase PWM output
H_DW	22	DO	High-side W-phase PWM output
VCC	23	P	Power input. The voltage range is determined by VCC_MODE, with an external filter capacitor of 10μF or above. ■ High-voltage single-power supply mode: External power supply 5V~24V is connected to VCC pin, and internal LDO supplies VDD5 voltage. ■ Low-voltage single-power supply mode: External power supply 3V~5.5V is connected to VDD5 pin, and VDD5 pin is shorted to VCC pin.
VSS	24	P	Ground
VDD5	25	P	Mid-voltage power input or 5V LDO power output is determined by VCC_MODE. See descriptions on VCC pin for power connection. It is connected with a 1μF~4.7μF external capacitor. ■ When VCC_MODE = 0, internal LDO outputs 5V power supply. ■ When VCC_MODE = 1, 3V~5.5V external power is supplied.
RSTN/ FICEK	26	DI/ DI	Input of external reset; Built-in pull-up resistor FICE SCL
VDD18	27	P	1.8V LDO output with an external 1μF ~ 4.7μF capacitor
P1.2/ FICED	28	DB/ DB	GPIO, configurable as INT1 input FICE SDA
P1.3/ HBIAS/ C1PS/ C5P/ A3O/ AD12	29	DB/ DO/ AI/ AI/ DO/ AI	GPIO Hall bias power supply, internally connected to VDD5 via a switch CMP1 positive input after function switching CMP5 positive input AMP3 output Input of ADC channel 12
P1.4/ C0P/ A3M/ AD10/ HAL0S	30	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP0 positive input AMP3 negative input Input of ADC channel 10 Hall0 logic level input after function switching
P1.5/ C0M/ C2PS/ C5M/ A3P/ AD13/ DA2	31	DB/ AI/ AI/ AI/ AI/ AI/ AO	GPIO, configurable as INT1 input CMP0 negative input CMP2 positive input after function switching CMP5 negative input AMP3 positive input Input of ADC channel 13 DAC2 output, without buffer output
P1.6/ C1P/ A1P/ AD9/ HAL1S	32	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP1 positive input AMP1 positive input Input of ADC channel 9 Hall1 logic level input after function switching

2.4 FU6813N QFN32 Pinout Diagram

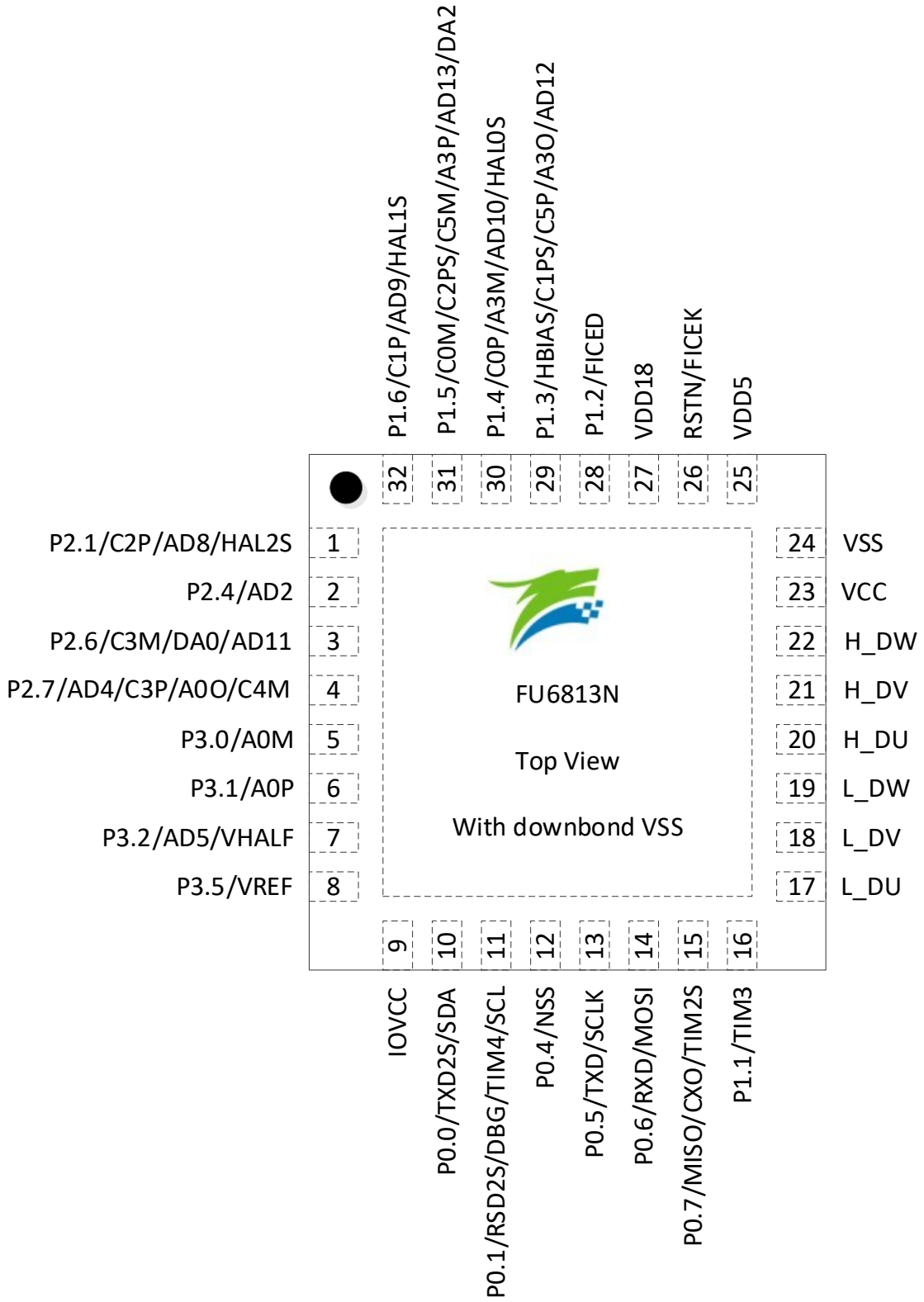


Figure 2-2 FU6813N QFN32 Pinout Diagram

2.5 FU6813P LQFP52 Pins

Table 2-3 FU6813P LQFP52 Pin Descriptions

Pin	FU6813P LQFP52	IO Type	Description
P2.3/ AD1/ A2O/ C4P/ DA1	1	DB/ AI/ AO/ AI/ DO	GPIO, configurable as INT1 input Input of ADC channel 1 AMP2 output CMP4 positive input DAC1 output, without Buffer output
P2.4/ AD2	2	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 2 or bus voltage signal
P2.5/ AD3	3	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 3
P2.6/ C3M/ DA0/ AD11	4	DB/ AI/ AO/ AI	GPIO, configurable as INT1 input CMP3 negative input DAC0 output, without Buffer output Input of ADC channel 11
P2.7/ AD4/ C3P/ A0O/ C4M	5	DB/ AI/ AI/ AO/ AI	GPIO, configurable as INT1 input Input of ADC channel 4 for bus current sampling CMP3 positive input AMP0 output CMP4 negative input
P3.0/ A0M	6	DB/ AI	GPIO AMP0 negative input
P3.1/ A0P	7	DB/ AI	GPIO AMP0 positive input
P3.2/ AD5/ VHALF	8	DB/ AI/ AO	GPIO Input of ADC channel 5 1/2 VREF voltage output, with a 1μF external capacitor
P3.3/ AD6	9	DB/ AI	GPIO Input of ADC channel 6
P3.4/ AD7	10	DB/ AI	GPIO Input of ADC channel 7
P3.5/ VREF	11	DB/ AI	GPIO ADC external VREF input or internal VREF output, with a 1μF~4.7μF external capacitor
NC	12		Not connected
VSS	13	P	Ground
IOVCC	14	P	GPIO power supply, ranging from 3V to 5.5V, with a 1μF ~ 10μF capacitor connected to the ground. IOVCC ≤ VDD5. IOVCC supplies P3.7~6, P0.x, P1.1~0, P4.2~1, H_DU, H_DV, H_DW, L_DU, L_DV and L_DW only, and VDD5 supplies other GPIOs.
P3.6/ HAL2/ RXD2	15	DB/ DI/ DI	GPIO Hall2 logic level input UART2 RXD input
P3.7/ HAL1/ TXD2	16	DB/ DI/ DO	GPIO Hall1 logic level input UART2 TXD output
P0.0/ TXD2S/ SDA	17	DO/ DB/ DO	GPIO, configurable as INT0 input UART2 TXD output after function switching I^2C SDA, configured as collector open-drain output with a 4.7k pull-up resistor

Pin	FU6813P LQFP52	IO Type	Description
P0.1/ RXD2S/ DBG/ TIM4/ SCL	18	DB/ DI/ DO/ DB/ DB	GPIO UART2 RXD input after function switching Debug port Timer4 input capture mode I^2C SCL, configured as collector open-drain output with a 4.7k pull-up resistor
P0.2/ LXIN/ HAL0	19	DB/ AI/ DI	GPIO 32768Hz crystal clock input Hall0 logic level input
P0.3/ LXOUT/ PFC	20	DB/ AO/ DO	GPIO 32768Hz crystal clock output PFC output
P0.4/ NSS	21	DB/ DB	GPIO SPI NSS
P0.5/ TXD/ SCLK	22	DB/ DO/ DB	GPIO UART1 TXD output SPI SCLK
P0.6/ RXD/ MOSI	23	DB/ DI/ DB	GPIO UART1 RXD input SPI MOSI, master output or slave input
P0.7/ MISO/ CXO/ TIM2S/ QEPA	24	DB/ DB/ DO/ DB/ DI	GPIO SPI MISO, master input or slave output Test pin for comparator output Timer2 input capture mode or PWM output after function switching QEP encode A input
P1.0/ TIM2/ QEPB	25	DB/ DB/ DI	GPIO, configurable as INT1 input Timer2 input capture mode or PWM output before function switching QEP encode B input
P1.1/ TIM3	26	DB/ DB	GPIO, configurable as INT1 input Timer3 input capture mode
P4.0	27	DB	GPIO
P4.1/ L_DX	28	DB/ DO	GPIO Low-side X-phase PWM output
P4.2/ H_DX	29	DB/ DO	GPIO High-side X-phase PWM output
VBB	30	P	Internal LDO outputs 15V power supply, ■ When VCC ≤ 15V, VBB = VCC; ■ When VCC > 15V, VBB = 15V. It is recommended that the 10μF and 0.1μF external capacitors be connected in parallel to GND for powering the internal pre-driver.
L_DU	31	DO	Low-side U-phase PWM output
L_DV	32	DO	Low-side V-phase PWM output
L_DW	33	DO	Low-side W-phase PWM output
H_DU	34	DO	High-side U-phase PWM output
H_DV	35	DO	High-side V-phase PWM output
H_DW	36	DO	High-side W-phase PWM output
NC	37		Not connected

Pin	FU6813P LQFP52	IO Type	Description
VCC	38	P	Power input. The voltage range is determined by VCC_MODE, with an external filter capacitor of 10μF or above. ■ High-voltage single-power supply mode: When VCC_MODE = 0, external power supply 5V~24V is connected to VCC pin, and internal LDO supplies VDD5 voltage. ■ Low-voltage single-power supply mode: When VCC_MODE = 1, external power supply 3V~5.5V is connected to VDD5 pin, and VDD5 pin is shorted to VCC pin. ■ Dual-power supply mode: When VCC_MODE = 1, external power supply 1 (5V~36V) is connected to VCC pin, and external power supply 2 (5V) is connected to VDD5 pin.
VSS	39	P	Ground
VDD5	40	P	Mid-voltage power input or 5V LDO power output is determined by VCC_MODE. See descriptions on VCC pin for power connection. It is connected with a 1μF~4.7μF external capacitor. ■ If VCC_MODE = 0, internal LDO outputs 5V power supply. ■ If VCC_MODE = 1, 3V~5.5V external power is supplied.
VCC_MODE	41	DI	Power supply mode control. See descriptions on VCC pin for details.
RSTN/	42	DI/	Input of external reset; Built-in pull-up resistor
FICEK		DI	FICE SCL
VDD18	43	P	1.8V LDO output with an external 1μF ~ 4.7μF capacitor
P1.2/	44	DB/	GPIO, configurable as INT1 input
FICED		DB	FICE SDA
P1.3/ HBIAS/ C1PS/ C5P/ A3O/ AD12	45	DB/ DO/ AI/ AI/ DO/ AI	GPIO Hall bias power supply, internally connected to VDD5 via a switch CMP1 positive input after function switching CMP5 positive input AMP3 output Input of ADC channel 12
P1.4/ C0P/ A3M/ AD10/ HAL0S	46	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP0 positive input AMP3 negative input Input of ADC channel 10 Hall0 logic level input after function switching
P1.5/ C0M/ C2PS/ C5M/ A3P/ AD13/ DA2	47	DB/ AI/ AI/ AI/ AI/ AI/ AO	GPIO, configurable as INT1 input CMP0 negative input CMP2 positive input after function switching CMP5 negative input AMP3 positive input Input of ADC channel 13 DAC2 output, without buffer output
P1.6/ C1P/ A1P/ AD9/ HAL1S	48	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP1 positive input AMP1 positive input Input of ADC channel 9 Hall1 logic level input after function switching
P1.7/ C1M/ A1M	49	DB/ AI/ AI	GPIO, configurable as INT1 input CMP1 negative input AMP1 negative input
P2.0/ AD0/ A1O	50	DB/ AI/ AO	GPIO, configurable as INT1 input Input of ADC channel 0 AMP1 output

Pin	FU6813P LQFP52	IO Type	Description
P2.1/ C2P/ A2P/ AD8/ HAL2S	51	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP2 positive input AMP2 positive input Input of ADC channel 8 Hall2 logic level input after function switching
P2.2/ C2M/ A2M	52	DB/ AI/ AI	GPIO, configurable as INT1 input CMP2 negative input AMP2 negative input

2.6 FU6813P LQFP52 Pinout Diagram

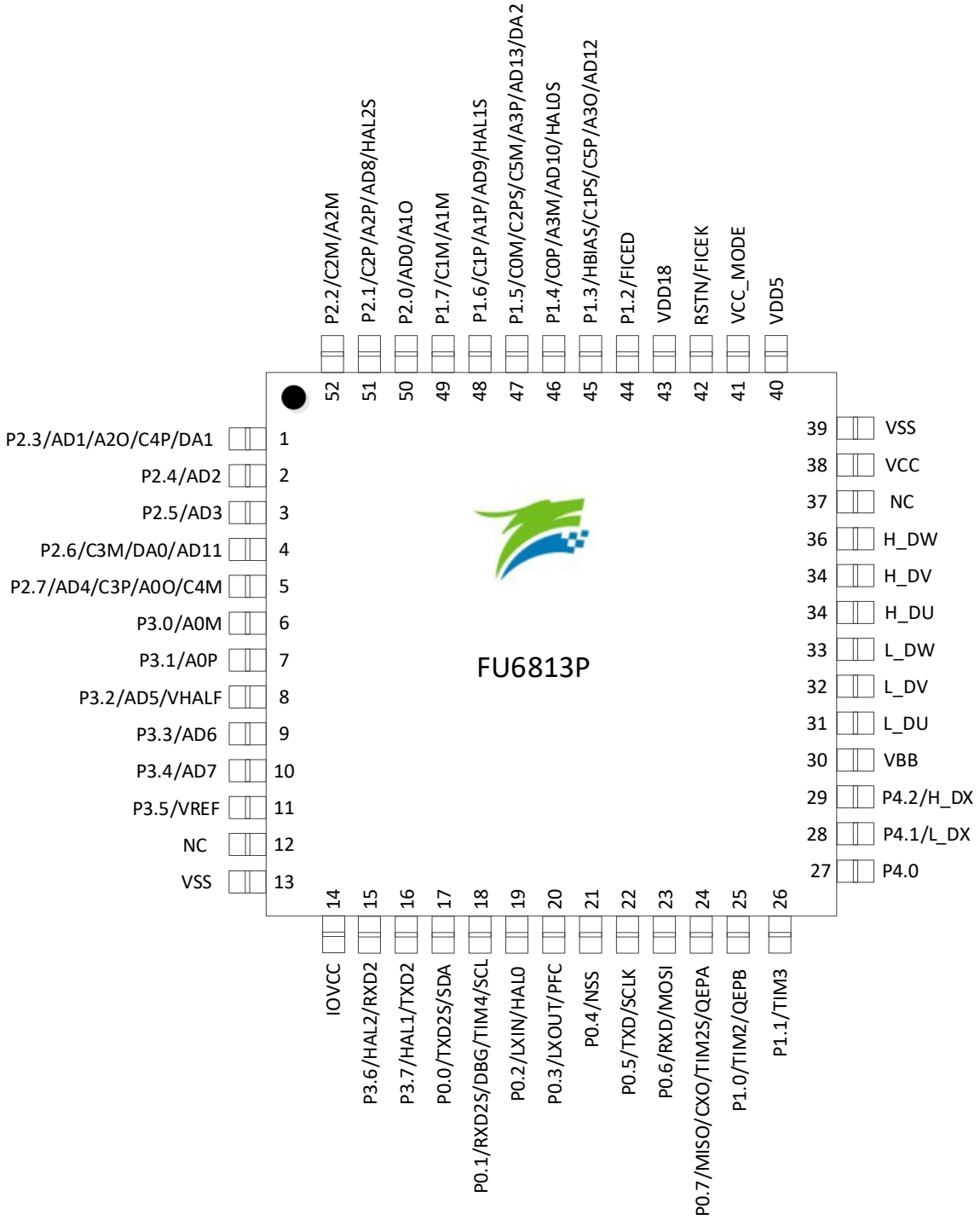


Figure 2-3 FU6813P LQFP52 Pinout Diagram

2.7 FU6863Q QFN56 Pins

Table 2-4 FU6863Q QFN56 Pin Descriptions

Pin	FU6863Q QFN56	IO Type	Description
VSU	1	P	6N pre-driver U-phase input, as GND reference for U-phase high-side bootstrap
HU	2	DO	6N pre-driver high-side U-phase PWM output
VBU	3	P	6N pre-driver high-side U-phase bootstrap power supply
VSV	4	P	6N pre-driver V-phase input, as GND reference for V-phase high-side bootstrap
HV	5	DO	6N pre-driver high-side V-phase PWM output
VBV	6	P	6N pre-driver high-side V-phase bootstrap power supply
VSW	7	P	6N pre-driver W-phase input, as GND reference for W-phase high-side bootstrap
HW	8	DO	6N pre-driver high-side W-phase PWM output
VBW	9	P	6N pre-driver high-side W-phase bootstrap power supply
VCC	10	P	Power input. The voltage range is determined by VCC_MODE, with an external filter capacitor of 10μF or above. ■ High-voltage single-power supply mode: When VCC_MODE = 0, external power supply 5V~24V is connected to VCC pin, and internal LDO supplies VDD5 voltage. ■ Dual-power supply mode: When VCC_MODE = 1, external power supply 1 (5V~36V) is connected to VCC pin, and external power supply 2 (5V) is connected to VDD5 pin.
VSS	11	P	Ground
VDD5	12	P	Mid-voltage power input or 5V LDO power output is determined by VCC_MODE. See descriptions on VCC pin for power connection. It is connected with a 1μF~4.7μF external capacitor. ■ If VCC_MODE = 0, internal LDO outputs 5V power supply. ■ If VCC_MODE = 1, 3V~5.5V external power is supplied.
VCC_MODE	13	DI	Power supply mode control. See descriptions on VCC pin for details.
RSTN/ FICEK	14	DI/ DI	Input of external reset; Built-in pull-up resistor FICE SCL
VDD18	15	P	1.8V LDO output with an external 1μF ~ 4.7μF capacitor
VSS	16	P	Ground
P1.2/ FICED	17	DB/ DB	GPIO, configurable as INT1 input FICE SDA
P1.3/ HBIAS/ C1PS/ C5P/ A3O/ AD12	18	DB/ DO/ AI/ AI/ DO/ AI	GPIO Hall bias power supply, internally connected to VDD5 via a switch CMP1 positive input after function switching CMP5 positive input AMP3 output Input of ADC channel 12
P1.4/ C0P/ A3M/ AD10/ HAL0S	19	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP0 positive input AMP3 negative input Input of ADC channel 10 Hall0 logic level input after function switching

Pin	FU6863Q QFN56	IO Type	Description
P1.5/ C0M/ C2PS/ C5M/ A3P/ AD13/ DA2	20	DB/ AI/ AI/ AI/ AI/ AI/ AO	GPIO, configurable as INT1 input CMP0 negative input CMP2 positive input after function switching CMP5 negative input AMP3 positive input Input of ADC channel 13 DAC2 output, without buffer output
P1.6/ C1P/ A1P/ AD9/ HAL1S	21	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP1 positive input AMP1 positive input Input of ADC channel 9 Hall1 logic level input after function switching
P1.7/ C1M/ A1M	22	DB/ AI/ AI	GPIO, configurable as INT1 input CMP1 negative input AMP1 negative input
P2.0/ AD0/ A1O	23	DB/ AI/ AO	GPIO, configurable as INT1 input Input of ADC channel 0 AMP1 output
P2.1/ C2P/ A2P/ AD8/ HAL2S	24	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP2 positive input AMP2 positive input Input of ADC channel 8 Hall2 logic level input after function switching
P2.2/ C2M/ A2M	25	DB/ AI/ AI	GPIO, configurable as INT1 input CMP2 negative input AMP2 negative input
P2.3/ AD1/ A2O/ C4P/ DA1	26	DB/ AI/ AO/ AI/ DO	GPIO, configurable as INT1 input Input of ADC channel 1 AMP2 output CMP4 positive input DAC1 output, without Buffer output
P2.4/ AD2	27	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 2 or bus voltage signal
P2.5/ AD3	28	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 3
P2.6/ C3M/ DA0/ AD11	29	DB/ AI/ AO/ AI	GPIO, configurable as INT1 input CMP3 negative input DAC0 output, without Buffer output Input of ADC channel 11
P2.7/ AD4/ C3P/ A0O/ C4M	30	DB/ AI/ AI/ AO/ AI	GPIO, configurable as INT1 input Input of ADC channel 4 for bus current sampling CMP3 positive input AMP0 output CMP4 negative input
P3.0/ A0M	31	DB/ AI	GPIO AMP0 negative input
P3.1/ A0P	32	DB/ AI	GPIO AMP0 positive input
P3.2/ AD5/ VHALF	33	DB/ AI/ AO	GPIO Input of ADC channel 5 1/2 VREF voltage output, with a 1μF external capacitor
P3.3/ AD6	34	DB/ AI	GPIO Input of ADC channel 6
P3.4/ AD7	35	DB/ AI	GPIO Input of ADC channel 7
P3.5/ VREF	36	DB/ AI	GPIO ADC external VREF input or internal VREF output, with a 1μF~4.7μF external capacitor

Pin	FU6863Q QFN56	IO Type	Description
VSS	37	P	Ground
IOVCC	38	P	GPIO power supply, ranging from 3V to 5.5V, with a 1μF ~ 10μF capacitor connected to the ground. IOVCC ≤ VDD5.IOVCC supplies P3.7~6, P0.x, P1.1~0, H_DU, H_DV, H_DW, L_DU, L_DV and L_DW= only, and VDD5 supplies other GPIOs.
P3.6/ HAL2/ RXD2	39	DB/ DI/ DI	GPIO Hall2 logic level input UART2 RXD input
P3.7/ HAL1/ TXD2	40	DB/ DI/ DO	GPIO Hall1 logic level input UART2 TXD output
P0.0/ TXD2S/ SDA	41	DO/ DB/ DO	GPIO, configurable as INT0 input UART2 TXD output after function switching I^2C SDA, configured as collector open-drain output with a 4.7k pull-up resistor
P0.1/ RXD2S/ DBG/TIM4/ SCL	42	DB/ DI/ DO/ DB/ DB	GPIO UART2 RXD input after function switching Debug port Timer4 input capture mode I^2C SCL, configured as collector open-drain output with a 4.7k pull-up resistor
P0.2/ LXIN/ HAL0	43	DB/ AI/ DI	GPIO 32768Hz crystal clock input Hall0 logic level input
P0.3/ LXOUT/ PFC	44	DB/ AO/ DO	GPIO 32768Hz crystal clock output PFC output
P0.4/ NSS	45	DB/ DB	GPIO SPI NSS
P0.5/ TXD/ SCLK	46	DB/ DO/ DB	GPIO UART1 TXD output SPI SCLK
P0.6/ RXD/ MOSI	47	DB/ DI/ DB	GPIO UART1 RXD input SPI MOSI, master output or slave input
P0.7/ MISO/ CXO/ TIM2S/ QEPA	48	DB/ DB/ DO/ DB/ DI	GPIO SPI MISO, master input or slave output Test pin for comparator output Timer2 input capture mode or PWM output after function switching QEP encode A input
P1.0/ TIM2/ QEPB	49	DB/ DB/ DI	GPIO, configurable as INT1 input Timer2 input capture mode or PWM output before function switching QEP encode B input
P1.1/ TIM3	50	DB/ DB	GPIO, configurable as INT1 input Timer3 input capture mode
VDRV	51	P	Internal LDO outputs 15V power supply, When VCC ≤ 15V, VDRV = VCC; When VCC > 15V, VDRV = 15V. It is recommended that the 10μF and 0.1μF external capacitors be connected in parallel to GND for powering the internal pre-driver.
VSS	52	P	Ground
NC	53		Not connected
LU	54	DO	6N pre-driver low-side U-phase PWM output
LV	55	DO	6N pre-driver low-side V-phase PWM output
LW	56	DO	6N pre-driver low-side W-phase PWM output

2.8 FU6863Q QFN56 Pinout Diagram

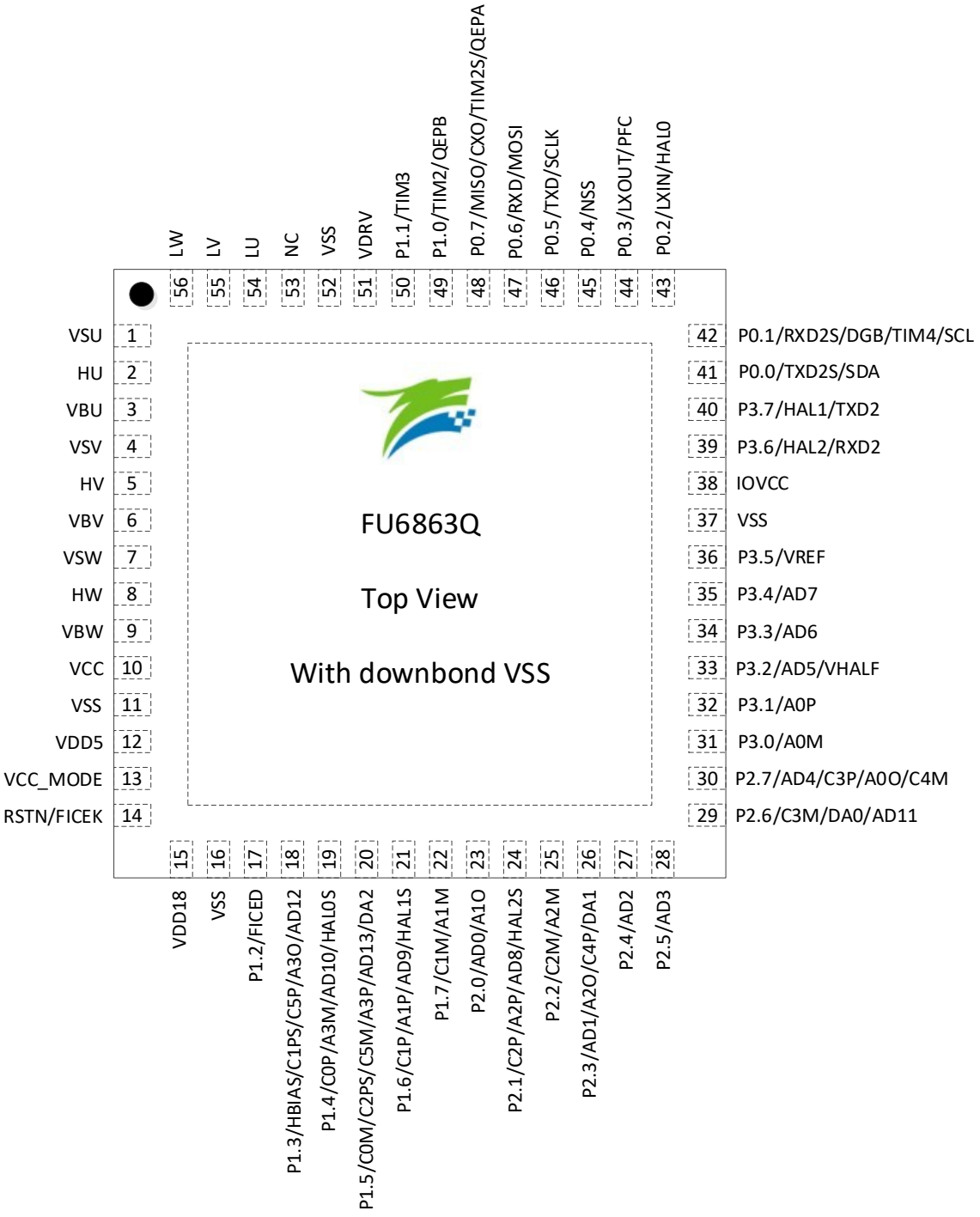
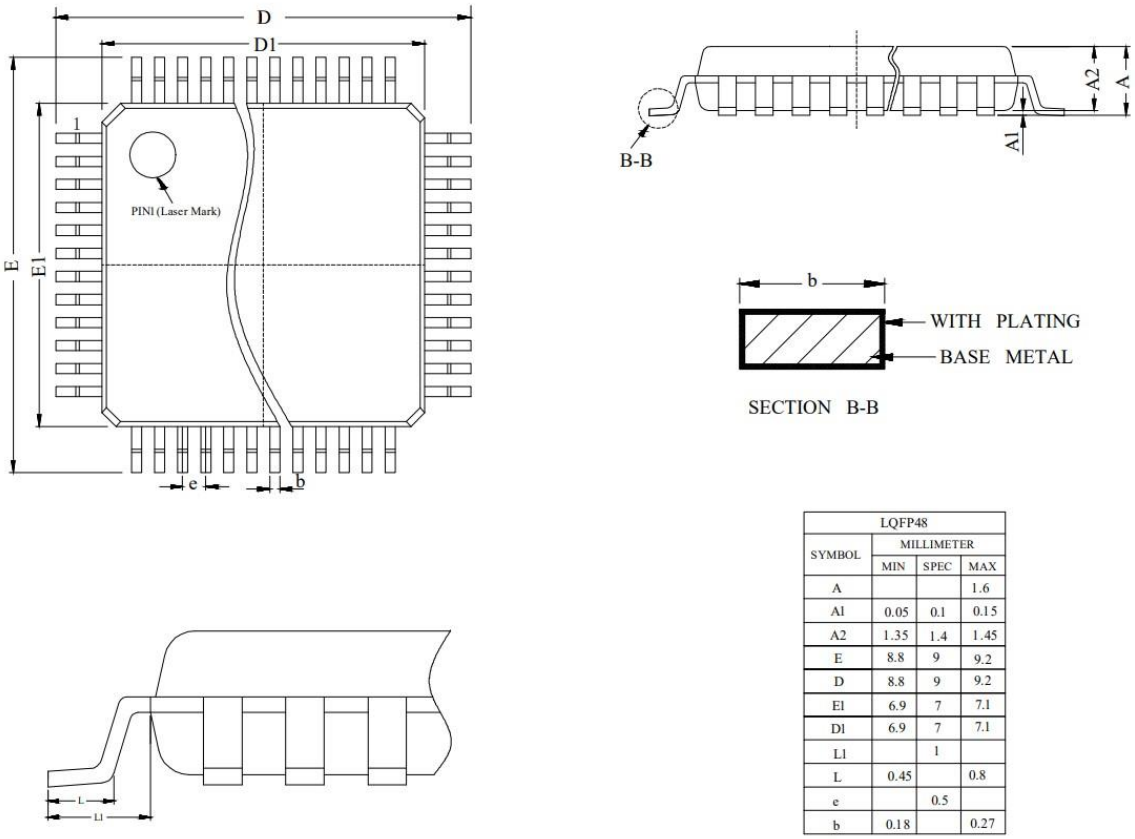


Figure 2-4 FU6863Q QFN56 Pinout Diagram

3 Package Information

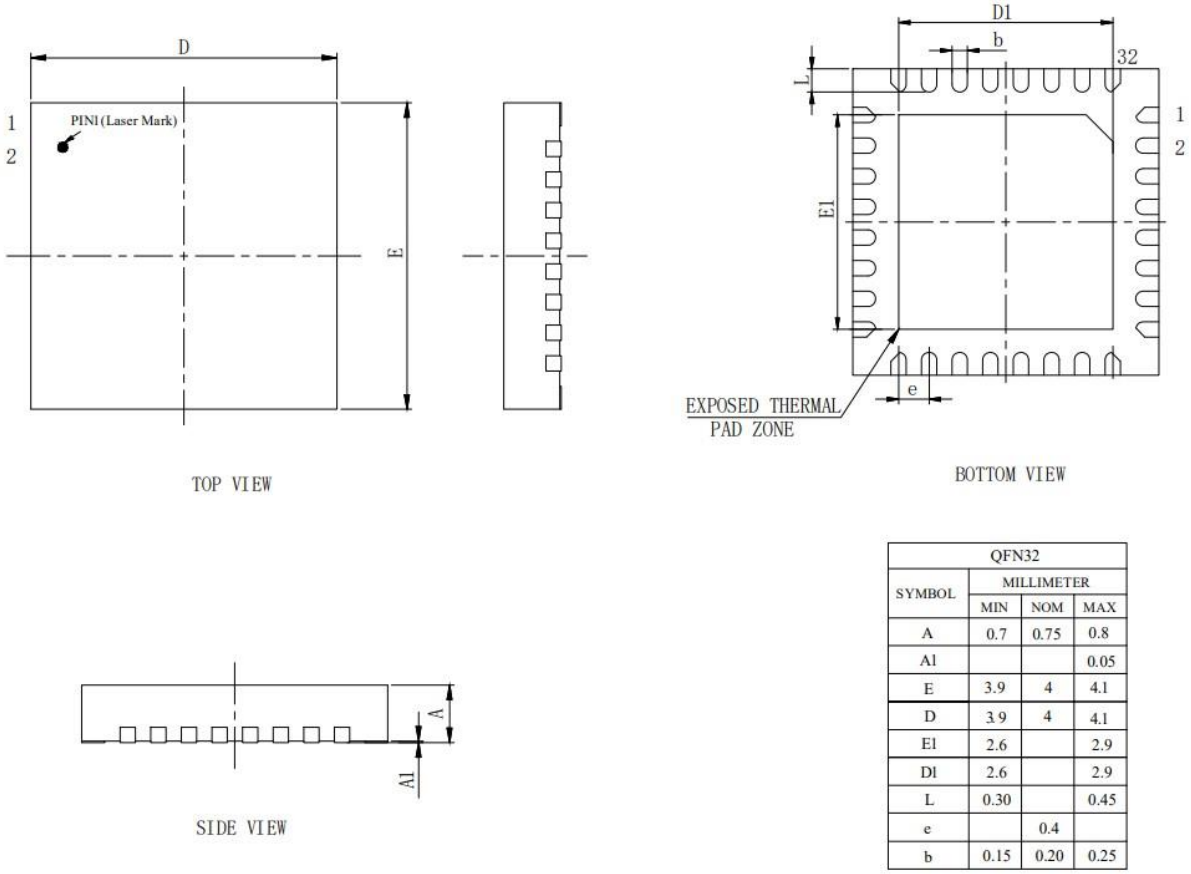
3.1 LQFP48_7X7 (FU6813L)



SYMBOL	LQFP48 MILLIMETER		
	MIN	SPEC	MAX
A			1.6
A1	0.05	0.1	0.15
A2	1.35	1.4	1.45
E	8.8	9	9.2
D	8.8	9	9.2
E1	6.9	7	7.1
D1	6.9	7	7.1
L1		1	
L	0.45		0.8
e		0.5	
b	0.18		0.27

Figure 3-1 LQFP48_7X7 Package Drawings and Dimensions

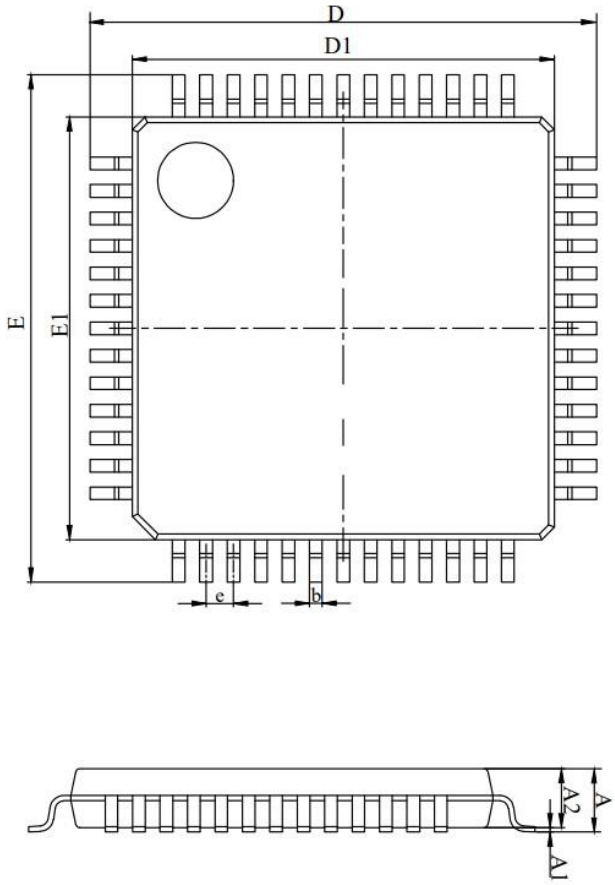
3.2 QFN32_4X4 (FU6813N)

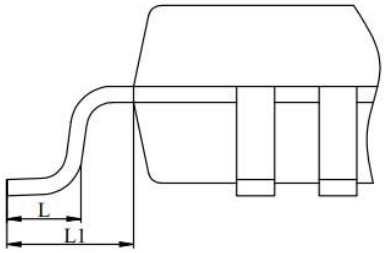


QFN32			
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.7	0.75	0.8
A1			0.05
E	3.9	4	4.1
D	3.9	4	4.1
E1	2.6		2.9
D1	2.6		2.9
L	0.30		0.45
e		0.4	
b	0.15	0.20	0.25

Figure 3-2 QFN32_4X4 Package Drawings and Dimensions

3.3 LQFP52_10X10 (FU6813P)

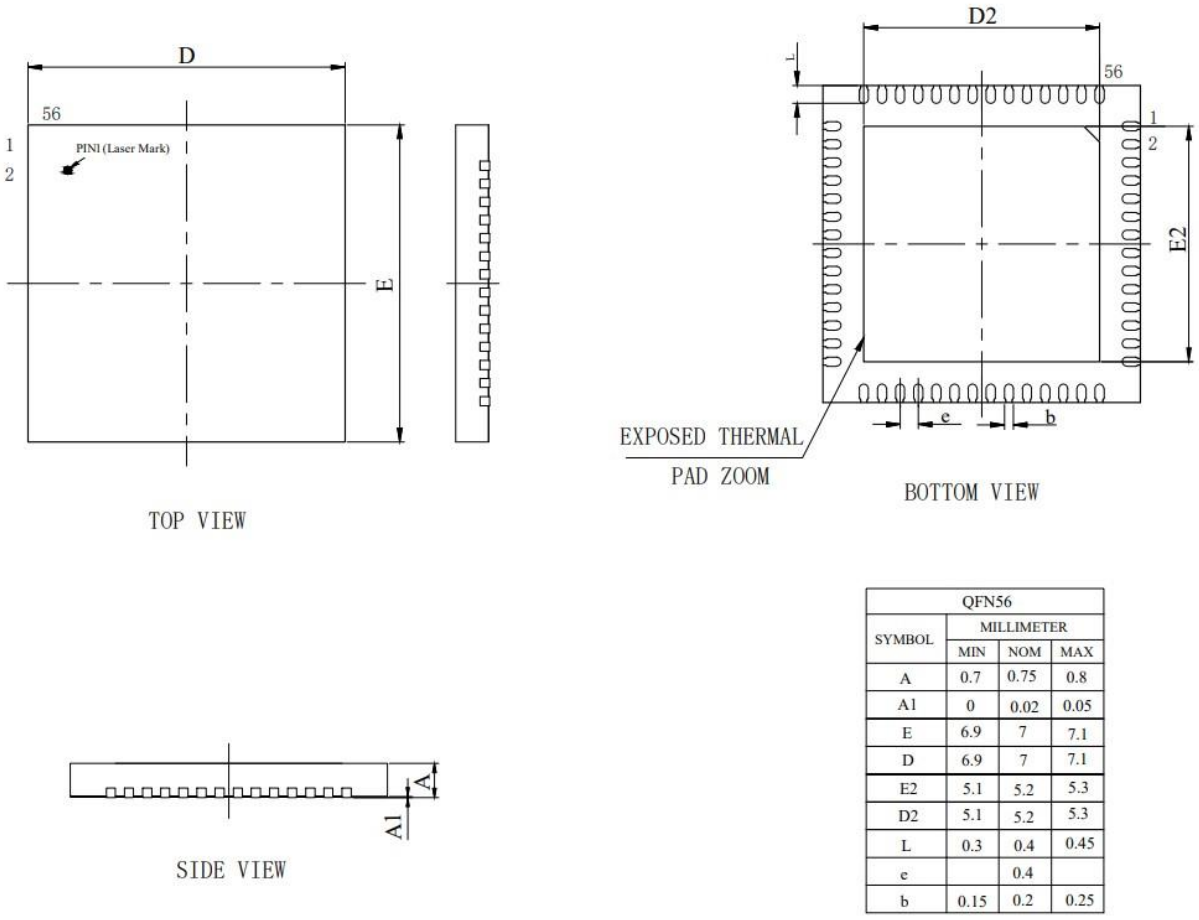




LQFP52			
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A			1.6
A1	0.05		0.2
A2	1.3	1.4	1.5
E	11.8	12	12.2
D	11.8	12	12.2
E1	9.9	10	10.1
D1	9.9	10	10.1
L1		1	
L	0.45	0.6	0.75
e	0.55	0.65	0.75
b	0.28		0.37

Figure 3-3 LQFP52_10X10 Package Drawings and Dimensions

3.4 QFN56_7X7 (FU6863Q)



QFN56			
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.7	0.75	0.8
A1	0	0.02	0.05
E	6.9	7	7.1
D	6.9	7	7.1
E2	5.1	5.2	5.3
D2	5.1	5.2	5.3
L	0.3	0.4	0.45
e		0.4	
b	0.15	0.2	0.25

Figure 3-4 QFN56_7X7 Package Drawings and Dimensions

4 Ordering Information

Table 4-1 Model Selections

Model	MIPS (Peak)	FLASH(KB)	XRAM(KB)	Clock Circuit				Driver Interface		Driver Type			I²C/UART/SPI	DMA	GPIO	Timer	Analog Peripherals								Lead-free	Package
																	ADC			DAC						
				Internal Fast Clock	External Fast Clock	Internal Slow Clock	External Slow Clock	6N Pre-driver	PWM	Square Wave	SVPWM	FOC					Number	Channel	Bits	Number	Bits	VREF	Operational Amplifier	Comparator		
FU6813L	24	32	1.5	√	—	√	√	—	√	√	√	√	√	√	34	6	1	14	12	3	9/8/6	√	4	4	√	LQFP48 (7x7 mm)
FU6813N	24	32	1.5	√	—	√	—	—	√	√	√	√	√	√	20	6	1	9	12	3	9/8/6	√	2	4	√	QFN32 (4x4 mm)
FU6813P	24	32	1.5	√	—	√	√	—	√	√	√	√	√	√	35	6	1	14	12	3	9/8/6	√	4	4	√	LQFP52 (10x10mm)
FU6863Q	24	32	1.5	√	—	√	√	√	—	√	√	√	√	√	32	6	1	14	12	3	9/8/6	√	4	4	√	QFN56 (7x7 mm)

5 Electrical Characteristics

5.1 Absolute Maximum Ratings

Stress values greater than "Absolute Maximum Ratings" listed in Table 5-1 ~ Table 5-4 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.

5.1.1 FU6813L

Table 5-1 Absolute Maximum Ratings of FU6813L

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Operating Junction Temperature T_J		-40	—	150	℃
Storage Temperature		-65	—	150	℃
VCC to VSS Voltage		-0.3	—	36	V
VDD5 to VSS Voltage		-0.3	—	6.5	V
RSTN/VCC_MODE/GPIO to VSS Voltage		-0.3	—	VDD5 + 0.3	V
Operating Ambient Temperature T_A		-40	—	85	℃
Operating Ambient Temperature T_A	Necessary Conditions: VCC ≤ 12V and I_{vcc} ≤ 30mA	-40	—	105	℃
Operating Ambient Temperature T_A	Low-voltage Single-power Supply Mode: VCC = VDD5 = 5V	-40	—	125	℃

5.1.2 FU6813N

Table 5-2 Absolute Maximum Ratings of FU6813N

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Operating Junction Temperature T_J		-40	—	150	℃
Storage Temperature		-65	—	150	℃
VCC to VSS Voltage		-0.3	—	36	V
VDD5 to VSS Voltage		-0.3	—	6.5	V
RSTN/VCC_MODE/GPIO to VSS Voltage		-0.3	—	VDD5 + 0.3	V
Operating Ambient Temperature T_A		-40	—	85	℃
Operating Ambient Temperature T_A	Necessary Conditions: VCC ≤ 12V and I_{vcc} ≤ 30mA	-40	—	105	℃
Operating Ambient Temperature T_A	Low-voltage Single-power Supply Mode: VCC = VDD5 = 5V	-40	—	125	℃

5.1.3 FU6813P

Table 5-3 Absolute Maximum Ratings of FU6813P

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Operating Junction Temperature T_J		-40	—	150	℃
Storage Temperature		-65	—	150	℃
VCC to VSS Voltage		-0.3	—	36	V
VDD5 to VSS Voltage		-0.3	—	6.5	V
RSTN/VCC_MODE/GPIO to VSS Voltage		-0.3	—	VDD5 + 0.3	V
Operating Ambient Temperature T_A		-40	—	85	℃
Operating Ambient Temperature T_A	Low-voltage Single-power Supply Mode: VCC = VDD5 = 5V	-40	—	125	℃

5.1.4 FU6863Q

Table 5-4 Absolute Maximum Ratings of FU6863Q

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Operating Junction Temperature T_J		-40	—	150	℃
Storage Temperature		-65	—	150	℃
VCC to VSS Voltage		-0.3	—	36	V
VDD5 to VSS Voltage		-0.3	—	6.5	V
RSTN/VCC_MODE/GPIO to VSS Voltage		-0.3	—	VDD5 + 0.3	V
Operating Ambient Temperature T_A		-40	—	85	℃
Operating Ambient Temperature T_A	Necessary Conditions: VCC ≤ 12V and I_{vcc} ≤ 30mA	-40	—	105	℃
VDRV to VSS Voltage		-0.3	—	22	V
Low-side Output Voltage $V_{LU,LV,LW}$		-0.3	—	VDRV + 0.3	V
High-side Floating Voltage $V_{BU,BV,BW}$		-0.3	—	205	V
High-side Floating Offset Voltage $V_{SU,SV,SW}$		$V_{BU,BV,BW}$ - 22	—	$V_{BU,BV,BW}$ + 0.3	V
High-side Output Voltage $V_{HU,HV,HW}$		$V_{SU,SV,SW}$ - 0.3	—	$V_{BU,BV,BW}$ + 0.3	V

5.2 Global Electrical Characteristics

5.2.1 FU6813

Table 5-5 Global Electrical Characteristics of FU6813

(T_A = 25℃ and VCC = 5V~24V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
VCC Operating Voltage[1]	High-voltage Single-power Supply Mode: VCC_MODE = 0, T_A = -40℃ ~ 85℃	5	—	24	V
	High-voltage Dual-power Supply Mode[2]: VCC_MODE = 1, VCC ≥ VDD5, T_A = -40℃ ~ 85℃	5	—	36	V
	Low-voltage Single-power Supply Mode[2]: VCC_MODE = 1; VCC pin is connected to VDD5 pin; T_A = -40℃ ~ 85℃	3	—	5.5	V
VDD5 Operating Voltage	VCC_MODE = 1 and VCC pin is connected with VDD5 pin[2]	3	—	5.5	V
I_{VCC} Operating Current[3]		—	24	—	mA
I_{VCC} Standby Current[3]		—	6	—	mA
I_{VCC} Sleep-mode Current		—	50	150	μA

5.2.2 FU6863

Table 5-6 Global Electrical Characteristics of FU6863

(T_A = 25℃ and VCC = 5V~24V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
VCC Operating Voltage[1]	High-voltage Single-power Supply Mode: VCC_MODE = 0, T_A = -40℃ ~ 85℃	5	—	24	V
	High-voltage Dual-power Supply Mode[2]: VCC_MODE = 1, VCC ≥ VDD5, T_A = -40℃ ~ 85℃	5	—	36	V
VDD5 Operating Voltage	VCC_MODE = 1; VCC pin is connected to VDD5 pin[2]; T_A = -40℃ ~ 85℃	3	—	5.5	V
High-side Floating Voltage $V_{BU,BV,BW}$		—	—	205	V
V_{BU} to V_{SU} Voltage V_{BV} to V_{SV} Voltage V_{BW} to V_{SW} Voltage		—	—	18	V
I_{VCC} Operating Current[3]		—	24	—	mA
I_{VCC} Standby Current[3]		—	6	—	mA
I_{VCC} Sleep-mode Current	T_A = 25℃	—	300	500	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 5V~5.5V during Flash write or erase.

[3] Characteristics may vary with different configurations.

5.3 GPIO Electrical Characteristics

Table 5-7 GPIO Electrical Characteristics

(T_A = 25℃, VCC = 5V~24V and VCC_MODE = 0 unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Output Rise Time	50pF load, from 10% to 90%, T_A = 25℃	—	15	—	ns
Output Fall Time	50pF load, from 90% to 10%, T_A = 25℃	—	13	—	ns
V_{OH} High-level Output Voltage	I_{OH} = 4mA, T_A = -40℃ ~ 85℃	VDD - 0.7	—	—	V
V_{OL} Low-level Output Voltage	I_{OL} = 4mA, T_A = -40℃ ~ 85℃	—	—	0.7	V
V_{IH} High-level Input Voltage[1]		0.7*VDD5	—	—	V
V_{IL} Low-level Input Voltage		—	—	0.2*VDD5	V
Pull-up Resistor[2]	V_{in} = 0V, T_A = 25℃	—	33	—	kΩ
Pull-up Resistor[3]	V_{in} = 0V, T_A = 25℃	—	5	—	kΩ

Notes:

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

[2] GPIOs except P0[2: 0], P1[6:3], P2[1] and P3[7:6].

[3] P0[2:0], P1[6:3], P2[1] and P3[7:6]

5.4 PWM IO Electrical Characteristics (FU6813)

Table 5-8 PWM IO Electrical Characteristics of FU6813

(T_A = 25℃, VCC = 5V~24V and VCC_MODE = 0 unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Output Source Current	P1_AN[HDIO] = 1	—	50	—	mA
Output Sink Current	P1_AN[HDIO] = 1	—	100	—	mA
Output Rise Time	50pF load, from 10% to 90%, T_A = 25℃	—	7	—	ns
Output Fall Time	50pF load, from 90% to 10%, T_A = 25℃	—	5	—	ns

5.5 6N Pre-driver IO Electrical Characteristics (FU6863)

Table 5-9 6N Pre-driver IO Electrical Characteristics of FU6863

(T_A = 25℃, VCC = 15V and VCC_MODE = 0 unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
High-level Peak Output Current		—	0.8	—	A
Low-level Peak Output Current		—	0.8	—	A
VDRV Operating Voltage		5	—	20	V
High-side Floating Voltage $V_{BU,BV,BW}$		—	—	180	V
High-side Floating Offset Voltage $V_{SU,SV,SW}$		$V_{BU,BV,BW}$ -20	—	$V_{BU,BV,BW}$ -5	V
VDRV UVLO Threshold Voltage		3.8	4.4	5	V
VDRV UVLO Release Voltage		3.5	4.1	4.7	V
VDRV UVLO Hysteresis Voltage		0.2	0.3	—	V
Output Rise Time	1nF load, from 10% to 90%	—	30	70	ns
Output Fall Time	1nF load, from 90% to 10%	—	30	70	nS
Deadtime	DT	—	100	—	ns

5.6 ADC Electrical Characteristics

Table 5-10 ADC Electrical Characteristics

(T_A = 25℃ and VCC = 5V~24V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
INL (Integral Nonlinearity)	12-bit	—	2	—	LSB
DNL (Differential Nonlinearity)	12-bit	—	1.5	—	LSB
OFFSET (Offset Error)	12-bit	—	10	—	LSB
SNR (Signal-to-noise Ratio)	f_{IN} = 350kHz	—	70.8	—	dB
ENOB (Effective Number of Bits)	f_{IN} = 350kHz	—	10.5	—	Bit
SFDR (Spurious-free Dynamic Range)	f_{IN} = 350kHz	—	68.2	—	dB
THD (Total Harmonic Distortion)	f_{IN} = 350kHz	—	67	—	dB
R_{IN} Input Resistance		—	800	—	Ω
C_{IN} Input Capacitance		—	30	—	pF
Conversion Time		—	13	—	ADCLK[1]
Sampling Time		3	—	63	ADCLK[1]

Note:

[1] ADCLK=12MHz

5.7 VREF and VHALF Electrical Characteristics

Table 5-11 VREF and VHALF Electrical Characteristics

(T_A = -40℃ ~85℃ and VCC =5V~24V)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
VREF	VREF_CR[VREFVSEL] = 00	4.3	4.5	4.7	V
	VREF_CR[VREFVSEL] = 01	—	VDD5	—	V
	VREF_CR[VREFVSEL] = 11	—	4	—	V
	VREF_CR[VREFVSEL] = 10	—	3	—	V
VHALF		VREF/2 - 0.2	VREF/2	VREF/2 + 0.2	V

5.8 Operational Amplifier Electrical Characteristics

Table 5-12 Operational Amplifier Electrical Characteristics

(T_A = 25℃ and VCC = 5V~24V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_{ICMR} Common-mode Input Voltage Range		0	—	VDD5 - 0.5	V
V_{OS} Operational Amplifier Offset Voltage	T_A = 25℃	—	5	10	mV
A_{OL} Open-loop Gain	R_L = 100kΩ	—	80	—	dB
Unity-gain Bandwidth (UGBW)	CL = 40pF	6	10	—	MHz
Slew Rate (SR)	CL = 40pF	10	15	—	V/μs

5.9 Hall/BEMF Electrical Characteristics

Table 5-13 Hall/BEMF Electrical Characteristics

(T_A = 25℃, VCC = 5V~24V and VCC_MODE = 0 unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
BEMF Built-in Resistor		5.4	6.8	8.2	kΩ
Relative Accuracy between BEMF Built-in Resistors		—	1	—	%

5.10 OSC Electrical Characteristics

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.

Table 5-14 OSC Electrical Characteristics

(TA = -40℃ ~85℃, VCC = 5V~24V and VCC_MODE = 0)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
System Clock Rate		23.5	24	24.5	MHz
Low-speed Clock Rate		29	32.8	37	kHz

5.11 Reset Electrical Characteristics

Table 5-15 Reset Electrical Characteristics

(T_A = 25℃, VCC = 5V~24V and VCC_MODE = 0 unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Minimum Time for RSTN Released to Low		—	25	50	μs
VDD5 Reset Threshold	Reset Voltage LVR = 3.0V	2.8	3.0	3.2	V

5.12 LDO Electrical Characteristics

Table 5-16 LDO Electrical Characteristics

(T_A = 25℃, VCC = 5V~24V and VCC_MODE = 0 unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
VDD5 Voltage	VCC = 7V ~ 24V, VCC_MODE = 0	4.7	5	5.3	V
VDD18 Voltage		—	1.85	—	V

5.13 Package Thermal Resistance

5.13.1 FU6813L LQFP48

Table 5-17 FU6813L LQFP48 Package Thermal Resistance

Parameter	Test Conditions	Value	Unit
Junction-to-ambient Temperature Thermal Resistance θ_{JA}[1]	JEDEC standard, 2S2P PCB	52.4	℃/W
	JEDEC standard, 1S0P PCB	72.2	℃/W
Junction-to-case Temperature Thermal Resistance θ_{JC}[1]	JEDEC standard, 1S0P PCB	17	℃/W

5.13.2 FU6813N QFN32

Table 5-18 FU6813N QFN32 Package Thermal Resistance

Parameter	Test Conditions	Value	Unit
Junction-to-ambient Temperature Thermal Resistance θ_{JA}[1]	JEDEC standard, 2S2P PCB	47	℃/W
	JEDEC standard, 1S0P PCB	74	℃/W
Junction-to-case Temperature Thermal Resistance θ_{JC}[1]	JEDEC standard, 1S0P PCB	20	℃/W

5.13.3 FU6813P LQFP52

Table 5-19 FU6813P LQFP52 Package Thermal Resistance

Parameter	Test Conditions	Value	Unit
Junction-to-ambient Temperature Thermal Resistance θ_{JA}[1]	JEDEC standard, 2S2P PCB	43	℃/W
Junction-to-case Temperature Thermal Resistance θ_{JC}[1]	JEDEC standard, 2S2P PCB	13	℃/W

5.13.4 FU6863Q QFN56

Table 5-20 FU6863Q QFN56 Package Thermal Resistance

Parameter	Test Conditions	Value	Unit
Junction-to-ambient Temperature Thermal Resistance θ_{JA}[1]	JEDEC standard, 2S2P PCB	33	℃/W
	JEDEC standard, 1S0P PCB	55	℃/W
Junction-to-case Temperature Thermal Resistance θ_{JC}[1]	JEDEC standard, 1S0P PCB	9.2	℃/W

Note:

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

6 Reset Control

6.1 Reset Source (RST_SR)

The chip includes a reset circuitry with 8 reset sources:

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

The reset flag is queryable and recorded in register RST_SR. Following the last reset, the affected reset flag is set to "1" and all other reset flags are cleared to "0". In order to clear a reset flag, you can set RST_SR[RSTCLR] flag to "1".

6.2 Reset Enable

See corresponding control registers. The control bits EOSRSTEN, LVDENB and WDTRSTEN enable or disable EOS, LVD and WDT reset sources respectively.

6.3 External Reset and Power-on Reset

The chip resets when RSTN pin remains low for 25μs. After reset, MCU starts the program from address 0x0000.

6.4 Low Voltage Detector Reset

The chip's internal circuitry monitors VDD. When VDD voltage 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 the WDT. When WDT reaches its maximum value, it 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 the software using MOVX instruction for read/write/erase operations (See Code Protection for details). A Flash error detector reset (RSTFED) occurs if a Flash erase

is attempted targeting the last sector or a Flash write is attempted targeting the last byte. RSTFED is always enabled and cannot be disabled.

6.7 Reset Registers

6.7.1 RST_SR (0xC9)

Bit	7	6	5	4	3	2	1	0
Name	RSTPOW/ RSTCLR	RSTEXT	RSTLVD	RSTEOS	RSTWDT	RSTFED	RSTDBG	SOFTR
Type	R/W	R	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: This bit is used to clear analog reset flag register. 1: RST_SR[7:3] is 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 pin reset.
[5]	RSTLVD	Low Voltage Detection (LVD) Reset Flag 0: Last reset was not an LVD reset. 1: Last reset was an LVD reset.
[4]	RSTEOS	Electrical Overstress Reset Flag 0: Last reset was not an electrical overstress reset. 1: Last reset was an electrical overstress reset.
[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 Control

7.1 Introduction

The chip includes an interrupt system with a total of 15 interrupt sources. Each interrupt source can be individually programmed in IP0 ~ IP2 registers with one of four priority levels. Interrupt flags (IF) are located in an SFRs or XSFRs. The associated IF is set by the hardware to 1 when the internal circuitry or an external source 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 generated and 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 interrupted 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 level, 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. The MCU does not respond to any interrupt request when IE[EA] = 0.

Each interrupt source 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. Once the enable bit is set to 1, the MCU immediately enters the interrupt subroutine. 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. They both can be configured as interrupt on rising edge, interrupt on falling edge or interrupt on edge changes (rise or fall).

The digital input signals from P0.0~P0.6 and output signals from CMP4 can be used to trigger an INT0. The interrupt source is selected through LVSR[EXT0CFG] bit. These trigger sources share one interrupt entry point, 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 P1.0~P1.7 and P2.0~P2.7 can be used to trigger an INT1. P1_IF and P2_IF are interrupt flag bits, and P1_IE and P2_IE are interrupt enable bits. Each trigger source has a

corresponding interrupt flag bit and an interrupt enable bit. INT1 can select multiple trigger sources that are recognized by P1_IF and P2_IF in the interrupt subroutine. These 15 interrupt sources share one interrupt entry and one interrupt enable bit IE[EX1]. To enable INT1, first set IE[EX1] to "1" and then configure the corresponding enable bit. The interrupt edge is configured by TCON[IT1] bit, and the priority level by IP0[PX1] bit. See 7.5.7 P1_IE (0xD1) ~ 7.5.10 P2_IF (0xD4) for INT1 interrupt flags and enable registers.

7.4 Interrupt Summary

Table 7-1 Interrupt Summary

Interrupt Source	Priority Order	Interrupt Vector	Interrupt Flag	Cleared by Software?	Enable Bit	Priority Control
Reset	Highest	0x0000	None	N	Always enabled	Highest
LVW Interrupt TSD Interrupt	0	0x0003	LVSR[0] TCON[5]	Y	CCFG1[6] IE[1]	IP0[1:0]
INT0	1	0x000B	TCON[2]	Y	IE[0]	IP0[3:2]
INT1	2	0x0013	P1_IF[7:0] P4_IF[7:0]	Y	IE[2]	IP0[5:4]
FG Interrupt DRV Compare Match Interrupt	3	0x001B	DRV_SR[5:4]	Y	DRV_SR[3] DRV_SR[2:0]	IP0[7:6]
Timer2 Interrupt	4	0x0023	TIM2_CR1[7:5]	Y	TIM2_CR1[4:3] TIM2_CR0[3]	IP1[1:0]
Timer1 Interrupt	5	0x002B	TIM1_SR[5:0]	Y	TIM1_IER[5:0]	IP1[3:2]
ADC Interrupt	6	0x0033	ADC_CR[0]	Y	ADC_CR[1]	IP1[5:4]
CMP0/1/2 Interrupt Hall Interrupt	7	0x003B	CMP_SR[6:4] HALL_CR[7]	Y	CMP_CR0[5:0] HALL_CR[6]	IP1[7:6]
RTC Interrupt	8	0x0043	RTC_STA[6]	Y	IE[6]	IP2[1:0]
Timer3 Interrupt	9	0x004B	TIM3_CR1[7:5]	Y	TIM3_CR1[4:3] TIM3_CR0[3]	IP2[3:2]
Systick Interrupt	10	0x0053	DRV_SR[7]	Y	DRV_SR[6]	IP2[5:4]
Timer4 Interrupt	11	0x005B	TIM4_CR1[7:5]	Y	TIM4_CR1[4:3] TIM4_CR0[3]	IP2[7:6]
RSV	12	0x0063		Y		
I^2C Interrupt UART1 Interrupt	13	0x006B	I2C_SR[0] UT_CR[1:0]	Y	I2C_CR[0] IE[4]	IP3[3:2]
SPI Interrupt UART2 Interrupt	14	0x0073	SPI_CR1[7:4] UT2_CR[1:0]	Y	IE[3] UT2_BAUDH[5]	IP3[5:4]
DMA Interrupt	15	0x007B	DMA0_CR0[0] DMA1_CR0[0]	Y	DMA0_CR0[2]	IP3[7:6]

The chip includes an interrupt system with a total of 15 interrupt sources. Each interrupt source can be individually programmed in IP0 ~ IP3 registers with one of four priority levels. A low priority interrupt can

be interrupted by a high priority interrupt. If two interrupt requests are generated at the same time, the interrupt with the higher priority is serviced first. See the above table for the interrupt sources and default priority orders, where the lower the mark the higher the priority level. An interrupt will not be preempted by another of the same priority level.

IE[EA] is the global interrupt enable bit. The MCU does not respond to any interrupt request when IE[EA] = 0.

7.5 Interrupt Registers

7.5.1 IE (0xA8)

Bit	7	6	5	4	3	2	1	0
Name	EA	RTCIE	RSV	ES0	SPIIE	EX1	TSDIE	EX0
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]	EA	Enable All Interrupts 0: Disable 1: Enable
[6]	RTCIE	RTC Interrupt Enable 0: Disable 1: Enable
[5]	RSV	Reserved
[4]	ES0	UART1 Interrupt Enable 0: Disable 1: Enable
[3]	SPIIE	SPI Interrupt Enable 0: Disable 1: Enable
[2]	EX1	External Interrupt 1 (INT1) Enable 0: Disable 1: Enable
[1]	TSDIE	TSD Interrupt Enable 0: Disable 1: Enable
[0]	EX0	External Interrupt 0 (INT0) 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	FG/DRV Compare Match Interrupt Priority Setting
[5:4]	PX1	External Interrupt 1 (INT1) Priority Setting
[3:2]	PX0	External Interrupt 0 (INT0) Priority Setting
[1:0]	PLVW_TSD	LVW/TSD Interrupt Priority Setting

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	PCMP		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]	PCMP	CMP0/1/2 Interrupt Priority Setting
[5:4]	PADC	ADC Interrupt Priority Setting
[3:2]	PTIM1	Timer1 Interrupt Priority Setting
[1:0]	PTIM2	Timer2 Interrupt Priority Setting

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		PSYSTICK		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 Setting
[5:4]	PSYSTICK	Systick Interrupt Priority Setting
[3:2]	PTIM3	Timer3 Interrupt Priority Setting
[1:0]	PRTC	RTC Interrupt Priority Setting

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	DMA		PSPI_UT2		PI2C_UT1		RSV	
Type	R/W	R/W	R/W	R/W	R/W	R/W	-	-
Reset	0	0	0	0	0	0	-	-

Bit	Name	Description
[7:6]	DMA	DMA Interrupt Priority Setting
[5:4]	PSPI_UT2	SPI/UART2 Interrupt Priority Setting
[3:2]	PI2C_UT1	I^2C/UART1 Interrupt Priority Setting
[1:0]	RSV	Reserved

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	RSV		TSDIF	IT1		IF0	IT0	
Type	-	-	R/W0	R/W	R/W	R/W0	R/W	R/W
Reset	-	-	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	TSDIF	TSD Interrupt Flag This bit is set by hardware to “1” 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 is often used with the overtemperature status bit LVSR[TSDF].
[4:3]	IT1	External Interrupt 1 (INT1) Edge Select 00: Interrupt on rising edge 01: Interrupt on falling edge 1X: Interrupt on edge changes (rise or fall)
[2]	IF0	External Interrupt 0 (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	External Interrupt 0 (INT0) Edge Select 00: Interrupt on rising edge 01: Interrupt on falling edge 1X: Interrupt on edge changes (rise or fall)

7.5.7 P1_IE (0xD1)

Bit	7	6	5	4	3	2	1	0
Name	P17_IE	P16_IE	P15_IE	P14_IE	P13_IE	P12_IE	P11_IE	P10_IE
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]	P17_IE	P1.7 INT1 Enable 0: Disable 1: Enable
[6]	P16_IE	P1.6 INT1 Enable 0: Disable 1: Enable
[5]	P15_IE	P1.5 INT1 Enable 0: Disable 1: Enable
[4]	P14_IE	P1.4 INT1 Enable 0: Disable 1: Enable
[3]	P13_IE	P1.3 INT1 Enable 0: Disable 1: Enable

[2]	P12_IE	P1.2 INT1 Enable 0: Disable 1: Enable
[1]	P11_IE	P1.1 INT1 Enable 0: Disable 1: Enable
[0]	P10_IE	P1.0 INT1 Enable 0: Disable 1: Enable

7.5.8 P1_IF (0xD2)

Bit	7	6	5	4	3	2	1	0
Name	P17_IF	P16_IF	P15_IF	P14_IF	P13_IF	P12_IF	P11_IF	P10_IF
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]	P17_IF	P1.7 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[6]	P16_IF	P1.6 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[5]	P15_IF	P1.5 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[4]	P14_IF	P1.4 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[3]	P13_IF	P1.3 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[2]	P12_IF	P1.2 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[1]	P11_IF	P1.1 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[0]	P10_IF	P1.0 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending

7.5.9 P2_IE (0xD3)

Bit	7	6	5	4	3	2	1	0
Name	P27_IE	P26_IE	P25_IE	P24_IE	P23_IE	P22_IE	P21_IE	P20_IE
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]	P27_IE	P2.7 INT1 Enable 0: Disable 1: Enable
[6]	P26_IE	P2.6 INT1 Enable 0: Disable 1: Enable

[5]	P25_IE	P2.5 INT1 Enable 0: Disable 1: Enable
[4]	P24_IE	P2.4 INT1 Enable 0: Disable 1: Enable
[3]	P23_IE	P2.3 INT1 Enable 0: Disable 1: Enable
[2]	P22_IE	P2.2 INT1 Enable 0: Disable 1: Enable
[1]	P21_IE	P2.1 INT1 Enable 0: Disable 1: Enable
[0]	P20_IE	P2.0 INT1 Enable 0: Disable 1: Enable

7.5.10 P2_IF (0xD4)

Bit	7	6	5	4	3	2	1	0
Name	P27_IF	P26_IF	P25_IF	P24_IF	P23_IF	P22_IF	P21_IF	P20_IF
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]	P27_IF	P2.7 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[6]	P26_IF	P2.6 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[5]	P25_IF	P2.5 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[4]	P24_IF	P2.4 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[3]	P23_IF	P2.3 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[2]	P22_IF	P2.2 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[1]	P21_IF	P2.1 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending
[0]	P20_IF	P2.0 INT1 Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending

8 I²C

8.1 I²C Introduction

The I²C module provides an industry standard two-wire serial interface and is a simple bi-directional synchronous serial bus for communication between MCU and external I²C devices, as shown in Figure 8-1. The bus consists of two serial lines: SDA (serial data line) and SCL (serial clock line). P0.0 serves as SDA port and P0.1 as SCL port. After I²C is enabled, P0.0 and P0.1 automatically shifts into open-drain outputs.

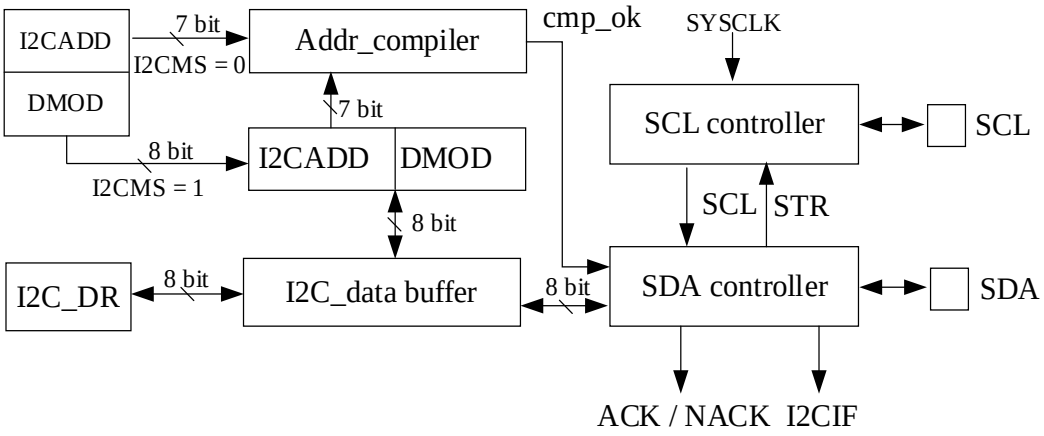


Figure 8-1 I²C Block Diagram

Features:

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

Both SDA and SCL lines are high level 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 I²C 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.

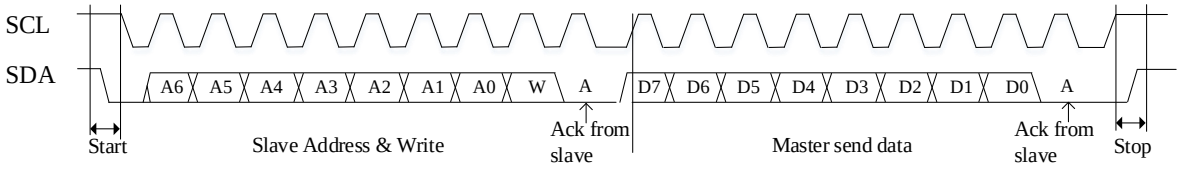


Figure 8-2 Master Transmits Data to Slave

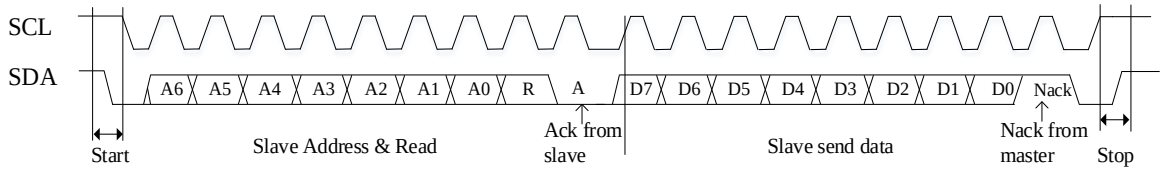


Figure 8-3 Master Receives Data from Slave

8.2 I²C Operations

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 I²C;
6. Set I2C_SR[I2CSTA] to “1” to transmit 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. Sending Data: Write the data to I2C_DR register. The master starts to transmit data when I2C_SR[STR] is reset and SCL is released. After the data is transmitted and ACK/NACK is received, I2C_SR[STR] is set to “1” by hardware and SCL is pulled LOW by the master;
8. Receiving Data: The master starts to receive data when I2C_SR[STR] is reset and SCL is released. After 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 transmit 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” and 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 I²C;
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. Sending Data: 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 to “1” by hardware and SCL is pulled LOW by the slave;
6. Receiving Data: 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 to “1” by hardware and SCL is pulled LOW by the slave;
7. RESTART: If the slave is processing a service when receiving START signal, it stops the current routine and waits for receiving address.

8.2.3 I²C Interrupt Sources

The interrupt sources of I²C 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 I²C 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	I²C Enable Enable the associated GPIO and switch to I²C mode, serving as collector open-drain output. The pull-up setting decides whether to pull I²C HIGH. 0: Disable 1: Enable
[6]	I2CMS	Master/Slave Mode Selection 0: Slave 1: Master
[5:3]	RSV	Reserved
[2:1]	I2CSPD	I²C transfer rate setting, valid only in Master Mode 00: 100kHz 01: 400kHz 10: 1MHz 11: Reserved

[0]	I2CIE	I²C 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, valid only in Slave Mode 0: General call is disabled 1: General call is enabled namely, 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	I²C 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	I²C Busy Flag When I2C_CR[I2CEN] = 0, I2C_SR[I2CBSY] is cleared to “0” by hardware. Master Mode: After START is transmitted, I2C_SR[I2CBSY] is set to “1” by hardware; after STOP is transmitted, I2C_SR[I2CBSY] is cleared to “0” by hardware. Slave Mode: After START is received and address matches, I2C_SR[I2CBSY] is set to “1” by hardware; after STOP is received, I2C_SR[I2CBSY] is cleared to “0” by hardware.
[6]	DMOD	I²C R/W Flag 0: WRITE (master transmits the data, and slave receives the data) 1: READ (master receives the data, and slave transmits the 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”, 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 bytes
1: Transmit START or RESTART and address bytes

Slave Mode:
This bit is set to “1” after hardware receives START and address matches, and cleared to “0” by software.

Table 8-1 Mapping of I2C_SR[I2CSTA] and I2C_SR[I2CSTP] in Slave Mode
<table><tr><th>I2CSTA</th><th>I2CSTP</th><th>I²C 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 bytes</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] = 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 I2C_SR[I2CSTA] and I2C_SR[I2CSTP] are written to “1” at the same time and I2C_SR[I2CBSY] is “1”, I²C first sends STOP, 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.
See Table 8-1 for status flags.

Note: When I2C_CR[I2CEN] = 0, I2C_SR[I2CSTP] is automatically cleared to “0” by hardware.</td></tr>
<tr><td>[2]</td><td>STR</td><td>I²C 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:
When DATA byte is received or START is received and address matches, I2C_SR[STR] is set to “1” by hardware and SCL is pulled LOW. SCL is released after I2C_SR[STR] is cleared by software.

Note: The 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 I²C. 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 transmit ACK/NACK after the 8th bit of data is received.</td></tr>
</table>

		0: Bit9 transmits ACK 1: Bit9 transmits NACK When the device is in WRITE mode, I2C_SR[NACK] is read to receive ACK/NACK after the 8th bit of data is transmitted. 0: Bit9 receives ACK 1: Bit9 receives NACK
[0]	I2CIF	I2C Interrupt Flag 0: No Interrupt Pending 1: Interrupt Pending When I2C_SR[STR] = 1, an interrupt is generated in both Master and Slave modes. When I2C_SR[I2CSTP] = 1, an interrupt is generated only in Slave mode.

9 SPI

9.1 SPI Introduction

SPI provides access to a high-speed and full-duplex synchronous serial bus, with its block diagram shown in Figure 9-1. SPI can operate as a master or slave device in 3-wire or 4-wire mode, and supports multiple masters and slaves on a single SPI bus.

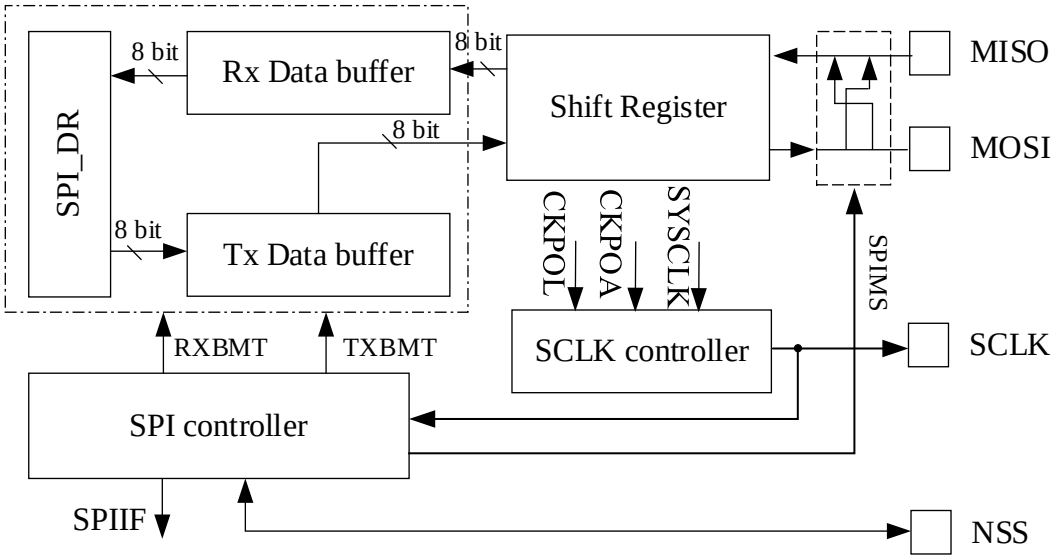


Figure 9-1 SPI Block Diagram

9.2 SPI Operations

9.2.1 Signal Descriptions

The four signals for SPI are MOSI, MISO, SCLK and NSS.

9.2.1.1 Master Output, Slave Input (MOSI)

The master-out, slave-in (MOSI) signal is an output from a master device and an input to slave devices. It is used to serially transfer data from the master to the slave. Data is transferred with most-significant bit (MSB) first, namely, the master begins its transmission by driving the MSB of the shift register on its MOSI pin.

9.2.1.2 Master Input, Slave Output (MISO)

The MISO signal is an output from a slave device and an input to the master device. The MISO pin is placed in a high-impedance state when the SPI module is disabled or when the SPI operates in 4-wire mode as a slave that is not selected. When the SPI acts as a slave in 3-wire mode or operates in 4-wire mode as a slave that is selected, MISO is used to serially transfer data from the slave to the master. Data is transferred with most-significant bit (MSB) first, namely, the master begins its transmission by driving the MSB of the shift register on its MOSI pin.

9.2.1.3 Serial Clock (SCLK)

The serial clock (SCLK) signal is an output from the master device and an input to slave devices. It is used to synchronize serial data transmission between the master and slave. SCLK signal is generated by SPI operating as a master. In 4-wire slave mode, SCLK is ignored when the slave device is not selected (NSS=1).

9.2.1.4 Slave Select (NSS)

The slave-select (NSS) is dependent on the setting of SPI_CR1[NSSMID] bit. SPI may operate in 3-Wire Mode, 4-Wire Slave/Multi-Master Mode or 4-Wire Single Master Mode. When SPI operates in 4-Wire Slave/Multi-Master Mode, NSS is enabled as an input. In this mode, a particular SPI master function is disabled to prevent SPI bus collision where two or more masters simultaneously initiate data transfer. When SPI operates in 4-Wire Single Master Mode, the master NSS is configured as chip select output. When SPI operates in 3-Wire Mode, NSS is disabled. When SPI operates as a master, multiple addressed slave devices can be selected using general-purpose I/O pins.

When SPI_CR1[NSSMID] = 00, SPI operates in 3-Wire Mode. NSS port is not necessary in this mode and there is only one master and one slave on the SPI bus. The connection diagram is shown in Figure 9-2.

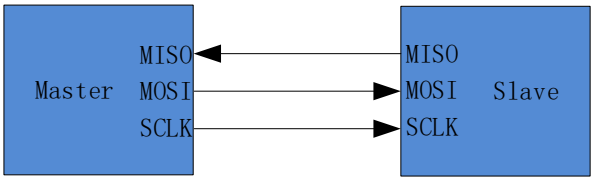


Figure 9-2 Connection Diagram of 3-Wire SPI Mode

When SPI_CR1[NSSMID] = 01, SPI operates in 4-Wire Slave/Multi-Master Mode. In this mode, NSS pins on the SPI bus are configured as inputs, waiting to be addressed by the master. When SPI_CR0[SPIMS] = 0, SPI operates in 4-Wire Slave Mode. If NSS is set to “0”, the slave is selected; while NSS is set to “1”, the slave is not selected. When SPI_CR0[SPIMS] = 1, SPI operates in Master Mode and defaults to Multi-Master Mode. When SPI operates in Multi-Master Mode, NSS is configured as an input to disable the master SPI. When NSS pin of a master on the SPI bus is pulled low, SPI_CR0[SPIMS] and SPI_CR1[SPIEN] are cleared to “0” by hardware to disable the SPI, and the Mode Fault Flag SPI_CR1[MODF] is set to “1”. In this case, SPI communication remains halted before the SPI is re-enabled by software. In this mode, multiple masters are allowed for communication on the SPI bus. The connection diagram is shown in Figure 9-3.

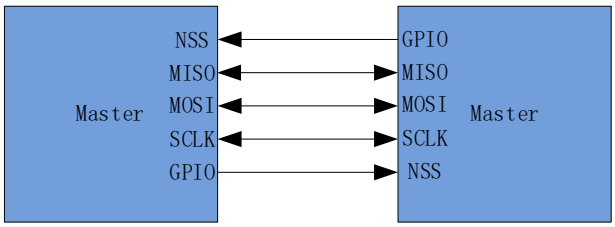


Figure 9-3 Connection Diagram of 4-Wire Multi-Master Mode

When SPI_CR1[NSSMID] = 1X, SPI operates in 4-Wire Single Master Mode. In this mode, NSS pin of the master on the bus is configured as an output, and NSS pin of the slave devices are configured as inputs. SPI_CR1[NSSMID] setting decides the output level of NSS pin serving as signal to select a slave. Other slaves can be selected using GPIO pins. The connection diagram is shown in Figure 9-4.

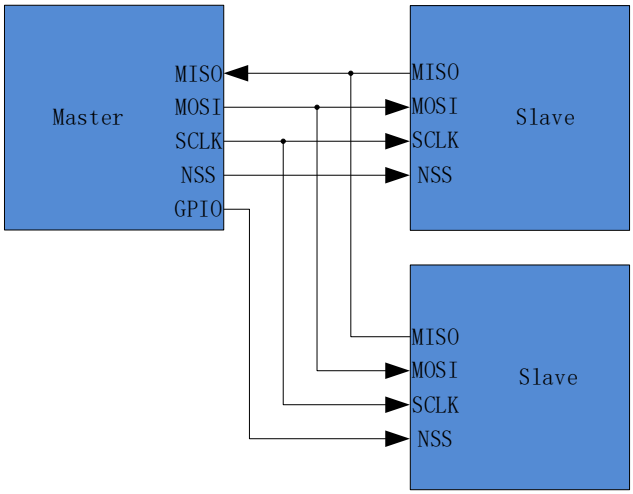


Figure 9-4 Connection Diagram of 4-Wire Single Master Mode

9.2.2 SPI Master Mode

When SPI_CR0[SPIMS] = 1, SPI operates in master mode, which provides SCLK signal for SPI bus. When the data is written to SPI_DR, it is firstly written to the transmit buffer and SPI_CR1[TXBMT] is cleared to “0”. If the shift register is empty, the data in the transmit buffer will be transferred to the shift register for the transmission. The master SPI begins its transmission by driving the MSB of shift register on its MOSI pin. After the transmission is completed, SPI_CR1[SPIIF] and SPI_CR1[TXBMT] are set to “1”. While the SPI master transfers data to a slave on the MOSI line, the addressed SPI slave simultaneously transfers data in the shift register to the SPI master on the MISO line in a full-duplex operation. Therefore, SPI_CR1[SPIIF] flag serves as both a transmit-complete flag and a receive-data-ready flag, and the data in the shift register is that received by MISO, which is transferred to the receive buffer. The data from SPI_DR is that of the receive buffer. If the data is written to SPI_DR when SPI_CR1[TXBMT] is “0”, the write conflict flag bit SPI_CR1[WCOL] will be set to “1” and the data in the transmit buffer keeps unchanged.

9.2.2.1 Master Mode Configuration

1. Configure SPI_CR1[NSSMID] to set the SPI operating mode;
2. Configure SPI_CR0[CPOL] to set the clock polarity;
3. Configure SPI_CR0[CPHA] to set the clock phase;
4. Set SPI_CR0[SPIMS] to “1” to select master mode;
5. Configure SPI_CLK to set the SCLK rate;
6. Set SPI_CR1[SPIEN] to “1” to enable SPI;
7. Write the data to SPI_DR. SPI transmits data for each write;
8. After SPI_CR1[SPIIF] is set to “1”, SPI_DR is read to receive the data.

9.2.3 SPI Slave Mode

When SPI_CR0[SPIMS] = 0, SPI operates in slave mode. In this mode, SCLK signal is sent by the master SPI. The data is shifted from MOSI pin and shifted out from MISO pin. If no SCLK signal is input, shift register of the slave is in the stop state. If the signal of SCLK is input, the shift register of slave starts to receive and transmit data through MOSI and MISO pins. The slave device cannot initiate transfers. The data sent to the master device is pre-loaded into the shift register by writing to SPI_DR. If the shift register is empty, the data in the transit buffer is transferred into the shift register. After the transmission is completed, SPI_CR1[SPIIF] and SPI_CR1[TXBMT] are set to “1”. The received data that is transferred into receive butter and receive buffer empty flag bit SPI_CR0[RXBMT] is cleared, indicating the new data has not been read. If SPI_CR0[RXBMT] is “0” and there is new data ready to be sent to the receive buffer, SPI_CR1[RXOVRN] is set to “1” and the data in the receive buffer remains unaffected. When data is written to SPI_DR, SPI_CR1[TXBMT] is cleared. If data is written in this case, the write conflict flag bit SPI_CR1[WCOL] is set to “1” and the data in the transmit buffer keeps unchanged.

9.2.3.1 Slave Mode Configuration

1. Configure SPI_CR1[NSSMID] to set the SPI operating mode;
2. Configure SPI_CR0[CPOL] to set the clock polarity;
3. Configure SPI_CR0[CPHA] to set the clock phase;
4. Set SPI_CR0[SPIMS] to “0” to select slave mode;
5. Set SPI_CR1[SPIEN] to 1 to enable SPI;
6. Write data to SPI_DR and wait for the master to transmit the clock signal.

9.2.4 SPI Interrupt Sources

The interrupt sources of SPI include:

- SPI interrupt flag SPI_CR1[SPIIF] is set to “1” each time after the byte is transferred.
- If SPI_DR is written when the data in transmit buffer has not been transferred to the shift register,

the write conflict flag SPI_CR1[WCOL] is set to “1” and the write operation will not be implemented.

- When SPI works as a master in a multi-master system and NSS pin is pulled LOW, the mode error flag SPI_CR1[MODF] is set to “1”. When a mode error occurs, SPI_CR0[SPIMS] and SPI_CR1[SPIEN] are cleared. SPI is forbidden to allow another master to control the bus.
- The receive overflow flag SPI_CR1[RXOVRN] is set to “1” when SPI operates in slave mode and a transmission is completed while the receive buffer still holds unread data from a previous transfer. And the received data will not be transferred to the receive buffer.

9.2.5 Serial Clock Timing

Four combinations of serial clock phase and idle polarity can be selected using the CPHA and CPOL bits in the SPI_CR0 Register. SPI_CR0[CPHA] selects the clock phase (the edge of the SCLK signal used to latch the data in shift register). SPI_CR0[CPOL] selects the polarity. Both master and slave devices must be configured with the same clock phase and polarity. When the clock phase and polarity is configured, SPI shall be disabled (SPI_CR1[SPIEN] = 0). The timing relationships of SCL and SDA in clock phase and polarity combinations are shown in Figure 9-5 and Figure 9-6.

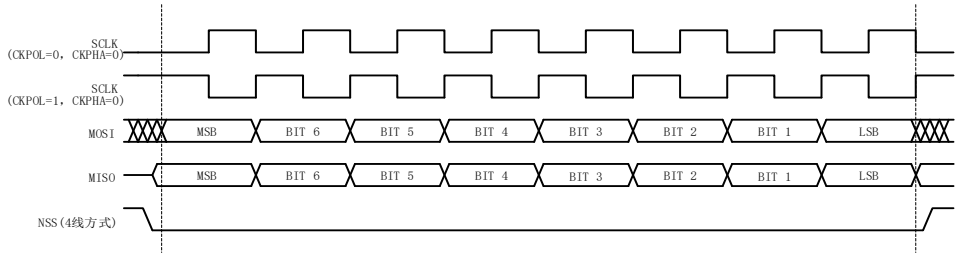


Figure 9-5 SDA/SCL Line Timing Diagram (SPI_CR0[CPHA] = 0)

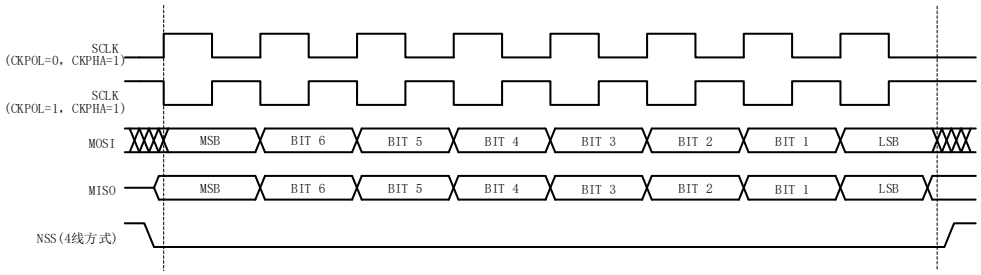


Figure 9-6 SDA/SCL Line Timing Diagram (SPI_CR0[CPHA] = 1)

9.3 SPI Registers

9.3.1 SPI_CR0 (0x4030)

Bit	7	6	5	4	3	2	1	0
Name	SPIBSY	SPIMS	CPHA	CPOL	SLVSEL	NSSIN	SRMT	RXBMT
Type	R	R/W	R/W	R/W	R	R	R	R
Reset	0	0	0	0	0	1	1	1

Bit	Name	Description
[7]	SPIBSY	SPI Busy Flag 0: No data is transferring via SPI transfer. 1: Data is transferring via SPI.
[6]	SPIMS	Master/Slave Mode Selection 0: Slave 1: Master
[5]	CPHA	Clock Phase 0: Data received on leading edge and transmitted on trailing edge of active SCLK. 1: Data transmitted on leading edge and received on trailing edge of active SCLK.
[4]	CPOL	Clock Idle Polarity 0: Low level in idle state. 1: High level in idle state.
[3]	SLVSEL	NSS Select Flag This bit is set to “1” when the filtered signal of NSS is low, indicating that the device is selected as slave. When NSS is high, this bit is cleared to “0”, indicating that the device is not selected as slave. 0: The device is NOT selected as slave. 1: The device is selected as slave.
[2]	NSSIN	NSS real-time signal, unfiltered.
[1]	SRMT	Shift Register Empty Flag (valid only in Slave Mode) 0: Data has been shifted out of the Transit Buffer into the shift register or SCLK changes. 1: There is no data in the shift register or transmit and receive buffer. Note: SPI_CR0[SRMT] = 1 in Master Mode.
[0]	RXBMT	Receive Buffer Empty Flag (valid only in Slave Mode) 0: New data in the receive buffer has not been read. 1: Data has been read and there is no new data in the receive buffer. Note: SPI_CR0[RXBMT] = 1 in Master Mode.

Notes: Clock phase and idle polarity modes SPI_CR0[CPHA:CPOL]:

- 00: Receive data on rising edge, and send data on falling edge. Idle level is low.
- 01: Send data on rising edge, and receive data on falling edge. Idle level is high.
- 10: Send data on rising edge, and receive data on falling edge. Idle level is low.
- 11: Receive data on rising edge, and send data on falling edge. Idle level is high.

9.3.2 SPI_CR1 (0x4031)

Bit	7	6	5	4	3	2	1	0
Name	SPIIF	WCOL	MODF	RXOVRN	NSSMID		TXBMT	SPIEN
Type	R/W0	R/W0	R/W0	R/W0	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	1	0

Bit	Name	Description
[7]	SPIIF	SPI Interrupt Flag This bit is set to “1” by hardware each time after a data frame (8-bit) is transferred. Read: 0: No Interrupt Pending . 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect.
[6]	WCOL	Write Conflict Interrupt Flag When SPI_CR1[TXBMT] is “0”, a write to SPI_DR sets this bit to 1. This bit can be cleared to “0” by software only. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect.
[5]	MODF	Master Mode Fault Interrupt Flag This bit is set to “1” when a master mode conflict is detected (SPI_CR0[NSSIN] = 0, SPI_CR1[SPIMS] = 1 and SPI_CR1[NSSMID] = 01). This bit can be cleared to “0” by software only. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect.
[4]	RXOVRN	Receive Overflow Interrupt Flag (Slave Mode only) This bit is set to “1” by hardware (and generates a SPI interrupt) when the Receive Buffer still holds unread data from a previous transfer and the last bit of the current transfer has been shifted into the SPI shift register. This bit cannot be cleared to “0” automatically by hardware, and can be cleared by software only. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect.
[3:2]	NSSMID	SPI Mode Selection 00: 3-Wire Slave or 3-wire Master Mode. NSS signal is not routed to a port pin. 01: 4-Wire Slave or Multi-Master Mode (Default). NSS pin is configured as an input. 1X: 4-Wire Single-Master Mode. NSS pin is configured as output and outputs SPI_CR1[2] value.
[1]	TXBMT	Transmit Buffer Empty Flag This bit is cleared to “0” when new data is written to the Transit Buffer. It is set to “1” when the data in the Transit Buffer is transferred to the SPI shift register, indicating that it is safe to write a new byte to the transmit buffer. 0: A new byte is written to the transmit buffer. 1: Data in the transmit buffer has been transferred to the shift register.

[0]	SPIEN	SPI Enable 0: Disable 1: Enable

9.3.3 SPI_CLK (0x4032)

Bit	7	6	5	4	3	2	1	0
Name	SPI_CLK							
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]	SPI_CLK	SPI Baud Rate Setting This bit is valid in master mode only, and can be written only when SPI_CR1[SPIEN] = 0. Baud rate = SYSCLK/2/(SPI_CLK + 1) Example: If baud rate = 2400kHz, then SPI_CLK = (24M/2/2400k) - 1 = 4, i.e. 0x04.

9.3.4 SPI_DR (0x4033)

Bit	7	6	5	4	3	2	1	0
Name	SPI_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]	SPI_DR	SPI Data Register SPI_DR Register is used to send and receive SPI data. Read: Receive the data of Receive Buffer Write: Write the data into Transit Buffer and initiate a transfer

10 UART

10.1 UART Operations

10.1.1 UART Mode0

UART mode0, or shifting mode, is used to expand the IO ports. In this mode, TXD pin is configured as clock output, and RXD as data bus. The clock frequency is set at fcpu_clk/12, and the data is sent starting from the least significant bit. UT_CR[REN] decides the data is received or sent. When UT_CR[REN] = 0, the data is sent, and when UT_CR[REN] = 1, the data is received.

Sending Data: Write the data to UT_DR and clear UT_CR[TI]. TXD pin outputs shift pulse and RXD pin sends the data held by UT_DR. The output clock frequency is fcpu_clk/12. After the data is sent, UT_CR[TI] is set to “1”.

Receiving Data: Set UT_CR[REN] to “1” and clear UT_CR[RI]. TXD pin outputs shift pulse and RXD pin receives the data. The shift pulse frequency is fcpu_clk/12. After the data is received, UT_CR[RI] is set to “1” and UT_DR is read to obtain the data.

10.1.2 UART Mode1

UART1 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, 1-bit stop) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Sending Data: Write the data to UT_DR and clear UT_CR[TI]. TXD pin outputs 10-bit data. After the data is sent, UT_CR[TI] is set to “1”.

Receiving Data: Set UT_CR[REN] to “1” to receive the data and clear UT_CR[RI]. The data is received via RXD pin. After the data is received, UT_CR[RB8] and UT_CR[RI] are set to “1” and UT_DR is read to obtain the data.

10.1.3 UART Mode2

UART1 mode2 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, namely 1-bit start, 9-bit data (UT_DR + UT_CR[RB8]/UT_CR[TB8]) and 1-bit stop, to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Sending Data: Write the first 8 bits of the data to UT_DR and the 9^{th} bit to UT_CR[TB8], and clear UT_CR[TI]. TXD pin outputs 11-bit data. After the data is sent, UT_CR[TI] is set to “1”.

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

10.1.4 UART Mode3

The operations are the same as those for UART mode2, but baud rate is the same as that for UART mode1.

10.1.5 UART Interrupt Sources

After UART interrupt is enabled (IE[ES0] = 1), an interrupt is generated when any of the two following flag bits are set to “1”.

Notes: These flag bits can be cleared to “0” by software only.

- After the data (8-bit data for UART mode0 and mode1 and 9-bit data for UART mode2 and mode3) 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.

10.2 UART1 Registers

10.2.1 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: UART Mode0: Shift Register Baud rate = fcpu_clk/12 01: UART Mode1: 8-bit UART Baud rate = fcpu_clk / (16 / (1 + UT_BAUD[BAUD_SEL])) / (UT_BAUD + 1) 10: UART Mode2: 9-bit UART Baud rate = fcpu_clk/ (32 - 16* UT_BAUD[BAUD_SEL]) 11: UART Mode3: 9-bit UART Baud rate = fcpu_clk / (16 / (1 + UT_BAUD[BAUD_SEL])) / (UT_BAUD + 1)
[5]	SM2	Single-device Communication or Multi-device Communication Selection 0: Single-device Communication 1: Multi-device Communication
[4]	REN	UART Receive Enable 0: Disable 1: Enable
[3]	TB8	Bit9 of the sent data in UART mode2 and UART mode3. This bit can be cleared by hardware as required.
[2]	RB8	Bit9 of the received data in UART mode2 and UART mode3. If the bit SM2 is set as “0”, it serves as the STOP bit. It does not work in UART mode0.
[1]	TI	Data Sending Completed Interrupt Flag. This bit is set to “1” after the data is sent, and can be cleared to “0” by software only.
[0]	RI	Data Receiving Completed Interrupt Flag. This bit is set to “1” after the data is received, and can be cleared to “0” by software only.

10.2.2 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	Sent/ Receive Data Read: Data received Write: Data to be sent Note: The UART1 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.

10.2.3 UT_BAUD (0x9A,0x9B)

UT_BAUDH(0x9B)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	BAUD_SEL	RSV			BAUD[11:8]			
Type	R/W	-	-	-	R/W	R/W	R/W	R/W
Reset	0	-	-	-	0	0	0	0
UT_BAUDL(0x9A)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	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
[14:12]	RSV	Reserved
[11:0]	BAUD	Baud rate setting in UART mode1 and UART mode3

10.3 UART2 Registers

10.3.1 UT2_CR (0x8A)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UT2MOD		UT2SM2	UT2REN	UT2TB8	UT2RB8	UT2TI	UT2RI
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W0	R/W0
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[7:6]	UT2MOD	Mode Selection 00: UART Mode0——Shift Register Baud rate = fcpu_clk/12 01: UART Mode1: 8-bit UART Baud rate = fcpu_clk / (16 / (1 + UT_BAUD[BAUD_SEL])) / (UT_BAUD + 1) 10: UART Mode2: 9-bit UART Baud rate = fcpu_clk/ (32 - 16* UT_BAUD[BAUD_SEL]) 11: UART Mode3: 9-bit UART Baud rate = fcpu_clk / (16 / (1 + UT_BAUD[BAUD_SEL])) / (UT_BAUD + 1)
[5]	UT2SM2	Single-device Communication or Multi-device Communication Selection 0: Single-device Communication 1: Multi-device Communication
[4]	UT2REN	UART Receive Enable 0: Disable 1: Enable
[3]	UT2TB8	Bit9 of the sent data in UART mode2 and UART mode3. This bit can be cleared by hardware as required.
[2]	UT2RB8	Bit9 of the received data in UART mode2 and UART mode3. If the bit SM2 is set as “0”, it serves as the STOP bit. It does not work in UART mode0.
[1]	UT2TI	Data Sending Completed Interrupt Flag. This bit is set to “1” after the data is sent, and can be cleared to “0” by software only.
[0]	UT2RI	Data Receiving Completed Interrupt Flag. This bit is set to “1” after the data is received, and can be cleared to “0” by software only.

Note: Before UT2TI or UT2RI is cleared, it is recommended that you write “1” to the flag bit that is not required to be cleared, in order to prevent the flag bit is cleared by software.

10.3.2 UT2_DR (0x89)

Bit	7	6	5	4	3	2	1	0
Name	UT2_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]	UT2_DR	Sent/ Receive Data Read: Data received Write: Data to be sent Note: The UART2 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.

10.3.3 UT2_BAUD (0x4042,0x4043)

UT2_BAUDH(0x4042)								
Bit	15	14	13	12	11	10	9	8
Name	BAUD2_SEL	UART2CH	UART2IEN	RSV	UT2_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
UT2_BAUDL(0x4043)								
Bit	7	6	5	4	3	2	1	0
Name	UT2_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]	BAUD2_SEL	UART2 Frequency Multiplier Enable 0: Disable 1: Enable
[14]	UART2CH	UART2 Function Switching 0: Disable. P3.6 serves as UART2 RXD, and P3.7 as UART2 TXD. 1: Enable. P0.1 serves as UART2 RXD, and P0.0 as UART2 TXD.
[13]	UART2IEN	UART2 Interrupt Enable 0: Disable 1: Enable
[12]	RSV	Reserved
[11:0]	UT2_BAUD	Baud rate setting in UART mode1 and UART mode3

11 MDU

11.1 MDU Introduction

MDU module is a built-in computing co-processor that assists the CPU in processing complex operations efficiently. It supports multiplication, division, trigonometric operation, LPF operation and PID operation. MDU module can be invoked in different interrupt services and master programs, and the results are independent from each other.

11.2 MDU Features

The MDU module has the following features:

- Support invocation with nested interrupt
- Hardware acceleration to reduce CPU load
- Support the following modes:
  - 16-bit signed multiplication
  - 16-bit signed multiplication (result shifted left by one-bit)
  - 16-bit unsigned multiplication
  - 32-bit/16-bit unsigned division
  - Low-pass filter (LPF)
  - Coordinate transformation (SIN/COS)
  - Arctangent (ATAN)

11.3 MDU Instructions

11.3.1 MDU Operations

MDU is operated as follows.

1. Set MDU_CR[MDURUN] as “1”;
2. Configure MDU_MD register to select computing mode of the MDU module;
3. Write the data to MDU_A, MDU_B, MDU_C and MDU_D. MDU module starts operation after it detects data is written to MDU_C[7:0];
4. Wait for MDU_CR[MDUBUSY] to be cleared to “0”;
5. Set MDU_CR[MDUDONE] to “1” after MDU module completes operation.

Notes:

- To ensure the application of multi-interrupt, set MDU_CR[MDURUN] to “1” before MDU module starts operation and set MDU_CR[MDUDONE] to “1” after module completes operation.
- Write data to MDU_C[7:0] after other data and operation mode are already written.

11.3.2 16-bit Signed Multiplication with the Result Shifted Left by One Bit

When MDU_MD[2:0] = 000, MDU module works in the 16-bit signed multiplication mode with the result shifted left by one bit. As shown in Table 11-1, after 16-bit signed data is written to MDU_A and MDU_C as multiplicand and multiplier respectively, 32-bit signed data is obtained by the product shifting left by one bit. The high-order 16 bits of the result are held by MDU_A register, and the low-order 16 bits by MDU_B register.

Table 11-1 Register Definitions in 16-bit Signed Multiplication Mode with Result Shifted Left by One Bit

Data Register	Input	Output
MDU_A	Multiplicand	High-order 16 bits of the product
MDU_B	-	Low-order 16 bits of the product
MDU_C	Multiplier	-
MDU_D	-	-

11.3.3 16-bit Signed Multiplication

When MDU_MD[2:0] = 001, MDU module works in the 16-bit signed multiplication mode. As shown in Table 11-2, 31-bit signed data is obtained after 16-bit signed data is written to MDU_A and MDU_C as multiplicand and multiplier respectively. The high-order 16 bits of the result are held by MDU_A register, and the low-order 16 bits by MDU_B register.

Table 11-2 Register Definitions in 16-bit Signed Multiplication Mode

Data Register	Input	Output
MDU_A	Multiplicand	High-order 16 bits of the product
MDU_B	-	Low-order 16 bits of the product
MDU_C	Multiplier	-
MDU_D	-	-

11.3.4 16-bit Unsigned Multiplication

When MDU_MD[2:0] = 010, MDU module works in the 16-bit unsigned multiplication mode. As shown in Table 11-2, 32-bit unsigned data is obtained after 16-bit unsigned data is written to MDU_A and MDU_C as multiplicand and multiplier respectively. The high-order 16 bits of the result are held by MDU_A register, and the low-order 16 bits by MDU_B register.

Table 11-3 Register Definitions in 16-bit Unsigned Multiplication Mode

Data Register	Input	Output
MDU_A	Multiplicand	High-order 16 bits of the product
MDU_B	-	Low-order 16 bits of the product
MDU_C	Multiplier	-
MDU_D	-	-

11.3.5 32-bit/16-bit Unsigned Division

When MDU_MD[2:0] = 011, MDU module works in the 32-bit/16-bit unsigned division mode. As shown in Table 11-4, 32-bit unsigned quotient and 16-bit unsigned remainder is obtained after a 16-bit unsigned divisor is written to MDU_C and a 32-bit unsigned dividend is written to MDU_A (high-order 16 bits of the dividend) and MDU_B (low-order 16 bits of the dividend) respectively. The high-order 16 bits of the quotient are held by MDU_A register, and the low-order 16 bits by MDU_B register. The remainder is held by MDU_C.

Table 11-4 Register Definitions in the Unsigned Division Mode

Data Register	Input	Output
MDU_A	High-order 16 bits of the dividend	High-order 16 bits of the quotient
MDU_B	Low-order 16 bits of the dividend	Low-order 16 bits of the quotient
MDU_C	Divisor	Remainder
MDU_D	-	-

11.3.6 LPF

When MDU_MD[2:0] = 110, MDU module works in LPF mode.

The calculation formula of LPF is:

$$Y_k = Y_{k-1} + K \times (X_k - Y_{k-1})$$

As shown in Table 11-5, Y_k and Y_{k-1} are 32-bit signed data, X_k is 16-bit signed data and K is 8-bit unsigned data. The result Y_k is obtained after high-order 16 bits of Y_{k-1} are written to MDU_A, low-order 16 bits of Y_{k-1} to MDU_B, K to MDU_D and X_k to MDU_C. The high-order 16 bits of Y_k are held by MDU_A register, and the low-order 16 bits by MDU_B register.

Table 11-5 Register Definitions in LPF Mode

Data Register	Input	Output
MDU_A	X_k	-
MDU_B	$Y_{k-1}[31:16]$	$Y_k[31:16]$
MDU_C	$Y_{k-1}[15:0]$	$Y_k[15:0]$
MDU_D	K	-

11.3.7 Coordinate Transformation (SIN/COS)

When MDU_MD[2:0] = 100, MDU module works in Coordinate Transformation mode.

The formula for coordinate transformation is:

$$sin_o = cos_i \times sin\,\theta + sin_i \times cos\,\theta$$
$$cos_o = cos_i \times cos\,\theta + sin_i \times sin\,\theta$$

In particular, when $sin_i = 0$, the coordinate transformation is a sine and cosine calculation with cos_o as the amplitude, calculated as:

$$sin_o = cos_i \times sin\,\theta$$
$$cos_o = cos_i \times cos\,\theta$$

As shown in Table 11-6, cos_i, sin_i, θ, cos_o and sin_o are all 16-bit signed data. cos_i is written to MDU_A, θ to MDU_B and sin_i to MDU_C to calculate cos_o and sin_o. The results cos_o and sin_o are held by MDU_A and MDU_C respectively.

Table 11-6 Register Definitions in the Coordinate Transformation Mode

Data Register	Input	Output
MDU_A	cos_i	cos_o
MDU_B	θ	-
MDU_C	sin_i	sin_o
MDU_D	-	-

11.3.8 Arctangent

When MDU_MD[2:0] = 101, MDU module works in arctangent (ATAN) mode.

ATAN calculates the amplitude and angle of a vector based on sine and cosine inputs. The calculation formula is:

$$U = \sqrt{sin\,\theta^2 + cos\,\theta^2}$$
$$\theta = tan^{-1}\left(\frac{sin\,\theta}{cos\,\theta}\right)$$

As shown in Table 11-7, cos, sin, U and θ are all 16-bit signed data. cos is written to MDU_A and sin to MDU_C to calculate U and θ. The results U and θ are held by MDU_A and MDU_C respectively.

Table 11-7 Register Definitions in ATAN Mode

Data Register	Input	Output
MDU_A	cos	U
MDU_B	-	-
MDU_C	sin	θ
MDU_D	-	-

11.4 MDU Registers

11.4.1 MDU_CR(0xC1)

Bit	7	6	5	4	3	2	1	0
Name	MDUBUSY	MDUDONE	MDURUN	RSV				
Type	R	W	W	-	-	-	-	-
Reset	0	0	0	-	-	-	-	-

Bit	Name	Description
[7]	MDUBUSY	MDU Busy Flag MDU module starts when data is written to MDU_C[7:0]. The bit is set to “1” after MDU completes operations.
[6]	MDUDONE	MDU Ending Flag This bit can be set to “1” after MDU module completes operations. To avoid calculation error, it must be set to “1” when MDU module is used in the main program and interrupt program.
[5]	MDURUN	MDU Starting Flag This bit can be set to “1” before MDU module starts. To avoid calculation error, it must be set to “1” when MDU module is used in the main program and interrupt program.
[4:0]	RSV	Reserved

11.4.2 MDU_MD(0xCA)

Bit	7	6	5	4	3	2	1	0
Name	RSV					MDUMOD2	MDUMOD1	MDUMOD0
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	-	-	-	-	-	0	0	0

Bit	Name	Description
[7:3]	RSV	Reserved
[2:0]	MDUMOD [2:0]	MDU Mode Selection 000: 16-bit Signed Multiplication (the result shifted left by one-bit) 001: 16-bit Signed Multiplication 010: 16-bit Unsigned Multiplication 011: 32-bit/16-bit Unsigned Division 100: Coordinate Transformation (SIN/COS) 101: ATAN 110: LPF 111: Reserved

11.4.3 MDU_A(0xC7,0xC6)

MDU_AH(0xC7)								
Bit	15	14	13	12	11	10	9	8
Name	MDU_A[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
MDU_AL(0xC6)								
Bit	7	6	5	4	3	2	1	0

Name	MDU_A[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]	RSV	Data register A of MDU module. Inputs and outputs of the register in different modes are shown as below. Table 11-8 Inputs and Outputs of Data Register A

MDU_MD[2:0]	Input	Output
000	Multiplicand	High-order 16 bits of the product
001	Multiplicand	High-order 16 bits of the product
010	Multiplicand	High-order 16 bits of the product
011	High-order 16 bits of the dividend	High-order 16 bits of the quotient
100	X	-
101	cos_i	cos_o
110	cos	U

11.4.4 :MDU_B(0xC5,0xC4)

MDU_BH(0xC5)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	MDU_B[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
MDU_BL(0xC4)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	MDU_B[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]	RSV	Data register B of MDU module. Inputs and outputs of the register in different modes are shown as below. Table 11-9 Inputs and Outputs of Data Register B

MDU_MD[2:0]	Input	Output
000	--	Low 16 bits of the dividend
001	--	Low 16 bits of the dividend
010	--	Low 16 bits of the dividend
011	Low-order 16 bits of the dividend	Low 16 bits of the dividend
100	$Y_{k-1}[31:16]$	$Y_k[31:16]$
101	θ	-
110	-	θ

11.4.5 MDU_C(0xC3,0xC2)

MDU_CH(0xC3)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	MDU_C[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
MDU_CL(0xC2)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	MDU_C[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]	RSV	Data register C of MDU module. MDU module starts when data is written to MDU_C[7:0]. Inputs and outputs of the register in different modes are shown as below.

Table 11-10 Inputs and Outputs of Data Register C

MDU_MD[2:0]	Input	Output
000	Multiplier	--
001	Multiplier	--
010	Multiplier	--
011	Divisor	Remainder
100	$Y_{k-1}[15:0]$	$Y_k[15:0]$
101	cos_i	cos_o
110	sin	

11.4.6 MDU_D(0xCB)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	MDU_D[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
[7:0]	MDU_D	Data register D of MDU module. This register is used to input *K* in coordinate transformation mode (MDU_MD[2:0] = 100).

12 PFC

12.1 PFC Operating Instructions

12.1.1 PFC Introduction

Power Factor Correction (PFC) improves power efficiency and power density, optimizes voltage control over the system and reduces electromagnetic compatibility and electromagnetic interference.

PFC module has the following features:

- Full-automatic hardware
- ADC automatic sampling
- Over-current protection and cycle-by-cycle current limiting

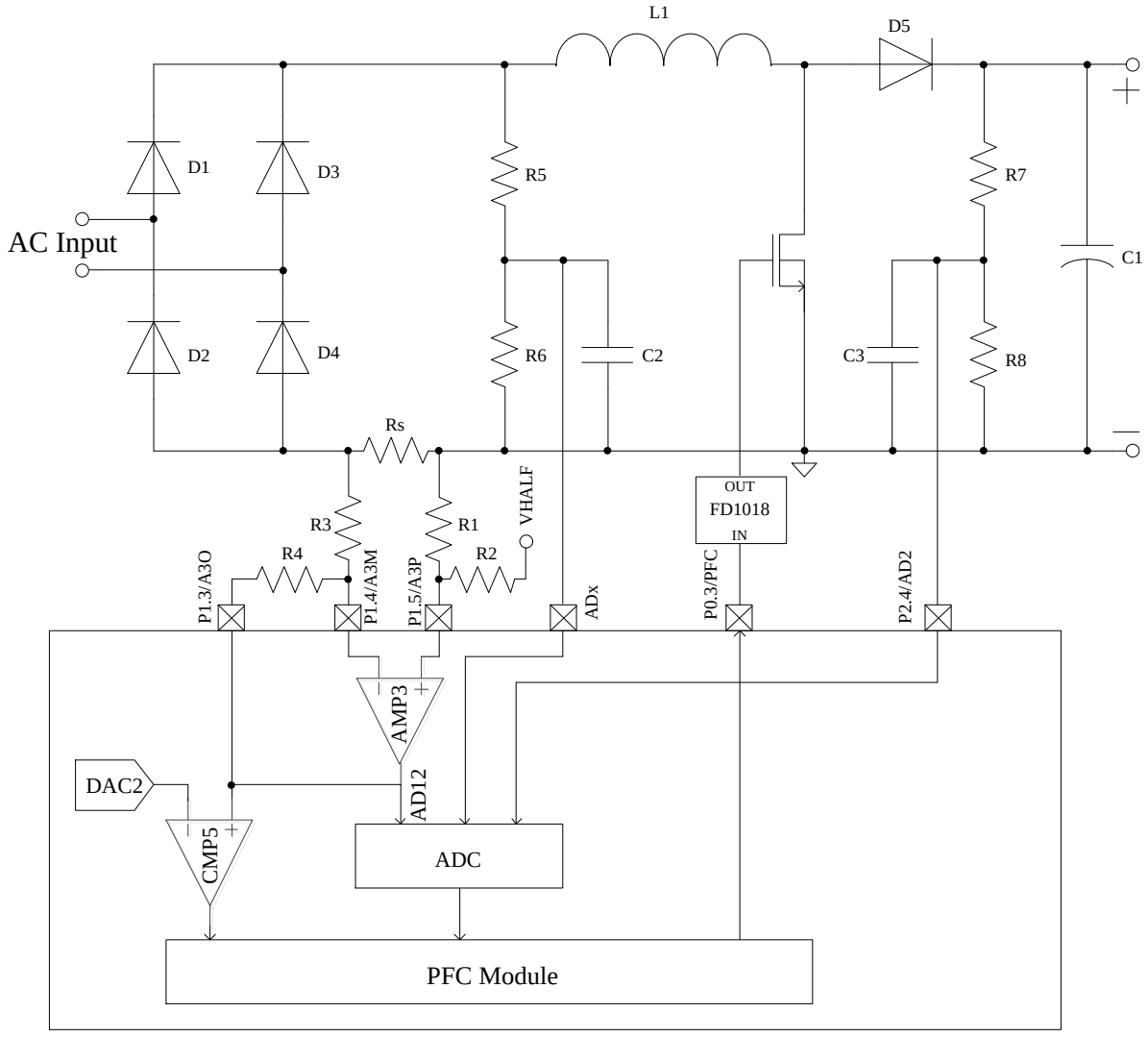


Figure 12-1 Structure Diagram of PFC Module

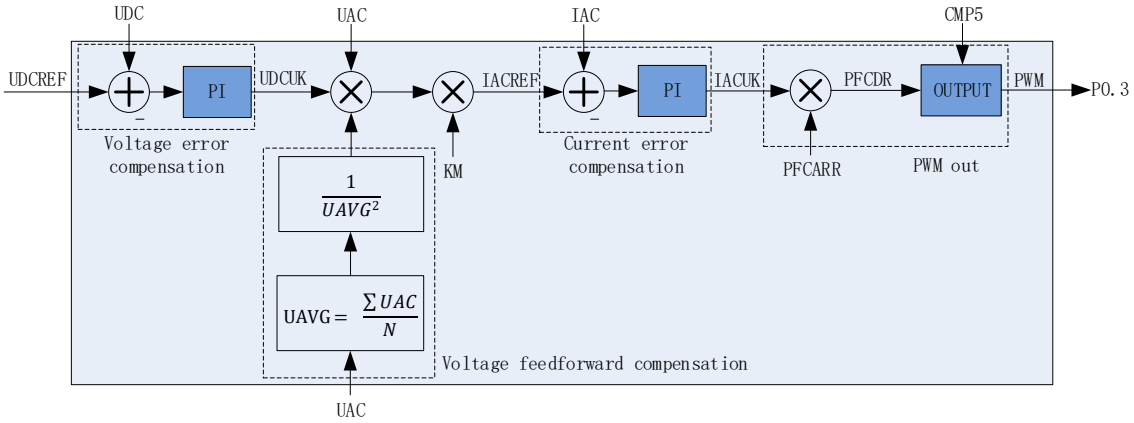


Figure 12-2 Block Diagram of PFC Module

PFC module includes voltage error compensation module, voltage feed-forward compensation module, current error compensation module and PWM output module.

12.1.2 Voltage Error Compensation Module

Voltage error compensation module is outer loop of the PFC module. Its input is the difference between UDCREF (user defined DCV reference) and UDC (sampled DCV by ADC), which is transmitted to PI controller to generate control output UDCUK. Outer loop frequency = Inner loop frequency / PFC_OUTARR = 24M /PFC_ARR / PFC_OUTARR.

12.1.3 Voltage Feed-forward Compensation Module

The voltage feed-forward compensation module is mainly used to maintain a constant output power under unstable input AC voltage.

12.1.4 Calculation of Average Voltage (UAVG)

UAVG is the rectified average voltage of AC voltage UAC. The PFC module calculates UAVG by hardware automatically. This function shall be disabled for some special applications and UAVG will be calculated by software. UAVG is calculated as

$$\mathrm{UAVG} = \frac{\sum UAC}{N}$$

where, UAC is the input AC voltage, N is the sampling number in time period T_s

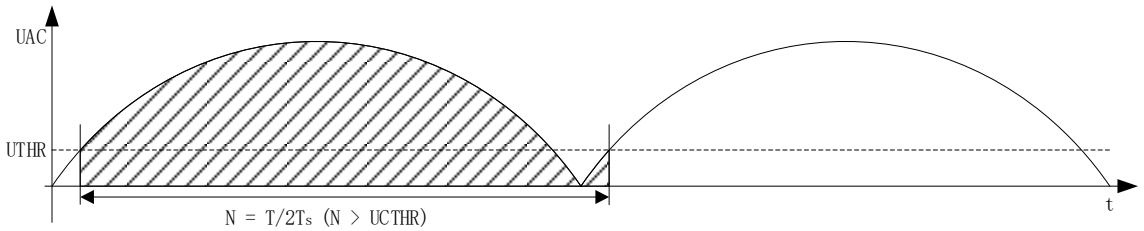


Figure 12-3 Calculation of UAVG

PFC_CR0[UACGCDIS] is set to "0" to start the calculation of UAVG by hardware. The calculation period of UAVG is a half period of power frequency *T*. UTHR is the first and the last value of sampling voltage. Ts is the sampling period of UAC. UCTHR is the minimum value of UAC sampling number. When UAC in this sampling period is bigger than UTHR and the former one is smaller, this sampling period is the start or the end of a calculation period. To decrease the influence of sampling distortion, N should be adequate, larger than UCTHR.

PFC_CR0[UACGCDIS] is set to "1" to start the calculation of UAVG by software. When PFC_CR0[UAVGSW] = 1, the calculation period is the time between the end of last calculation period and this sampling period (PFC_CR0[UAVGSW]). SYS_TICK or other Timer can be used to generate a frequency for the accurate calculation of UAVG.

12.1.5 Current Error Compensation Module

Current error compensation module is inner loop of the PFC module. Its input is the difference between IACREF (current reference calculated by outer loop) and IAC (sampled current by ADC), which is transmitted to PI controller to generate control output IACUK. Inner loop frequency = 24MHz/PFC_ARR.

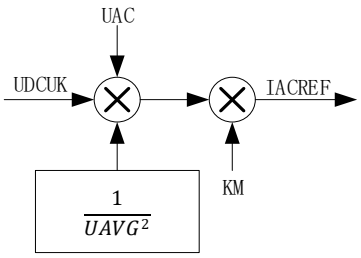


Figure 12-4 Block Diagram of the Calculation of IACREG

As shown above, multiply UDCUK, UAC, the output voltage feed-forward compensation module and constant KM, the result is IACREF.

12.1.6 PWM Output Module

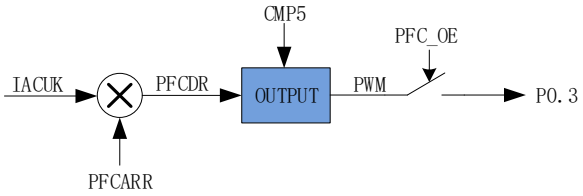


Figure 12-5 Block Diagram of PWM Output

IACUK is the duty cycle of PWM module, duty cycle = 100% × IACUK/32768, and PWM signal is generated by comparator and output to P0.3. As shown above, PFCDR, the value IACUK × PFCARR/32768, is compared with PFC timer to generate PWM signal. When PFCDR >PFCCNTR, PWM module outputs logical 1, and when PFCDR ＜PFCCNTR, PWM module outputs logical 0. If PFC_CR0[PFCOE] is set to "1", P0.3 serves as output of the PWM module.

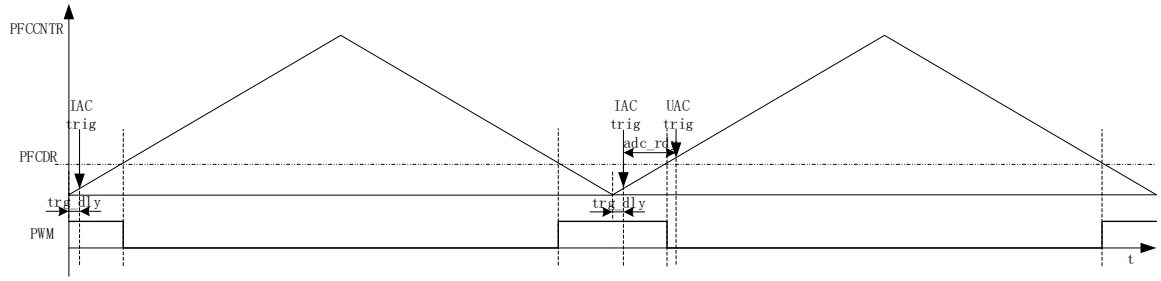


Figure 12-6 Diagram of PWM Output and IAC/UAC Sampling

12.1.7 Over-current Protection and Cycle-by-Cycle Current Limiting

Configuring CMP_CR4[CMP5EN] to "1" enables CMP5 and over-current protection feature. The filter factor of CMP5 is determined by PFC_CR0[CPM5DIV]. When the input of CMP5 is logical 1, over-current protection forces PWM module to output low voltage level. Configuring PFC_CR0[PFCOE] to "0" disables the over-current protection feature.

12.1.8 UAC/IAC/UDC Sampling

12.1.8.1 UDC Sampling

- UDC is sampled by FOC module every carrier period.
- ADC channel 2 is always used.

12.1.8.2 IAC Sampling

- Sample once on overflow point or underflow point of every inner loop period
- The time delay for sampling is configurable by PFC_TRGDLY. For example, if MCU clock runs

at 24MHz(41.67ns) and PFC_TRGDLY = 5, then the sampling is delayed by 41.67*2*5=416ns.

- ADC channel 6 is always used.
- When PFC_CR0[CCHSEL] = 0, a data can be written to PFC_CSO to set IAC offset. Providing the voltage range of ADC is 0~5V and the reference is 2.5V, then PFC_CSO = 32768 × 2.5 / 5V = 16384 (0x4000).

12.1.8.3 UAC Sampling

- Select UAC sampling period by PFC_CR1[UACSAMSEL], once every 1/2/4/8 inner loop periods. UAC is sampled after sampling of IAC.
- ADC channel 5 is used by default. Set UAC_TRIG_CH to select other ADC channels if necessary.
- When PFC_CR0[CCHSEL] = 1, a data can be written to PFC_CSO to set UAC offset. Providing the voltage range of ADC is 0~5V and the reference is 2.5V, then PFC_CSO = 32768 × 2.5 / 5V = 16384 (0x4000).

12.2 PFC Registers

12.2.1 PFC_CR2 (0x409E)

Bit	7	6	5	4	3	2	1	0
Name	UDCPISTA	IACPISTA	RSV					
Type	R/W	R/W	-	-	-	-	-	-
Reset	0	0	-	-	-	-	-	-

Bit	Name	Description
[7]	UDCPISTA	UDC_PI Enable When PFC module is disabled (PFC_CR0[PFCEN] = 0), UDC_PI works as a general PI controller. This bit is set by software to “1”, and will be cleared by hardware at next clock. A write of “0” to this bit has no effect. 0: Disable. 1: Enable.
[6]	IACPISTA	IAC_PI Enable When PFC module is disabled (PFC_CR0[PFCEN] = 0), IAC_PI works as a general PI controller. This bit is set by software to “1”, and will be cleared by hardware at next clock. A write of “0” to this bit has no effect. 0: Disable. 1: Enable.
[5: 0]	RSV	Reserved

12.2.2 PFC_CR0 (0x40E0)

Bit	7	6	5	4	3	2	1	0
Name	UAVGSW	CMP5DIV		UAVGDIS	PFCOA	CCHSEL	PFCOE	PFCEN
Type	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]	UAVGSW	Start UAVG Calculation Start UAVG calculation and update the result to the value of UAVG. This bit is set to “1” by software, and will be cleared by hardware at next clock. A write of “0” to this bit has no effect. 0: UAVG calculation doesn’t start. 1: UAVG calculation starts.
[6:5]	CMP5DIV	Filter Period Selection of CMP5 When input pulse width of CMP5 is less than the set value, it will be considered as noise and the hardware will automatically filter it out. 00: No filtering 01: 4 system clocks 10: 12 system clocks 11: 24 system clocks
[4]	UAVGCDIS	UAVG Calculation by Hardware Enable When this bit is disabled, the hardware does not calculate UAVG at each power frequency period. You need to configure PFC_CR0[UAVGSW] to start UAVG calculation. UAVG calculation starts automatically when internal timer overflows. 0: Enable 1: Disable
[3]	PFCOA	Cycle-by-cycle Current Limiting Feature Enable After CMP5 is enabled, over-current protection feature is enabled by default. When over-current event occurs, PFC module turns off the outputs. After PFCOA is enabled, the hardware automatically restores the output after the over-current protection state is released, that is, cycle-by-cycle current limiting feature. 0: Disable 1: Enable
[2]	CCHSEL	ADC Offset Channel Selection This bit is used to select the ADC offset channel, IAC or UAC. 0: ADC offset by IAC sampling IAC_TRIG_CH 1: ADC offset by UAC sampling UAC_TRIG_CH
[1]	PFCOE	PFC Output Enable With this bit enabled, PWM signal generated by PFC module outputs to P0.3. 0: Disable 1: Enable
[0]	PFCEN	PFC Module Enable 0: Disable 1: Enable

12.2.3 PFC_CR1 (0x40F2)

Bit	7	6	5	4	3	2	1	0
Name	UACSAMSEL		UTHR	UCTHR				
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]	UACSAMSEL	UAC Sampling Period Sample UAC once every x PFC periods. 00: 1 PFC period 01: 2 PFC periods 10: 4 PFC periods 11: 8 PFC periods
[5]	UTHR	UAVG Calculation Period Threshold The sampling value, which is higher or lower than this threshold, is set as the first and the last value of a calculation period. The threshold is based on UAC_BASE. 1: 1/8*UAC_BASE 0: 1/16*UAC_BASE

[4:0]	UCTHR	Minimum Number of UAC Sampling Times The calculation value of UAVG is reasonable under sampling times no less than this value. The minimum sampling times = UCTHR × 32

Note: PFC_CR1 is valid only when PFC module is enabled (PFC_CR0[PFCEN] = 1).

12.2.4 PFC_ADCCH (0x40E1)

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

Bit	Name	Description
[7:4]	IAC_TRIG_CH	ADC Channel Selection for IAC Sampling Choose ADC channel 6 when PFC is enabled, otherwise the operational amplifier is unavailable. Table 12-1 ADC Channel Selection for IAC Sampling

IAC_TRIG_CH	ADC Channel	IAC_TRIG_CH	ADC Channel
0000	Channel 0	0001	Channel 1
0010	Channel 2	0011	Channel 3
0100	Channel 4	0101	Channel 5
0110	Channel 6	0111	Channel 7
1000	Channel 8	1001	Channel 9
1010	Channel 10	1011	Channel 11
1100	Channel 12	1101	Channel 13
1110	Reserved	1111	Reserved

Bit	Name	Description
[3:0]	UAC_TRIG_CH	ADC Channel Selection for UAC Sampling Table 12-2 ADC Channel Selection for UAC Sampling

UAC_TRIG_CH	ADC Channel	UAC_TRIG_CH	ADC Channel
0000	Channel 0	0001	Channel 1
0010	Channel 2	0011	Channel 3
0100	Channel 4	0101	Channel 5
0110	Channel 6	0111	Channel 7
1000	Channel 8	1001	Channel 9
1010	Channel 10	1011	Channel 11
1100	Channel 12	1101	Channel 13
1110	Reserved	1111	Reserved

12.2.5 PFC_CSO (0x40E2, 0x40E3)

PFC_CSOH(0x40E2)								
Bit	15	14	13	12	11	10	9	8
Name	PFC_CSO[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PFC_CSOL(0x40E3)								
Bit	7	6	5	4	3	2	1	0

Name	PFC_CSO[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]	PFC_CSO	IAC/UAC Sampling Reference Configure PFC_CR0[CCHSEL] to write data to PFC_CSO and set IAC/UAC ADC offset. Range: (0, 32767). MSB is always 0. Providing ADC voltage range is 0~5V, the offset is 2.5V, then PFC_CSO = 32768 × 2.5V/5V = 16384 (0x4000).

12.2.6 PFC_ARR (0x40E4, 0x40E5)

PFC_ARRH(0x40E4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				PFC_ARR[11:8]			
Type	-	-	-	-	W	W	W	W
Reset	-	-	-	-	0	0	0	0
PFC_ARRL(0x40E5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PFC_ARR[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:0]	PFC_ARR	Reload Value of PFC Timer. It configures the carrier period and operation mode (center-alignment mode). An overflow event occurs when PFC timer counts from 0 and reaches PFC_ARR, and then it counts down to 0. This register is write-only. Range: (0, 4095)

12.2.7 PFC_UAVG (0x40E4, 0x40E5)

PFC_UAVGH(0x40E4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PFC_UAVG[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
PFC_UAVGL(0x40E5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PFC_UAVG[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]	PFC_UAVG	Calculation value of average UAC in one power frequency Range: (-32768, 32767)

12.2.8 PFC_DR (0x40E6, 0x40E7)

PFC_DRH(0x40E6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				PFC_DR[11:8]			
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
PFC_DRL(0x40E7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PFC_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]	PFC_DR	Comparison Value by PFC to Generate PWM Signal When PFC_DR >PFC_CNTR, PWM module outputs logical 1, and when PFC_DR <PFC_CNTR, PWM module outputs logical 0. When PFC module is enabled, PFC_DR is automatically updated by hardware. Range: (0, 4095)

12.2.9 UDC_REF/UDC_EK (0x40E8, 0x40E9)

UDC_REFH/UDC_EK(0x40E8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_REF/UDC_EK[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_REFL/UDC_EK(0x40E9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_REF/UDC_EK[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]	UDC_REF	This register is the UDC reference when PFC is enabled (PFC_CR0[PFCEN] = 1). This register is EK of UDC_PI when PFC is disabled (PFC_CR0[PFCEN] = 0). Range: (-32768, 32767)

12.2.10 UDC_UK (0x40EA, 0x40EB)

UDC_UKH(0x40EA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_UK[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_UKL(0x40EB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_UK[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]	UDC_UK	Output value UK of UDC_PI controller Range: (-32768, 32767)

12.2.11 UDC_KP (0x40EC, 0x40ED)

UDC_KPH(0x40EC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_KP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_KPL(0x40ED)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_KP[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]	UDC_KP	KP coefficient of UDC_PI controller Range: (0, 32767). MSB is always 0. The data format is Q10.

12.2.12 UDC_KI (0x40EE, 0x40EF)

UDC_KIH(0x40EE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_KI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_KIL(0x40EE)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_KI[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]	UDC_KI	KI coefficient of UDC_PI controller Range: (0, 32767). MSB is always 0. The data format is Q15.

12.2.13 UDC_UKMAX (0x40F0, 0x40F1)

UDC_UKMAXH(0x40F0)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_UKMAX[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_UKMAXL(0x40F1)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_UKMAX[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]	UDC_UKMAX	Maximum output of UDC_PI controller Range (-32768, 32767)

12.2.14 UDC_UKMIN (0x40F2, 0x40F3)

UDC_UKMINH(0x40F2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_UKMIN[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_UKMINL(0x40F3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_UKMIN[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]	UDC_UKMIN	This bit is the minimum output of UDC_PI controller when PFC module is disabled (PFC_CR0[PFCEN] = 0). Range (-32768, 32767) This bit is for PFC_CR1 and PFC_KM registers when PFC module is enabled (PFC_CR0[PFCEN] = 1). In this case, UDC_UKMIN is 0 by default.

12.2.15 PFC_KM (0x40F3)

Bit	7	6	5	4	3	2	1	0
Name	PFC_KM							
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]	PFC_KM	KM coefficient of PFC Range: (0, 255) Note: This bit is valid only when PFC module is enabled (PFC_CR0[PFCEN] = 1).

12.2.16 IAC_REF/IAC_EK (0x40F4, 0x40F5)

IAC_REFH/IAC_EK(0x40F4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_REF/IAC_EK[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_REFL/IAC_EK(0x40F5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_REF/IAC_EK[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]	IAC_REF/ IAC_EK	This register is the IAC reference when PFC is enabled (PFC_CR0[PFCEN] = 1). This register is EK of IAC_PI when PFC is disabled (PFC_CR0[PFCEN] = 0). Range: (-32768, 32767)

12.2.17 IAC_UK (0x40F6, 0x40F7)

IAC_UKH(0x40F6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_UK[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_UKL(0x40F7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_UK[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]	IAC_UK	Output value UK of IAC_PI controller Range: (-32768, 32767)

12.2.18 IAC_KP (0x40F8, 0x40F9)

IAC_KPH(0x40F8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_KP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_KPL(0x40F9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_KP[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]	IAC_KP	KP coefficient of IAC_PI controller Range: (0, 32767). MSB is always 0. The data format is Q10.

12.2.19 IAC_KI (0x40FA, 0x40FB)

IAC_KIH(0x40FA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_KI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_KIL(0x40FB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_KI[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]	IAC_KI	KI coefficient of IAC_PI controller Range: (0, 32767). MSB is always 0. The data format is Q15.

12.2.20 IAC_UKMAX (0x40FC, 0x40FD)

IAC_UKMAXH(0x40FC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_UKMAX[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_UKMAXL(0x40FD)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_UKMAX[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]	IAC_UKMAX	Maximum output of IAC_PI controller Range (-32768, 32767)

12.2.21 IAC_UKMIN (0x40FE, 0x40FF)

IAC_UKMINH(0x40E4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_UKMIN[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_UKMINL(0x40FF)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_UKMIN[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]	IAC_UKMIN	This bit is the minimum output of IACPI controller when PFC module is disabled (PFC_CR0[PFCEN] = 0). Range (-32768, 32767) This bit is for PFC_TRGDLY and PFC_OUTARR registers when PFC module is enabled (PFC_CR0[PFCEN] = 1). In this case, IAC_UKMIN is 0 by default.

12.2.22 PFC_TRGDLY/PFC_OUTARR (0x40FE, 0x40FF)

PFC_OUTARRH(0x40FE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PFC_TRGDLY				PFC_OUTARR[11:8]			
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	0	0	0	0	0
PFC_OUTARRL(0x40FF)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PFC_OUTARR[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]	PFC_TRGDLY	ADC Sampling Delay of IAC/UDC Channel According to MCU clock, 24MHz (41.67ns), if PFC_TRGDLY = 5, the sampling time is delayed by 41.67ns × 2 × 5 = 416.7ns. Range (0, 15) Note: This bit is valid only when PFC module is enabled (PFC_CR0[PFCEN] = 1).
[11:0]	PFC_OUTARR	Period of PFC outer loop This bit configures the period of PFC outer loop, i.e., UDC_PI controller. Period of outer loop = Period of inner loop / PFC_OUTARR = 24M/ PFC_ARR / PFC_OUTARR According to MCU clock, or 24MHz (41.67ns), when PFC_ARR = 150, PFC_OUTARR = 200, the period of inner loop = 24000000 / 2 /PFC_ARR = 80000Hz; the period of outer loop = 80000 /PFC_OUTARR = 400Hz. Range: (0, 2047) Note: This bit is valid only when PFC module is enabled (PFC_CR0[PFCEN] = 1).

13 PI

13.1 PI Introduction

PI regulator is a linear controller, where the output is generated by linear combination of error proportional, integral and differential actions, and then implemented by an actuator. In motor control system, it is used to for speed and position control.

PI algorithm:

$$U(K) = U(k-1) + Kp*(E(k) – E(k-1)) + Ki*E(k) \text{ ----- } (Uk\_min < U(k) < Uk\_max)$$

13.2 PI Features

The PI module has the following features:

- A general-purpose PI regulator
  - Parameter range is configurable
  - Support multiple invocations but not with nested interrupt
  - Produce a 32-bit result PIx_UK
  - Results are read after Busy Flag is reset to “0”.
- Two special-purpose PI regulators, which can be used as general-purpose PI regulators when PFC module is disabled.
  - Data format Q10 is applied to PI_KP by default, and data format Q15 to PI_KI by default.
  - Fast calculation speed. The calculation is completed after 4 clock cycles.

13.3 PI Operations

13.3.1 General-purpose PI Regulators

1. Configuring PI_CR[PISTA] = 1 enables the PI regulator. PI_CR[PIBUSY] is automatically set to “1”. And then, read PI_CR[PIBUSY] bit by software. When this bit is 0, it indicates that the calculation is completed, and calculation result PI_UK is updated;
2. Data format Q10 or Q15 can be selected for PI registers. Note that Q10 is applied to PI_KP by default and Q15 to other registers (including PI_KI).
3. PI_UK and PI_EK1 values default to the previous calculated U_K and E_k. The related values change after PI_EK1 and PI_UK are written.

When PI regulator is invoked repeatedly, relevant parameters shall be saved after each PI operation, and initialized before the next PI operation. Initialization codes are shown as below:

```
PI_EK1 = X;                    //Initialize E(k-1)
PI_UK  = Y1;                   //Initialize high-order 16 bits of U(k-1)
PI_UKS = Y2;                   //Initialize low-order 16 bits of U(k-1)
```

13.3.2 Special-purpose PI Regulators (UDC_PI/IAC_PI)

1. The special-purpose PI regulators are used only when the PFC is disabled.
2. Configuring PFC_CR2[UDCPISTA]/PFC_CR2[IACPISTA] = 1 enables the PI regulator. The calculation is completed after 4 clock cycles and calculation result PI_UK is updated.
3. PI_UK and PI_EK1 values default to the previous calculated U_K and E_k. The related values change after PI_EK1 and PI_UK are written.

13.4 PI Registers

13.4.1 PI_CR (0xF9)

Bit	7	6	5	4	3	2	1	0
Name	T2SS	RSV					PISTA/PIBUSY	RSV
Type	R	-	-	-	-	-	R/W	-
Reset	0	-	-	-	-	-	0	-

Bit	Name	Description
[7]	T2SS	Input Mode Selection of TIM2 Step Motor 0: P1.0 for direction input and P0.7 for pulse input 1: P1.0 for backward pulse input and P0.7 for forward pulse input
[6:2]	RSV	Reserved
[1]	PISTA/ PIBUSY	PI_STA (write only) PI regulator starts. This bit is set by software to “1”, and will be cleared by hardware at next clock. 0: PI regulator does not start. 1: PI regulator starts. PI_BUSY (read only) 0: PI regulator is idle. 1: PI regulator is busy.
[0]	RSV	Reserved

13.4.2 PI_EK (0xEA, 0Xeb)

PI_EKH(0xEB)								
Bit	15	14	13	12	11	10	9	8
Name	PI_EK[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
PI_EKL(0xEA)								
Bit	7	6	5	4	3	2	1	0
Name	PI_EK[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]	PI_EK	Input deviation EK Range (-32768, 32767)

13.4.3 PI_EK1 (0xE2, 0xE3)

PI_EK1H(0xE3)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PI_EK1[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
PI_EK1L(0xE2)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PI_EK1[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]	PI_EK1	Previous input deviation e(k-1) Range (-32768, 32767)

13.4.4 PI_UK (0xEC, 0xED)

PI_UKH(0xED)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PI_UK[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
PI_UKL(0xEC)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PI_UK[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]	PI_UKL	High-order 16 bits of the output result UK The output result UK is 32-bit data within (-1,1) and generally high-order 16 bits of the data are used. Range (-32768, 32767)

13.4.5 PI_UKS (0xE4, 0xE5)

PI_UKSH(0xE5)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PI_UKS[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
PI_UKSL(0xE4)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PI_UKS[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]	PI_UKS	Low-order 16 bits of the output result UK This register stores low-order 16 bits of the output result for multiple invocation. Range (-32768, 32767)

13.4.6 PI_KP (0xEE, 0xEF)

PI_KPH(0xEF)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PI_KP[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
PI_KPL(0xEE)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PI_KP[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]	PI_KP	KP coefficient Range (-32768, 32767)

13.4.7 PI_KI (0xF2, 0xF3)

PI_KIH(0xF3)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PI_KI[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
PI_KIL(0xF2)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PI_KI[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]	PI_KI	KI coefficient Range (-32768,32767)

13.4.8 PI_UKMAX (0xF4, 0xF5)

PI_UKMAXH(0xF5)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PI_UKMAX[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
PI_UKMAXL(0xF4)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PI_UKMAX[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]	PI_UKMAX	Maximum output of UK Range (-32768, 32767)

13.4.9 PI_UKMIN (0xF6, 0xF7)

PI_UKMINH(0xF7)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	PI_UKMIN[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
PI_UKMINL(0xF6)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	PI_UKMIN[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]	PI_UKMIN	Minimum output of UK Range (-32768, 32767)

13.4.10 PFC_CR2 (0x409E)

Bit	7	6	5	4	3	2	1	0
Name	UDCPISTA	IACPISTA	RSV					
Type	W	W	-	-	-	-	-	-
Reset	0	0	-	-	-	-	-	-

Bit	Name	Description
[7]	UDCPISTA	UDC_PI Regulator Enable UDC_PI regulator can work as a general-purpose PI regulator when the PFC is disabled (PFC_CR0[PFCEN] = 0). This bit is set by software to “1”, and will be cleared by hardware at next clock. A write of “0” to this bit has no effect. 0: Disable 1: Enable
[6]	IACPISTA	IAC_PI Regulator Enable IAC_PI regulator can work as a general-purpose PI regulator when the PFC is disabled (PFC_CR0[PFCEN] = 0). This bit is set by software to “1”, and will be cleared by hardware at next clock. A write of “0” to this bit has no effect. 0: Disable 1: Enable
[5: 0]	RSV	Reserved

13.4.11 UDC_EK (0x40E8, 0x40E9)

UDC_EKH(0x40E8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_EK[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_EKL(0x40E9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**

Name	UDC_EK[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]	UDC_EK	EK of the UDC_PI regulator when PFC is disabled (PFC_CR0[PFCEN] = 0). Range (-32768, 32767)

13.4.12 UDC_UK (0x40EA, 0x40EB)

UDC_UKH(0xEA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_UK[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
UDC_UKL(0xEB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_UK[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]	UDC_UK	The output UK of UDC_PI regulator Range (-32768, 32767)

13.4.13 UDC_KP (0x40EC, 0x40ED)

UDC_KPH(0x40EC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_KP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_KPL(0x40ED)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_KP[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]	UDC_KP	KP coefficient of UDC_PI regulator Range (0, 32767); MSB is always 0; Q10 format

13.4.14 UDC_KI (0x40EE, 0x40EF)

UDC_KIH(0x40EE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_KI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_KIL(0x40EE)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**

Name	UDC_KI[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]	UDC_KI	KI coefficient of UDC_PI regulator Range (0, 32767); MSB is always 0; Q15 format

13.4.15 UDC_UKMAX (0x40F0, 0x40F1)

UDC_UKMAXH(0x40F0)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_UKMAX[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
UDC_UKMAXL(0x40F1)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_UKMAX[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]	UDC_UKMAX	Maximum output of UDC_PI regulator Range (-32768, 32767)

13.4.16 UDC_UKMIN (0x40F2, 0x40F3)

UDC_UKMINH(0x40F2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	UDC_UKMIN[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
UDC_UKMINL(0x40F3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	UDC_UKMIN[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]	UDC_UKMIN	Minimum output of UDC_PI regulator when the PFC is disabled (PFC_CR0[PFCEN] = 0). Range (-32768, 32767) This register is fixed to “0” and used for PFC_CR1 and PFC_KM when the PFC is enabled (PFC_CR0[PFCEN] = 1).

13.4.17 IAC_EK (0x40F4, 0x40F5)

IAC_EKH(0x40F4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_EK[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
IAC_EKL(0x40F5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_EK[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]	IAC_EK	EK of the IAC_PI regulator when PFC is disabled (PFC_CR0[PFCEN] = 0). Range (-32768, 32767)

13.4.18 IAC_UK (0x40F6, 0x40F7)

IAC_UKH(0x40F6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_UK[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
IAC_UKL(0x40F7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_UK[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]	IAC_UK	The output UK of IAC_PI regulator Range (-32768, 32767)

13.4.19 IAC_KP (0x40F8, 0x40F9)

IAC_KPH(0x40F8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_KP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_KPL(0x40F9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_KP[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]	IAC_KP	KP coefficient of IAC_PI regulator Range (0, 32767); MSB is always 0; Q10 format

13.4.20 IAC_KI (0x40FA, 0x40FB)

IAC_KIH(0x40FA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_KI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_KIL(0x40FB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_KI[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]	IAC_KI	KI coefficient of IAC_PI regulator Range (0, 32767); MSB is always 0; Q15 format

13.4.21 IAC_UKMAX (0x40FC, 0x40FD)

IAC_UKMAXH(0x40FC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_UKMAX[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_UKMAXL(0x40FD)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_UKMAX[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]	IAC_UKMAX	Maximum output of IAC_PI regulator Range (-32768, 32767)

13.4.22 IAC_UKMIN (0x40FE, 0x40FF)

IAC_UKMINH(0x40FE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	IAC_UKMIN[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
IAC_UKMINL(0x40FF)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IAC_UKMIN[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]	IAC_UKMIN	Minimum output of IAC_PI regulator when the PFC is disabled (PFC_CR0[PFCEN] = 0) Range (-32768, 3767)

14 FOC

14.1 FOC Overview

14.1.1 FOC Introduction

The FOC module is used in sensorless and sensored FOC motor drive applications and SVPWM-based motor control applications. When DRV_CR[FOC_EN] = 0, FOC module is inactivated and FOC clock stops. The relevant FOC registers are forced into the reset state and cannot be written.

The FOC module consists of an angle estimator, a PI controller, a coordinate transform module, a current sampling module and a PWM output module. The angle estimator uses the sampling motor current to estimate the rotor position and implement sensorless FOC-based motor control. MCU can also process the signals from the position sensor to implement sensored FOC-based control.

- Sensorless FOC: Angle for coordinate transformation is obtained by angle estimator, and the motor speed is estimated for speed closed-loop control.
- Sensor-based FOC: FOC module provides the angle input interface. MCU samples position sensor signals and calculates electrical angle of the motor. Software sends the result to FOC module for coordinate transformation.

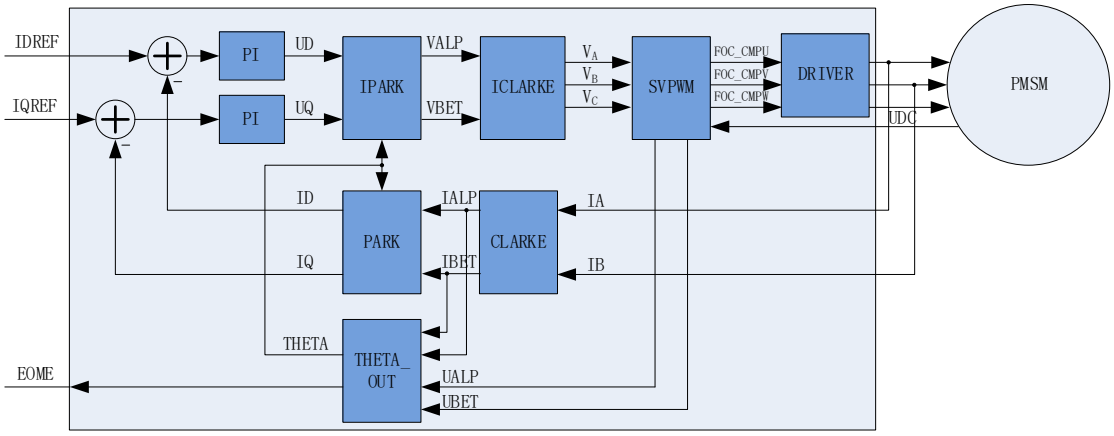


Figure 14-1 FOC Block Diagram

14.1.2 Reference Voltage (VREF) Input

The current loop of FOC module uses the d-axis current reference value FOC_IDREF and the q-axis current reference value FOC_IQREF as the reference, and uses the d-axis current sampling value FOC__ID and the q-axis current sampling value FOC__IQ as the feedback. FOC module outputs real-time estimated motor speed FOC__EOME. MCU can use FOC__EOME as the feedback to build speed loop and send the output of speed loop to FOC_IQREF to implement the speed-current dual closed loop control.

14.1.3 PI Controllers

FOC module uses four PI controllers, which are respectively applied to:

1. Rotor flux control: PI controller of axis D, reference current FOC_IDREF minus feedback current FOC__ID as deviation input, proportional coefficient FOC_DQKP and integral coefficient FOC_DQKI to adjust the performance of PI controller, FOC_DMAX and FOC_DMIN to limit the output, and finally output D axis voltage FOC__UD.
2. Torque control: PI controller of q-axis current, with current reference FOC_IQREF minus feedback current FOC__IQ as deviation input, proportional coefficient FOC_DQKP and the integral coefficient FOC_DQKI for adjustment of PI performance, and FOC_QMAX and FOC_QMIN for limiting of the output amplitude. The output is voltage reference of q-axis FOC__UQ.
3. Angle estimation: PI controller of the estimator, proportional coefficient FOC_EKP and integral coefficient FOC_EKI adjust the performance of PI controller, and finally output estimated angle FOC__ETHETA.
4. PLL estimation: PI controller of PLL estimator, proportional coefficient FOC_PLLKP and integral coefficient FOC_PLLKI adjust the performance of PI controller, and finally output estimated BEMF FOC__EALP and FOC__EBET.

14.1.4 Coordinate Transformation

14.1.4.1 Inverse Park Transformation

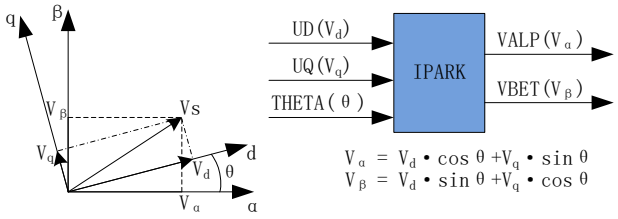


Figure 14-2 Park Transformation

Inverse Park transformation is used to transform two voltage vectors obtained by PI controller, FOC__UD and FOC__UQ, from dq-axis coordinate to αβ-axis coordinate.

14.1.4.2 Inverse Clarke Transformation

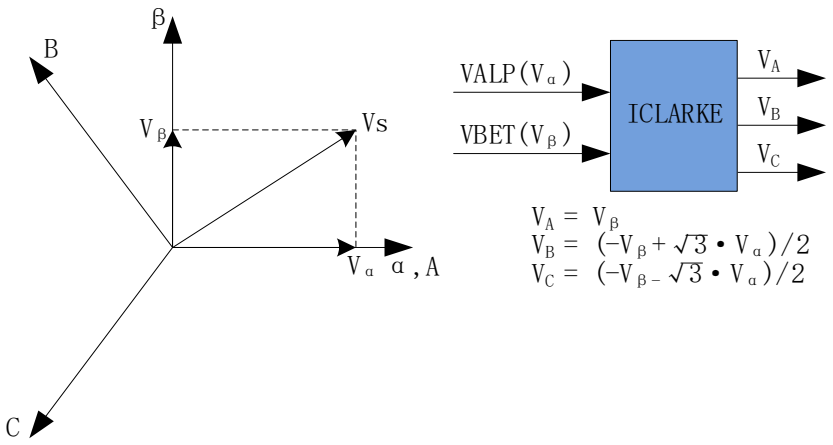


Figure 14-3 Inverse Clarke Transformation

Inverse Clarke transformation is used to transform voltage vector from αβ-axis coordinate to 3-phase stationary coordinate.

14.1.4.3 Clarke Transformation

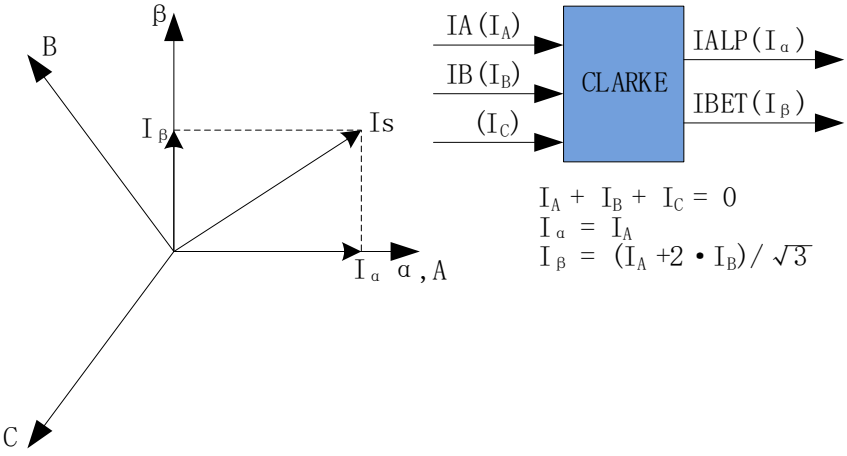


Figure 14-4 Clarke Transform

Clarke transformation is used to transform the sampled current from 3-phase stationary coordinate to αβ-axis coordinate.

14.1.4.4 Park Transformation

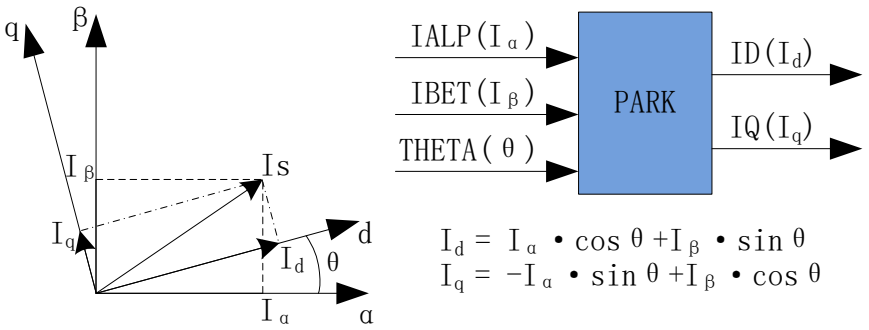


Figure 14-5 Park Transform

Park transformation is used to transform the current vectors, obtained after Clarke transformation, from α/β-axis coordinate to d/q-axis coordinate to get the sampled d/q-axis current FOC__ID and FOC__IQ.

14.1.5 SVPWM

SVPWM algorithm is an important part of FOC. The main idea is to obtain quasi-circular rotating magnetic field by switching the inverter space voltage vectors. This method decreases harmonic components of the phase current, harmonic losses of the motor and torque ripple, and achieves high voltage utilization.

SVPWM generates pulse-width modulation signals for the 3-phase motor voltage control, whose process can be reduced to a few simple equations. Since high side and low side of the inverter cannot be turned on simultaneously, there are two states for a phase, i.e., phase connected to bus voltage (represented by 1) or phase connected to ground (represented by 0). Therefore, voltage vector output of THE inverter has a total of $2^3 = 8$ possible states. $X_C X_B X_A$ represents the voltage vectors, where X_C represents the state of C-phase, X_B represents the state of B-phase and X_A represents the state of A-phase. For example, “100” represents the state that C-phase voltage is connected to bus voltage and A, B-phases are connected to ground. When the states of 3-phase are all 1 or 0, there is no voltage drop between two phases and the state is called inactive state or zero voltage vector. The other six states which have voltage output are active voltage vectors with an adjacent state rotation offset of 60 degrees.

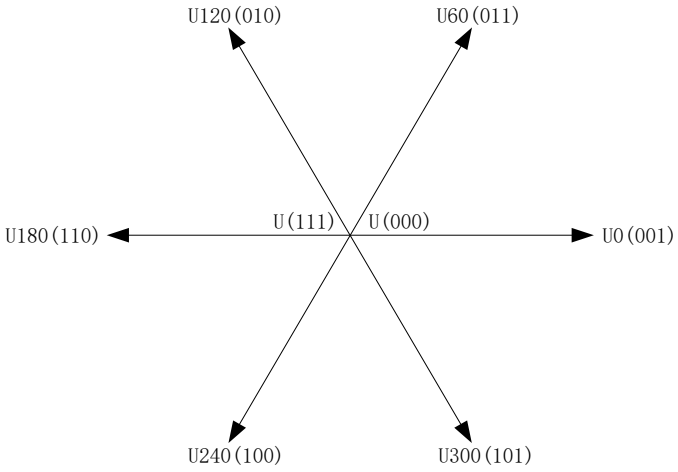


Figure 14-6 SVPWM Voltage Vector

SVPWM uses the sum of two adjacent vectors to generate any voltage vector located in the voltage vector space. As shown in Figure 14-7, U_{OUT} is the desired vector and it is in the sector between U60 and U0. Based on the principle of equal impulse, the effect, U0 applied 2*T1 time and U60 applied 2*T2 time, is equivalent to the U_{OUT}. The rest of time (T0) is applied by zero voltage vector.

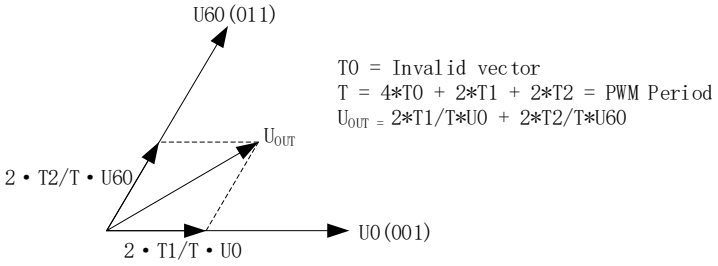


Figure 14-7 SVPWM Voltage Vector Synthesis

Table 14-1 States of SVPWM Inverter

Phase C	Phase B	Phase A	U_{ALP}	U_{BET}	Vector
0	0	0	0	0	000
0	0	1	$2/3*U_{DC}$	0	001
0	1	1	$1/3*U_{DC}$	$1/3*U_{DC}$	011
0	1	0	$-1/3*U_{DC}$	$1/3*U_{DC}$	010
1	1	0	$-2/3*U_{DC}$	0	110
1	0	0	$-1/3*U_{DC}$	$-1/3*U_{DC}$	100
1	0	1	$1/3*U_{DC}$	$-1/3*U_{DC}$	101
1	1	1	0	0	111

14.1.5.1 Continuous SVPWM

In single-shunt current sampling mode, continuous SVPWM is always used. In dual/triple-shunt current sampling mode, FOC_CR2[F5SEG] is set to “0” to select continuous SVPWM as the output mode.

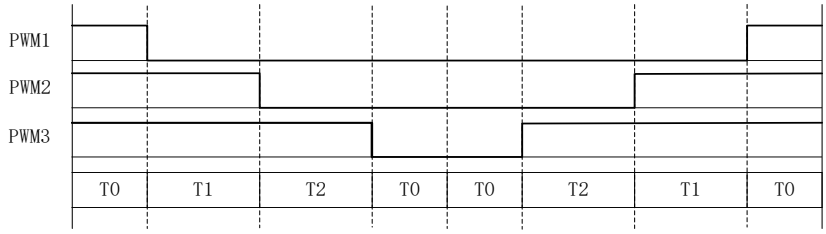


Figure 14-8 Output Level of Continuous SVPWM

14.1.5.2 Discontinuous SVPWM

Discontinuous SVPWM is available in dual/triple-shunt current sampling mode. FOC_CR2[F5SEG] is set to “1” to activate this mode.

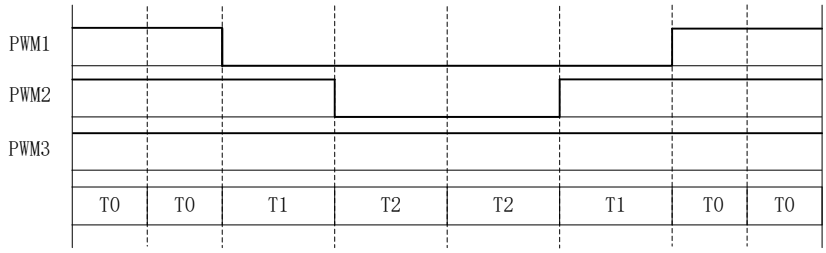


Figure 14-9 Output Level of Discontinuous SVPWM

14.1.6 Overmodulation

Overmodulation is available in single/dual/triple-shunt current sampling mode. Configuring FOC_CR1[OVMDL] = 1 enables overmodulation feature. FOC__UD, FOC__UQ, related limit amplitudes and voltage output are multiplied by 1.15 in this mode.

14.1.7 Deadtime Compensation

Deadtime compensation is available in dual/triple-shunt current sampling mode. The compensation value of deadtime is configured by FOC_TSMIN. This mode improves the quality of phase current at low speed.

14.1.8 Current and Voltage Sampling

In FOC mode, bus voltage and phase current are sampled by hardware automatically. Before the FOC module operates, ADC and operational amplifier shall be enabled and the corresponding control registers be configured. No configuration is required for ADC channel and mode. Single/dual/triple-shunt current sampling mode is selected by setting FOC_CR1[CSM]. In single-shunt current sampling mode, ADC channel

4 is the default sampling channel of the bus current (itrip). In dual-shunt current sampling mode, ADC channel 0 and channel 1 are the default sampling channels of A-phase current (ia) and B-phase current (ib) respectively. In triple-shunt current sampling mode, ADC channel 0, channel 1 and channel 4 are the default sampling channels of ia, ib and C-phase current (ic) respectively. Channel 2 can be selected for bus voltage sampling.

14.1.8.1 Single-shunt Current Sampling Mode

FOC_CR1[CSM] is set to “00” to select the single-shunt current sampling mode. In this mode, FOC module samples itrip (channel 4) twice during DRV timer counting-up operation, and samples bus voltage during DRV timer counting-down operation and after FOC module completes the calculation.

Since deadtime affects the accuracy of current sampling, FOC module samples within T1’ and T2’, which is the applied time of active voltage vector with deadtime removed. FOC_TRGDLY is the register which advance or delay the current sampling time, and this register shall be configured reasonably to ensure sampling is completed within T1’ and T2’. For example, if FOC_TRGDLY = 5, the sampling time is delayed by 5*T = 208ns; and if FOC_TRGDLY = 0xFB(–5), the sampling time is advanced by 5*T = 208ns.

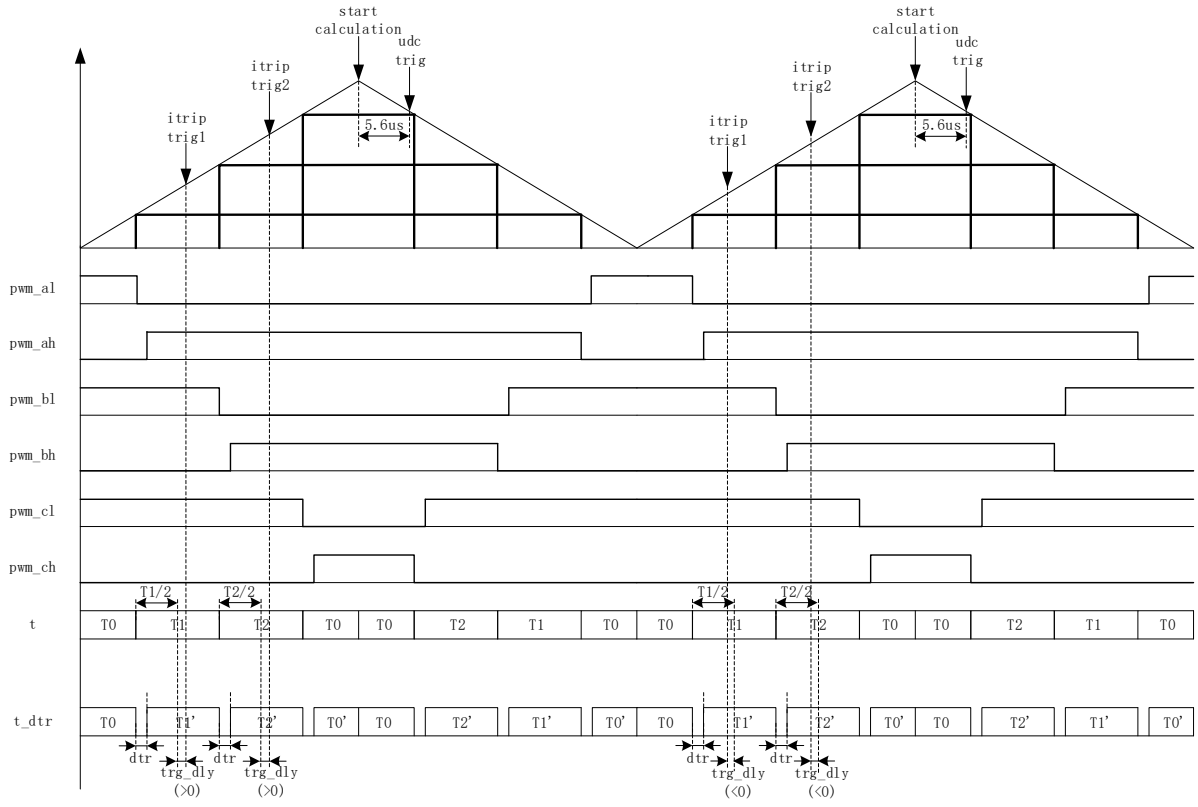


Figure 14-10 Single-shunt Current Sampling Timing

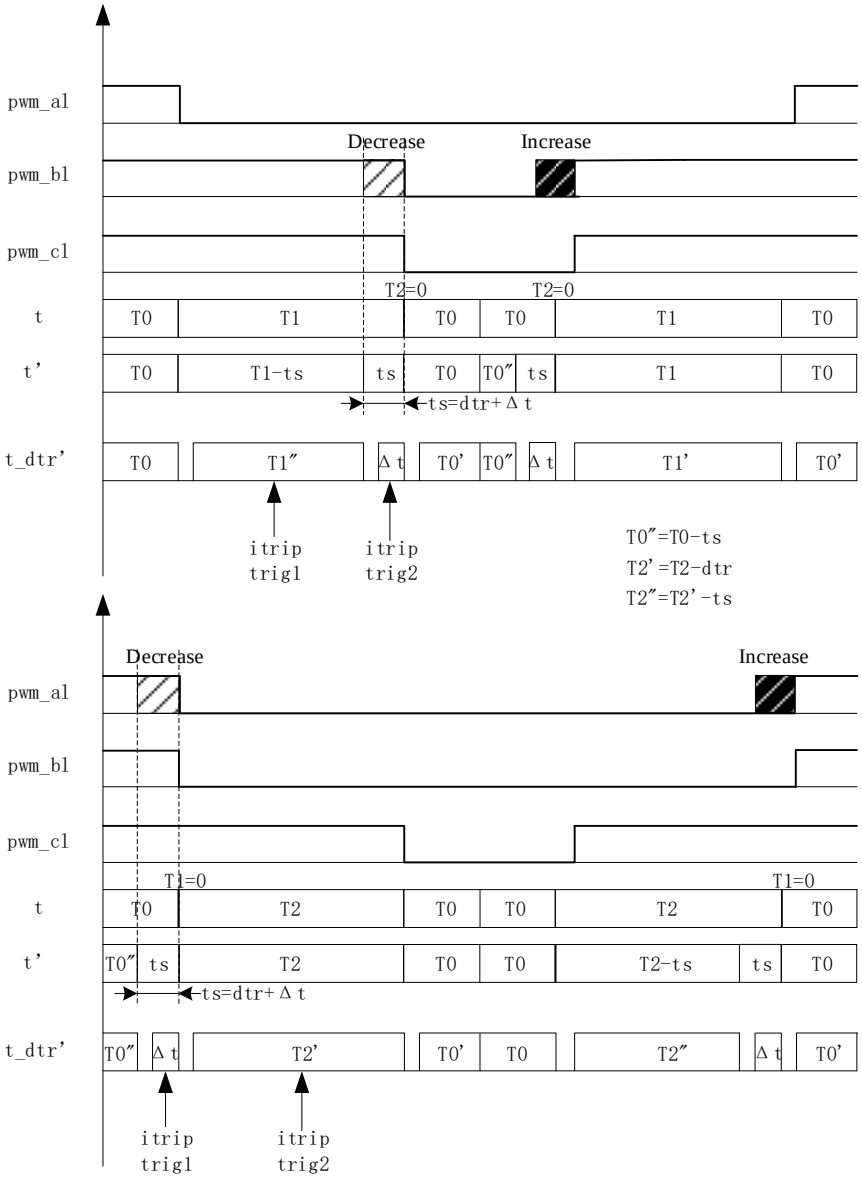


Figure 14-11 Single-shunt Current Sampling Time Compensation

The time of single-shunt current sampling window may be not enough to sample the current in low modulation index and sector switching area. PWM waveform shall be adjusted to ensure the minimum sampling window required in the case. FOC_TSMIN (FOC_TSMIN = minimum sampling window + deadtime) is used to configure the compensation value of deadtime, and FOC module adjusts the PWM waveform automatically.

14.1.8.2 Dual/Triple-shunt Current Sampling Mode

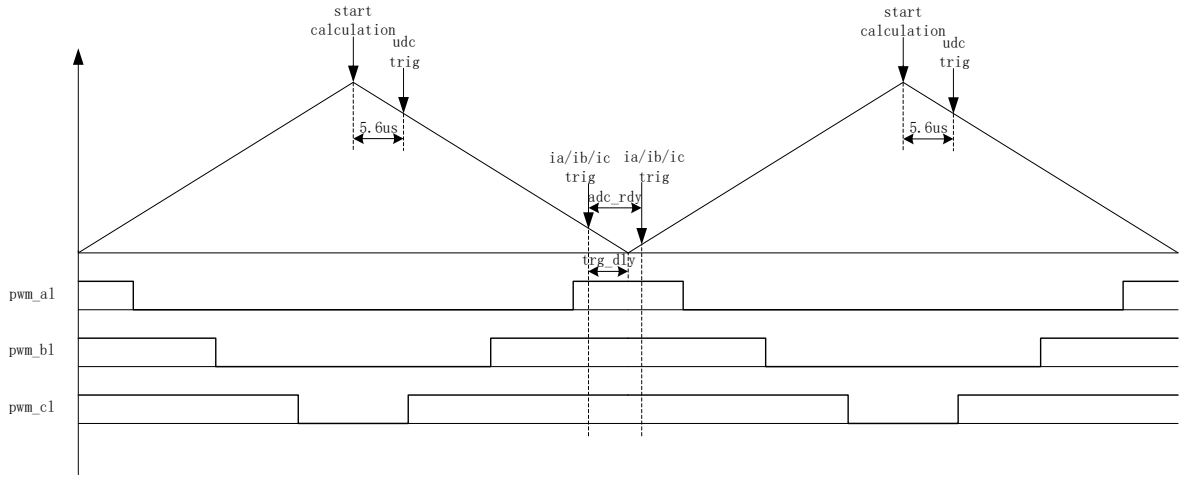


Figure 14-12 Dual/Triple-shunt Sequential Current Sampling Mode

FOC_CR1[CSM] is set to “10/11” and FOC_CR2[DSS] to “0” to select dual/triple-shunt sequential current sampling mode. In triple-shunt sequential current sampling mode, FOC_TRGDLY is used to configure the sampling time of a phase current (ia/ib/ic is determined according to the sector), and other phases are sampled at the end of previous sampling. In dual-shunt sequential current sampling mode, FOC_TRGDLY is used to configure the sampling time of ia, and ib is sampled at the end of ia sampling. TRG_DLY shall be configured reasonably to ensure current sampling time is within zero voltage vector (000). For example, when FOC_TRGDLY = 0xB2 and FOC timer counts down, ia/ib/ic is sampled at 50*T = 2.08μs before an underflow event, and then the other phases of ia/ib/ic are sampled.

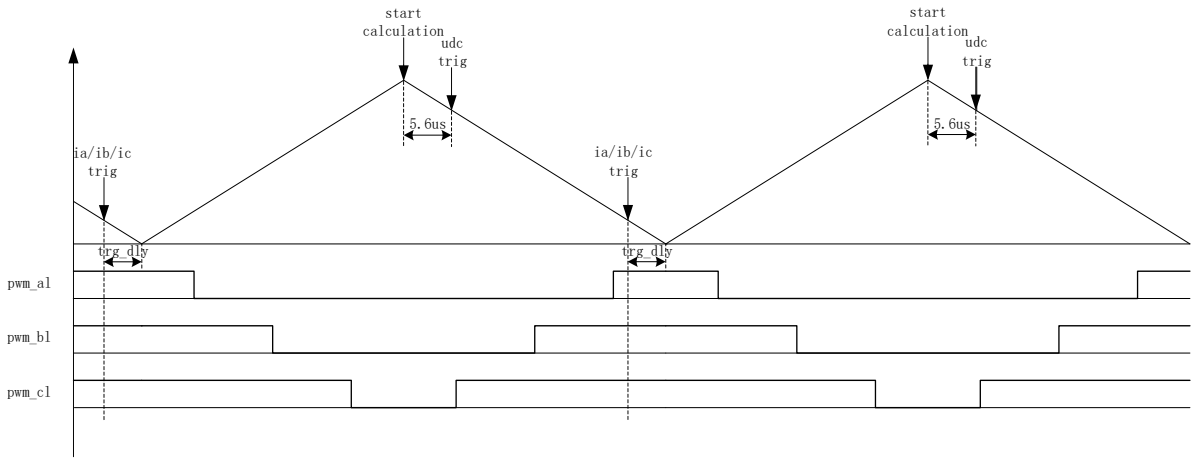


Figure 14-13 Dual/Triple-shunt Alternate Current Sampling Mode

FOC_CR1[CSM] is set to “10/11” and FOC_CR2[DSS] to “1” to select dual/triple-shunt alternating current sampling mode. In this mode, FOC module performs calculation in every PWM cycle. However, only one phase current is sampled at each PWM cycle (ia/ib/ic is determined according to the sector). The first

carrier cycle samples one phase of the ia/ib/ic, and the second carrier cycle samples the current of the other phase, so as to alternately sample the current of two phases in three phases. FOC_TRGDLY is used to configure the sampling time of ia (channel 0), ib (channel 1) and ic (channel 4). TRG_DLY shall be configured reasonably to ensure sampling time for the ia/ib/ic current is within zero voltage vector (000). For example, when FOC_TRGDLY = 0xB2 and FOC timer counts-down, phase current is sampled at 50*T = 2.08μs before an underflow event.

In dual/triple-shunt current sampling mode, bus voltage is sampled when driver timer is down-counting and FOC module completes the calculation.

14.1.8.3 Current Sampling Offset

The current sampling offset voltage shall be added to sample full range of current due to the existence of the positive and negative phase current. When phase current is 0, ADC result is the offset value. ADC result minus this value, 0x4000 default, is the sampling current. Since ADC reference voltage and hardware are nonideal, there is a deviation between the default value and the real value. Therefore, it is necessary to calibrate the offset. The calibration procedure is as follows. When there is no current in three phases, MCU starts to sample the corresponding channel several times, averages all the sampled value, and writes the averaged value to FOC_CSO. Providing ADC sampling range is 0 ~ 5V and the offset is 2.5V, FOC_CSO = 2.5V/5V*32768= 16384 (0x4000).

- When FOC_CHC[CSOC] = 00/11, FOC_CSO is written to modify the offset of itrip and ic.
- When FOC_CHC[CSOC] = 01, FOC_CSO is written to modify the offset of ia.
- When FOC_CHC[CSOC] = 10, FOC_CSO is written to modify the offset of ib.

14.1.9 Angle Mode

Angle module includes angle estimation module, ramping module and estimated angle smooth switching module. The sources of angle are as follows:

- Forced ramping angle
- Forced pulling angle
- Estimated angle of estimator
- Forced angle of estimator

Table 14-2 Sources of Angle

FOC_CR1[RFAE]	FOC_CR1[ANGM]	FOC_CR1[EFAE]	Source
1	X	X	Forced ramping angle
0	0	X	Forced pulling angle
0	1	0	Estimated angle of estimator
0	1	1	omega > FOC_EFREQMIN: Estimated angle of estimator omega < FOC_EFREQMIN: Forced angle of estimator

14.1.9.1 Forced Ramping Angle

Forced ramping angle is controlled by angle register FOC__THETA, speed register FOC__RTHESTEP, acceleration register FOC_RTHEACC and ramping counter FOC_RTHECNT. The formula is:

FOC__RTHESTEP (32 bits) = FOC__RTHESTEP (32 bits) + FOC_RTHEACC (low-order 16 bits)

THETA_OL (16 bits) = THETA_OL (16 bits) + FOC__RTHESTEP (high-order 16 bits)

Where, THETA_OL is an internal variable of the chip. In forced pulling angle mode, THETA_OL is written to FOC__THETA as the used angle. If the software writes a value to FOC__THETA, this value is written to THETA_OL as well.

Forced ramping angle has the highest priority. Configuring FOC_CR1[RFAE] to "1" enables the ramping feature. Ramping module makes a ramping operation in every PWM cycle and the counter is added by 1. When the value of the counter reaches the set value by FOC_RTHECNT, FOC_CR1[RFAE] is cleared by hardware, and then the ramping is completed. Thereafter, according to the value of FOC_CR1[ANGM], the angle comes from estimator (FOC_CR1[ANGM] = 1) or forced pulling angle (FOC_CR1[ANGM] = 0).

14.1.9.2 Forced Pulling Angle

Forced pulling angle is controlled by angle FOC__THETA and speed FOC__RTHESTEP. The formula is: THETA_OL (16-bit) = THETA_OL (16-bit) + FOC__RTHESTEP (high-order 16 bits)

Where, THETA_OL is an internal variable of the chip. In forced pulling angle mode, THETA_OL is written to FOC__THETA as the used angle. If the software writes a value to FOC__THETA, this value is written to THETA_OL as well.

- When FOC_CR1[RFAE] is set to "1" and FOC_CR1[ANGM] to "0", MCU switches to forced pulling angle mode after forced ramping angle mode. The speed is the cumulative result after ramp force angle mode. This mode implements a forced uniform speed control.
- When FOC_CR1[RFAE] is set to "0" and FOC_CR1[ANGM] to "0", the angle is the forced pulling angle and the speed FOC__RTHESTEP is the initial speed written by software. Configuring FOC__RTHESTEP to "0" enables the pre-position feature. The sensor-based FOC is implemented after the motor speed is set with FOC_RTHESTEP. (Principle of Sensor-based FOC: The angle and speed are written to FOC__THETA and FOC__RTHESTEP by software, and FOC module generates an angle in each PWM cycle based on the written values.)

14.1.9.3 Estimator Output Angle

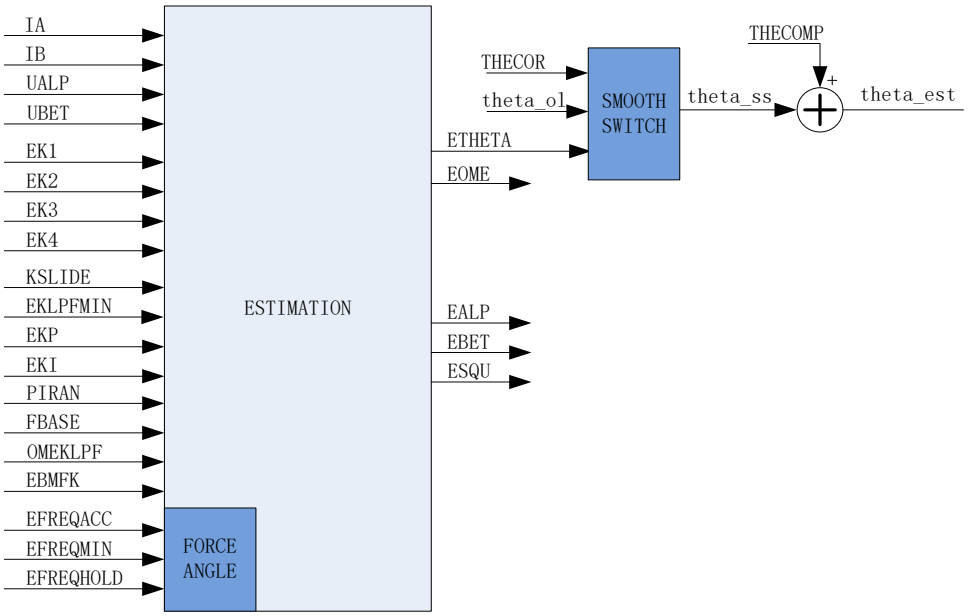


Figure 14-14 Schematic Block Diagram of Estimator

14.1.9.3.1 Estimated Angle of Estimator

The estimator builds the motor model based on the motor parameters and control parameters, and outputs the estimated angle based on the sampled current and voltage. The estimator works in PLL mode or SMO mode by configuring the FOC_CR2[ESEL] bit.

14.1.9.3.2 Forced Angle of Estimator

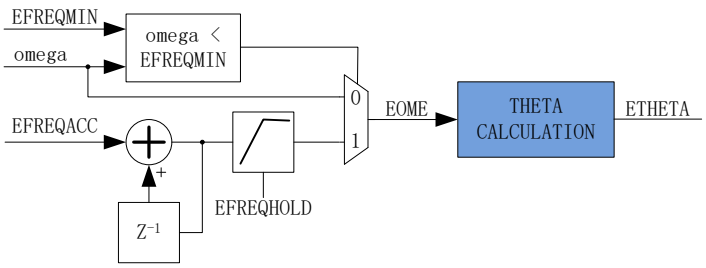


Figure 14-15 Schematic Diagram of Forced Angle of Estimator

This feature is similar to the ramping feature. Due to the low speed at motor starting process, there may be a deviation in angle and speed estimation with the small effective signal, resulting in startup failure. In this case, the estimator outputs the forced angle to ensure the motor start normally.

The forced angle feature of the estimator is enabled when FOC_CR1[RFAE] is set to “0”, FOC_CR1[ANGM] to “1” and FOC_CR1[EFAE] to “1”. As shown in Figure 14-15, the estimator compares

the value of real-time estimated speed (omega) and FOC_EFREQMIN to determine omega or forced speed (FOC__ETHETA) as the used speed (OME). When omega < FOC_EFREQMIN, the forced speed is selected as OME. The forced speed starts with 0 and increases by FOC_EFREQACC in each PWM cycle, with the maximum value FOC_EFREQHOLD. When omega ≥ FOC_EFREQMIN, omega is selected as OME.

Estimated speed of the estimator FOC__EOME is the low-pass filtering result of OME with the coefficient set by FOC_OMEKLPF.

14.1.9.3.3 Angle Smooth Switching

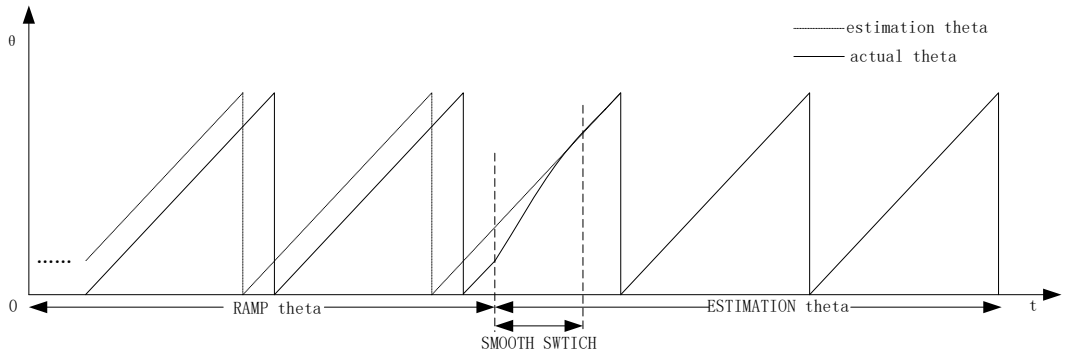


Figure 14-16 Angle Smooth Switching Curve

When FOC_CR1[RFAE] is set to “1” and FOC_CR1[ANGM] to “1”, the motor starts with ramping feature, and it switches to estimator angle mode after the ramping. However, there is usually a deviation between the estimated angle (FOC__ETHETA) and the forced ramping angle (THETA_OL). If the angle is switched from forced ramping angle to estimated angle directly, motor jitter may occur due to such a sudden change. To deal with this problem, a smooth switching is preferred.

After ramping, if the deviation between FOC__ETHETA and THETA_OL is less than or equal to FOC_THECOR, FOC__ETHETA is selected as the output angle. But if the deviation is larger than FOC_THECOR, THETA_OL is modified smoothly with the step of FOC_THECOR at every PWM cycle until it is close to FOC__ETHETA. After the deviation is less than FOC_THECOR, FOC__ETHETA is selected as the output angle.

14.1.9.3.4 Angle Compensation

Angle compensation value FOC_THECOMP is used to compensate for the estimated angle FOC__ETHETA. If FOC_THECOMP is negative (MSB is 1), the lag angle is compensated; if it is positive (MSB is 0), the lead angle is compensated.

14.1.10 Motor Real-time Parameters

MCU monitors the state of motor using the following real time variables provided by FOC module:

- Used angle FOC__THETA
- Estimated angle FOC__ETHETA and estimated speed FOC__EOME
- d-axis voltage FOC__UD and q-axis voltage FOC__UQ
- d-axis current FOC__ID and q-axis current FOC__IQ
- α-axis voltage FOC__VALP, β-axis voltage FOC__VBET
- Bus voltage FOC__UDCFLT
- Phase current FOC__IA, FOC__IB , FOC__IC and maximum phase current FOC__IAMAX, FOC__IBMAX, FOC__ICMAX
- α-axis current (equal to FOC__IA) and β-axis current FOC__IBET
- α-axis BEMF FOC__EALP and β-axis BEMF FOC__EBET
- Motor power FOC__POW

14.1.10.1 Motor Power

FOC module calculates motor power based on the sampling current, modulation index of SVPWM and bus voltage.

14.2 FOC Registers

14.2.1 FOC_CR1 (0x40A0)

Bit	7	6	5	4	3	2	1	0
Name	OVMDL	EFAE	RFAE	ANGM	CSM		SPWMSEL	SVPWMEN
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]	OVMDL	Overmodulation Enable 0: Disable 1: Enable
[6]	EFAE	Forced Angle of Estimator Enable When this feature is enabled, angle mode is determined by the estimator, and it switches to estimated angle mode automatically. 0: Disable 1: Enable
[5]	RFAE	Forced Ramping Angle Enable When this feature is enabled, angle mode is determined by the ramping module. After ramping, it switches to estimated mode or forced pulling mode according to FOC_CR1[ANGM]. FOC_CR1[RFAE] is cleared to "0" by hardware as well. 0: Disable 1: Enable
[4]	ANGM	Angle Mode When FOC_CR1[RFAE] = 0, angle mode is determined by this bit. When FOC_CR1[RFAE] = 1, angle mode is determined by this bit after ramping. 0: Forced Pulling Angle Mode 1: Estimated Angle of Estimator Mode
[3:2]	CSM	Current Sampling Mode 00: Single-shunt Current Sampling 01: Dual-shunt Current Sampling 10: Reserved 11: Triple-shunt Current Sampling
[1]	SPWMSEL	SPWM Output Mode 0: Unipolar 1: Bipolar
[0]	SVPWMEN	SVPWM/SPWM Mode Selection 0: SPWM 1: SVPWM

14.2.2 FOC_CR2 (0x40A1)

Bit	7	6	5	4	3	2	1	0
Name	ESEL	ICLR	F5SEG	DSS	CSOC		UQD	UDD
Type	R/W	R/W1	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]	ESEL	Angle Estimator Mode Selection 0: SMO 1:PLL (phase-locked loop). FOC_KSILDE register is FOC_PLLKP of PI controller, and FOC_KLPFMIN register is FOC_PLLKI of PI controller.

[6]	ICLR	Clear FOC__IAMAX/FOC__IBMAX/FOC__ICMAX to “0” 0: No effect 1: This bit is automatically set to “0” after FOC__IAMAX/ FOC__IBMAX/ FOC__ICMAX are cleared to “0”.
[5]	F5SEG	SVPWM Mode Selection 0: Continuous SVPWM 1: Discontinuous SVPWM (cannot be selected in single-shunt current sampling mode)
[4]	DSS	Dual/Triple-shunt Current Sampling Mode 0: Sequential Sampling Mode, where current values of two phases are sampled in each carrier period. 1: Alternate Sampling Mode. FOC module completes the calculation in every PWM cycle. The current of one phase is sampled in each PWM cycle, and the current of two phases are sampled alternately in two adjacent PWM cycles.
[3:2]	CSOC	Current Sampling Offset Calibration This bit is written to select the offset of FOC_CSO. In single-shunt current sampling mode, “00” or “11” is written to calibrate itrip offset. In dual-shunt current sampling mode, “01” is written to calibrate ia offset and “10” to calibrate ib offset. In triple-shunt current sampling mode, “01” is written to calibrate ia offset, “10” to calibrate ib offset and “00” or “11” to calibrate ic offset. 00: itrip & ic 01: ia 10: ib 11: itrip & ic
[1]	UQD	q-axis PI controller is disabled, where FOC__UQ value is no longer updated by the PI controller. 0: q-axis PI controller is enabled 1: q-axis PI controller is disabled
[0]	UDD	d-axis PI controller is disabled, where the FOC__UD value is no longer updated by the PI controller. 0: d-axis PI controller is enabled 1: d-axis PI controller is disabled

14.2.3 FOC_TSMIN (0x40A2)

Bit	7	6	5	4	3	2	1	0
Name	FOC_TSMIN							
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]	FOC_TSMIN	Single-shunt Current Sampling Mode: minimum window for sampling Dual/triple-shunt Current Sampling Mode: deadtime compensation Range [0, 255] TSMIN = sampling window T_{window} + deadtime T_{DT} Example: Assuming that T_{window} = 1μs, T_{DT} = 1μs, TSMIN = 2μs and carrier period = 62.5μs, then FOC_TSMIN = (1 + 1)/62.5*4096 = 131.

14.2.4 FOC_TGLI (0x40A3)

Bit	7	6	5	4	3	2	1	0
Name	FOC_TGLI							
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]	FOC_TGLI	Narrow Pulse Elimination for High-side of the Bridge This feature is designed for high-voltage applications. The high-side of bridge must be longer than a certain time. After this bit is configured, high-side of the bridge is not turned on when the conducting time is less than this value. Range [0, 255] Example: Assuming that it is required to remove narrow pulses with less than 1μs width, deadtime T_{DT} = 1μs, and carrier period = 62.5μs, then FOC_TGLI = (1 + 1)/62.5*4096 =131.

14.2.5 FOC_TBLO (0x40A4)

Bit	7	6	5	4	3	2	1	0
Name	FOC_TBLO							
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]	FOC_TBLO	Sampling Masking Time in Triple-shunt Current Sampling Mode If low-side of the bridge is turned on for less than FOC_TBLO, the current of this phase is not sampled and obtained through special process. Range [0, 255] Example: Assuming that the phase current is not sampled if the low-side is turned on for less than 1μs, then FOC_TBLO = 1000ns/41.67ns = 24.

14.2.6 FOC_TRGDLY (0x40A5)

Bit	7	6	5	4	3	2	1	0
Name	FOC_TRGDLY							
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]	FOC_TRGDLY	Time Configuration for Current Sampling When FOC_TRGDLY is set to "0", FOC module samples the current as follows. Single-shunt Current Sampling Mode: Midpoint between deadtime and applied time of active voltage vector Dual/Triple-shunt Current Sampling Mode: Midpoint of vector 000 (Driver count value = 0). Range [-128, 127] Single-shunt Current Sampling Mode: If FOC_TRGDLY = 5, it delays by 5*T = 208ns to sample the current, and if FOC_TRGDLY = 0xFB (complement) or FOC_TRGDLY = -5, it advances by 5*T=208ns. Dual-shunt/Triple-shunt Current Sampling Mode: If FOC_TRGDLY = 0x85 (the highest bit, and the remaining 7 bits are absolute values) and Driver timer counts down, it samples the current at 5*T = 208ns before an overflow event occurs. If FOC_TRGDLY = 5 and Driver timer counts up, it samples the current at 5*T = 208ns after an overflow event occurs.

14.2.7 FOC_CSO (0x40A6, 0x40A7)

FOC_CSOH(0x40A6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_CSO[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	0	0	0	0	0
FOC_CSOL(0x40A7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_CSO[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]	FOC_CSO	Current Sampling Offset FOC_CR2[CSOC] is configured to select the current, and FOC_CSO is written to calibrate current sampling offset of itrip in single-shunt current sampling mode, ia, ib in dual-shunt current sampling mode and ia, ib and ic in triple-shunt current sampling mode. Range [0, 32767]. The MSB is always 0 Example: Assuming that the ADC voltage falls within 0V ~ 5V with a reference value of 2.5V, then FOC_CSO = 2.5V/5V*32768 = 16384(0x4000)

14.2.8 FOC__RTHESTEP (0x40A8, 0x40A9)

FOC__RTHESTEPH(0x40A8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__RTHESTEP[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
FOC__RTHESTEPL(0x40A9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__RTHESTEP[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]	FOC__RTHESTEP	Speed of Ramping Module FOC__RTHESTEP is an internal 32-bit variable. MSB is sign bit. High-order 16 bits are written by software. Range [-32768, 32767] FOC__RTHESTEP (32 bits) = FOC__RTHESTEP (32 bits) + FOC_RTHEACC (low-order 16 bits) THETA_OL (16 bits) = THETA_OL (16 bits) + FOC__RTHESTEP (high-order 16 bits)

14.2.9 FOC_RTHEACC (0x40AA, 0x40AB)

FOC_RTHEACCH(0x40AA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_RTHEACC[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
FOC_RTHEACCL(0x40AB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_RTHEACC[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]	FOC_RTHEACC	Ramping Acceleration; FOC_RTHEACC is an internal 32-bit variable. MSB is sign bit. Low-order 16 bits are written by software, and high-order 16 bits are always 0. Range [-32768, 32767] FOC__RTHESTEP (32 bits) = FOC__RTHESTEP (32 bits) + FOC_RTHEACC (low-order 16 bits) THETA_OL (16 bits) = THETA_OL (16 bits) + FOC__RTHESTEP (high-order 16 bits)

14.2.10 FOC_RTHECNT (0x40AC)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_RTHECNT							
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]	FOC_RTHECNT	Max. ramping counts = FOC_RTHECNT*256 When ramping feature is enabled, the ramping angle increases in each PWM cycle. After FOC_RTHECNT*256 times, ramping feature is disabled.

14.2.11 FOC_THECOR (0x40AD) (Shared with BLDC Motor Control)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_THECOR							
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	1

Bit	**Name**	**Description**
[7:0]	FOC_THECOR	Angle Smooth Switching Correction The step value of angle smooth switching after ramping. The format is the same as FOC_THETA. Range [0, 255]

14.2.12 FOC_THECOMP (0x40AE, 0x40AF)

FOC_THECOMPH(0x40AE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_THECOMP[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
FOC_THECOMPL(0x40AF)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_THECOMP[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]	FOC_THECOMP	Angle Compensation Value The output angle FOC__THETA is derived from the estimator estimated angle FOC__ETHETA + compensation value; the format is same with that of FOC__THETA. Range [-32768,32767]

14.2.13 FOC_DMAX (0x40B0, 0x40B1)

FOC_DMAXH(0x40B0)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_DMAX[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
FOC_DMAXL(0x40B1)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_DMAX[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]	FOC_DMAX	Max. output of d-axis PI controller Range [-32768,32767]

14.2.14 FOC_DMIN (0x40B2, 0x40B3)

FOC_DMINH(0x40B2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_DMIN[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
FOC_DMINL(0x40B3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_DMIN[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]	FOC_DMIN	Min. output of d-axis PI controller Range [-32768,32767]

14.2.15 FOC_QMAX (0x40B4, 0x40B5)

FOC_QMAXH(0x40B4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_QMAX[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
FOC_QMAXL(0x40B5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_QMAX[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]	FOC_QMAX	Max. output of q-axis PI controller Range [-32768,32767]

14.2.16 FOC_QMIN (0x40B6, 0x40B7)

FOC_QMINH(0x40B6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_QMIN[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
FOC_QMINL(0x40B7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_QMIN[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]	FOC_QMIN	Min. output of q-axis PI controller Range [-32768,32767]

14.2.17 FOC__UD (0x40B8, 0x40B9)

FOC__UDH(0x40B8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__UD[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
FOC__UDL(0x40B9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__UD[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]	FOC__UD	d-axis voltage calculated by d-axis PI controller Range [-32768,32767]

14.2.18 FOC__UQ (0x40BA, 0x40BB)

FOC__UQH(0x40BA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__UQ[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
FOC__UQL(0x40BB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__UQ[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]	FOC__UQ	q-axis voltage calculated by q-axis PI controller Range [-32768,32767]

14.2.19 FOC__ID (0x40BC, 0x40BD)

FOC__IDH(0x40BC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__ID[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IDL(0x40BD)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__ID[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]	FOC__ID	d-axis current from coordinate transformation. Range [-32768,32767]

14.2.20 FOC__IQ (0x40BE, 0x40BF)

FOC__IQH(0x40BE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IQ[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IQL(0x40BF)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IQ[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]	FOC__IQ	q-axis current from coordinate transformation; Range [-32768,32767]

14.2.21 FOC__IBET (0x40C0, 0x40C1)

FOC__IBETH(0x40C0)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IBETL(0x40C1)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IBET[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]	FOC__IBET	β-axis current from coordinate transformation. Range [-32768,32767]

14.2.22 FOC__VBET (0x40C2, 0x40C3)

FOC__VBETH(0x40C2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__VBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__VBETL(0x40C3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__VBET[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]	FOC__VBET	β-axis output voltage of FOC module Range [-32768,32767]

14.2.23 FOC__VALP (0x40C4, 0x40C5)

FOC__VALPH(0x40C4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__VALP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__VALPL(0x40C5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__VALP[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]	FOC__VALP	α-axis output voltage of FOC module Range [-32768,32767]

14.2.24 FOC_UDCPS (0x40C2, 0x40C3)

FOC_UDCPSH(0x40C2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_UDCPS[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_UDCPSL(0x40C3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_UDCPS[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_UDCPS	d-axis Voltage Compensation Value The result of d-axis PI controller (FOC__UD) added to FOC_UDCPS is transferred to the next module. Range [-32768,32767]

14.2.25 FOC_UQCPS (0x40C4, 0x40C5)

FOC_UQCPSH(0x40C4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_UQCPS[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_UQCPSL(0x40C5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_UQCPS[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_UQCPS	q-axis Voltage Compensation Value The result of q-axis PI controller (FOC__UD) added to FOC_UQCPS is transferred to the next module. Range [-32768,32767]

14.2.26 FOC__IC (0x40C6, 0x40C7)

FOC__ICH(0x40C6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IC[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__ICL(0x40C7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IC[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]	FOC__IC	Sampled Phase-C Current Range [-32768,32767]

14.2.27 FOC__IB (0x40C8, 0x40C9)

FOC__IBH(0x40C8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IB[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IBL(0x40C9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IB[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]	FOC__IB	Sampled Phase-B Current Range [-32768,32767]

14.2.28 FOC__IA (0x40CA, 0x40CB)

FOC__IAH(0x40CA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IA[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IAL(0x40CB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IA[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]	FOC__IA	Sampled Phase-A Current Range [-32768,32767]

14.2.29 FOC__THETA (0x40CC, 0x40CD)

FOC__THETAH(0x40CC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__THETA[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
FOC__THETAL(0x40CD)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__THETA[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]	FOC__THETA	Output Angle of FOC Module Range [-32768, 32767] The bit value -32768 ~ 32767 corresponds to angle range -180°~ 180°. Example: Assuming that FOC__THETA = 8192, the output angle is 8192/32768*180°= 45°.

14.2.30 FOC__ETHETA (0x40CE, 0x40CF)

FOC__ETHETAH(0x40CE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__ETHETA[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
FOC__ETHETAL(0x40CF)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__ETHETA[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]	FOC__ETHETA	Read: Output Angle of Estimator (angle before FOC_THECOMP is applied); the format is same as that of FOC_THETA. Write: Start Angle of Estimator Range [-32768,32767]

14.2.31 FOC__EALP (0x40D0, 0x40D1)

FOC__EALPH(0x40D0)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__EALP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__EALPL(0x40D1)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__EALP[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]	FOC__EALP	α-axis estimated BEMF Range [-32768,32767]

14.2.32 FOC__EBET (0x40D2, 0x40D3)

FOC__EBETH(0x40D2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__EBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__EBETL(0x40D3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**

Name	FOC__EBET[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]	FOC__EBET	β-axis estimated BEMF Range [-32768,32767]

14.2.33 FOC__EOME (0x40D4, 0x40D5)

FOC__EOMEH(0x40D4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__EOME[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
FOC__EOMEL(0x40D5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__EOME[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]	FOC__EOME	Output Speed of Estimator Range [-32768,32767]

14.2.34 FOC__UQEX (0x40D6, 0x40D7)

FOC__UQEXH(0x40D6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__UQEX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__UQEXL(0x40D7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__UQEX[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]	FOC__UQEX	Overflow Value of q-axis PI Controller Equation: FOC__UQ - FOC_QMAX FOC__UQEX is positive when FOC__UQ > FOC_QMAX FOC__UQEX is negative when FOC__UQ < FOC_QMAX FOC__UQEX can be used to realize weak magnetic flux control. Range [-32768,32767]

14.2.35 FOC__POW (0x40D8, 0x40D9)

FOC__POWH(0x40D8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__POW[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__POWL(0x40D9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__POW[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]	FOC__POW	Motor Power Range [-32768,32767]

14.2.36 FOC__IAMAX (0x40DA, 0x40DB)

FOC__IAMAXH(0x40DA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IAMAX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IAMAXL(0x40DB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IAMAX[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]	FOC__IAMAX	Max. Phase-A Current Recorded maximum value of phase-A current; This value may be unreliable unless the motor rotates in a full electrical period. This value will not be cleared to “0” automatically unless FOC_CR2[ICLR] is set to “1”. Range [-32768,32767]

14.2.37 FOC__IBMAX (0x40DC, 0x40DD)

FOC__IBMAXH(0x40DC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IBMAX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IBMAXL(0x40DD)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IBMAX[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]	FOC__IBMAX	Max. Phase-B Current Recorded maximum value of phase-B current; This value may be unreliable unless the motor rotates in a full electrical period. This value will not be cleared to “0” automatically unless FOC_CR2[ICLR] is set to “1”. Range [-32768,32767]

14.2.38 FOC__ICMAX (0x40DE, 0x40DF)

FOC__ICMAXH(0x40DE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__ICMAX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__ICMAXL(0x40DF)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__ICMAX[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]	FOC__ICMAX	Max. Phase-C Current Recorded maximum value of phase-C current; This value may be unreliable unless the motor rotates in a full electrical period. This value will not be cleared to “0” automatically unless FOC_CR2[ICLR] is set to “1”. Range [-32768,32767]

14.2.39 FOC_EKP (0x4074, 0x4075) (Shared with BLDC Motor Control)

FOC_EKPH(0x4074)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EKP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EKPL(0x4075)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EKP[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]	FOC_EKP	KP of PI controller used for estimated angle of the estimator. MSB is always 0. Q12 format. Range [0,32767]

14.2.40 FOC_EKI (0x4076, 0x4077) (Shared with BLDC Motor Control)

FOC_EKIH(0x4076)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EKI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EKIL(0x4077)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EKI[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]	FOC_EKI	KI of PI controller used for estimated angle of the estimator; MSB is always 0; Q15 format. Range [0,32767]

14.2.41 FOC_EBMFK (0x407C, 0x407D) (Shared with BLDC Motor Control)

FOC_EBMFKH (0x407C)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EBMFK[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EBMFKL(0x407D)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EBMFK[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]	FOC_EBMFK	Coefficient used to calculate BEMF low-pass filter coefficient EKLPF Range [0,32767]; Q15 format. EKLPF = EBMFK * OMEGA EBMFK = 2 * PI * fbase * △T

14.2.42 FOC_KSLIDE (0x4078, 0x4079) (Shared with BLDC Motor Control)

FOC_KSLIDEH(0x4078)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_KSLIDE/FOC_PLLKP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_KSLIDEL(0x4079)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_KSLIDE/FOC_PLLKP[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]	FOC_KSLIDE/FOC_PLLKP	FOC_CR2[ESEL] = 0: SMO gain factor; Q15 format FOC_CR2[ESEL] = 1: KP of PI controller on PLL; Q12 format Range [0, 32767]. MSB is always 0.

14.2.43 FOC_EKLPFMIN (0x407A, 0x407B) (Shared with BLDC Motor Control)

FOC_EKLPFMINH(0x407A)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EKLPFMIN/FOC_PLLKPI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EKLPFMINH(0x407B)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EKLPFMIN/FOC_PLLKPI[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]	FOC_EKLPFMIN/FOC_PLLKI	FOC_CR2[ESEL] = 0: The minimum value of BEMF low pass filter factor. EKLPF is forced to be this value when it is lower than this value. Q15 format. FOC_CR2[ESEL] = 1:PI controller KI coefficient on PLL. Q15 format. Range [0, 32767], MSB is always 0.

14.2.44 FOC_OMEKLPF (0x407E, 0x407F)

FOC_OMEKLPFH(0x407E)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_OMEKLPF[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_OMEKLPFL(0x407F)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_OMEKLPF[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]	FOC_OMEKLPF	LPF factor of estimated speed of the estimator. MSB is always 0. Q15 format. Range [0,32767]

14.2.45 FOC_FBASE (0x4080, 0x4081)

FOC_FBASEH(0x4080)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_FBASE[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
FOC_FBASEL(0x4081)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**

Name	FOC_FBASE[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]	FOC_FBASE	Frequency Base of Estimator Range [0, 32767] FOC_FBASE = fbase*Ts*32768 Example: Assuming that fbase = 200Hz, Ts = 62.5μs, then FOC_FBASE = 200*0.0000625*32768 = 409(0x199)

14.2.46 FOC_EFREQACC (0x4082, 0x4083) (Shared with BLDC Motor Control)

FOC_EFREQACCH(0x4082)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EFREQACC[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
FOC_EFREQACCL(0x4083)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EFREQACC[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]	FOC_EFREQACC	Speed Increment of the Forced Angle Mode. FOC_EFREQACC is an internal 24-bit variable and MSB is sign bit. Low-order 16 bits are written by software. Range [0, 65535] Example: Assuming that fbase = 200Hz and pp (Pole_Pairs) = 4, then speed_base = 60*fbase/pp = 3000rpm. If speed increment = 3rpm, then FOC_EFREQACC = 3rpm/speed_base*32768*256 = 8388(0x20C4).

14.2.47 FOC_EFREQMIN (0x4084, 0x4085) (Shared with BLDC Motor Control)

FOC_EFREQMINH(0x4084)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EFREQMIN[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
FOC_EFREQMINL(0x4085)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EFREQMIN[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]	FOC_EFREQMIN	Switch Threshold of the Estimated Angle FOC_EFREQMIN is an internal 24–bit variable, and MSB is sign bit. High-order 16 bits are written by software. With Forced Angle of Estimator Mode enabled, FOC module outputs forced angle when the estimated angle is smaller than the bit value. Range [-32768, 32767]

		Example: Assuming that fbase = 200Hz and pp (Pole_Pairs) = 4, then speed_base = 60*fbase/pp = 3000rpm. Assuming that the min. switching speed = 30rpm, then FOC_EFREQMIN = 30rpm/speed_base*32768 = 327(0x147).

14.2.48 FOC_EFREQHOLD (0x4086, 0x4087) (Shared with BLDC Motor Control)

FOC_EFREQHOLDH(0x4086)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EFREQHOLD[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
FOC_EFREQHOLDL(0x4087)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EFREQHOLD[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]	FOC_EFREQHOLD	Maximum Value of Forced Speed of the Estimator FOC_EFREQHOLD is an internal 24-bit variable, and MSB is sign bit. High-order 16 bits are written by the software. When estimated speed by the estimator is less than FOC_EFREQMIN, the speed is forced to increase to this value. Range [-32768, 32767] Example: Assuming that fbase = 200Hz and pp (Pole_Pairs) = 4, then speed_base = 60*fbase/pp = 3000rpm. If max. forced speed = 60rpm, then FOC_EFREQHOLD = 60rpm/speed_base*32768 = 655(0x028F).

14.2.49 FOC_EK3 (0x4088, 0x4089)

FOC_EK3H(0x4088)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EK3[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EK3L(0x4089)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EK3[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]	FOC_EK3	The 3rd coefficient of the current model in estimator, and MSB is always 0. Q15 format. Range [0,32767]

14.2.50 FOC_EK4 (0x408A, 0x408B)

FOC_EK4H(0x408A)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EK4[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
FOC_EK4L(0x408B)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EK4[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]	FOC_EK4	The 4th coefficient of the current model in estimator. Q15 format. Range [-32768,32767]

14.2.51 FOC_EK1 (0x408C, 0x408D)

FOC_EK1H(0x408C)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EK1[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EK1L(0x408D)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EK1[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]	FOC_EK1	The 1st coefficient of the current model in estimator, and MSB is always 0. Q15 format. Range [0,32767]

14.2.52 FOC_EK2 (0x408E, 0x408F)

FOC_EK2H(0x408E)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_EK2[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EK2L(0x408F)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_EK2[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]	FOC_EK2	The 2nd coefficient of the current model in estimator, and MSB is always 0. Q15 format. Range [0,32767]

14.2.53 FOC_IDREF (0x4090, 0x4091) (Shared with BLDC Motor Control)

FOC_IDREFH(0x4090)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_IDREF[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
FOC_IDREFL(0x4091)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_IDREF[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]	FOC_IDREF	User-defined d-axis Current Range [-32768,32767]

14.2.54 FOC_IQREF (0x4092, 0x4093) (Shared with BLDC Motor Control)

FOC_IQREFH(0x4092)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_IQREF[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
FOC_IQREFL(0x4093)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_IQREF[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]	FOC_IQREF	User-defined q-axis Current Range [-32768,32767]

14.2.55 FOC_DQKP (0x4094, 0x4095) (Shared with BLDC Motor Control)

FOC_DQKPH(0x4094)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_QKP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_DQKPL(0x4095)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_DQKP[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]	FOC_DQKP	KP coefficient of d/q-axis PI controller. MSB is always 0. Q12 format. Range [0, 32767] corresponds to range of Q12 [0, 8].

14.2.56 FOC_DQKI (0x4096, 0x4097) (Shared with BLDC Motor Control)

FOC_DQKIH(0x4096)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_QKI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_DQKIL(0x4097)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_DQKI[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]	FOC_DQKI	KI coefficient of d/q-axis PI controller. MSB is always 0. Q15 format. . Range [0, 32767] corresponds to range of Q15 [0, 1].

14.2.57 FOC__UDCFLT (0x4098, 0x4099)

FOC__UDCFLTH(0x4098)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__UDCFLT[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__UDCFLTL(0x4099)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__UDCFLT[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]	FOC__UDCFLT	Filtered Bus Voltage FOC module samples the bus voltage and filters it to obtain FOC__UDCFLT. ADC channel 2 (external voltage divider) can be selected. Range [0, 32767] Example: The bus voltage is scaled down by 1/6 before feeding into the ADC module, ADC VREF = 5V (namely, the sampling range is 0V~30V) and FOC__UDCFLT = 19661(0x4CCD), then bus voltage = 19661/32768*5V*6 = 18V.

15 SPWM

15.1 SPWM Operating Instructions

15.1.1 SPWM Introduction

SPWM module, as an independent module, is included in FOC module and used for single-phase motor and step motor drive applications. FOC clock stops when SPWM module is not working. Configuring DRV_CR[FOC_EN] = 1 enables SPWM module, otherwise its related registers are in the reset state and cannot be written.

SPWM module contains an angle estimator, a PI controller, a coordinate transform module, a current sampling module and a PWM output module. It also includes a closed current loop, which outputs 8-channel (4-pair) PWM signals to drive the motor based on user-defined ID and IQ. Phase-U/V of PWM signals generate phase-β voltage, and phase-W/X generate phase-α voltage.

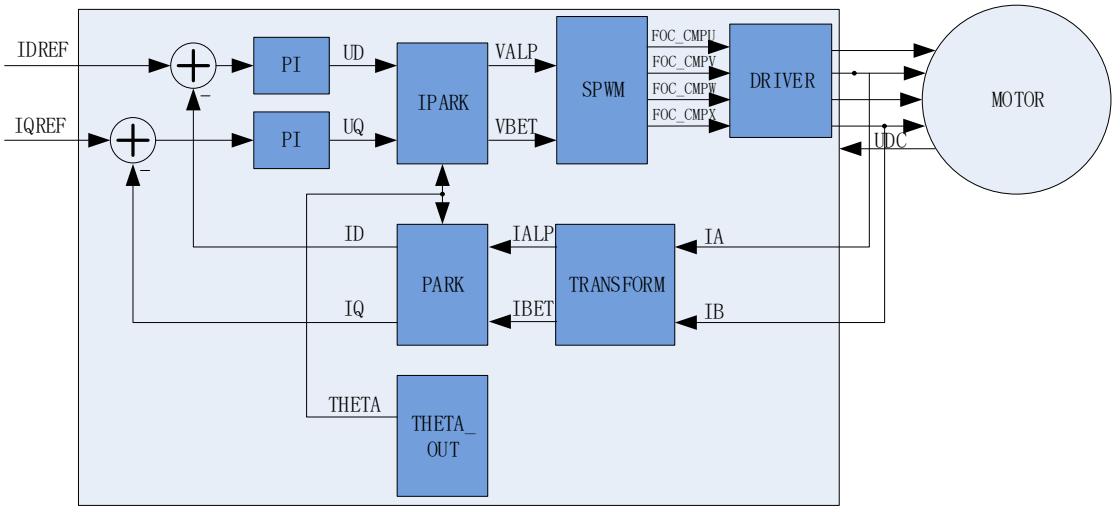


Figure 15-1 Block Diagram of SPWM Module

15.1.2 Reference Voltage (VREF) Input

The current loop of SPWM module uses d-axis current FOC_IDREF and q-axis current FOC_IQREF as the reference inputs to implement closed-loop control. If it is necessary to directly control α/β-axis current, configure FOC__THETA = 0 to make d/q-axis coincide with α/β-axis. The control of d/q-axis current is to control α/β axis current.

15.1.3 PI Controllers

SPWM module integrates two PI controllers, which are respectively applied to:

1. Rotor flux control: PI controller of d-axis, reference current FOC_IDREF minus feedback current FOC__ID as deviation input, proportional coefficient FOC_DQKP and integral coefficient FOC_DQKI to adjust the performance of PI controller, FOC_DMAX and FOC_DMIN to limit the

output, and finally output d-axis voltage FOC__UD.

2. Torque control: PI controller of axis-q current, with current reference FOC_IQREF minus feedback current FOC__IQ as deviation input, proportional coefficient FOC_DQKP and the integral coefficient FOC_DQKI for adjustment of PI performance, and FOC_QMAX and FOC_QMIN for limiting of the output amplitude. The output is voltage reference of q-axis FOC__UQ.

15.1.4 Coordinate Transformation

15.1.4.1 Inverse Park Transformation

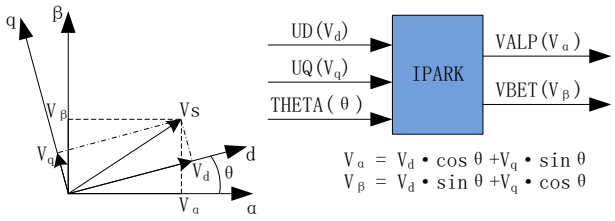


Figure 15-2 Park Transformation

Inverse Park transformation is used to transform two voltage vectors obtained by PI controller, FOC__UD and FOC__UQ, from d/q-axis coordinate to α/β-axis coordinate.

15.1.4.2 Park Transformation

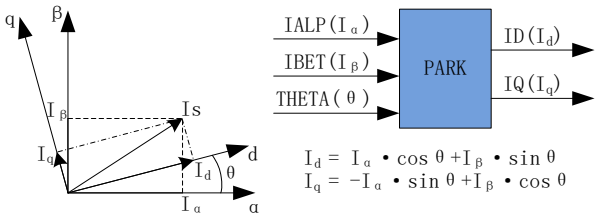


Figure 15-3 Park Transform

Park transformation is used to transform the current vectors, obtained after Clarke transformation, from α/β-axis coordinate to d/q-axis coordinate to get the sampled d/q-axis current FOC__ID and FOC__IQ.

15.1.5 SPWM

15.1.5.1 Unipolar SPWM Mode

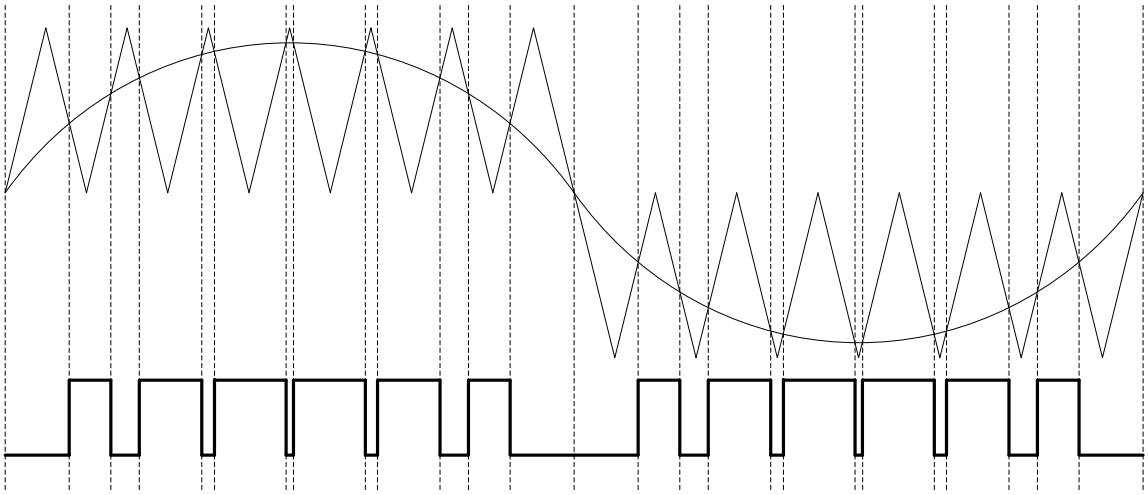


Figure 15-4 Unipolar SPWM Mode

When FOC_CR1[SPWMSEL] = 0, SPWM module works in unipolar SPWM mode.

15.1.5.2 Bipolar SPWM Mode

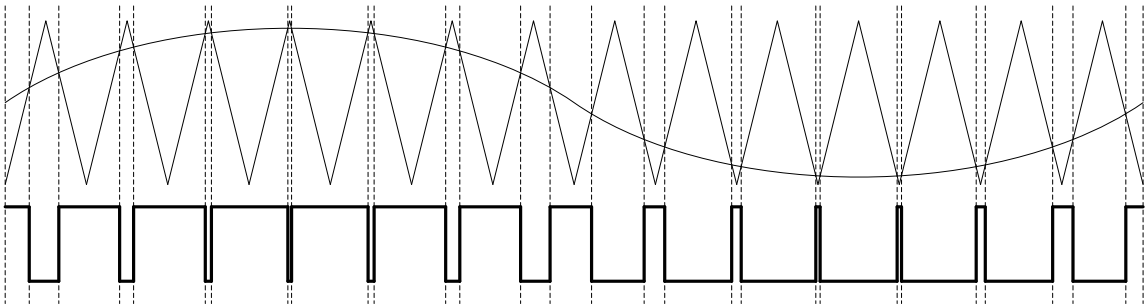


Figure 15-5 Bipolar SPWM Mode

When FOC_CR1[SPWMSEL] = 1, SPWM module works in bipolar SPWM mode.

15.1.6 Current and Voltage Sampling

In SPWM mode, bus voltage and phase current are sampled by hardware automatically. Before the SPWM module operates, ADC and operational amplifier shall be enabled and the corresponding control registers be configured. No configuration is required for ADC channel and mode. Single/dual/triple-shunt current sampling mode is selected by setting FOC_CR1[CSM]. In single-shunt current sampling mode, ADC channel 4 is the default sampling channel of the bus current (itrip). In dual-shunt current sampling mode, ADC channel 0 and channel 1 are the default sampling channels of A-phase current (ia) and B-phase current (ib) respectively. In triple-shunt current sampling mode, ADC channel 0, channel 1 and channel 4 are the default sampling channels of ia, ib and C-phase current (ic) respectively. Channel 2 can be selected for bus voltage sampling.

15.1.6.1 Current Sampling Offset

The current sampling offset voltage shall be added to sample full range of current due to the existence of the positive and negative phase current. When phase current is 0, ADC result is the offset value. ADC result minus this value, 0x4000 default, is the sampling current. Since ADC reference voltage and hardware are nonideal, there is a deviation between the default value and the real value. Therefore, it is necessary to calibrate the offset. The calibration procedure is as follows. When there is no current in three phases, MCU starts to sample the corresponding channel several times, averages all the sampled value, and writes the averaged value to FOC_CSO. Providing ADC sampling range is 0 ~ 5V and the offset is 2.5V, FOC_CSO = 2.5V/5V*32768= 16384 (0x4000).

- When FOC_CHC[CSOC] = 01, FOC_CSO is written to modify the offset of ia.
- When FOC_CHC[CSOC] = 10, FOC_CSO is written to modify the offset of ib.

15.1.7 Angle Mode

The angle module offers forced ramping angle and forced pulling angle.

15.1.7.1 Forced Ramping Angle

Forced ramping angle is controlled by angle register FOC__THETA, speed register FOC__RTHESTEP, acceleration register FOC_RTHEACC and ramping counter FOC_RTHECNT. The formula is:

FOC__RTHESTEP (32 bits) = FOC__RTHESTEP (32 bits) + FOC_RTHEACC (low-order 16 bits)

THETA_OL (16 bits) = THETA_OL (16 bits) + FOC__RTHESTEP (high-order 16 bits)

Where, THETA_OL is an internal variable of the chip. In forced pulling angle mode, THETA_OL is written to FOC__THETA as the used angle. If the software writes a value to FOC__THETA, this value is written to THETA_OL as well.

Forced ramping angle has the highest priority. Configuring FOC_CR1[RFAE] = 1 enables the ramping feature. Ramping module makes a ramping operation in every PWM cycle and the counter is added by 1. When the value of the counter reaches the set value by FOC_RTHECNT, FOC_CR1[RFAE] is cleared by hardware, and then the ramping is completed. Thereafter, according to the value of FOC_CR1[ANGM], the angle comes from estimator (FOC_CR1[ANGM] = 1) or forced pulling angle (FOC_CR1[ANGM] = 0).

15.1.7.2 Forced Pulling Angle

Forced pulling angle is controlled by angle FOC__THETA and speed FOC__RTHESTEP. The formula is: THETA_OL (16 bits) = THETA_OL (16 bits) + FOC__RTHESTEP (high-order 16 bits)

Where, THETA_OL is an internal variable of the chip. In forced pulling angle mode, THETA_OL is written to FOC__THETA as the used angle. If the software writes a value to FOC__THETA, this value is written to THETA_OL as well.

- When FOC_CR1[RFAE] is set to “1” and FOC_CR1[ANGM] to “0”, MCU switches to forced

pulling angle mode after forced ramping angle mode. The speed is the cumulative result after ramp force angle mode. This mode implements a forced uniform speed control.

- When FOC_CR1[RFAE] is set to “0” and FOC_CR1[ANGM] to “0”, the angle is the forced pulling angle and the speed FOC__RTHESTEP is the initial speed written by software. Configuring FOC__RTHESTEP to “0” enables the pre-position feature. The sensor-based FOC is implemented after the motor speed is set with FOC_RTHESTEP. (Principle of Sensor-based FOC: The angle and speed are written to FOC__THETA and FOC__RTHESTEP by software, and FOC module generates an angle in each PWM cycle based on the written values.)

15.1.8 Motor Real-time Parameters

MCU monitors the state of motor using the following real time variables provided by SPWM module:

- Used angle FOC__THETA
- d-axis voltage FOC__UD and q-axis voltage FOC__UQ
- d-axis current FOC__ID and q-axis current FOC__IQ
- α-axis voltage FOC__VALP, β-axis voltage FOC__VBET
- Bus voltage FOC__UDCFLT
- α-axis current (equal to FOC__IA) and β-axis current FOC__IBET

15.2 SPWM Registers

15.2.1 FOC_CR1 (0x40A0)

Bit	7	6	5	4	3	2	1	0
Name	OVMDL	EFAE	RFAE	ANGM	CSM		SPWMSEL	SVPWMEN
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]	OVMDL	IALPHA Reverse Enable 0: Disable 1: Enable
[6]	EFAE	IBETA Reverse Enable 0: Disable 1: Enable
[5]	RFAE	Forced Ramping Angle Enable When this feature is enabled, angle mode is determined by the ramping module. After ramping, it switches to estimated mode or forced pulling mode according to FOC_CR1[ANGM]. FOC_CR1[RFAE] is cleared to “0” by hardware as well. 0: Disable 1: Enable
[4]	ANGM	In SPWM mode, this bit must be set as “0”.
[3:2]	CSM	In SPWM mode, this bit must be set as “01”.
[1]	SPWMSEL	SPWM Output Mode 0: Unipolar 1: Bipolar

[0]	SVPWMEN	SVPWM/SPWM Mode Selection 1: SVPWM 0: SPWM

15.2.2 FOC_CR2 (0x40A1)

Bit	7	6	5	4	3	2	1	0
Name	ESEL	RSV	F5SEG	DSS	CSOC		UQD	UDD
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]	ESEL	IALPHA/IBETA Automatic Reverse Enable When this bit is set to “1”, IALPHA/IBETA will be positive all the time. 0: Disable 1: Enable
[6]	RSV	Reserved
[5]	F5SEG	In SPWM mode, this bit is invalid
[4]	DSS	Current Sampling Mode 0: Sequential Sampling Mode, where the current of two phases are sampled in each carrier period. 1: Alternate Sampling Mode, where FOC module completes the calculation in every PWM cycle, and the current of two phases are sampled alternately in two adjacent PWM cycles.
[3:2]	CSOC	Current Sampling Offset Calibration This bit is written to select the offset of FOC_CSO. In SPWM mode, a write of “01” calibrates IALPHA and “10” calibrates IBETA. 00/11: No effect 01: IALPHA 10: IBETA
[1]	UQD	q-axis PI controller is disabled, where FOC__UQ value is no longer updated by the PI controller. 0: q-axis PI controller is enabled 1: q-axis PI controller is disabled
[0]	UDD	d-axis PI controller is disabled, where the FOC__UD value is no longer updated by the PI controller. 0: d-axis PI controller is enabled 1: d-axis PI controller is disabled

15.2.3 FOC_TRGDLY (0x40A5)

Bit	7	6	5	4	3	2	1	0
Name	FOC_TRGDLY							
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]	FOC_TRGDLY	Time Configuration for Current Sampling When FOC_TRGDLY is set to “0”, SPWM module samples the current as follows. Single-shunt Current Sampling Mode: Midpoint between deadtime and applied time of active voltage vector Dual/Triple-shunt Current Sampling Mode: Midpoint of vector 000 (Driver count value = 0). Range [-128, 127] Single-shunt Current Sampling Mode: If FOC_TRGDLY = 5, it delays by 5*T = 208ns to sample the current, and if FOC_TRGDLY = 0xFB

		(complement) or FOC_TRGDLY = -5, it advances by 5*T=208ns. Dual-shunt/Triple-shunt Current Sampling Mode: If FOC_TRGDLY = 0x85 (the highest bit, and the remaining 7 bits are absolute values) and Driver timer counts down, it samples the current at 5*T = 208ns before an overflow event occurs. If FOC_TRGDLY = 5 and Driver timer counts up, it samples the current at 5*T = 208ns after an overflow event occurs.

15.2.4 FOC_CSO (0x40A6, 0x40A7)

FOC_CSOH(0x40A6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_CSO[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	0	0	0	0	0
FOC_CSOL(0x40A7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_CSO[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_CSO	Current Sampling Offset FOC_CR2[CSOC] is configured to select the current, and FOC_CSO is written to calibrate current sampling offset of itrip in single-shunt current sampling mode, ia, ib in dual-shunt current sampling mode and ia, ib and ic in triple-shunt current sampling mode. Range [0, 32767]. The MSB is always 0 Example: Assuming that the ADC voltage falls within 0V ~ 5V with a reference value of 2.5V, then FOC_CSO = 2.5V/5V*32768 = 16384(0x4000)

15.2.5 FOC__RTHESTEP (0x40A8, 0x40A9)

FOC__RTHESTEPH(0x40A8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__RTHESTEP[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
FOC__RTHESTEPL(0x40A9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__RTHESTEP[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]	FOC__RTHESTEP	Speed of Ramping Module FOC__RTHESTEP is an internal 32-bit variable. MSB is sign bit. High-order 16 bits are written by software. Range [-32768, 32767] FOC__RTHESTEP (32 bits) = FOC__RTHESTEP (32 bits) + FOC_RTHEACC (low-order 16 bits) THETA_OL (16 bits) = THETA_OL (16 bits) + FOC__RTHESTEP (high-order 16 bits)

15.2.6 FOC_RTHEACC (0x40AA, 0x40AB)

FOC_RTHEACCH(0x40AA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_RTHEACC[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_RTHEACCL(0x40AB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_RTHEACC[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_RTHEACC	Ramping Acceleration; FOC_RTHEACC is an internal 32-bit variable. MSB is sign bit. Low-order 16 bits are written by software, and high-order 16 bits are always 0. Range [-32768, 32767] FOC__RTHESTEP (32 bits) = FOC__RTHESTEP (32 bits) + FOC_RTHEACC (low-order 16 bits) THETA_OL (16 bits) = THETA_OL (16 bits) + FOC__RTHESTEP (high-order 16 bits)

15.2.7 FOC_RTHECNT (0x40AC)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_RTHECNT							
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]	FOC_RTHECNT	Max. ramping counts = FOC_RTHECNT*256 When ramping feature is enabled, the ramping angle increases in each PWM cycle. After FOC_RTHECNT*256 times, ramping feature is disabled.

15.2.8 FOC_DMAX (0x40B0, 0x40B1)

FOC_DMAXH(0x40B0)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_DMAX[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_DMAXL(0x40B1)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_DMAX[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_DMAX	Max. output of d-axis PI controller Range [-32768,32767]

15.2.9 FOC_DMIN (0x40B2, 0x40B3)

FOC_DMINH(0x40B2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_DMIN[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_DMINL(0x40B3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_DMIN[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_DMIN	Min. output of d-axis PI controller Range [-32768,32767]

15.2.10 FOC_QMAX (0x40B4, 0x40B5)

FOC_QMAXH(0x40B4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_QMAX[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
FOC_QMAXL(0x40B5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_QMAX[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]	FOC_QMAX	Max. output of q-axis PI controller Range [-32768,32767]

15.2.11 FOC_QMIN (0x40B6, 0x40B7)

FOC_QMINH(0x40B6)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_QMIN[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
FOC_QMINL(0x40B7)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_QMIN[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]	FOC_QMIN	Min. output of q-axis PI controller Range [-32768,32767]

15.2.12 FOC__UD (0x40B8, 0x40B9)

FOC__UDH(0x40B8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__UD[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
FOC__UDL(0x40B9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__UD[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]	FOC__UD	d-axis voltage calculated by d-axis PI controller Range [-32768,32767]

15.2.13 FOC__UQ (0x40BA, 0x40BB)

FOC__UQH(0x40BA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__UQ[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
FOC__UQL(0x40BB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__UQ[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]	FOC__UQ	q-axis voltage calculated by q-axis PI controller Range [-32768,32767]

15.2.14 FOC__ID (0x40BC, 0x40BD)

FOC__IDH(0x40BC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__ID[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IDL(0x40BD)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__ID[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]	FOC__ID	d-axis current from coordinate transformation. Range [-32768,32767]

15.2.15 FOC__IQ (0x40BE, 0x40BF)

FOC__IQH(0x40BE)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IQ[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IQL(0x40BF)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IQ[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]	FOC__IQ	q-axis current from coordinate transformation; Range [-32768,32767]

15.2.16 FOC__IBET (0x40C0, 0x40C1)

FOC__IBETH(0x40C0)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IBETL(0x40C1)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IBET[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]	FOC__IBET	β-axis current from coordinate transformation. Range [-32768,32767]

15.2.17 FOC__VBET (0x40C2, 0x40C3)

FOC__VBETH(0x40C2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__VBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__VBETL(0x40C3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__VBET[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]	FOC__VBET	β-axis output voltage of FOC module Range [-32768,32767]

15.2.18 FOC__VALP (0x40C4, 0x40C5)

FOC__VALPH(0x40C4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__VALP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__VALPL(0x40C5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__VALP[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]	FOC__VALP	α-axis output voltage of FOC module Range [-32768,32767]

15.2.19 FOC_UDCPS (0x40C2, 0x40C3)

FOC_UDCPSH(0x40C2)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_UDCPS[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_UDCPSL(0x40C3)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_UDCPS[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_UDCPS	d-axis Voltage Compensation Value The result of d-axis PI controller (FOC__UD) added to FOC_UDCPS is transferred to the next module. Range [-32768,32767]

15.2.20 FOC_UQCPS (0x40C4, 0x40C5)

FOC_UQCPSH(0x40C4)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_UQCPS[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_UQCPSL(0x40C5)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_UQCPS[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description

[15:0]	FOC_UQCPS	q-axis Voltage Compensation Value The result of q-axis PI controller (FOC__UD) added to FOC_UQCPS is transferred to the next module. Range [-32768,32767]

15.2.21 FOC__IB (0x40C8, 0x40C9)

FOC__IBH(0x40C8)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IB[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IBL(0x40C9)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IB[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]	FOC__IB	Sampled Phase-B Current Range [-32768,32767]

15.2.22 FOC__IA (0x40CA, 0x40CB)

FOC__IAH(0x40CA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IA[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IAL(0x40CB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IA[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]	FOC__IA	Sampled Phase-A Current Range [-32768,32767]

15.2.23 FOC__THETA (0x40CC, 0x40CD)

FOC__THETAH(0x40CC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__THETA[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
FOC__THETAL(0x40CD)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__THETA[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]	FOC__THETA	Output Angle of FOC Module Range [-32768, 32767] The bit value -32768 ~ 32767 corresponds to angle range -180°~ 180°. Example: Assuming that FOC__THETA = 8192, the output angle is 8192/32768*180°= 45°.

15.2.24 FOC__IAMAX (0x40DA, 0x40DB)

FOC__IAMAXH(0x40DA)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IAMAX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IAMAXL(0x40DB)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IAMAX[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]	FOC__IAMAX	Max. Phase-A Current Recorded maximum value of phase-A current; This value may be unreliable unless the motor rotates in a full electrical period. This value will not be cleared to “0” automatically unless FOC_CR2[ICLR] is set to “1”. Range [-32768,32767]

15.2.25 FOC__IBMAX (0x40DC, 0x40DD)

FOC__IBMAXH(0x40DC)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__IBMAX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IBMAXL(0x40DD)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__IBMAX[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]	FOC__IBMAX	Max. Phase-B Current Recorded maximum value of phase-B current; This value may be unreliable unless the motor rotates in a full electrical period. This value will not be cleared to “0” automatically unless FOC_CR2[ICLR] is set to “1”. Range [-32768,32767]

15.2.26 FOC_IDREF (0x4090, 0x4091) (Shared with BLDC Motor Control)

FOC_IDREFH(0x4090)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_IDREF[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
FOC_IDREFL(0x4091)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_IDREF[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]	FOC_IDREF	User-defined d-axis Current Range [-32768,32767]

15.2.27 FOC_IQREF (0x4092, 0x4093) (Shared with BLDC Motor Control)

FOC_IQREFH(0x4092)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_IQREF[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
FOC_IQREFL(0x4093)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_IQREF[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]	FOC_IQREF	User-defined q-axis Current Range [-32768,32767]

15.2.28 FOC_DQKP (0x4094, 0x4095) (Shared with BLDC Motor Control)

FOC_DQKPH(0x4094)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_QKP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_DQKPL(0x4095)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_DQKP[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]	FOC_DQKP	KP coefficient of d/q-axis PI controller. MSB is always 0. Q12 format. Range [0, 32767] corresponds to range of Q12 [0, 8].

15.2.29 FOC_DQKI (0x4096, 0x4097) (Shared with BLDC Motor Control)

FOC_DQKIH(0x4096)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC_QKI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_DQKIL(0x4097)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC_DQKI[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]	FOC_DQKI	KI coefficient of d/q-axis PI controller. MSB is always 0. Q15 format. . Range [0, 32767] corresponds to range of Q15 [0, 1].

15.2.30 FOC_UDCFLT (0x4098, 0x4099)

FOC__UDCFLTH(0x4098)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	FOC__UDCFLT[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__UDCFLTL(0x4099)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	FOC__UDCFLT[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]	FOC__UDCFLT	Filtered Bus Voltage SPWM module samples the bus voltage and filters it to obtain FOC__UDCFLT. ADC channel 2 (external voltage divider) can be selected. Range [0, 32767] Example: The bus voltage is scaled down by 1/6 before feeding into the ADC module, ADC VREF = 5V (namely, the sampling range is 0V~30V and FOC__UDCFLT = 19661(0x4CCD), then bus voltage = 19661/32768*5V*6 = 18V.

16 Timer1

16.1 Timer1 Operations

Timer1 consists of a 16-bit up-counting Base Timer and a 16-bit up-counting Reload Timer. Timer1 can be used in the applications of square-wave controlled BLDC motor drive. Timer1 features as follows.

- The 16-bit up-counting Base Timer is used to record the time between twice position detected events or twice phases commutations (60 degree time) and also can be used for forced phase commutation control when phase detection fails.
- The 16-bit up-counting Reload Timer is used to control the time from position detected to phase commutation, as well as masking time for diode freewheeling after phase commutation (prohibit position detection time).
- The 3-bit programmable frequency prescaler divides the system clock. The divided clock is used as the clock source of the two timers.
- Configurable filtering signals and sampling delay for position detection
- Position detection module generates the position signal required for phase commutation according to the input signal
- 7 groups state register control comparator and pre-driver output
- 6 interrupt sources

The internal structure of Timer1 is shown in Figure 16-1.

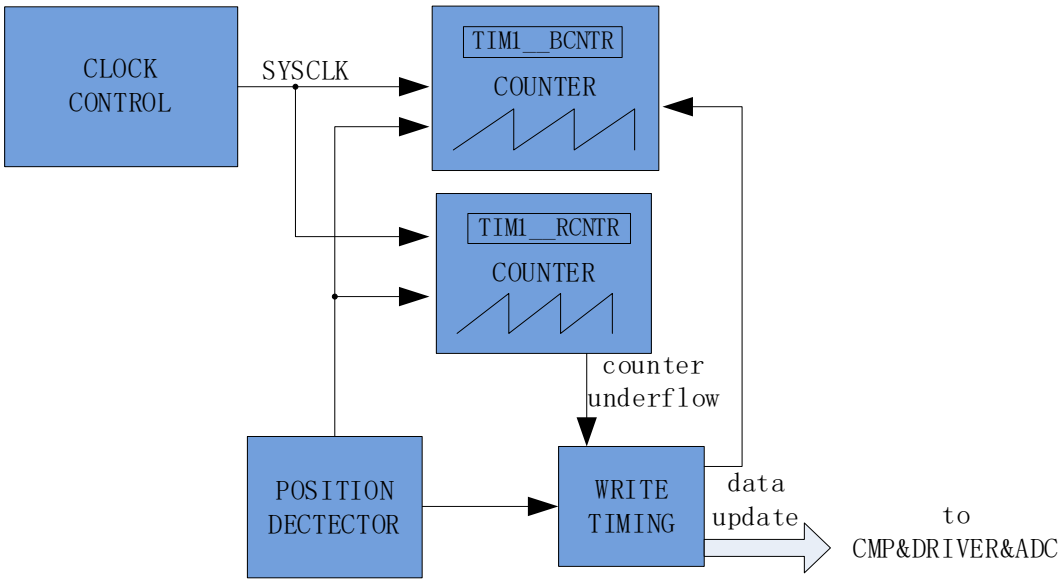


Figure 16-1 Timer1 Internal Structure

16.1.1 Timer1 Counter Module

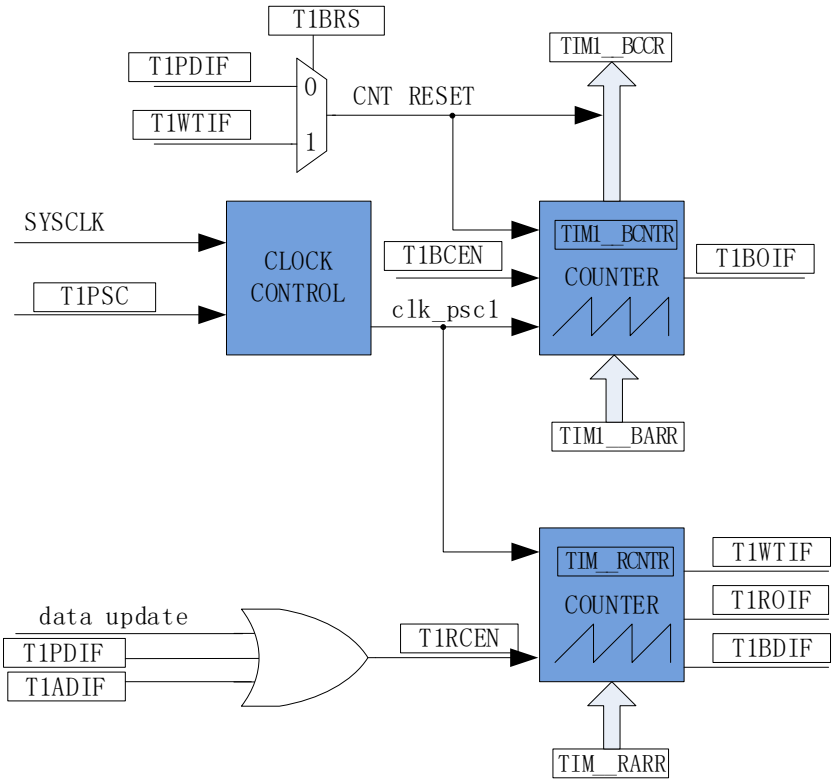


Figure 16-2 Timebase Unit

Timer1 consists of a frequency prescaler, an 16-bit up-counting Base Timer and an 16-bit up-counting Reload Timer.

16.1.1.1 Prescaler

Prescaler divides the system clock frequency and generates the counter clock source for Base Timer and Reload Timer. It offers 8 division coefficients and can be selected through TIM1_CR3[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 both the Basic Timer and Reload Timer are not working. The clock rate clk_psc1 = SYSCLK/(2^TIM1_CR3[T1PSC]). The clock rate corresponding to TIM1_CR3[T1PSC] is shown in Table 16-1.

Table 16-1 Mapping between Clock Rate and TIM1_CR3[T1PSC] Bit

TIM1_CR3[T1PSC]	Division Factor	clk_psc1(Hz)	TIM1_CR3[T1PSC]	Division Factor	clk_psc1(Hz)
000	1	24M	100	16	1.5M
001	2	12M	101	32	750k
010	4	6M	110	64	375k
011	8	3M	111	128	187.5k

16.1.1.2 Base Timer

The Base Timer is a 16-bit up timer with its count value held in the TIM1__BCNTR register. TIM1__BCNTR value is loaded into Capture Register TIM1__BCCR upon a Position Detected Interrupt TIM1_SR[T1PDIF] or a Write Timing Interrupt TIM1_SR[T1WTIF] (selected by TIM1_CR2[T1BRS] bit). Meanwhile, TIM1__BCNTR bit is cleared to "0" and restarts the counter cycle. TIM1__BCCR captures the time between two Position Detected Interrupts or two Write Timing Interrupts (i.e. 60° commutation time). These time inputs are averaged multiple times (programmed by the TIM1_CR0[T1CFLT] bit) before loading the average as a 60° commutation base into the TIM1_BCOR register. When Auto-load Register TIM1__BARR is enabled (TIM1_CR1[BAPE] is set to "1"), TIM1__BARR loads the value of TIM1_BCOR by hardware. When count value of TIM1__BCNTR increases to TIM1__BARR, overflow interrupt flag TIM1_SR[T1BOIF] of the Basic Timer is set to "1". If forced commutation feature is enabled, phase commutation occurs and the Basic Timer Register is cleared to "0". Otherwise, the Basic Timer Register will not be cleared until it counts up to 0xFFFF and becomes overflowed.

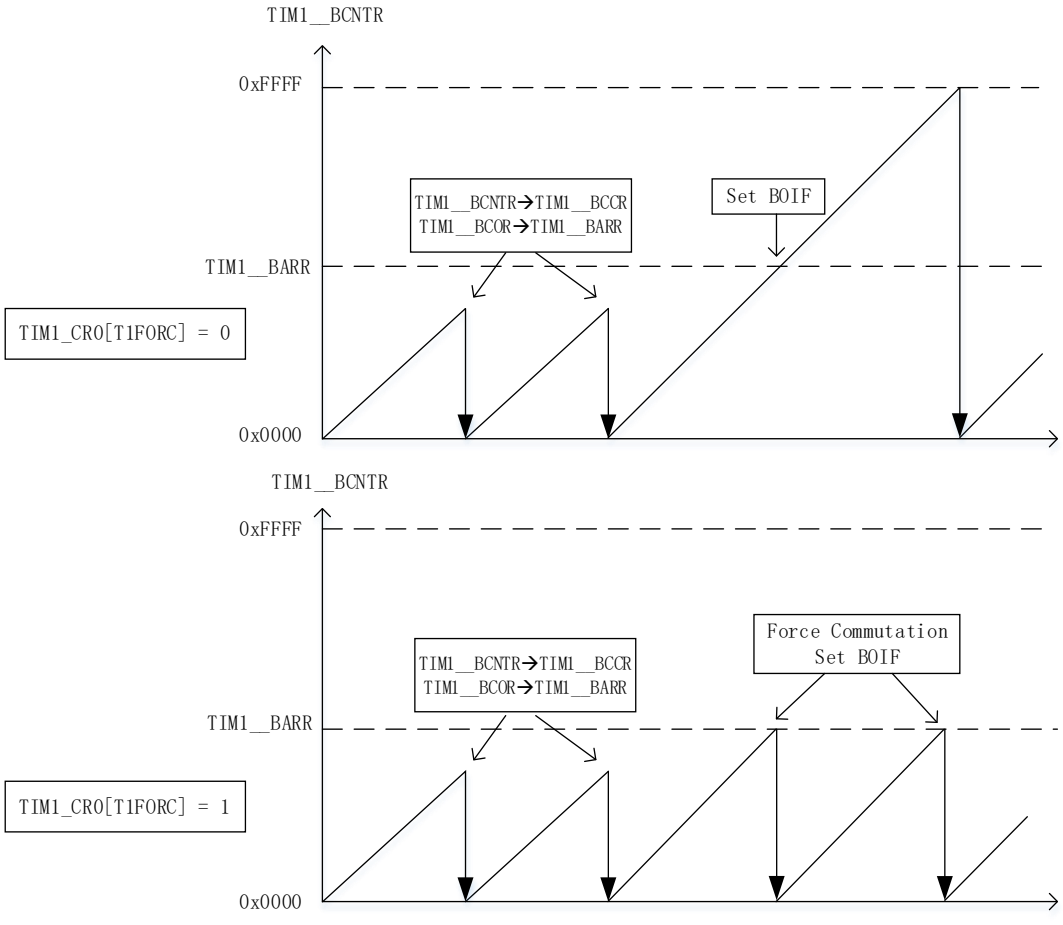


Figure 16-3 Waveform of Base Timer

16.1.1.3 Reload Timer

The Reload Timer is a 16-bit up timer with its count value held in TIM1__RCNTR. The timer overflows when TIM1__RCNTR increases to TIM1__RARR. It stops counting when TIM1_SR[T1ROIF] (overflow interrupt flag of the reload counter) is set to “1”, and TIM1__RCNTR and TIM1_CR0[T1RCEN] are cleared to “0”. TIM1_CR0[T1RCEN] is set to “1” to restart Reload Timer when position detection interrupt or write timing interrupt is generated.

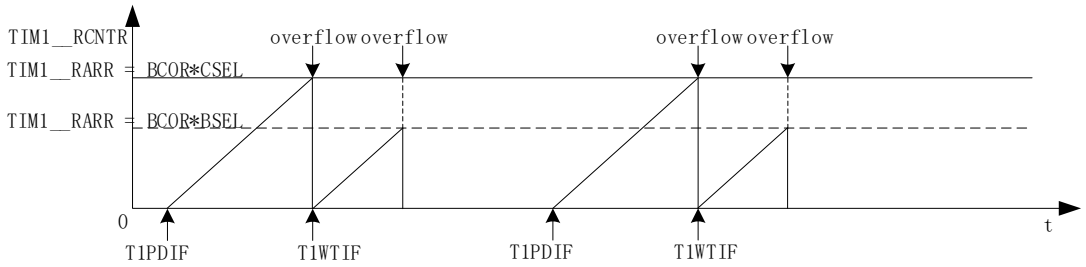


Figure 16-4 Waveform of Reload Timer

16.1.2 Position Detection

16.1.2.1 Position Detection Signal

The TIM1_CR3[T1TIS] bit selects the sources of Position Detection signal, including CMP0/1/2, (CMP Position Detection), GPIO (Hall Sensor Position Detection) or ADC (ACD Position Detection). HALL_CR[HALLSEL] bit is used to configure GPIO sourced by P1.4/P1.6/P2.1 (Hall signal input after function switching) or P0.2/P3.7/P3.6. TIM1_CR3[T1INM] bit decides whether CMP/GPIO signal is filtered. A Position Detected Interrupt is generated upon the completion of position detection. Position Detected Interrupts are divided into CMP/GPIO Position Detected Interrupt and ADC Position Detected Interrupt.

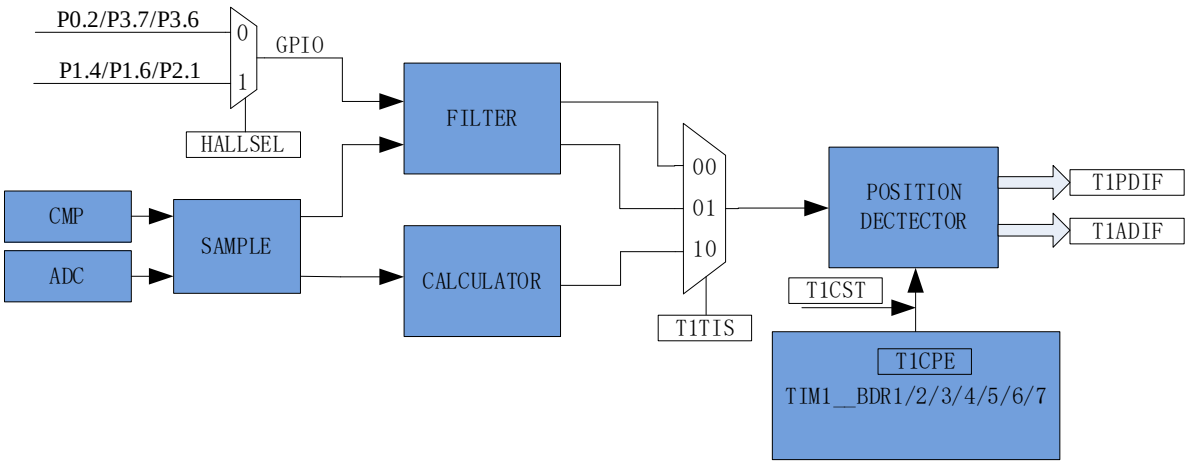


Figure 16-5 Functional Block Diagram of Position Detection

16.1.2.2 CMP/GPIO Position Detection Event

The register bank TIM1_DBR1/2/3/4/5/6/7[T1CPE] is configured to select the active edge of position detection signal. When an active edge of CMP/GPIO Position Detection signal is detected, it indicates the position detection is successfully done, allowing the CMP/GPIO Position Detected Interrupt Flag TIM1_SR[T1PDIF] bit to become "1". TIM1_CR4[T1CST] bit selects TIM1_DBR1/2/3/4/5/6/7[T1CPE] timing.

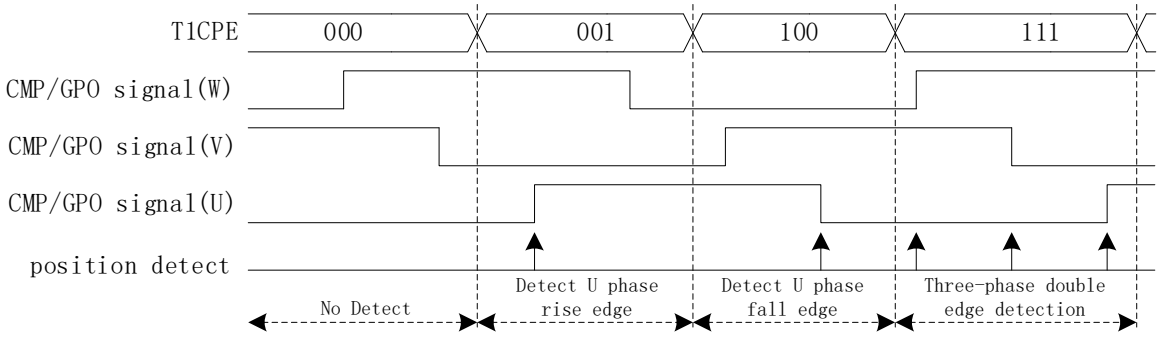


Figure 16-6 Timing Diagram of CMP/GPIO Position Detection

The relation between active edge and TIM1_DBR1/2/3/4/5/6/7[T1CPE] is shown in Table 16-2.

Table 16-2 Mapping between Active Edge and TIM1_DBR1/2/3/4/5/6/7[T1CPE]

T1CPE	Description	T1CPE	Description
000	0	100	U-phase corresponding comparator is enabled when falling edge of U-phase is detected.
001	U-phase corresponding comparator is enabled when rising edge of U-phase is detected.	101	W-phase corresponding comparator is enabled when rising edge of W-phase is detected.
010	W-phase corresponding comparator is enabled when falling edge of W-phase is detected.	110	V-phase corresponding comparator is enabled when falling edge of V-phase is detected.
011	V-phase corresponding comparator is enabled when rising edge of V-phase is detected.	111	U+W+V-phase corresponding comparator is enabled when rising or falling edge of U+W+V-phase is detected.

16.1.2.3 ADC Position Detection Event

TIM1_CR3[T1TIS] is configured to select the position detection signal from ADC. Timer1 controls ADC to sample the voltage of active phase and floating phase, which are calculated in the following equation:

$$TIM1\_URES = K \times TIM1\_UCOP - TIM1\_UFLP$$

Where,

K: ADC Position Detection Coefficient

TIM1__UCOP: ADC sampled value of active phase

TIM1__UFLP: ADC sampled value of floating phase

K/TIM1__UCOP/TIM1__UFLP definitions are determined by TIM1_DBR1/2/3/4/5/6/7[T1CPE] bit, as detailed in Table 16-3.

Table 16-3 Relation between TIM1_DBR1/2/3/4/5/6/7[T1CPE] and K, TIM1__UCOP and TIM1__UFLP

T1CPE	Description
000	Reserved
001	TIM1__KR for K, W-phase voltage for TIM1__UCOP, and U-phase voltage for TIM1__UFLP
010	TIM1__KF for K, U-phase voltage for TIM1__UCOP, and W-phase voltage for TIM1__UFLP
011	TIM1__KR for K, U-phase voltage for TIM1__UCOP, and V-phase voltage for TIM1__UFLP
100	TIM1__KF for K, V-phase voltage for TIM1__UCOP, and U-phase voltage for TIM1__UFLP
101	TIM1__KR for K, V-phase voltage for TIM1__UCOP, and W-phase voltage for TIM1__UFLP
110	TIM1__KF for K, W-phase voltage for TIM1__UCOP, and V-phase voltage for TIM1__UFLP
111	Reserved

When TIM1__URES has a negative step or a positive step, an ADC Position Detected Interrupt is generated and TIM1_SR[T1ADIF] (Position Detected Interrupt Flag) is set to “1”. The position at which ADC Position Detected Interrupt is generated is controlled by setting the coefficient K. In this case, the phase commutation degree can be controlled flexibly.

16.1.2.4 Sampling

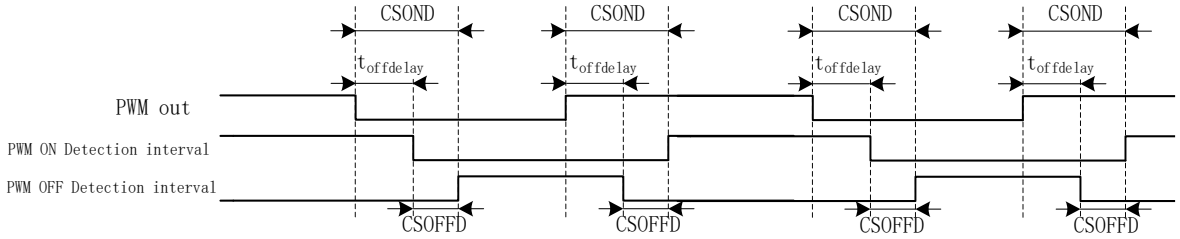


Figure 16-7 Timing Diagram of Sampling

Affected by switching rate of the power device, BEMF signal lags behind PWM output. CMP_SAMR[CSOFFD] and CMP_SAMR[CSOND] bits shall be set reasonably to adjust the sampling interval and obtain the valid position detection signal. When TIM1_CR3[T1TIS] = 01 or 10, Timer1 enables CMP0/1/2 to output the compare results between phase BEMF and neutral point, or starts ADC module to sample floating voltage.

See section 30.1.5 for details.

16.1.2.5 Filtering

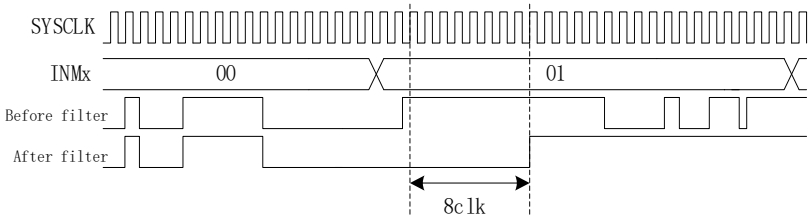


Figure 16-8 Timing Diagram of Filtering Module

According to TIM1_CR3[T1INM], the filtered pulse width of input noise can be selected as 8/32/64 system clock. After this feature is enabled, the signal is lagged behind about 8/32/64 system clocks.

16.1.3 Write Timing Interrupt

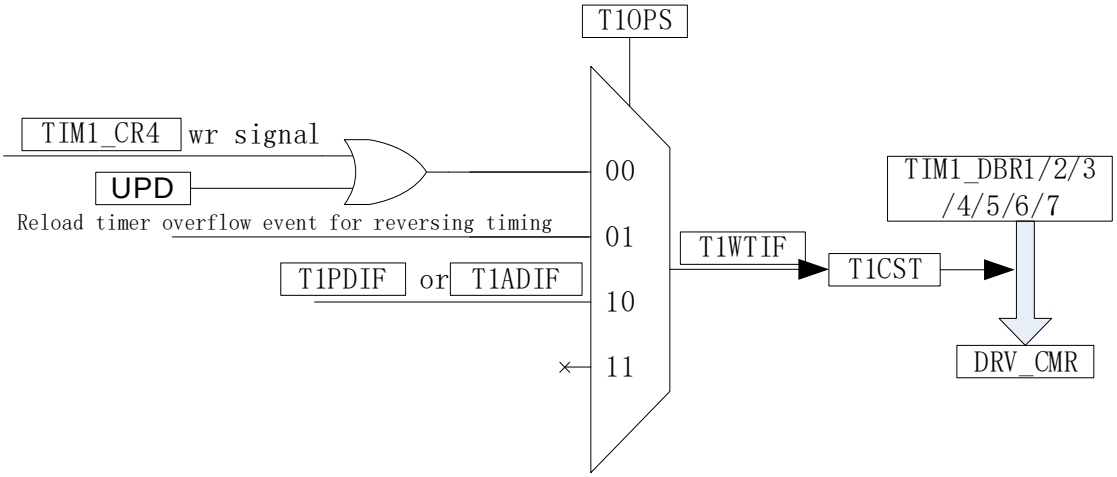


Figure 16-9 Write Timing Block Diagram

When the control logic, predefined in TIM1_DBR1/2/3/4/5/6/7, is sent to driver register DRV_CMR, a writing sequence interrupt is generated. The triggered source is selected by the configuration of TIM1_CR0[T1OPS], and software, Reload Timer overflow event or position detection event can be selected. When a write sequence interrupt is generated, write sequence interrupt flag TIM1_SR[T1WTIF] is set to “1”. If TIM1_CR4[T1CST] ranges in 001~110, TIM1_CR4[T1CST] adds 1 automatically.

16.1.4 Timer1 Interrupt

Timer1 supports 6 interrupt sources:

- Base Timer overflow interrupt
- Reload Timer overflow interrupt
- Write sequence interrupt
- Diode Freewheeling End Interrupt

- CMP/GPIO Position Detected Interrupt
- ADC Position Detected Interrupt

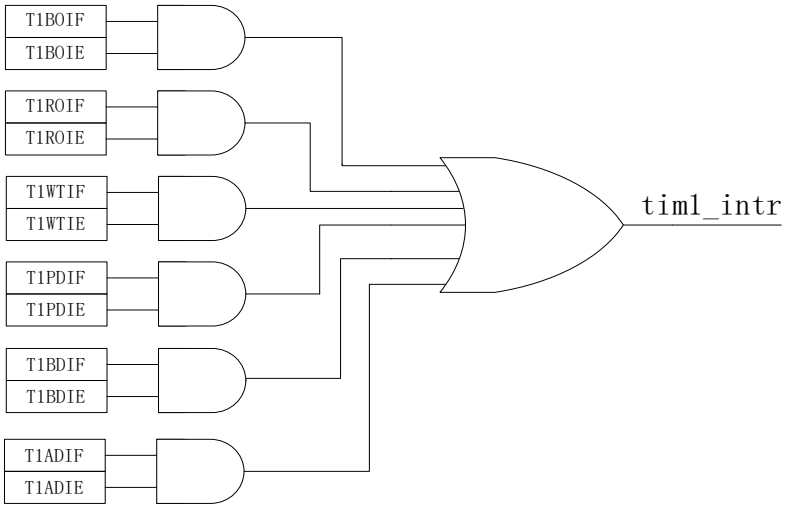


Figure 16-10 Timer1 Interrupt Sources

16.2 Square-wave Control Based BLDC Motor

For BLDC motor square-wave control application, Timer1 works with CMP0/1/2 and Driver module to achieve the following features:

- Automatic record of 60 degree time, filtered as 60 degree reference time
- Automatic forced phase commutation when no position signal is detected
- Automatic diode freewheeling masking, i.e., stopping comparator detection during diode freewheeling
- Automatic control of the time from position detected to phase commutation to achieve automatic commutation
- Take over CMP_CR2[CMP0SEL] to control CMP0/1/2 automatically
- Comparator signal can be set to avoid ringing at the switch node on power IC, and the signal can be filtered after sampling
- Take over DRV_CMR register to control six-way PWM outputs automatically

16.2.1 Six-step Phase Commutation of Square Wave Control

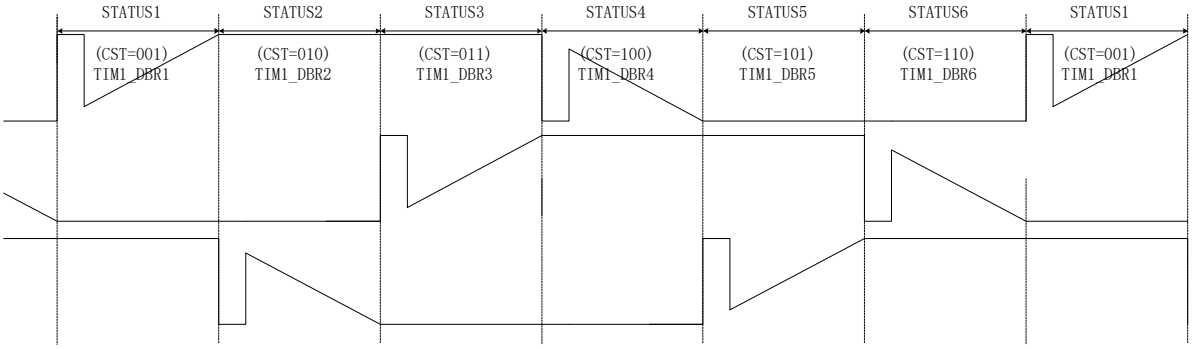


Figure 16-11 Diagram of Six-step Phase Commutation of Square Wave Control

TIM1_CR4[T1CST] is the commutation state machine. Among them, state 0 is used to output off state, and state 7 is customizable for braking, pre-charging, pre-positioning, startup, etc. States 1~6 are used for six-step automatic commutation, and the state machine TIM1_CR4[T1CST] automatically adds 1 after phase commutation.

The states 1~7 maps to the TIM1_DBR1~7. When write sequence interrupt occurs, TIM1_DBRx corresponding to the current state is automatically transferred to DRV_CMR and CMP_CR2[CMP0SEL] for phase commutation and position detection.

16.2.2 Square Wave Control Working Principle

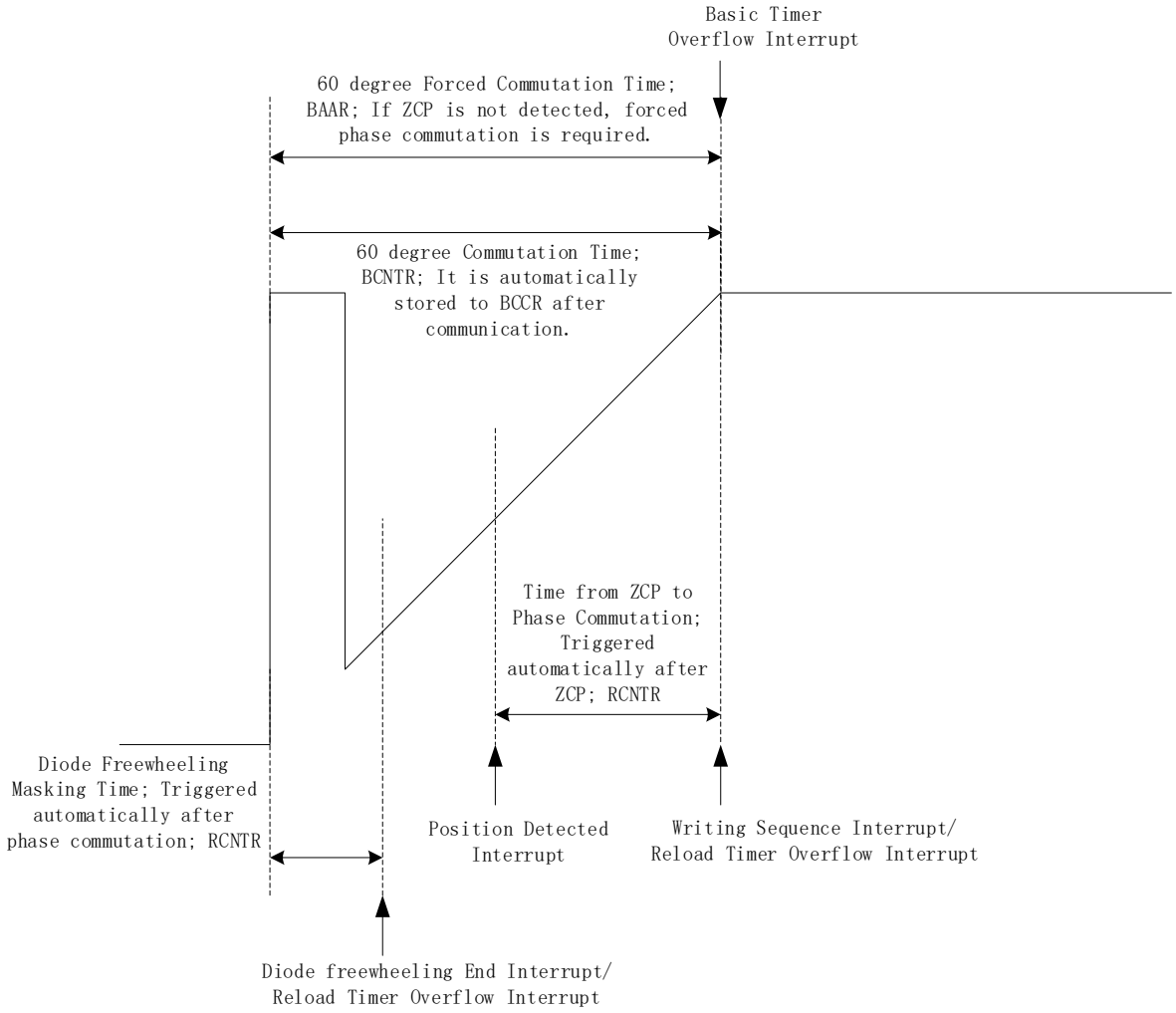


Figure 16-12 Square Wave Control Working Principle

16.2.2.1 60° Commutation Base Time

TIM1__BCCR captures the time of last 60 degree. TIM1_CR2[T1BRS] is set to “0” to capture the time between two write sequence interrupts and TIM1_CR2[T1BRS] to “1” to capture the time between two position detected interrupts.

TIM1_BCOR is the filtered 60 degree time, i.e., 60 degree base time. TIM1_CR0[T1CFLT] can select the last 1/2/4/8 TIM1__BCCR averaged to obtain TIM1_BCOR.

In square-wave control mode, the diode freewheeling masking time, the time from position detected to commutation, and the time to forced commutation are determined by the 60 degree base time TIM1_BCOR.

When Base Timer is auto-load enabled (TIM1_CR1[T1BAPE] = 1), and is reset due to a position detection interrupt or a write timing interrupt, TIM1_BCOR is transferred to TIM1__BARR to control the forced phase commutation.

16.2.2.2 Forced Commutation at 60°

When the motor rotates smoothly, ZCP is generally detected after 30 degrees of rotation after a phase commutation and a position detection interrupt is generated. If ZCP is not detected in 60 degree after the phase commutation, position detection fails and a forced phase commutation is required.

In this case, TIM1_CR0[T1FORC] is set to "1" to enable the forced commutation feature. During previous commutation, the timer TIM1__BCNTR is cleared to "0" by timing interrupt and restarts counting, while TIM1__BCCR captures the count value held in TIM1__BCNTR, which is filtered and stored in TIM1_BCOR as the 60 degree base time. When auto-load feature is enabled (TIM1_CR1[T1BAPE] = 1), the value held in TIM1_BCOR is loaded into TIM1__BARR after the Base Timer is cleared. If no ZCP is detected in 60 degree after commutation (TIM1__BCNTR matches TIM1__BARR), TIM1_SR[T1BOIF] (overflow interrupt flag of the Basic Timer) is set to "1" for forced phase commutation, and the timer TIM1__BCNTR is cleared to "0". But if an ZCP is detected within 60 degrees after phase commutation, even if TIM1__BCNTR > TIM1__BARR, the forced commutation will not be triggered and TIM1_SR[T1BOIF] will not be set to "1". When forced commutation feature is disabled (TIM1_CR0[T1FORC] = 0) and TIM1__BCNTR > TIM1__BARR, the interrupt flag TIM1_SR[T1BOIF] is set to "1" and no forced phase commutation is automatically performed. Phase commutation can be performed manually by Basic Timer overflow interrupt flag and the position detected interrupt flag.

16.2.2.3 Diode Freewheeling Masking

After commutation, inductance energy of the phase is released to the power supply or ground through the diode since the original active phase becomes a floating phase. During diode freewheeling, the floating phase BEMF signal cannot be measured. By masking comparator signal or ADC sampling value during diode freewheeling time, wrong commutation caused by wrong signal generated by the freewheeling is avoided. After freewheeling masking, the freewheeling masking end interrupt flag TIM1_SR[T1BDIF] is generated.

Freewheeling masking time is set by TIM1_CR1[BSEL] with the formula: Masking angle = TIM1_CR1[BSEL]/128*60°.

16.2.2.4 Angle of Position Detected to Commutation

After commutation, a ZCP is detected (generating a position detected interrupt) and the hardware starts counting according to the software-set time between ZCP and the commutation. After the counting ends, the hardware automatically implements phase commutation and generates the write sequence interrupt flag TIM1_SR[T1WTIF].

The time between ZCP and commutation is set by TIM1_CR2[CSEL] with the formula: Commutation angle = TIM1_CR2[CSEL]/128*60°.

16.2.2.5 Cycle-by-cycle Current Limiting

See section 30.1.1.2.

16.3 Timer1 Registers

16.3.1 TIM1_CR0 (0x4068)

Bit	7	6	5	4	3	2	1	0
Name	T1RWEN	T1CFLT		T1FORC	T1OPS		T1BCEN	T1RCEN
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]	T1RWEN	Write to TIM1_CR0[T1RCEN] Enable 0: No effect 1: When TIM1_CR0 is updated, TIM1_CR0[T1RWEN] and TIM1_CR0[T1RCEN] shall be configured simultaneously to enable or disable TIM1_CR0[T1RCEN]. A write of “0x81” to TIM1_CR0 enables TIM1_CR0[T1RCEN], and “0x80” to disables TIM1_CR0[T1RCEN].
[6:5]	T1CFLT	60 Degree Base Time Filtering Selection The average of previous x times 60 degree (TIM1__BCCR) is used as the base time (TIM1__BCOR). 00: 1 times 60 degree 01: 2 times 60 degree 10: 4 times 60 degree 11: 8 times 60 degree
[4]	T1FORC	Forced Phase Commutation at 60° Enable 0: Disable 1: Enable Note: If a ZCP is detected, forced phase commutation will not be implemented even if this bit is enabled.
[3:2]	T1OPS	Commutation Trigger Signal Selection The bit selects the trigger signal for TIM1_DBRx to transfer data to DRV_CMR. 00: The transfer is triggered upon a write of “1” to TIM1_IER[T1UPD] in software or a write to TIM1_CR4[T1CST]. 01: The transfer is triggered upon an overflow interrupt of reload timer commutation counter. 10: The transfer is triggered upon a Position Detected Interrupt. 11: Reserved
[1]	T1BCEN	Base Timer Enable 0: Disable 1: Enable
[0]	T1RCEN	Reload Timer Enable When TIM1_CR0 is updated, TIM1_CR0[T1RWEN] and TIM1_CR0[T1RCEN] must be configured simultaneously to enable or disable TIM1_CR0[T1RCEN]. A write of “0x81” to TIM1_CR0 enables TIM1_CR0[T1RCEN] and “0x80” disables TIM1_CR0[T1RCEN]. TIM1_CR0[T1RCEN] is automatically enabled upon a Position Detected Interrupt and a Write Timing Interrupt. TIM1_CR0[T1RCEN] is cleared to “0” by hardware upon a Reload Timer Overflow Interrupt. 0: Disable 1: Enable

16.3.2 TIM1_CR1 (0x4069)

Bit	7	6	5	4	3	2	1	0
Name	T1BAPE	BSEL						
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]	T1BAPE	TIM1__BARR Register Auto-load Enable With this bit enabled, TIM1__BCOR is written to TIM1__BARR when Basic Timer is reset due to a Position Detected Interrupt or a Write Timing Interrupt. 0: Disable 1: Enable
[6:0]	BSEL	Diode Freewheeling Masking Angle Selection This bit is used to configure the angle of diode freewheeling masking after phase commutation. Position is not detected during diode freewheeling masking. Equation: Diode freewheeling masking angle = TIM1_CR1[BSEL]/128*60°

16.3.3 TIM1_CR2 (0x406A)

Bit	7	6	5	4	3	2	1	0
Name	T1BRS	CSEL						
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]	T1BRS	Base Timer Reset Source Selection 0: Write Timing Reset 1:Position Detected Interrupt Reset
[6:0]	CSEL	Phase Commutation Angle Selection After a position detected event, phase commutation is implemented after the degree configured by TIM1_CR2[CSEL]. Equation: Commutation angle = TIM1_CR2[CSEL]/128*60°

16.3.4 TIM1_CR3 (0x406B)

Bit	7	6	5	4	3	2	1	0
Name	RSV	T1PSC			T1TIS		T1INM	
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	1	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:4]	T1PSC	Timer Clock Source Frequency Selection This bit is configured to divide the system clock as the clock source for Base Timer and Reload Timer. The clock source frequency of the two timers: 000: 24MHz 001: 12MHz 010: 6MHz 011: 3MHz 100: 1.5MHz 101: 750kHz 110: 375kHz 111: 187.5kHz
[3:2]	T1TIS	Position Detection Signal Selection 00: GPIO (select P1.4, P1.6, P2.1 or P0.2, P3.7, P3.6 according to HALL_CR[HALLSEL] bit) 01: Output signal of CMP0/1/2 10: Output signal of ADC 11: Reserved

[1:0]	T1INM	Noise Pulse Width Selection for TI0/TI1/TI2 When pulse width of the noise is less than the set value, it is filtered as noise. Assuming that MCU clock runs at 24MHz(41.67ns): 00: Disable 01: 8 system clock cycles, 8 x 41.67ns 10: 32 system clock cycles, 32 x 41.67ns 11: 64 system clock cycles, 64 x 41.67ns

16.3.5 TIM1_CR4 (0x406C)

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

Bit	Name	Description
[7:3]	RSV	Reserved
[2:0]	T1CST	Commutation State Machine The state machine corresponds to different TIM1_DBRx at different states. When TIM1_CR4[T1CST] reads 001~111, Timer1 automatically enables or disables CMP0/1/2 according to the TIM1_DBRx[T1CPE]. When TIM1_CR4[T1CST] reads 001~110, Timer1 automatically adds by "1" each cycle upon a Write Timing Interrupt.

Table 16-4 Mapping between TIM1_CR4[T1CST] and TIM1_DBRx

TIM1_CR4[T1CST]	TIM1_DBRx	TIM1_CR4[T1CST]	TIM1_DBRx
000	0	100	TIM1_DBR4
001	TIM1_DBR1	101	TIM1_DBR5
010	TIM1_DBR2	110	TIM1_DBR6
011	TIM1_DBR3	111	TIM1_DBR7

16.3.6 TIM1_IER (0x406D)

Bit	7	6	5	4	3	2	1	0
Name	T1UPD	RSV	T1ADIE	T1BOIE	T1ROIE	T1WTIE	T1PDIE	T1BDIE
Type	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]	T1UPD	When TIM1_CR0[T1OPS] = 00, a write of "1" to this bit enables data transfer. This bit is cleared to "0" by hardware after "1" is written.
[6]	RSV	Reserved
[5]	T1ADIE	ADC Position Detected Interrupt Enable 0: Disable 1: Enable
[4]	T1BOIE	Base Timer Overflow Interrupt Enable 0: Disable 1: Enable
[3]	T1ROIE	Reload Timer Overflow Interrupt Enable 0: Disable 1: Enable
[2]	T1WTIE	Write Timing Interrupt Enable 0: Disable 1: Enable
[1]	T1PDIE	CMP/GPIO Position Detected Interrupt Enable 0: Disable 1: Enable

[0]	T1BDIE	Diode Freewheeling Masking Interrupt Enable 0: Disable 1: Enable

16.3.7 TIM1_SR (0x406E)

Bit	7	6	5	4	3	2	1	0
Name	RSV		T1ADIF	T1BOIF	T1ROIF	T1WTIF	T1PDIF	T1BDIF
Type	-	-	R/W0	R/W0	R/W0	R/W	R/W0	R/W0
Reset	-	-	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	T1ADIF	ADC Position Detected Interrupt Flag A Position Detected Interrupt is generated when TIM1_DBRx[T1CPE] matches ACD Position Detection signal. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect
[4]	T1BOIF	Base Timer Overflow Interrupt Flag An overflow event occurs when Basic Timer counts up and TIM1__BCNTR matches TIM1__BARR. 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 An overflow event occurs and TIM1__RCNTR is cleared to “0” when TIM1__RCNTR matches TIM1__RARR. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect
[2]	T1WTIF	Write sequence interrupt Flag Write sequence interrupt is generated when TIM1_DBRx is transferred to DRV_CMR. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect.
[1]	T1PDIF	CMP/GPIO Position Detected Interrupt Flag A position detected interrupt is generated when CMP/GPIO Position Detection matches TIM1_DBRx[T1CPE]. Read:

		0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect
[0]	T1BDIF	Diode Freewheeling Masking End Interrupt Flag Diode freewheeling masking starts after phase commutation and an interrupt is generated at end. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect

16.3.8 TIM1_BCOR (0x4070, 0x4071)

TIM1__BCORH(0x4070)								
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__BCORL(0x4071)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__BCOR[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__BCOR	This bit is configured to capture filtered count values held in the Base Timer. TIM1__BCCR hods the filtered count value, i.e., 60 Degree Base Time.

16.3.9 TIM1_CR5 (0x4072)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1POP	RSV			ITRIP_DIS	UCOP_DIS	T1AFL	
Type	R	-	-	-	R/W	R/W	R/W	R/W
Reset	0	-	-	-	0	0	0	0

Bit	**Name**	**Description**
[7]	T1POP	Data Transfer Triggered by Driver Timer Overflow This bit is valid only when TIM_CR0[T1OPS] = 00. With it enabled, data transfer is triggered by Driver Timer Overflow, namely, commuting the phase once every PWM cycle 0: Disable 1: Enable
[6:4]	RSV	Reserved
[3]	ITRIP_DIS	Bus Current Sampling Disable 0: Enable 1: Disable
[2]	UCOP_DIS	Active Phase Voltage Sampling Disable 0: Enable 1: Disable

[1:0]	T1AFL[1:0]	ADC Sampled Voltage Calculation Filtering Counts 00: 1 01: 2 10: 4 11: 8

16.3.10 TIM1_DBR1 (0x4074, 0x4075)

TIM1_DBR1H(0x4074)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0
TIM1_DBR1L(0x4075)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1UHP	T1ULP	T1WHE	T1WLE	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
[15]	RSV	Reserved
[14:12]	T1CPE	Position Detection Input Edge Polarity and Comparator Enable Selection This bit is used to define the edge of Position Detection Input and enable or disable the associated comparators. The detected edge in input signal, corresponding to the configuration, generates a position detected interrupt. See CMP/GPIO Position Detection Event and Table 16-2.
[11]	T1WHP	High-side Output Polarity of W-phase 0: Active High 1: Active Low
[10]	T1WLP	Low-side Output Polarity of W-phase 0: Active High 1: Active Low
[9]	T1VHP	High-side Output Polarity of V-phase 0: Active High 1: Active Low
[8]	T1VLP	Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[7]	T1UHP	High-side Output Polarity of U-phase 0: Active High 1: Active Low
[6]	T1ULP	Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[5]	T1WHE	High-side Output Enable of W-phase 0: Disable 1: Enable
[4]	T1WLE	Low-side Output Enable of W-phase 0: Disable 1: Enable
[3]	T1VHE	High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	T1VLE	Low-side Output Enable of V-phase 0: Disable 1: Enable

[1]	T1UHE	High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	Low-side Output Enable of U-phase 0: Disable 1: Enable

Note: The high-side and low-side outputs of W, V and U-phases are complementary and deadtime is automatically added (same for TIM1_DBR2~TIM1_DBR7) when TIM1_DBR1[T1WLE] and TIM1_DBR1[T1WHE], TIM1_DBR1[T1VLE] and TIM1_DBR1[T1VHE] or TIM1_DBR1[T1ULE] and TIM1_DBR1[T1UHE] are set to "1".

16.3.11 TIM1_DBR2 (0x4076, 0x4077)

TIM1_DBR2H(0x4076)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0
TIM1_DBR2L(0x4077)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1UHP	T1ULP	T1WHE	T1WLE	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**
[15]	RSV	Reserved
[14:12]	T1CPE	Position Detection Input Edge Polarity and Comparator Enable Selection This bit is used to define the edge of Position Detection Input and enable or disable the associated comparators. The detected edge in input signal, corresponding to the configuration, generates a position detected interrupt. See CMP/GPIO Position Detection Event and Table 16-2.
[11]	T1WHP	High-side Output Polarity of W-phase 0: Active High 1: Active Low
[10]	T1WLP	Low-side Output Polarity of W-phase 0: Active High 1: Active Low
[9]	T1VHP	High-side Output Polarity of V-phase 0: Active High 1: Active Low
[8]	T1VLP	Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[7]	T1UHP	High-side Output Polarity of U-phase 0: Active High 1: Active Low
[6]	T1ULP	Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[5]	T1WHE	High-side Output Enable of W-phase 0: Disable 1: Enable
[4]	T1WLE	Low-side Output Enable of W-phase 0: Disable 1: Enable

[3]	T1VHE	High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	T1VLE	Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	T1UHE	High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	Low-side Output Enable of U-phase 0: Disable 1: Enable

16.3.12 TIM1_DBR3 (0x4078, 0x4079)

TIM1_DBR3H(0x4078)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0
TIM1_DBR3L(0x4079)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1UHP	T1ULP	T1WHE	T1WLE	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**
[15]	RSV	Reserved
[14:12]	T1CPE	Position Detection Input Edge Polarity and Comparator Enable Selection This bit is used to define the edge of Position Detection Input and enable or disable the associated comparators. The detected edge in input signal, corresponding to the configuration, generates a position detected interrupt. See CMP/GPIO Position Detection Event and Table 16-2.
[11]	T1WHP	High-side Output Polarity of W-phase 0: Active High 1: Active Low
[10]	T1WLP	Low-side Output Polarity of W-phase 0: Active High 1: Active Low
[9]	T1VHP	High-side Output Polarity of V-phase 0: Active High 1: Active Low
[8]	T1VLP	Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[7]	T1UHP	High-side Output Polarity of U-phase 0: Active High 1: Active Low
[6]	T1ULP	Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[5]	T1WHE	High-side Output Enable of W-phase 0: Disable 1: Enable
[4]	T1WLE	Low-side Output Enable of W-phase 0: Disable 1: Enable

[3]	T1VHE	High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	T1VLE	Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	T1UHE	High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	Low-side Output Enable of U-phase 0: Disable 1: Enable

16.3.13 TIM1_DBR4 (0x407A, 0x407B)

TIM1_DBR4H(0x407A)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0
TIM1_DBR4L(0x407B)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1UHP	T1ULP	T1WHE	T1WLE	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**
[15]	RSV	Reserved
[14:12]	T1CPE	Position Detection Input Edge Polarity and Comparator Enable Selection This bit is used to define the edge of Position Detection Input and enable or disable the associated comparators. The detected edge in input signal, corresponding to the configuration, generates a position detected interrupt. See CMP/GPIO Position Detection Event and Table 16-2.
[11]	T1WHP	High-side Output Polarity of W-phase 0: Active High 1: Active Low
[10]	T1WLP	Low-side Output Polarity of W-phase 0: Active High 1: Active Low
[9]	T1VHP	High-side Output Polarity of V-phase 0: Active High 1: Active Low
[8]	T1VLP	Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[7]	T1UHP	High-side Output Polarity of U-phase 0: Active High 1: Active Low
[6]	T1ULP	Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[5]	T1WHE	High-side Output Enable of W-phase 0: Disable 1: Enable
[4]	T1WLE	Low-side Output Enable of W-phase 0: Disable 1: Enable

[3]	T1VHE	High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	T1VLE	Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	T1UHE	High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	Low-side Output Enable of U-phase 0: Disable 1: Enable

16.3.14 TIM1_DBR5 (0x407C, 0x407D)

TIM1_DBR5H(0x407C)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0
TIM1_DBR5L(0x407D)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1UHP	T1ULP	T1WHE	T1WLE	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**
[15]	RSV	Reserved
[14:12]	T1CPE	Position Detection Input Edge Polarity and Comparator Enable Selection This bit is used to define the edge of Position Detection Input and enable or disable the associated comparators. The detected edge in input signal, corresponding to the configuration, generates a position detected interrupt. See CMP/GPIO Position Detection Event and Table 16-2.
[11]	T1WHP	High-side Output Polarity of W-phase 0: Active High 1: Active Low
[10]	T1WLP	Low-side Output Polarity of W-phase 0: Active High 1: Active Low
[9]	T1VHP	High-side Output Polarity of V-phase 0: Active High 1: Active Low
[8]	T1VLP	Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[7]	T1UHP	High-side Output Polarity of U-phase 0: Active High 1: Active Low
[6]	T1ULP	Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[5]	T1WHE	High-side Output Enable of W-phase 0: Disable 1: Enable
[4]	T1WLE	Low-side Output Enable of W-phase 0: Disable 1: Enable

[3]	T1VHE	High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	T1VLE	Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	T1UHE	High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	Low-side Output Enable of U-phase 0: Disable 1: Enable

16.3.15 TIM1_DBR6 (0x407E, 0x407F)

TIM1_DBR6H(0x407E)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0
TIM1_DBR6L(0x407F)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1UHP	T1ULP	T1WHE	T1WLE	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**
[15]	RSV	Reserved
[14:12]	T1CPE	Position Detection Input Edge Polarity and Comparator Enable Selection This bit is used to define the edge of Position Detection Input and enable or disable the associated comparators. The detected edge in input signal, corresponding to the configuration, generates a position detected interrupt. See CMP/GPIO Position Detection Event and Table 16-2.
[11]	T1WHP	High-side Output Polarity of W-phase 0: Active High 1: Active Low
[10]	T1WLP	Low-side Output Polarity of W-phase 0: Active High 1: Active Low
[9]	T1VHP	High-side Output Polarity of V-phase 0: Active High 1: Active Low
[8]	T1VLP	Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[7]	T1UHP	High-side Output Polarity of U-phase 0: Active High 1: Active Low
[6]	T1ULP	Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[5]	T1WHE	High-side Output Enable of W-phase 0: Disable 1: Enable
[4]	T1WLE	Low-side Output Enable of W-phase 0: Disable 1: Enable

[3]	T1VHE	High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	T1VLE	Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	T1UHE	High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	Low-side Output Enable of U-phase 0: Disable 1: Enable

16.3.16 TIM1_DBR7 (0x4080, 0x4081)

TIM1_DBR7H(0x4080)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0
TIM1_DBR7L(0x4081)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	T1UHP	T1ULP	T1WHE	T1WLE	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**
[15]	RSV	Reserved
[14:12]	T1CPE	Position Detection Input Edge Polarity and Comparator Enable Selection This bit is used to define the edge of Position Detection Input and enable or disable the associated comparators. The detected edge in input signal, corresponding to the configuration, generates a position detected interrupt. See CMP/GPIO Position Detection Event and Table 16-2.
[11]	T1WHP	High-side Output Polarity of W-phase 0: Active High 1: Active Low
[10]	T1WLP	Low-side Output Polarity of W-phase 0: Active High 1: Active Low
[9]	T1VHP	High-side Output Polarity of V-phase 0: Active High 1: Active Low
[8]	T1VLP	Low-side Output Polarity of V-phase 0: Active High 1: Active Low
[7]	T1UHP	High-side Output Polarity of U-phase 0: Active High 1: Active Low
[6]	T1ULP	Low-side Output Polarity of U-phase 0: Active High 1: Active Low
[5]	T1WHE	High-side Output Enable of W-phase 0: Disable 1: Enable
[4]	T1WLE	Low-side Output Enable of W-phase 0: Disable 1: Enable

[3]	T1VHE	High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	T1VLE	Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	T1UHE	High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	T1ULE	Low-side Output Enable of U-phase 0: Disable 1: Enable

16.3.17 TIM1__BCNTR (0x4082, 0x4083)

TIM1__BCNTRH(0x4082)								
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	1	0	0	0	0	0	0	0
TIM1__BCNTRL(0x4083)								
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	This bit holds count values of the Base Timer and is used for clocking commutation at 60°. Note: TIM1__BCNTR register selects the reset source according to TIM1_CR2[T1BRS], and TIM1__BCNTR does not restart when TIM1__BCNTR overflow interrupt is generated.

16.3.18 TIM1__BCCR (0x4084, 0x4085)

TIM1__BCCRH(0x4084)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__BCCR[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__BCCRL(0x4085)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__BCCR[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__BCCR	This bit is configured to capture count values held in Base Timer. When the Base Timer is reset on a Position Detected Interrupt or a Write Timing Interrupt, the count values before the reset are stored into TIM1__BCCR.

16.3.19 TIM1__BARR (0x4086, 0x4087)

TIM1__BARRH(0x4086)

Bit	15	14	13	12	11	10	9	8
Name	TIM1__BARR[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__BARRL(0x4087)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__BARR[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__BARR	Reload Value in Base Timer When the count value of the Base Timer equals to TIM1__BARR value, an overflow interrupt is generated and the timer is cleared to “0”.

16.3.20 TIM1__RARR (0x4088, 0x4089)

TIM1__RARRH(0x4088)								
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(0x4089)								
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	Auto-Reload Value in Reload Timer When count of the Reload Timer is equal to TIM1__RARR, an overflow interrupt is generated and the value of the timer is cleared to “0”. The value of diode freewheeling masking angle held in TIM1_CR1[BSEL] is updated to TIM1__RARR when a Write Sequence Interrupt is generated. The value of commutation angle held in TIM1_CR2[CSEL] is updated to TIM1__RARR when a Position Detected Interrupt occurs.

16.3.21 TIM1__RCNTR (0x408A, 0x408B)

TIM1__RCNTRH(0x408A)								
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	1	1	1	1	1	1	1	1
TIM1__RCNTRL(0x408B)								
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	1	1	1	1	1	1	1	1

Bit	Name	Description
[15:0]	TIM1__RCNTR	Count value of the Reload Timer for counting numbers of diode freewheeling masking and ZCP to phase commutation.

16.3.22 TIM1__UCOP (0x408C, 0x408D)

TIM1__UCOPH(0x408C)								
Bit	**7**	**6**	**5**	4	**3**	**2**	**1**	**0**
Name	TIM1__UCOP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__UCOPL(0x408D)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__UCOP[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__UCOP	ADC sampled value of active phase voltage (second-highest bit alignment)

16.3.23 TIM1__UFLP (0x408E, 0x408F)

TIM1__UFLPH(0x408E)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__UCOP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__UFLPL(0x408F)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__UCOP[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__UFLP	ADC sampled value of floating phase voltage (second-highest bit alignment)

16.3.24 TIM1__URES (0x4090, 0x4091)

TIM1__URESH(0x4090)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__URES[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__URESL(0x4091)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__URES[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__URES	Result of ADC Position Detection formula; Q15 format

16.3.25 TIM1__KF (0x4094, 0x4095)

TIM1__KFH(0x4094)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__KF[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__KFL(0x4095)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__KF[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__KF	ADC Position Detection Coefficient when floating phase voltage drops. Range [0,32767]

16.3.26 TIM1__KR (0x4096, 0x4097)

TIM1__KRH(0x4096)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__KR[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1__KRL(0x4097)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__KR[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__KR	ADC Position Detection Coefficient when floating phase voltage rises Range [0,32767]

16.3.27 TIM1__UIGN (0x4092, 0x4093)

TIM1__UIGNH(0x4092)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__UIGN[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__UFLPL(0x4093)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__UIGN[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__UIGN	When ADC sampled voltage of active phase is lower than TIM1__UIGN, the voltage is ignored.

16.3.28 TIM1__ITRIP (0x4098, 0x4099)

TIM1__ITRIPH(0x4098)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIM1__ITRIP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
TIM1__ITRIPL(0x4099)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIM1__ITRIP[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]	TIM1__ITRIP	Filtered Bus Current The hardware automatically samples the bus current and filters it for software application. The default channel is ADC channel 4. Range [0,32767]

17 Timer2

17.1 Timer2 Instructions

Timer2 has the following five working modes:

- Output mode: PWM generation
- Input capture mode: Detect duration of high and low level of input PWM to calculate duty cycle
- Input counter mode: Detect input time of the set PWM wave numbers
- QEP & RSD mode: Quadrature Encoder Pulse & Rotating State Detection (tailwind/headwind detection) mode
- Step Mode: Detect rotation direction, position and speed of step motor.

Timer2 features:

- 3-bit programmable prescaler divides the system clock
- 16-bit up-counting Base Timer; Counting clock source serves as the output of prescaler
- 16-bit up/down-counting special timer for Input Count Mode, QEP&RSD Mode and Step Mode, with external input signal selected as clock source.
- Input filter module
- Edge detection module
- PWM generation module
- Interrupt event

17.1.1 Prescaler

Prescaler divides the system clock frequency and generates clock source for Base Timer. 8 frequency division coefficients of prescaler are available and can be selected by TIM2_CR0[T2PSC]. Since this register has no buffer, the clock source frequency is updated immediately after TIM2_CR0[T2PSC] is written. Therefore, the frequency division coefficients shall be configured when Basic Timer is not working. The clock source frequency formula is: clk_psc2 = SYSCLK/(2^TIM2_CR0[T2PSC]). The clock rate corresponding to different TIM2_CR0[T2PSC] value as shown in Table 17-1.

Table 17-1 Mapping between Clock Rate and TIM2_CR0[T2PSC]

TIM2_CR0[T2PSC]	Division Factor	clk_psc2(Hz)	TIM2_CR0[T2PSC]	Division Factor	clk_psc2(Hz)
000	1	24M	100	16	1.5M
001	2	12M	101	32	750k
010	4	6M	110	64	375k
011	8	3M	111	128	187.5k

17.1.2 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 Base Timer shall be disabled before the write operation. When

reading TIM2__CNTR, the software reads the high bytes first, and the hardware synchronously caches the low bytes. When reading the low bytes, the software reads the cached data.

17.1.3 Output Mode

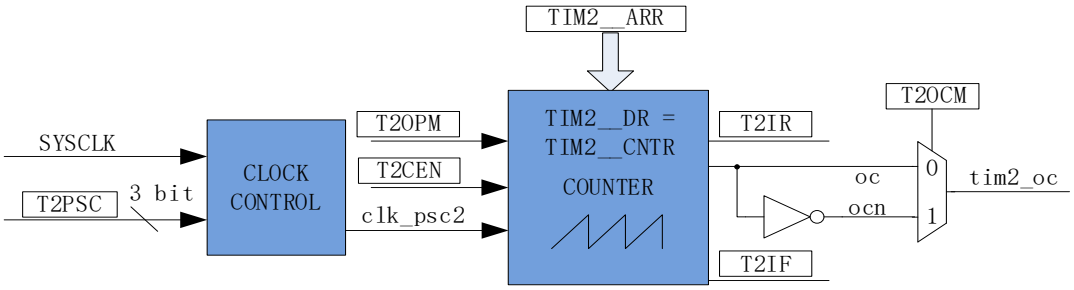


Figure 17-1 Output Mode Block Diagram

The output mode 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.

17.1.3.1 Reading and Writing of TIM2__ARR/TIM2__DR

In output mode, TIM2__ARR/TIM2__DR contains preload registers and shadow registers. When the software writes TIM2__ARR/TIM2__DR register, the data is saved in the preload register. When the overflow event TIM2_CR1[T2IF] is generated or the Base Timer stops working (TIM2_CR1[T2CEN] = 0), the set value is transferred to the shadow register.

TIM2__ARR/TIM2__DR is a 16-bit register, which requires to write the high bytes first and then the low bytes. The hardware ensures that the data in the preload register is not transferred to the shadow register after the high bytes are written or before the low bytes are written.

For example, TIM2__DR is a preload register and DR_SH is a shadow register. PWM is generated by comparing TIM2__CNTR with DR_SH. When software writes TIM2__DR, TIM2__DR is not updated to DR_SH immediately, and is updated to TIM2__DR at the end of a PWM (TIM2__CNTR overflow event).

17.1.3.2 High-/Low-level Output Mode

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

Note: The output signal is always high/low only when TIM2__DR = TIM2__ARR. Configuring TIM2__DR = 0 generates a pulse of one clock cycle.

17.1.3.3 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 is increased to TIM2__ARR, the output signal is reversed.

17.1.3.4 Interrupt Event

- 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.
- 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 then restarts.

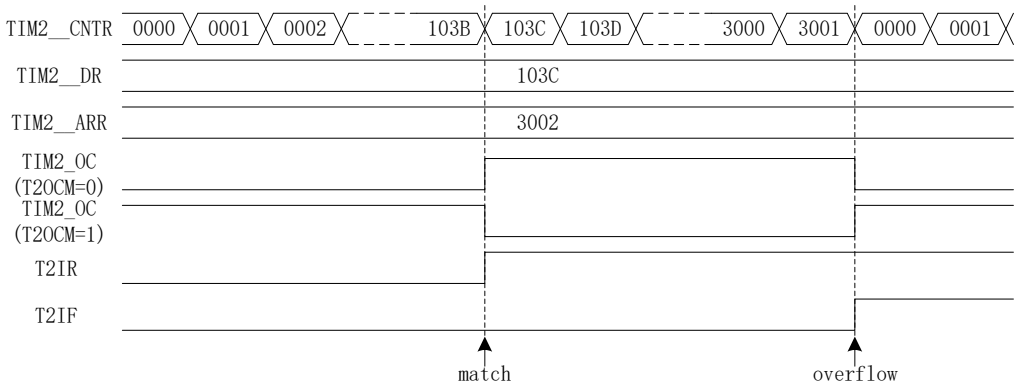


Figure 17-2 Output Mode Waveform

17.1.4 Input Signal Filtering and Edge Detection

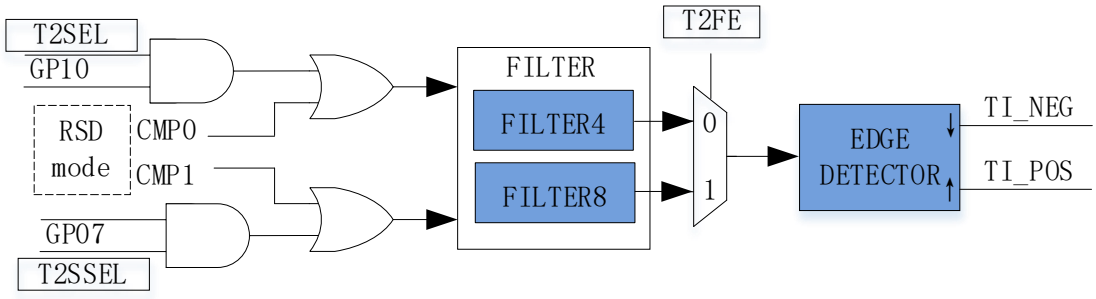


Figure 17-3 Block Diagram of Input Signal Filtering and Edge Detection

The input signal of Timer2 comes from P0.7 or P1.0, set by PH_SEL[T2SEL] and PH_SEL [T2SSEL] (see section 23.3.14). The filter of input signal is optional.

The filtering circuit filters out the input noise below 4 system cycles. The filtering period is selected by setting TIM2_CR1[T2FE]. When TIM2_CR1[T2FE] is set to “0”, the filtering circuit is disabled; and when TIM2_CR1[T2FE] is set to “1”, the filtering circuit filters signals every 4 system cycles. The filtered signal

is about 4 clock cycles later than the signal before filtering. TIM2_CR0[T2CES] determines the active edge to count.

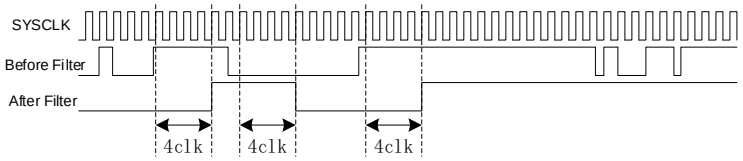


Figure 17-4 Timing Diagram of Filter Module

The edge detection module detects filtered input signals and records rising edge and falling edge for input capture mode or input counting mode.

17.1.5 Input Capture Mode

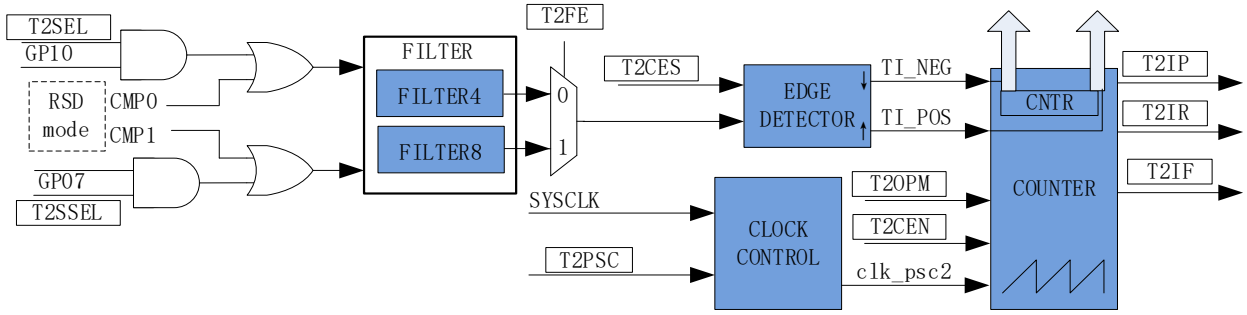


Figure 17-5 Schematic Diagram of Input Capture Mode

The input capture mode detects duty cycle and period of the PWM signal. When TIM_CR0[T2CES] = 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 TIM_CR0[T2CES] = 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). When the predefined edge arrives, the count value TIM2__CNTR is stored in TIM2__DR and TIM2__ARR respectively to calculate the period and duty cycle of PWM waveform.

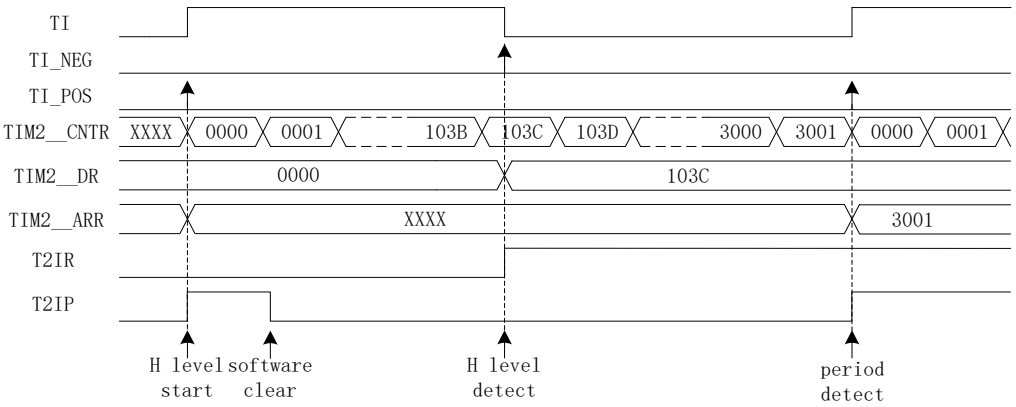


Figure 17-6 Timing Diagram of Input Capture Mode (TIM2_CR0[T2CES] = 0)

For example, when TIM2_CR0[T2CES] = 0, TIM2_CR1[T2CEN] is set to "1" to enable the Base Timer. When the first rising edge of the input (falling edge is invalid) is detected, TIM2__CNTR is cleared and restarts. When falling edge of the input is detected, the value of TIM2__CNTR is stored in TIM2__DR, while the interrupt flag TIM2_CR1[T2IR] is set to "1", and TIM2__CNTR continues to count. When the second rising edge of input is detected, the value of TIM2__CNTR is stored in TIM2__ARR. Meanwhile, the interrupt flag TIM2_CR1[T2IP] is set to "1", and TIM2__CNTR is cleared to "0" and 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 TIM2_CR1[T2IF] is set to "1", and TIM2__CNTR is cleared to "0" and restarts. At this point, TIM2__ARR value is 0xFFFF.

17.1.6 Input Counter Mode

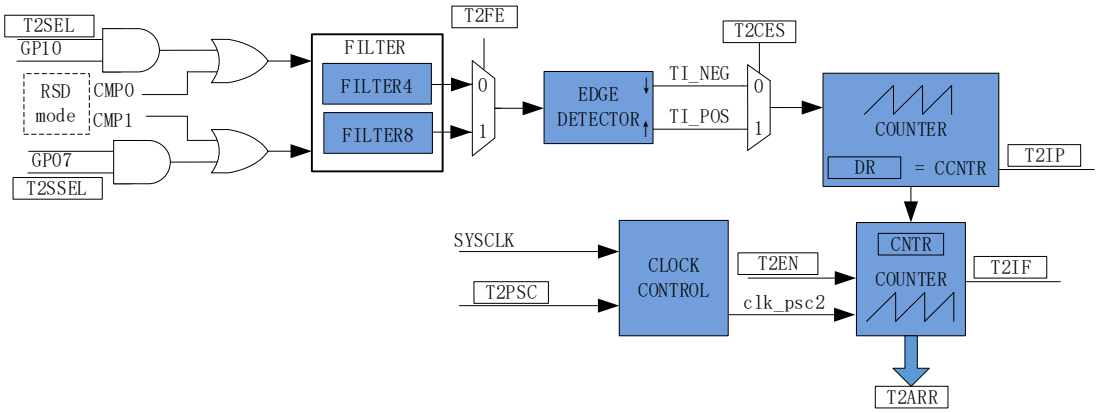


Figure 17-7 Schematic Diagram of Input Counter Mode

In input counter mode, TIM2__DR includes preload register and shadow register. When the software writes TIM2__DR register, the data is saved in the preload register first, and then sent to the shadow register in case of compare match event (TIM2_CR1[T2IP] = 1), overflow event (TIM2_CR1[T2IF] = 1) or special

timer disable (TIM2_CR1[T2CEN] = 0). TIM2__DR is a 16-bit register, which requires the software writes the high bytes first and then the low bytes. The hardware ensures that the data in the preload register is not updated to the shadow register after the high bytes are written and before the low bytes are written.

The input counter mode is used to detect the time to input the set PWM wave. When the number of input PWM counted by the special timer CCNTR reaches the set value (TIM2_DR), TIM2__CNTR of the Base Timer is stored in TIM2__ARR. When TIM2_CR0[T2CES] is set to “1”, the rising edge of the input PWM signal serves as the active counting edge of the special timer; when TIM2_CR0[T2CES] is set to “0”, the falling edge of the input signal as the active edge.

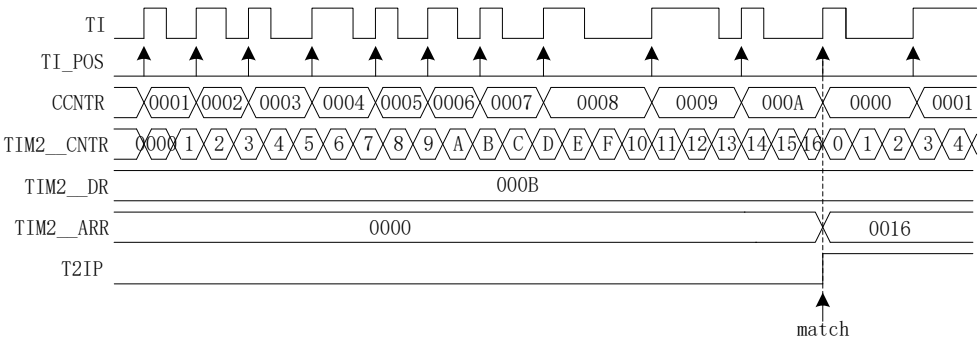


Figure 17-8 Timing Diagram of Input Counter Mode

The Basic Timer is enabled when TIM2_CR1[T2CEN] is set to “1”. If the first active edge of the input signal is detected, TIM2__CNTR is cleared to “0” and restarts. Whenever active edge of the input signal arrives, one is added to the count value of the special timer CCNTR. When the count value reaches TIM2__DR, TIM2__CNTR is stored in TIM2__ARR. When TIM2_CR1[T2IP] is set to “1”, TIM2__CNTR and CCNTR are cleared to “0” and restart.

When the number of input PWM does not reach the set value and TIM2__CNTR reaches 0xFFFF, an overflow event generates, and the interrupt flag TIM2_CR1[T2IF] is set to “1”. TIM2__CNTR is cleared to “0” with CCNTR uncleared. TIM2__CNTR starts counting from 0, and CCNTR continues counting with the previous value.

17.1.7 QEP&RSD Mode

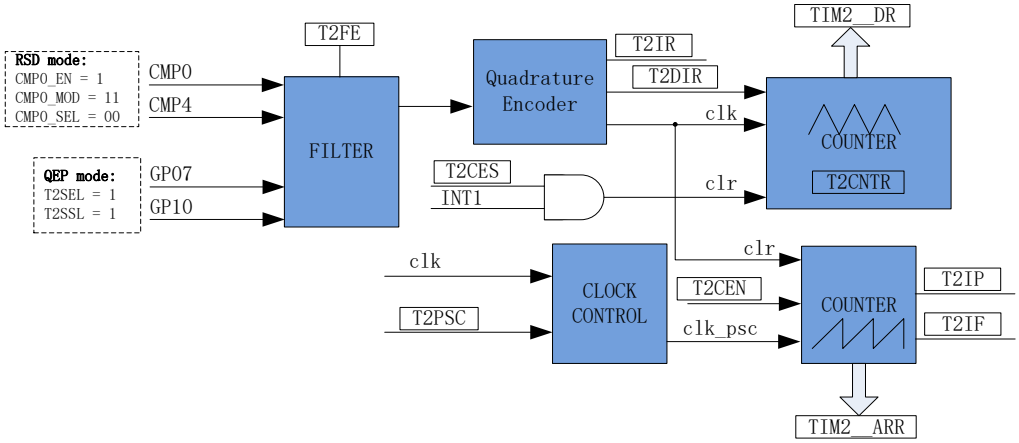


Figure 17-9 Schematic Diagram of QEP&RSD Mode

QEP & RSD mode obtains relative position, direction and speed of the motor by detecting orthogonal signals on two channels. P0.7 and P1.0 (QEP mode) or CMP0, CMP4 (RSD mode) are the input signal sources, which are sent to the quadrature decoding module from the filtering module to obtain active edge and direction (TIM2_CR1[T2DIR]). TIM2_CR1[T2IF] interrupt flag is generated when the direction changes.

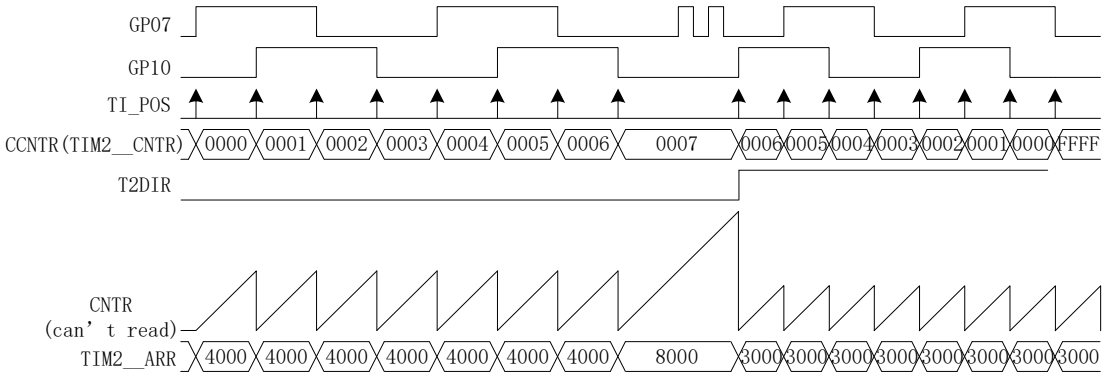


Figure 17-10 Timing Diagram of QEP&RSD Mode

The special timer is an up/down timer, and the signal source is the active edge from orthogonal decoding module. If TIM2_CR1[T2DIR] = 0, the direction is positive, and special timer counts upward. When the active edge arrives, the timer increases by one. If TIM2_CR1[T2DIR] = 1, the direction is reverse and special timer counts down. When the active edge arrives, the timer decreases by one. The special timer can be cleared by external interrupt INT1 and the value held by the timer is stored in TIM2__DR, after mechanical zero of the encoder is connected with any port of INT1, INT1 interrupt is enabled and TIM2_CR0[T2CES] = 1. If count value of the special timer reaches 65535 from 0, it is automatically cleared to "0". If it decreases from 65535 to 0, it is automatically set to 65535. TIM2__CNTR is read to obtain the value of special timer.

The Base Timer is an up counter, which uses the output of prescaler as the clock source to record the time between two active counting edges. When active counting edge arrives, the value of Basic Timer is stored in TIM2__ARR and then cleared to “0”, and TIM2_CR1[T2IP] interrupt flag bit is set to “1”. When Base Timer counts to 0xFFFF, the count overflows and (TIM2_CR1[T2IF]) interrupt flag is generated.

17.1.7.1 RSD Comparator Sampling

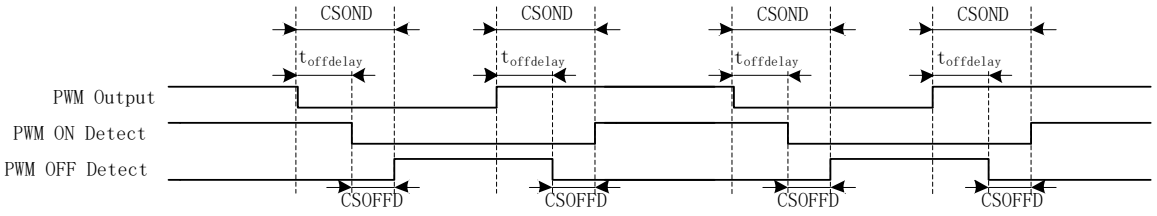


Figure 17-11 PWM Sampling Mode

The Start of Sampling (“SoS”) time delay and End of Sampling (“EoS”) time advance must be set in order to sample correct BEMF comparison signals in RSD sampling mode.

See section 30.1.5 for details.

17.1.8 Step Mode

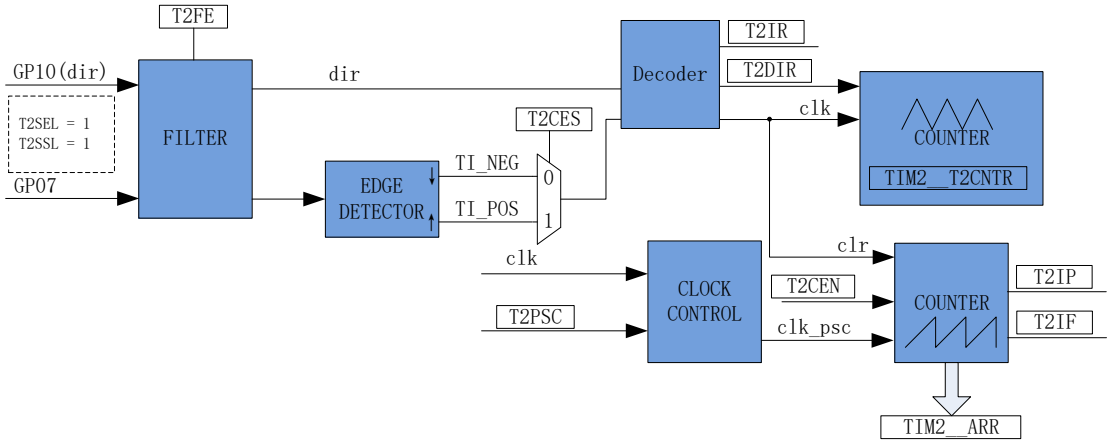


Figure 17-12 Schematic Diagram of Step Mode

In step mode, relative position, direction and speed of the step motor are obtained by detecting inputs of the two channel. P1.0 is direction input, and P0.7 is pulse input. Setting TIM2_CR0[T2CES] to select the rising edge or falling edge as the active edge. The input signals are sent to decoding module from the filtering module to obtain the active edge and direction TIM2_CR1[T2DIR]. TIM2_CR1[T2IF] interrupt flag is generated when the direction changes.

Note: TIM2_CR1[T2DIR] will not change unless transition occurs at P1.0 and active edge is detected at P0.7. To generate an interrupt immediately after P1.0 changes, use INT1.

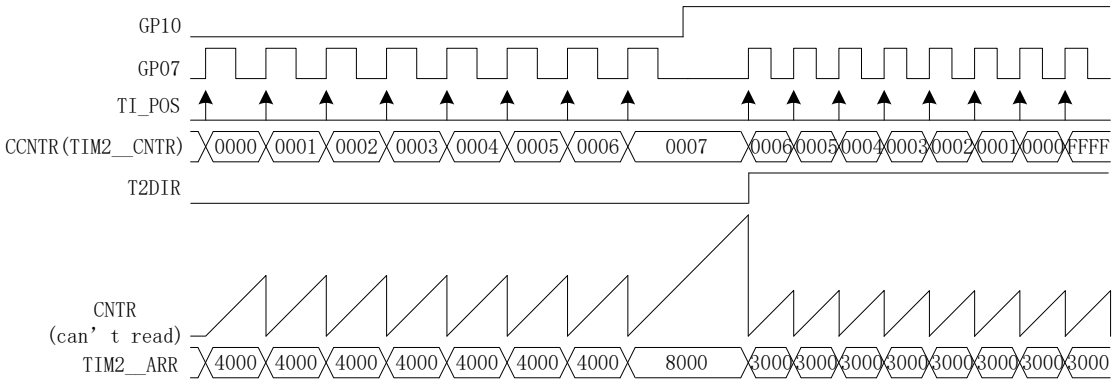


Figure 17-13 Timing Diagram of Step Mode

The special timer is an up/down-counter, and the signal source is active edge of the encoding module. When P1.0 = 0, TIM2_CR1[T2DIR] = 0, the direction is forward. If active edge of P0.7 arrives, the special timer increases by 1. When P1.0 = 1, TIM2_CR1[T2DIR] = 1 and the direction is reverse. If active edge of P0.7 arrives, it decreases by 1. If count value of the special timer reaches 65535 from 0, it is automatically cleared to "0". If it decreases from 65535 to 0, it is automatically reset to 65535. TIM2__CNTR is read to obtain the value of special timer.

The Base Timer is an up counter, which uses the output of prescaler as the clock source to record the time between two active counting edges. When active counting edge arrives, the value of Basic Timer is stored in TIM2__ARR and then cleared to "0", and TIM2_CR1[T2IP] interrupt flag bit is set to "1". When Base Timer counts to 0xFFFF, the count overflows and TIM2_CR1[T2IF] interrupt flag is generated..

17.2 Timer2 Registers

17.2.1 TIM2_CR0(0xA1)

Bit	7	6	5	4	3	2	1	0
Name	T2PSC			T2OCM	T2IRE	T2CES	T2MOD	
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
[7:5]	T2PSC	Base Timer Clock Prescaler Selection It is configured to divide the system clock frequency and generate the clock source for Base Timer. The prescaled clock rates are configured as follows: 000: 24MHz 001: 12MHz 010: 6MHz 011: 3MHz 100: 1.5MHz 101: 750kHz 110: 375kHz 111: 187.5kHz
[4]	T2OCM	Output Mode: Output Mode Selection 0:“0” is output when TIM2__CNTR < TIM2__DR, and “1” is output when TIM2__CNTR ≥ TIM2__DR 1: “1” is output when TIM2__CNTR < TIM2__DR, and “0” is output when TIM2__CNTR ≥ TIM2__DR Input Count Mode: No effect Input Capture Mode: No effect QEP&RSD Mode and Step Mode Selection 0: QEP&RSD Mode 1: Step Mode
[3]	T2IRE	Output Mode: Match Interrupt Enable Input Capture Mode: Pulse Width Detection Interrupt Enable Input Count Mode: No effect QEP&RSD Mode and Step Mode: Direction Change Interrupt Enable 0: Disable 1: Enable
[2]	T2CES	Output Mode: No effect Input Capture Mode: Counting 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). Input Counter Mode and Step Mode: Active Edge Selection 0: Falling Edge Count 1: Raising Edge Count QEP&RSD Mode: Pulse Counter Cleared Enable upon Interrupt INT1 (Z Signal) 0: Disable 1: Enable
[1:0]	T2MOD	Mode Selection 00: Input Capture Mode 01: Output Mode 10: Input Counter Mode 11: QEP&RSD Mode or Step Mode

17.2.2 TIM2_CR1 (0xA9)

Bit	7	6	5	4	3	2	1	0
Name	T2IR	T2IP	T2IF	T2IPE	T2IFE	T2FE	T2DIR	T2CEN
Type	R/W0	R/W0	R/W0	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T2IR	Output Mode: Match Interrupt Flag The bit is set to “1” by the hardware when TIM2__CNTR matches TIM2__DR. It is cleared to “0” by software. Input Capture Mode: Pulse Width Detection Interrupt Flag This bit is set to “1” by hardware when an input pulse width is detected (Setting TIM2_CR0[T2CES] to select the rising edge or falling edge as the active edge). It is cleared to “0” by software. Input Counter Mode: No effect QEP&RSD Mode or Step Mode: Direction Change Flag Bit 0: No Interrupt Pending 1: Interrupt Pending
[6]	T2IP	Output Mode: No effect Input Capture Mode: PWM Cycle Detection Interrupt Flag This bit is set to “1” by hardware when an input PWM cycle is detected (Setting TIM2_CR0[T2CES] to select the rising edge or falling edge as the active edge). It is cleared to “0” by software. Input Counter Mode: PWM Input Counter Match Interrupt Flag This bit is set to “1” by hardware when the number of input PWM reaches TIM2__DR. It is cleared to “0” by software. QEP&RSD Mode or Step Mode: Active Edge Detection Interrupt Flag This bit is set to “1” by hardware when the input edge is detected as an active edge. It is cleared to “0” by software. 0: No Interrupt Pending 1: Interrupt Pending
[5]	T2IF	Output Mode: Timer Overflow Interrupt Flag. This bit is set to “1” by hardware and TIM2__CNTR is cleared to “0” when TIM2__CNTR matches TIM2__ARR. It is cleared to “0” by software. Input Capture Mode: Timer Overflow Interrupt Flag Setting TIM2_CR0[T2CES] to select the rising edge or falling edge as the active edge. This bit is set to “1” by hardware and TIM2__CNTR is cleared to “0” when the Timer has not detected an input PWM cycle but the timer TIM2__CNTR reaches 0xFFFF (overflow occurs). It is cleared to “0” by software. Input Counter Mode: Base Timer Overflow Interrupt Flag This bit is set to “1” by hardware and TIM2__CNTR is cleared to “0” when the input PWM cycle has not reached the preset TIM2__DR value but Base Timer TIM2__CNTR value reaches 0xFFFF (overflow occurs). It is cleared to “0” by software. QEP&RSD Mode or Step Mode: Base Timer Overflow Interrupt Flag This bit is set to “1” by hardware and Basic Timer is cleared to “0” when Basic Timer reaches to 0xFFFF (overflow occurs). It is cleared to “0” by software. 0: No Interrupt Pending 1: Interrupt Pending
[4]	T2IPE	Output Mode: No effect Input Capture Mode: PWM Cycle Detection Interrupt Enable Input Counter Mode: PWM Input Counter Match Interrupt Enable QEP&RSD Mode and Step Mode: Active Edge Detection Interrupt Enable 0: Disable 1: Enable
[3]	T2IFE	Output Mode: Base Timer Overflow Interrupt Enable Input Capture Mode: Base Timer Overflow Interrupt Enable

		Input Counter Mode: Base Timer Overflow Interrupt Enable QEP&RSD Mode or Step Mode: Base Timer Overflow Interrupt Enable 0: Disable 1: Enable
[2]	T2FE	Input Signal Filter Selection When input signals are filtered out as noise if the pulse width is less than 4 clock cycle. Assuming that MCU clock runs at 24MHz (41.67ns) , then the pulse width for filtering is 166.67ns. 0: Disable 1: Enable
[1]	T2DIR	QEP&RSD Mode or Step Mode: Indicator of Motor Rotation Direction 0: Forward 1: Backward
[0]	T2CEN	Timer Enable 0: Disable 1: Enable

17.2.3 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	Output Mode/Input Capture Mode/Input Count Mode: Count values held in the Base Timer QEP&RSD Mode or Step Mode: count values held in the special timer

17.2.4 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 value (written by software) Input Capture Mode: Count value of the detected input pulse width based on TIM2_CR0[T2CES] (written by hardware) Input Counter Mode: PWM cycles to be counted (written by software)

		QEP&RSD Mode: Value of the special timer when TIM2_CR0[T2CES] = 1 and INT1 (zero point) is detected (written by hardware) Step Mode: No effect

17.2.5 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: Overload value (written by software) Input Capture Mode: Count value of a PWM cycle based on TIM2_CR0[T2CES] (written by hardware) Input Counter Mode: Count value held in Base Timer when the input PWM count matches (written by hardware) QEP&RSD Mode or Step Mode: Count value held in Base Timer when the input signal is detected as an active edge (written by hardware)

18 Timer3/Timer4

18.1 Timer3/Timer4 Instructions

Timer3/Timer4 support output and input modes:

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

Timer3/Timer4 Features:

- 3-bit programmable prescaler divides system clock as the clock source for Base Timer
- 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

18.1.1 Prescaler

Prescaler divides the system clock frequency and generates counter clock source for Base Timer. 8 frequency division coefficients of prescaler are available and can be selected by TIMx_CR0[TxPSC]. Since this register has no buffer, the clock source frequency is updated immediately after TIMx_CR0[TxPSC] is written. Therefore, the frequency division coefficients shall be configured when Basic Timer is not working. The clock source frequency formula is: clk_pscx = SYSCLK/(2^TxPSC). The clock rate corresponding to different TIMx_CR0[TxPSC] value as shown in Table 18-1.

Table 18-1 Mapping between Clock Rate and TIMx_CR0[TxPSC]

TIMx_CR0[TxPSC]	Division Factor	clk_pscx(Hz)	TIMx_CR0[TxPSC]	Division Factor	clk_pscx(Hz)
000	0x1	24M	100	0x10	1.5M
001	0x2	12M	101	0x20	750k
010	0x4	6M	110	0x40	375k
011	0x8	3M	111	0x80	187.5k

18.1.2 Reading, Writing and Counting of TIMx_CNTR

TIMx__CNTR starts when TIMx_CR1[TxEN] = 1. The write operation to TIMx__CNTR directly changes the value of the register, so it is required to disable the timer before the write operation. When reading TIMx__CNTR, the software reads high bytes first and then low bytes, and the hardware caches the low bytes simultaneously. When reading the low bytes, the software reads the cached data.

18.1.3 Output Mode

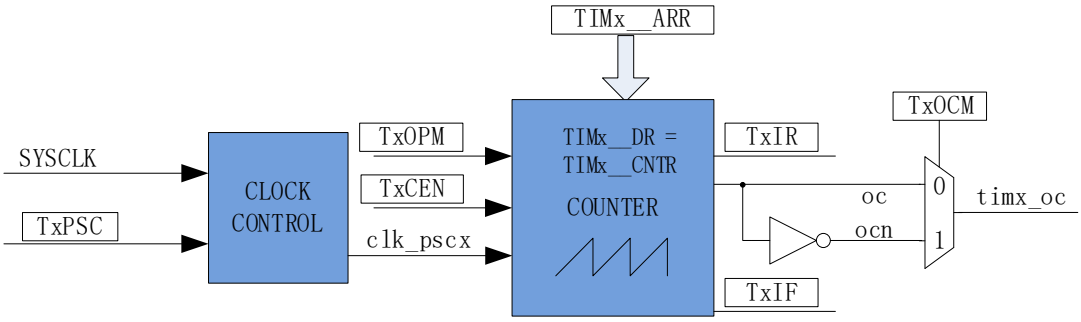


Figure 18-1 Output Mode Block Diagram

The output mode generate output signals according to TIMx_ CR0[TxOCM], and the comparison results between TIMx__CNTR and registers TIMx__DR, TIMx__ARR. Meanwhile, corresponding interrupts are generated.

18.1.3.1 High-/Low-level Output Mode

When TIMx_CR0[TxOCM] = 0 and TIMx__DR > TIMx__ARR, the output signals are always low. When TIMx_CR0[TxOCM] = 1 and TIMx__DR > TIMx__ARR, the output signals are always high.

18.1.3.2 PWM Generation

In PWM generation mode, TIMx__ARR determines PWM cycle, and TIMx__DR determines the duty cycle, and duty cycle = TIMx__DR/TIMx__ARR*100%. If TIMx_CR0[TxOCM] = 0, the low level is output when TIMx__CNTR < TIMx__DR, and the high level is output when TIMx__CNTR > TIMx__DR. If TIMx_CR0[TxOCM] = 1, the high level is output when TIMx__CNTR < TIMx__DR, and low level is output when TIMx__CNTR > TIMx__DR. When TIMx__CNTR > TIMx_ARR, the output signal is reversed.

18.1.3.3 Interrupt Event

- When TIMx__CNTR = TIMx__DR, a compare match interrupt is generated. The interrupt flag TIMx_CR1[TxIR] is set to “1”, and the timer continues.
- When TIMx__CNTR = TIMx__ARR, an overflow event is generated. The interrupt flag TIMx_CR1[TxIF] is set to “1”, and the timer is cleared to “0”. TIMx_CR0[TxOPM] determines whether the timer recounts. The timer stops when TIMx_CR0[TxOPM]= 1, and restarts when TIMx_CR0[TxOPM]= 0.

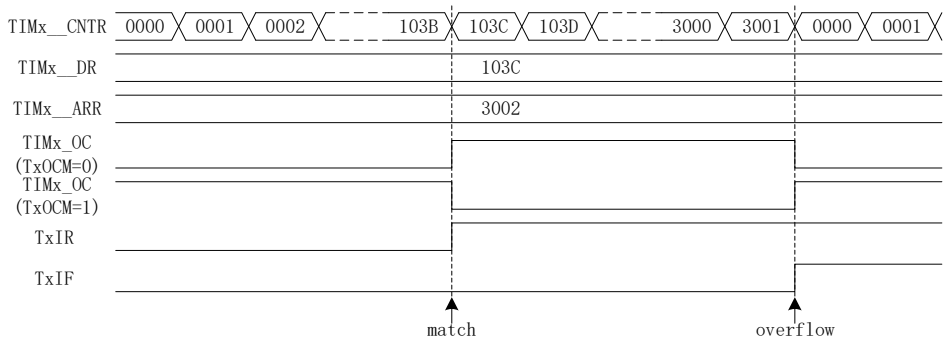


Figure 18-2 Output Waveform of Output Mode

18.1.4 Input Signal Filtering and Edge Detection

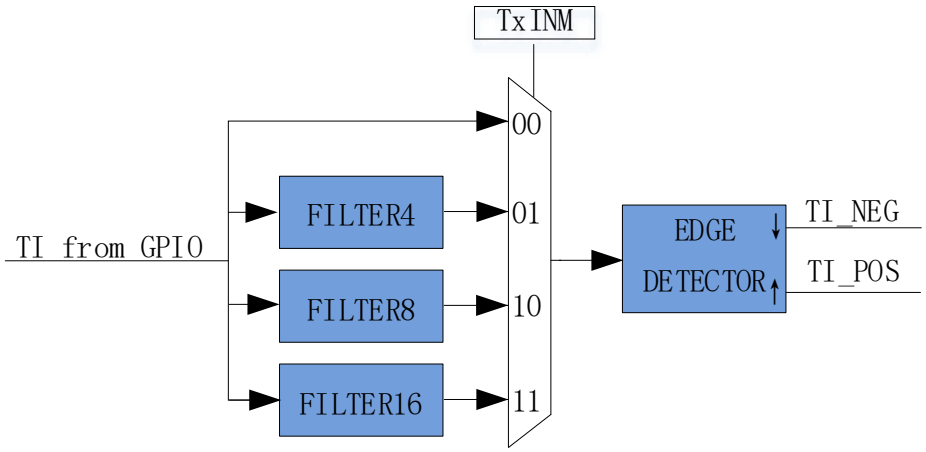


Figure 18-3 Block Diagram of Input Signal Filtering and Edge Detection

The input signals of Timer3/Timer4 come from GPIO pin. TIMx_CR1[TxINM] 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.

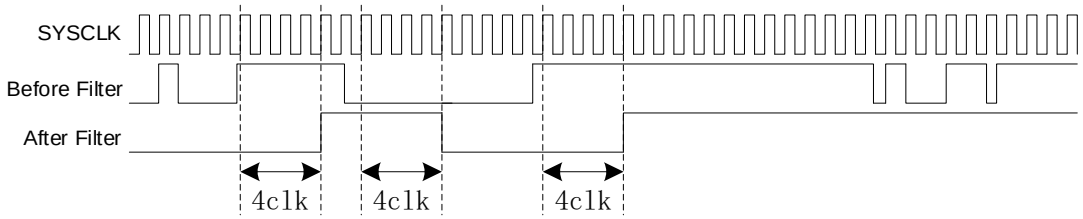


Figure 18-4 Timing Diagram of Filter Module

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

18.1.5 Input Capture Mode

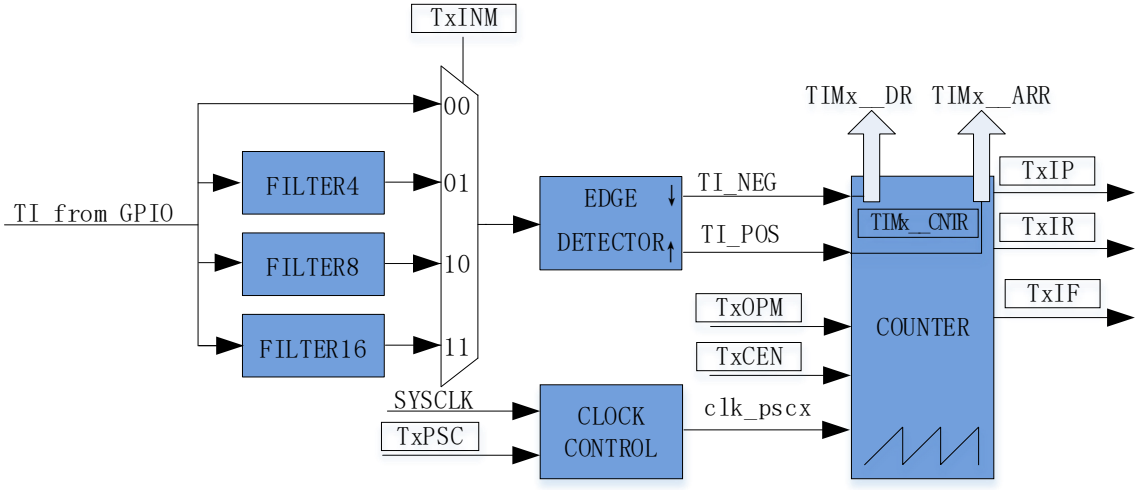


Figure 18-5 Schematic Diagram of Input Capture Mode

The Input Capture Mode detects pulse width and waveform period of the input PWM signals. When TIMx_CR0[TxOCM] = 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 TIMx_CR0[TxOCM] = 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 TIMx__CNTR are stored in TIMx__DR and TIMx__ARR respectively.

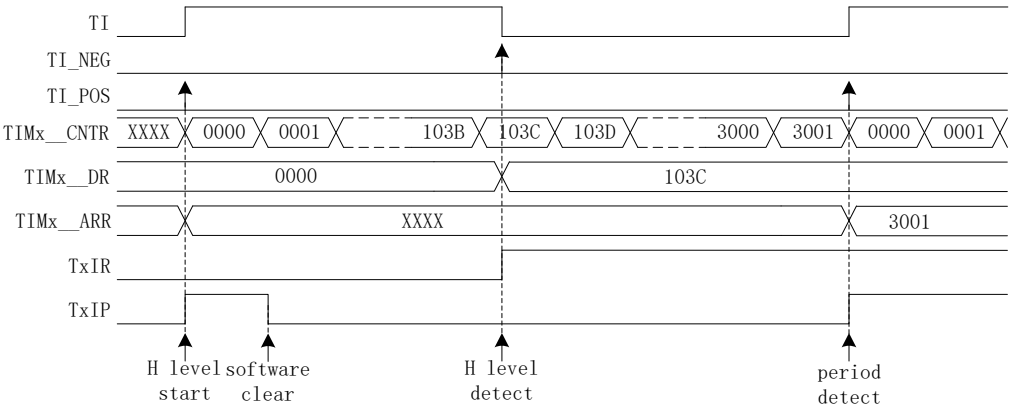


Figure 18-6 Timing Diagram of Input Capture Mode (TIMx_CR0[TxOCM] = 0)

For example, when TIMx_CR0[TxOCM]= 0, TIMx_CR1[TxEN] is set to “1” to enable the timer. The Base Timer is cleared to “0” and restarts when the first raising edge is detected. When the falling edge is detected, the value of TIMx__CNTR is stored into TIMx__DR. Meanwhile, the interrupt flag TIMx_CR1[TxIR] is set to “1”, and TIMx__CNTR continues to count. When the second rising edge is detected, the value of TIMx__CNTR is saved into TIMx__ARR. The interrupt flag TIMx_CR1[TxIP] is set

to “1” and TIMx__CNTR is cleared to “0”. TIMx_CR0[TxOPM] determines whether the timer restarts. If TIMx_CR0[TxOPM] = 1, the timer stops; and if TIMx_CR0[TxOPM] = 0, it restarts.

An overflow event occurs if Timer3/Timer4 does not detect the second rising edge of the input and TIMx__CNTR reaches 0xFFFF. In this case, the interrupt flag bit TIMx_CR1[TxIF] is set to “1”, and TIMx__CNTR is cleared to “0”. TIMx_CR0[TxOPM] determines whether the timer restarts. If TIMx_CR0[TxOCM]= 1, the timer stops counting, and if TIMx_CR0[TxOPM] = 0, it restarts. At this point, TIMx__ARR is 0xFFFF, and TIMx__DR is determined by the input level and TIMx_CR0[TxOCM] XOR.

18.2 Timer3/Timer4 Registers

18.2.1 TIMx_CR0 (0x9C/0x9E) (x=3/4)

Bit	7	6	5	4	3	2	1	0
Name	TxPSC			TxOCM	TxIRE	RSV	TxOPM	TxMOD
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
[7:5]	TxPSC	Base Timer Clock Prescaler Selection It is configured to divide the system clock frequency and generate the clock source for Base Timer. The prescaled clock rates are configured as follows: 000: 24MHz 001: 12MHz 010: 6MHz 011: 3MHz 100: 1.5MHz 101: 750kHz 110: 375kHz 111: 187.5kHz
[4]	TxOCM	Output Mode: Output Mode Selection 0: “0” is output when TIMx__CNTR<TIMx__DR, and “1” is output when TIMx__CNTR>TIMx__DR. 1: “1” is output when TIMx__CNTR<TIMx__DR, and “0” is output when TIMx__CNTR>TIMx__DR. Input Capture Mode: TIMx__DR indicates the input level to be selected when the active edge is detected or the timer becomes overflowed. 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). TIMx__DR indicates the input level to be selected when the timer becomes overflowed.
[3]	TxIRE	Output Mode: Compare Match Interrupt Enable Input Capture Mode: Pulse Width Detection Interrupt Enable 0: Disable 1: Enable
[2]	RSV	Reserved
[1]	TxOPM	Single Mode Base Timer stops in any of the following events: Output Mode: Base Timer overflow event Input Capture Mode: PWM Cycle Detection or Base Timer overflow event 0: Base Timer does not stop 1: Base Timer stops (TIMx_CR1[TxEN] is reset to “0”)
[0]	TxMOD	Working Mode Selection 0: Input Capture Mode 1: Output Mode

18.2.2 TIMx_CR1 (0x9D/0x9F) (x=3/4)

Bit	7	6	5	4	3	2	1	0
Name	TxIR	TxIP	TxIF	TxIPE	TxIFE	TxINM		TxEN
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]	TxIR	Output Mode: Compare Match Interrupt Flag Input Capture Mode: Pulse Width Detection Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	TxIP	Output Mode: No effect Input Capture Mode: PWM Cycle Detection Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	TxIF	Output Mode: Base Timer Overflow Interrupt Flag, which is set to”1” when TIMx__CNTR matches TIMx__ARR. Input Capture Mode: Base Timer Overflow Interrupt Flag, which is set to “1” when the Timer does not detect an input PWM cycle but TIMx__CNTR reaches 0xFFFF. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	TxIPE	Output Mode: No effect Input Capture Mode: PWM Cycle Detection Interrupt Enable 0: Disable 1: Enable
[3]	TxIFE	Output Mode: Base Timer Overflow Interrupt Enable Input Capture Mode: Base Timer Overflow Interrupt Enable 0: Disable 1: Enable
[2:1]	TxINM	Input Signal Filtering Pulse Width Selection Input signals are filtered as noise if pulse width is less than the defined value. 00: Disable 01: Filtered on every 4 SYSCLK cycles 10: Filtered on every 8 SYSCLK cycles 11: Filtered on every 16 SYSCLK cycles
[0]	TxEN	Base Timer Enable 0: Disable 1: Enable

18.2.3 TIMx__CNTR (0xA2, 0xA3/0x92, 0x93) (x = 3/4)

TIMx__CNTRH(0xA3/0x93)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIMx__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
TIMx__CNTRL(0xA2/0x92)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIMx__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]	TIMx__CNTR	Count values held in Base Timer

18.2.4 TIMx__DR(0xA4, 0xA5/0x94,0x95) (x=3/4)

TIMx__DRH(0xA5/0x95)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIMx__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
TIMx__DRL(0xA4/0x94)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIMx__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]	TIMx__DR	Output Mode: Compare match values (written by software) Input Capture Mode: Count value of the detected input pulse width (written by hardware)

18.2.5 TIMx__ARR (0xA6, 0xA7/0x96, 0x97) (x = 3/4)

TIMx__ARRH(0xA7/0x97)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	TIMx__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
TIMx__ARRL(0xA6/0x96)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	TIMx__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]	TIMx__ARR	Output Mode: Reload value (written by software). Input Capture Mode: Count value of a detected PWM cycle (written by hardware)

19 Systick

19.1 Systick Instructions

The chip can generate Systick interrupts at a fixed interval, 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.

19.2 Systick Registers

19.2.1 DRV_SR (0x4061)

Bit	7	6	5	4	3	2	1	0
Name	SYSTIF	SYSTIE	FGIF	DCIF	FGIE	DCIP	DCIM	
Type	R/W0	R/W	R/W0	R/W0	R/W	R/W	R/W	R/W
Reset	0	0	0	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]	FGIF	FG Interrupt Flag When FOC Drive/Square Wave Drive is enabled, an FGIF Interrupt is generated in each rotation cycle (electrical cycle). Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	DCIF	Driver Compare Match Interrupt Flag When the Driver timer 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]	FGIE	FG Interrupt Enable When FOC Drive/Square Wave Drive is enabled, an FG Interrupt is generated in each rotation cycle (electrical cycle). 0: Disable 1: Enable
[2]	DCIP	Number of PWM cycles required to generate a Driver Compare Match Interrupt 0: 1 interrupt in 1 PWM cycle 1: 1 interrupt in 2 PWM cycles
[1:0]	DCIM	Driver Compare Match Interrupt Mode Selection The system decides whether to generate an interrupt according to DRV_SR[DCIM]

		when Driver counter value is equal to DRV_COMR value. 00: No interrupt is generated. 01: An interrupt is generated when the counter counts up 10: An interrupt is generated when the counter counts down 11: An interrupt is generated when the counter counts up/down

19.2.2 SYST_ARR (0x4064, 0x4065)

SYST_ARRH(0x4064)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	SYST_ARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	1	1	1	0	1
SYST_ARRL(0x4065)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	SYST_ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	1	1	1	1	1	1

Bit	**Name**	**Description**
[15:0]	SYST_ARR	Systick Reloaded Value This bit determines the cycle at which Systick interrupts are generated, which defaults to 1ms. Calculation formula is as follows: Systick interrupt rate = SYSCLK/(SYST_ARR[15:0] + 1) Range [0, 65535]

20 Driver

20.1 Driver Instructions

20.1.1 Driver Introduction

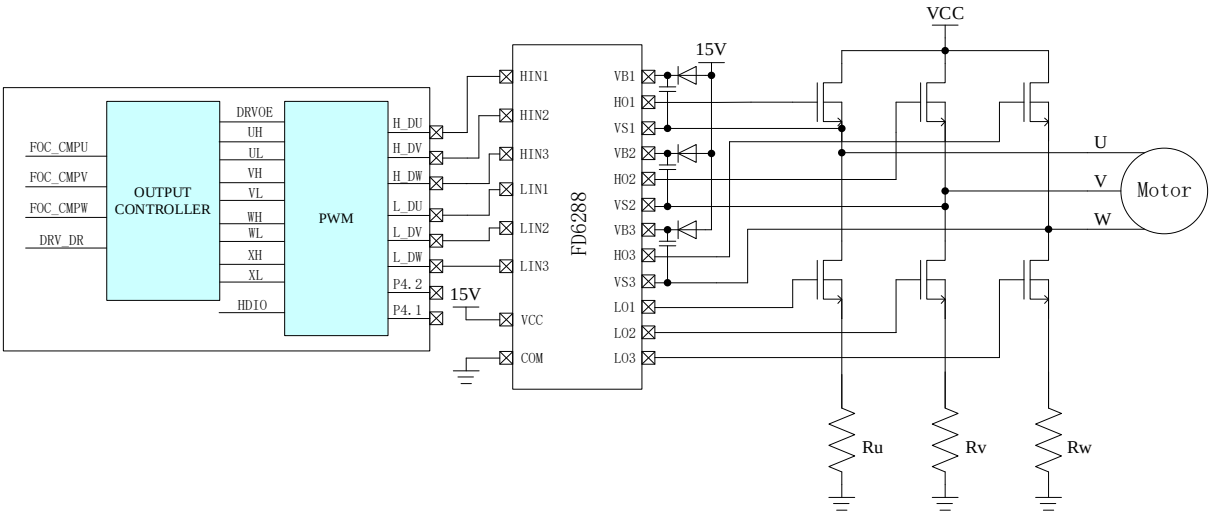


Figure 20-1 Block Diagram of FU6813 Driver Module

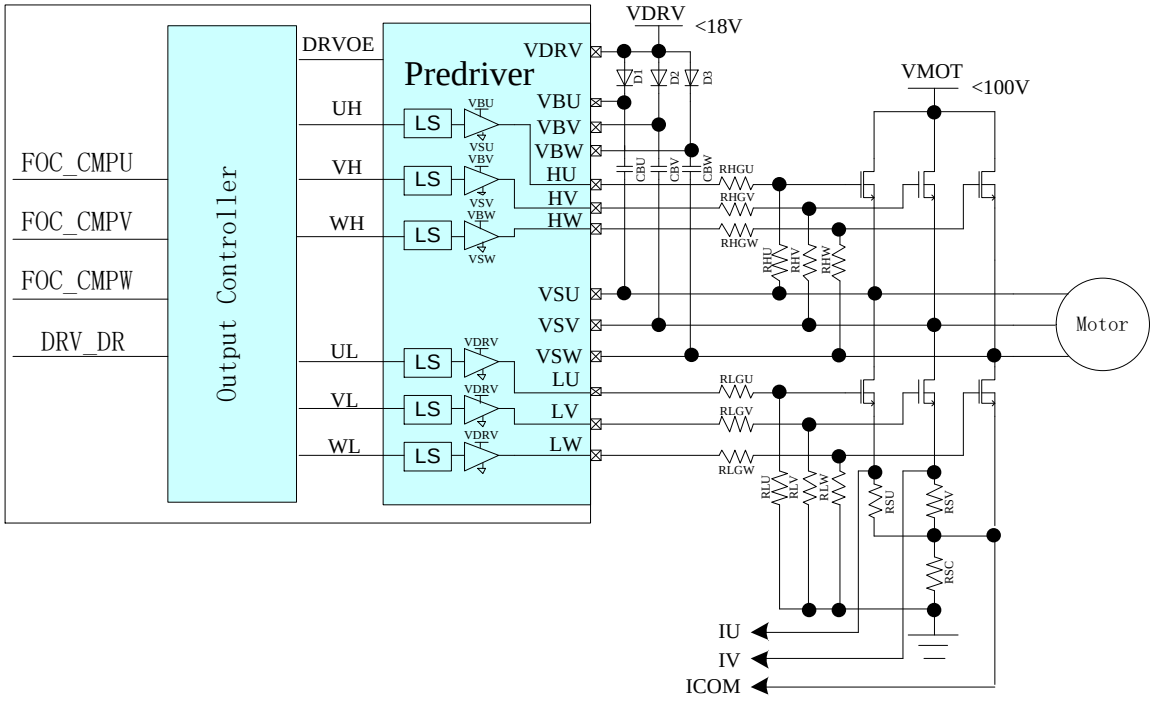


Figure 20-2 Block Diagram of FU6863 Driver Module

FOC_CMPU/V/W is the three-way comparison value output by FOC module, and DRV_ DR is the comparison value set by the software. The above comparison value outputs four sets of level signals U/V/W/X to PWM output (FU6813) or three sets of level signals U/V/W to pre-driver (FU6863) after passing

through the output control module. The U/V/W three-way output is applied to DC brushless motor control, and the U/V/W/X four-way output to step motor control.

20.1.2 Output Control Module

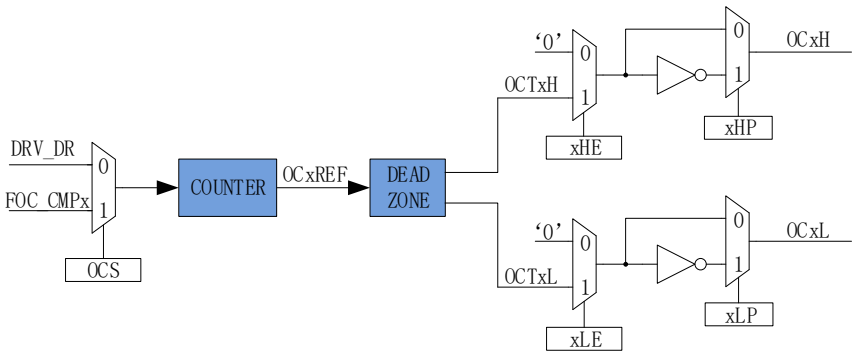


Figure 20-3 Block Diagram of Output Control Module

Before Driver module works, DRV_CR[MESEL] is set to "1" to select FOC/SVPWM/SPWM mode or to "0" to select square-wave control mode.

When DRV_CR[OCS] = 0, comparison value of PWM comes from DRV_DR, and OCTxH serves as the reference for output PWM signal. If OCxH and OCxL are output at the same time, OCTxL is output in reverse phase. When DRV_CR[OCS] = 1, comparison value of PWM comes from FOC module, and OCTxL serves as the reference for output PWM signal. If OCxH and OCxL are output at the same time, OCTxH is output in reverse phase.

20.1.2.1 Count and Compare Module

DRV_CR[OCS] is configured to select the comparison value of PWM from FOC_CMPU/V/W of FOC module or DRV_DR set by software. The comparison value is sent to the counter for comparison to obtain a 3-way PWM signal OCxREF, and DRV_ DR is used for motor pre-charging, braking and square-wave control. If DRV__CNTR is smaller than the comparison value, OCxREF outputs high-level signal, and if DRV__CNTR is larger than DRV_DR, OCxREF outputs low-level signal.

When DRV_CR[OCS] = 1, FOC_CMPU/V/W is compared with the count value to generate the duty cycle OC1REF/OC2REF/OC3REF.

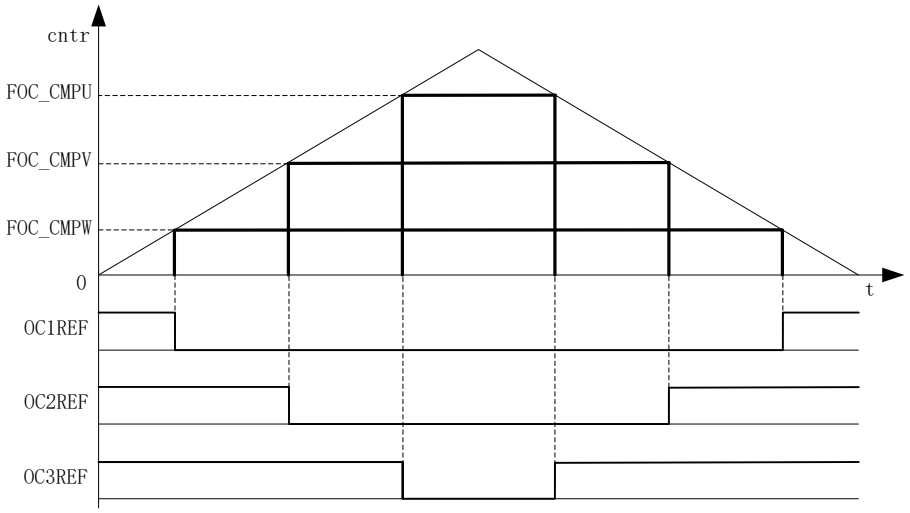


Figure 20-4 PMW Generation

When DRV_CR[OCS] = 0, DRV_DR set by software is compared with the count value to generate OC1REF/OC2REF/OC3REF with the same duty cycle. Duty cycle = DRV_DR/DRV_ARR*100%.

20.1.2.2 Deadtime Module

OCxREF supports deadtime insertion. For complementary outputs, the deadtime insertion is enabled when DRV_DTR is not “0”. Each channel has an 8-bit deadtime generator, and four channels have the same deadtime, which is set by DRV_DTR. When rising edge signals are detected, output high level of OCxL is delayed for a period of time set in DRV_DTR. When falling edge signals are detected, output high level of OcxH is delayed for a period of time set in DRV_DTR.

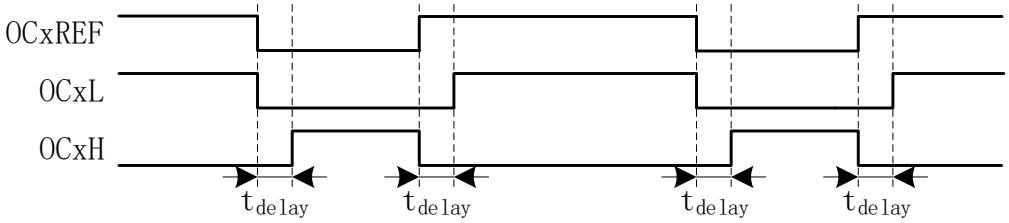


Figure 20-5 Complementary Outputs with Deadtime Insertion

20.1.2.3 Enable and Polarity of Output Signals

DRV_CMR[xHE] and DRV_CMR[xLE] are configured by software to enable high and low sides of the bridge, and DRV_CMR[xHP] and [xLP] to select the polarity of output. For square-wave control, Timer1 automatically controls DRV_CMR to implement phase commutation. Configuring DRV_CR[MESEL] = 0 enables the square wave drive Mode. After Timer1 generates a phase commutation event, the data stored in the corresponding TIM1_DBRx are transferred to DRV_CMR.

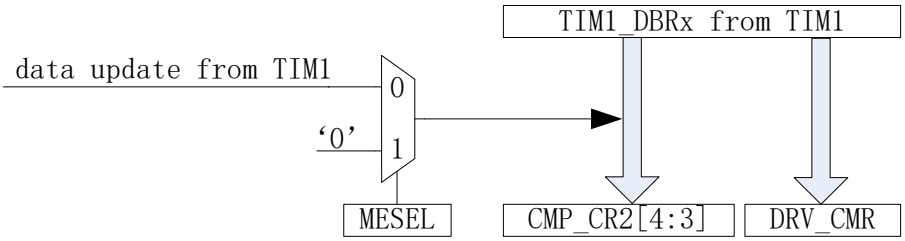


Figure 20-6 Timer1 Automatic Control of DRV_CMR and CMP_CR2[4:3]

DRV_DR, DRV_ARR and DRV_CMR can be configured to implement pre-charging, brake, etc. DRV_DR and DRV_ARR control the duty cycle and frequency of PWM. DRV_OUT[MOE] decides the output mode as idle level or PWM.

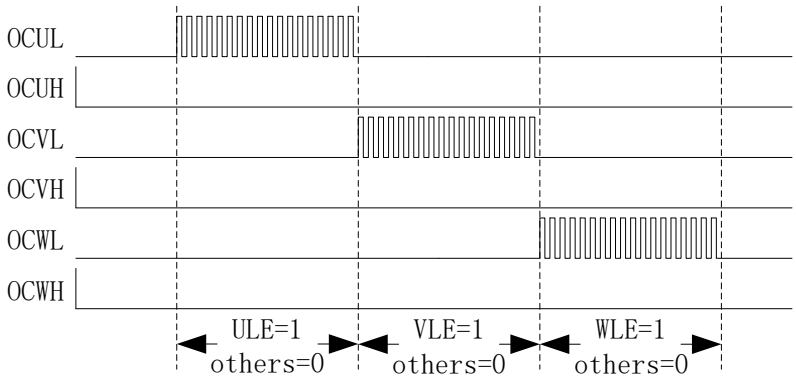


Figure 20-7 Pre-charge Waveform

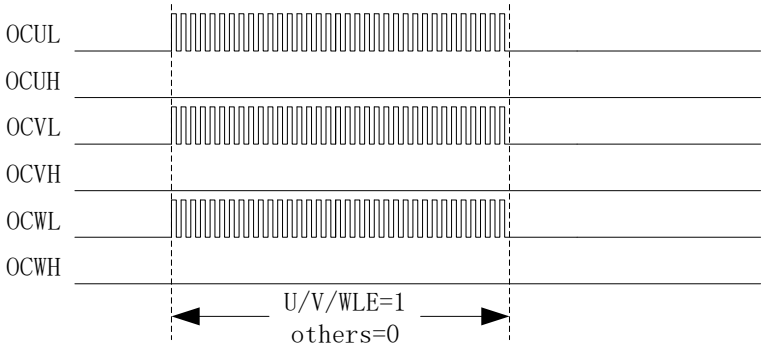


Figure 20-8 Brake Waveform

20.1.2.4 Main Output Enable (MOE)

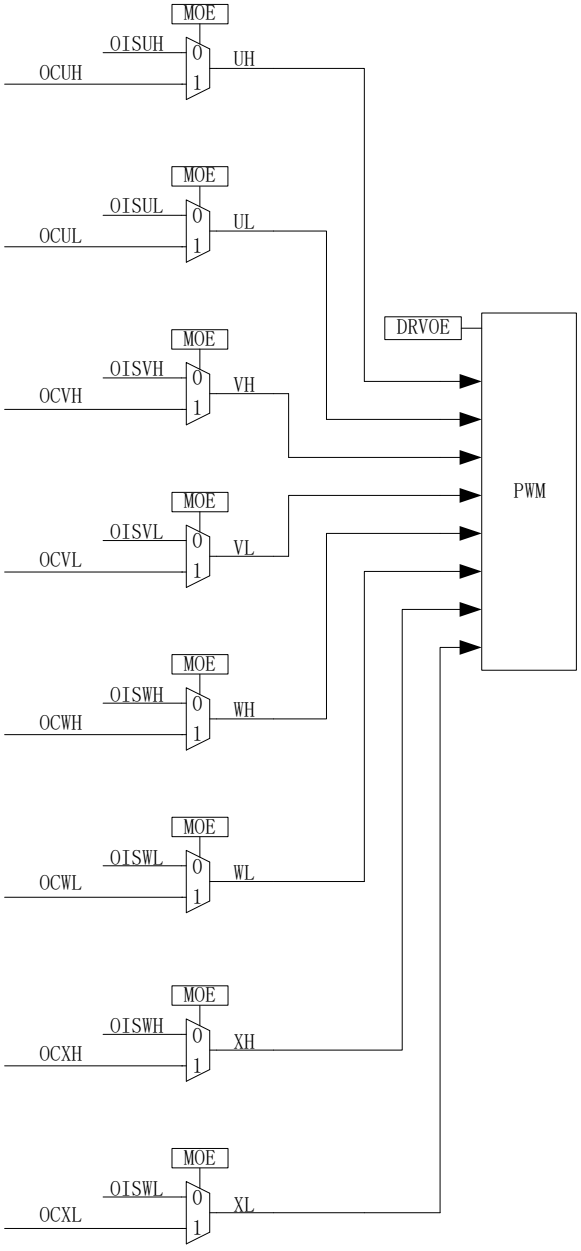


Figure 20-9 Block Diagram of Output Control Module

When DRV_OUT[MOE] is enabled, MOE module outputs PWM signals for motor control. When DRV_OUT[MOE] is disabled, the module outputs idle level set by the software to keep the motor at shutdown state.

20.1.2.5 Interrupt

20.1.2.5.1 Compare Match Interrupt

The generation conditions and time for compare match interrupt are configured by DRV_SR[DCIM] and DRV_COMR respectively. When the timer reaches the value set in DRV_COMR and the conditions set

by DRV_SR[DCIM] are met, a compare match interrupt is generated and the interrupt flag DRV_SR[DCIF] is set to “1” by hardware.

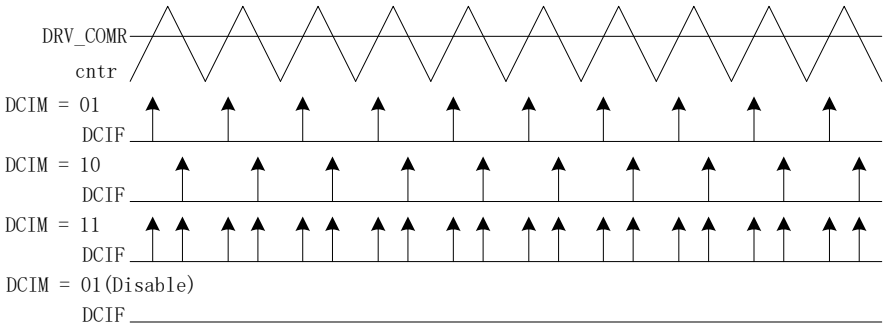


Figure 20-10 Driver Compare Match Interrupt

20.1.2.5.2 FG Interrupt

FG interrupt is enabled when DRV_SR[FGIE] is set to “1”. The motor generates an interrupt for every electrical cycle.

20.1.3 PWM Mode (FU6813)

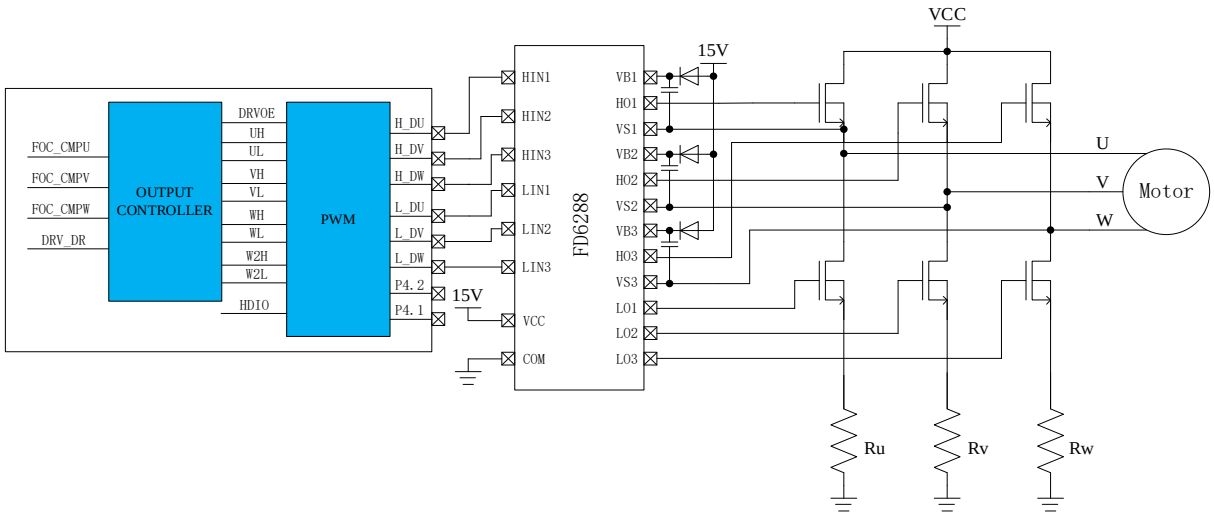


Figure 20-11 Block Diagram of PWM Mode

FU6813 adopts PWM output IC, as shown in Figure 20-11. Configuring DRV_CR[DRVOE] enables PWM mode, where the PWM output is connected to HVIC that drives the MOS gate.

20.1.4 6N Pre-driver Mode (FU6863)

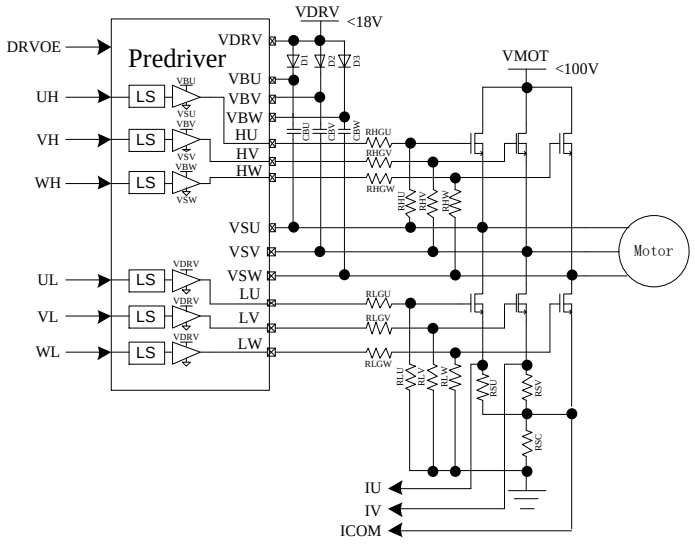


Figure 20-12 Block Diagram of 6NPre-driver Mode

The features of 6N pre-driver are shown in Figure 20-12. Configuring DRV_CR[DRVOE] enables pre-driver mode, where the pre-driver output is wired to 6 NMOS respectively to drive U/V/W-phases.

Table 20-1 Output Truth Values of FU6863Q Built-in Pre-driver

Input		Output	
UH/VH/WH	UL/VL/WL	HU/HV/HW	LU/LV/LW
L	L	L	L
L	H	L	H
H	L	H	L
H	H	L	L

20.2 Driver Registers

20.2.1 PI_CR (0xF9)

Bit	7	6	5	4	3	2	1	0
Name	T2TSS	RSV				DRVMD	HINV	LINV
Type	R/W	-	-	-	-	R/W	R/W	R/W
Reset	0	-	-	-	-	0	0	0

Bit	Name	Description
[7]	T2TSS	Input Mode Selection of TIM2 Step Motor 0: Direction +Pulse Input Mode. P1.0 for direction input, and P0.7 for pulse input 1: Bidirectional Pulse Input Mode. P1.0 for backward pulse input, and P0.7 for forward pulse input
[6:3]	RSV	Reserved
[2]	DRVMD	Count Mode 0: Triangular Wave Mode 1: Sawtooth Wave Mode (FOC disabled)
[1]	HINV	High Side Reverse Enable 0: Disable 1: Enable
[0]	LINV	Low Side Reverse Enable 0: Disable 1: Enable

20.2.2 DRV_CR (0x4062)

Bit	7	6	5	4	3	2	1	0
Name	DRVEN	DDIR	FOCEN	DRPE	OCS	MESEL	RSV	DRVOE
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]	DRVEN	Counter Enable 0: Disable 1: Enable
[6]	DDIR	Output Direction (Forward/Reverse) Switch motor rotation directions; Valid in both square-wave drive and FOC drive modes. In sensorless FOC mode, setting this bit changes motor rotation. In sensored FOC mode, it is also required to modify the angle by the software. In square-wave control mode, parameters related to Timer1 shall be configured. 0: Forward 1: Reverse
[5]	FOCEN	FOC Module Enable 0: Disable 1: Enable
[4]	DRPE	DRV_DR Pre-load Enable When preload is enabled, the data written to DRV_DR is updated after a timer underflow event occurs. When preload is disabled, the data written to DRV_DR is updated immediately. 0: Disable 1: Enable
[3]	OCS	Comparison Source Selection 0: DRV_DR 1: FOC Module

[2]	MESEL	ME Operating Mode Selection 0: Square Wave Drive 1: FOC Drive
[1]	RSV	Reserved
[0]	DRVOE	Driver Enable 0: Disable 1: Enable

20.2.3 DRV_SR (0x4061)

Bit	7	6	5	4	3	2	1	0
Name	SYSTIF	SYSTIE	FGIF	DCIF	FGIE	DCIP	DCIM	
Type	R/W0	R/W	R/W0	R/W0	R/W	R/W	R/W	R/W
Reset	0	0	0	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]	FGIF	FG Interrupt Flag Read 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	DCIF	Driver Match Interrupt Flag When the Driver count 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]	FGIE	FG Interrupt Enable After the interrupt feature is enabled, an FG Interrupt is generated in each electric cycle under FOC/square-wave control mode. 0: Disable 1: Enable
[2]	DCIP	Number of PWM cycles to generate a Compare Match Interrupt 0: 1 PWM cycle 1: 2 PWM cycles
[1:0]	DCIM	Compare Match Interrupt Mode Selection When the Driver count value is equal to DRV_COMR, the system decides whether to generate an interrupt according to DRV_SR[DCIM]. 00: No interrupt is generated. 01: An interrupt is generated when the timer counts up. 10: An interrupt is generated when the timer counts down. 11: An interrupt is generated when the timer counts up/down.

20.2.4 DRV_OUT (0xF8)

Bit	7	6	5	4	3	2	1	0
Name	MOE	RSV	OISWL	OISWH	OISVL	OISVH	OISUL	OISUH
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]	MOE	Main Output Enable This bit selects the sources for high and low sides of the bridge of 3-phase output signals. It can be set to “1” and “0” by software. When bus current protection occurs, it is automatically cleared to “0” to turn off the output (see section 30.1.1.1). 0: Disable, with output sourced from the idle levels set by DRV_OUT[OISUH]/DRV_OUT[OISVH]/DRV_OUT[OISWH] and DRV_OUT[OISUL]/DRV_OUT[OISVL]/DRV_OUT[OISWL]. 1: Enable, with output sourced from the comparison value of the timer.
[6]	RSV	Reserved
[5]	OISWL	Output idle level of WL/XL See descriptions on the bit OISUH. Note: DRV_OUT[OISWL] bit is configured as WL/XL output in IDLE state.
[4]	OISWH	Output idle level of WH/XH See descriptions on the bit OISUH. Note: DRV_OUT[OISWH] bit is configured as WH/XH output in IDLE state.
[3]	OISVL	Output idle level of VL See descriptions on the bit OISUH.
[2]	OISVH	Output idle level of VH See descriptions on the bit OISUH.
[1]	OISUL	Output idle level of UL See descriptions on the bit OISUH.
[0]	OISUH	Output idle level of UH This bit sets the UH output in IDLE state. When DRV_OUT[MOE] = 0, it outputs idle level to disable MOS. 0: Low 1: High

20.2.5 DRV_CMR (0x405C, 0x405D)

DRV_CMRH(0x405C)								
Bit	15	14	13	12	11	10	9	8
Name	XHP	XHL	XHE	XLE	WHP	WLP	VHP	VLP
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
DRV_CMRL(0x405D)								
Bit	7	6	5	4	3	2	1	0
Name	UHP	ULP	WHE	WLE	VHE	VLE	UHE	ULE
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]	XHP	High-side Polarity Control of X-phase 0: Active High 1: Active Low
[14]	XLP	Low-side Polarity Control of X-phase 0: Active High 1: Active Low

[13]	XHE	High-side Output Enable of X-phase 0: Disable 1: Enable
[12]	XLE	Low-side Output Enable of X-phase 0: Disable 1: Enable
[11]	WHP	High-side Polarity Control of W-phase 0: Active High 1: Active Low
[10]	WLP	Low-side Polarity Control of W-phase 0: Active High 1: Active Low
[9]	VHP	High-side Polarity Control of V-phase 0: Active High 1: Active Low
[8]	VLP	Low-side Polarity Control of V-phase 0: Active High 1: Active Low
[7]	UHP	High-side Polarity Control of U-phase 0: Active High 1: Active Low
[6]	ULP	Low-side Polarity Control of U-phase 0: Active High 1: Active Low
[5]	WHE	High-side Output Enable of W-phase 0: Disable 1: Enable
[4]	WLE	Low-side Output Enable of W-phase 0: Disable 1: Enable
[3]	VHE	High-side Output Enable of V-phase 0: Disable 1: Enable
[2]	VLE	Low-side Output Enable of V-phase 0: Disable 1: Enable
[1]	UHE	High-side Output Enable of U-phase 0: Disable 1: Enable
[0]	ULE	Low-side Output Enable of U-phase 0: Disable 1: Enable

Notes:

- When DRV_CMR[W/V/ULE] and DRV_CMR[W/V/UHE] are set to “1”, high-side and low-side outputs of W/V/U-phases are complementary to generate PWM signals with deadtime insertion.
- For square-wave control, Timer1 automatically controls DRV_CMR register.

20.2.6 DRV_ARR (0x405E,0x405F)

DRV_ARRH(0x405E)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV		DRV_ARR[13:8]					
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	0	0	0
DRV_ARRL(0x405F)								

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/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:14]	RSV	Reserved
[13:0]	DRV_ARR	Timer reload value, which determines PWM frequency (center-aligned) Driver timer counts from 0 to DRV_ARR and an overflow event occurs. Then it down-counts to 0. Calculation formula: $f_{Carrier} = f_{mcu}/2/(DRV\_ARR + 1)$ Range [0,4095]

20.2.7 DRV_COMR (0x405A, 0x405B)

DRV_COMRH(0x405A)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				DRV_COMR[11:8]			
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
DRV_COMRL(0x405B)								
Bit	7	6	5	4	3	2	1	0
Name	DRV_COMR[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_COMR	Timer Compare Match Value The compare match interrupt is generated when the count value is equal to DRV_COMR and the conditions set in DRV_SR[DCIM] are met. The clock rate for the calculation is 12MHz. Duty cycle at the match point = DRV_COMR*4/DRV_ARR*100% DRV_COMR value is calculated using 12MHz clock rate, which falls within the range [0, 4095].

20.2.8 DRV_DR (0x4058,0x4059)

DRV_DRH(0x4058)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				DRV_DR[11:8]			
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	0	0	0
DRV_DRL(0x4059)								
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 in Software When DRV_CR[OCS] = 0, DRV_CNTR is compared with DRV_DR to output PWM. “1” is output when DRV_CNTR is smaller than DRV_DR, and “0” is output when DRV_CNTR is larger than DRV_DR. 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. Range [0, 4095]

20.2.9 DRV_DTR (0x4060)

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 = (DRV_DTR + 1)*T Example: If DRV_DTR is configured to “11”, the deadtime = 12*41.67ns = 500ns. Note: If DRV_DTR is configured to “0”, deadtime insertion is disabled.

21 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 MCU, and the program starts running from address 0. WDT has to be initialized at regular intervals during operation, and WDT rolls over to WDT_ARR and restart counting.

21.1 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 timer overflow.

21.2 WDT Operations

1. Set CCFG1[WDT_EN] = 1 to start 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, and WDT rolls over to WDT_ARR setting.

21.3 WDT Registers

21.3.1 WDT_CR (0x4026)

Bit	7	6	5	4	3	2	1	0
Name	RSV						WDTF	WDTRF
Type	-	-	-	-	-	-	R/W0	R/W
Reset	-	-	-	-	-	-	0	0

Bit	Name	Description
[7:2]	RSV	Reserved
[1]	WDTF	WDT Reset Flag
[0]	WDTRF	WDT Initialization 0: No effect 1: WDT rolls over to WDT_ARR setting and restarts counting.

21.3.2 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 This bit sets high-order 8 bits of the initialized value of WDT.

21.3.3 CCFG1 (0x401E)

Bit	7	6	5	4	3	2	1	0
Name	LVWENB	LVWIE	WDTEN	RSV				
Type	R/W	R/W	R/W	-	-	-	-	-
Reset	-	0	0	-	-	-	-	-

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

22 RTC

22.1 RTC Functional Block Diagram

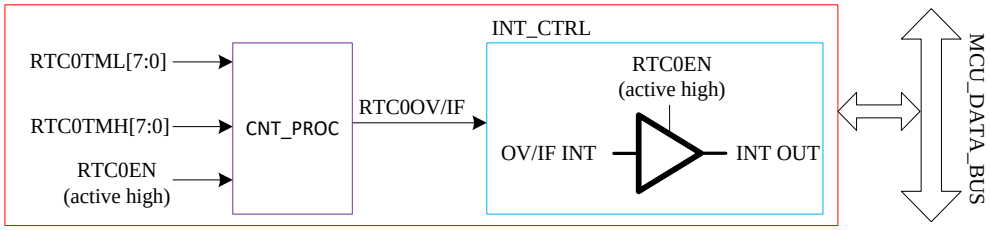


Figure 22-1 RTC Functional Block Diagram

22.2 RTC Operations

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

22.3 RTC Registers

22.3.1 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 Count Register Read: Instantaneous value of the timer Write: RTC timer up-counts at a rate of 32768Hz from 0 to the written value and becomes overflowed. Meanwhile, an interrupt request is generated, causing the timer to be cleared and restart counting.

22.3.2 RTC_STA (0x402E)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RTC_EN	RTC_OV/ RTC_IF	SCK_SEL	ISOSCEN	ESOSCAE	ESOSCEN	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]	RTC_EN	RTC Enable 0: Disable 1: Enable
[6]	RTC_OV/ RTC_IF	RTC Overflow/Interrupt Flag When IE[RTCIE] is “1”, an interrupt event is generated after this bit get

		overflowed. This bit is cleared to “0” by software. When IE[RTCIE] is “0”, an interrupt flag, instead of an interrupt event, is generated after this bit get overflowed. MCU can read and clear the flag. 0: RTC does not overflow. 1: RTC overflows, and it is cleared to “0” by software. Note: This bit will not be automatically cleared to “0”. It shall be cleared to “0” by software when an RTC interrupt is generated.
[5]	SCK_SEL	RTC Slow Clock Source Selection 0: Internal Slow Clock 1: External Slow Clock Note: This bit is also used for clock calibration. See section Clock Calibration for details.
[4]	ISOSCEN	Internal Slow Clock Enable 0: Disable 1: Enable
[3]	ESOSCAE	External Slow Clock Analog Form 0: Digital form. 1: Analog form. When an External Slow Clock is selected, the analog form must be configured.
[2]	ESOSCEN	External Slow Clock Enable 0: Disable 1: Enable
[1:0]	RSV	Reserved

22.4 Clock Calibration

22.4.1 Introduction

Clock calibration is a feature that uses internal slow clock to calibrate the internal fast clock. Working principles: A 12-bit timer 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 in software to start the calibration. Read CAL_CR0[CAL_BUSY] flag bit to check if the calibration process is completed. 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 4 slow clock cycles.

22.4.2 Clock Calibration Registers

22.4.2.1 CAL_CR0 (0x4044) CAL_CR1 (0x4045)

CAL_CR0(0x4044)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	CAL_STA/ CAL_BUSY	RSV			CAL_ARR[11:8]			
Type	R/W1	-	-	-	R/W	R/W	R/W	R/W
Reset	1	-	-	-	0	0	0	0
CAL_CR1(0x4045)								
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/ CAL_BUSY	Clock Calibration Enable Read: 0: Calibration is completed. 1: Calibration is in progress. Write: 0: No effect 1: Clock Calibration starts.
[14:12]	RSV	Reserved
[11:0]	CAL_ARR	Calibration Counts The count values of the fast clock to continuously count four slow clock cycles Note: When this value is 0, it indicates that no corresponding slow clock input exists, and when this value is 0xFFF , it indicates that the count overflows (slow clock is too slow or fast clock is too fast).

23 IO

23.1 IO Introduction

FU6813L has up to 34 GPIOs, including P0.0 ~ P0.7, P1.0 ~ P1.7, P2.0 ~ P2.7, P3.0 ~ P3.7 and P4.1 ~ P4.2.

FU6813N has up to 20 GPIOs, including P0.0 ~ P0.1, P0.4~ P0.7, P1.1 ~ P1.6, P2.1, P2.4, P2.6~ P2.7, P3.0 ~ P3.2 and P3.5.

FU6813P has up to 35 GPIOs, including P0.0 ~ P0.7, P1.0 ~ P1.7, P2.0~ P2.7, P3.0 ~ P3.7 and P4.0 ~ P4.2.

FU6863Q has up to 32 GPIO, including P0.0 ~ P0.7, P1.0 ~ P1.7, P2.0 ~ P2.7 and P3.0 ~ P3.7.

23.2 IO Operations

Each GPIO port pin has relevant 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.

Notes:

- The enable bits of pull-up resistors and pull-down resistors are configured to “1”. See 23.3.9 P0_PU (0x4053) ~ 23.3.13 P4_PU (0x4057) 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 P1_AN, P2_AN and P3_AN registers are configured to “1” to activate analog signal mode. See 23.3.6 P1_AN (0x4050) ~ 23.3.8 P3_AN (0x4052) 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 P1, P2 and P3 registers.
- DRV_CTL[OCS] selects Timer0, Timer1 or FOC module to output OCUH/OCVH/OCWH and OCUL/OCVL/OCWL of U/V/W-phase, and DRV_OUT[MOE] selects the output signal sourced from idle level DRV_OUT[OISUH]/[OISVH]/[OISWH] and DRV_OUT[OISUL]/[OISVL]/[OISWL] or output sources OCUH/OCVH/OCWH and OCUL/OCVL/OCWL to deliver to Driver module.
- Timer0 output, T0_OC4 (from channel 4 of Timer0) or idle level OIS4, is determined by DRV_OUT[MOE].
- “0” or “1” can be written to DRV_OUT[MOE] by software. When over-current protection event occurs, the bit DRV_OUT[MOE] is automatically cleared to ”0” by hardware.
- IO Priority:
  - GPIO has the lowest priority
  - P0.1: I^2C > TIMER4 > DBG_SIG > GPIO
  - P0.5: SPI > UART > GPIO
  - P0.6: SPI > UART > GPIO
  - P0.7: TIMER2 > CMP > SPI >GPIO

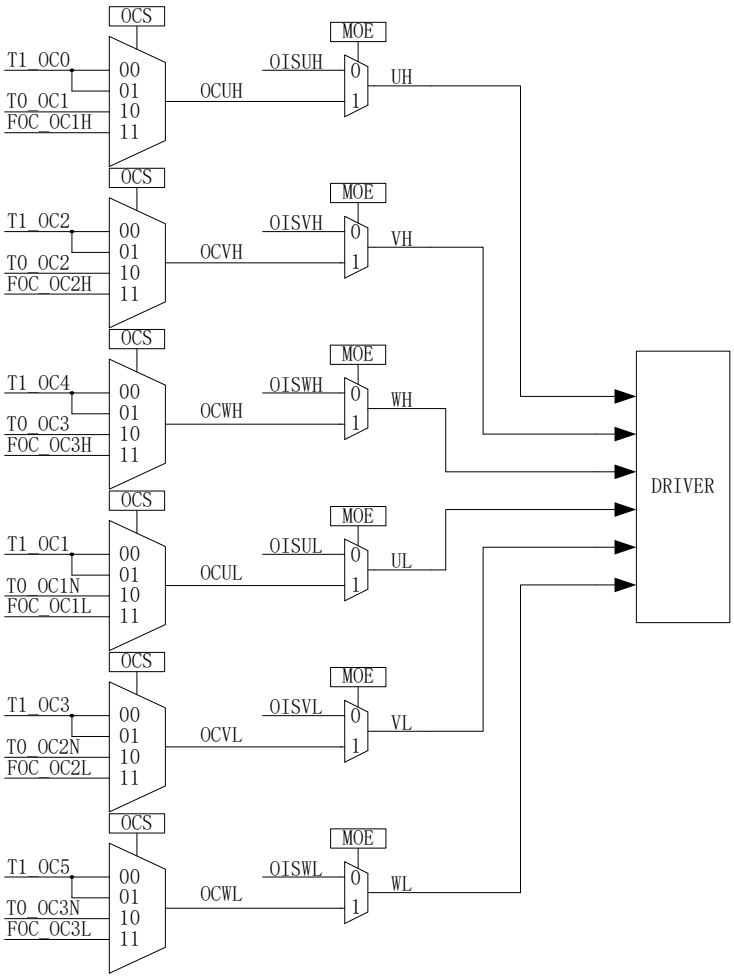


Figure 23-1 Phase-U/V/W Output Configurations

23.3 IO Registers

23.3.1 P0_OE (0xFC)

Bit	7	6	5	4	3	2	1	0
Name	P0_OE							
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]	P0_OE	P0.0 ~ P0.7 Digital I/O Selection 0: Input 1: Output

23.3.2 P1_OE (0xFD)

Bit	7	6	5	4	3	2	1	0
Name	P1_OE							
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]	P1_OE	P1.0 ~ P1.7 Digital I/O Selection 0: Input 1: Output

23.3.3 P2_OE (0xFE)

Bit	7	6	5	4	3	2	1	0
Name	P2_OE							
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]	P2_OE	P2.0 ~ P2.7 Digital I/O Selection 0: Input 1: Output

23.3.4 P3_OE (0xFF)

Bit	7	6	5	4	3	2	1	0
Name	P3_OE							
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]	P3_OE	P3.0 ~ P3.7 Digital I/O Selection 0: Input 1: Output

23.3.5 P4_OE (0xE9)

Bit	7	6	5	4	3	2	1	0
Name	RSV					P4_OE[2:0]		
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	-	-	-	-	-	0	0	0

Bit	Name	Description
[7:3]	RSV	Reserved
[2:0]	P4_OE[2:0]	P4.0 ~ P4.2 Digital I/O Selection 0: Input 1: Output

23.3.6 P1_AN (0x4050)

Bit	7	6	5	4	3	2	1	0
Name	P1_AN				HBMOD	HDIO	ODE1	ODE0
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
[7:4]	P1_AN	P1.4 ~ P1.7 Analog Mode Enable 0: Disable 1: Enable
[3]	HBMOD	P1.3 mode configuration, which determines the functional mode of P1.3 in combination with P1_OE[3], as shown in Table 23-1.

		Table 23-1 P1.3 Mode Setting HBMOD / P1_OE[3] / P1.3 Pin Mode 0 / 0 / Digital Input 0 / 1 / Digital Output 1 / 0 / Analog Mode 1 / 1 / Digital enhanced drive output mode. The drive mode of low level output is the same as that of the digital output mode.
[2]	HDIO	IO Driver Capability for PWM Output. It is valid for L_DU, L_DV, L_DW, H_DU, H_DV and H_DW only. 0: Normal drive capability 1: High drive capability
[1]	ODE1	P0.1 Collector Open-Drain Output Enable 0: Disable 1: Enable
[0]	ODE0	P0.0 Collector Open-Drain Output Enable 0: Disable 1: Enable

23.3.7 P2_AN (0x4051)

Bit	7	6	5	4	3	2	1	0
Name	P2_AN							
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]	P2_AN	P2.0 ~ P2.7 Analog Mode Enable 0: Disable 1: Enable

23.3.8 P3_AN (0x4052)

Bit	7	6	5	4	3	2	1	0
Name	RSV		P3_AN					
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]	P3_AN	P3.0 ~ P3.5 Analog Mode Enable 0: Disable 1: Enable

23.3.9 P0_PU (0x4053)

Bit	7	6	5	4	3	2	1	0
Name	P0_PU							
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]	P0_PU	P0.0 ~ P0.7 Pull-up Resistor Enable 0: Disable 1: Enable

23.3.10 P1_PU (0x4054)

Bit	7	6	5	4	3	2	1	0
Name	P1_PU							
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]	P1_PU	P1.0 ~ P1.7 Pull-up Resistor Enable 0: Disable 1: Enable

23.3.11 P2_PU (0x4055)

Bit	7	6	5	4	3	2	1	0
Name	P2_PU							
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]	P2_PU	P2.0 ~ P2.7 Pull-up Resistor Enable 0: Disable 1: Enable

23.3.12 P3_PU (0x4056)

Bit	7	6	5	4	3	2	1	0
Name	P3_PU							
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]	P3_PU	P3.0 ~ P3.7 Pull-up Resistor Enable 0: Disable 1: Enable

23.3.13 P4_PU (0x4057)

Bit	7	6	5	4	3	2	1	0
Name	RSV					P4_PU[2:0]		
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	-	-	-	-	-	0	0	0

Bit	Name	Description
[7:3]	RSV	Reserved
[2:0]	P4_PU	P4.0 ~ P4.2 Pull-up Resistor Enable 0: Disable 1: Enable

23.3.14 PH_SEL (0x404C)

Bit	7	6	5	4	3	2	1	0
Name	SPITMOD	UART1EN	UART2EN	T4SEL	T3SEL	T2SEL	T2SSEL	XOE
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]	SPITMOD	MISO port status after SPI slave device completes transmission 0: Output State 1: High-impedance State
[6]	UART1EN	UART1 Enable 0: Disable 1: Enable
[5]	UART2EN	UART2 Enable 0: Disable 1: Enable
[4]	T4SEL	Port Pins Multiplexed as Timer4 0: P0.1 is multiplexed as GPIO 1: P0.1 multiplexed as Timer4 I/O pins Note: I^2C has a higher priority than Timer4. When I^2C is enabled, P0.1 acts as SCL for I^2C communication.
[3]	T3SEL	Port Pins Multiplexed as Timer3 0: P1.1 is multiplexed as GPIO 1: P1.1 multiplexed as Timer3 I/O pins
[2]	T2SEL	Port Pins Multiplexed as Timer2 0: P1.0 is multiplexed as GPIO 1: P1.0 multiplexed as Timer2 I/O pins
[1]	T2SSEL	Port Pins Multiplexed as Timer2 Port 2 0: P0.7 is multiplexed as GPIO. 1: P0.7 is multiplexed as I/O pins of Timer2 port 2. Note: Timer2 has the highest priority, then the comparator output and SPI MISO.
[0]	XOE	XH/L Port Enable 0: P4.2/P4.1 is multiplexed as GPIO 1: P4.2/P4.1 is multiplexed as XH/XL output pin.

23.3.15 P0 (0x80)

Port output register P0/1/2/3/4 supports read and write access. RMW commands are used to access the register value (see Table 23-2 for RMW commands), and other commands are used to access PORT pin.

Bit	7	6	5	4	3	2	1	0
Name	GP07	GP06	GP05	GP04	GP03	GP02	GP01	GP00
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]	GP07	GP07 pin
[6]	GP06	GP06 pin
[5]	GP05	GP05 pin
[4]	GP04	GP04 pin
[3]	GP03	GP03 pin
[2]	GP02	GP02 pin

[1]	GP01	GP01 pin
[0]	GP00	GP00 pin

23.3.16 P1 (0x90)

Bit	7	6	5	4	3	2	1	0
Name	GP17	GP16	GP15	GP14	GP13	GP12	GP11	GP10
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]	GP17	GP17 pin
[6]	GP16	GP16 pin
[5]	GP15	GP15 pin
[4]	GP14	GP14 pin
[3]	GP13	GP13 pin
[2]	GP12	GP12 pin
[1]	GP11	GP11 pin
[0]	GP10	GP10 pin

23.3.17 P2 (0xA0)

Bit	7	6	5	4	3	2	1	0
Name	GP27	GP26	GP25	GP24	GP23	GP22	GP21	GP20
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]	GP27	GP27 pin
[6]	GP26	GP26 pin
[5]	GP25	GP25 pin
[4]	GP24	GP24 pin
[3]	GP23	GP23 pin
[2]	GP22	GP22 pin
[1]	GP21	GP21 pin
[0]	GP20	GP20 pin

23.3.18 P3 (0xB0)

Bit	7	6	5	4	3	2	1	0
Name	GP37	GP36	GP35	GP34	GP33	GP32	GP31	GP30
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]	GP37	GP37 pin
[6]	GP36	GP36 pin
[5]	GP35	GP35 pin
[4]	GP34	GP34 pin
[3]	GP33	GP33 pin

[2]	GP32	GP32 pin
[1]	GP31	GP31 pin
[0]	GP30	GP30 pin

23.3.19 P4 (0xE8)

Bit	7	6	5	4	3	2	1	0
Name	GP47	GP46	GP45	GP44	GP43	GP42	GP41	GP40
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]	GP47	GP47 pin
[6]	GP46	GP46 pin
[5]	GP45	GP45 pin
[4]	GP44	GP44 pin
[3]	GP43	GP43 pin
[2]	GP42	GP42 pin
[1]	GP41	GP41 pin
[0]	GP40	GP40 pin

Note: P4 has three pins, and their corresponding output registers are P4[2:0] respectively.

Table 23-2 RMW Commands

Command	Description
ANL	Bitwise logical AND operation
ORL	Bitwise logical OR operation
XRL	Bitwise logical XOR operation
JBC	Jump if the bit is set to“1” and then cleared to “0”
CPL	Bitwise logical converse operation
INC, DEC	+1, -1 logical operation
DJNZ	Jump if the bit is not “0”
MOV Px, y, C	Assign carry bit C to Px, y
CLR Px, y	Px, y is cleared to “0”
SETB Px, y	Px, y is set to “1”

24 ADC

24.1 ADC Introduction

The ADC module is a 12-bit successive approximation register ADC with 14 channels inside. The sampling mode supports sequential sampling (that is, from ADC Channel 0 to ADC channel 13 in sequence) and trigger sampling (including FOC triggered sampling mode and Timer1 triggered sampling mode). The results of sequential sampling are stored in ADCx_DR (x = 0 ~ 13) in a right-aligned or left-second-high-aligned format. The result of triggered sampling is sent to FOC module or Timer1 module instead of ADCx_DR for motor control. The relevant registers of the FOC module or Timer1 module are always left-second-high-aligned to store the triggered sample results. Triggered sampling is done automatically by hardware, and sequential sampling is controlled by software. The priority of triggered sampling is higher than that of sequential sampling. If both triggered sampling and sequential sampling are applied at the same time, the triggered sampling is performed first, and ADC automatically stores sequential sampling mode upon completion of triggered sampling.

The clock source of ADC is at a rate of 12MHz and the sampling time is set by DAC_CR[5:2] and ADC_SCYC. See ADC Electrical Characteristics for sample time and conversion time.

24.2 ADC Block Diagram

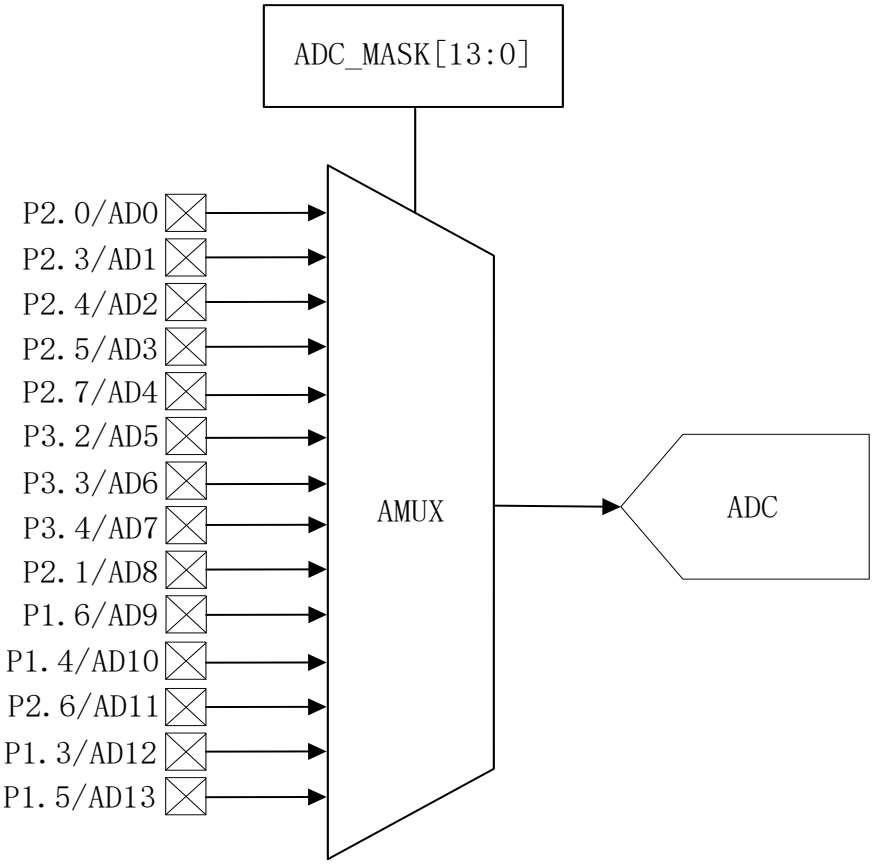


Figure 24-1 ADC Multiplexer Block Diagram

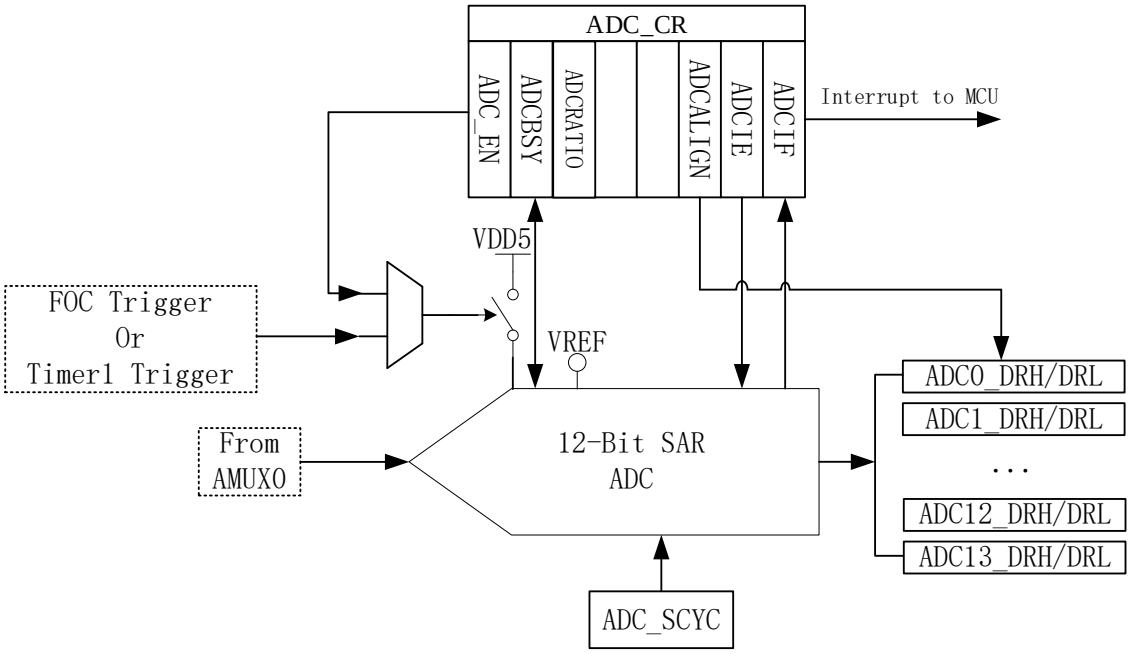


Figure 24-2 ADC Functional Block Diagram

24.3 ADC Operations

24.3.1 Sequential Sampling Mode

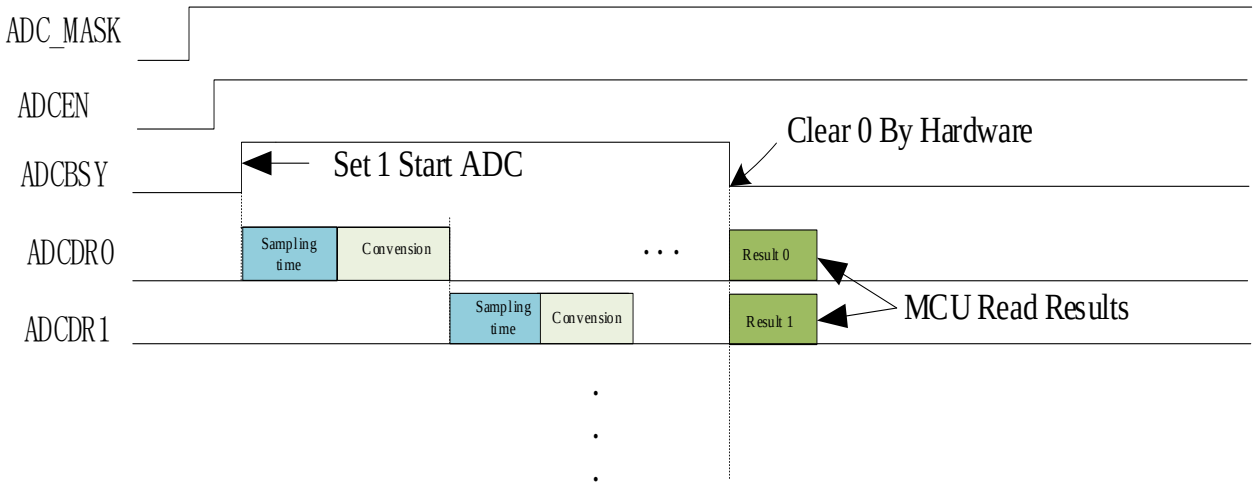


Figure 24-3 ADC Sequential Sampling Timing

ADC operations:

1. Set the appropriate ADC VREF. It shall be noted that if internal VREF is selected as the reference voltage and VREF = 5V, VDD5 must be greater than 5.3V, that is, VREF = 5V is not available in high-voltage mode (VCC_MODE = 0).
2. Configure ADC_MASK to enable the corresponding channel required to sample;

3. Set sampling cycle ADC_SCYC for each channel (minimum value is 3);
4. Configure ADC_CR[ADCEN] = 1 to enable ADC;
5. Configure ADC_CR[ADCBSY] = 1 to start ADC;
6. Read ADC results when ADC_CR[ADCBSY] = 0.

Note: The ADC conversion sequence is from low to high based on the enabled channel (i.e., when channel 2/3/4 is enabled, the signal is sampled in order of 2/3/4, and then a single conversion result is read after ADC_CR[ADCBSY] = 0).

24.3.2 Triggered Sampling Mode

When FOC module is enabled, ADC channel 0/1/2/4 can be used to FOC trigger sampling. Channel 2 is used for bus voltage trigger sampling. In single-shunt current sampling mode, channel 4 is used for itrip sampling. In dual-shunt current sampling mode, channel 0 is used for ia sampling and channel 1 for ib sampling. In triple-shunt current sampling mode, channel 0 is used for ia sampling, channel 1 for ib sampling, and channel 4 for ic sampling.

When Timer1 is enabled, channel 4 is used for bus current sampling. TIM1_CR3[T1TIS] is configured to select the input source of position detection as ADC. When CMP0_CR4[CMP0FS] = 0, channel 10 is used for U-phase voltage sampling, channel 9 for V-phase voltage sampling, and channel 8 for W-phase voltage sampling. When CMP0_CR4[CMP0FS] = 1, channel 10 is used for U-phase voltage sampling, channel 12 for V-phase voltage sampling, and channel 13 for W-phase voltage sampling.

24.4 ADC Registers

24.4.1 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/W0
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: Writing “1” to this bit has no effect when ADC_MASK = 0.
[5:3]	RSV	Reserved
[2]	ADCALIGN	ADC Data Format Selection 0: ADC output is right-aligned, and ADC result = ADCx_DR[11:0] 1: ADC output is left-second-high-aligned, and ADC result = ADCx_DR[14:3]

		Note: The results of triggered sampling mode are always left-second-high-aligned.
[1]	ADCIE	ADC Interrupt Enable (excluding triggered sampling mode interrupt) 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

24.4.2 ADC_MASK (0x4036 ~ 0x4037)

ADC_MASKH(0x4036)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV		CH13EN	CH12EN	CH11EN	CH10EN	CH9EN	CH8EN
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	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**
[15:14]	RSV	Reserved
[13]	CH13EN	ADC Channel 13 Enable
[12]	CH12EN	ADC Channel 12 Enable
[11]	CH11EN	ADC Channel 11 Enable
[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

Note: In triggered sampling mode, it is not required to configure ADC_MASK.

24.4.3 DAC_CR (0x4035)

DAC_CR(0x4035)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	DAC0_1EN	DACMOD	ADC_SCYCH[3:0]				DAC2EN	RSV
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-
Reset	0	0	0	0	1	1	0	-

Bit	Name	Description
[7]	DAC0_1EN	See section DAC_CR (0x4035) in DAC chapter.
[6]	DACMOD	See section DAC_CR (0x4035) in DAC chapter.
[5:2]	ADC_SCYCH [3:0]	ADC Sampling Cycle for ADC Channel 8 ~ 13 ADC_SCYCH[3] = 0: The sampling cycle is ADC_SCYCH[2:0] ADC clock cycles. ADC_SCYCH[3] = 1: The sampling cycle is (ADC_SCYCH[2:0]*8 + 7) ADC clock cycles.
[1]	DAC2EN	See section DAC_CR (0x4035) in DAC chapter.
[0]	RSV	Reserved

24.4.4 ADC_SCYC (0x4038)

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

Bit	Name	Description
[7:4]	ADC_SCYC [7:4]	ADC Sampling Cycle for ADC Channel 5 ~ 7 ADC_SCYC[7] = 0: The sampling cycle is ADC_SCYC[6:4] ADC clock cycles. ADC_SCYC[7] = 1: The sampling cycle is (ADC_SCYC[6:4]*8 + 7) ADC clock cycles.
[3:0]	ADC_SCYC [3:0]	ADC Sampling Cycle for ADC Channel 0 ~ 3 ADC_SCYC[3] = 0: The sampling cycle is ADC_SCYC[2:0] ADC clock cycles. ADC_SCYC[3] = 1: The sampling cycle is (ADC_SCYC[2:0]*8 + 7) ADC clock cycles.

24.4.5 ADC0_DR (0x0600 ~ 0x0601)

ADC0_DRH(0x0600)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC0_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC0_DRL(0x0601)								
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
[15:12]	RSV	Reserved
[11:8]	ADC0_DR[11:8]	High-order 4 bits of the result by ADC channel 0 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC0_DR[7:0]	High-order 8 bits of the result by ADC channel 0 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

Note: ADC results are not updated to this register in triggered sampling mode.

24.4.6 ADC1_DR (0x0602 ~ 0x0603)

ADC1_DRH(0x0602)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC1_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC1_DRL(0x0603)								
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
[15:12]	RSV	Reserved
[11:8]	ADC1_DR[11:8]	High-order 4 bits of the result by ADC channel 1 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC1_DR[7:0]	High-order 8 bits of the result by ADC channel 1 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.7 ADC2_DR (0x0604 ~ 0x0605)

ADC2_DRH(0x0604)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC2_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC2_DRL(0x0605)								
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
[15:12]	RSV	Reserved
[11:8]	ADC2_DR[11:8]	High-order 4 bits of the result by ADC channel 2 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC2_DR[7:0]	High-order 8 bits of the result by ADC channel 2 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.8 ADC3_DR (0x0606 ~ 0x0607)

ADC3_DRH(0x0606)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC3_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC3_DRL(0x0607)								
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
[15:12]	RSV	Reserved
[11:8]	ADC3_DR[11:8]	High-order 4 bits of the result by ADC channel 3 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC3_DR[7:0]	High-order 8 bits of the result by ADC channel 3 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.9 ADC4_DR (0x0608 ~ 0x0609)

ADC4_DRH(0x0608)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC4_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC4_DRL(0x0609)								
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
[15:12]	RSV	Reserved
[11:8]	ADC4_DR[11:8]	High-order 4 bits of the result by ADC channel 4 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC4_DR[7:0]	High-order 8 bits of the result by ADC channel 4 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.10 ADC5_DR (0x060A ~ 0x060B)

ADC5_DRH(0x060A)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC5_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC5_DRL(0x060B)								
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
[15:12]	RSV	Reserved
[11:8]	ADC5_DR[11:8]	High-order 4 bits of the result by ADC channel 5 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC5_DR[7:0]	High-order 8 bits of the result by ADC channel 5 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.11 ADC6_DR (0x060C ~ 0x060D)

ADC6_DRH(0x060C)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC6_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC6_DRL(0x060D)								
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
[15:12]	RSV	Reserved
[11:8]	ADC6_DR[11:8]	High-order 4 bits of the result by ADC channel 6 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC6_DR[7:0]	High-order 8 bits of the result by ADC channel 6 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.12 ADC7_DR (0x060E ~ 0x060F)

ADC7_DRH(0x060E)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC7_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC7_DRL(0x060F)								
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
[15:12]	RSV	Reserved
[11:8]	ADC7_DR[11:8]	High-order 4 bits of the result by ADC channel 7 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC7_DR[7:0]	High-order 8 bits of the result by ADC channel 7 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.13 ADC8_DR (0x0610 ~ 0x0611)

ADC8_DRH(0x0610)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC8_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC8_DRL(0x0611)								

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
[15:12]	RSV	Reserved
[11:8]	ADC8_DR[11:8]	High-order 4 bits of the result by ADC channel 8 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC8_DR[7:0]	High-order 8 bits of the result by ADC channel 8 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.14 ADC9_DR (0x0612 ~ 0x0613)

ADC9_DRH(0x0612)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				ADC9_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC9_DRL(0x0613)								
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
[15:12]	RSV	Reserved
[11:8]	ADC9_DR[11:8]	High-order 4 bits of the result by ADC channel 9 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC9_DR[7:0]	High-order 8 bits of the result by ADC channel 9 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.15 ADC10_DR (0x0614 ~ 0x0615)

ADC10_DRH(0x0614)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				ADC10_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC10_DRL(0x0615)								
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
[15:12]	RSV	Reserved
[11:8]	ADC10_DR[11:8]	High-order 4 bits of the result by ADC channel 10 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC10_DR[7:0]	High-order 8 bits of the result by ADC channel 10 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.16 ADC11_DR (0x0616 ~ 0x0617)

ADC11_DRH(0x0616)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC11_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC11_DRL(0x0617)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC11_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:8]	ADC11_DR[11:8]	High-order 4 bits of the result by ADC channel 11 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC11_DR[7:0]	High-order 8 bits of the result by ADC channel 11 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.17 ADC12_DR (0x0618 ~ 0x0619)

ADC12_DRH(0x0618)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC12_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC12_DRL(0x0619)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC12_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:8]	ADC12_DR[11:8]	High-order 4 bits of the result by ADC channel 12 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC12_DR[7:0]	High-order 8 bits of the result by ADC channel 12 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

24.4.18 ADC13_DR (0x061A ~ 0x061B)

ADC13_DRH(0x061A)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV				ADC13_DR[11:8]			
Type	-	-	-	-	R	R	R	R
Reset	-	-	-	-	0	0	0	0
ADC13_DRL(0x061B)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADC13_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	**Name**	**Description**
[15:12]	RSV	Reserved
[11:8]	ADC13_DR[11:8]	High-order 4 bits of the result by ADC channel 13 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].
[7:0]	ADC13_DR[7:0]	High-order 8 bits of the result by ADC channel 13 after ADC conversion The data is aligned according to ADC_CR[ADCALIGN].

25 DAC

25.1 DAC Introduction

The chip integrates three DAC modules, where DAC0 is a 9-bit digital-to-analog converter, DAC1 is a 6-bit digital-to-analog converter and DAC2 is a 8-bit digital-to-analog converter

25.2 DAC0 Functional Block Diagram

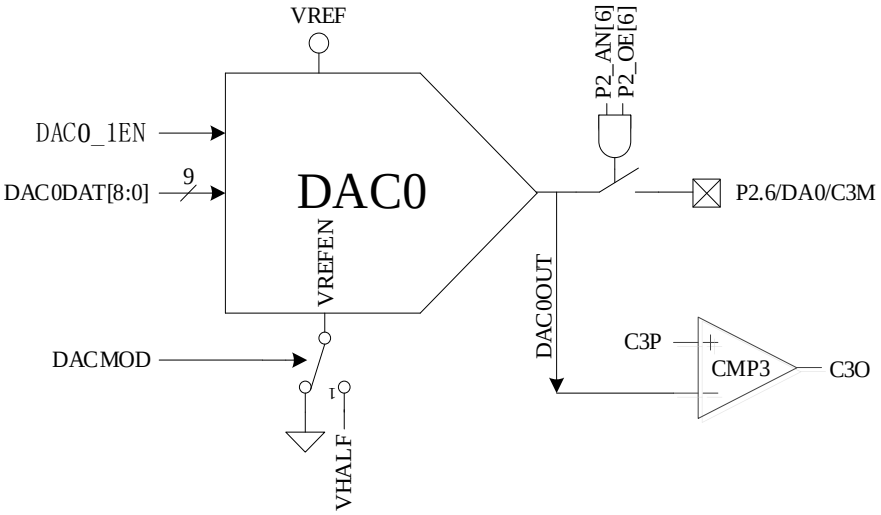


Figure 25-1 DAC0 Functional Block Diagram

As shown in Figure 25-1, DAC0 converts 9-bit digital data into analog voltage and sends the voltage to CMP3 negative input for bus over-current protection. P2.6 pin can be configured as the analog output.

Note: DAC0 output has no current drive capability and can only carry capacitive load. To carry resistive load, operational amplifiers are used to follow the voltage output.

DAC0 operations are as follows:

1. Configure P2_AN[6] = 1 and P2_OE[6] = 1, and DAC0 output to P2.6 pin;
2. Configure VREF_CR[VREFEN] = 1 and DAC_CR[DAC0_1EN] = 1, and VREF is used as DAC0 reference voltage;
3. The range of output voltage is set by DAC_CR[DACMOD]. When DAC_CR[DACMOD] = 0, full-voltage output mode is active, and the range of output voltage is 0~VREF. When DAC_CR[DACMOD] = 1, half-voltage output mode is active, the range of output voltage is VHALF~VREF. Output voltage of DAC0DAT under different configuration is shown in Table 25-1.

Table 25-1 Output Voltage of DAC0 under Different Configurations

DAC0DAT[8:0]	DAC Output Voltage (DAC_CR[DACMOD] = 0)	DAC Output Voltage (DAC_CR[DACMOD] = 1)
0x000	0	VHALF
0x100	VREF/2	(VREF - VHALF)/2 + VHALF
0x1FF	VREF*511/512	(VREF - VHALF)*511/512 + VHALF

25.3 DAC1 Functional Block Diagram

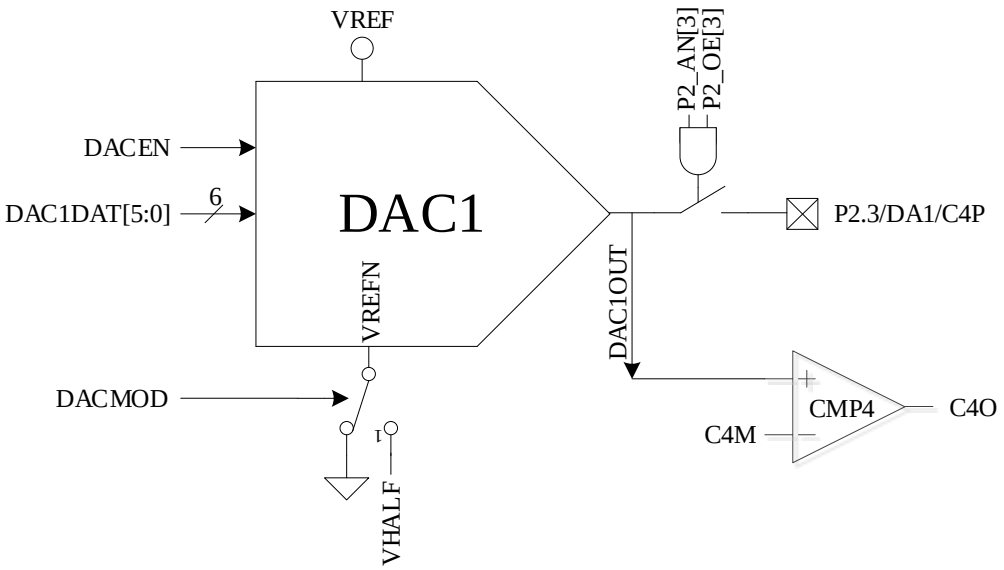


Figure 25-2 DAC1 Functional Block Diagram

DAC1 output has no current drive capability and can only carry capacitive load. To carry resistive load, operational amplifiers are used to follow the voltage output.

DAC1 operations are as follows:

1. Configure P2_AN[3] = 1 and P2_OE[3] = 1, and DAC1 output to P2.3/DA1 pin;
2. Configure VREF_VHALF_CR[VREFEN] = 1 and DAC_CR[DAC0_1EN] = 1, and VREF is used as DAC1 reference voltage.

25.4 DAC2 Functional Block Diagram

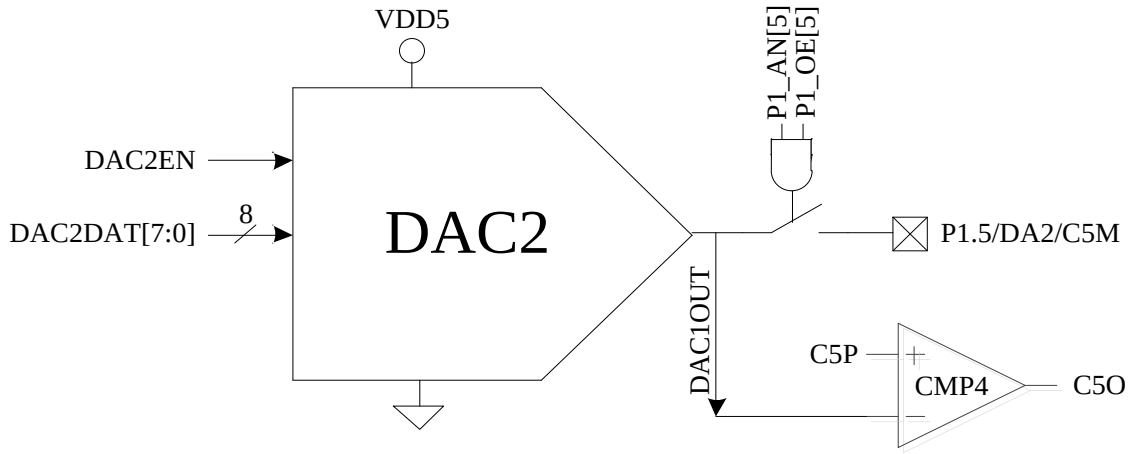


Figure 25-3 DAC2 Functional Block Diagram

DAC2 output has no current drive capability and can only carry capacitive load. To carry resistive load,

operational amplifiers are used to follow the voltage output.

DAC2 operations are as follows:

1. Configure P1_AN[5] = 1 and P1_OE[5] = 1, and DAC2 output to P1.5/DA2 pin;

25.5 DAC Registers

25.5.1 DAC_CR (0x4035)

Bit	7	6	5	4	3	2	1	0
Name	DAC0_1EN	DACMOD	ADC_SCYCH[3:0]				DAC2EN	RSV
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-
Reset	0	0	0	0	1	1	0	-

Bit	Name	Description
[7]	DAC0_1EN	DAC0, DAC1 Enable 0: Disable 1: Enable
[6]	DACMOD	DAC Mode Setting 0: Full-voltage Output Mode 1: Half-voltage Output Mode
[5:2]	ADC_SCYCH[3:0]	See section DAC_CR (0x4035) in chapter ADC.
[1]	DAC2EN	DAC2 Enable 0: Disable 1: Enable
[0]	RSV	Reserved

25.5.2 DAC0_DR (0x404B)

Bit	7	6	5	4	3	2	1	0
Name	DAC0DAT[8:1]							
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]	DAC0DAT[8:1]	High-order 8 bits input of DAC0 controller

25.5.3 DAC1_DR (0x404A)

Bit	7	6	5	4	3	2	1	0
Name	DAC0_DR_0	RSV	DAC1DAT					
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]	DAC0_DR_0	LSB input of DAC0 controller
[6]	RSV	Reserved
[5:0]	DAC1DAT	6-bit data input of DAC1 controller

25.5.4 DAC2_DR (0x4049)

Bit	7	6	5	4	3	2	1	0
Name	DAC2DAT							
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]	DAC2DAT	8-bit data input of DAC2 controller

26 DMA

26.1 DMA Instructions

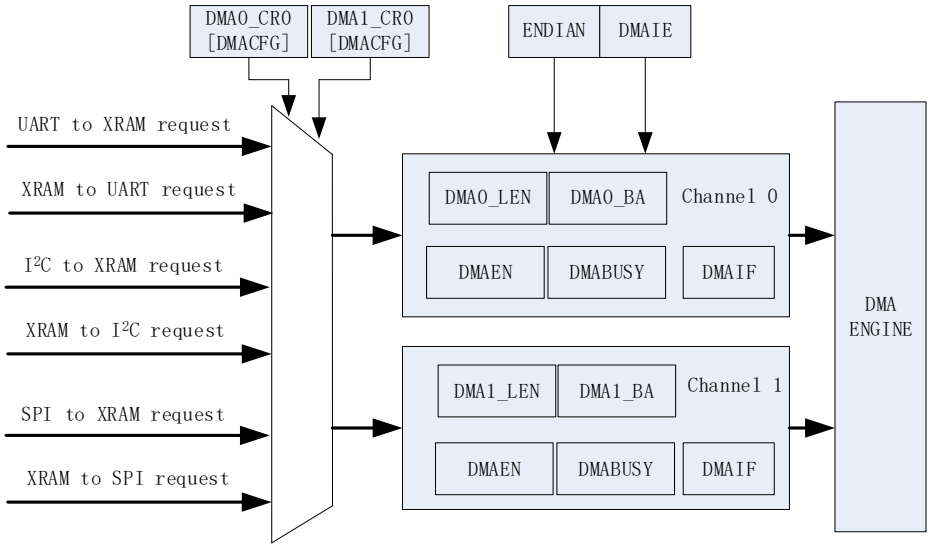


Figure 26-1 DMA Functional Block Diagram

DMA module is a dual-channel DMA controller, which performs direct data transfer between peripherals (SPI, UART, I²C) and XRAM (IRAM data invalid). DMA accessing to XRAM does not interfere with the normal CPU read/write operation to XRAM. The length of the transferred data and the start address of XRAM access is configurable. Data transfer mode is configurable and interrupt can be enabled.

DMA instructions are as follows:

1. Configure and enable the peripheral, and set input and output channels taken over by DMA by DMAx_CR0[DMACFG];
2. Configure DMA interrupt enable, transfer order, transfer length and XRAM start address. Write “1” to DMAx_CR0[DMAEN] and DMAx_CR0[DMABSY] to start DMA;
3. After data transfer, the interrupt flag bit DMAx_CR0[DMAIF] is set to “1” by hardware and it is cleared to “0” by software;
4. Set DMAx_CR0[DMABSY] to “1” to start DMA again.

26.2 DMA Registers

26.2.1 DMA0_CR0 (0x403A)

Bit	7	6	5	4	3	2	1	0
Name	DMAEN	DMABSY	DMACFG			DMAIE	ENDIAN	DMAIF
Type	R/W	R/W1	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DMAEN	DMABSY	DMACFG			DMAIE	ENDIAN	DMAIF
Type	R/W	R/W1	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]	DMAEN	DMA Channel 0 Enable 0: Disable 1: Enable
[6]	DMABSY	DMA Channel 0 Start/Busy Flag Read: 0: Channel 0 Idle 1: Channel 0 Busy Write: 0: No effect 1: Channel 0 starts for data transfer
[5:3]	DMACFG	DMA Channel 0 Peripherals and Transfer Direction Selection 000: From UART1 to XRAM 001: From XRAM to UART1 010: From I^2C to XRAM 011: From XRAM to I^2C 100: From SPI to XRAM 101: From XRAM to SPI 110: From UART2 to XRAM 111: From XRAM to UART2 Note: It cannot be configured when Channel 0 is busy.
[2]	DMAIE	DMA Interrupt Enable 0: Disable 1: Enable
[1]	ENDIAN	DMA Data Transfer Sequence 0: High bytes are received or sent first 1: Low bytes are received or sent first Note: This bit is set for 16-bit data mode, and shall be configured to “0” for 8-bit data mode. It cannot be configured when channel 0 or 1 is busy.
[0]	DMAIF	DMA Channel 0 Transfer Interrupt Event Flag Bit Read:

		0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: The interrupt event is generated.

26.2.2 DMA1_CR0 (0x403B)

Bit	7	6	5	4	3	2	1	0
Name	DMAEN	DMABSY	DMACFG			DMAIE	ENDIAN	DMAIF
Type	R/W	R/W1	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]	DMAEN	DMA Channel 1 Enable 0: Disable 1: Enable
[6]	DMABSY	DMA Channel 1 Start/Busy Read: 0: Channel 1 Idle 1: Channel 1 Busy Write: 0: No effect 1: Channel 1 starts for data transfer
[5:3]	DMACFG	DMA Channel 1 Peripherals and Direction Selection 000: From UART1 to XRAM 001: From XRAM to UART1 010: From I^2C to XRAM 011: From XRAM to I^2C 100: From SPI to XRAM 101: From XRAM to SPI 110: From UART2 to XRAM 111: From XRAM to UART2 Note: It cannot configured when Channel 1 is busy.
[2]	DBGSW	Sector Targeted in Debug Mode 0: XSFR as the Debug area (export address space: 0x4020 ~ 0x40FF) 1: XRAM as the Debug area (export address space: 0x0000 ~ 0x0317)
[1]	DBGEN	Debug Mode Enable DMA works in Debug mode when DMA1_CR0[DMACFG] is set to “101” and DMA1_CR0[DBGEN] to “1”. After SPI is enabled, DMA automatically sends relevant data in the sector defined by DMA1_CR0[DBGSW] via MOSI. DMA1_BA/DMA1_LEN defines the start address and range of the relevant data. 0: Disable 1: Enable Note: DMA Channel 1 Interrupt is automatically disabled in Debug mode.

[0]	DMAIF	DMA Channel 1 Transfer Interrupt Event Flag Bit Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: The interrupt event is generated.

26.2.3 DMA0_LEN (0x403C)

Bit	7	6	5	4	3	2	1	0
Name	RSV		DMA0_LEN					
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]	DMA0_LEN	Transfer Length of DMA Channel 0 Read: The number of the byte that is currently transferred by DMA Channel 0 (0 denotes the first byte) Write: XRAM data transfer length of DMA Channel 0 Note: It cannot be configured when Channel 0 is busy. When DMA0_CR0[ENDIAN] = 1 (low bytes are received or sent first) , it is recommended that DMA0_LEN be set to an odd number.

26.2.4 DMA0_BA (0x403E, 0x403F)

DMA0_BAH(0x403E)								
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Name	RSV					DMA0_BA[10:8]		
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
DMA0_BAL(0x403F)								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	DMA0_BA[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
[15:11]	RSV	Reserved
[10:0]	DMA0_BA	Start address of data transfer by DMA Channel 0 Start address of XRAM data transfer by DMA Channel 0 It cannot be configured when Channel 0 is busy. Note: XRAM address space for data transfer by Channel 0: DMA0_BA[10:0] ~ (DMA0_BA[10:0] + DMA0_LEN[5:0])

26.2.5 DMA1_LEN (0x403D)

Bit	7	6	5	4	3	2	1	0
Name	RSV		DMA1_LEN					
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]	DMA1_LEN	Transfer length of DMA Channel 1 Read: The number of the bytes that is currently transferred by DMA Channel 1 (0 denotes the first byte) Write: XRAM data transfer length of DMA Channel 1 Note: It cannot be configured when Channel 1 is busy. When DMA0_CR0[ENDIAN] = 1 (low bytes are received or transmitted first), it is recommended that DMA1_LEN be set to an odd number.

26.2.6 DMA1_BA (0x4040, 0x4041)

DMA1_BAH(0x4040)								
Bit	15	14	13	12	11	10	9	8
Name	RSV					DMA1_BA[10:8]		
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
DMA1_BAL(0x4041)								
Bit	7	6	5	4	3	2	1	0
Name	DMA1_BA[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
[15:11]	RSV	Reserved
[10:0]	DMA1_BA	Start address of data transfer by DMA Channel 1 Start address of XRAM data transfer by DMA Channel 1 It cannot be configured when Channel 1 is busy. Note: XRAM address space for data transfer by Channel 1: DMA1_BA[10:0] ~ (DMA1_BA[10:0] + DMA1_LEN[5:0])

Note: When I^2C is selected as DMA channel peripherals (including from I^2C to XRAM and from XRAM to I^2C) , START + Address interrupt of I^2C communication still requires to be cleared to “0” by MCU software. In I^2C slave mode, if STOP is received, I2C_SR[I2CSTP] = 0 is configured to clear I^2C interrupt and restart the DMA transfer.

27 VREF

27.1 VREF Instructions

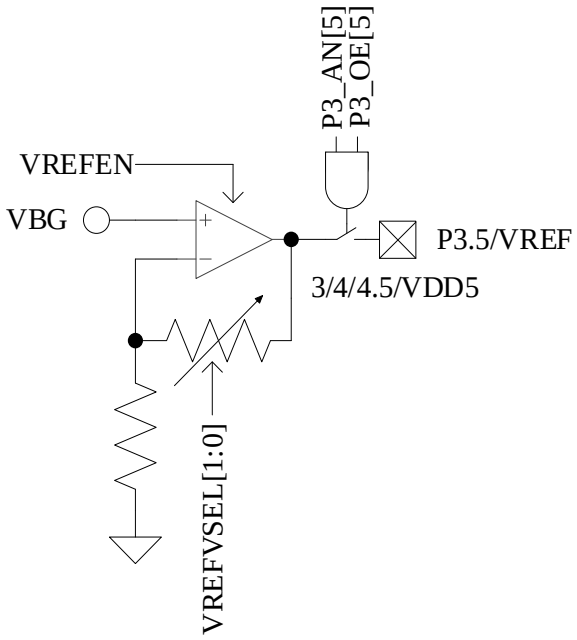


Figure 27-1 I/O Pins of VREF Module

The input and output ports of the VREF module are shown in Figure 27-1. VREF is the voltage reference generation block that provides internal voltage reference to ADC and DAC modules. VBG is the voltage supplied by the chip internally.

VREF is enabled when VREF_CR[VREFEN] is set to “1”. The output voltage is selected by configuring VREF_CR[VREFVSEL]. When P3_AN[5] = 1 and P3_OE[5] = 1, VREF is output to P3.5 pin.

27.2 VREF Register

27.2.1 VREF_VHALF_CR (0x404F)

Bit	7	6	5	4	3	2	1	0
Name	VREFVSEL		RSV	VREFEN	RSV			VHALFEN
Type	R/W	R/W	-	R/W	-	-	-	R/W
Reset	0	0	-	0	-	-	-	0

Bit	Name	Description
[7:6]	VREFVSEL	VREF Module Output Voltage Selection 00: 4.5V 01: VDD5 10: 3V 11: 4V
[5]	RSV	Reserved
[4]	VREFEN	VREF Module Enable 0: Disable. P3_AN[5] is set to “1”, and external VREF is input from P3.5 pin 1: Enable. P3_AN[5] is set to “1”, and internal VREF is output to P3.5 pin. A 0.1μF ~ 1μF external capacitor is added to improve the stability of VREF.
[3:1]	RSV	Reserved
[0]	VHALFEN	VHALF Enable 0: Disable 1: Enable

28 VHALF

28.1 VHALF Instructions

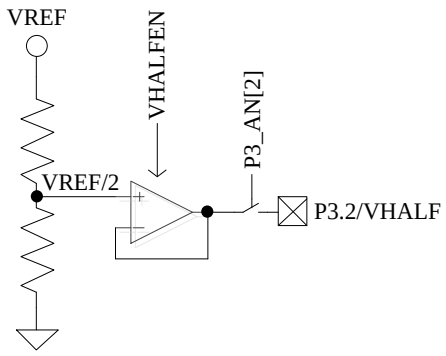


Figure 28-1 I/O Pins of VHALF Module

The input and output ports of VHALF module are shown in Figure 28-1. This module generates the voltage reference.

VHALF is enabled when VREF_CR[VHALFEN] is set to “1”, and the voltage is output to P3.2. A 1μF external capacitor is added.

28.2 VHALF Register

See VREF_VHALF_CR (0x404F) for details.

29 Operational Amplifier

The chip integrates four high-speed independent operational amplifiers, AMP0, AMP1, AMP2 and AMP3. Each operational amplifier has a separate enable bit. For FU6813N, it integrates two high-speed independent operational amplifiers, AMP0 and AMP3.

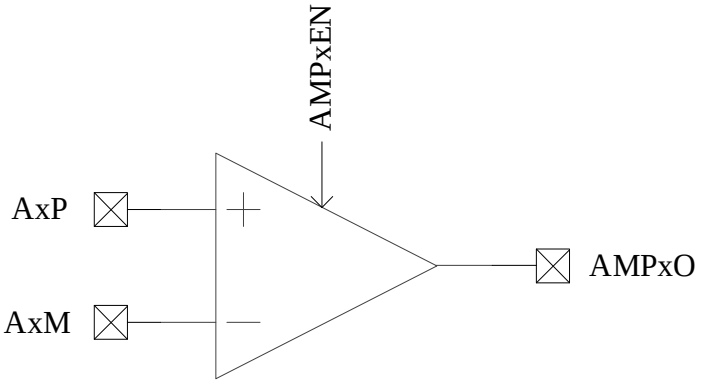


Figure 29-1 Schematic Diagram of Operational Amplifier Module

29.1 Operational Amplifier Instructions

29.1.1 Bus Current Sampling Operational Amplifier (AMP0)

Connection diagram of AMP0 is shown in Figure 29-2.

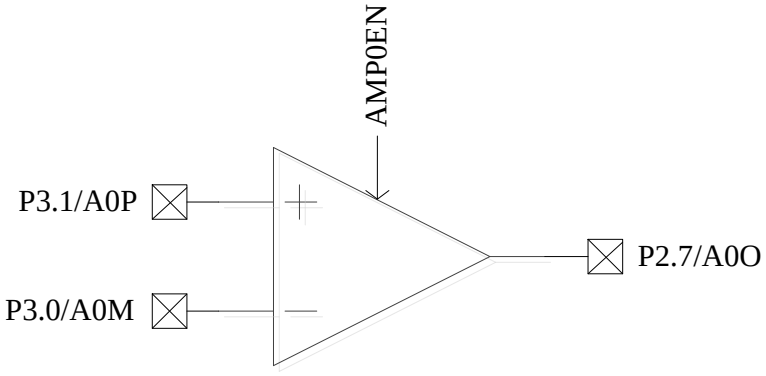


Figure 29-2 AMP0 I/O Pins

The I/O pins of AMP0 are shown in Figure 29-2. AMP0 is enabled when AMP_CR[AMP0EN] = 1, and P2.7, P3.0 and P3.1 are automatically configured to analog signal mode by the hardware.

29.1.2 Phase Current Operational Amplifier (AMP1/AMP2)

29.1.2.1 AMP1

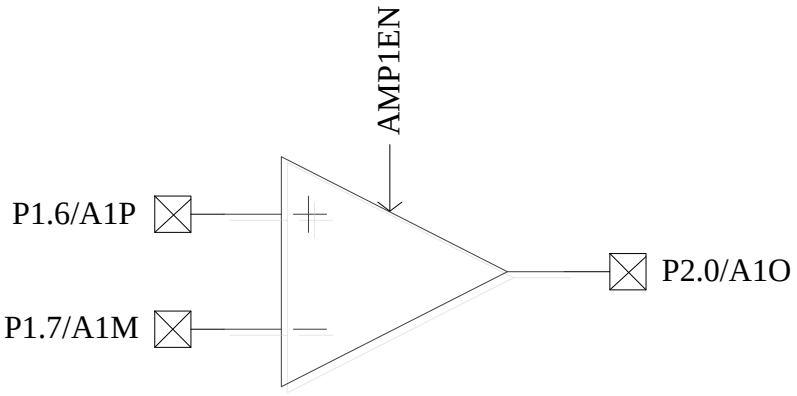


Figure 29-3 AMP1 I/O Pins

The I/O pins of AMP1 are shown in Figure 29-3. AMP1 is enabled when AMP_CR[AMP1EN] = 1, and P1.6, P1.7 and P2.0 pins are automatically configured to analog signal mode by the hardware. P1_AN[7:6] is set to “11” and P2_AN[0] to “1”.

29.1.2.2 AMP2

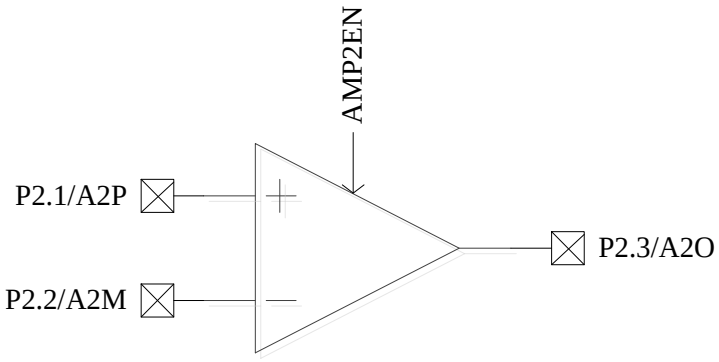


Figure 29-4 AMP2 I/O Pins

The I/O pins of AMP2 are shown in Figure 29-4. AMP2 is enabled when AMP_CR[AMP2EN] = 1, and P2.1, P2.2 and P2.3 are automatically configured to analog signal mode by the hardware. P2_ AN[3:1] is set to “111”.

29.1.3 Operational Amplifier AMP3

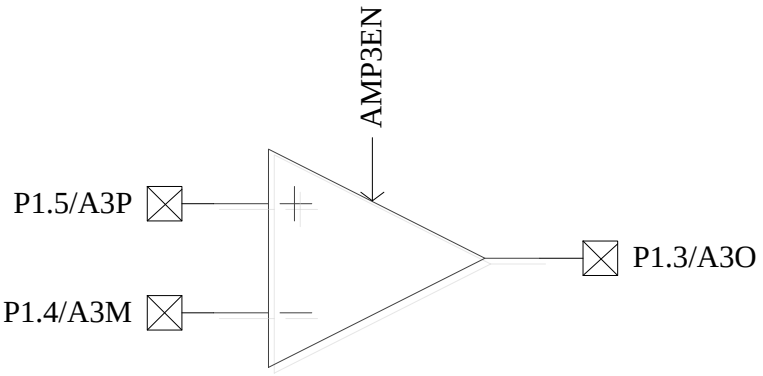


Figure 29-5 AMP3 I/O Pins

The I/O pins of AMP3 are shown in Figure 29-5. AMP3 is enabled when AMP_CR[AMP3EN] = 1, and P1.5, P1.4 and P1.3 are automatically configured to analog input mode by the hardware. P1_AN[5:3] is set to “111”.

29.2 Operational Amplifier Register

29.2.1 AMP_CR (0x404E)

Bit	7	6	5	4	3	2	1	0
Name	RSV				AMP3EN	AMP2EN	AMP1EN	AMP0EN
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0

Bit	Name	Description
[7:4]	RSV	Reserved
[3]	AMP3EN	AMP3 Enable 0: Disable 1: Enable
[2]	AMP2EN	AMP2 Enable 0: Disable 1: Enable
[1]	AMP1EN	AMP1 Enable 0: Disable 1: Enable
[0]	AMP0EN	AMP0 Enable 0: Disable 1: Enable

30 Comparator

30.1 Comparator Operations

30.1.1 Comparator CMP3

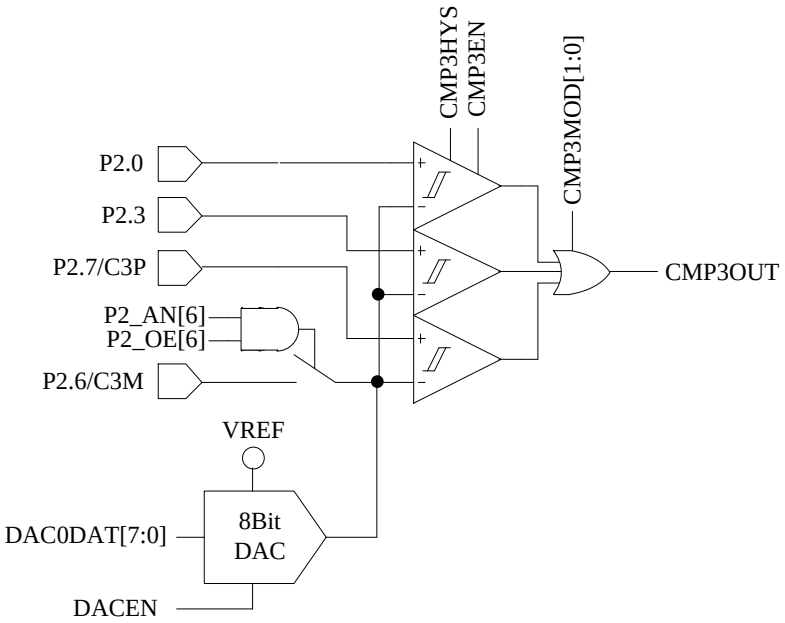


Figure 30-1 CMP3 I/O Pins

The I/O pins of CMP3 are shown in Figure 30-1. CMP3 configurations are as follows:

1. Configure P2_AN[6] and P2_OE[6] to “1” to enable VREF on the negative input of CMP3. The VREF source can be on-chip DAC0 output voltage or external circuit input voltage. Select DAC0 output, and place an external capacitor between P2.6 pin and GND (the recommended capacitance value is 100pF, and the output voltage stabilizes after DAC0 output for a period of time);
2. Configure CMP_CR1[CMP3MOD] to select single-comparator input, dual-comparator input, or triple-comparator input mode;
   - When CMP_CR1[CMP3MOD] = 00, CMP3 works in Single-comparator Input Mode. The connection of input and output pins are shown in Figure 30-2.
   - When CMP_CR1[CMP3MOD] = 01, CMP3 works in Dual-comparator Input Mode. The connection of input and output pins are shown in Figure 30-3.
   - When CMP_CR1[CMP3MOD] = 1X, CMP3 works in Triple-comparator Input Mode. The connection of input and output pins are as shown in Figure 30-4.
3. Configure CMP_CR1[CMP3HYS] to enable or disable hysteresis;
4. Set CMP_CR1[CMP3EN] = 1 to enable CMP3.

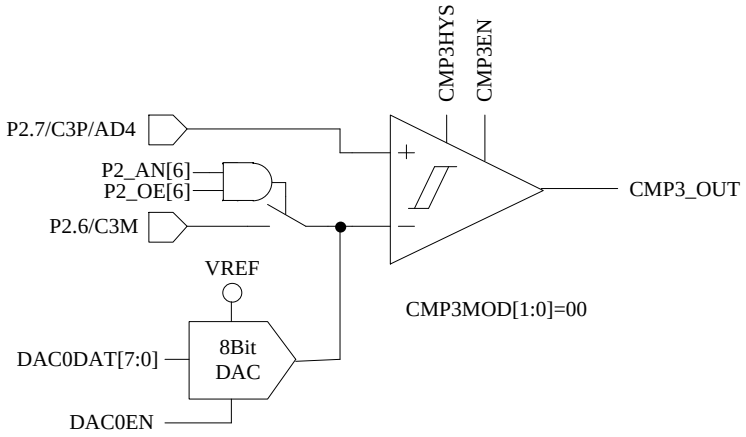


Figure 30-2 Single-comparator Input Mode

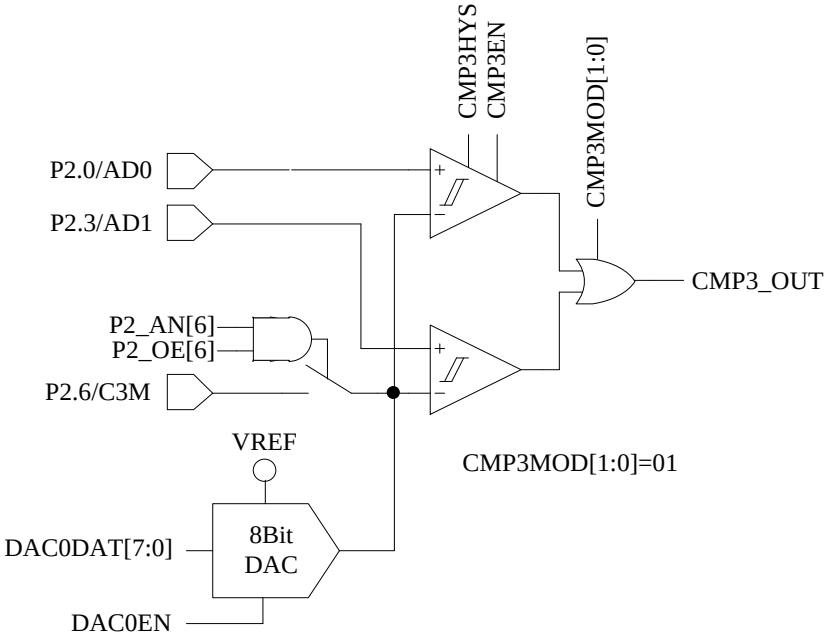


Figure 30-3 Dual-comparator Input Mode

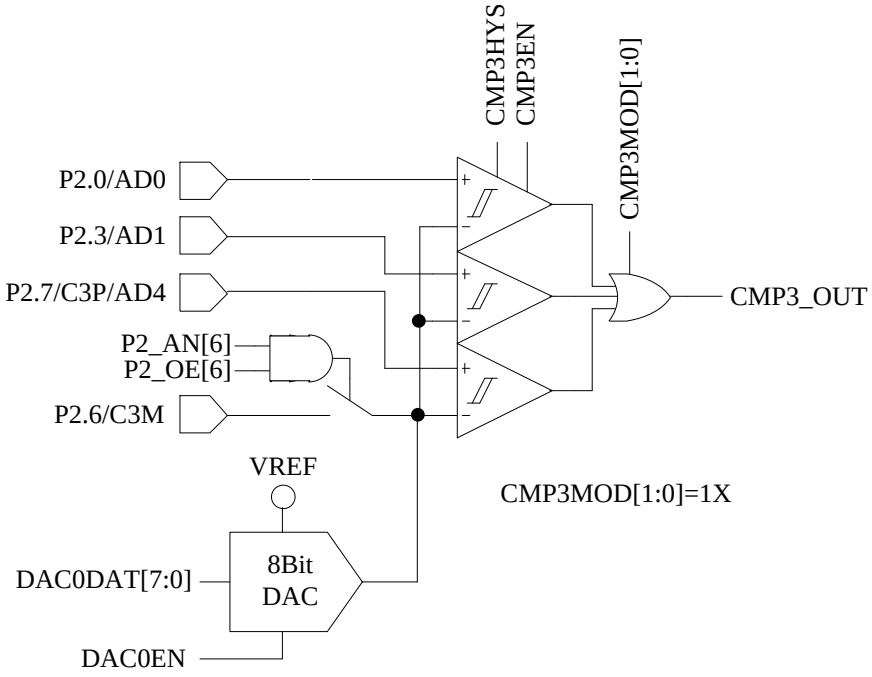


Figure 30-4 Triple-comparator Input Mode

30.1.1.1 Over-current Protection (OCP) on Bus

When an over-current protection signal is generated on bus, DRV_OUT[MOE] is automatically cleared to output idle voltage to stop motor drive for chip and motor protection. OCP feature is enabled when EVT_FILT[MOEMD] = 01, which automatically turns off the output and generates an OCP interrupt request if the current exceeds the threshold. When EVT_FILT[MOEMD]=00, the output is not automatically turned off if the current exceeds the threshold. However, an OCP request is generated by the hardware.

The source of OCP interrupt on bus can be selected by configuring EVT_FILT[EFSRC], namely CMP3 interrupt or external interrupt INT0 (P0.0 is connected to output FALUT signal of the IPM module). When EVT_FILT[EFSRC] = 1, TCON[IT0] bit is programmed to select the trigger edge of the external interrupt INT0 which generates an OCP signal. At this time, the source of OCP interrupt is INT0. When EVT_FILT[EFSRC] = 0 and bus current exceeds the threshold, CMP3 generates an OCP signal. At this time, the source of OCP interrupt is CMP3.

Configuring EVT_FILT[EFDIV]!= 0 enables the filtering of interrupt signals for OCP, and programming EVT_FILT[EFDIV] = 01/10/11 selects filter width of 4/8/16 clock cycles. When the filtering feature is enabled, the filtered signal is delayed by 4 ~ 5/8 ~ 9/16 ~ 17 clock cycles compared to the signal before filtering.

30.1.1.2 Cycle-by-cycle Current Limiting

The cycle-by-cycle current limiting feature is applied to square-wave-based drive control of BLDC

motors. When EVT_FILT[MOEMD] = 10, MOE is automatically enabled to turn off outputs and DRV_OUT[MOE] is automatically enabled upon DRV timer overflow events. When EVT_FILT[MOEMD] = 11, MOE is automatically enabled to turn off outputs and DRV_OUT[MOE] is automatically enabled upon DRV timer overflow/underflow events or every 5 μs.

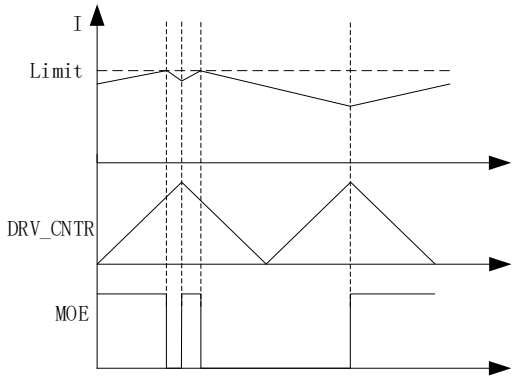


Figure 30-5 Cycle-by-cycle Current Limiting Waveform at EVT_FILT[MOEMD] = 10

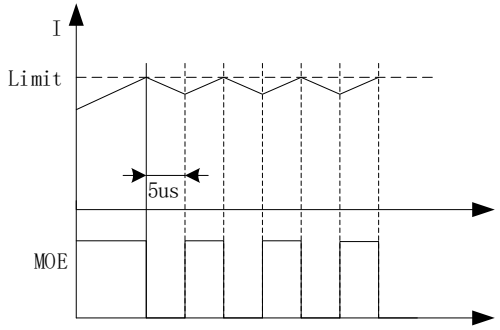


Figure 30-6 Cycle-by-cycle Current Limiting Waveform at EVT_FILT[MOEMD] = 11

30.1.2 Comparator CMP4

CMP4 is a hysteresis comparator, as shown in Figure 30-7. CMP4OUT can be read by software or reversed on external interrupt INT0. When CMP3 is used for cycle-by-cycle current limiting protection, CMP4 is used for bus current protection. After bus current OCP feature of CMP4 is triggered, output must be turned off by software.

CMP4 configurations are as follows:

1. The VREF source can be on-chip DAC1 output voltage or external circuit input voltage. Select DAC1 output, and place an external capacitor between P2.3 pin and GND (the recommended capacitance value is 100pF, and the output voltage stabilizes after DAC1 output for a period of time);
2. Configure P2_AN[7] = 1 to assign P2.7 pin to analog signal;
3. Configure CMP_CR2[CMP4EN] = 1 to enable CMP4;
4. Clear INT0 flag bit to enable INT0;

5. Set LVSR[EXT0CFG] = 111 to select CMP4 as the source of INT0;
6. Configure TCON[IT0] = 01 to select falling edge triggered INT0.

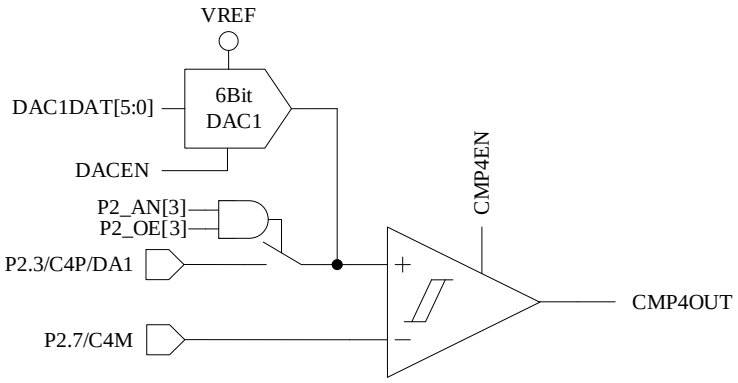


Figure 30-7 Schematic Diagram of CMP4 Module

30.1.3 Comparator CMP5

CMP5 is a hysteresis comparator, as shown in Figure 30-8. CMP5OUT can be read by software. When CMP5 is used for cycle-by-cycle current limiting protection, PFC module is used for cycle-by-cycle current limiting or current protection.

CMP5 configurations are as follows:

1. Configure P1_AN[3] = 1 and P1_AN[5] = 1 to assign P1.3/C5P and P1.5/C5M pins to analog signal;
2. Configure CMP_CR4[CMP5EN] = 1 to enable CMP5.

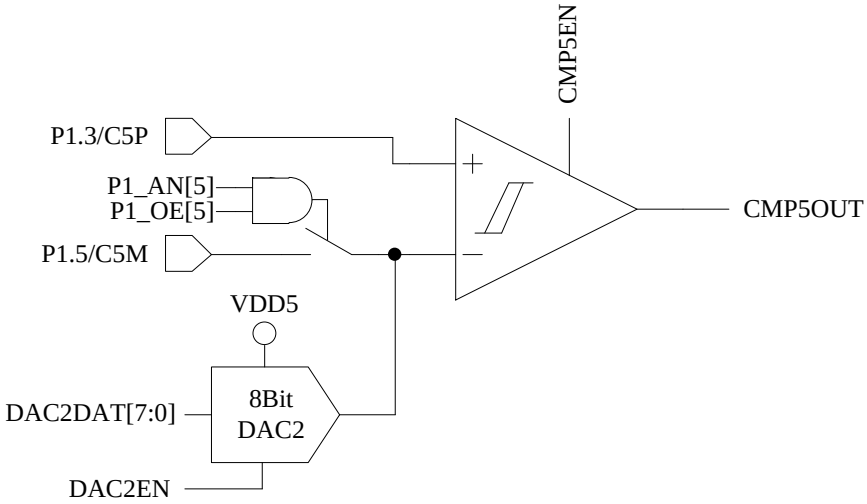


Figure 30-8 Schematic Diagram of CMP5 Module

30.1.4 Comparator Group (CMPG)

Comparator Group (CMPG) is a collection of CMP0, CMP1 and CMP2, with multiple comparison modes for different applications.

When CMP_CR2[CMP0MOD] = 00, CMPG works in the mode of three comparators without built-in

resistor. The I/O pins are shown in Figure 30-9. It is used for BEMF detection with the external virtual neutral point resistors. Negative inputs of the three comparators are connected together to P1.5 pin, and positive inputs are connected to P1.4, P1.6 and P2.1 respectively. The outputs are CMP0OUT, CMP1OUT and CMP2OUT respectively. The number of comparators working in this mode is defined by CMP_CR2[CMP0SEL]. When CMP_CR2[CMP0SEL] = 00, CMP0, CMP1 and CMP2 work simultaneously, which is the recommended configuration. When CMP_CR2[CMP0SEL] = 01, only CMP0 works. When CMP_CR2[CMP0SEL] = 10, only CMP1 works. When CMP_CR2[CMP0SEL] = 11, only CMP2 works.

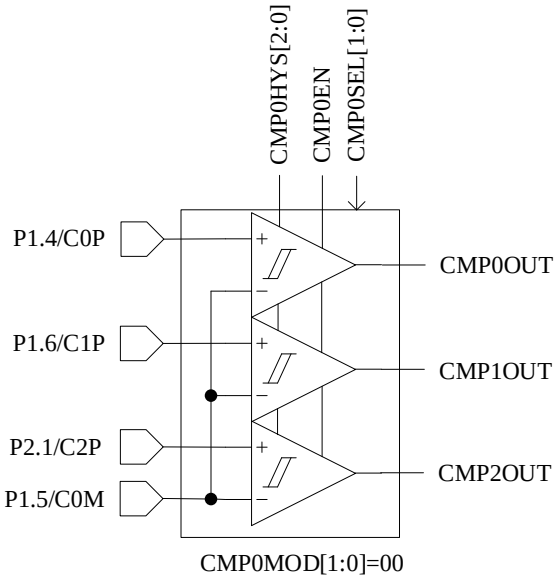


Figure 30-9 CMPG Mode with Built-in Three Comparators

(without Built-in Resistor)

When CMP_CR2[CMP0MOD] = 01, CMPG works in the mode of three comparators with built-in resistors. It is used for BEMF detection with the internal virtual neutral point resistors. The input port is selected by setting the function switching bit CMP_CR4[CMP0FS]. The number of comparators operating in this mode is defined by CMP_CR2[CMP0SEL]. When CMP_CR2[CMP0SEL] = 00, CMP0, CMP1 and CMP2 work simultaneously, which is the recommended configuration. When CMP_CR2[CMP0SEL] = 01, only CMP0 works. When CMP_CR2[CMP0SEL] = 10, only CMP1 works. When CMP_CR2[CMP0SEL] = 11, only CMP2 works.

When CMP_CR4[CMP0FS] = 0, the I/O pins are shown in Figure 30-10. Negative inputs of the three comparators are connected together to the center point of the built-in resistor. Positive inputs are connected to P1.4, P1.6 and P2.1 respectively, and the outputs are CMP0OUT, CMP1OUT and CMP2OUT respectively.

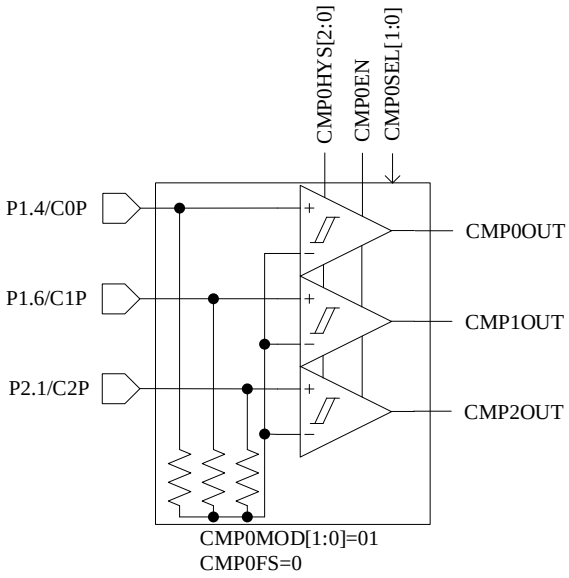


Figure 30-10 CMPG Mode with Built-in Three Comparators and Resistors

(without Functional Switching)

When CMP_CR4[CMP0FS] = 1, the I/O pins are shown in Figure 30-11. Negative inputs of the three comparators are connected together to the center point of the built-in resistor. Positive inputs are connected to P1.4, P1.3 and P1.5 respectively, and the outputs are CMP0OUT, CMP1OUT and CMP2OUT respectively.

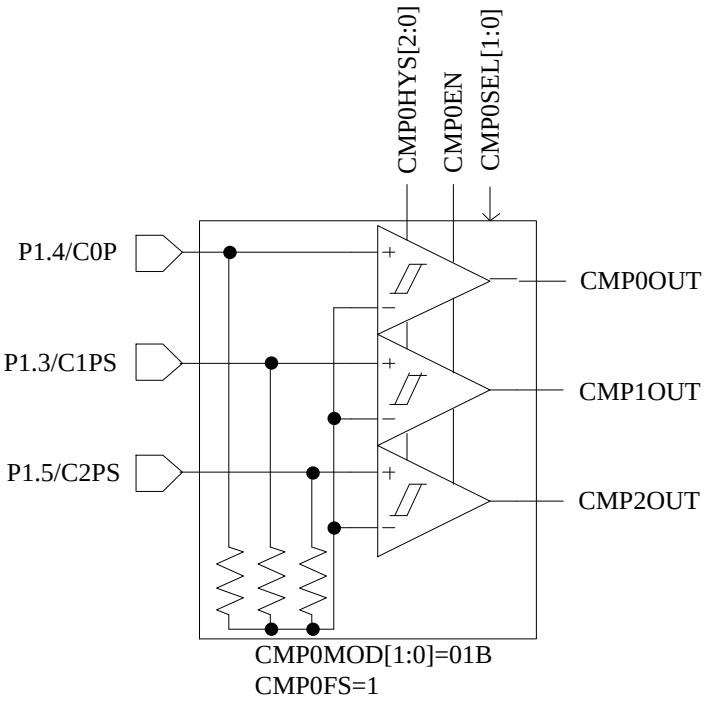


Figure 30-11 CMPG Mode with Built-in Three Comparators (with Functional Switching)

When CMP_CR2[CMP0MOD] = 10, CMPG mode with three differential comparators is selected for

the differential Hall sensor to detect the motor rotor position. The input and output pins are shown in Figure 30-12. Negative inputs of the three comparators are respectively connected to P1.5, P1.7 and P2.2, and positive inputs are respectively connected to P1.4, P1.6 and P2.1. The outputs are CMP0OUT, CMP1OUT and CMP2OUT respectively. The number of comparators working in this mode is defined by CMP_CR2[CMP0SEL]. When CMP_CR2[CMP0SEL] = 00, CMP0, CMP1 and CMP2 work simultaneously, which is the recommended configuration. When CMP_CR2[CMP0SEL] = 01, only CMP0 works. When CMP_CR2[CMP0SEL] = 10, only CMP1 works. When CMP_CR2[CMP0SEL] = 11, only CMP2 works.

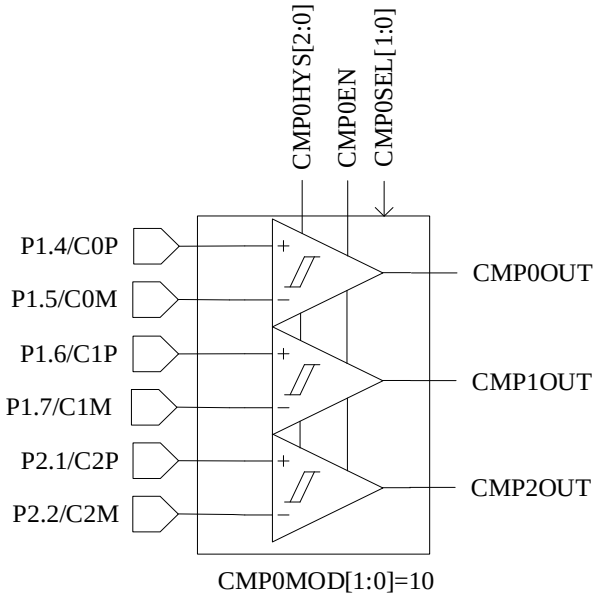


Figure 30-12 CMPG Mode with Three Differential Comparators

When CMP_CR2[CMP0MOD] = 11, CMPG mode with two comparators is selected for motor speed detection. The I/O pins are shown in Figure 30-13. Negative inputs of the two comparators are connected together to P1.5, and positive inputs are connected to P1.4 and P1.3 respectively. The outputs are CMP0OUT and CMP1OUT respectively. The number of comparators in this mode is defined by CMP_CR2[CMP0SEL]. When CMP_CR2[CMP0SEL] = 00, CMP0 and CMP1 work simultaneously, which is the recommended configuration. When CMP_CR2[CMP0SEL] = 01, only CMP0 works. When CMP_CR2[CMP0SEL] = 10, only CMP1 works.

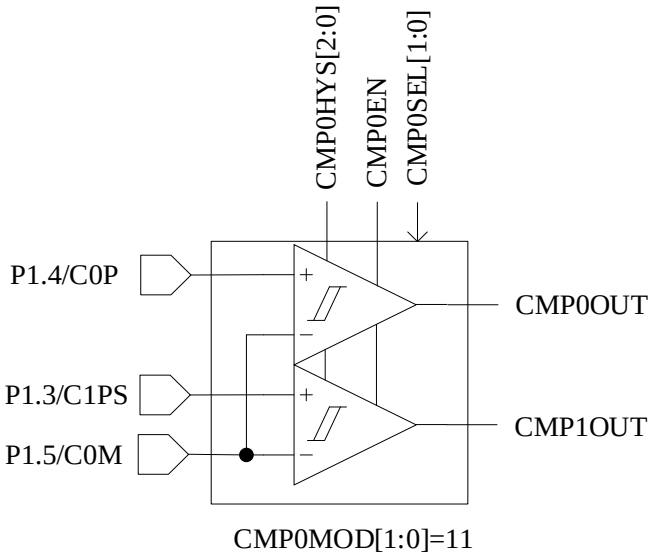


Figure 30-13 CMPG Mode with Two Comparators

The output signals of CMP0/CMP1/CMP2 are sent to Timer1 after filtering and sampling modules.

30.1.5 Comparator Sampling

The comparator sampling feature is mainly used for the square-wave control and RSD (tailwind/headwind detection), which eliminates the switching interference from driving circuit. See section Sampling for square-wave control and section RSD Comparator Sampling for RSD.

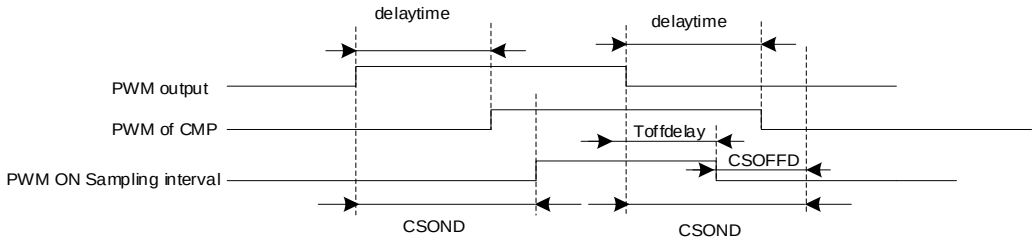


Figure 30-14 PWM ON Sampling Mode

There is a delay from the PWM output to the output of the comparator, which is mainly affected by the following factors: resistance value of drive resistor, switching speed of the power device, and input delay and hysteresis settings of the comparator. As shown in Figure 30-14, the delay-time is from the chip output to the comparator output. When high-level sampling is performed, the sampling interval shall be enveloped by actual high-level output of the comparator. First, the sampling ON-delayed time CMP_SAMR[CSOND] is set to overcome the output delay and the oscillation interval of the power device. At the end of the sampling interval, CMP_SAMR[CSOND] is delayed after the falling edge of PWM, at which time the actual sampling

window has exceeded the corresponding high-level interval. The sampling OFF-lead time CMP_SAMR[CSOFFD] is set to stop sampling Toffdelay after the PWM output falling edge, where Toffdelay = CMP_SAMR[CSOND] – CMP_SAMR[CSOFFD]. By configuring CMP_SAMR[CSOND] and CMP_SAMR[CSOFDD], the sampling interval can be located in the high-level interval of the actual output of the comparator.

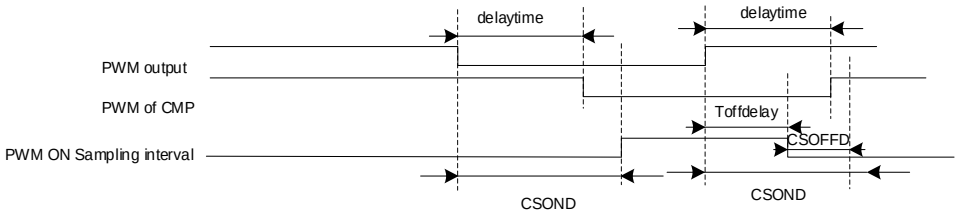


Figure 30-15 PWM OFF Sampling Mode

Similarly, when low-level sampling is performed, the sampling ON-delayed time CMP_SAMR[CSOND] and the sampling OFF-lead time CMP_SAMR[CSOFFD] are set reasonably to ensure that the actual sampling interval is located in the low-level output interval of the comparator.

Method for measuring the delay of PWM output to comparator: Set CMP_CR3[SAMSEL] = 00 to disable the comparator sampling delay feature. Set CMP_CR3[CMPSEL] to select the corresponding comparator output to test pin P0.7. Enable the PWM output and comparator, manually rotate the motor to change the comparator value, and measure the delay between the PWM output and the comparator output.

30.1.6 Comparator Output

CMP_CR3[CMPSEL] is configured to output results of one comparator to P0.7.

30.2 Comparator Registers

30.2.1 CMP_CR0 (0xD5)

Bit	7	6	5	4	3	2	1	0
Name	CMP3IM		CMP2IM		CMP1IM		CMP0IM	
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]	CMP3IM	CMP3 Interrupt Mode See descriptions on CMP_CR0[CMP0IM].
[5:4]	CMP2IM	CMP2 Interrupt Mode See descriptions on CMP_CR0[CMP0IM].
[3:2]	CMP1IM	CMP1 Interrupt Mode See descriptions on CMP_CR0[CMP0IM].
[1:0]	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 both rising/falling edges.

30.2.2 CMP_CR1 (0xD6)

Bit	7	6	5	4	3	2	1	0
Name	HALLSEL	CMP3MOD		CMP3EN	CMP3HYS	CMP0HYS		
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]	HALLSEL	Hall Input Selection 0: P0.2/P3.7/P3.6 1: P1.4/P1.6/P2.1
[6:5]	CMP3MOD	CMP3 Mode Selection Negative input is connected to P2.6 or DAC0 output 00: Single-comparator input mode, where P2.7 is connected to the positive input, as shown in Figure 30-2. 01: Dual-comparator input mode, where P2.0 and P2.3 are connected to the positive input, as shown in Figure 30-3. 1X: Three-comparator input mode, where P2.0, P2.3 and P2.7 are connected to the positive input, as shown in Figure 30-4.
[4]	CMP3EN	CMP3 Enable 0: Disable 1: Enable
[3]	CMP3HYS	CMP3 Hysteresis Voltage Selection 0: No hysteresis 1: Hysteresis voltage is selected
[2:0]	CMP0HYS	CMP0/1/2 Hysteresis Voltage Selection 000: No hysteresis 001: ±2.5mV 010: -5MV 100: +5mV 011: ±5mV 101: -10mV 110: +10mV 111: ±10mV

30.2.3 CMP_CR2 (0xDA)

Bit	7	6	5	4	3	2	1	0
Name	CMP4EN	CMP0MOD		CMP0SEL		RSV		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]	CMP4EN	CMP4 Enable 0: Disable 1: Enable
[6:5]	CMP0MOD	CMPG Mode Setting 00: CMPG Mode with built-in three comparators (without built-in resistor), as shown in Figure 30-9. 01: CMPG Mode with built-in three comparators and resistors, where function switching is selected by configuring CMP_CR4[CMP0FS], as shown in Figure 30-10 and Figure 30-11. 10: CMPG Mode with three differential comparators, as shown in Figure 30-12. 11: CMPG Mode with two comparators , where only CMP0 and CMP1 work, as shown in Figure 30-13.
[4:3]	CMP0SEL	CMPG Pin Combination Selection, used with CMP_CR2[CMP0MOD] bit. It is set to 00 by default. In square-wave drive application, TIM1_DBRx[T1CPE] automatically controls CMP_CR2[CMP0SEL] to enable or disable each comparator. Table 30-1 Function Description of CMPG Port and CMP_CR2[CMP0MOD] Combination

CMP0MOD	CMP0SEL	Description
00	00	CMP0/1/2 work simultaneously, as shown in Figure 30-9. The negative input of these comparators are connected to C0M. The hardware automatically compares the positive inputs C0P, C1P and C2P with C0M, and the output results are transferred to CMP0OUT, CMP1OUT and CMP2OUT respectively.
	01	Only CMP0 works. The positive input is connected to C0P, and the negative input to C0M. The output results are transferred to CMP0OUT.
	10	Only CMP1 works. The positive input is connected to C1P, and the negative input to C0M. The output results are transferred to CMP1OUT.
	11	Only CMP2 works. The positive input is connected to C2P, and the negative input to C0M. The output results are transferred to CMP2OUT.
01	00	CMP0/1/2 work simultaneously, as shown in Figure 30-10 and Figure 30-11. The negative inputs of these 3 comparators are connected to the center of built-in resistor. When CMP_CR4[CMP0FS] = 0, the hardware automatically compares the positive inputs C0P, C1P and C2P with C0M. When CMP_CR4[CMP0FS] = 1, the hardware automatically compares the positive inputs C0P, C1PS and C2PS with C0M. The output results are transferred to CMP0OUT, CMP1OUT and CMP2OUT respectively.

				01	Only CMP0 works. The positive input is connected to C0P, and the negative input to the center of BEMF built-in resistor. The output results are transferred to CMP0OUT.
				10	Only CMP1 works. When CMP_CR4[CMP0FS] = 0, the positive input is connected to C1P, and when CMP_CR4[CMP0FS] = 1, it is connected to C1PS. The negative input is connected to the center of BEMF built-in resistor. The output results are transferred to CMP1OUT.
				11	Only CMP2 works. When CMP_CR4[CMP0FS] = 0, the positive input is connected to C2P, and when CMP_CR4[CMP0FS] = 1, it is connected to C2PS. The negative input is connected to the center of BEMF built-in resistor. The output results are transferred to CMP2OUT.
			10	00	CMP0/1/2 work simultaneously, as shown in Figure 30-12. The positive inputs of these comparators are connected to C0P, C1P and C2P respectively, and the negative inputs are connected to C0M, C1M and C2M respectively. The output results are transferred to CMP0OUT, CMP1OUT and CMP2OUT respectively.
				01	Only CMP0 works. The positive input is connected to C0P, and the negative input to C0M, and the output results are transferred to CMP0OUT.
				10	Only CMP1 works. The positive input is connected to C1P, and the negative input to C1M. The output results are transferred to CMP1OUT.
				11	Only CMP2 works. The positive input is connected to C2P, and the negative input to C2M. The output results are transferred to CMP2OUT.
			11	00	CMP0/1 work simultaneously, as shown in Figure 30-13. The positive inputs are connected to C0P and C1PS respectively, and the negative inputs to C0M. The outputs results are transferred to CMP0OUT and CMP1OUT respectively.
				01	Only CMP0 works. The positive input is connected to C0P, and the negative input to C0M. The output results are transferred to CMP0OUT.
				10	Only CMP1 works. The positive input is connected to C1PS, and the negative input to C0M. The output results are transferred to CMP1OUT.
				11	Reserved
[2:1]	RSV	Reserved			
[0]	CMP0EN	CMP0 Enable 0: Disable 1: Enable			

30.2.4 CMP_CR3 (0xDC)

Bit	7	6	5	4	3	2	1	0
Name	CMPDTEN	DBGSEL		SAMSEL		CMPSEL		
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]	CMPDTEN	Comparator Deadtime Sampling Enable 0: Disable 1: Enable
[6:5]	DBGSEL	Debug Output Selection, connected to P0.1 pin 00: Debug Output Disable 01: Freewheeling shielding is completed and ZCP signal is detected 10: ADC Trigger Signal 11: Comparator Sampling Interval
[4:3]	SAMSEL	Sampling delay enable of CMP0, CMP1, CMP2 and ADC in PWM ON/OFF modes 00: Sampling at both PWM ON and OFF modes without time delay 01: Sampling at PWM OFF mode, with time delay according to CMP_SAMR 10: Sampling at PWM ON mode, with time delay according to CMP_SAMR 11: Sampling at both PWM ON and OFF, with time delay according to CMP_SAMR
[2:0]	CMPSEL	Comparator Output Selection Output signals of one selected comparator to P0.7. 000: No output 001: CMP0 010: CMP1 011: CMP2 100: CMP3 101: CMP4 110: CMP5 111: Omega Start Flag (Estimator Output Angle Flag, see section 14.1.9.3 for details)

30.2.5 CMP_CR4 (0xE1)

Bit	7	6	5	4	3	2	1	0
Name	CMP4OUT	CMP5OUT	RSV				CMP0FS	CMP5EN
Type	R	R	-	-	-	-	R/W	-
Reset	1	0	-	-	-	-	0	-

Bit	Name	Description
[7]	CMP4OUT	CMP4 comparison result
[6]	CMP5OUT	CMP5 comparison result
[5: 2]	RSV	Reserved
[1]	CMP0FS	CMP1/CMP2 Function Switching 0: Disable. See Figure 30-10 1: Enable. It is valid only when CMP_CR2[CMP0_MOD] = 01, as shown in Figure 30-11.
[0]	CMP5EN	Comparator CMP5 Enable 0: Disable 1: Enable

30.2.6 CMP_SAMR (0x40AD)

Bit	7	6	5	4	3	2	1	0
Name	CSOND				CSOFFD			
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	1

Bit	Name	Description
[7:4]	CSOND	CMP0/CMP1/CMP2 ON-delayed Sampling Time When PWM module switches from OFF to ON or from ON to OFF, turn-on/off of the power IC affects input signal of the comparator. In this case, CMP_SAMR[CSOND] is configured to delay the sampling of CMP0/CMP1/CMP2. The delay generated by drive circuit shall be taken into account when CMP_SAMR[CSOND] is calculated. If MCU clock runs at 24MHz(41.67ns), ON-delayed Sampling Time = CSOND x 41.67 x 8ns Notes: ■ CMP_SAMR[CSOND] must be greater than or equal to CMP_SAMR[CSOFFD]. ■ See section Sampling for BLDC drive application. ■ See section RSD Comparator Sampling for RSD application.
[3:0]	CSOFFD	CMP0/CMP1/CMP2 OFF-delayed Sampling Time When PWM module switches from OFF to ON or from ON to OFF, turn-on/off of the power IC affects input signal of the comparator. In this case, CMP_SAMR[CSOFFD] is configured to reduce comparator interference. If MCU clock runs at 24MHz(41.67ns), OFF-delayed Sampling Time = CSOFFD x 41.67 x 8ns Notes: ■ CMP_SAMR[CSOND] must be greater than or equal to CMP_SAMR[CSOFFD]. ■ See section Sampling for BLDC drive application. ■ See section RSD Comparator Sampling for RSD application.

30.2.7 CMP_SR (0xD7)

Bit	7	6	5	4	3	2	1	0
Name	CMP3IF	CMP2IF	CMP1IF	CMP0IF	CMP3OUT	CMP2OUT	CMP1OUT	CMP0OUT
Type	R/W0	R/W0	R/W0	R/W0	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	CMP3IF	CMP3 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	CMP2IF	CMP2 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect

[5]	CMP1IF	CMP1 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]	CMP3OUT	CMP3 comparison result
[2]	CMP2OUT	CMP2 comparison result
[1]	CMP1OUT	CMP1 comparison result
[0]	CMP0OUT	CMP0 comparison result

30.2.8 EVT_FILT (0xD9)

Bit	7	6	5	4	3	2	1	0
Name	TSDEN	TSDADJ		MOEMD		EFSRC	EFDIV	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	1	0	0	0	0	0

Bit	Name	Description
[7]	TSDEN	TSD Enable 0: Disable 1: Enable
[6:5]	TSDADJ	TSD Temperature Selection 00: 105℃ 01: 120℃ 10: 135℃ 11: 150℃
[4:3]	MOEMD	MOE Cleared and Enabled by Hardware MOE is cleared and enabled by hardware upon over-/under-current protection event. 00: MOE is not automatically cleared. 01: MOE is automatically cleared. 10: MOE is automatically cleared and enabled by hardware upon DRV timer overflow events (for square-wave drive). 11: MOE is automatically cleared and enabled automatically upon DRV timer overflow/underflow events or every 5 μs (for square-wave drive).
[2]	EFSRC	Input Source of Filtering Module Upon Current Protection Event 0: CMP3 interrupt 1: INT0(P0.0) interrupt
[1:0]	EFDIV	Filter Width for Current Protection 00: Disable 01: 4 system clock cycles 10: 8 system clock cycles 11: 16 system clock cycles

31 Power Supply

31.1 LDO

31.1.1 LDO Operations

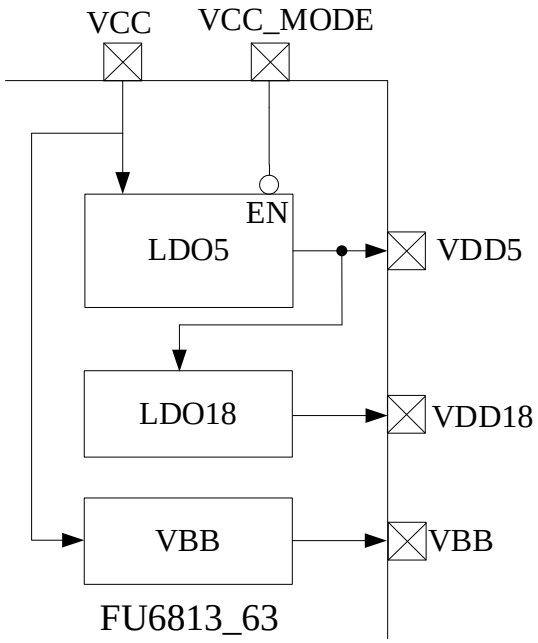


Figure 31-1 Functional Block Diagram of Power Supply

The I/O pins of LDO module is shown in Figure 31-1. The LDO module converts the input supply voltage to 5V (VDD5) and 1.8V (VDD18) as the power supply for built-in analog and digital modules. Internal LDO5 or external supply for VDD5 is selected by configuring VCC_MODE. VBB is enabled or disabled by IDE. As shown in Figure 31-2, VBB is enabled by default to power pre-driver. In this case, VBB = 15V if VCC > 15V and VBB = VCC if VCC ≤ 15V. When VBB is disabled, VBB has no voltage output.

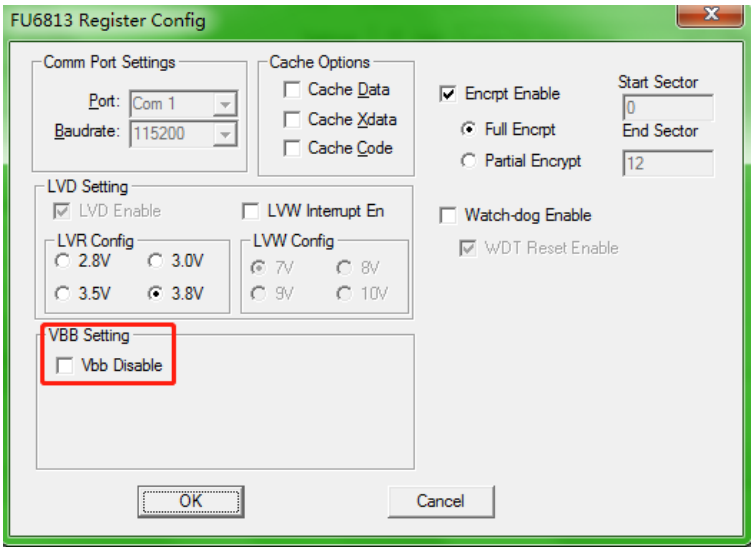


Figure 31-2 VBB Setting

- FU6813
  - High-voltage Single-power Supply Mode (VCC_MODE = 0): VCC = 5 ~ 24V. See Figure 31-3 for details.
  - Dual-power Supply Mode (VCC_MODE = 1): VCC≥VDD5, VCC = 5 ~ 36V and VDD5 = 5V. See Figure 31-4 for details.
  - Low-voltage Single-power Supply Mode(VCC_MODE = 1): VCC = VDD5 = 3 ~ 5.5V. See Figure 31-5 for details.
- FU6863
  - Mode 1 (VCC_MODE = 0): VCC = 5V ~ 24V and VDRV = 7V ~ 18V
  - Mode 2 (VCC_MODE = 1): VCC = VDD5 = 3V ~ 5.5V and VDRV = 7V ~ 18V

Note: When VCC_MODE = 1, VDD5 voltage is supplied.

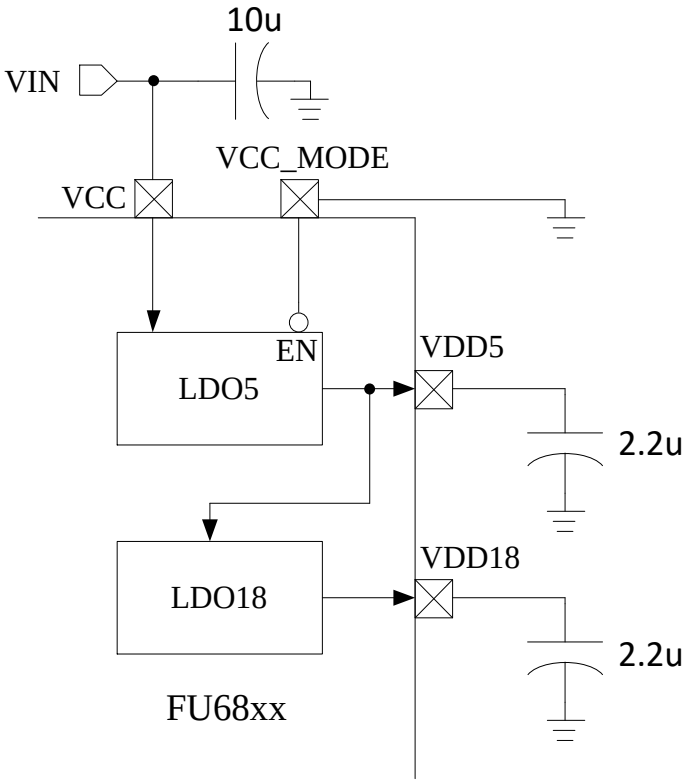


Figure 31-3 High-voltage Single-power Supply Mode

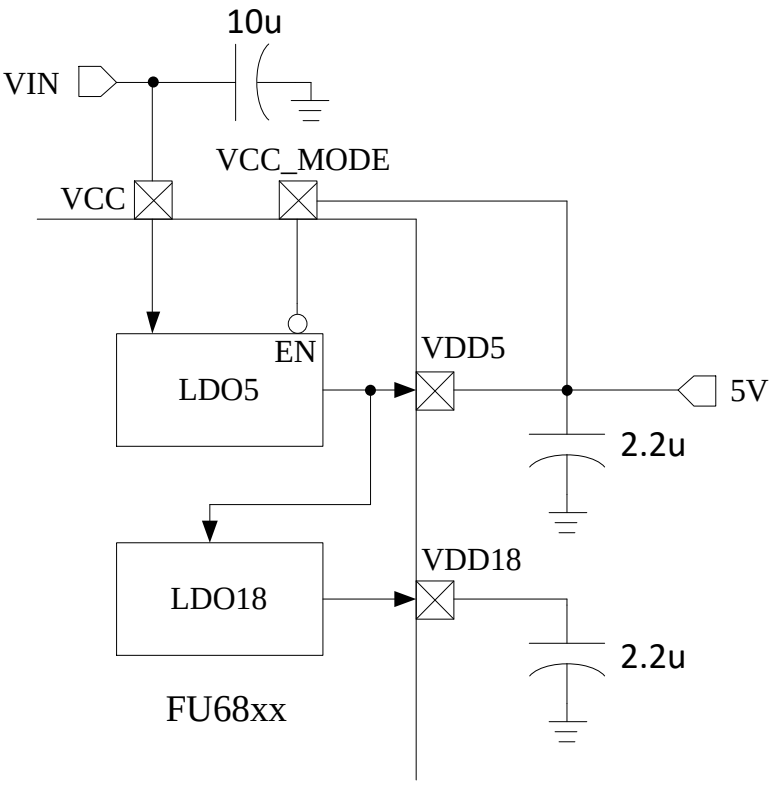


Figure 31-4 Dual-power Supply Mode

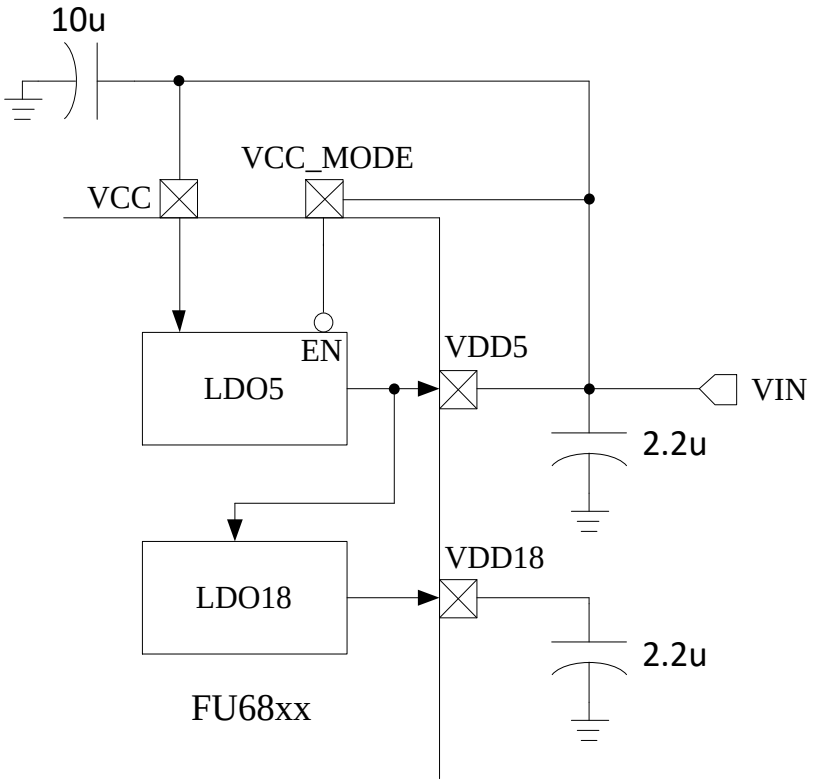


Figure 31-5 Low-voltage Single-power Supply Mode

31.2 Low Voltage Detector (LVD)

31.2.1 LVD Operations

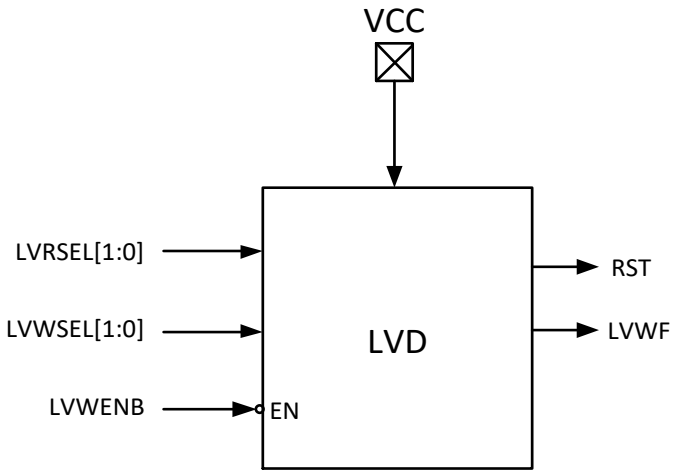


Figure 31-6 LVD Module

Configuring CCFG1[LVDENB] = 0 enables LVD module. Low voltage warning threshold voltage is selected by CCFG2[LVWSEL]. The low voltage reset module is always enabled, and low voltage reset threshold voltage is selected by CCFG2[LVRSEL].

31.2.2 CCFG2 (0x401D)

Bit	7	6	5	4	3	2	1	0
Name	LVRSEL		WDTBTEN	WDTRSTEN	EOSRSTEN	EOSGATEN	LVWSEL	
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]	LVRSEL	Low Voltage Reset Threshold Voltage. VDD5 is detected for the low voltage reset. 00: Low voltage reset is enabled when VDD5 is lower than 2.8V. 01: Low voltage reset is enabled when VDD5 is lower than 3.0V. 10: Low voltage reset is enabled when VDD5 is lower than 3.5V. 11: Low voltage reset is enabled when VDD5 is lower than 3.8V.
[5]	WDTBTEN	WDT Boot Enable Upon Overflow Reset 0: Disable 1: Enable
[4]	WDTRSTEN	WDT Reset Enable 0: Disable 1: Enable
[3]	EOSRSTEN	EOS Reset Enable 0: Disable 1: Enable
[2]	EOSGATEN	EOS Gate Clock Enable
[1: 0]	LVWSEL	Low Voltage Warning Threshold Voltage. VCC is detected for the low voltage warning. 00: Low voltage warning is enabled when VCC is lower than 7V. 01: Low voltage warning is enabled when VCC is lower than 8V. 10: Low voltage warning is enabled when VCC is lower than 9V. 11: Low voltage warning is enabled when VCC is lower than 10V.

31.2.3 LVSR (0xDB)

Bit	7	6	5	4	3	2	1	0
Name	RSV		EXT0CFG			TSDF	LVWF	LVWIF
Type	-	-	R/W	R/W	R/W	R	R	R/W0
Reset	-	-	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5:3]	EXT0CFG	INT0 Pin Selection 000: P0.0 001: P0.1 010: P0.2 011: P0.3 100: P0.4 101: P0.5 110: P0.6 111: CMP4 Output
[2]	TSDF	Over Temperature State Indicator 0: The current temperature does not exceed the threshold 1: The current temperature exceeds the threshold Note: This flag bit often works with TSD interrupt flag TCON[5].
[1]	LVWF	VCC Low Voltage(LV) Flag

		This bit indicates whether the chip is in the low voltage state. 0: The chip is not in the LV warning state. 1: The chip is in the LV warning state.
[0]	LVWIF	VCC LV 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.

32 Flash

32.1 Flash Introduction

The chip provides 32k bytes of Flash space. It supports page erasure, page pre-programing and write.

Main features:

- 128 sectors in total, each with a size of 256 bytes
- Last sector (address range: 0x7F00~0x7FFF) cannot be erased at any time
- Programming is enabled when FLA_CR[FLAEN] is set to “1”, where page pre-programming, page erase or write and other Flash operations are activated with MOVX instructions.

32.2 Flash Operations

- Flash memory must be unlocked before erase and programming operations. The Flash software programming feature is activated after “0x5A” and “0x1F” are written to register FLA_KEY in sequence. If the sequence is incorrect or other values are written, Flash space is frozen until the next reset. After unlocking, any write to FLA_CR register causes the FLA_ KEY to be locked again.
- CRC results change if Flash memory is rewritten during program execution.
- Page pre-programming must be done before page erase.
- Configuring LA_CR = 0x03 enables page erase, FLA_CR = 0x05 enables page pre-programming and FLA_CR = 0x01 enables write operations.

Note: All interrupts must be disabled before self-programming to ensure the security of Flash operations and avoid mis-operation of Flash using MOVX instruction during interrupt processing.

32.3 Flash Registers

32.3.1 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 it is erased) 0: Disable 1: Enable Note: FLA_CR[FLAPRE] is valid only when FLA_CR[FLAEN] = 1.
[1]	FLAERS	Sector Erase Enable 0: Disable 1: Enable Note: FLA_CR[FLAERS] is valid only when FLA_CR[FLAEN] = 1.
[0]	FLAEN	Programming Enable 0: Disable 1: Enable

32.3.2 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 Flash operations; Write any value to FLA_CR bit to lock Flash operations.

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: Release

33 CRC

33.1 CRC Functional Block Diagram

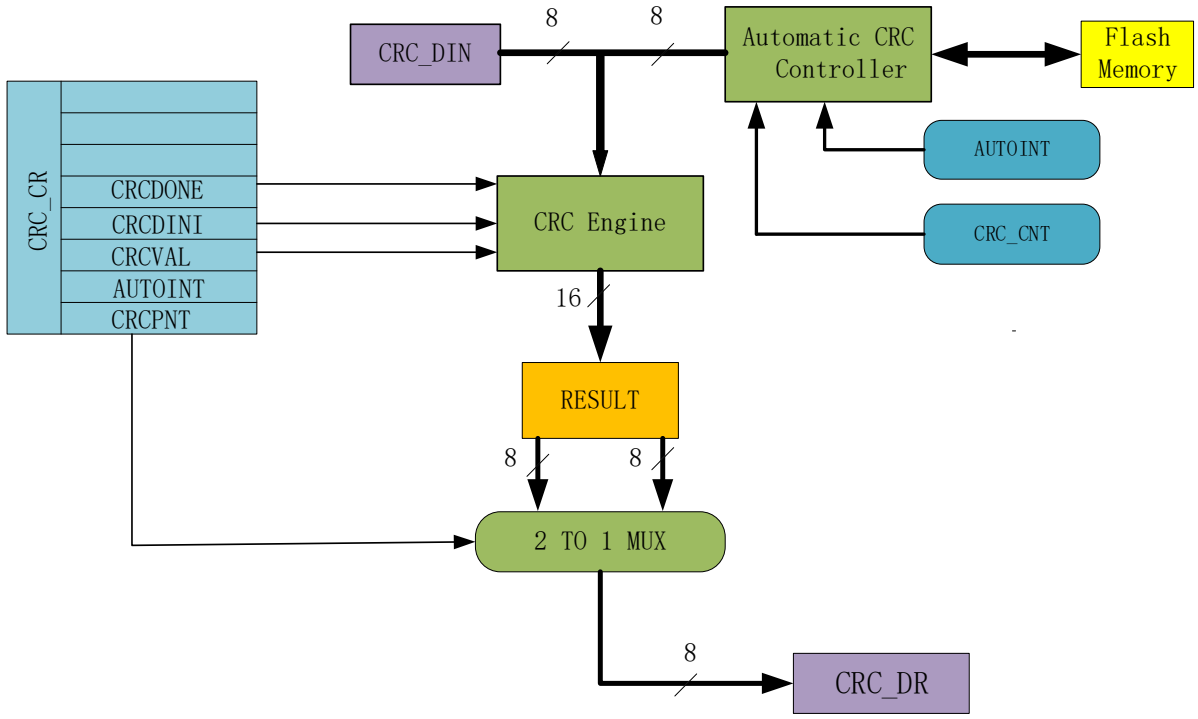


Figure 33-1 CRC Functional Block Diagram

CRC module outputs the result of CRC calculation for any 8-bit data based on a fixed polynomial. As shown in Figure 33-1, 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 33-1 CRC Criteria and Polynomials

No.	CRC Criteria	Polynomial	Hexadecimal Representation
1	CRC12	x^12+x^11+x^3+x^2+x+1	0x80F
2	CRC16	x^16+x^15+x^2+1	0x8005
3	CRC16/CCITT-FALSE	x^16+x^12+x^5+1	0x1021
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

33.2 CRC16 Polynomial

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

33.3 CRC16 Logic Diagram

The schematic diagram of CRC16 is shown in Figure 33-2. The chip implementation is based on parallel algorithm. For each input byte, MCU calculates the results within one system clock cycle.

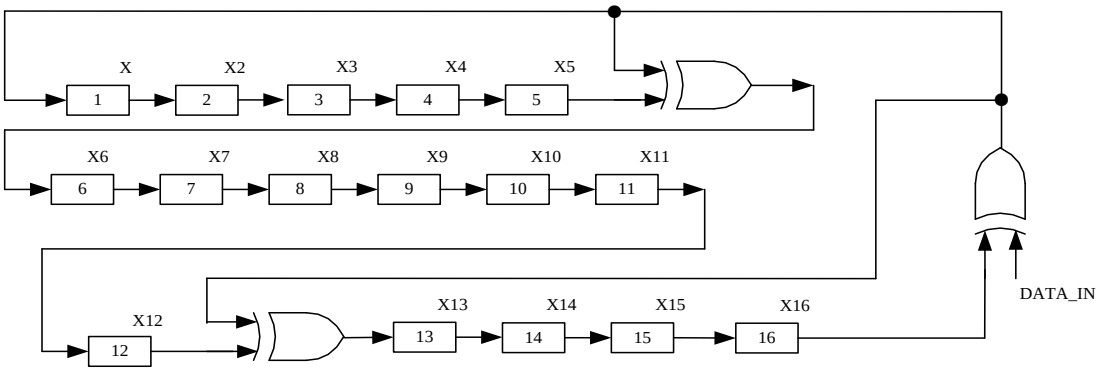


Figure 33-2 CRC16 Schematic Diagram

33.4 CRC Operations

33.4.1 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.

33.4.2 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.

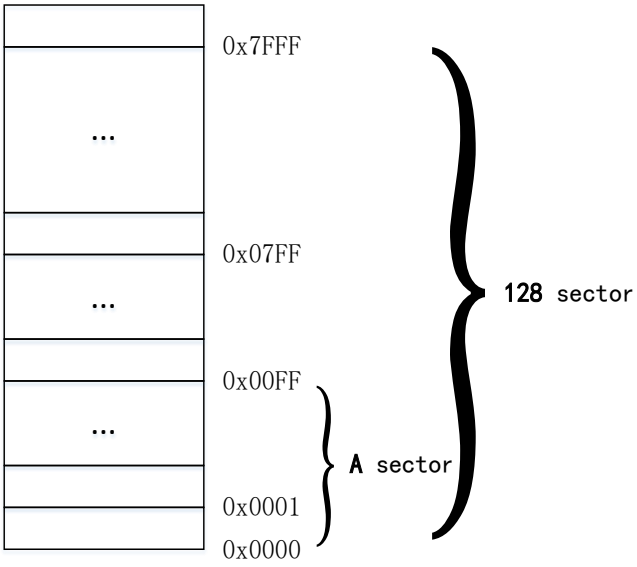


Figure 33-3 ROM Sectors

As shown in Figure 33-3, ROM contains 32k bytes and is divided into 128 sectors, numbered from sector0 to sector127. Each sector contains 256 bytes. For CRC calculation of sectors, the value of CRC_BEG (the starting sector) can be any value falling between 0x00 and 0xFF, including 0x00 and 0x7F. The CRC_CNT (total number of sectors to be calculated) can be any value between 0x00~0x7F, including 0x00 and 0xFF.

As CRC_BEG increases, CRC_CNT decreases accordingly. For example, if CRC_BEG is 0x7F, 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.

33.5 CRC Registers

33.5.1 CRC_CR (0x4022)

Bit	7	6	5	4	3	2	1	0
Name	RSV			CRCDONE	CRCDINI	CRCVAL	AUTOINT	CRCPNT
Type	-	-	-	R	R/W	R/W	R/W	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” 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 low-order 8 bits of the 16-bit CRC result 1: Read CRC_DR to access high-order 8 bits of the 16-bit CRC result

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

33.5.2 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.

33.5.3 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, CRC_CR[CRCPNT] determines to access high-order or low-order 8 bits of the CRC result.

Note: In addition to software, the value of this register can be changed by other signals. In this case, the register is placed inside CRC module, instead of the register-specific module.

33.5.4 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 Automatic CRC Calculation Example: If CRC_BEG is set to “1”, CRC calculation starts from location 1*256 = 256, or rather from the first byte of sector 2.

33.5.5 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.

34 Sleep Mode

34.1 Introduction

The chip operates in three modes: normal mode, standby mode and sleep mode. 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 34-1.

Table 34-1 Power Consumption Modes

Power Mode	Description	Wakeup 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 other functional modules are either enabled or disabled depending on their control bit setting. WDT stops.	Any interrupt; Reset/Debug on external interrupt	Low power performance with flexible performance
Sleep	Flash Deep Sleep. The analog fast clock circuit is disconnected and MCU software shall ensure that ADC, FOC, and driver modules are idle before the chip enters the Sleep Mode. WDT is disabled.	External interrupt; RTC interrupt; Reset/Debug on external interrupt	Extremely low power performance with flexible performance

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

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

34.2 Sleep Mode Register

34.2.1 PCON(0x87)

Bit	7	6	5	4	3	2	1	0
Name	RSV		GF3	GF2	GF1	LDOM	STOP	IDLE
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]	GF3	General-purpose flag bit 3
[4]	GF2	General-purpose flag bit 2
[3]	GF1	General-purpose flag bit 1
[2]	LDOM	0: LDO5 works normally when “1” is written to the bit STOP (sleep mode). 1: LDO5 works in low-power consumption mode when “1” is written to the bit STOP (sleep mode).
[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: {STOP, IDLE} = 1x, the system idis in sleep mode. {STOP, IDLE} = 01, the system is in standby mode. {STOP, IDLE} = 00, the system is in normal mode.

35 Code Protection

35.1 Introduction

The chip supports 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.

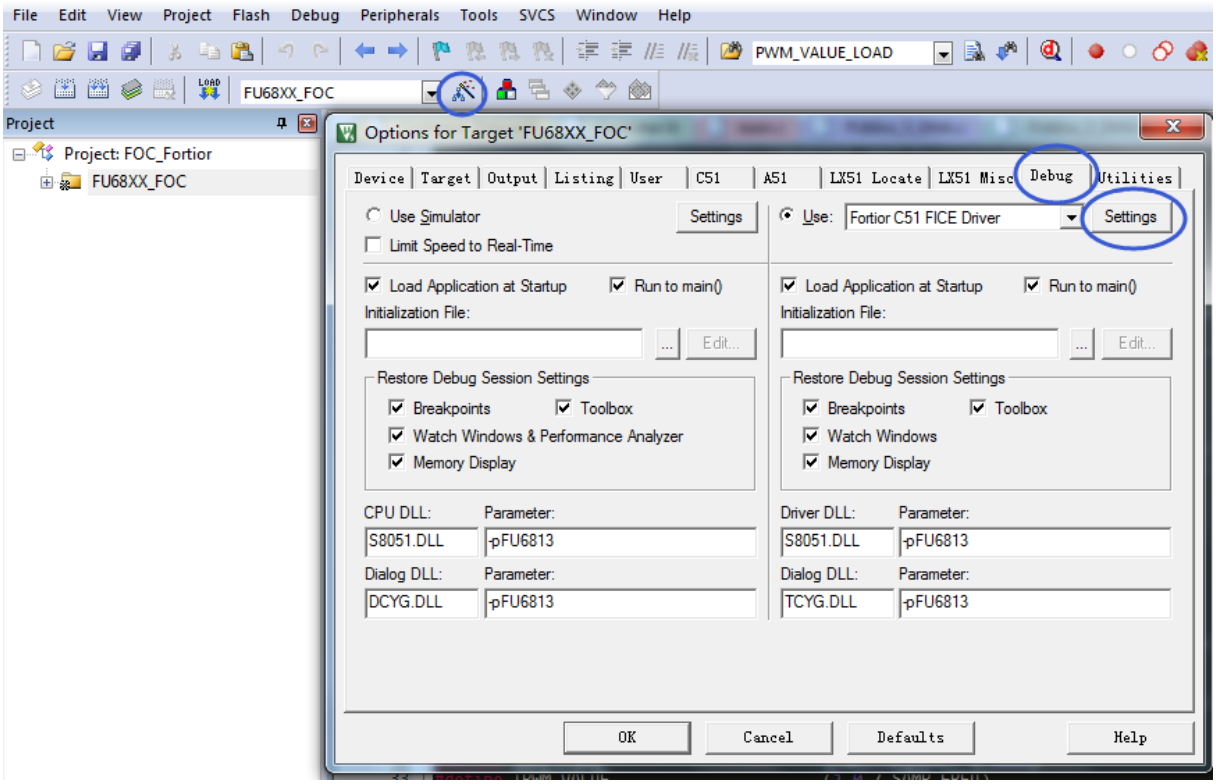


Figure 35-1 Code Protection Configurations

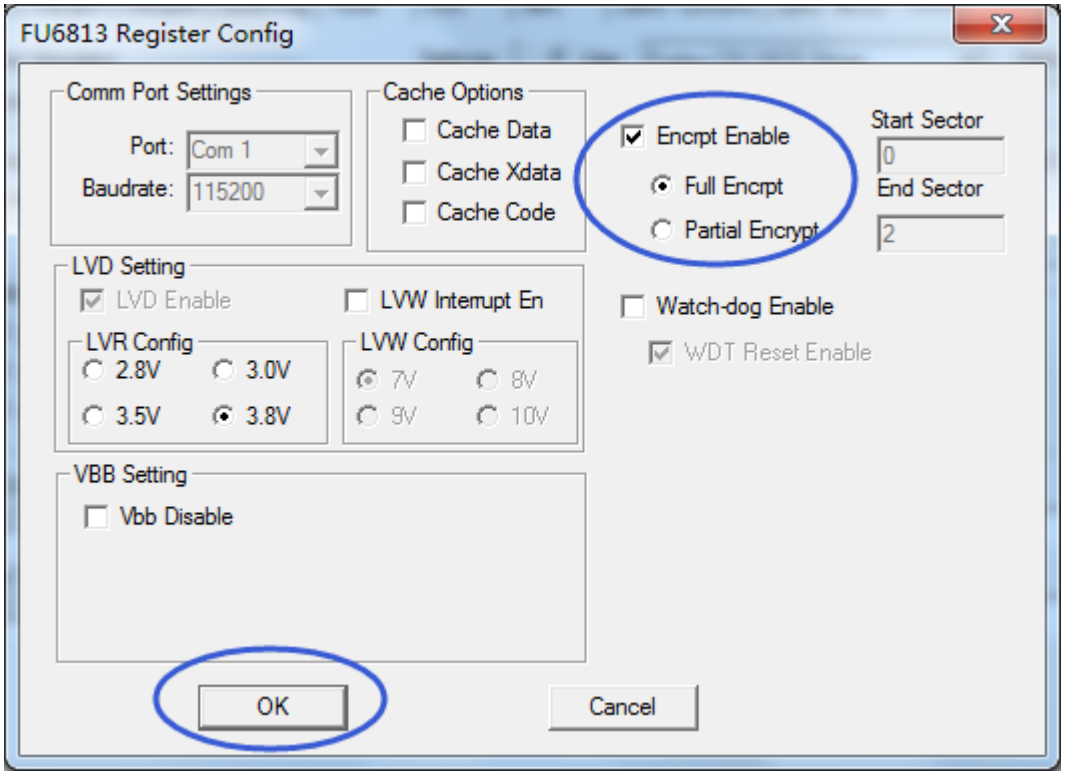


Figure 35-2 Full Code Protection Mode

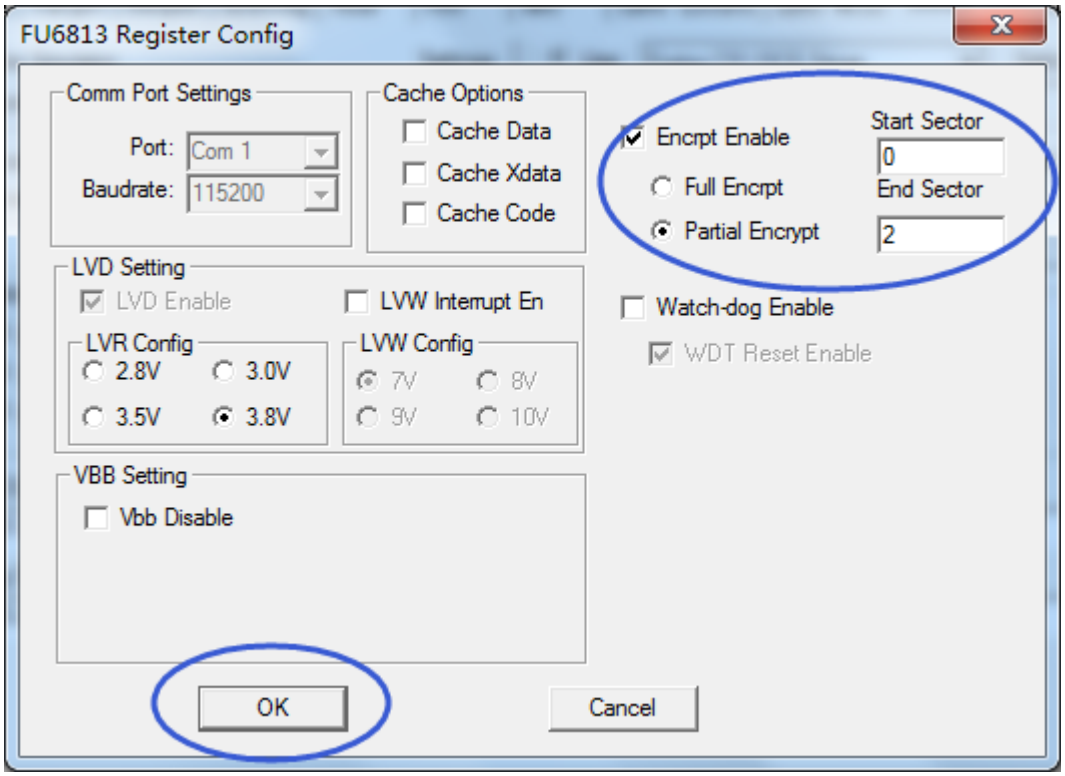


Figure 35-3 Partial Code Protection Mode

Operation steps are as follows:

1. Start 8051 IDE, enter Target Options and select Debug tab. As shown in Figure 35-1, click Settings to proceed with the setting;
2. Select the options as shown in Figure 35-2, and click OK. Then compile the project and download it. Get the BIN file and program it to Flash.

The chip support full code protection mode and partial code protection mode. After Full Encrypt is selected as shown in Figure 35-2, all codes in Flash are protected. After Partial Encrypt is selected as shown in Figure 35-3, the codes from sector 0 to END SECTOR are protected. The last sector is always protected in any case.

36 Revision History

Rev.	Description	Date	Prepared By
V1.0	First release. This document corresponds to internal document V0.42.	2020/05/09	Bruce Long
V1.1	This document corresponds to internal document V0.44. 1. Added descriptions on DAC0/1 operations; 2. Added descriptions on chip models in section 1.3.	2020/11/4	Bruce Long
V1.2	Added the chip model FU6813P. This document corresponds to internal document V0.48.	2021/01/14	Bruce Long
V1.3	Added the chip mode lFU6863Q and modified descriptions on VBB. This document corresponds to internal document V0.50.	2022/03/28	Bruce Long
V1.4	1. Corrected “Gate Driver” as “PWM” in Figure 1-1 Functional Block Diagram of FU6813L; 2. Updated chapter 2 Pin Definitions; 3. Updated chapter 3 Package Information; 4. Updated section 5.1 Absolute Maximum Ratings; 5. Updated Table 5-5 Global Electrical Characteristics of FU6813; 6. Updated Table 5-6 Global Electrical Characteristics of FU6863; 7. Updated Table 5-7 GPIO Electrical Characteristics; 8. Modified the test conditions “T_A=25℃” of Output Source Current and Output Sink Current in Table 5-8 PWM IO Electrical Characteristics as “P1_AN[HDIO] = 1” 9. Updated Table 5-9 6N Pre-driver IO Electrical Characteristics of FU6863; 10. Modified positions of the endnotes in Table 5-10 ADC Electrical Characteristics; 11. Updated Table 5-11 VREF and VHALF Electrical Characteristics; 12. Updated section 5.10 OSC Electrical Characteristics; 13. Updated section 5.11 Reset Electrical Characteristics; 14. Modified the test condition “VCC = 7V～30V” in Table 5-16 LDO Electrical Characteristics as “VCC = 7V～24V”; 15. Updated section 5.13 Package Thermal Resistance; 16. Modified the formula of baud rate “ fcpu_clk / (16 × (1+UT_BAUD[BAUD_SEL]) × (UT_BAUD + 1))” for 01 and 11 mode of the bit [7:6] in section 10.2.1 UT_CR(0x98) as “ fcpu_clk/(16/(1+ UT_BAUD[BAUD_SEL])) / (UT_BAUD+1)”; 17. Updated the default data format for PI_KP and PI_KI in section 13 PI; 18. Deleted the bit PI_CR[2] PIRANGE; 19. Updated descriptions in 14.2.6 FOC_TRGDLY(0x40A5); 20. Added more detailed descriptions in section 16.2.2.2 Forced Commutation at 60°. 21. Modified the name of the bit name [3] “T1RUIE” in section 16.3.6 TIM1_IER(0x406D) as “T1ROIE”; 22. Modified DRV_CMRH(0x405C) [7:0] in section 20.2.4 DRV_CMR(0x405C, 0x405D) as DRV_CMRH(0x405C) [15:8]; 23. Added descriptions on the bits [6:0] in section 31.2.3 CCFG1:CK RST CFG (0x401E); 24. Modified “CRC16-CCITT” in section 33 CRC as “CRC16-CCITT-FALSE”.	2023/11/23	Eric Deng
V1.5	1. Separated section 1.4 Functional Block Diagram by chip models; 2. Separated chapter 5 Electrical Characteristics by chip models; 3. Proofread the overall document and corrected grammar mistakes and wrong sentences; 4. Standardized document format.	2023/12/29	Eric Deng

Copyright Notice

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

Contained herein