Fortior Tech

FU6813/63
MCU Embedded and Configurable 3-Phase PMSM Motor Controller

Datasheet

Table of Contents

1 System Overview

1.1 Features

- Power Supply

FU6813L/P:

Single Supply HV Mode (VCC_MODE = 0): VCC = 5~24V

Dual Supply Mode (VCC_MODE = 1), VCC ≥ VDD5: VCC = 5~36V, VDD5 = 5V

Single Supply LV Mode (VCC_MODE = 1): VCC = VDD5 = 3~5.5V

FU6813N:

Single Supply HV Mode: VCC = 5~24V

Single Supply LV Mode: VCC = VDD5 = 3~5.5V

FU6863Q:

Mode 1: VCC_MODE = 0, VCC = 5~24V

Mode 2: VCC_MODE = 1, VCC = 5~24V, VDD5 = 3~5.5V

- Dual core: 8051 core and ME
- Instruction cycles: 1T or 2T
- 32K Byte Flash, with CRC verification. Support self-write and code protection
- 256 Byte IRAM, 1.5K Byte XRAM
- ME: integrated LPF, PI, BLDC module, FOC module
- 1T 16 ×16 multiplier, 16T 32/16 divider
- 4 priority level interrupt, 15 interrupt source
- GPIO number:

  FU6813L: 34

  FU6813P: 35

  FU6813N: 20

  FU6863Q: 32
- Timer

  2 capture timer

  1 QEP timer

  1 general timer

  1 RTC timer
- 1 SPI
- 1 I2C

- 2 UART
- 2 channel DMA. Support I2C / SPI / UART
- Analog peripherals:

  12 bit ADC. Conversion time is 1μs. Select voltage reference: Internal or External VREF

  FU6813L: 14 channel

  FU6813P: 14 channel

  FU6813N: 9 channel

  FU6863Q: 14 channel

  Configurable VREF: 3V, 4V, 4.5, VDD5Select VHALF (1/2 VREF) output

  4 operational amplifiers (2 for FU6813N)

  4 analog comparators

  DAC: 1 channel 9 bit, 1 channel 6 bit, 1 channel 8 bit
- Pre-driver mode:

  Gate Driver (For FU6813)

  6N Pre-driver (For FU6863)
- BLDC drive mode: automatic commutation, current limiting, HALL sensors, BEMF ZCP feedback
- FOC drive mode: single, dual or triple resistors current sampling (Only single resistor for FU6813N), over modulation
- PFC
- Clock source:

  System clock source: internal clock 24MHz±2%

  32.8KHz low speed clock

  32768 crystal oscillator
- Watch-dog
- Two wire FICE for on chip debugging

(1) FU6813L/P/N package is LQFP48, LQFP52 and QFN32.

(2) FU6863Q package is QFN56.

1.2 Applications

FU6813/63 can be applied for many domestic and industrial products using sensor/sensorless BLDC/PMSM, or induction motor. The typical applications include ceiling fan, stand fan, cooling fan, exhaust fan, electric tool, electric bicycle, vacuum, pump, compressor and drone.

1.3 Description

FU6813/63 series is a dual core chip integrated with 8051 processor core and motor drive engine (ME) specially designed for motor drive. The 8051 core is used for routine operation processing while ME is used for real-time motor control processing. Most of instruction cycle of the 8051 core is 1T, or 2T. IC is integrated high speed ADC, high speed amplifiers, comparator, Pre-driver, CRC, SPI, I2C, UART, timer, PWM, and HV LDO. These are using to SVPWM/SPWM、FOC control of BLDC/PMSM.

FU68x3 series is formed by 2 types of chip: FU6813 and FU6863. These 2 chips are designed for different applications. The detailed information can be found in Chapter-4. FU6813 uses Gate Driver output, FU6863 uses 6N Pre-driver output.

By integrating dual core with rich set of peripherals, FU6813/63 is the platform of choice for next-generation applications that require programmability, leading-edge signal processing and robust hardware support in one integrated package. These applications span a wide array of markets, from servo control, ultrahigh speed drive and automotive to domestic and industrial-based applications that require high speed processing.

1.4 Block Diagram

1.4.1 FU6813 Block Diagram

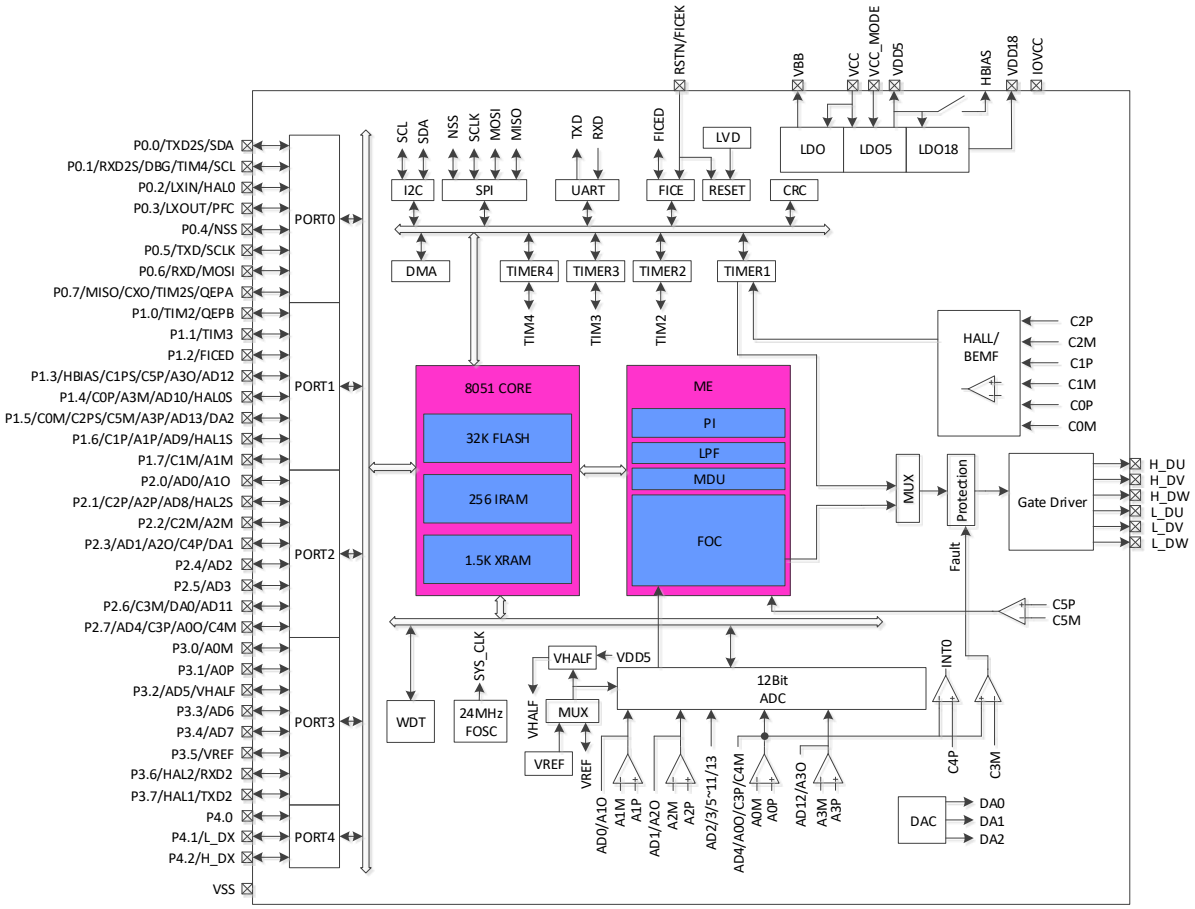


Figure 1-1 FU6813 Block Diagram

1.4.2 FU6863 Block Diagram

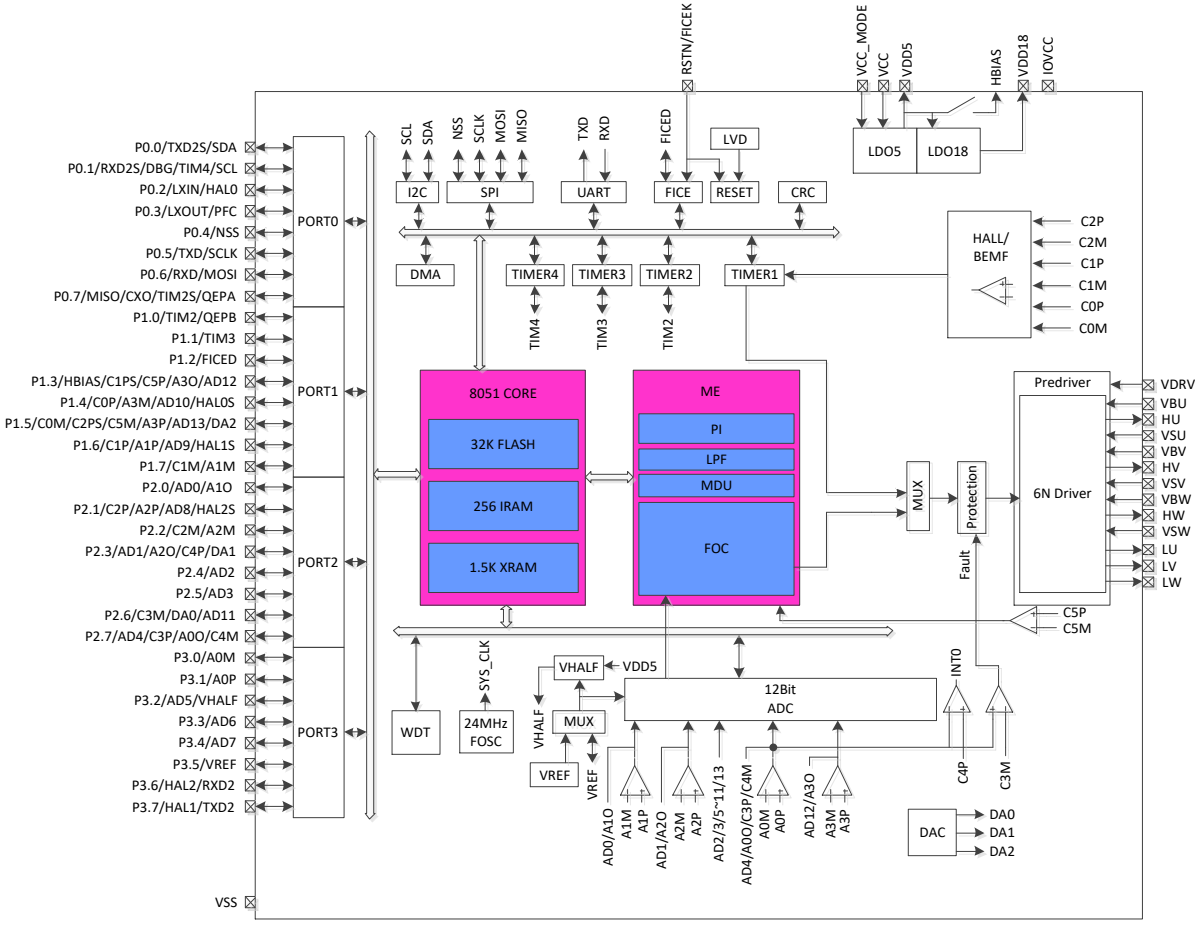


Figure 1-2 FU6863 Block Diagram

1.5 Memory Organization

The memory organization is similar to the standard of 8051. There are two separate memory spaces: program memory and data memory, accessed via different instruction types. The memory organization is shown in.

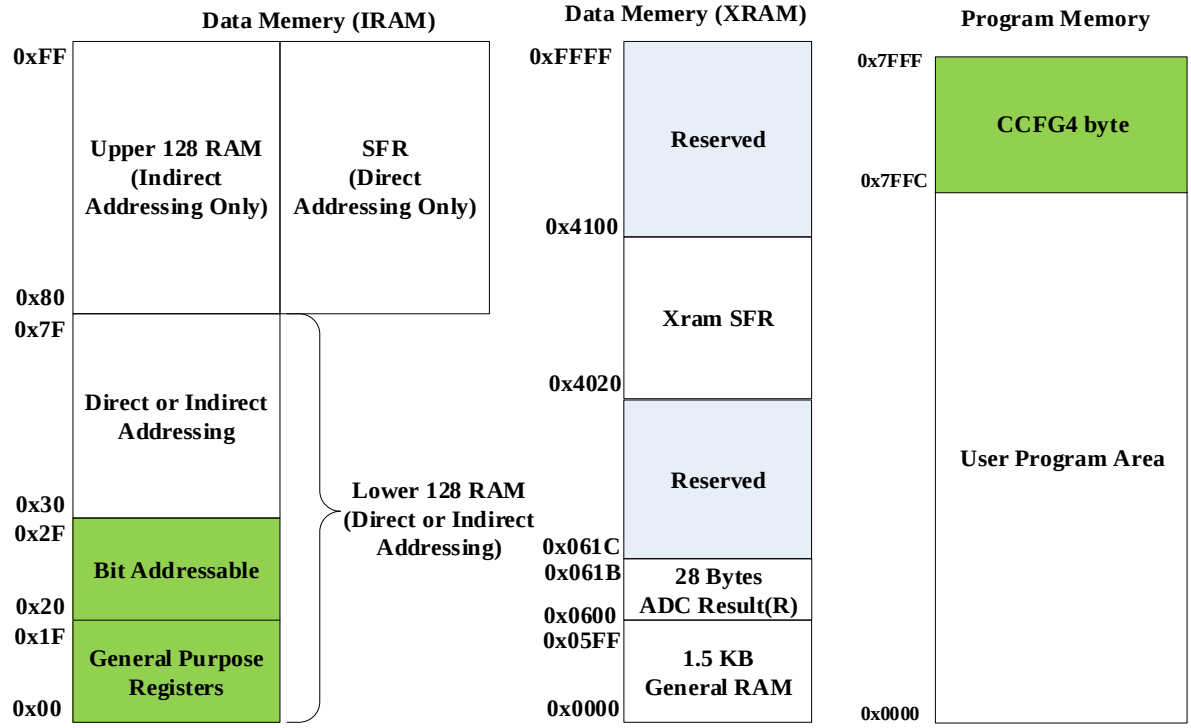


Figure 1-3 Memory Space Allocation

1.5.1 Program Memory

The FU68x3 series implements 32kB of the program memory space, re-programmable flash memory, organized in a contiguous block from addresses 0x0000 to 0x7FFF.

1.5.2 Data Memory

The FU6813/63 series consists of the following of RAM data memory: External RAM (XRAM), Internal RAM (IRAM) and Special Function Registers (SFR). The data memory map is shown in Figure 1-3.

The MOVX instruction in an 8051 device is typically used to access external data memory. On the FU68x3 devices, the MOVX instruction is normally used to read and write on-chip XRAM, but can be re-configured to write and erase on-chip flash memory space.

There are 256 bytes of IRAM mapped into the data memory space from 0x00 through 0xFF. The lower 128 bytes of data memory are used for general purpose registers and scratch pad memory. Either direct or

indirect addressing may be used to access the lower 128 bytes of data memory. Locations 0x00 through 0x1F are addressable as four banks of general purpose registers, each bank consisting of eight byte-wide registers. The next 16 bytes, locations 0x20 through 0x2F, may either be addressed as bytes or as 128 bit locations accessible with the direct addressing mode.

The upper 128 bytes of data memory are accessible only by indirect addressing. This region occupies the same address space as the SFR but is physically different from the SFR space. The addressing mode used by an instruction when accessing locations above 0x7F depends on if the upper 128 bytes of data memory space or the SFRs are being accessed. Instructions that use direct addressing will access the SFR space. Instructions using indirect addressing above 0x7F access the upper 128 bytes of data memory.

1.5.3 SFR

There are two parts of special function register: SFR and External SFR (XSFR). The XSFR locates in the XRAM address range. The way of accessing XSFR is same as XRAM.

Table 1-1 Address Map of Special Function Register (SFR)

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

Note 1: Bit is addressable for the SFR when the lower 4 bit is 0 or 8.

Note 2: Use a variable to read the value of register with double underscore.

1.5.4 XSFR

Table 1-2 Address Map of Extended Special Function Register (XSFR)

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_CR1	UDC_UKMINL/ PFC_KM	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
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	FOC__IDREFL	FOC__IQREFH	FOC__IQREFL	FOC_DQKPH	FOC_DQKPL	FOC_DQKIH	FOC_DQKIL
0x4088	FOC_EK3H/ TIM1__RARRH	FOC_EK3L/ TIM1__RARRL	FOC_EK4H/ TIM1__RCNTRH	FOC_EK4L/ TIM1__RCNTRL	FOC_EK1H	FOC_EK1L	FOC_EK2H	FOC_EK2L
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			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
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_BARRH	DMA1_BARRL	UT2_BAUDH	UT2_BAUDL	CAL_CR0	CAL_CR1		
0x4038	ADC_SCYC	ADC_CR	DMA0_CR0	DMA1_CR0	DMA0_LEN	DMA1_LEN	DMA0_BARRH	DMA1_BARRL
0x4030	SPI_CR0	SPI_CR1	SPI_CLK	SPI_DR		DAC_CR	ADC_MASKH	ADC_MASKL
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

2 Pin Configuration and Functions

2.1 FU6813 LQFP48 Pin Definition

Table 2-1 FU6813 LQFP48 Pin Definition

PAD Name	FU6813 LQFP48	IO Type	Description
P2.2/ C2M/ A2M	1	DB/ AI/ AI	1. GPIO P2.2 configurable as external interrupt-1 2. Negative input to comparator CMP2 3. Negative input to Op Amp A2
P2.3/ AD1/ A2O/ C4P/ DA1	2	DB/ AI/ AO/ AI/ DO	1. GPIO P2.3 configurable as external interrupt-1 2. ADC Channel-1 input 3. Output Op Amp A2 4. Positive input to comparator CMP4 5. Output DAC1, without buffer
P2.4/ AD2	3	DB/ AI	1. GPIO P2.4 configurable as external interrupt-1 2. ADC Channel-2 input, used for sampling of power supply voltage
P2.5/ AD3	4	DB/ AI	1. GPIO P2.5 configurable as external interrupt-1 2. ADC Channel-3 input
P2.6/ C3M/ DA0/ AD11	5	DB/ AI/ AO/ AI	1. GPIO P2.6 configurable as external interrupt-1 2. Negative input to comparator CMP3 3. Output DAC0, without buffer 4. ADC Channel-11 input
P2.7/ AD4/ C3P/ A0O/ C4M	6	DB/ AI/ AI/ AO/ AI	1. GPIO P2.7 configurable as external interrupt-1 2. ADC Channel-4 input, used for sampling of power supply current 3. Positive input to comparator CMP3 4. Output Op Amp A0 5. Negative input to comparator CMP4
P3.0/ A0M	7	DB/ AI	1. GPIO P3.0 2. Negative input to Op Amp A0
P3.1/ A0P	8	DB/ AI	1. GPIO P3.1 2. Positive input to Op Amp A0
P3.2/ AD5/ VHALF	9	DB/ AI/ AO	1. GPIO P3.2 2. ADC Channel-5 input 3. Voltage reference output. Configurable as 1/2 VREF. Shunt a 1μF capacitance to ground.
P3.3/ AD6	10	DB/ AI	1. GPIO P3.3 2. ADC Channel-6 input
P3.4/ AD7	11	DB/ AI	1. GPIO P3,4 2. ADC Channel-7 input

PAD Name	FU6813 LQFP48	IO Type	Description
P3.5/ VREF	12	DB/ AI	1. GPIO P3.5 2. Configurable as ADC external reference input or VREF voltage reference output. Shunt a 1~4.7μF capacitance to ground.
VSS	13	P	Digital ground
IOVCC	14	P	Power supply of IO, input range is 3~5.5V. Shunt a 1~10μF capacitance to ground. IOVCC ≤ VDD5, P3.7~P3.6, P0.x, P1.1~P1.0, P4.2~P4.1, H_DU, H_DV, H_DW, L_DU, L_DV and L_DW are powered by IOVCC. The remaining IOVCC are powered by VDD5.
P3.6/ HAL2/ RXD2	15	DB/ DI/ DI	1. GPIO P3.6 2. HALL2 logical input 3. UART2 RXD
P3.7/ HAL1/ TXD2	16	DB/ DI/ DO	1. GPIO P3.7 2. HALL1 logical input 3. UART2 TXD
P0.0/ TXD2S/ SDA	17	DO/ DB/ DO	1. GPIO P0.0 configurable as external interrupt-0 2. UART2 TXD when UART function transfer is enable 3. I2C SDA with optional 4.7K pull-up resistance
P0.1/ RXD2S/ DBG/ TIM4/ SCL	18	DB/ DI/ DO/ DB/ DB	1. GPIO P0.1 2. UART2 RXD when UART function transfer is enable 3. Debug port 4. Timer 4 capture input 5. I2C SCL with optional 4.7K pull-up resistance
P0.2/ LXIN/ HAL0	19	DB/ AI/ DI	1. GPIO P0.2 2. 32768Hz crystal-oscillator input 3. HALL0 logical input
P0.3/ LXOUT/ PFC	20	DB/ AO/ DO	1. GPIO P0.3 2. 32768Hz crystal-oscillator output 3. PFC output
P0.4/ NSS	21	DB/ DB	1. GPIO P0,4 2. SPI NSS
P0.5/ TXD/ SCLK	22	DB/ DO/ DB	1. GPIO P0.5 2. UART1 TXD when UART function transfer is disable 3. SPI clock
P0.6/ RXD/ MOSI	23	DB/ DI/ DB	1. GPIO P0.6 2. UART1 RXD when UART function transfer is disable 3. SPI MOSI

PAD Name	FU6813 LQFP48	IO Type	Description
P0.7/ MISO/ CXO/ TIM2S/ QEPA	24	DB/ DB/ DO/ DB/ DI	1. GPIO P0.7 2. SPIO MISO 3. Comparator output test pin 4. Timer 2 capture input when Timer 2 function transfer is enable/PWM output 5. QEP A input
P1.0/ TIM2/ QEPB	25	DB/ DB/ DI	1. GPIO P1.0 configurable as external interrupt-1 2. Timer 2 capture input when Timer 2 function transfer is disable/PWM output 3. QEP B input
P1.1/ TIM3	26	DB/ DB	1. GPIO P1.1 configurable as external interrupt-1 2. Timer 3 capture input
P4.1/ L_DX	27	DB/ DO	1. GPIO P4.1 2. Gate Driver phase X low side output
P4.2/ H_DX	28	DB/ DO	1. GPIO P4.2 2. Gate Driver phase X high side output
L_DU	29	DO	Gate Driver phase U low side output
L_DV	30	DO	Gate Driver phase V low side output
L_DW	31	DO	Gate Driver phase W low side output
H_DU	32	DO	Gate Driver phase U high side output
H_DV	33	DO	Gate Driver phase V high side output
H_DW	34	DO	Gate Driver phase W high side output
VCC	35	P	Power supply. Input voltage range select VCC_MODE. Shunt a 10μF capacitance to ground. 1. Single Supply HV Mode: VCC_MODE = 0, VCC input range is 5~24V, VDD5 is driven by internal LDO. 2. Single Supply LV Mode: VCC_MODE = 1 (i.e., connected with VDD5), VDD5 input range is 3~5.5V and short with VCC. 3. Dual Supply Mode: VCC_MODE = 1 (i.e., connected to VDD5), VCC input range is 5~36V, and VDD5 input is 5V.
VSS	36	P	Digital ground
VDD5	37	P	1. VDD supply input or internal LDO output select VCC_MODE. Shunt a 1~4.7μF capacitance to ground. 2. Please refer to VCC pin's description.
VCC_MODE	38	DI	Power supply mode selection. Please refer to VCC pin's description.

PAD Name	FU6813 LQFP48	IO Type	Description
RSTN/ FICEK	39	DI/ DI	1. External reset input with internal pull-up. 2. FICE clock
VDD18	40	P	1.8V LDO output. Shunt a 1~4.7μF capacitance to ground.
P1.2/ FICED	41	DB/ DB	1. GPIO P1.2 configurable as external interrupt-1 2. FICE data
P1.3/ HBIAS/ C1PS/ C5P/ A3O/ AD12	42	DB/ DO/ AI/ AI/ DO/ AI	1. GPIO P1.3 2. Gated VDD5 output 3. Positive input to comparator CMP1 when CMP1 function transfer is enable 4. Positive input to comparator CMP5 5. Output Op Amp A3 6. ADC Channel-12 input
P1.4/ C0P/ A3M/ AD10/ HAL0S	43	DB/ AI/ AI/ AI/ DI	1. GPIO P1.4 configurable as external interrupt-1 2. Positive input to comparator CMP0 3. Negative input to Op Amp A3 4. ADC Channel-10 input 5. HALL0 logical input when HALL0 function transfer is enable
P1.5/ C0M/ C2PS/ C5M/ A3P/ AD13/ DA2	44	DB/ AI/ AI/ AI/ AI/ AI/ AO	1. GPIO P1.5 configurable as external interrupt-1 2. Negative input to comparator CMP0 3. Positive input to comparator CMP2 when CMP2 function transfer is enable 4. Negative input to comparator CMP5 5. Positive input to Op Amp A3 6. ADC Channel-13 input 7. Output DAC2, without buffer
P1.6/ C1P/ A1P/ AD9/ HAL1S	45	DB/ AI/ AI/ AI/ DI	1. GPIO P1.6 configurable as external interrupt-1 2. Positive input to comparator CMP1 3. Positive input to Op Amp A1 4. ADC Channel-9 input 5. HALL1 logical input when HALL1 function transfer is enable
P1.7/ C1M/ A1M	46	DB/ AI/ AI	1. GPIO P1.7 configurable as external interrupt-1 2. Negative input to comparator CMP1 3. Negative input to Op Amp A1
P2.0/ AD0/ A1O	47	DB/ AI/ AO	1. GPIO P2.0 configurable as external interrupt-1 2. ADC Channel-0 input 3. Output Op Amp A1
P2.1/ C2P/ A2P/ AD8/ HAL2S	48	DB/ AI/ AI/ AI/ DI	1. GPIO P2.1 configurable as external interrupt-1 2. Positive input to comparator CMP2 3. Positive input to Op Amp A2 4. ADC Channel-8 input 5. HALL2 logical input when HALL2 function transfer is enable

Note：

IO Type Definition:

DI: Digital input,

DO: Digital output,

DB: Bidirectional input and output,

AI: Analog input,

AO: Analog output,

P: Power supply or ground pin

2.2 FU6813L LQFP48 Footprint

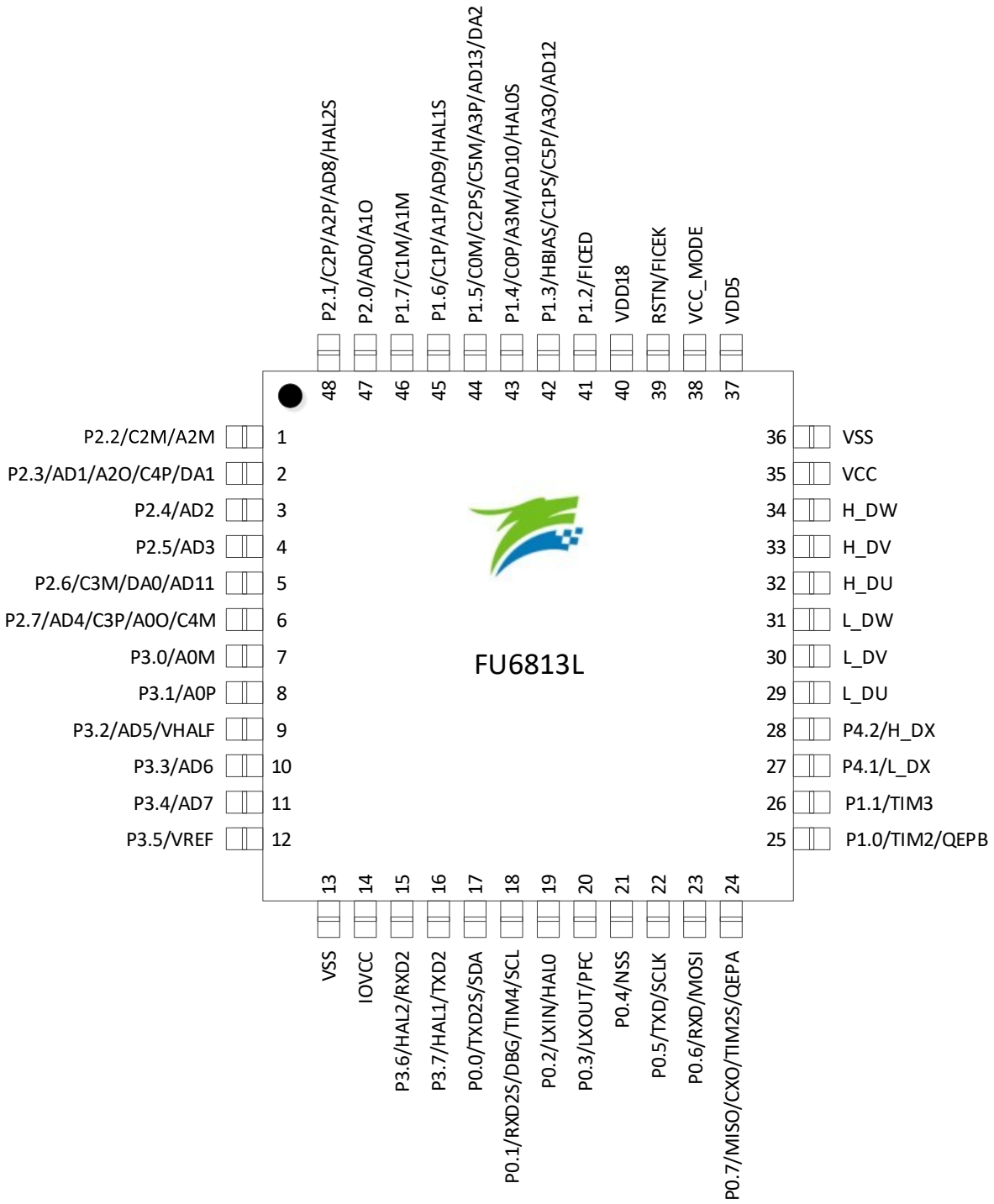


Figure 2-1 FU6813L /LQFP48 Footprint Diagram

2.3 FU6813 QFN32 Pin Definition

Table 2-2 FU6813 QFN32 Pin Definition

PAD Name	FU6813 QFN32	IO Type	Description
P2.1/ C2P/ A2P/ AD8/ HAL2S	1	DB/ AI/ AI/ AI/ DI	1. GPIO P2.1 configurable as external interrupt-1 2. Positive input to comparator CMP2 3. Positive input to Op Amp A2 4. ADC Channel-8 input 5. HALL2 logical input when HALL2 function transfer is enable
P2.4/ AD2	2	DB/ AI	1. GPIO P2.4 configurable as external interrupt-1 2. ADC Channel-2 input, used for sampling of power supply voltage
P2.6/ C3M/ DA0/ AD11	3	DB/ AI/ AO/ AI	1. GPIO P2.6 configurable as external interrupt-1 2. Negative input to comparator CMP3 3. Output DAC0, without buffer 4. ADC Channel-11 input
P2.7/ AD4/ C3P/ A0O/ C4M	4	DB/ AI/ AI/ AO/ AI	1. GPIO P2.7 configurable as external interrupt-1 2. ADC Channel-4 input, used for sampling of power supply current 3. Positive input to comparator CMP3 4. Output Op Amp A0 5. Negative input to comparator CMP4
P3.0/ A0M	5	DB/ AI	1. GPIO P3.0 2. Negative input to Op Amp A0
P3.1/ A0P	6	DB/ AI	1. GPIO P3.1 2. Positive input to Op Amp A0
P3.2/ AD5/ VHALF	7	DB/ AI/ AO	1. GPIO P3.2 2. ADC Channel-5 input 3. Voltage reference output. Configurable as 1/2 VREF. Shunt a 1μF capacitance to ground.
P3.5/ VREF	8	DB/ AI	1. GPIO P3.5 2. Configurable as ADC external reference input or VREF voltage reference output. Shunt a 1~4.7μF capacitance to ground.
IOVCC	9	P	Power supply of IO, input range is 3~5.5V. Shunt a 1~10μF capacitance to ground. IOVCC ≤ VDD5, P0.x, P1.1, H_DU, H_DV, H_DW, L_DU, L_DV and L_DW are powered by IOVCC. The remaining IOVCC are powered by VDD5.
P0.0/ TXD2S/ SDA	10	DO/ DB/ DO	1. GPIO P0.0 configurable as external interrupt-0 2. UART2 TXD when UART function transfer is enable 3. I2C SDA with optional 4.7K pull-up resistance

PAD Name	FU6813 QFN32	IO Type	Description
P0.1/ RXD2S/ DBG/ TIM4/ SCL	11	DB/ DI/ DO/ DB/ DB	1. GPIO P0.1 2. UART2 RXD when UART function transfer is enable 3. Debug port 4. Timer 4 capture input 5. I2C SCL with optional 4.7K pull-up resistance
P0.4/ NSS	12	DB/ DB	1. GPIO P0,4 2. SPI NSS
P0.5/ TXD/ SCLK	13	DB/ DO/ DB	1. GPIO P0.5 2. UART1 TXD when UART function transfer is disable 3. SPI clock
P0.6/ RXD/ MOSI	14	DB/ DI/ DB	1. GPIO P0.6 2. UART1 RXD when UART function transfer is disable 3. SPI MOSI
P0.7/ MISO/ CXO/ TIM2S/ QEPA	15	DB/ DB/ DO/ DB/ DI	1. GPIO P0.7 2. SPIO MISO 3. Comparator output test pin 4. Timer 2 capture input when Timer 2 function transfer is enable/PWM output 5. QEP A input
P1.1/ TIM3	16	DB/ DB	1. GPIO P1.1 configurable as external interrupt-1 2. Timer 3 capture input
L_DU	17	DO	Gate Driver phase U low side output
L_DV	18	DO	Gate Driver phase V low side output
L_DW	19	DO	Gate Driver phase W low side output
H_DU	20	DO	Gate Driver phase U high side output
H_DV	21	DO	Gate Driver phase V high side output
H_DW	22	DO	Gate Driver phase W high side output
VCC	23	P	Power supply. Input voltage range select VCC_MODE. Shunt a 10μF, or more, capacitance to ground. 1. Single Supply HV Mode: VCC_MODE = 0, VCC input range is 5~24V, VDD5 is driven by internal LDO. 2. Single Supply LV Mode: VCC_MODE = 1 (i.e., connected with VDD5), VDD5 input range is 3~5.5V and short with VCC.
VSS	24	P	Digital ground
VDD5	25	P	1. VDD supply input or internal LDO output select VCC_MODE. Shunt a 1~4.7μF capacitance to ground. 2. Please refer to VCC pin's description.
RSTN/ FICEK	26	DI/ DI	1. External reset input with internal pull-up. 2. FICE clock

PAD Name	FU6813 QFN32	IO Type	Description
VDD18	27	P	1.8V LDO output. Shunt a 1~4.7μF capacitance to ground.
P1.2/ FICED	28	DB/ DB	1. GPIO P1.2 configurable as external interrupt-1 2. FICE data
P1.3/ HBIAS/ C1PS/ C5P/ A3O/ AD12	29	DB/ DO/ AI/ AI/ DO/ AI	1. GPIO P1.3 2. Gated VDD5 output 3. Positive input to comparator CMP1 when CMP1 function transfer is enable 4. Positive input to comparator CMP5 5. Output Op Amp A3 6. ADC Channel-12 input
P1.4/ C0P/ A3M/ AD10/ HAL0S	30	DB/ AI/ AI/ AI/ DI	1. GPIO P1.4 configurable as external interrupt-1 2. Positive input to comparator CMP0 3. Negative input to Op Amp A3 4. ADC Channel-10 input 5. HALL0 logical input when HALL0 function transfer is enable
P1.5/ C0M/ C2PS/ C5M/ A3P/ AD13/ DA2	31	DB/ AI/ AI/ AI/ AI/ AI/ AO	1. GPIO P1.5 configurable as external interrupt-1 2. Negative input to comparator CMP0 3. Positive input to comparator CMP2 when CMP2 function transfer is enable 4. Negative input to comparator CMP5 5. Positive input to Op Amp A3 6. ADC Channel-13 input 7. Output DAC2, without buffer
P1.6/ C1P/ A1P/ AD9/ HAL1S	32	DB/ AI/ AI/ AI/ DI	1. GPIO P1.6 configurable as external interrupt-1 2. Positive input to comparator CMP1 3. Positive input to Op Amp A1 4. ADC Channel-9 input 5. HALL1 logical input when HALL1 function transfer is enable

Note:

IO Type Definition:

DI: Digital input,

DO: Digital output,

DB: Bidirectional input and output,

AI: Analog input,

AO: Analog output,

P: Power supply or ground pin

2.4 FU6813N QFN32 Footprint

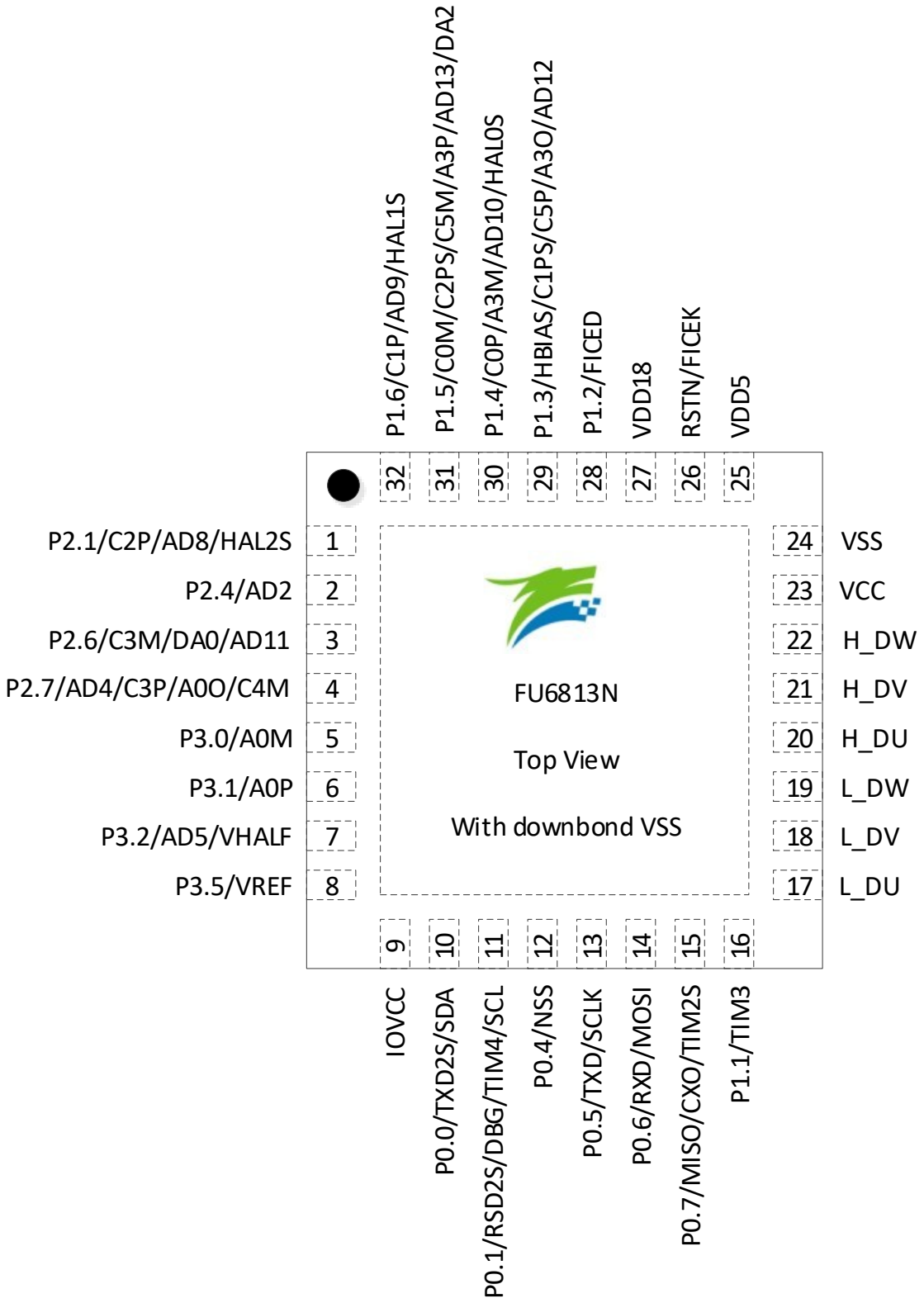


Figure 2-2 FU6813N/QFN32 Footprint Diagram

2.5 FU6863 QFN56 Pin Definition

Table 2-3 FU6863 QFN56 Pin Definition

PAD Name	FU6863 QFN56	IO Type	Description
VSU	1	P	Phase U input for 6N Pre-driver, and for the grounding reference of the high side of phase U self-boost.
HU	2	DO	High side Output phase U Pre-driver PWM, 6N control mode, drive NMOS gate.
VBU	3	P	Self-boost power supply for the high side of phase U 6N Pre-driver
VSV	4	P	Phase V input for 6N Pre-driver, and for the grounding reference of the high side of phase V self-boost
HV	5	DO	High side Output phase V Pre-driver PWM, 6N control mode, drive NMOS gate.
VBV	6	P	Self-boost power supply for the high side of phase V 6N Pre-driver
VSW	7	P	Phase W input for 6N Pre-driver, and for the grounding reference of the high side of phase W self-boost
HW	8	DO	High side Output phase W Pre-driver PWM, 6N control mode, drive NMOS gate.
VBW	9	P	Self-boost power supply for the high side of phase W 6N Pre-driver
VCC	10	P	Power supply. Input voltage range select VCC_MODE. Shunt a 10μF, or more, capacitance to ground. 1. Single Supply HV Mode: VCC_MODE = 0, VCC input range is 5~24V, VDD5 is driven by internal LDO. 2. Dual Supply Mode: VCC_MODE = 1 (i.e., connected to VDD5), VCC input range is 5~36V, and VDD5 input is 5V.
VSS	11	P	Digital ground
VDD5	12	P	1. VDD supply input or internal LDO output select VCC_MODE. Shunt a 1~4.7μF capacitance to ground. 2. Please refer to VCC pin's description.
VCC_MODE	13	DI	Power supply mode selection. Please refer to VCC pin's description.
RSTN/ FICEK	14	DI/ DI	1. External reset input with internal pull-up. 2. FICE clock
VDD18	15	P	1.8V LDO output. Shunt a 1~4.7μF capacitance to ground.
VSS	16	P	Digital ground

PAD Name	FU6863 QFN56	IO Type	Description
P1.2/ FICED	17	DB/ DB	1. GPIO P1.2 configurable as external interrupt-1 2. FICE data
P1.3/ HBIAS/ C1PS/ C5P/ A3O/ AD12	18	DB/ DO/ AI/ AI/ DO/ AI	1. GPIO P1.3 2. Gated VDD5 output 3. Positive input to comparator CMP1 when CMP1 function transfer is enable 4. Positive input to comparator CMP5 5. Output Op Amp A3 6. ADC Channel-12 input
P1.4/ C0P/ A3M/ AD10/ HAL0S	19	DB/ AI/ AI/ AI/ DI	1. GPIO P1.4 configurable as external interrupt-1 2. Positive input to comparator CMP0 3. Negative input to Op Amp A3 4. ADC Channel-10 input 5. HALL0 logical input when HALL0 function transfer is enable
P1.5/ C0M/ C2PS/ C5M/ A3P/ AD13/ DA2	20	DB/ AI/ AI/ AI/ AI/ AI/ AO	1. GPIO P1.5 configurable as external interrupt-1 2. Negative input to comparator CMP0 3. Positive input to comparator CMP2 when CMP2 function transfer is enable 4. Negative input to comparator CMP5 5. Positive input to Op Amp A3 6. ADC Channel-13 input 7. Output DAC2, without buffer
P1.6/ C1P/ A1P/ AD9/ HAL1S	21	DB/ AI/ AI/ AI/ DI	1. GPIO P1.6 configurable as external interrupt-1 2. Positive input to comparator CMP1 3. Positive input to Op Amp A1 4. ADC Channel-9 input 5. HALL1 logical input when HALL1 function transfer is enable
P1.7/ C1M/ A1M	22	DB/ AI/ AI	1. GPIO P1.7 configurable as external interrupt-1 2. Negative input to comparator CMP1 3. Negative input to Op Amp A1
P2.0/ AD0/ A1O	23	DB/ AI/ AO	1. GPIO P2.0 configurable as external interrupt-1 2. ADC Channel-0 input 3. Output Op Amp A1
P2.1/ C2P/ A2P/ AD8/ HAL2S	24	DB/ AI/ AI/ AI/ DI	1. GPIO P2.1 configurable as external interrupt-1 2. Positive input to comparator CMP2 3. Positive input to Op Amp A2 4. ADC Channel-8 input 5. HALL2 logical input when HALL2 function transfer is enable
P2.2/ C2M/ A2M	25	DB/ AI/ AI	1. GPIO P2.2 configurable as external interrupt-1 2. Negative input to comparator CMP2 3. Negative input to Op Amp A2

PAD Name	FU6863 QFN56	IO Type	Description
P2.3/ AD1/ A2O/ C4P/ DA1	26	DB/ AI/ AO/ AI/ DO	1. GPIO P2.3 configurable as external interrupt-1 2. ADC Channel-1 input 3. Output Op Amp A2 4. Positive input to comparator CMP4 5. Output DAC1, without buffer
P2.4/ AD2	27	DB/ AI	1. GPIO P2.4 configurable as external interrupt-1 2. ADC Channel-2 input, used for sampling of power supply voltage
P2.5/ AD3	28	DB/ AI	1. GPIO P2.5 configurable as external interrupt-1 2. ADC Channel-3 input
P2.6/ C3M/ DA0/ AD11	29	DB/ AI/ AO/ AI	1. GPIO P2.6 configurable as external interrupt-1 2. Negative input to comparator CMP3 3. Output DAC0, without buffer 4. ADC Channel-11 input
P2.7/ AD4/ C3P/ A0O/ C4M	30	DB/ AI/ AI/ AO/ AI	1. GPIO P2.7 configurable as external interrupt-1 2. ADC Channel-4 input, used for sampling of power supply current 3. Positive input to comparator CMP3 4. Output Op Amp A0 5. Negative input to comparator CMP4
P3.0/ A0M	31	DB/ AI	1. GPIO P3.0 2. Negative input to Op Amp A0
P3.1/ A0P	32	DB/ AI	1. GPIO P3.1 2. Positive input to Op Amp A0
P3.2/ AD5/ VHALF	33	DB/ AI/ AO	1. GPIO P3.2 2. ADC Channel-5 input 3. Voltage reference output. Configurable as 1/2 VREF. Shunt a 1μF capacitance to ground.
P3.3/ AD6	34	DB/ AI	1. GPIO P3.3 2. ADC Channel-6 input
P3.4/ AD7	35	DB AI	1. GPIO P3,4 2. ADC Channel-7 input
P3.5/ VREF	36	DB/ AI	1. GPIO P3.5 2. Configurable as ADC external reference input or VREF voltage reference output. Shunt a 1~4.7μF capacitance to ground.
VSS	37	P	Digital ground
IOVCC	38	P	Power supply of IO, input range is 3~5.5V. Shunt a 1~10μF capacitance to ground. IOVCC ≤ VDD5, P3.7~P3.6, P0.x, P1.1~P1.0, H_DU, H_DV, H_DW, L_DU, L_DV and L_DW are powered by IOVCC. The remaining IOVCC are powered by VDD5.

PAD Name	FU6863 QFN56	IO Type	Description
P3.6/ HAL2/ RXD2	39	DB/ DI/ DI	1. GPIO P3.6 2. HALL2 logical input 3. UART2 RXD
P3.7/ HAL1/ TXD2	40	DB/ DI/ DO	1. GPIO P3.7 2. HALL1 logical input 3. UART2 TXD
P0.0/ TXD2S/ SDA	41	DO/ DB/ DO	1. GPIO P0.0 configurable as external interrupt-0 2. UART2 TXD when UART function transfer is enable 3. I2C SDA with optional 4.7K pull-up resistance
P0.1/ RXD2S/ DBG/ TIM4/ SCL	42	DB/ DI/ DO/ DB/ DB	1. GPIO P0.1 2. UART2 RXD when UART function transfer is enable 3. Debug port 4. Timer 4 capture input 5. I2C SCL with optional 4.7K pull-up resistance
P0.2/ LXIN/ HAL0	43	DB/ AI/ DI	1. GPIO P0.2 2. 32768Hz crystal-oscillator input 3. HALL0 logical input
P0.3/ LXOUT/ PFC	44	DB/ AO/ DO	1. GPIO P0.3 2. 32768Hz crystal-oscillator output 3. PFC output
P0.4/ NSS	45	DB/ DB	1. GPIO P0,4 2. SPI NSS
P0.5/ TXD/ SCLK	46	DB/ DO/ DB	1. GPIO P0.5 2. UART1 TXD when UART function transfer is disable 3. SPI clock
P0.6/ RXD/ MOSI	47	DB/ DI/ DB	1. GPIO P0.6 2. UART1 RXD when UART function transfer is disable 3. SPI MOSI
P0.7/ MISO/ CXO/ TIM2S/ QEPA	48	DB/ DB/ DO/ DB/ DI	1. GPIO P0.7 2. SPIO MISO 3. Comparator output test pin 4. Timer 2 capture input when Timer 2 function transfer is enable/PWM output 5. QEP A input
P1.0/ TIM2/ QEPB	49	DB/ DB/ DI	1. GPIO P1.0 configurable as external interrupt-1 2. Timer 2 capture input when Timer 2 function transfer is disable/PWM output 3. QEP B input
P1.1/ TIM3	50	DB/ DB	1. GPIO P1.1 configurable as external interrupt-1 2. Timer 3 capture input

PAD Name	FU6863 QFN56	IO Type	Description
VDRV	51	P	Internal LDO, output 15V. When VCC ≤ 15V, VBB = VCC, When VCC > 15V, VBB = 15V. Shunt 0.1μF and 10μF capacitor to ground, available for Pre-driver.
VSS	52	P	Digital ground
NC	53		NC Pin. Keep floating.
LU	54	DO	Low side Output phase U Pre-driver PWM, 6N control mode, drive NMOS gate.
LV	55	DO	Low side Output phase V Pre-driver PWM, 6N control mode, drive NMOS gate.
LW	56	DO	Low side Output phase W Pre-driver PWM, 6N control mode, drive NMOS gate.

Note:

IO Type Definition:

DI: Digital input,

DO: Digital output,

DB: Bidirectional input and output,

AI: Analog input,

AO: Analog output,

P: Power supply or ground pin

2.6 FU6863Q QFN56 Footprint

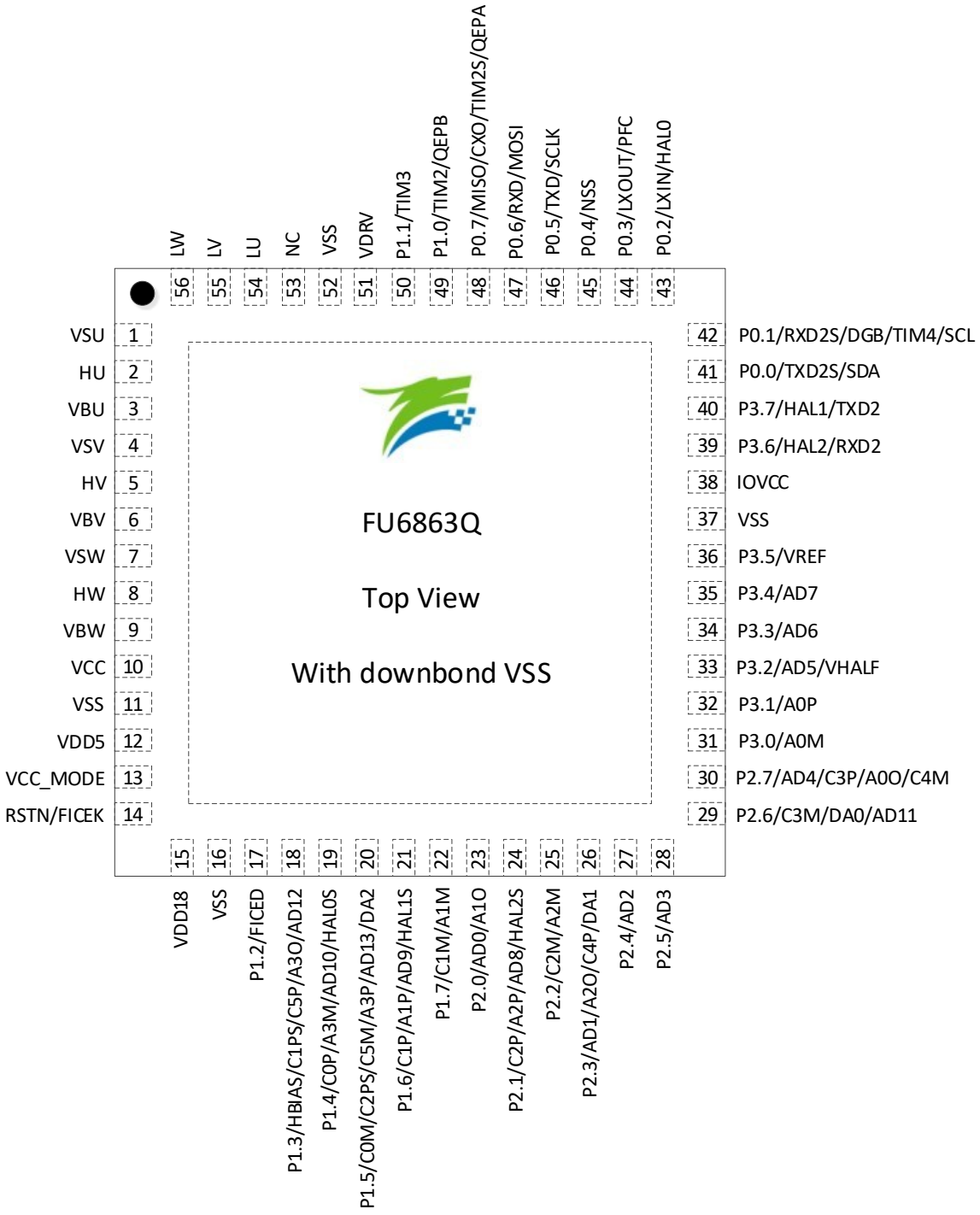


Figure 2-3 FU6863Q/QFN56 Footprint Diagram

2.7 FU6813 LQFP52 Pin Definition

Table 2-4 FU6813 LQFP52 Pin Definition

PAD Name	FU6813 LQFP52	IO Type	Description
P2.3/ AD1/ A2O/ C4P/ DA1	1	DB/ AI/ AO/ AI/ DO	1. GPIO P2.3 configurable as external interrupt-1 2. ADC Channel-1 input 3. Output Op Amp A2 4. Positive input to comparator CMP4 5. Output DAC1, without buffer
P2.4/ AD2	2	DB/ AI	1. GPIO P2.4 configurable as external interrupt-1 2. ADC Channel-2 input, used for sampling of power supply voltage
P2.5/ AD3	3	DB/ AI	1. GPIO P2.5 configurable as external interrupt-1 2. ADC Channel-3 input
P2.6/ C3M/ DA0/ AD11	4	DB/ AI/ AO/ AI	1. GPIO P2.6 configurable as external interrupt-1 2. Negative input to comparator CMP3 3. Output DAC0, without buffer 4. ADC Channel-11 input
P2.7/ AD4/ C3P/ A0O/ C4M	5	DB/ AI/ AI/ AO/ AI	1. GPIO P2.7 configurable as external interrupt-1 2. ADC Channel-4 input, used for sampling of power supply current 3. Positive input to comparator CMP3 4. Output Op Amp A0 5. Negative input to comparator CMP4
P3.0/ A0M	6	DB/ AI	1. GPIO P3.0 2. Negative input to Op Amp A0
P3.1/ A0P	7	DB/ AI	1. GPIO P3.1 2. Positive input to Op Amp A0
P3.2/ AD5/ VHALF	8	DB/ AI/ AO	1. GPIO P3.2 2. ADC Channel-5 input 3. Voltage reference output. Configurable as 1/2 VREF. Shunt a 1μF capacitance to ground.
P3.3/ AD6	9	DB/ AI	1. GPIO P3.3 2. ADC Channel-6 input
P3.4/ AD7	10	DB/ AI	1. GPIO P3,4 2. ADC Channel-7 input
P3.5/ VREF	11	DB/ AI	1. GPIO P3.5 2. Configurable as ADC external reference input or VREF voltage reference output. Shunt a 1~4.7μF capacitance to ground.
NC	12		NC
VSS	13	P	Digital ground

PAD Name	FU6813	IO	Description
IOVCC	14	P	Power supply of IO, input range is 3~5.5V. External shunt of 1~10μF. IOVCC ≤ VDD5, P3.7～P3.6, P0.x, P1.1～P1.0, P4.2～P4.1 , H_DU, H_DV, H_DW, L_DU, L_DV and L_DW are powered by IOVCC. The remaining IOVCC are powered by VDD5.
P3.6/ HAL2/ RXD2	15	DB/ DI/ DI	1. GPIO P3.6 2. HALL2 logical input 3. UART2 RXD
P3.7/ HAL1/ TXD2	16	DB/ DI/ DO	1. GPIO P3.7 2. HALL1 logical input 3. UART2 TXD
P0.0/ TXD2S/ SDA	17	DO/ DB/ DO	1. GPIO P0.0 configurable as external interrupt-0 2. UART2 TXD when UART function transfer is enable 3. I2C SDA with optional 4.7K pull-up resistance
P0.1/ RXD2S/ DBG/ TIM4/ SCL	18	DB/ DI/ DO/ DB/ DB	1. GPIO P0.1 2. UART2 RXD when UART function transfer is enable 3. Debug port 4. Timer 4 capture input 5. I2C SCL with optional 4.7K pull-up resistance
P0.2/ LXIN/ HAL0	19	DB/ AI/ DI	1. GPIO P0.2 2. 32768Hz crystal-oscillator input 3. HALL0 logical input
P0.3/ LXOUT/ PFC	20	DB/ AO/ DO	1. GPIO P0.3 2. 32768Hz crystal-oscillator output 3. PFC output
P0.4/ NSS	21	DB/ DB	1. GPIO P0,4 2. SPI NSS
P0.5/ TXD/ SCLK	22	DB/ DO/ DB	1. GPIO P0.5 2. UART1 TXD when UART function transfer is disable 3. SPI clock
P0.6/ RXD/ MOSI	23	DB/ DI/ DB	1. GPIO P0.6 2. UART1 RXD when UART function transfer is disable 3. SPI MOSI
P0.7/ MISO/ CXO/ TIM2S/ QEPA	24	DB/ DB/ DO/ DB/ DI	1. GPIO P0.7 2. SPIO MISO 3. Comparator output test pin 4. Timer 2 capture input when Timer 2 function transfer is enable/PWM output 5. QEP A input
P1.0/ TIM2/ QEPB	25	DB/ DB/ DI	1. GPIO P1.0 configurable as external interrupt-1 2. Timer 2 capture input when Timer 2 function transfer is disable/PWM output 3. QEP B input

PAD Name	FU6813	IO	Description
P1.1/ TIM3	26	DB/ DB	1. GPIO P1.1 configurable as external interrupt-1 2. Timer 3 capture input
P4.0	27	DB	GPIO P4.0
P4.1/ L_DX	28	DB/ DO	1. GPIO P4.1 2. Gate Driver phase X low side output
P4.2/ H_DX	29	DB/ DO	1. GPIO P4.2 2. Gate Driver phase X high side output
VBB	30	P	Internal LDO, output 15V. When VCC ≤ 15V, VBB = VCC, When VCC > 15V, VBB = 15V. Shunt 0.1μF and 10μF capacitor to ground, available for Pre-driver.
L_DU	31	DO	Gate Driver phase U low side output
L_DV	32	DO	Gate Driver phase V low side output
L_DW	33	DO	Gate Driver phase W low side output
H_DU	34	DO	Gate Driver phase U high side output
H_DV	35	DO	Gate Driver phase V high side output
H_DW	36	DO	Gate Driver phase W high side output
NC	37		NC Pin. Keep floating
VCC	38	P	Power supply. Input voltage range select VCC_MODE. Shunt a 10μF, or more, capacitor to ground. 1. Single Supply HV Mode: VCC_MODE = 0, VCC input range is 5~24V, VDD5 is driven by internal LDO. 2. Single Supply LV Mode: VCC_MODE = 1 (i.e., connected with VDD5), VDD5 input range is 3~5.5V and short with VCC. 3. Dual Supply Mode: VCC_MODE = 1 (i.e., connected to VDD5), VCC input range is 5~36V, and VDD5 input is 5V.
VSS	39	P	Digital ground
VDD5	40	P	1. VDD supply input or internal LDO output select VCC_MODE. Shunt a 1~4.7μF capacitor to ground. 2. Please refer to VCC pin's description.
VCC_MODE	41	DI	Power supply mode selection. Please refer to VCC pin's description.
RSTN/ FICEK	42	DI/ DI	1. External reset input with internal pull-up. 2. FICE clock
VDD18	43	P	1.8V LDO output. Shunt a 1~4.7μF capacitor to ground.
P1.2/ FICED	44	DB/ DB	1. GPIO P1.2 configurable as external interrupt-1 2. FICE data

PAD Name	FU6813	IO	Description
P1.3/ HBIAS/ C1PS/ C5P/ A3O/ AD12	45	DB/ DO/ AI/ AI/ DO/ AI	1. GPIO P1.3 2. Gated VDD5 output 3. Positive input to comparator CMP1 when CMP1 function transfer is enable 4. Positive input to comparator CMP5 5. Output Op Amp A3 6. ADC Channel-12 input
P1.4/ C0P/ A3M/ AD10/ HAL0S	46	DB/ AI/ AI/ AI/ DI	1. GPIO P1.4 configurable as external interrupt-1 2. Positive input to comparator CMP0 3. Negative input to Op Amp A3 4. ADC Channel-10 input 5. HALL0 logical input when HALL0 function transfer is enable
P1.5/ C0M/ C2PS/ C5M/ A3P/ AD13/ DA2	47	DB/ AI/ AI/ AI/ AI/ AI/ AO	1. GPIO P1.5 configurable as external interrupt-1 2. Negative input to comparator CMP0 3. Positive input to comparator CMP2 when CMP2 function transfer is enable 4. Negative input to comparator CMP5 5. Positive input to Op Amp A3 6. ADC Channel-13 input 7. Output DAC2, without buffer
P1.6/ C1P/ A1P/ AD9/ HAL1S	48	DB/ AI/ AI/ AI/ DI	1. GPIO P1.6 configurable as external interrupt-1 2. Positive input to comparator CMP1 3. Positive input to Op Amp A1 4. ADC Channel-9 input 5. HALL1 logical input when HALL1 function transfer is enable
P1.7/ C1M/ A1M	49	DB/ AI/ AI	1. GPIO P1.7 configurable as external interrupt-1 2. Negative input to comparator CMP1 3. Negative input to Op Amp A1
P2.0/ AD0/ A1O	50	DB/ AI/ AO	1. GPIO P2.0 configurable as external interrupt-1 2. ADC Channel-0 input 3. Output Op Amp A1
P2.1/ C2P/ A2P/ AD8/ HAL2S	51	DB/ AI/ AI/ AI/ DI	1. GPIO P2.1 configurable as external interrupt-1 2. Positive input to comparator CMP2 3. Positive input to Op Amp A2 4. ADC Channel-8 input 5. HALL2 logical input when HALL2 function transfer is enable
P2.2/ C2M/ A2M	52	DB/ AI/ AI	1. GPIO P2.2 configurable as external interrupt-1 2. Negative input to comparator CMP2 3. Negative input to Op Amp A2

Note:

IO Type Definition:

DI: Digital input,

DO: Digital output,

DB: Bidirectional input and output,

AI: Analog input,

AO: Analog output,

P: Power supply or ground pin

2.8 FU6813P LQFP52 Footprint

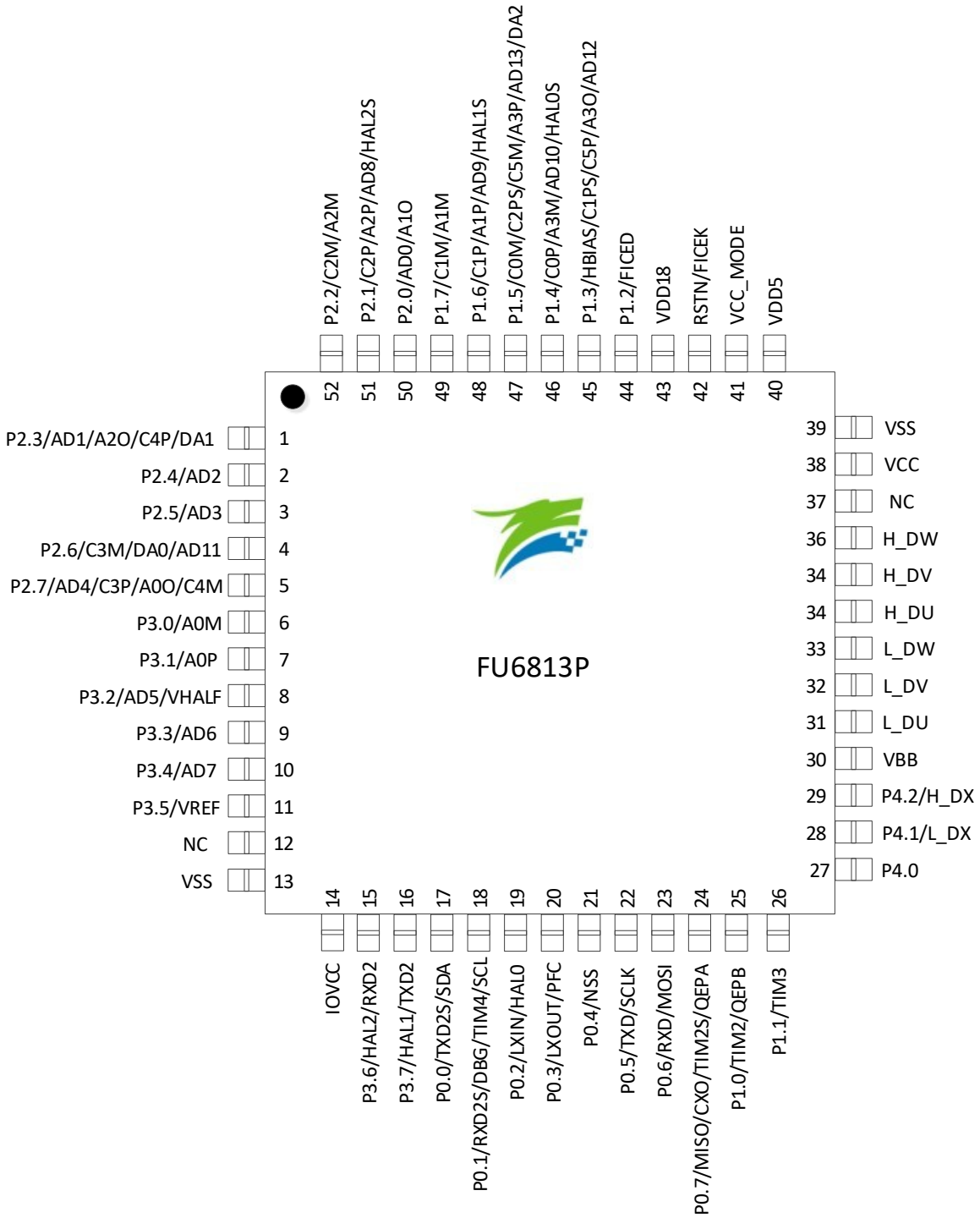
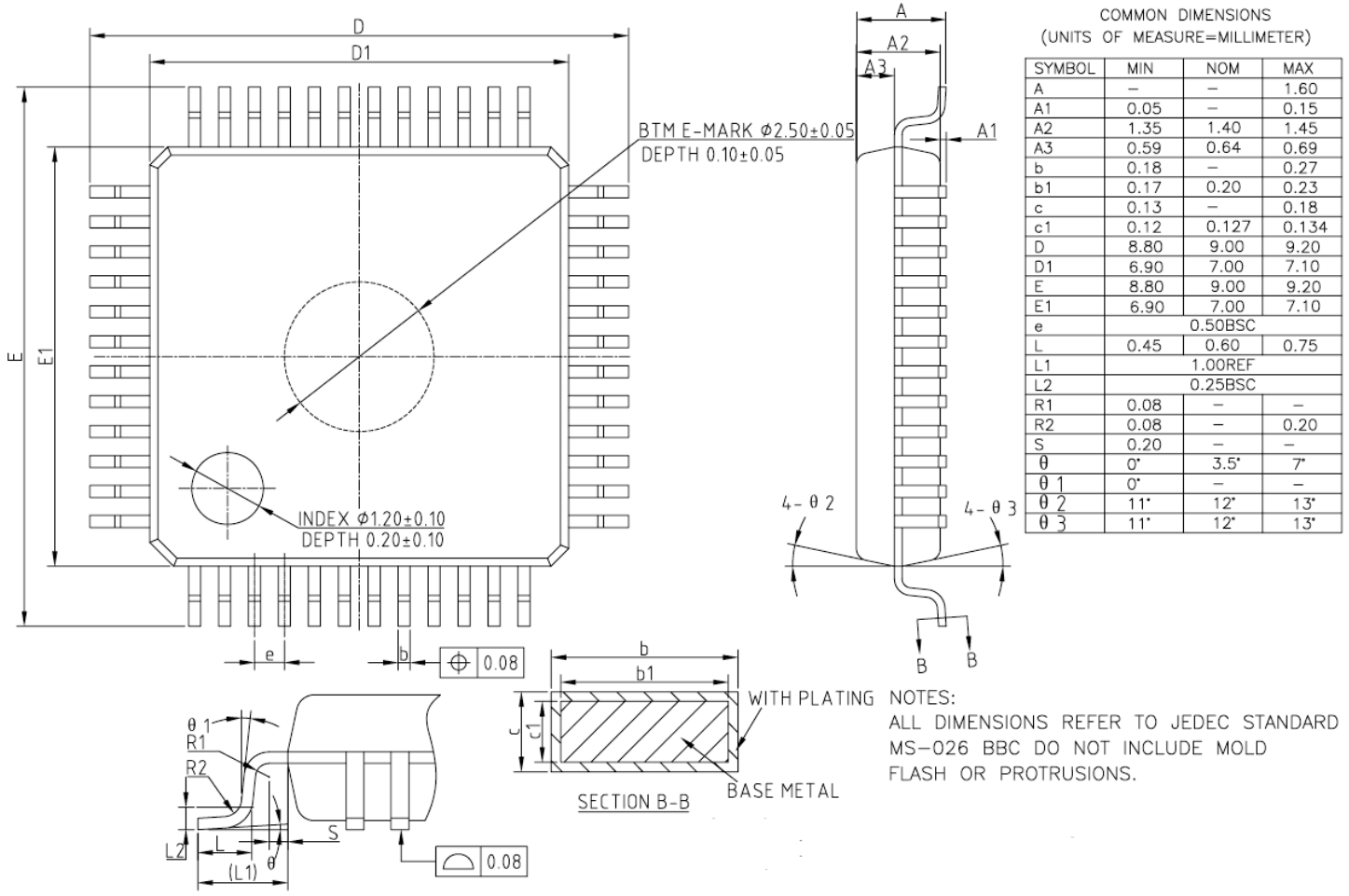


Figure 2-4 FU6813P/LQFP52 Footprint Diagram

3 Package

3.1 LQFP48_7X7

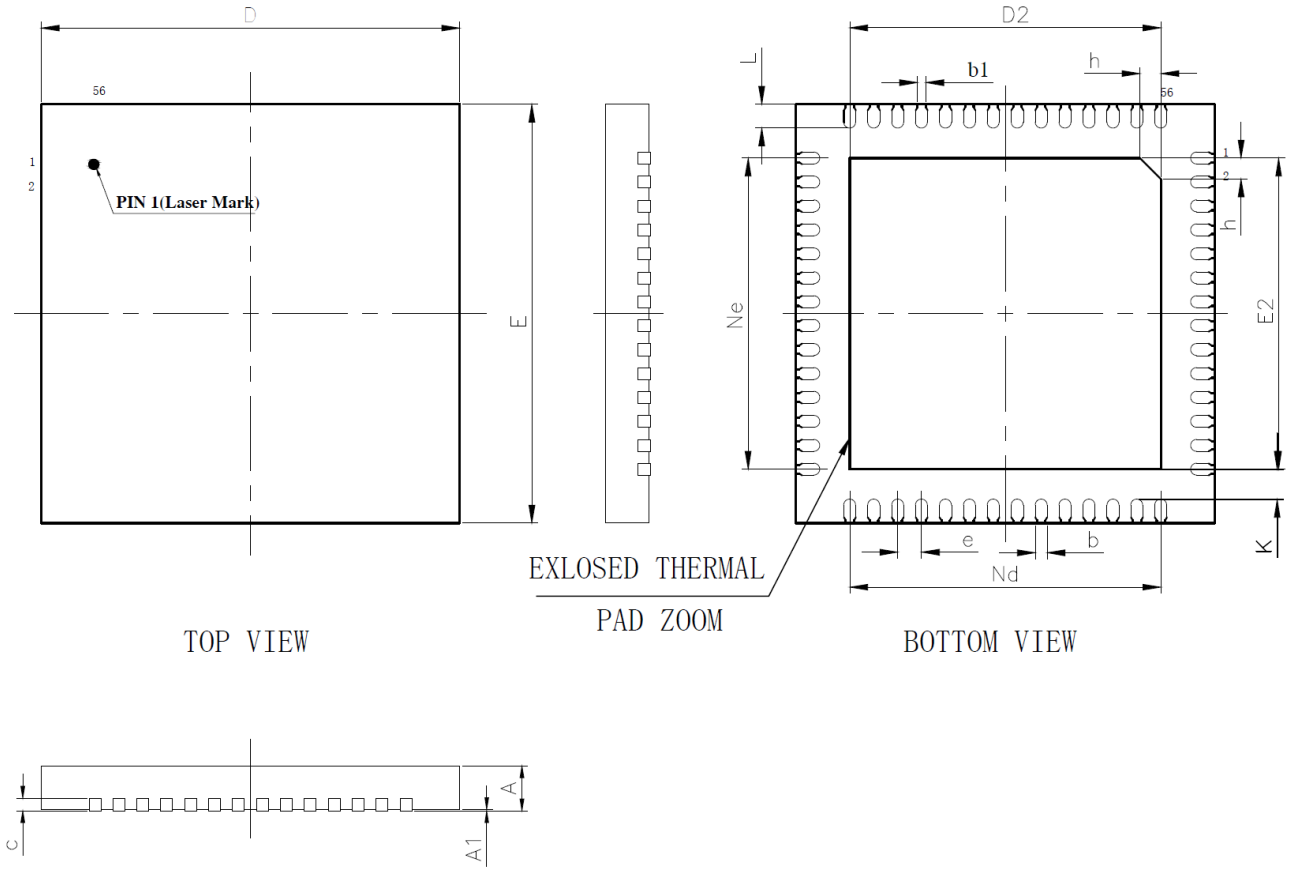


COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	–	–	1.60
A1	0.05	–	0.15
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.18	–	0.27
b1	0.17	0.20	0.23
c	0.13	–	0.18
c1	0.12	0.127	0.134
D	8.80	9.00	9.20
D1	6.90	7.00	7.10
E	8.80	9.00	9.20
E1	6.90	7.00	7.10
e	0.50BSC		
L	0.45	0.60	0.75
L1	1.00REF		
L2	0.25BSC		
R1	0.08	–	–
R2	0.08	–	0.20
S	0.20	–	–
θ	0°	3.5°	7°
θ 1	0°	–	–
θ 2	11°	12°	13°
θ 3	11°	12°	13°

Figure 3-1 LQFP48_7X7 Package Diagram

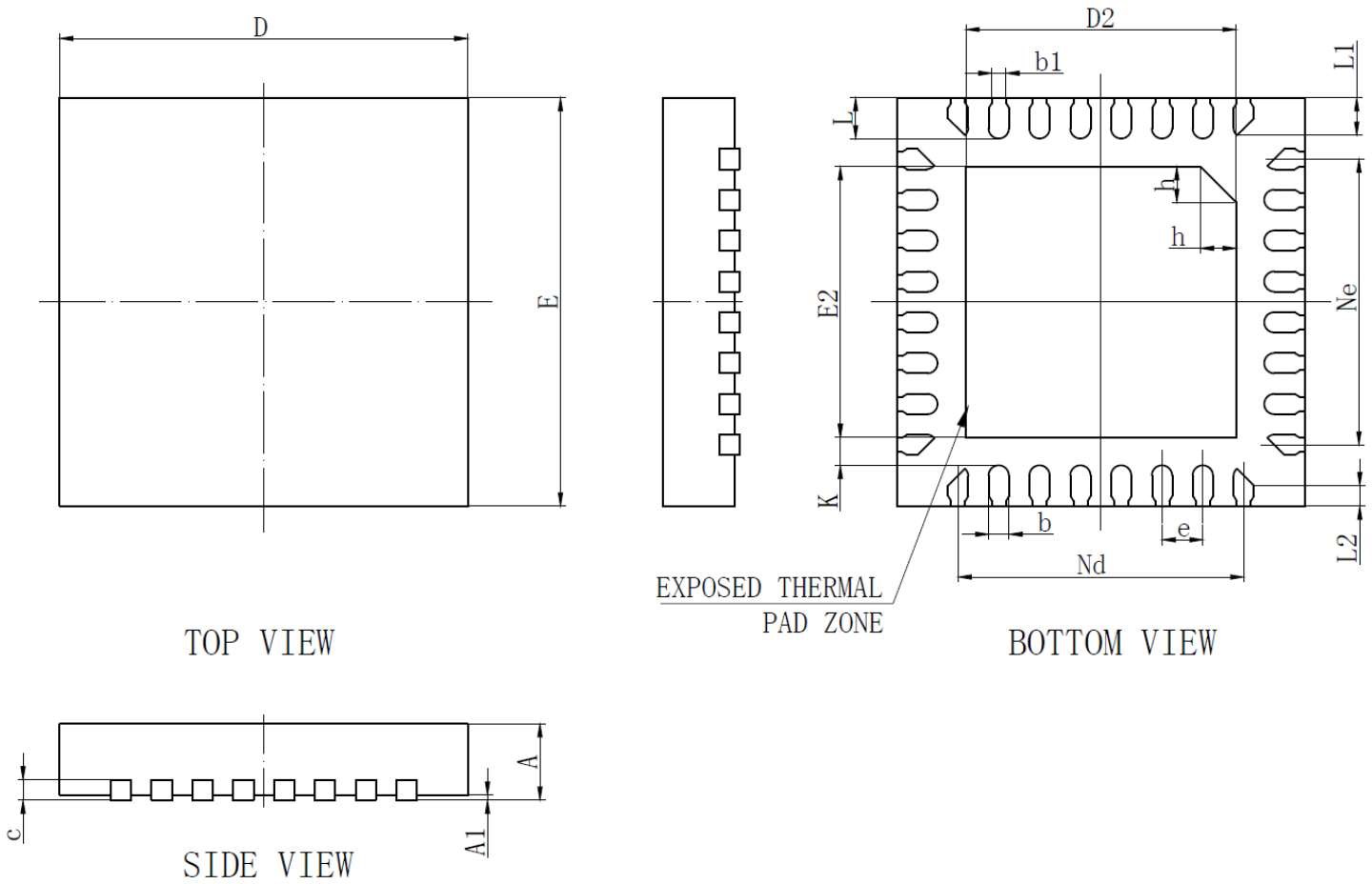
3.2 QFN56_7X7



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.70	0.75	0.80
A1	—	0.02	0.05
b	0.15	0.20	0.25
b1	0.14REF		
c	0.18	0.20	0.25
D	6.90	7.00	7.10
D2	5.10	5.20	5.30
e	0.40BSC		
Nd	5.20BSC		
Ne	5.20BSC		
E	6.90	7.00	7.10
E2	5.10	5.20	5.30
K	0.20	--	--
L	0.35	0.40	0.45
h	0.30	0.35	0.40

Figure 3-2 QFN56_7X7 Package Diagram

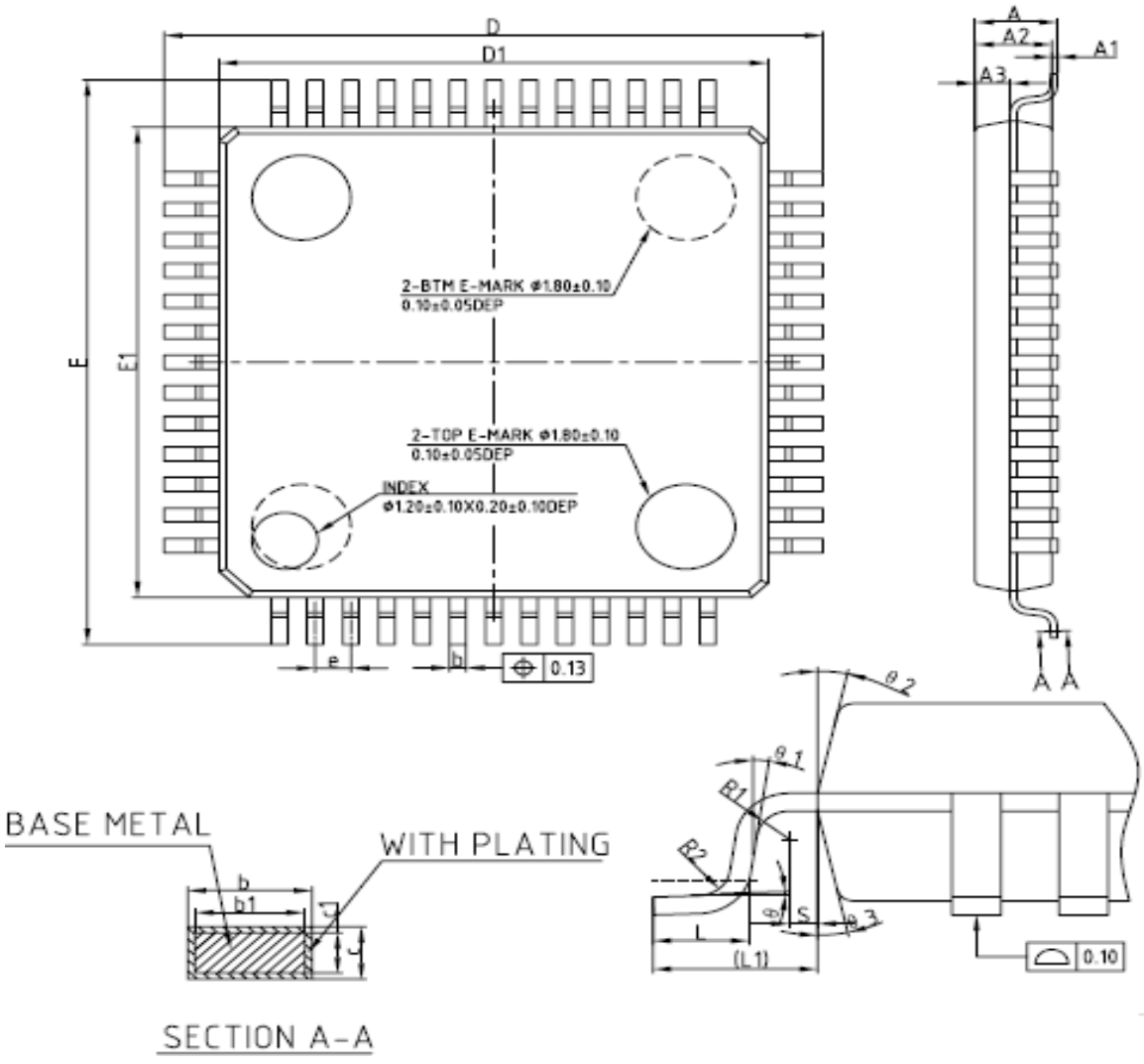
3.3 QFN32_4X4



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.70	0.75	0.80
A1	0	0.02	0.05
b	0.15	0.20	0.25
b1		0.14REF	
c	0.18	0.20	0.25
D	3.90	4.00	4.10
D2	2.60	2.65	2.70
e		0.40BSC	
Nd		2.80BSC	
E	3.90	4.00	4.10
E2	2.60	2.65	2.70
Ne		2.80BSC	
K	0.20	-	-
L	0.35	0.40	0.45
L1	0.30	0.35	0.40
L2	0.15	0.20	0.25
h	0.30	0.35	0.40

Figure 3-3 QFN32_4X4 Package Diagram

3.4 LQFP52_10X10



COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	–	–	1.60
A1	0.05	–	0.20
A2	1.30	1.40	1.50
A3	0.59	0.64	0.69
b	0.28	–	0.37
b1	0.27	0.30	0.33
c	0.13	–	0.18
c1	0.12	0.13	0.14
D	11.80	12.00	12.20
D1	9.90	10.00	10.10
E	11.80	12.00	12.20
E1	9.90	10.00	10.10
e	0.55	0.65	0.75
L	0.45	0.60	0.75
L1		1.00REF	
L2		0.25BSC	
R1	0.08	–	–
R2	0.08	–	–
S	0.15	–	–
θ	0°	3.5°	7°
θ 1	0°	–	–
θ 2	11°	12°	13°
θ 3	11°	12°	13°

NOTES:
ALL DIMENSIONS REFER TO JEDEC STANDARD MS-026 BCE DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS.

Figure 3-4 QFN32_4X4 Package Diagram

4 Ordering Information

Table 4-1 Product Selection Guide

Type	MIPS(Peak)	FLASH(KB)	XRAM(KB)	Oscillation Mode				Pre-driver Interface			Drive Control			I2C/UART/SPI	DMA	GPIO	Timer	Analog Peripherals								Leadfree	Package
																		ADC			DAC						
				Internal Fast Clock	External Fast Clock	Internal Slow Clock	External Slow Clock	6N Pre-driver	3P3N Pre-driver	Gate Driver	BLDC	SVPWM	FOC					Number	Channel	Bits	Number	Bits	VREF	Amplifier	Comparator		
FU6813L	24	32	1.5	√	—	√	√	—	—	√	√	√	√	√	√	34	6	1	14	12	3	8/6	√	4	4	√	LQFP48 (7x7 mm)
FU6813N	24	32	1.5	√	—	√	√	—	—	√	√	√	√	√	√	20	6	1	9	12	3	8/6	√	2	4	√	QFN32 (4x4 mm)
FU6863Q	24	32	1.5	√	—	√	√	√	—	—	√	√	√	√	√	32	6	1	14	12	3	8/6	√	4	4	√	QFN56 (7x7 mm)
FU6813P	24	32	1.5	√	—	√	√	—	—	√	√	√	√	√	√	35	6	1	14	12	3	8/6	√	4	4	√	LQFP52 (10x10mm)

5 Electrical Characteristics

5.1 Absolute Maximum Ratings

Table 5-1 Absolute value of electrical characteristics

Parameter	Conditions	Min	Typ	Max	Units
Ambient temperature T_A		-40	—	85	℃
Ambient temperature T_A	For FU6813L, FU6813N and FU6863Q, VCC ≤ 12V, Ivcc ≤ 30mA	-40	—	105	℃
Ambient temperature T_A	For FU6813, Single Supply LV Mode, VCC = VDD5 = 5V	-40	—	125	℃
Junction temperature T_J		-40	—	150	℃
Storage temperature		-65	—	150	℃
VCC relative to VSS		-0.3	—	36	V
VDD5 relative to VSS		-0.3	—	6.5	V
VDRV relative to VSS	For FU6863	-0.3	—	22	V
VBU, VBV, VBW (floating voltage) relative to VSS	For FU6863	-0.3	—	100	V
VSU to VSV or VSW , and vice versa	For FU6863	-0.3	—	VBU-22, VBV-22, VBW-22	V
RSTN, VCC_MODE and GPIO relative to VSS		-0.3	—	VDD5+0.3	V

Note:

Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

5.2 Global Electrical Characteristics

Table 5-2 Global Electrical Characteristics (For FU6813)

T_A = 25℃, VCC = 5V~24V unless specified

Parameter	Conditions	Min	Typ	Max	Units
VCC relative to VSS	Single Supply HV Mode VCC_MODE = 0	5	—	24	V
	Dual Sipply Mode VCC_MODE = 1, VCC ≥ VDD5 (2)	5	—	36	V
	Single Supply LV Mode VCC_MODE = 1, short with VDD5	3	—	5.5	V

	(2)				
VDD5 relative to VSS	VCC_MODE = 1, short with VCC (2)	3	—	5.5	V
System clock		—	24	—	MHz
Operating current I_{VCC}	(1)	—	24	—	mA
Standby current I_{VCC_STB}	(1)	—	6	—	mA
Sleep current I_{VCC_SLP}		—	50	150	μA

Note:

1. The characteristics may vary for different program code
2. Maintain VDD5 withing 5~5.5V when erasing or programing of Flash
3. Unless specified, for VCC_MODE = 1, VCC_MODE = VDD5

Table 5-3 Global Electrical Characteristics (for FU6863)

T_A = 25℃, VCC = 5V~24V unless specified

Parameter	Condition	Min	Typ	Max	Unit
VCC relative to VSS	Single Supply HV Mode VCC_MODE = 0	5	—	24	V
	Dual Sipply Mode VCC_MODE = 1, VCC ≥ VDD5 (2)	5	—	36	V
VDD5 relative to VSS	VCC_MODE = 1, short with VCC (2)	3	—	5.5	V
VDRV relative to VSS		7	—	18	V
VBU, VBV, VBW (floating voltage) relative to VSS		—	—	100	V
VSU to VSV or VSW , and vice versa		—	—	18	V
System clock		—	24	—	MHz
Operating current I_{VCC}	(1)	—	24	—	mA
Standby current I_{VCC_STB}	(1)	—	6	—	mA
Sleep current I_{VCC_SLP}	T_A = 25℃	—	300	500	μA

Note:

1. The characteristics may vary for different program code
2. Maintain VDD5 withing 5~5.5V when erasing or programing of Flash

5.3 GPIO Electrical Characteristics

Table 5-4 GPIO Electrical Characteristics

T_A = 25℃, VCC = 5V~24V, VCC_MODE = 0 unless sepcified

Parameter	Conditions	Min	Typ	Max	Units
Output rising time	50pF load, measure from 10% to 90%, T_A = 25 ℃	—	15	—	ns
Output falling time	50pF load, measure from 90% to 10%, T_A = 25 ℃	—	13	—	ns
Output high voltage V_{OH}	I_{OH} = 4mA	VDD-0.7	—	—	V
Output low voltage V_{OL}	I_{OL} = 8mA	—	—	0.7	V
Input high voltage V_{IH}	(1)	0.7×VDD5	—	—	V
Input low voltage V_{IL}		—	—	0.2×VDD5	V
Pull-up resistance, excluding P0[2:0], P1[6:3], P2[1], P3[7:6]	Vin = 0V, T_A = 25℃	—	33	—	kΩ
Pull-up resistor, P0 [2:0], P1 [6:3], P2 [1] , P3[7:6]	Vin=0V, T_A=25℃	—	5	—	kΩ

(1) When VDD5=5V, the minimum V_{IH} value can be 0.6*VDD5

5.4 Electrical Characteristics of Gate Driver IO (for FU6813)

Table 5-5 Electrical Characteristics of Gate Driver IO

T_A = 25℃,VCC = 5V~24V, VCC_MODE = 0 unless specified

Parameter	Conditions	Min	Typ	Max	Units
Output source current	T_A = 25℃	—	50	—	mA
Output sink current	T_A = 25℃	—	100	—	mA
Output rising time	50pF load, measure from 10% to 90%, T_A = 25℃	—	7	—	ns
Output falling time	50pF load, measure from 90% to 10%, T_A = 25℃	—	5	—	ns

5.5 Predriver 6N IO Electrical Characteristics (for FU6863)

Table 5-6 Predriver 6N IO Electrical Characteristics

Parameter	Conditions	Min	Typ	Max	Units
Output high peak current		—	0.6	—	A
Output low peak current		—	0.6	—	A
Output rising time	1nF load, measure from 10% to 90% time	—	15	30	ns
Output falling time	1nF load, measure from 90% to 10% time	—	15	30	ns

5.6 ADC Electrical Characteristics

Table 5-7 ADC Electrical Characteristics

T_A = 25℃, VCC = 5V~24V unless specified

Parameter	Conditions	Min	Typ	Max	Units
INL	12 bit mode (1)	—	2	—	LSB
DNL	12 bit mode (1)	—	1.5	—	LSB
OFFSET	12 bit mode (1)	—	10	—	LSB
SNR	f_{IN} = 350kHz (1)	—	70.8	—	dB
ENOB	f_{IN} = 350kHz (1)	—	10.5	—	Bit
SFDR	f_{IN} = 350kHz (1)	—	68.2	—	dB
THD	f_{IN} = 350kHz (1)	—	67	—	dB
Input resistance R_{IN}	(1)	—	800	—	Ω
Input capacitance C_{IN}	(1)	—	30	—	pF
Conversion time	(1)	—	13	—	ADCLK
Sampling time	(1)	3	—	63	ADCLK

Note：

1. ADCLK = 12MHz

5.7 Electrical Characteristics of Reference Voltage

Table 5-8 VREF& VHALF

T_A = -40~85℃, VCC = 5V~24V unless specified

Parameter	Conditions	Min	Typ	Max	Units
VREF	VREFVSEL = 00B	—	4.5	—	V
	VREFVSEL = 01B	—	VDD5	—	V
	VREFVSEL = 11B	—	4	—	V
	VREFVSEL = 10B	—	3	—	V
VHALF		—	VREF/2	—	V

5.8 Electrical Characteristics of Operation Amplifier

Table 5-9 Electrical Characteristics of Operation Amplifier

T_A = 25℃, VCC = 5V~24V unless specified

Parameter	Conditions	Min	Typ	Max	Units
Common mode input range V_{ICMR}		0	—	VDD5-0.5	V
Input offset voltage V_{OS}	T_A = 25℃	—	5	10	mV
Open loop gain A_{OL}	R_L = 100kΩ	—	80	—	dB
UGBW, unit gain bandwidth	C_L = 40pF	6	10	—	MHz

SR, slew rate of Op Amp	C_L = 40pF	10	15	—	V/μs

5.9 Electrical Characteristics of HALL/BEMF

Table 5-10 Electrical Characteristics of HALL/BEMF

T_A = 25℃, VCC = 5V~24V, VCC_MODE = 0 unless specified

Parameter	Conditions	Min	Typ	Max	Units
On-chip BEMF resistance		5.4	6.8	8.2	kΩ
on-chip BEMF resistance mismatch value		—	1	—	%

5.10 Electrical Characteristics of Oscillator

Table 5-11 Electrical Characteristics of OSC

T_A = -40~85℃, VCC = 5V~24V, VCC_MODE = 0 unless specified

Parameter	Conditions	Min	Typ	Max	Units
Internal oscillator frequency		23.5	24	24.5	MHz
WDT oscillator frequency		29	32.8	37	kHz

5.11 Electrical Characteristics of Reset

Table 5-12 Electrical Characteristics of Reset

T_A = 25℃, VCC = 5V~24V, VCC_MODE = 0 unless specified

Parameter	Conditions	Min	Typ	Max	Units
Minimum negative pulse width for reset		—	25	50	μs

5.12 Electrical Characteristics of LDO

Table 5-13 LDO Electrical Characteristics of LDO

T_A = 25℃, VCC = 5V~24V, VCC_MODE = 0 unless specified

Parameter	Conditions	Min	Typ	Max	Units
VDD5 Voltage	VCC = 7V~30V, VCC_MODE = 0	4.7	5	5.3	V
VDD18 Voltage		—	1.85	—	V

5.13 Thermal Information

Table 5-14 LQFP48 Package Thermal Resistance

Parameter	Conditions	Value	Units
θ_{JA} junction-to-ambient thermal resistance	(1), (3)	52.4	℃/W
	(2), (3)	72.2	℃/W
θ_{JC} junction-to-case(top)	(2), (3)	17	℃/W

thermal resistance			

(1) JEDEC Standard, 2S2P PCB

(2) JEDEC Standard, 1S0P PCB

(3) The results of test will vary if the application situation is changed

Table 5-15 QFN56 Package Thermal Resistance

Parameter	Conditions	Value	Units
θ_{JA} junction-to-ambient thermal resistance	(1), (3)	33	℃/W
	(2), (3)	55	℃/W
θ_{JC} junction-to-case(top) thermal resistance	(1), (3)	9.2	℃/W

(1) JEDEC standard, 2S2P PCB

(2) JEDEC standard, 1S0P PCB

(3) The results of test will vary if the application situation is changed

Table 5-16 QFN32 Package Thermal Resistance

Parameter	Conditions	Value	Units
θ_{JA} junction-to-ambient thermal resistance	(1), (3)	47	℃/W
	(2), (3)	74	℃/W
θ_{JC} junction-to-case(top) thermal resistance	(1), (3)	20	℃/W

(1) JEDEC standard, 2S2P PCB

(2) JEDEC standard, 1S0P PCB

(3) The results of test will vary if the application situation is changed

6 Reset Source

6.1 Reset Source (RST_SR)

The chip has 8 reset sources:

(1) Power on reset;

(2) External reset

(3) Low voltage reset

(4) Electrical over stress reset

(5) Watch-dog reset

(6) Flash operation error reset

(7) Debug reset

(8) Software reset

The reset flag is saved in a register, RST_SR, which can be read to check the reset source. The latest reset will clear the entire register and update it with the latest flag. Write operation of 1 to RST_SR (RSTCLR) will clear all reset flags.

Table 6-1 Reset sources control RST_SR (0xC9)

Bit	7	6	5	4	3	2	1	0
Name	RSTPOW/ RSTCLR	RSTEXT	RSTLVD	RSTEOS	RSTWDT	RSTFED	RSTDBG	SOFTR
Type	W/R	R	R	R	R	R	R	W/R
Reset	X	X	X	X	X	X	X	X

Field	Name	Description
[7]	RSTPOW/ RSTCLR	Reading: Power-on reset flag 0: Last reset not caused by power-on reset. 1: Last reset caused by power-on reset. Writing: Rest clear register. Write operation of '1' will clear all reset flags.
[6]	RSTEXT	External reset flag 0: Last reset not caused by external reset. 1: Last reset caused by external reset.
[5]	RSTLVD	Low voltage reset flag 0: Last reset not caused by low voltage reset. 1: Last reset caused by low voltage reset.
[4]	RSTEOS	Electrical over stress reset flag 0: Last reset not caused by electrical over stress reset. 1: Last reset caused by electrical over stress reset.
[3]	RSTWDT	Watch-dog overflow reset flag 0: Last reset not caused by watch dog overflow reset. 1: Last reset caused by watch-dog overflow reset.
[2]	RSTFED	Flash operation error flag 0: Last reset not caused by flash operation error reset. 1: Last reset caused by flash operation error reset.
[1]	RSTDBG	Debug reset flag 0: Last reset not caused by debug reset. 1: Last reset caused by debug reset.
[0]	SOFTR	Software reset Writing: Write operation of '1' to reset and trigger a boot process. Write operation of '0' is invalid. Reading: 0: Last reset not caused by software reset. 1: Last reset caused by software reset.

6.2 Power-on Reset/External Reset

Powering on or keeping RSTN low voltage longer 25μs will respectively generate an internal or external device reset. The MCU will execute the program from address 0x0000.

6.3 Low Voltage Reset (LVR)

Device has low voltage detect circuit to check on VDD5. When enabled, the LVR circuit will generate a reset signal to reset system if VDD5 voltage is lower than voltage gate. Voltage gate level is configurable in registers.

6.4 Watch-dog Reset

Watch-dog counter overflow can generate a system reset. If the watch-dog counter function is enabled, the MCU regularly restarts the watch-dog counter to avoid the watch-dog reset.

6.5 RSTFED Reset

Flash operation module provides user commands MOVX @dptr, A to write or erase flash, and MOVC to read flash. However, if these commands attempt to operate on code-protected sectors, a flash-error detect reset will be generated. This function is a precaution to protect user's flash code area. RSTFED rest source is always enabled.

7 Interrupt Handler

7.1 Interrupt Register

7.1.1 IE (0xA8)

Table 7-1 IE (0xA8) Interrupt Enable

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

Bit	Name	Function
[7]	EA	Globally disable all interrupts. It overrides individual interrupt mask settings. 0: Disable all interrupt sources. 1: Enable each interrupt according to its individual setting.
[6]	RTCIE	Enable Real-Time-Clock interrupt 0: Disable RTC interrupt 1: Enable RTC interrupt
[5]	RSV	Reserved
[4]	ES0	Enable UART1 interrupt 0: Disable UART1 interrupt 1: Enable UART1 interrupt
[3]	SPIIE	Enable SPI Interrupt 0: Disable SPI Interrupt 1: Enable SPI Interrupt
[2]	EX1	Enable external interrupt 1 0: Disable external interrupt 1 from INT1 (P1/P2) inputs 1: Enable external interrupt 1 from INT1 (P1/P2) inputs
[1]	TSDIE	Enable Temperature-Sensor-Detect (TSD) interrupt 0: Disable TSD interrupt 1: Enable TSD interrupt
[0]	EX0	Enable external interrupt 0 0: Disable external interrupt 0 from INT0 (P1/P2) inputs 1: Enable external interrupt 0 from INT0 (P1/P2) inputs

7.1.2 IP0 (0xB8)

Table 7-2 IP0 (0xB8) Interrupt Priority-0

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	Function
[7:6]	PDRV	Driver interrupt priority
[5:4]	PX1	INT1 (External Interrupt 1) priority
[3:2]	PX0	INT0 (External Interrupt 0) priority
[1:0]	PLVW_TSD	LVW (Low-voltage-Warning) interrupt and TSD interrupt priority

Note: Interrupt priority value is arranged from 0 to 3. Bigger value means the higher interrupt priority level.

7.1.3 IP1 (0xC0)

Table 7-3 IP1 (0xC0) Interrupt Priority-1

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	Function
[7:6]	PCMP	Analog comparator interrupt priority
[5:4]	PADC	ADC interrupt priority
[3:2]	PTIM1	Timer 1 interrupt priority
[1:0]	PTIM2	Timer 2 interrupt priority

Note: Interrupt priority value is arranged from 0 to 3. Bigger value means the higher interrupt priority level.

7.1.4 IP2 (0xC8)

Table 7-4 IP2 (0xC8) Interrupt Priority-2

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	Function
[7:6]	PTIM4	Timer 4 interrupt priority
[5:4]	PSYSTICK	SYSTICK interrupt priority
[3:2]	PTIM3	Timer 3 interrupt priority
[1:0]	PRTC	RTC interrupt priority

Note: Interrupt priority value is arranged from 0 to 3. Bigger value means the higher interrupt priority level.

7.1.5 IP3 (0xD8)

Table 7-5 IP3 (0xD8) Interrupt Priority-3

Bit	7	6	5	4	3	2	1	0
Name	PDMA		PSPI_UT2		PI2C_UT1		RSV	
Type	R/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	Function
[7:6]	PDMA	DMA interrupt priority
[5:4]	PSPI_UT2	SPI and UART2 interrupt priority
[3:2]	PI2C_UT1	I2C and UART1 interrupt priority
[1:0]	RSV	Reserved

Note: Interrupt priority value is arranged from 0 to 3. Bigger value means the higher interrupt priority level.

7.1.6 TCON (0x88)

Table 7-6 TCON (0x88)

Bit	7:6	5	4：3	2	1：0
Name	RSV	TSDIF	IT1	IF0	IT0
Type	R	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0

Bit	Name	Function
[7:6]	RSV	Reserved
[5]	TSDIF	TSD interrupt flag 0: No TSD interrupt request. 1: TSD interrupt request. A software write of 0, set the bit to 0. This flag bit is often used with the Temperature Sensor Detect Flag (TSDF).
[4:3]	IT1[1:0]	INT1 external interrupt-1 sensitivity level setting 2’b00: Active rising edge. 2’b01: Active falling edge. 2’b1x: Voltage level changes.
[2]	IF0	INT0 (external interrupt-0 flag) 0: No INT0 interrupt is requested. 1: INT0 interrupt is requested. A software write of 0, set the bit to 0.
[1:0]	IT0[1:0]	INT0 external interrupt-1 trigger event: 2’b00: Active rising edge. 2’b01: Active falling edge. 2’b1x: Voltage level changes.

7.2 Interrupt Summary

Table 7-7 Interrupt Summary

Interrupt source	Priority Order	Interrupt Vector	Interrupt Flag	Can clear by SW	Interrupt Enable	Priority
Reset	Top	0x0000	N/A	N/A	Always enabled	Highest
LVW/ TSD	0	0x0003	LVSR[0]/ TCON[5]	Y	CCFG1[6]/ IE[1]	IP0[1:0]
External interrupt (INT0)	1	0x000B	TCON[2]	Y	IE[0]	IP0[3:2]
External interrupt (INT1)	2	0x0013	P1IF[7:0]/ P2IF[7:0]	Y	IE[2]	IP0[5:4]
DRV interrupt	3	0x001B	DRV_SR[5:4]	Y	DRV_SR[2:0]	IP0[7:6]
TIM2 interrupt	4	0x0023	TIM2_CR1[7:5]	Y	TIM2_CR1[4:3] TIM2_CR0[3]	IP1[1:0]
TIM1 interrupt	5	0x002B	TIM1_SR[4:0]	Y	TIM_IER[4:0]	IP1[3:2]
ADC interrupt	6	0x0033	ADC_CR[0]	Y	ADC_CR[1]	IP1[5:4]
CMP interrupt	7	0x003B	CMP_SR[7:4]	Y	CMP_CR0[7:0]	IP1[7:6]
RTC interrupt	8	0x0043	RTC_STA[6]	Y	IE[6]	IP2[1:0]
TIM3 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]
TIM4 interrupt	11	0x005B	TIM4_CR1[7:5]	Y	TIM4_CR1[4:3] TIM4_CR0[3]	IP2[7:6]
RSV	12	0x0063		Y		IP3[1:0]
I2C 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]/ UT2_RI UT2_TI	Y	IE[3]/ UT2_BAUDH[5]	IP3[5:4]
DMA interrupt	15	0x007B	DMA:CH0INT DMA:CH1INT	Y	DMAIE	IP3[7:6]

The FU6813/63 includes an extended interrupt system, which supports sixteen interrupt sources with four priority levels. User software can set the IP0~IP3 register to control the four priority levels of the sources. Higher priority interrupt can processed during the operation of lower priority interrupt. Same priority interrupt sources will affect the current operation. When many same priority interrupts are requested at the same time, the priority level is shown in Table 5 7.

IE[EA] is interrupt main enabling, EA = 0 disable all interrupts request.

7.3 External Interrupt (INT0/INT1)

External interrupt has two interrupt sources INT0 and INT1. Setting P0.0~P0.6 as digital IO input or enabling CMP4, these ports and CMP4 become external interrupt sources when EX0 = 1. Setting P1.0~P1.7/P2.0~P2.7 as digital IO inputs, these ports can use external INT-1 with setting EX1=1, the 16 interrupt sources use 1 interrupt vector.

INT0 enable register bit is EX0, interrupt flag is IF0, interrupt active sensitivity level setting is IT0. INT0 interrupt source is configurable by LVSR[EXT0CFG] and all sources use 1interrupt vector.

The main enable register bit of INT1 is EX1, 16 PINs interrupt enable bits are controlled by P1IE and P2IE. Interrupt flags are P1IF and P2IF, interrupt active sensitivity level setting is IT1.

NOTE:

When software clears the INT1 interrupt flags, MCU should clear the corresponding bit and shouldn't clear other flag bit. For example, when clear P1IF[0], the software code should be : mov 0D2h,#0FEh.

Table 7-8 IO corresponding to external interrupt 1

SFR ADDR.	Field	Name	Description	R/W	RES. VAL.
0xD1	[7:0]	P1IE[7:0]	P1.7~P1.0 is used as external interrupt sources, enable 0: Disable the port external interrupt. 1: Enable the port external interrupt.	R/W	0x00
0xD2	[7:0]	P1IF[7:0]	P1.7~P1.0 is used as external interrupt sources, interrupt flag 0: No interrupt request from the port. 1: An interrupt request from the port. Software clears the interrupt flag. NOTE: software should not clear the other flag.	R/W	0x00
0xD3	[7:0]	P2IE[7:0]	P2.7~P2.0 is used as external interrupt sources, enable 0: Disable the port external interrupt. 1: Enable the port external interrupt.	R/W	0x00
0xD4	[7:0]	P2IF[7:0]	P2.7~P2.0 is used as external interrupt sources, interrupt flag 0: No interrupt request from the port. 1: Interrupt request exist from the port. Software clears the interrupt flag. NOTE: software should not clear the other flag.	R/W	0x00

8 I2C

I2C is a bi-directional synchronous serial bus with two-wire, Serial Data Line (SDA) and serial clock line (SCL). The SDA and SCL pins both are bi-directional. They each must be connected to VCC using a pull-up resistance. When the bus is free, both pins are high. During the data transfer, only one master device and at least one slave device are connected to the I2C-bus. Other devices will wait for the transfer unless the I2C-bus is free. Master device, which initiates a data transfer on the bus, send the clock signals by SCL bus and slave device address, Send/Receive mode by SDA bus. During this transfer, any device addressed by this master is considered a slave. Each device can operate as transmitter or receiver, depending on the function of device. The process of transfer is shown in Figure 8-1 and Figure 8-2. First, Master device and slave device are connected by slave device address. Then, either master device or slave device sends the data. At last, transfer is ended by master device.

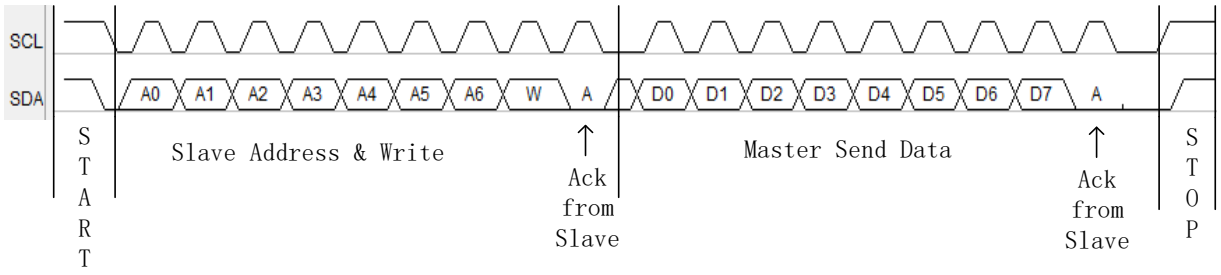


Figure 8-1 Master Device Send Data to Slave Device

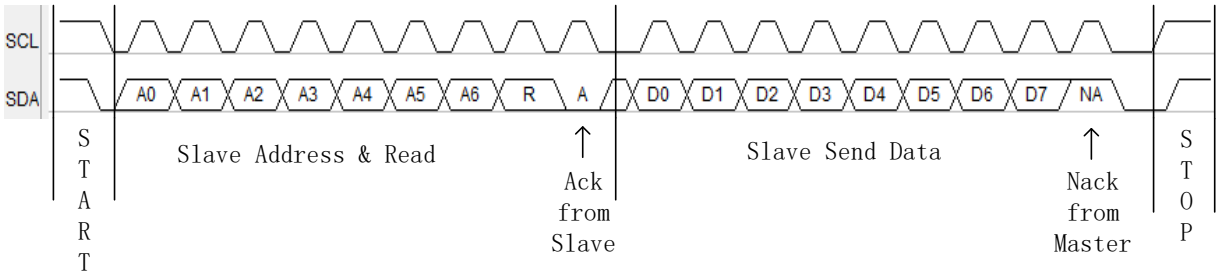


Figure 8-2 Slave Device Send Data to Master Device

The fastest speed of transfer is 1MHz. The process is controlled by STA, DMOD, STR, NACK and STP when I2C is already configured.

8.1 Operating Instructions

8.1.1 Master Mode

1. Configure I2C_CR[I2CMS] = 1 as master mode;
2. Configure I2C_CR[I2CSPD] to select the clock frequency SCL.
3. Configure I2C_ID to set target device address.

4. Configure I2C_SR[DMOD] to set the direction of the write/read.
5. Configure I2C_CR[I2CEN] = 1 to enable the I2C.
6. Configure I2C_SR[I2CSTA] = 1 to send START signal and address.
7. After sending the address and receiving the ACK/NACK, I2C_SR[STR] is set to 1 by hardware and SCL is pulled down by the master, then waiting for the next motion.
8. In the SEND operation, set I2C_SR[STR] = 0 to release SCL after writing I2C_DR register. At this time, the master begins to send data and waits for the ACK/NACK from the slave. When finished the transmit and received the ACK/NACK from the slave, I2C_SR[STR] is set to 1by hardware and SCL is pulled down by the master, then waiting for the next motion;
9. In the RECEIVE operation, set I2C_SR[STR] = 0 to release SCL. At this time, the master begins to receive data. When finished it, I2C_SR[STR] is set to 1by hardware and SCL is pulled down by the master, waiting for setting I2C_SR[NACK] to send ACK/NACK by software. Sending ACK/NACK after setting the I2C_SR[STR] = 0 and releasing SCL. After receiving the next byte, the SCL is forced to pull down by the master.
10. Send STOP condition. If I2C_SR[STOP] is set to 1 during the process of transmit/receive, the master will transmit the STOP after finishing sending/receiving the current byte.

8.1.2 Slave Mode

1. Configure I2C_CR[I2CMS] = 0 as slave mode；
2. Configure I2C_ID[I2CADD] to set the slave address, or configure the I2C_ID[GC] = 1 to enable the General Call mode.
3. Configure I2C_CR[I2CEN] = 1 to enable the I2C.
4. Wait for receiving the “start” command and the address data. After receiving the “start” and address, SCL is pulled down. I2C_SR[I2CSTA] and STR(I2C_STA[2]) is set to 1 by hardware, and then wait for the I2C_SR[NACK] to send ACK/NACK by software. Meanwhile, determine if in slave send mode, then write I2C_DAT register. Set I2C_SR[STR] = 0 and release SCL, send data after sending ACK/NACK. After the slave sends the data and receives the ACK/NACK from the master, SCL is forced to be pulled down by slave. Meantime, I2C_SR[STR] is set 1 by hardware.
5. In RECEIVE mode, the slave set I2C_SR[STR] = 0. Then it is ready for receiving the data. Release SCL, send data, and wait for the salve reception of the data. After that, I2C_SR[STR] is set to 1 by hardware, and SCL is forced to pull down by the master. Then the system waits for I2C_SR[NACK]. After setting I2C_SR[STR] to 0,SCL is released, the system sends out ACK/NACK and receive the next byte and pulled down SCL by the slave.
6. RESTART function: when the slave receives the START signal in the busy state, the current work is suspended, and then waiting for the address data.

8.1.3 I2C Interrupt Sources

If I2C interrupt is allowed, the interrupt will be generated at the following cases.

1. I2C_SR[STR], this interrupt flag is available in both master mode and slave mode. It is used for pull down SCL, and then waiting for the next operation.
2. IS2_SR[I2CSTP] in the slave mode. This flag indicates a reception of the STOP signal in the slave mode (NOTE: STOP bit is the function of stop signal generation).

8.2 I2C Register

8.2.1 I2C_CR (0x4028)

Table 8-1 I2C_CR (0x4028)

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

Bit	Name	Function
[7]	I2CEN	I2C enable 0: I2C is disabled 1: I2C is enabled. The relate GPIO change to I2C mode with OPEN DRAIN Output. Whether to turn on the pull-up of I2C depends on the setting of the IO port.
[6]	I2CMS	I2C mode selection 0: Master mode 1: Slave mode
[5:3]	RSV	Reserved
[2:1]	I2CSPD	I2C speed setting, only valid in the master mode 00: 100KHz transmit speed 01: 400KHz transmit speed 10: 1MHz transmit speed 11: Not support, reserved
[0]	I2CIE	Interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled with the request generated by I2C_SR[I2CIF]

8.2.2 I2C_ID (0x4029)

Table 8-2 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	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:1]	I2CADD	I2C slave address
[0]	GC	This bit is used for supporting the General Call, only valid in the slave mode. 0: The General Call is not supported. 1: The General Call is supported, which means that address 0x00 can response.

8.2.3 I2C_DR (0x402A)

Table 8-3 I2C_DR (0x402A)

Bit	7	6	5	4	3	2	1	0
Name	DATA							
Type	R/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	Function
[7:0]	I2C_DR	I2C data register

8.2.4 I2C_SR (0x402B)

Table 8-4 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	R/W	R/W	R/W0	R/W	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	I2CBSY	I2C operation flag This bit will be zero automatically when I2CEN=0. Master mode: This bit is set to 1 by hardware for sending “START” command. This bit is cleared by hardware for sending “STOP” command. Slave mode: This bit is set to 1 by hardware after receiving “START” command. This bit is cleared by hardware after receiving “STOP” command.
6	DMOD	I2C read/write mode Master mode: 0: Write mode (data are sent from the master and received by the slave) 1: Read mode (data are received by the master and sent from the slave) In Master mode, this bit is modified validly only when: <a> I2CSTA bit is 1 <b> Change the DMOD by setting the I2CSTA bit to 1

<table>
<tr><td></td><td></td><td>Slave mode:
0: Write mode (data are sent from the master and received by the slave)
1: Read mode (data are received by the master and sent from the slave)</td></tr>
<tr><td>5</td><td>RSV</td><td>Reserved</td></tr>
<tr><td>4</td><td>I2CSTA</td><td>Master mode:
This bit is set to 1 by software, and it sends “START” command and the address data with hardware when SCL and SDA are in high-level. It is cleared automatically by hardware when data sending is completed. Setting the I2CSTA to 1, the process of sending data will be continued until all the data have been sent out, and then start to send the START and address data. It is cleared when I2C_CR[I2CEN] = 0.
0: Non-START and the address bytes
1: Send START or RESTART and the address bytes
Slave mode:
This bit set to 1 when the hardware receives the START with the address bytes matched. It is cleared by software.
If the slave receives the START with the address unmatched, this bit will not be set to 1, and will be ignored until receiving the next START.
In slave mode, the current data condition is decided by START and STOP:
Table 8-5 I2C State Flag
<table>
<tr><td>START</td><td>STOP</td><td>Description</td></tr>
<tr><td>0</td><td>0</td><td>The current write/read is the data bytes</td></tr>
<tr><td>0</td><td>1</td><td>The current receiving is STOP</td></tr>
<tr><td>1</td><td>0</td><td>The current receiving is START + address bytes</td></tr>
<tr><td>1</td><td>1</td><td>The current receiving is STOP, and then received one is START +address bytes</td></tr>
</table>
Note: START will be cleared automatically by hardware when I2CEN=0.</td></tr>
<tr><td>3</td><td>I2CSTP</td><td>Master mode:
The software write 1 validly when I2CBSY = 1, and then send STOP by hardware. It is cleared automatically by hardware at the end of the transmission. If I2CSTA and I2CSTP are write at the same time with I2CBSY = 1, I2C will send STOP first. At the end of the transmission of STOP, START and the address bytes are transmitted, at this point STOP interrupt will not be ignored until the transmission of the address bytes is completed. I2CSTP will be forbidden to write till the end of the transmission. It is cleared when I2C_CR[I2CEN] = 0.
0: Not send STOP
1: Send STOP
Slave mode:
This bit will be set to 1 by hardware and be cleared by software after receiving the STOP condition.
NOTE: I2CSTP will be cleared by hardware automatically when I2CEN = 0.</td></tr>
<tr><td>2</td><td>STR</td><td>I2C event completion flag
Being set to1 by hardware and cleared by software. STR will be cleared by</td></tr>
</table>

		hardware automatically when I2CEN = 0. Master mode: STR will be set to 1 at the end of sending the START and address bytes, or data bytes. At the same time, SCL will be pulling down and released till STR is cleared. If I2CSTA and I2CSTP are both logic 1 at the same time, STR will not be set to 1 by hardware at the end STOP transmission. Slave mode: STR will be set to 1 after receiving the START and address bytes, or data bytes. Meanwhile, SCL will be pull down till the STR is cleared.
1	NACK	The state of the 9th bit (acknowledge bit) at the end of byte transmission. It is cleared automatically when I2CEN = 0. 0: ACK 1: NACK Master read mode (I2CMS = 1,DMOD = 1): the acknowledge bit of data bytes. slave write mode (I2CMS = 0,DMOD = 0): the acknowledge bit of data bytes. Pull down the SCL after receiving the data of the 8th bit. 0: Send ACK at the 9th bit 1: Send NACK at the 9th bit. Master write mode (I2CMS = 1, DMOD = 0): the acknowledge bit of data bytes and address bytes. Master read mode (I2CMS = 1, DMOD = 1): the acknowledge bit of address bytes. Slave read mode (I2CMS = 0, DMOD = 1): the acknowledge bit of data bytes. 0: Receive ACK at the 9th bit 1: Receive NACK at the 9th bit. NOTE: whether I2C is in master mode or slave mode, if this bit is I2C SEND acknowledge bit, STR will set to 1 after receiving the 8th bit, and SCL is pulled down to 0. The value of the NACK means that the 9th bit is going to transmit. If this bit is I2C RECEIVE acknowledge bit, STR will be set to 1 after receiving the 9th bit, and SCL is pulled down to 0. The value of the NACK means the 9th bit is in receiving.
0	I2CIF	I2C interrupt requesting flag: clearing this bit will allow the I2C to keep the data transmitting state. 0: Have no I2C interrupt request 1: Have a I2C interrupt request Master mode: I2CIF is logic 1 when STR = 1, otherwise it is logic 0. Slave mode: I2CIF is logic 1 when I2CSTP = 1 or STR = 1, otherwise it is logic 0.

9 SPI

Serial Peripheral Interface (SPI) is a high-speed synchronous serial input/output port. It may work as master device or slave device with 3-wire or 4-wire modes, and it may support many master devices and slave devices on a bus. The SPI module includes MOSI, MISO, SLK and NSS.

MOSI, data signal, is the output pin of master device, and input pin of slave device.

MISO, data signal, is the input pin of master device, and output pin of slave device. MISO pin will be in high-impedance state when SPI is disabled, or has not been selected on the 4-wire slave mode.

SLK, clock signal, is synchronous serial clock reference of data signal. SCLK is output from master device and input to slave device.

NSS is enable signal which is configurable. When SPI work in 3-wire mode, NSS is GPIO. When SPI work in slave mode, NSS is configurable as input pin to detect NSS signal from master device. When SPI work in single-master single-slave mode, NSS pin of master device is configurable as output pin to enable slave device. When SPI work in multi-master mode, NSS pin is configurable as input pin to detect whether bus is busy. When SPI work in single-master multi-slave mode, several IO is configurable as NSS pin to select slave device.

9.1 Operating Declaration

9.1.1 SPI Master Mode Operation

SPI work in master mode when SPI_CR0[SPIMS] = 1. Unless shift register is empty, SPI will not initiate a transfer. The data written to SPI_DR is written to the buffer and SPI_CR1[TXBMT] is set to 0. If the SPI shift register is empty, the data is transferred from send buffer to shift register, and data transmission will be proceeded. The SPI master sends out the data to the MOSI pin, and also transmits a serial clock signal on SCK. The SPI_CR1[SPIF] and SPI_CR1[TXBMT] are set to logic 1 at the end of the transfer. Shift register receive data from MISO and send it to receive buffer. The data can be read by SPI_DR. If data is written to SPI_DR under SPI_CR1[TXBMT] = 0, SPI_CR1[WCOL] will be set to 1 and the data in the send buffer will be maintained.

9.1.1.1 Master Mode Configuration

1. Configure SPI_CR1[NSSMID] to select SPI work mode
2. Configure SPI_CR0[CKPOL] to set clock polarity;
3. Configure SPI_CR0[CKPHA] to set clock phase;
4. Configure SPI_CR0[SPIMS] = 1to set as master mode;
5. Configure SPI_CLK to set SCK frequency;
6. Configure SPI_CR1[SPIEN] = 1 to enable SPI;
7. Configure SPI_DR to write an operation data during sending and receiving a data.

9.1.2 SPI Slave Mode

SPI work in slave mode when SPI_CR0[SPIMS] = 0. SCK signal of slave device is received from master mode (3MHz highest). MOSI pin and MISO pin of slave device will transfer data when SCK signal inputs. At the end of transfer, SPI_CR1[SPIF] and SPI_CR1[TXBMT] are set to logic 1, and SPI_CR0[RXBMT] is set to 0, indicating the unread data. If new data is received under SPI_CR0[RXBMT] = 0, receive buffer will be maintained. If writing data to SPI_DR under SPI_CR1[TXBMT] = 0, SPI_CR1[WCOL] is set to 1 and send buffer data will be maintained. When SPI work in 4-wire mode, NSS pin is input pin and the bit counter is reset on a falling edge of NSS

9.1.2.1 Slave Mode Configuration

1. Configure SPI_CR1[NSSMID] to select 3-wire slave mode or 4-wire slave mode.
2. Configure SPI_CR1[CKPOL] to set clock polarity.
3. Configure SPI_CR1[CKPHA] to set clock phase.
4. Configure SPI_CR1[SPIMS] = 0 to set as slave mode.
5. Configure SPI_CR1[SPIEN] = 1 to enable SPI.
6. Configure SPI_DR, write operation data, wait clock signal send by master device.

9.1.3 SPI Interrupt Source

When SPI interrupts are enabled (SPIIE = 1), the following four flags will generate an interrupt if they are set to logic 1:

Note: All of the following bits must be cleared to 0 by software.

1. The SPI Interrupt Flag, SPIF, is set to logic 1 at the end of each byte transfer. This flag can be enabled under all SPI modes.
2. The Write Collision Flag, WCOL, is set to logic 1 if a write to SPI_DR is attempted in the cases where the send buffer has not been emptied to the SPI shift register. When this happens, the write to SPI_DR will be ignored, and the send buffer will not be written. This flag can be enabled under all SPI modes.
3. The Mode Fault Flag, MODF, is set to logic 1 when SPI is configured as a master in multi-master mode and the NSS pin is pulled down. When a Mode Fault occurs, the SPIMS and SPIEN bits are set to logic 0 to disable SPI. This will allow another master device to access the bus.
4. The Receive Overrun Flag, RXOVRN, is set to logic 1 when SPI is configured as a slave, the receive buffer still holds the unread byte from a previous transfer until the transfer is completed. The new byte is not transferred to the receive buffer, and it allows the previously received data to be read.

The data byte which caused the overrun is lost.

.

9.1.4 SPI Operation Modes

Several operation mode, 3-wire mode, 4-wire slave mode, 4-wire single-master mode and 4-wire multi-master mode, can be configured by SPI_CR1[NSSMID] .

3-wire mode is selected when SPI_CR1[NSSMID] = 00. Under this mode, NSS is not used, and is not mapped to external port pin. Since the absence of NSS, the slave device is unique. Figure 9-1 shows a connection diagram between master device in 3-wire master mode and slave device in 3-wire slave mode.

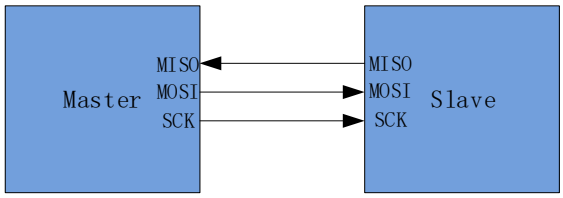


Figure 9-1 Connection Diagram between Master Device in 3-Wire Master Mode and Slave Device in 3-Wire Slave Mode

4-wire mode is selected when SPI_CR1[NSSMID] = 01, NSS is input pin as enable signal. When SPI_CR0[METEN] = 1, SPI work in multi-master mode. When SPI_CR0[METEN] = 0, SPI work in slave mode. For multi-master mode, the master device, which NSS is pulled down, will work in slave mode and disable SPI. For slave mode, the NSS signal must be driven low at least 2 system clocks before the first active edge of SCK for each byte transfer. Figure 9-2 shows a connection diagram between devices in 4-wire multi-master mode.

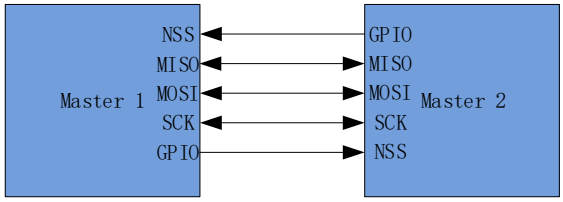


Figure 9-2 Connection Diagram between Devices in 4-Wire Multi-Master Modes

4-wire mode is selected when SPI_CR1[NSSMID] = 1x (only for master mode). NSS is output pin and can be configured by SPI_CR1[NSSMID0]. When SPI_CR1[NSSMID0] = 1, NSS outputs high level. SPI_CR1[NSSMID0] = 0, NSS outputs low level. Figure 9-3 shows a connection diagram between master device in 4-wire single-master mode and slave device in 4-wire slave mode.

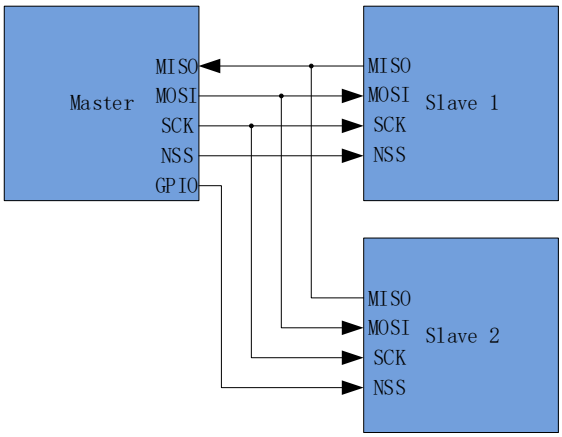


Figure 9-3 Connection Diagram between Master Device in 4-Wire Single-Master Mode and Slave Device in 4-Wire Slave Mode

9.1.5 Serial Clock Timing Sequence

Four combinations of serial clock phase and polarity can be selected by using the clock control bits (SPI_CFG) in the SPI configuration register. The CKPHA bit selects one of two clock phases (edge used to latch the data). The CKPOL bit selects the value in the active-high or active-low clock. Both master and slave devices must be configured to use the same clock phase and polarity. SPI should be disabled (by clearing the SPIEN bit) when changing the clock phase or polarity. The clock and data relationships for master mode are shown in Figure 9-4. For slave mode, the clock and data relationships are shown in Figure 9-5 and Figure 9-6.

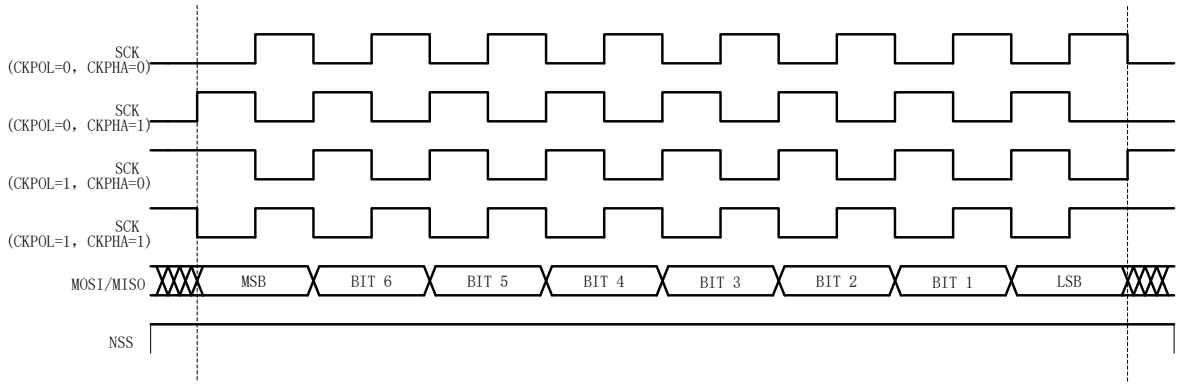


Figure 9-4 Master Mode Data/Clock Timing

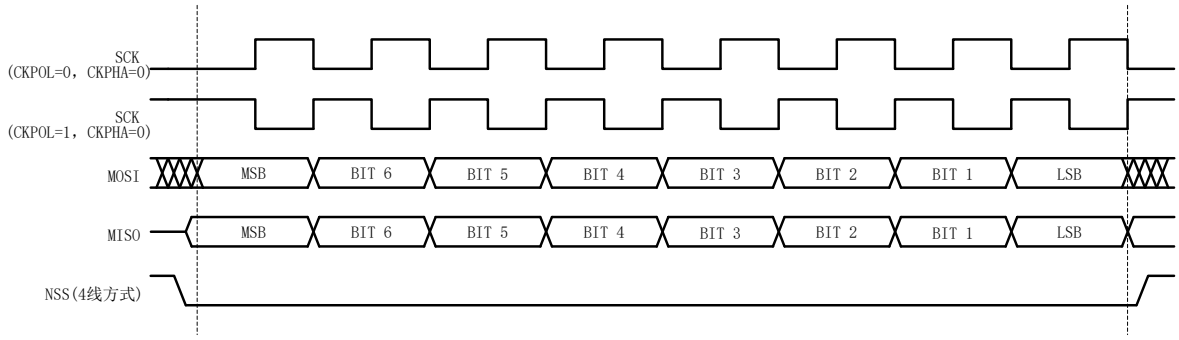


Figure 9-5 Slave Mode Data/Clock Timing (CKPHA=0)

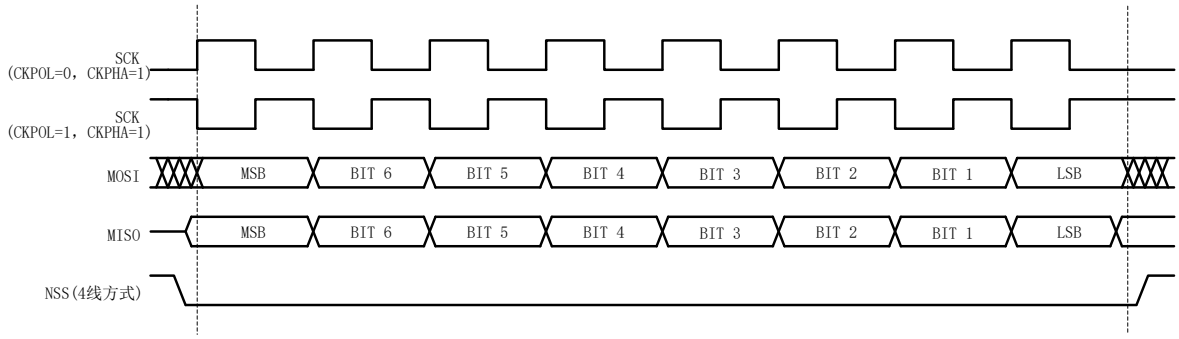


Figure 9-6 Slave Mode Data/Clock Timing (CKPHA=1)

9.2 SPI Register

9.2.1 SPI_CR0 (0x4030)

Table 9-1 SPI_CR0 (0x4030)

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

Bit	Name	Function
[7]	SPIBSY	This bit is set to logic 1 when SPI transfer is in progress (master or slave mode).
[6]	SPIMS	Master/slave mode setting 0: Slave mode 1: Master mode
[5]	CKPHA	SPI clock phase. 0: Data centered on the first edge of SCK period. 1: Data centered on the second edge of SCK period.
[4]	CKPOL	SPI clock polarity 0: SCK line is low in idle state. 1: SCK line is high in idle state.

[3]	SLVSEL	This bit is set to logic 1 whenever the NSS pin is low indicating SPI is the selected slave. It is cleared to logic 0 when NSS is high (not selected as slave device). This bit does not indicate the instantaneous value at the NSS pin, but the noise filter signal input of the pin.
[2]	NSSIN	This bit shows the instantaneous value of the NSS port pin when the register is read.
[1]	SRMT	Shift register empty (valid in slave mode only). Under slave mode, this bit is set to logic 1 when all data has been transferred in/out of the shift register, and there is no new data to be read from the send buffer, or write to the receive buffer. It returns to logic 0 when a data byte is transferred to the shift register from the send buffer, or by a transition on SCK. Under master mode, this bit is set to logic 1.
[0]	RXBMT	Receive buffer empty (valid in slave mode only). Under slave mode, this bit is set to logic 1 if the receive buffer has been read and the buffer does not contain new data. If new data is available in the receive buffer and the data has not been read, this bit returns to logic 0. Under master mode, this bit is set to logic 1.
Phase Mode/Clock Polarity: 00：Rising edge receive, falling edge send, idle level is low 01：Rising edge send, falling edge receive, idle level is high 10：Rising edge send, falling edge receive, idle level is low 11：Rising edge receive, falling edge send, idle level is high		

9.2.2 SPI_CR1 (0x4031)

Table 9-2 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	Function
[7]	SPIIF	SPI interrupt flag This bit is set to logic 1 by hardware at the end of a data transfer. If SPI interrupts are enabled, an interrupt will be generated. This bit is not automatically cleared by hardware, and must be cleared by software.
[6]	WCOL	Write collision flag This bit is set to logic 1 if a write to SPI_DR is attempted when TXBMT is 0. When this occurs, the write to SPI_DR will be ignored, and the transmit buffer will not be written. If SPI interrupts are enabled, an interrupt will be generated. This bit is not automatically

		cleared by hardware, and must be cleared by software.
[5]	MODF	Mode fault flag This bit is set to logic 1 by hardware when a master mode collision is detected (NSS is low, MSTEN = 1, and NSSMD[1:0] = 01). If SPI interrupts are enabled, an interrupt will be generated. This bit is not automatically cleared by hardware, and must be cleared by software.
[4]	RXOVRN	Receive overrun flag (valid in slave mode only) This bit is set to logic 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 is shifted into the SPI shift register. This bit is not automatically cleared by hardware. It must be cleared by software.
[3:2]	NSSMID	SPI Mode selection Selects between the following NSS operation modes: 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 is an input to the device. 1x: 4-wire single-master mode. NSS signal is mapped as an output from the device and will assume the value of NSSMD0.
[1]	TXBMT	Send buffer empty flag This bit will be set to logic 0 when new data has been written to the send buffer. This bit will be set to logic 1 when data in the send buffer is transferred to the SPI shift register, indicating that it is safe to write a new byte to the send buffer.
[0]	SPIEN	SPI Enable. 0: SPI disabled. 1: SPI enabled.

9.2.3 SPI_CLK (0x4032)

Table 9-3 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	Function
[7:0]	SPI_CLK	These bits determine the frequency of the SCK output when the SPI module is configured for master mode operation. fsck = sysclk/(2 × (SPI_SCR[7:0] + 1)) for 0 ≤ SPI_CLK ≤ 255 Example: if sysclk = 24MHz,SPI_SCR=0x04, fsck = 24000000/(2 × (4+1)) = 2400kHz

9.2.4 SPI_DR (0x4033)

Table 9-4 SPI_DR (0x4033)

Bit	7	6	5	4	3	2	1	0
Name	SPI_DAT							
Type	R/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	Function
[7:0]	SPI_DAT	SPI send and receive data The SPI_DAT register is used to transmit and receive SPI data. Writing data to SPI_DAT places the data into the send buffer and initiates a transfer when in Master Mode. A read of SPI_DAT returns the contents of the receive buffer.

10 UART

10.1 UART Operating Descriptions

10.1.1 Mode 0

Mode 0 works in shift mode and can used to expand IO port. TXD is clock bus, RXD is data bus and clock frequency is fcpu_clk/12. The data is transmitted from lowest bit to highest bit. When UT_CR[REN] = 0, UART work in send mode. UT_CR[REN] = 1, UART work in receive mode.

To send data, write the data to UT_DR and reset UT_CR[TI]. TXD will output shift pulse and RXD will output data in UT_DR. UT_CR[TI] is set to 1 at the end of transmission.

To receive data, reset UT_CR[RI] and set UT_CR[REN] to 1. TXD will output shift pulse and RXD will receive the data. UT_CR[RI] is set to 1 at the end of receiving data and UT_DR can be read to get the received data.

10.1.2 Mode 1

Mode 1 supports full-duplex and half-duplex modes. TXD is send data bus, RXD is receive data bus. 10bit protocol: 1bit start + 8 data bits (UT_DR) + 1bit stop. The baud rate is configured with frequency division according to UT_BAUD.

To send data, write the data to UT_DR and reset UT_CR[TI]. TXD will output 10 bit data, and UT_CR[TI] is set to 1 at the end of transmission.

To receive data, set UT_CR[REN] to 1 and then reset UT_CR[RI]. RXD receive the data by UART. UTCR[RB8] and UT_CR[RI] are set to 1 at the end of receiving data, and UT_DR can be read to get the received data.

10.1.3 Mode 2

Mode 2 supports full-duplex and half-duplex modes. TXD is send data bus, RXD is receive data bus. 11bit protocol: 1bit start + 9 data bits(UT_DR + UT_CR[RB8]/UT_CR[TB8]) + 1bit stop. The baud rate is configured with frequency division according to UT_BAUD.

To send data, write the data to UT_DR, configure UT_CR[TB8] and reset UT_CR[TI]. TXD will output 11 bit data, and UT_CR[TI] is set to 1 at the end of transmission.

To receive data, set UT_CR[REN] to 1 and then reset UT_CR[RI]. RXD receive the data by UART. UT_CR[RI] are set to 1 at the end of receiving data, and the data is 8 data bits(UT_DR) + 9^{th} bit(UT_CR[RB8]).

10.1.4 Mode 3

The basic operation is the sane as Mode 2, however, the baud rate configuration is the same as Mode 1.

10.1.5 UART Interrupt Source

When UART interrupts are enabled (ES0 = 1), the following two flags will generate an interrupt when they are set to logic 1.

Note: All of the following bits must be cleared by software

1. Send interrupt flag TI will be set to 1 by hardware after UART has completed sending a group of data (8bit for Mode 0 & Mode 1, 9bit for Mode 2 & Mode 3)
2. Receive interrupt flag RI will be set to 1 by hardware after UART has completed receiving a group of data and stop bit.

10.2 UART1 Register

10.2.1 UT_CR (0x98)

Table 10-1 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	Function
[7:6]	MOD	Mode Control 00: Mode 0: Shift register The baud rate is fixed at 2MHz in this mode Baud rate = f_{cpu_clk} / 12= 24MHz / 12 = 2MHz 01: Mode 1:8-bit UART The baud rate is calculated by the follow formula: f_{cpu_clk} / (16 × (1+UT_BAUD[BAUD_SEL]) × (UT_BAUD + 1)) 10: Mode 2: 9-bit UART The baud rate is fixed at 750KHz in this mode Baud rate = f_{cpu_clk} / 32 = 24MHz / 32 = 750KHz 11: Mode3: 9-bit UART The baud rate is calculated by the follow formula: f_{cpu_clk} / (16 × (1 + UT_BAUD[BAUD_SEL]) × (UT_BAUD + 1))
[5]	SM2	0: Does not allow multi-thread cpu operation 1: Allow multi-thread cpu operation
[4]	REN	0: Does not allow serial input operation, 1: Allow serial input operation and cleared by software
[3]	TB8	To set the 9th bit of data transmitting in the Mode 2 and Mode 3. It is cleared by hardware.
[2]	RB8	To set the 9th bit of data receiving in the Mode 2 and Mode 3. It will work as stop bit when SM2 is 0. This bit cannot be used in the mode 0

[1]	TI	Send finished interrupt flag. It will be set to 1 by hardware after completing the data transfer. It is cleared by software.
[0]	RI	Receive finished interrupt flag, it will be set to 1 by hardware after data has been finished to receive. It is be cleared by software.

10.2.2 UT_DR (0x99)

Table 10-2 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	Function
[7:0]	UT_DR	Send/receive data

10.2.3 UT_BAUD (0x9A, 0x9B)

Table 10-3 UT_BAUDH (0x9B)

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

Table 10-4 UT_BAUDL (0x9A)

Bit	7	6	5	4	3	2	1	0
Name	UT_BAUDL							
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	Function
[11:0]	UT_BAUD	Setting of baud rate in the Mode 1 & Mode 3

10.3 UART2 Register

10.3.1 UT2_CR (0x8A)

Table 10-5 UT2_CR (0x8A)

Bit	7:6	5	4	3	2	1	0
Name	MOD	SM2	REN	TB8	RB8	UT2_TI	UT2_RI
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	Function

[7:6]	MOD	Mode Control 00: Mode 0: Shift register The baud rate is fixed at 2MHz in this mode Baud rate = fcpu_clk / 12= 24MHz / 12 = 2MHz 01: Mode 1:8-bit UART The baud rate is calculated by the follow formula: fcpu_clk / (16 × (1+UT_BAUD[BAUD_SEL]) × (UT_BAUD + 1)) 10: Mode 2: 9-bit UART The baud rate is fixed at 750KHz in this mode Baud rate = fcpu_clk / 32 = 24MHz / 32 = 750KHz 11: Mode3: 9-bit UART The baud rate is calculated by the follow formula: fcpu_clk / (16 × (1 + UT_BAUD[BAUD_SEL]) × (UT_BAUD + 1))
[5]	SM2	0: Does not allow multi-thread cpu operation 1: Allow multi-thread cpu operation
[4]	REN	0: Does not allow serial input operation, 1: Allow serial input operation and cleared by software
[3]	TB8	To set the 9th bit of data transmitting in the Mode 2 and Mode 3. It is cleared by hardware.
[2]	RB8	To set the 9th bit of data receiving in the Mode 2 and Mode 3. It will work as stop bit when SM2 is 0. This bit cannot be used in the mode 0
[1]	UT2_TI	Send finished interrupt flag. It will be set to 1 by hardware after completing the data transfer. It is cleared by software.
[0]	UT2_RI	Receive finished interrupt flag, it will be set to 1 by hardware after data has been finished to receive. It is be cleared by software.

10.3.2 UT2_DR (0x89)

Table 10-6 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	Function
[7:0]	UT2_DR	Send/receive data

10.3.3 UT2_BAUD (0x4042, 0x4043)

Table 10-7 UT2_BAUDH (0x4042)

Bit	7	6	5	4	3：0
Name	BAUD2_SEL	UART2CH	UART2IEN	RSV	UT2_BAUDH
Type	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0

Table 10-8 UT2_BAUDL (0x4043)

Bit	7	6	5	4	3	2	1	0
Name	UT2_BAUDL							
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	Function
[15]	BAUD2_SEL	UART2 baud rate selection
[14]	UART2CH	UART2 function transfer enable 0: UART2 function transfer is disabled, P3.6 is RXD of UART2, P3.7 is TXD of UART2 1: UART2 port function transfer is enabled, P0.1 is RXD of UART2, P0.0 is TXD of UART2
[13]	UART2IEN	UART2 interrupt enable 0: Disable 1: Enable
[12]	RSV	
[11:0]	UT2_BAUD	Setting of baud rate in the Mode 1 & Mode 3

11 MDU

11.1 Introduction

Multiply and Division Unit (MDU) is a built-in calculation co-process unit. In addition to multiplication and division, trigonometric function operation and Low-Pass Filter (LPF) operation are embedded. MDU module can be used in interrupt program or the main program

11.2 Features

MDU has the following features:

- Supports multi-interrupt application
- Fast calculation by Hardware
- Following calculation modes available
  - 16 bit signed multiplication
  - 16 bit signed multiplication (left shift 1 bit)
  - 16 bit unsigned multiplication
  - 32 bit / 16 bit unsigned division
  - LPF
  - Coordinate transformation
  - Arc tangent function

11.3 Function

11.3.1 Operation Description

Process of MDU:

1. MDU_CR[MDURUN] set to 1 to initiate MUD.
2. Configure MUD_MD register to select the MDU operation mode.
3. Write data to MDU_A, MDU_B, MDU_C, and MDU_D. The operation will initiate when data is written to MDU_C[7:0].
4. Calculation is finished when MDU_CR[MDUBUSY] is set to 0.
5. Set MDU_CR[MDUDONE] to 1

Notes:

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

11.3.2 16 Bit Signed Multiplication (Shift Left 1 Bit)

MDU works in 16 bit signed multiplication (shift left 1 bit) mode when MDU_MD[2:0] = 000. As shown in Table 11-1, MDU_A and MDU_C, which are 16 bit signed data, are written as multiplicand and multiplier. The result is a 31 bit signed data and will be shift 1 bit left to become a 32 bit signed bit. Software can read MDU_A to return high 16 bits of the result and MDU_B to return low 16 bits.

Table 11-1 Definition of Register in 16 Bit Signed Multiplication (Shift Left 1 Bit) Mode

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

11.3.3 16 Bit Signed Multiplication

MDU works in 16 bit signed multiplication mode when MDU_MD[2:0] = 001. As shown in Table 11-2, MDU_A and MDU_C, which are 16 bit signed data, are written as multiplicand and multiplier. The result is a 31 bit signed data. Software can read MDU_A to return high 16 bits of the result and MDU_B to return low 16 bits.

Table 11-2 Definition of Register in 16 Bit Signed Multiplication Mode

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

11.3.4 16 Bit Unsigned Multiplication

MDU works in 16 bit unsigned multiplication mode when MDU_MD[2:0] = 010. As shown in Table 11-3, MDU_A and MDU_C, which are 16 bit unsigned data, are written as multiplicand and multiplier. The result is a 32 bit unsigned data. Software can read MDU_A to return high 16 bits of the result and MDU_B to return low 16 bits.

Table 11-3 Definition of Register in 16 Bit Unsigned Multiplication Mode

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

11.3.5 32 Bit / 16 Bit Unsigned Division

MDU works in 32 bit / 16 bit unsigned division mode when MDU_MD[2:0] = 011. As shown in Table 11-4, MDU_A, MDU_B and MDU_C, which are 16 bit unsigned data, are written as high, low 16 bits of dividend and divisor. The result is a 32 bit unsigned quotient and a 16 bit unsigned remainder. Software can read MDU_A to return high 16 bits of the quotient and MDU_B to return low 16 bits. The remainder of result is read by MDU_C.

Table 11-4 Definition of Register in 32 bit / 16 Bit Unsigned Division Mode

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

MDU_D	--	--

11.3.6 Low-Pass Filter (LPF)

MDU works in LPF mode when MDU_MD[2:0] = 110.

LPF is calculated as

$$Y_K = Y_{K-1} + K \times (X_X - Y_{K-1})$$

As shown in Table 11-5, MDE_A, MDU_B, MDU_C and MDU_D are written as input X_X, 16 bit signed data, high and low 16 bits of last period output Y_{K-1}, a 32 bit signed data and K, 8 bit unsigned data. Software can read MDU_B to return high 16 bits of the result Y_K and MDU_C to return low 16 bits.

Table 11-5 Definition of Register in LPF Mode

Data Register	Input	Output
MDU_A	X_X	--
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

MDU works in coordinate transformation mode when MDU_MD[2:0] = 100.

Coordinate transformation are calculated as

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

As shown in Table 11-6, MDU_A, MDU_B and MDU_C, which are 16 bit signed data, are written as $\cos_i$, θ and $\sin_i$. Software can read MUD_A, 16 bit signed data, to return $\cos_o$ and MUD_C to return $\sin_o$.

Table 11-6 Meaning of registers in 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 Acr Tangent Function

MDU works in arc tangent function mode when MDU_MD[2:0] = 101.

Based on the input of sin and cos, arc tangent function is calculated as

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

As shown in Table 11-7, MDU_A and MDU_C, which are 16 bit signed data, are written as cos and sin. Software can read MUD_A, 16 bit signed data, to return *U* and MUD_C to return *θ*.

Table 11-7 Meaning of registers in Acr Tangent Function Mode

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

11.4 Register

11.4.1 MDU_CR (0xC1)

Table 11-8 MDU_CR (0xC1)

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

Bit	Name	Function
[7]	MDUBUSY	MDU busy flag When MDU_C[7:0] is written a data, MDU start to work and this bit is set to 1 until the process is completed.
[6]	MDUDONE	MDU ending flag Set this bit to 1 at the end of MDU process. To avoid calculation error, the step above is necessary when MDU is used in the main program and interrupt program.
[5]	MDURUN	MDU starting flag Set this bit to 1 before MDU configuration. To avoid calculation error, the step above is necessary when MDU is used in the main program and interrupt program.
[4:0]	RSV	Reserved

11.4.2 MDU_MD (0xCA)

Table 11-9 MDU_MD (0xCA)

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

Reset	0	0	0	0

Bit	Name	Function
[7:3]	RSV	
[2:0]	MDUMOD[2:0]	MDU mode selection 000: 16 bit signed multiplication (shift left 1 bit) 001: 16 bit signed multiplication 010: 16 bit unsigned multiplication 011: 32 bit / 16 bit unsigned division 100: Coordinate Transformation 101: Arc tangent function 110: LPF 111: RSV

11.4.3 MDU_A (0xC7, 0xC6)

Table 11-10 MDU_AH (0xC7)

Bit	7	6	5	4	3	2	1	0
Name	MDU_A[15:8]							
Type	R/W							
Reset	0							

Table 11-11 MDU_AL (0xC6)

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

Bit	Name	Function
[15:0]	MDU_A[15:0]	A data register of MDU, the content of this register is shown in the following table.

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

11.4.4 MDU_B (0xC5, 0xC4)

Table 11-12 MDU_BH (0xC5)

Bit	7	6	5	4	3	2	1	0
Name	MDU_B[15:8]							
Type	R/W							
Reset	0							

Table 11-13 MDU_BL (0xC4)

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

Bit	Name	Function
[15:0]	MDU_B[15:0]	B data register of MDU, the content of this register is shown in the following table.

MDU_MD[2:0]	Input	Output
000	--	Low 16 bits of the product
001	--	Low 16 bits of the product
010	--	Low 16 bits of the product
011	Low 16 bits of the dividend	Low 16 bits of the quotient
100	θ	--
101	--	--
110	Y_{K-1}[31:16]	Y_K[31:16]

11.4.5 MDU_C (0xC3, 0xC2)

Table 11-14 MDU_CH (0xC3)

Bit	7	6	5	4	3	2	1	0
Name	MDU_C[15:8]							
Type	R/W							
Reset	0							

Table 11-15 MDU_CL (0xC2)

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

Bit	Name	Function
[15:0]	MDU_C[15:0]	C data register of MDU. The operation will initiate when data is written to MDU_C[7:0]. The content of this register is shown in the following table.

		MDU_MD[2:0]	Input	Output
		000	Multiplier	--
		001	Multiplier	--
		010	Multiplier	--
		011	Divisor	Remainder
		100	$\sin_i$	$\sin_o$
		101	sin	θ
		110	Y_{K-1}[15:0]	Y_K[15:0]

11.4.6 MDU_D (0xCB)

Table 11-16 MDU_D (0xCB)

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

Bit	Name	Function
[15:0]	MDU_D[15:0]	The data register of MDU. Input *K* in coordinate transformation mode.

12 PFC

12.1 PFC Operating Descriptions

12.1.1 Introduction

Power Factor Correction (PFC) can improve the efficiency of supply, with high quality, and alleviate the problems of electromagnetic compatibility and electromagnetic interference.

Features:

1. Process by hardware
2. Automatic ADC sampling
3. Overcurrent protection and cycle-by-cycle current limiting

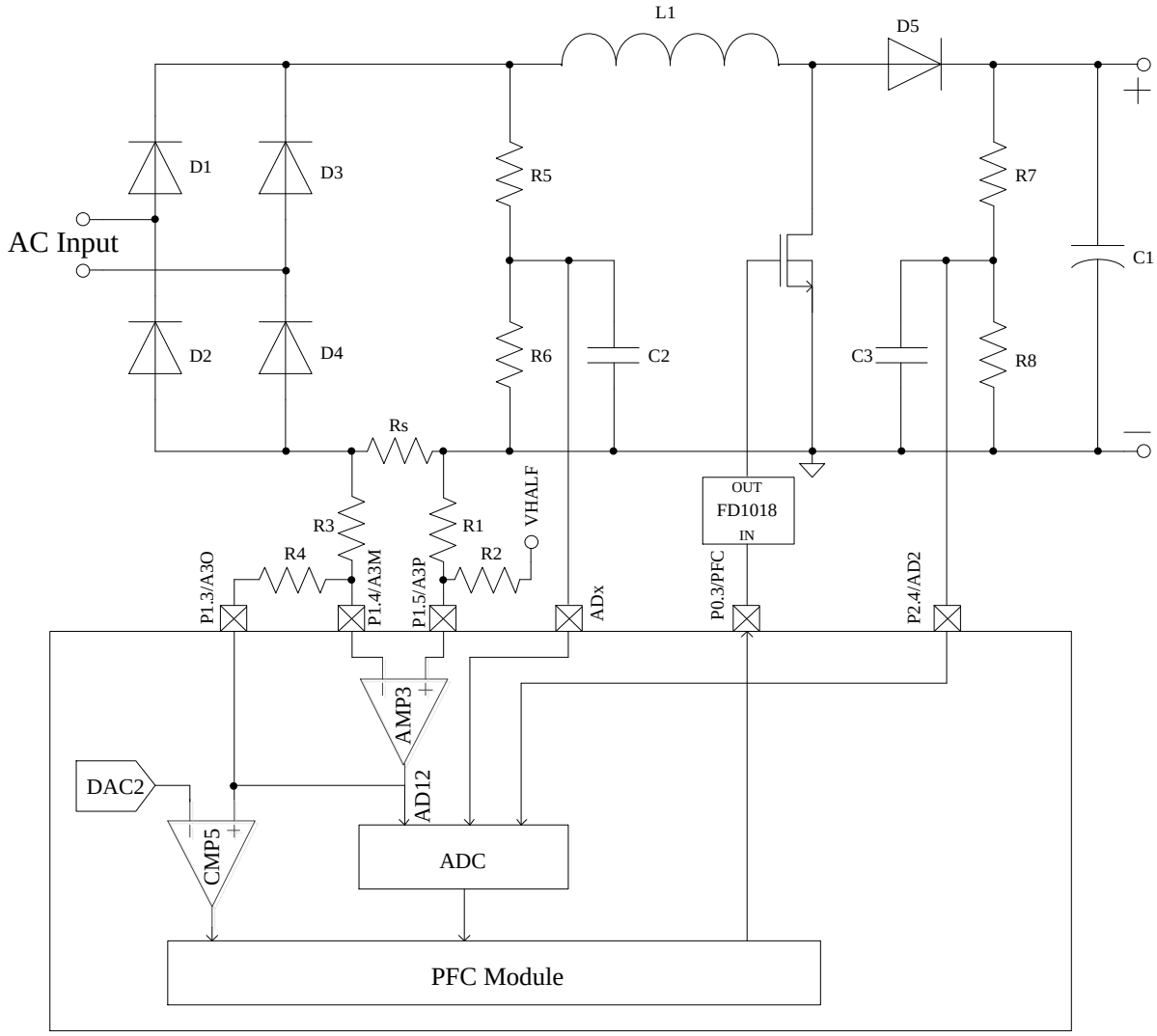


Figure 12-1 Structure Diagram of PFC Module

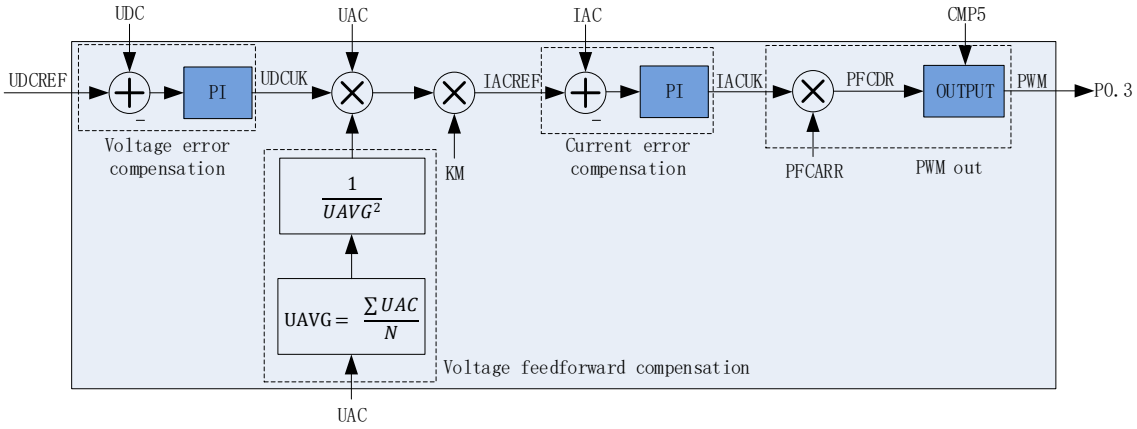


Figure 12-2 Block Diagram of PFC Module

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 the outer loop of the PFC module. Input is bus voltage reference UDCREF and the error with sampling bus voltage, and output is a control signal UDCUK used to calculate inner loop reference IACREF.

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 used to keep supply power stable under uncertain input AC voltage.

12.1.3.1 Calculation of Average Voltage

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

UAVG is calculated as

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

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

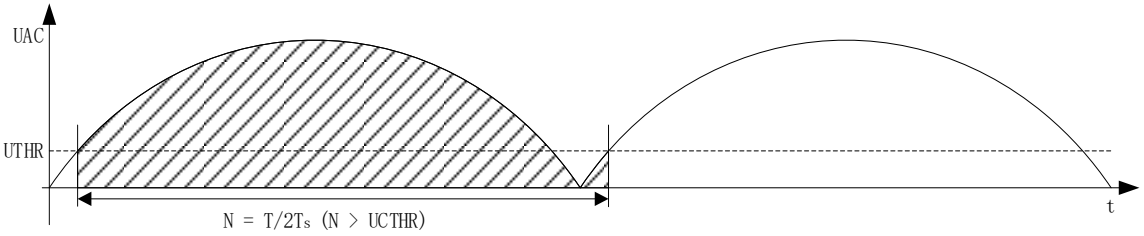


Figure 12-3 Calculation of UAVG

Set PFC_CR0[UACGCDIS] to 0 to configure 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.

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

12.1.4 Current Error Compensation Module

Current error compensation module is the inner loop of the PFC module. Input is current reference IACREF and error with the sampling current, and output is the duty cycle of PWM module IACUK.

Inner loop frequency = 24MHz / PFC_ARR

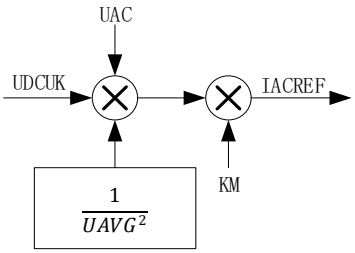


Figure 12-4 Block Diagram of the Calculation of IACREG

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

12.1.5 PWM Output Module

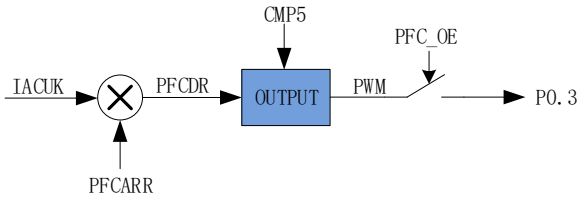


Figure 12-5 Block Diagram of PWM Output

IACUK is the duty cycle of PWM module, duty cycle = 100% × IACUK / 32768, and PWM signal is generated by comparator and GPIO. As shown above, PFCDR, the value is IACUK × PFCARR / 32768, is compared with PFC counter to generate PWM signal. Set PFC_OE to enable P0.3 as the Output the PWM module. When PFCDR > PFCCBTR, P0.3 output logical 1. When PFCDR < PFCCBTR, P0.3 output logical 0.

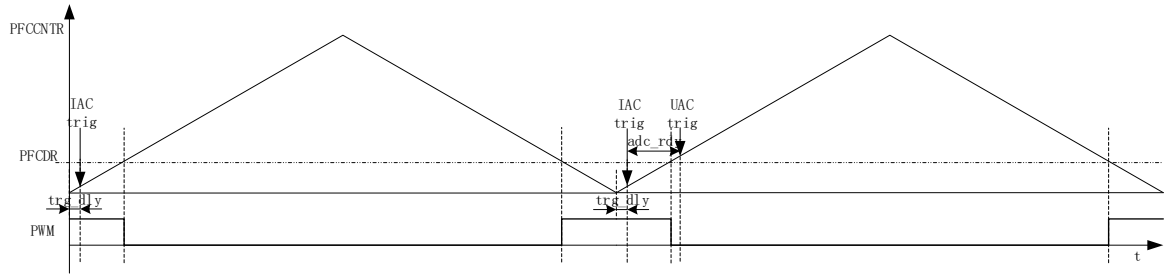


Figure 12-6 Diagram of PWM Output and IAC/UAC Sampling

12.1.5.1 Overcurrent Protection and Cycle-by-Cycle Current Limiting

Set CMP_CR4[CMP5EN] to 1 to enable overcurrent protection function. The filter factor of CMP5 is configured by PFC_CR0[CPM5DIV]. When the input of CMP5 is logical 1, overcurrent protection forces the Output PWM module to low voltage level. The overcurrent protection can be disabled by set PFC_CR0[PFC_OE] to 0.

12.1.6 UAC/IAC/UDC Sampling

12.1.6.1 UDC Sampling

- UDC is sampled by FOC module every carrier period.
- Using ADC Channel-2

12.1.6.2 IAC Sampling

- Sample once on the underflow point of every inner loop period

- The sampling time of every period is configurable by PFC_TRGDLY. According to the MCU clock, 41.67ns, if PFC_TRGDLY = 5, the sampling time is delayed for 41.67ns × 2 × 5 = 416.7ns.
- Using ADC Channel-6
- Set PFC_CR0[CCHSEL] to 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)

12.1.6.3 UAC Sampling

- Set the value of PFC_CR1[UACSAMSEL] to configure the sampling period, once every 1/2/4/8 inner loop periods. Sample after sampling of IAC.
- ADC Channel-5 is used by default. Set the value of UAC_TRIG_CH to select other ADC channels.
- Set PFC_CR0[CCHSEL] to 0, 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 Register

12.2.1 PFC_CR2 (0x409E)

Table 12-1 PFC_CR2 (0x409E)

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

Bit	Name	Function
[7]	UDCPISTA	Start UDC PI controller When PFC is not enabled (PFCEN = 0), the UDC PI controller can be used as a general PI controller. Set by software to logical 1. It will be cleared by hardware at next clock. It is invalid to write 0 to this bit. 0: No start 1: Start
[6]	IACPISTA	Start IAC PI controller When PFC is not enabled (PFCEN = 0), the IAC PI controller can be used as a general PI controller. To be set by software to logic 1. It will be cleared by hardware at next clock. It is invalid to write 0 to this bit. 0: No start 1: Start
[5]	RSV	Reserved

[4]	RSV	Reserved
[3]	RSV	Reserved
[2]	RSV	Reserved
[1]	RSV	Reserved
[0]	RSV	Reserved

12.2.2 PFC_CR0 (0x40E0)

Table 12-2 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	Function
[7]	UAVGSW	Start UAVG calculation Start UAVG calculation and update the result to the value of UAVG. Set by software to logical 1. It will be cleared by hardware at next clock. It is invalid to write 0 to this bit. 0: No start 1: Start
[6:5]	CMP5DIV	Filter period selection of comparator 5 The input will be ignored if the width of it is less than the configured value. When the input pulse width of comparator 5 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: 8 system clocks 11: 16 system clocks
[4]	UAVGCDIS	Calculation of UAVG by hardware disable When UAVGCDIS is set to 1, set UAVGSW to 1 to initiate the calculation of UAVG by software. 0: Enable 1: Disable
[3]	PFCOA	Cycle-by-cycle current limiting enable The overcurrent protection is enabled by default when comparator 5 is enabled. Set PFCOA to 1, the PWM output will be enabled when current is below the limit. 0: Disable 1: Enable
[2]	CCHSEL	ADC offset channel selection Select the ADC offset channel, IAC or UAC. 0: IAC_TRIG_CH corresponding to IAC

		1: UAC_TRIG_CH corresponding to UAC
[1]	PFCOE	PFC output enable Enable P0.3 as the PWM Output PFC. 0: Disable 1: Enable
[0]	PFCEN	PFC enable 0: Disable 1: Enable

12.2.3 PFC_CR1 (0x40F2)

Table 12-3 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	Function
[7:6]	UACSAMSEL	UAC sampling frequency division Sample UAC once every x PFC periods. 00: 1 PFC periods 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, will be set as the start or end of a calculation period. The threshold is based on UAC_BASE. 1: 1 / 8 × UAC_BASE 0: 1 / 16 × UAC_BASE
[4:0]	UCTHR	The 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 is enabled (PFCEN = 1).

12.2.4 PFC_ADCCH (0x40E1)

Table 12-4 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

<table>
<tr><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>[7:4]</td><td>IAC_TRIG_CH</td><td>ADC channel selection for sampling IAC
Choose ADC Channel-6 when PFC is enabled, otherwise the Op Amp is unavailable.
<table>
<tr><td>0000</td><td>Channel-0</td><td>0001</td><td>Channel-1</td></tr>
<tr><td>0010</td><td>Channel-2</td><td>0011</td><td>Channel-3</td></tr>
<tr><td>0100</td><td>Channel-4</td><td>0101</td><td>Channel-5</td></tr>
<tr><td>0110</td><td>Channel-6</td><td>0111</td><td>Channel-7</td></tr>
<tr><td>1000</td><td>Channel-8</td><td>1001</td><td>Channel-9</td></tr>
<tr><td>1010</td><td>Channel-10</td><td>1011</td><td>Channel-11</td></tr>
<tr><td>1100</td><td>Channel-12</td><td>1101</td><td>Channel-13</td></tr>
<tr><td>1110</td><td>RSV</td><td>1111</td><td>RSV</td></tr>
</table></td></tr>
<tr><td>[3:0]</td><td>UAC_TRIG_CH</td><td>ADC channel selection for sampling UAC
<table>
<tr><td>0000</td><td>Channel-0</td><td>0001</td><td>Channel-1</td></tr>
<tr><td>0010</td><td>Channel-2</td><td>0011</td><td>Channel-3</td></tr>
<tr><td>0100</td><td>Channel-4</td><td>0101</td><td>Channel-5</td></tr>
<tr><td>0110</td><td>Channel-6</td><td>0111</td><td>Channel-7</td></tr>
<tr><td>1000</td><td>Channel-8</td><td>1001</td><td>Channel-9</td></tr>
<tr><td>1010</td><td>Channel-10</td><td>1011</td><td>Channel-11</td></tr>
<tr><td>1100</td><td>Channel-12</td><td>1101</td><td>Channel-13</td></tr>
<tr><td>1110</td><td>RSV</td><td>1111</td><td>RSV</td></tr>
</table></td></tr>
</table>

12.2.5 PFC_CSO (0x40E2, 0x40E3)

Table 12-5 PFC_CSOH (0x40E2)

Bit	7	6	5	4	3	2	1	0
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

Table 12-6 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	Function
[15:0]	PFC_CSO	Current sampling reference Configure the PFC_CR0[CCHSEL] to set IAC/UAC ADC offset. Range: (0,32767)
Providing the ADC voltage range is 0~5V, the offset is 2.5V PFC_CSO = 32768 × 2.5V / 5V = 16384 (0x4000)		

12.2.6 PFC_ARR (0x40E4, 0x40E5)

Table 12-7 PFC_ARRH (0x40E4)

Bit	7	6	5	4	3	2	1	0
Name	PFC_ARR[11:8]							
Type	-	-	-	-	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Table 12-8 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	Function
[12:0]	PFC_ARR	The reload value of the PFC counter, which configures the carrier period and operation period When the PFC counter count from 0 to PFC_ARR, it generates an overflow event. And then count down to 0. This register is unreadable. Range: (0,4095)

12.2.7 PFC__UAVG (0x40E4, 0x40E5)

Table 12-9 PFC__UAVGH (0x40E4)

Bit	7	6	5	4	3	2	1	0
Name	PFC__UAVG [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 12-10 PFC__UAVG L (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	Function
[15:0]	PFC__UAVG	The calculation value of average UAC Range: (-32768, 32767)

12.2.8 PFC__DR (0x40E6, 0x40E7)

Table 12-11 PFC__DRH (0x40E6)

Bit	7	6	5	4	3	2	1	0
Name	PFC__DR[11:8]							
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 12-12 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	Function
[11:0]	PFC__DR	The value in the PFC__DR register is continuously compared to the PFC counter. When the value of the PFC counter is less than PFC__DR, it output 1, and PFC counter is more than PFC__DR, it output 0. PFC__DR will be updated by hardware when PFC is enabled. Range: (0,4095)

12.2.9 UDC_REF/UDC_EK (0x40E8, 0x40E9)

Table 12-13 UDC_REFH/UDC_EKH (0x40E8)

Bit	7	6	5	4	3	2	1	0
Name	UDC_REF/UDC_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

Table 12-14 UDC_REFL/UDC_EKL (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	Function
[15:0]	UDC_REF/UDC_EK	This register is the UDC reference when PFC is enabled This register is the value of EK when PFC is disabled Range: (-32768, 32767)

12.2.10 UDC_UK (0x40EA, 0x40EB)

Table 12-15 UDC_UKH (0x40EA)

Bit	7	6	5	4	3	2	1	0
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

Table 12-16 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	Function
[15:0]	UDC_UK	PI controller output value UK Range: (-32768, 32767)

12.2.11 UDC_KP (0x40EC, 0x40ED)

Table 12-17 UDC_KPH (0x40EC)

Bit	7	6	5	4	3	2	1	0
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

Table 12-18 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	Function
[15:0]	UDC_KP	KP factor of UDC PI controller Range: (0, 32767). The data format is Q10.

12.2.12 UDC_KI (0x40EE,0x40EF)

Table 12-19 UDC_KIH (0x40EE)

Bit	7	6	5	4	3	2	1	0
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

Table 12-20 UDC_KIL (0x40EF)

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	Function
[15:0]	UDC_KI	KI factor of UDC PI controller Range: (0, 32767). The data format is Q15.

12.2.13 UDC_UKMAX (0x40F0, 0x40F1)

Table 12-21 UDC_UKMAXH (0x40F0)

Bit	7	6	5	4	3	2	1	0
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

Table 12-22 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	Function
[15:0]	UDC_UKMAX	The upper limit of UDC PI controller output Range: (-32768, 32767)

12.2.14 UDC_UKMIN (0x40F2, 0x40F3)

Table 12-23 UDC_UKMINH (0x40F2)

Bit	7	6	5	4	3	2	1	0
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

Table 12-24 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	Function
[15:0]	UDC_UKMIN	The register is the lower limit of UDC PI controller output when PFC is disabled. Range: (-32768, 32767) This address is PFC_CR1 and PFC_KM registers when PFC is enabled. UDC_UKMIN is 0 by default.

12.2.15 PFC_KM (0x40F3)

Table 12-25 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	Function
[7:0]	PFC_KM	KM factor of PFC Range: (0,255) Note: Only available when PFC is enabled (PFCEN = 1)

12.2.16 IAC_REF/IAC_EK (0x40F4, 0x40F5)

Table 12-26 IAC_REFH/IAC_EKH (0x40F4)

Bit	7	6	5	4	3	2	1	0
Name	IAC_REF/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

Table 12-27 IAC_REFL/IAC_EKL (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	Function
[15:0]	IAC_REF/IAC_EK	This register is the IAC reference when PFC is enabled This register is the value of EK when PFC is disabled Range: (-32768, 32767)

12.2.17 IAC_UK (0x40F6, 0x40F7)

Table 12-28 IAC_UKH (0x40F6)

Bit	7	6	5	4	3	2	1	0

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

Table 12-29 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	Function
[15:0]	IAC_UK	IAC PI controller output UK Range: (-32768, 32767)

12.2.18 IAC_KP (0x40F8, 0x40F9)

Table 12-30 IAC_KPH (0x40F8)

Bit	7	6	5	4	3	2	1	0
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

Table 12-31 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	Function
[15:0]	IAC_KP	KP factor of IAC PI controller Range: (0, 32767). The data format is Q10

12.2.19 IAC_KI (0x40FA, 0x40FB)

Table 12-32 IAC_KIH (0x40FA)

Bit	7	6	5	4	3	2	1	0
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

Table 12-33 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	Function
[15:0]	IAC_KI	KI factor of IAC PI controller Range: (0, 32767). The data format is Q15.

12.2.20 IAC_UKMAX (0x40FC, 0x40FD)

Table 12-34 IAC_UKMAXH (0x40FC)

Bit	7	6	5	4	3	2	1	0
Name	IAC_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

Table 12-35 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	Function
[15:0]	IAC_UKMAX	The upper limit of IAC PI controller output Range: (-32768, 32767)

12.2.21 IAC_UKMIN (0x40FE, 0x40FF)

Table 12-36 IAC_UKMINH (0x40FE)

Bit	7	6	5	4	3	2	1	0
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

Table 12-37 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	Function
[15:0]	IAC_UKMIN	The register is the lower limit of the IAC PI controller output when

		PFC is disabled. Range: (-32768, 32767) The address is PFC_TRGDLY and PFC_OUTARR registers when PFC is enabled. IAC_UKMIN is 0 by default.

12.2.22 PFC_TRGDLY/PFC_OUTARR (0x40FE, 0x40FF)

Table 12-38 PFC_TRGDLY/PFC_OUTARRH (0x40FE)

Bit	7	6	5	4	3	2	1	0
Name	PFC_TRGDLY				PFC_OUTARR[11:8]			
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

Table 12-39 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	Function
[15:12]	PFC_TRGDLY	The sampling delay of IAC/UDC According to the MCU clock, 41.67ns, if PFC_TRGDLY = 5, the sampling time is delayed for 41.67ns × 2 × 5 = 416.7ns. Note: Only available when PFC is enabled
[11:0]	PFC_OUTARR	The period of PFC outer loop These bits configure the period of PFC outer loop. Period of outer loop = period of inner loop / PFC_OUTARR = 24M / (2 × PFC_ARR × PFC_OUTARR) According to the MCU clock, 41.67ns, if 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: Only available when PFC is enabled

13 PI

13.1 Introductions

PI is a linear controller. The input is the error between the reference value and the sampling value. The output is the sum of proportional module and integral module. It is widely used in the application of motor drive system.

The Output PI is calculated as：

$$U(k) = U(k-1) + Kp \times (E(k) - E(k-1)) + Ki \times E(k) \text{ ----- } (UK\_MIN < U(k) < UK\_MAX)$$

13.2 Features

- A general PI controller
  - Adjustable parameter range
  - Can be used for repeated applications. Nested structure is not supported.
  - 32 bit Operation result PI_UK
  - The calculation is finished when PIBUSY is 0.
- Two dedicated PI controllers. PFC PI controller will turn to be a general PI controller when PFC is not enabled.
  - KP is Q10 data format by default, KI is Q15 data format by default
  - The calculation will be finished in 4 system clocks.

13.3 PI Operating Descriptions

13.3.1 General PI Controller

1. Set PI_CR[PISTA] to 1 to initiate PI controller, PIBUSY is forced to be 1 by hardware. The calculation is finished when PIBUSY is set to 0. A read of PI_UK returns the update of output.
2. The data format of control parameters can be selected as Q12 or Q15. By default, the data format of PI_KP and PI_KI are Q12 and others are Q15.
3. The contents U(k – 1) and E(k – 1) are the previous updates of U(k) and E(k) by default. Writing data to PI_EK1 and PI_UK registers can configure the value of E(k – 1) and U(k – 1).

To using PI controller for repeated applications, the parameters should be saved after calculation for the proper initialization of the next calculation. The parameters in PI controller are the results in the last period by default. The PI controller should be initialized correctly as the following program for the first time.

```
PI_EK1  = X;        // Initialize E(k – 1)
PI_UK   = Y1;       // Initialize the higher 16 bits of U(k – 1)
PI_UKS  = Y2;       // Initialize the lower 16 bits of U(k – 1)
```

13.3.2 Dedicated PI Controller UDC_PI/IAC_PI

1. To using UDC_PI/IAC_PI, PFC should be disabled.
2. Set PFC_CR2[UDCPISTA / IACPISTA] to 1 to initiate the PI controller. The calculation will be completed in 4 MCU clocks, and the result is updated in PI_UK register.
3. The contents U(k – 1) and E(k – 1) are the previous updates of U(k) and E(k) by default. Writing data

to PI_EK1 and PI_UK registers can configure the value of E(k – 1) and U(k – 1).

13.4 PI Register

13.4.1 PI_CR (0xF9)

Table 13-1 PI_ CR (0xF9)

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

Bit	Name	Function
[7]	T2SS	Input mode selection of TIM2 stepper motor mode 0: P10 is the direction, P07 is the pulse count 1: P10 is the pulse count of negative direction, P07 is the pulse count of positive direction
[6:3]	RSV	Reserved
[2]	PIRANGE	PI parameter data format 0: Q12. The range of KP, KI (-32768, 32767) corresponds to the actual value (-8, 8) 1: Q15. The range of KP, KI (-32768, 32767) corresponds to the actual value (-1, 1)
[1]	PISTA/PIBUSY	PI_STA (write only) Start PI controller. To be set by software to logical 1. It will cleared by hardware at next clock. 0: Not start 1: Start PI_BUSY (read only) 0: PI is available 1: PI is busy
[0]	RSV	Reserved

13.4.2 PI_EK (0xEA, 0xEB)

Table 13-2 PI_EKH (0xEB)

Bit	7	6	5	4	3	2	1	0
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

Table 13-3 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	Function
[15:0]	PI_EK	The input error of PI controller E(k) Range: (-32768, 32767)

13.4.3 PI_EK1 (0xE2, 0xE3)

Table 13-4 PI_EK1H (0xE3)

Bit	7	6	5	4	3	2	1	0
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

Table 13-5 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	Function
[15:0]	PI_EK1	The previous input error of PI controller E (k – 1) Range: (-32768, 32767)

13.4.4 PI_UK (0xEC, 0xED)

Table 13-6 PI_UKH (0xED)

Bit	7	6	5	4	3	2	1	0
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

Table 13-7 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	Function

[15:0]	PI_UK	The higher 16 bits of output U(k) U(k) is a 32 bit data and the actual range is (-1,1). Generally, only the higher 16 bits are used. Range: (-32768, 32767)

13.4.5 PI_UKS (0xE4, 0xE5)

Table 13-8 PI_UKSH (0xE5)

Bit	7	6	5	4	3	2	1	0
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

Table 13-9 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	Function
[15:0]	PI_UKS	The lower 16 bits of output U(k) PI_UKS is used to save the U(k) for the repeated applications of PI controller. Range: (-32768, 32767)

13.4.6 PI_KP (0xEE, 0xEF)

Table 13-10 PI_KPH (0xEF)

Bit	7	6	5	4	3	2	1	0
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

Table 13-11 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	Function
[15:0]	PI_KP	KP factor of PI controller Range: (-32768, 32767)

13.4.7 PI_KI (0xF2, 0xF3)

Table 13-12 PI_KIH (0xF3)

Bit	7	6	5	4	3	2	1	0
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

Table 13-13 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	Function
[15:0]	PI_KI	KI factor of PI controller Range: (-32768, 32767)

13.4.8 PI_UKMAX (0xF4, 0xF5)

Table 13-14 PI_UKMAXH (0xF5)

Bit	7	6	5	4	3	2	1	0
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

Table 13-15 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	Function
[15:0]	PI_UKMAX	The upper limit of PI controller output U(k) Range (-32768, 32767)

13.4.9 PI_UKMIN (0xF6, 0xF7)

Table 13-16 PI_UKMINH (0xF7)

Bit	7	6	5	4	3	2	1	0
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

Table 13-17 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	Function
[15:0]	PI_UKMIN	The lower limit of PI controller output U(K) Range: (-32768, 32767)

13.4.10 PFC_CR2 (0x409E)

Table 13-18 PFC_CR2 (0x409E)

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

Bit	Name	Function
[7]	UDCPISTA	Start UDC PI controller When PFC is not enabled (PFCEN = 0), the UDC PI controller can be used as a general PI controller. Set by software to logical 1. It will be cleared by hardware at next clock. It is invalid to write 0 to this bit. 0: Not start 1: Start
[6]	IACPISTA	Start IAC PI controller When PFC is not enabled (PFCEN = 0), the IAC PI controller can be used as a general PI controller. To be set by software to logic 1. It will be cleared by hardware at next clock. It is invalid to write 0 to this bit. 0: Not start 1: Start
[5]	RSV	
[4]	RSV	
[3]	RSV	
[2]	RSV	
[1]	RSV	
[0]	RSV	

13.4.11 UDC_EK (0x40E8, 0x40E9)

Table 13-19 UDC_EKH (0x40E8)

Bit	7	6	5	4	3	2	1	0
Name	UDC_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

Table 13-20 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	Function
[15:0]	UDC_EK	This register is the value of EK when PFC is disabled Range: (-32768, 32767)

13.4.12 UDC_UK (0x40EA, 0x40EB)

Table 13-21 UDC_UKH (0x40EA)

Bit	7	6	5	4	3	2	1	0
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

Table 13-22 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	Function
[15:0]	UDC_UK	UDC PI controller output UK Range: (-32768, 32767)

13.4.13 UDC_KP (0x40EC, 0x40ED)

Table 13-23 UDC_KPH (0x40EC)

Bit	7	6	5	4	3	2	1	0
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

Table 13-24 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	Function
[15:0]	UDC_KP	KP factor of UDC PI controller Range: (0, 32767). The data format is Q10.

13.4.14 UDC_KI (0x40EE, 0x40EF)

Table 13-25 UDC_KIH (0x40EE)

Bit	7	6	5	4	3	2	1	0
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

Table 13-26 UDC_KIL (0x40EF)

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	Function
[15:0]	UDC_KI	KI factor of UDC PI controller Range: (0, 32767). The data format is Q15.

13.4.15 UDC_UKMAX (0x40F0, 0x40F1)

Table 13-27 UDC_UKMAXH (0x40F0)

Bit	7	6	5	4	3	2	1	0
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

Table 13-28 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	Function
[15:0]	UDC_UKMAX	The upper limit of UDC PI controller output Range: (-32768, 32767)

13.4.16 UDC_UKMIN (0x40F2, 0x40F3)

Table 13-29 UDC_UKMINH (0x40F2)

Bit	7	6	5	4	3	2	1	0
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

Table 13-30 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	Function
[15:0]	UDC_UKMIN	The register is the lower limit of UDC PI controller output when PFC is disabled. Range: (-32768, 32767) This address is PFC_CR1 and PFC_KM registers when PFC is enabled. UDC_UKMIN is 0 by default.

13.4.17 IAC_EK (0x40F4, 0x40F5)

Table 13-31 IAC_EKH (0x40F4)

Bit	7	6	5	4	3	2	1	0
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

Table 13-32 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	Function

[15:0]	IAC_EK	This register is the value of EK when PFC is disabled Range: (-32768, 32767)

13.4.18 IAC_UK (0x40F6, 0x40F7)

Table 13-33 IAC_UKH (0x40F6)

Bit	7	6	5	4	3	2	1	0
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

Table 13-34 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	Function
[15:0]	IAC_UK	IAC PI controller output UK Range: (-32768, 32767)

13.4.19 IAC_KP (0x40F8, 0x40F9)

Table 13-35 IAC_KPH (0x40F8)

Bit	7	6	5	4	3	2	1	0
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

Table 13-36 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	Function
[15:0]	IAC_KP	KP factor of IAC PI controller Range: (0, 32767). The data format is Q10

13.4.20 IAC_KI (0x40FA, 0x40FB)

Table 13-37 IAC_KIH (0x40FA)

Bit	7	6	5	4	3	2	1	0

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

Table 13-38 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	Function
[15:0]	IAC_KI	KI factor of IAC PI controller Range: (0, 32767). The data format is Q15.

13.4.21 IAC_UKMAX (0x40FC, 0x40FD)

Table 13-39 IAC_UKMAXH (0x40FC)

Bit	7	6	5	4	3	2	1	0
Name	IAC_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

Table 13-40 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	Function
[15:0]	IAC_UKMAX	The upper limit of IAC PI controller output Range: (-32768, 32767)

13.4.22 IAC_UKMIN (0x40FE, 0x40FF)

Table 13-41 IAC_UKMINH (0x40FE)

Bit	7	6	5	4	3	2	1	0
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

Table 13-42 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	Function
[15:0]	IAC_UKMIN	The register is the lower limit of the IAC PI controller output when PFC is disabled. Range: (-32768, 32767)

14 FOC/SVPWM

14.1 FOC/SVPWM Operating Descriptions

14.1.1 Introductions

FOC/SVPWM module is used in the applications of sensorless FOC, HALL-effected FOC, and HALL-effected SVPWM control. Since SVPWM is included in FOC module, FOC/SVPWM module is called FOC module for short. As an independent module, the FOC clock stops when the FOC module is not working. It must be enabled by setting the DRV_CR[FOC_EN] to 1 before the operation of the FOC module.

FOC module contains an angle module, a PI controller, a coordinate transformation module and an output module. Sensorless FOC control can be realized by using the internal angle estimation module. MCU may also process the HALL signals to realize the FOC control with Hall sensors. The internal in FOC module contains a closed loop current control, which can output six channel PWM signals to drive the motor. The reference value ID and IQ will be provided by the user, and together with the ADC, which samples the current signal, closed loop current control can be implemented.

A) Sensorless FOC: The angle estimation module is used for coordinate transformations. Meanwhile, the angular velocity and back-EMF are provided to realize the velocity closed loop control and the detection of motor starting.

B) HALL-effected FOC (single HALL/dual HALL/triple HALL): FOC module provides the angle input interface. The angle value is obtained by utilizing the HALL signals, with the angle value sent to the FOC module.

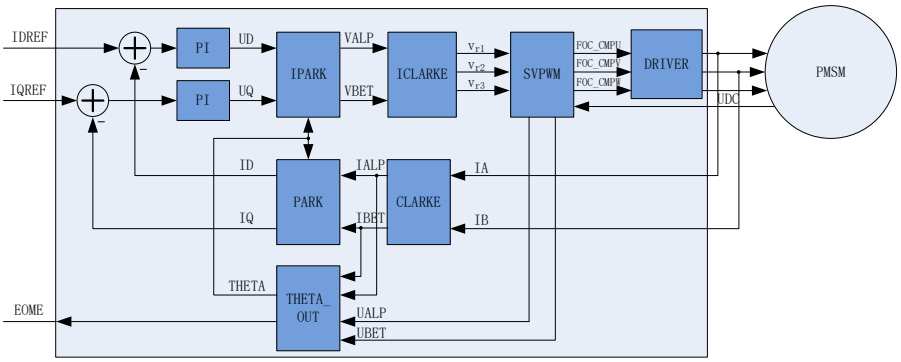


Figure 14-1 Functional Block Diagram of FOC

14.1.2 Reference Input

FOC module has closed loop current control, and d-axis current and q-axis current are adopted as the reference input. If the velocity and current dual closed-loop control is requested, the velocity outer loop control is achieved by using the estimated velocity signal EOME, together with the MCU or PI module.

14.1.3 PI Controller

FOC module contains 4 PI controllers, and they are applied to:

1. Flux control: PI controller of d-axis, reference current IDREF minus feedback current ID as the error input. Proportional coefficient DKP and the integral coefficient DKI for adjustment of PI performance. DMAX and DMIN for limiting of the output amplitude. The output, voltage of d-axis UD, is provided.
2. Torque control: PI controller of q-axis, reference current IQREF minus feedback current IQ as the error input. Proportional coefficient QKP and the integral coefficient QKI for adjustment of PI performance. QMAX and QMIN for limiting of output amplitude. The output, voltage of q-axis UQ, is provided.
3. Angle estimation: PI controller of estimator, proportional coefficient EKP and the integral coefficient EKI for adjustment of PI performance. The estimated angle ETHETA is provided.
4. PLL estimation: PI controller of PLL, proportional coefficient PLLKP and the integral coefficient PLLKI for adjustment of PI performance. The estimated BEMF EALPHA and EBETA are provided.

14.1.4 Coordinate Transformations

14.1.4.1 Inverse PARK Transformation

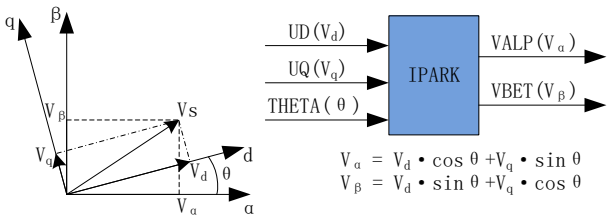


Figure 14-2 Inverse PARK Transformation

After the PI iteration, the two components of voltage vector on the rotating d-q axis can be obtained. From d-q axis, the transforms to α-β axis can thus be obtained with the inverse Park transform.

14.1.4.2 Inverse CLARKE Transformation

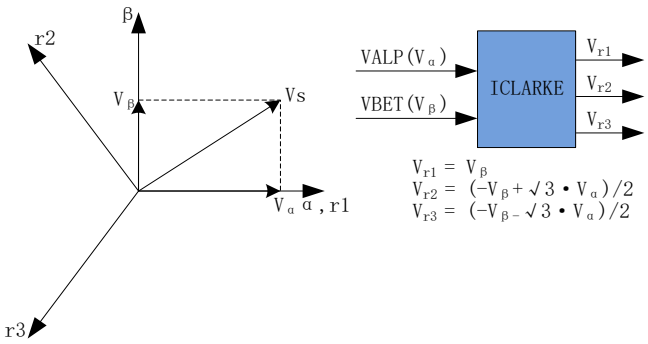


Figure 14-3 Inverse CLARKE Transformation

Inverse Clarke transformation is to transform the voltage vector from the α-β frame to the stationary three-axis, 3-phase reference frame of the stator.

14.1.4.3 CLARKE Transformation

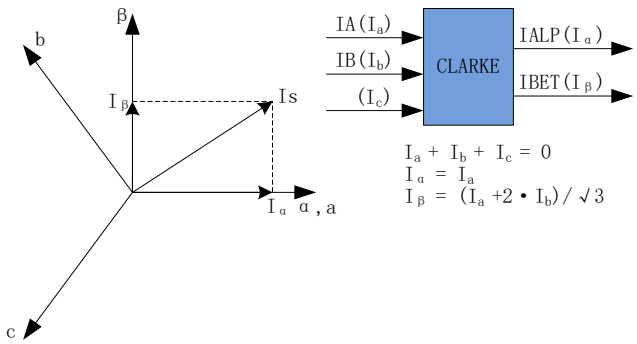


Figure 14-4 CLARKE Transformation

Clarke transformation converts the current components from a three-axis coordinate to a 2D coordinate system referenced to the stator.

14.1.4.4 PARK Transformation

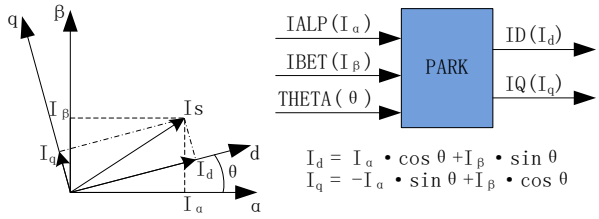


Figure 14-5 PARK Transformation

Park transformation converts the current components from a 2D coordinate system with the axis called α-β, into another 2D system called d-q, which is rotating with the rotor flux.

14.1.5 SVPWM

Space Vector Pulse Width Modulation (SVPWM) algorithm is an important part of FOC. The main idea is to use the switching of inverter space voltage vector to obtain the quasi-circular rotating magnetic field. This method can decrease the harmonic components of the inverter output current, the harmonic losses of the motor and the torque ripple, and have higher voltage utilization.

SVPWM generate pulse-width modulation signals for the 3-phase motor voltage control. The process of generating the pulse width is reduced to a few simple equations. Each of the inverter phase output can be in one of two states. The inverter phase output can be connected to either the plus (+) BUS rail or the minus

(–) BUS rail, which allows for $2^3 = 8$ possible states of the output voltage. There are two states in which all three outputs are connected to either the plus (+) BUS or the minus (–) BUS. These are considered null states because there is no line-to-line voltage across any of the phases. These are plotted at the origin of the SVPWM voltage space. The remaining six states are represented as active vectors with 60 degree rotation between each state.

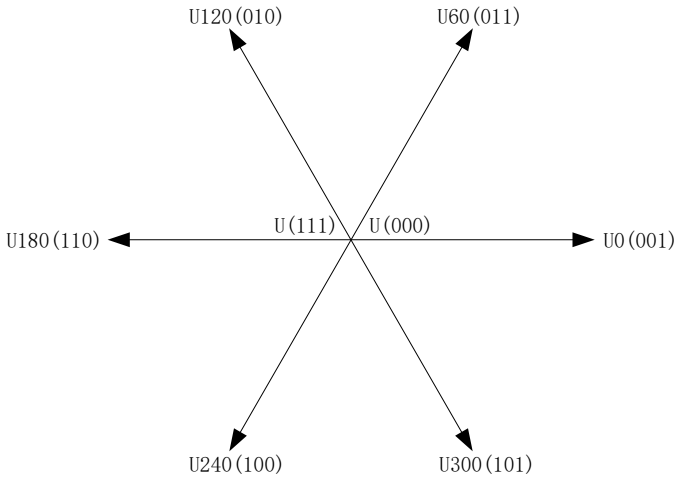


Figure 14-6 SVPWM Voltage Vector Space

SVPWM uses the sum of two adjacent vectors to generate any voltage vector located in the voltage vector space. Supposing that UOUT is the desired vector and it lies in the sector between U60 and U0. UOUT is then represented as an average voltage during a given PWM period T, U0 outputs for 2 × T1 / T and U60 outputs for 2 × T2 / T. T0 represents the time where no active voltage is applied into the windings, that is, where a null vector is applied.

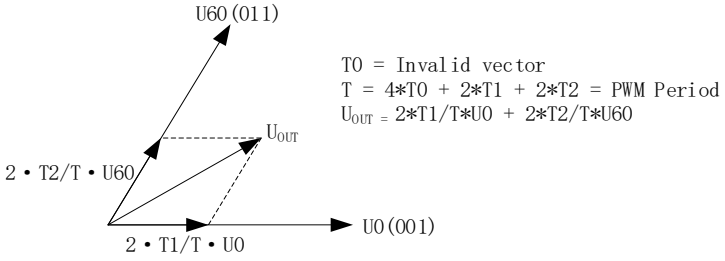


Figure 14-7 Average SVPWM

Table 14-1 Inverter states of SVPWM

Phase C	Phase B	Phase A	V_{ab}	V_{bc}	V_{ca}	V_{ds}	V_{qs}	Vector
0	0	0	0	0	0	0	0	U(000)
0	0	1	V_{DC}	0	$-V_{DC}$	$2/3V_{DC}$	0	U0
0	1	1	0	V_{DC}	$-V_{DC}$	$1/3V_{DC}$	$1/3V_{DC}$	U60

0	1	0	$-V_{DC}$	V_{DC}	0	$-1/3V_{DC}$	$1/3V_{DC}$	U120
1	1	0	$-V_{DC}$	0	V_{DC}	$-2/3V_{DC}$	0	U180
1	0	0	0	$-V_{DC}$	V_{DC}	$-1/3V_{DC}$	$-1/3V_{DC}$	U240
1	0	1	V_{DC}	$-V_{DC}$	0	$1/3V_{DC}$	$-1/3V_{DC}$	U300
1	1	1	0	0	0	0	0	U(111)

14.1.5.1 Seven-Segment SVPWM

In the single-resistance current sampling mode, seven-segment SVPWM is always used. In the dual-resistances current sampling mode, user can set FOC_CR2[F5SEG] to 0 select the seven-segment SVPWM as the output mode.

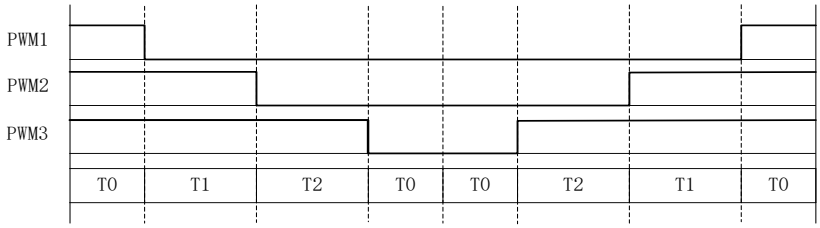


Figure 14-8 Seven-Segment SVPWM Output Voltage Level

14.1.5.2 Five-Segment SVPWM

Five-segment SVPWM is only used in the dual-resistances current sampling mode. User is required to set FOC_CR2[F5SEG] to 1.

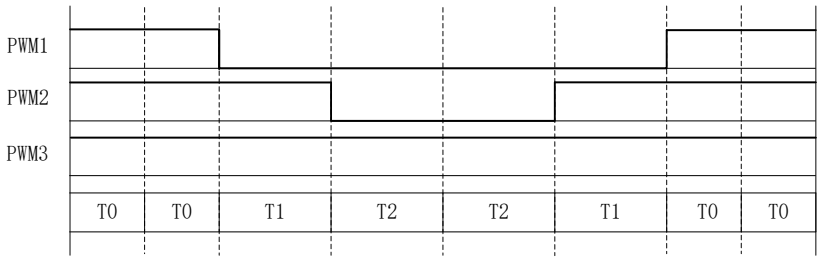


Figure 14-9 Five-Segment SVPWM Output Voltage Level

14.1.6 Overmodulation

Overmodulation can be used in single/dual/triple-resistance current sampling mode. Overmodulation is enabled when FOC_CR1[OVMDL] is set to 1. The output (i.e., FOC_DMAX and FOC_QMIN) will be multiplied by 1.15 in this mode.

14.1.7 Deadtime Compensation

Deadtime compensation is only available in dual/triple-resistances mode. The compensation value of

deadtime is configured by FOC_TSMIN. This mode can improve the quality of phase current at low speed.

14.1.8 Current and Voltage Sampling

FOC module needs to sample the BUS voltage of the inverter and phase current. Before using the FOC module, users should enable ADC (set ADC_STA[ADCEN] to 1), Op Amp, and related registers must be configured. However, there is no need to configure ADC channels and scanning mode. FOC_CR1[CSM] can be set to select single/dual/triple-resistances current sampling mode. In single-resistance current sampling mode, the default sampling channel of the BUS current is Channel-4. In dual-resistance mode, Channel-0 is the default sampling channel for current ia and Channel-1 is for ib. In triple-resistance mode, Channel-0, Channel-1 and Channel-2 are the default sampling channels for phase current ia, ib and ic. The default sampling BUS voltage is Channel-2.

14.1.8.1 Single-Resistance Sampling Mode

Set FOC_CR1[CSM] to 0 to select the single-resistance current sampling mode. Using this mode, FOC module samples the BUS current itrip (Channel-4) twice during the counter with a count-up operation, and samples the BUS voltage when FOC module finishing the operation during the counter with a count-down operation.

Since deadtime affects the sampling time of current sampling, FOC module will adjust the sampling time based on deadtime automatically to ensure the accuracy. Moreover, FOC_TRGDLY is used to adjust sampling time by software. According to the MCU clock, 41.67ns, if FOC_TRGDLY = 5, the sampling time is delayed for 41.67ns × 5 = 208.3ns. If FOC_TRGDLY = 0xFB (-5), the current is sampled 208.3ns in advance.

The time of sampling window should be large enough to sample the current. Users can set TS to configure the lower limit time of T1 and T2, and FOC module will adjust the value of T1 and T2 automatically

14.1.8.2 Dual/triple Resistances Sampling Mode

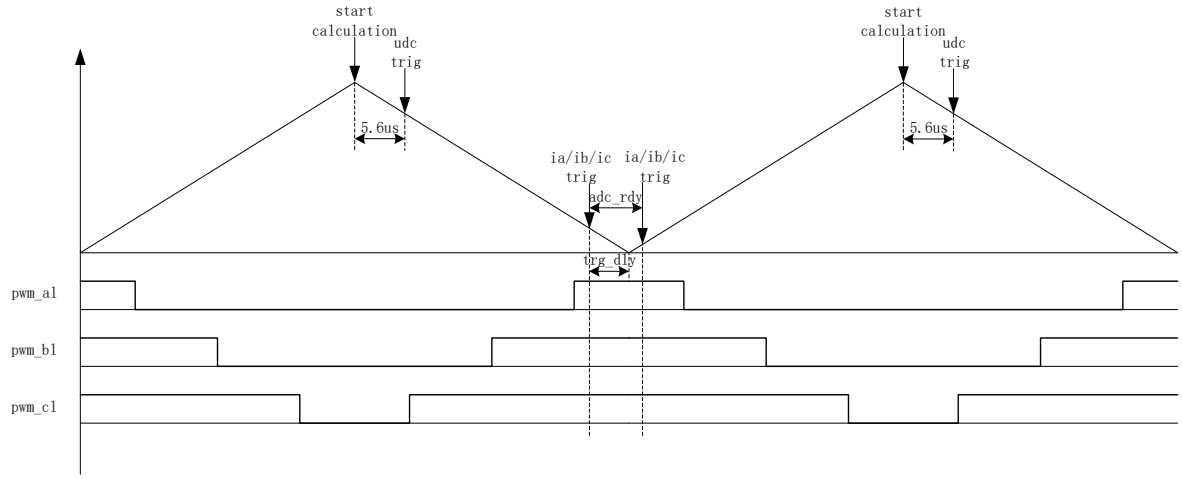


Figure 14-10 Dual/Triple Resistances Concurrent Sampling Mode

Set FOC_CR1[CSM] to 10/11 and FOC_CR2[DSS] to 0, to select dual/triple resistances concurrent current sampling mode. In this mode, the sampling time of a phase current (decided by the voltage reference) can be set by TRG_DLY register, and the other phase is sampled at the end of the periods sampling. The BUS voltage can be sampled after FOC module has finished the operation during a counter count-down operation. Attention should be paid such that the current sampling time is set to make the sampling points of ia, ib and ic all in the state U(000). According to the MCU clock, 41.67ns, if FOC_TRGDLY = 0xB2, the sampling time is 41.67*50=2.0835μs ahead the minima of the counter during the counter of the FOC with a count-down operation. The other phase current ia/ib/ic should be sampled after completing the previous sampling.

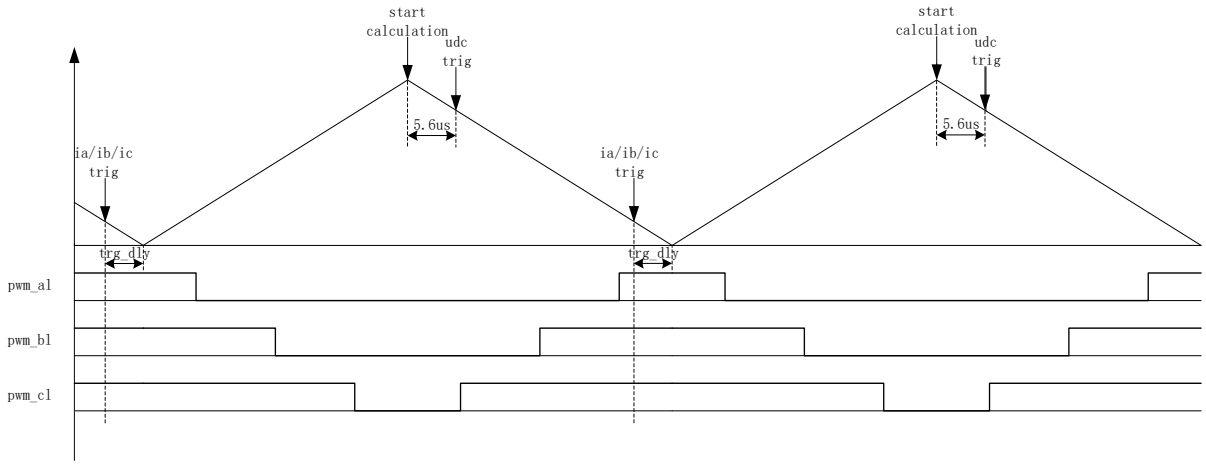


Figure 14-11 Dual/Triple Resistances Alternating Sampling Mode

Set FOC_CR1[CSM] to 10/11 and FOC_CR2[DSS] to 1, to select dual/triple resistances alternating current sampling mode. In this mode, only one phase current is sampled at each carrier period (i.e., a phase

current is sampled in the first carrier period and the other in the second period). At the end of the FOC module operation, BUS voltage will be sampled. The sampling time can be set by TRG_DLY register. Attention should be paid such that the current sampling time is set to make the sampling points of ia, ib and ic all in the state U(000). According to the MCU clock, 41.67ns, if FOC_TRGDLY = 0xB2, the sampling time is 41.67*50=2.0835μs ahead the minima of the counter during the counter of the FOC with a count-down operation.

14.1.8.3 Current Sampling Offset

The input current sampling range should be doubled before the sampling due to the existence of the positive and negative phase current. So it is required to minus an offset during the operation, which the default value is 0x4000. However, there is a deviation between the default value and the real value, owing to ADC and hardware implementation. Therefore, it is necessary to calibrate an offset. The calibration procedure is as the following: When FOC module doesn't work and there is no current in three phases, MCU starts to sample the corresponding channel and do a write to the FOC_CSO register after averaging all the sampled value. If the voltage range is 0~5V and the offset voltage is 2.5V, FOC_CSO = 32768 × 2.5V / 5V = 16384(0x4000).

1. When FOC_CHC[CSOC] = 00/11, write FOC_CSO to modify the offset value of ITRIP and IC.
2. When FOC_CHC[CSOC] = 01, write FOC_CSO to modify the offset value of IA.
3. When FOC_CHC[CSOC] = 10, write FOC_CSO to modify the offset value of IB.

14.1.9 Angle Mode

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

1. Forced Ramping angle
2. Forced pulling angle
3. Estimator estimated angle
4. Estimator forced angle

Table 14-2 Angle source

RFAE	ANGM	EFAE	Source of the angle
1	x	x	Forced ramping angle
0	0	x	Forced pulling angle
0	1	0	Estimator estimated angle
0	1	1	Estimated speed > EFREQMIN: Estimator estimated angle Estimated speed < EFREQMIN: Estimator forced angle

14.1.9.1 Forced Ramping Angle

Forced ramping angle consists of angle THETA, velocity RTHESTEP, acceleration RTHEACC and ramping counter RTHECNT. The formula is:

RTHESTEP (32bit) = RTHESTEP (32bit) + RTHEACC (32bit, the higher 16 bit are fixed to zero and the lower 16 bit are configurable).

THETA (16bit) = THETA (16bit) + RTHESTEP (higher 16 bit)

Forced ramping angle has a highest priority. Set FOC_CR1[RFAE] to 1 to enable the ramping function. Ramping module makes a ramping operation in every cycle. When the value of the counter reaches RTHECNT, RFAE is cleared by hardware, and then the ramping is over. Thereafter, according to the value of FOC_CR1[ANGM], the angle comes from estimator (ANGM = 1) or forced pulling angle (ANGM = 0).

14.1.9.2 Forced Pulling Angle

Forced pulling angle consists of angle THETA and velocity RTHESTEP. The formula is:

THETA (16bit) = THETA (16bit) + RTHESTEP (higher16 bit)

There are two states in the forced pulling angle:

1. Setting FOC_CR1[RFAE] to 1 and FOC_CR1[ANGM] to 0, forced pulling mode is started after ramping function. The speed RTHESTEP will be the cumulative result at the end of the ramping. This mode may achieve uniform and forced pulling without angle feedback function.
2. Setting FOC_CR1[RFAE] to 0 and FOC_CR1[ANGM] to 0, forced pulling angle is started without going by the ramping module. The speed RTHESTEP will be the initial speed writing in the register by software. When RTHESTEP is set to 0, localization function is realized. When RTHESTEP is not set to 0, HALL effected FOC is enabled. (HALL effected FOC principle: MCU calculate the existing angle and the speed when the HALL signal is received, and writes to the THETA and RTHESTEP to make modifications).

14.1.9.3 Estimated Angle of Estimator

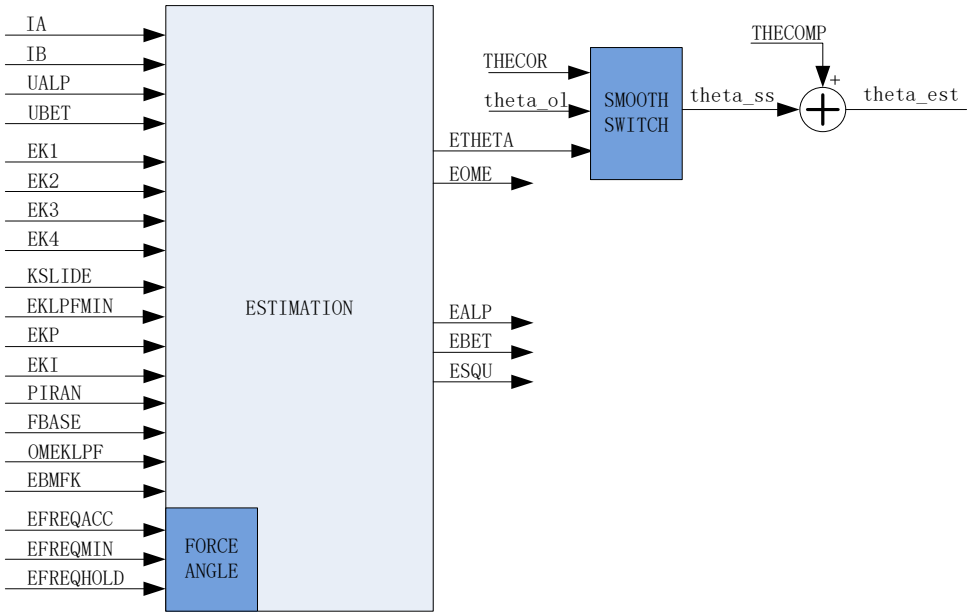


Figure 14-12 Functional Block Diagram of Estimator

The estimator samples the current and voltage of the motor and outputs the angle, speed and BEMF based on the motor parameters and control parameters.

1. Estimated angle of estimator

The estimator builds the motor model based on the motor parameters and control parameters, and samples the current and voltage of the motor to update the estimated value. After iterations, the estimator outputs the angles which are consistent with the real situation. The estimator can be selected as the PLL mode and sliding mode observer mode by configuring FOC_CR1[ESEL].

2. Forced estimated angle of estimator

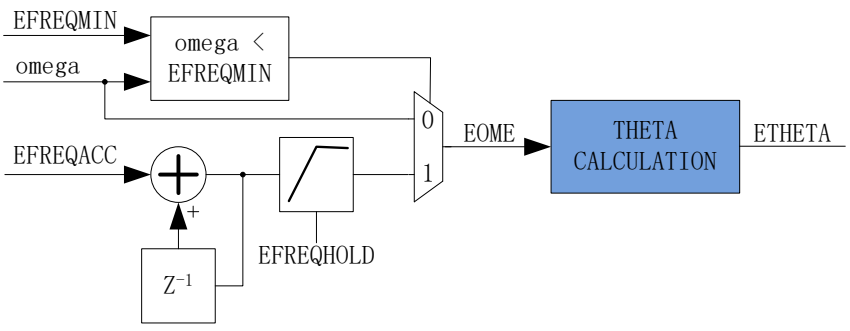


Figure 14-13 Functional Block Diagram of Forced Estimated Angle

This function is familiar as the ramping function. Due to the small Output at motor starting, there can be a huge difference in angle and speed estimation with the small sampling current. This may result in the

launch failure. In this case, the estimator outputs the forced pulling angle to make the motor start successfully.

Set FOC_CR1[EFAE] to 1. When the estimated speed from the estimator is less than the minimum value set, the system will be started compulsorily from zero speed. The speed is increased with the speed increment set in every operation cycle, but with the maximum value of the speed limited to EFREQHOLD. Estimator angle ETHETA is estimated using the final speed EOME which is the forced speed. When OMEGA is larger than or equal to EFREQMIN, the final speed EOME is the estimated speed.

3. Smooth switching on the estimated angle

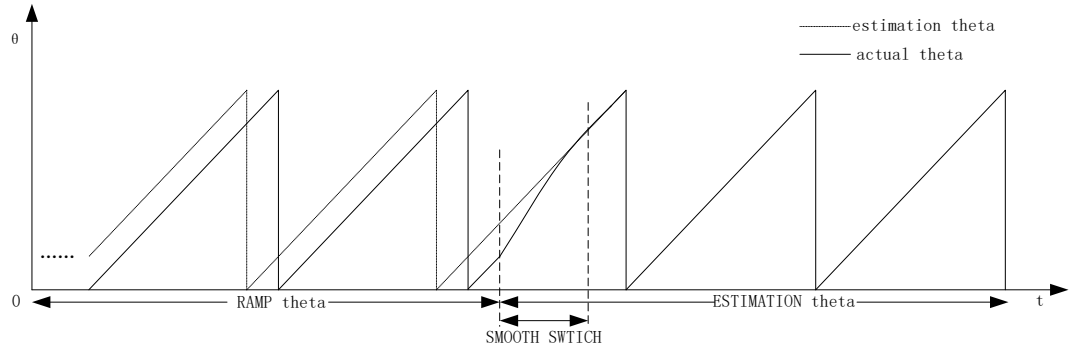


Figure 14-14 Smooth Switching Curve of the Estimated Angle

Setting the FOC_CR1[RFAE] and FOC_CR1[ANGM] to 1, the motor operates with the ramping function. At the end of the ramping, it switches to estimator mode. The estimator also estimates the angle in the ramping, but usually there is a deviation between the estimated angle and the forced ramping angle. If the angle is switched from forced ramping angle to estimated angle after the ramping, there could be motor jitter due to angle difference. Therefore, the smooth switching module of the estimated angle is preferred.

At the end of the ramping, if the devotion between estimated angle ETHETA and the forced ramping angle theta_ol is less than or equal to the THECOR, ETHETA is used directly as the output angle. If the deviation is larger than THECOR, theta_ol will be modified smoothly to close ETHETA with the step of the THECOR at every operation cycle. When the deviation is less than THECOR,ETHETA is used directly as the output angle.

4. Angle compensation

THECOMP is used as an offset of the estimated angle. When the first bit is 1, it is the negative compensation. When the first bit is 0, it is the positive compensation.

14.1.10 Motor Realtime Parameters

FOC module can generate the following realime parameters in motor control

1. Angle: THETA
2. Estimated angle: ETHETA. Estimated speed: EOME
3. d-Axis voltage: UD. q-Axis voltage: UQ
4. d-Axis voltage: ID. q-Axis voltage: IQ
5. α-Axis voltage: VALP. β-Axis voltage: VBET
6. BUS voltage: UDCFLT
7. Three phase current: IA, IB and IC. The maximum value of three phase current: IAMAX, IBMAX and ICMAX.
8. IALP and IBET obtained with Calrke transformation
9. αβ-Axis BEMF: EALP and EBET
10. The squared of BEMF: ESQU
11. Power: POW

14.1.10.1 Forward and Reverse Rotation Detection

FOC provides dedicated forward and reverse rotation detection module. Setting the reference input current IDREF and IQREF to 0. Enable FOC module, and the motor state is predicted by reading the estimated angle ETHETA and estimated speed EOME. If ETHETA decreases downward or EOME is a negative value, it is in a reverse rotation. It is then required to brake first and start motor with the forced angle mode. If ETHETA increases forward or EOME is a negative value, it is in a forward rotation. At this moment, it may switch to estimated angle mode directly to start motor.

14.1.10.2 BEMF Detection

Estimator estimates α-axis BEMF EALF and β-axis BEMF EBET with the input motor parameters, and outputs $e\alpha^2 + e\beta^2$, namely ESQU. The user can predict the launch state by using the value of ESQU to implement functions such as stall protection or unconnected phase protection.

14.1.10.3 Power

From the current, conduction time provided from the internal SVPWM module in FOC, together with the BUS voltage, power can be calculated.

14.2 FOC Register

14.2.1 FOC_CR1 (0x40A0)

Table 14-3 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	Function
[7]	OVMDL	Overmodulation enable 0: Disable 1: Enable
[6]	EFAE	Estimator forced angle enable When EFAE set 1, angle is provided by estimator, and auto switch to angle close loop 0: Disable 1: Enable
[5]	RFAE	Forced ramping angle enable When RFAE set 1, angle is provided by ramping module. At the end of ramping, RFAE is cleared by hardware and switch to estimator mode or forced pulling mode 0: Disable 1: Enable
[4]	ANGM	Angle mode RFAE = 0, the angle is from estimator mode or forced pulling mode. RFAE = 1, at the end of ramping, the angle is from estimator mode or forced pulling mode. 0: Estimator mode 1: Forced pulling mode
[3:2]	CSM	Current sampling mode. x0: Single-resistance sampling 01: Dual-resistances sampling 11: Triple-resistances sampling
[1]	SPWMSEL	SPWM mode selection 0: Unipolar mode 1: Bipolar mode
[0]	SVPWMEN	SVPWM/SPWM mode selection 1: SVPWM 0: SPWM

14.2.2 FOC_CR2 (0x40A1)

Table 14-4 FOC_CR2 (0x40A1)

Bit	7	6	5	4	3	2	1	0
Name	ESEL	ICLR	F5SEGn	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	Function

[7]	ESEL	Estimator mode selection. 0: Sliding Mode Observer (SMO) 1: Phase Locked Loop (PLL). When ESEL = 1, the FOC_KSILDE register is KP in PLL PI controller and the FOC_KLPFMIN register is KI in PLL PI controller
[6]	ICLR	Clear FOC__IA/B/CMAX to 0 Setting this bit to 1, the FOC__IA/B/CMAX register is cleared to 0. This bit is automatically cleared. 0: Invalid 1: Cleared
[5]	F5SEG	SVPWM mode at dual/triple-resistances sampling mode 0: Seven-segment selection 1: Five-segment selection
[4]	DSS	Sampling timing at dual/triple resistances sampling mode 0: Concurrent sampling mode 1: Alternating sampling mode
[3:2]	CSOC	Current sampling offset calibration Write this bit to set the current sample offset. For single-resistance sampling, write 00 or 11 to set ITRIP offset. For dual-resistances sampling, write 01 to set IA offset and write 10 to set IB offset. For triple-resistances sampling, write 01 to set IA offset, write 10 to set IB offset and write 00 or 11 to set IC offset. 00,11: ITRIP and IC 01: IA 10: IB
[1]	UQD	Disable q-axis PI controller. When UQD = 1, FOC__UQ is not updated by PI controller 0: Enable q-axis PI controller 1: Disable q-axis PI controller
[0]	UDD	Disable d-axis PI controller. When UDD = 1, FOC__UD is not updated by PI controller 0: Enable d-axis PI controller 1: Disable d-axis PI controller

14.2.3 FOC_TSMIN (0x40A2)

Table 14-5 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	Function
[7:0]	FOC_TSMIN	The minimum window for ADC sampling under single resistor sampling mode Deadtime compensation time in dual/triple resistances sampling mode

		Range: (0,255)
TS = sample window (ΔT) + deadtime (DT) E.g.: If ΔT = 1μs, DT = 1μs, TS = 2μs, and carrier period = 62.5μs, TS = 2048 × 2 / 62.5 = 65		

14.2.4 FOC_TGLI (0x40A3)

Table 14-6 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	Function
[7:0]	FOC_TGLI	High side of bridge narrow pulse elimination This function is used in HV applications. The high side of bridge should be on longer than a certain time. Set this register to configure a value, and high side of bridge will not turn on if the conducting time is less than this value. Range: (0,255)
Providing the value of narrow pulses elimination is 1μs, DT = 1μs and carrier cycle is 62.5μs, FOC_TGLI = 2048 × 2 / 62.5 = 65		

14.2.5 FOC_TBLO (0x40A4)

Table 14-7 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	Function
[7:0]	FOC_TBLO	For triple-resistances sampling mode, it is the unavailable time for current sampling. When the low side of bridge conducting time is less than FOC_TBLO, the current of this phase is not sampled and obtained by special processing. Range: (0,255)
Providing the value of unavailable time is 1μs, FOC_TBLO = 1000 / 41.67 = 24		

14.2.6 FOC_TRGDLY (0x40A5)

Table 14-8 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	Function
[7:0]	FOC_TRGDLY	Single-resistance sampling mode: current sampling triggering is delayed. Range: (-128,127) Dual/triple-resistance sampling mode: current sampling timing TRGDLY[7] = 0: counter up counting direction TRGDLY[7] = 1: counter down counting direction Range: (0, DRV_ARR[6:0])
Single-resistance sampling mode: MCU clock = 24MHz (41.67ns), TRGDLY = 5, delay 41.67ns × 5 = 208.3ns; TRGDLY = –5, advance 41.67ns × 5=208.3ns Dual/triple-resistance sampling mode: MCU clock = 24MHz (41.67ns), TRGDLY = 0x85, sample current in 41.67ns × 5 = 208.3ns before counter minima; TRGDLY = 0x05, sample current in 41.67ns × 5 = 208.3ns after counter minima		

14.2.7 FOC_CSO (0x40A6, 0x40A7)

Table 14-9 FOC_CSOH (0x40A6)

Bit	7	6	5	4	3	2	1	0
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

Table 14-10 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	Function
[15:0]	FOC_CSO	Current sampling offset Set FOC_CR1[CSOC] to select the corresponding current, write FOC_CSO to calibrate the current sampling offset of ITRIP in single-resistance sampling mode, IA, IB in dual-resistances sampling mode and IA, IB and IC in triple-resistances sampling mode. Range: (0,32767)
If the range of ADC is 0~5V and the offset is 2.5V, FOC_CSO = 32768 × 2.5V / 5V = 16384 (0x4000)		

14.2.8 FOC__RTHESTEP (0x40A8, 0x40A9)

Table 14-11 FOC__RTHESTEPH (0x40A8)

Bit	7	6	5	4	3	2	1	0

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

Table 14-12 FOC__RTHESTEPL (0x40A8)

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	Function
[15:0]	FOC__RTHESTEP	Ramping step Software write: Initial step value Software read: Current step value Range: (-32768,32767)
RTHESTEP (32bit) = RTHESTEP (32bit) + RTHEACC (32bit, the higher 16 bit are zero, and lower 16 bit are configurable); THETA (16bit) = THETA (16bit) + RTHESTEP (high16bit)		

14.2.9 FOC_RTHEACC (0x40AA, 0x40AB)

Table 14-13 FOC_RTHEACCH (0x40AA)

Bit	7	6	5	4	3	2	1	0
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

Table 14-14 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	Function
[15:0]	FOC_RTHEACC	Ramping acceleration, and format is the same as FOC__THETA Range: (-32768,32767) Note: FOC_RTHETACC is internally represented as 32bit, and the 1st bit is sign bit.
RTHESTEP (32bit) = RTHESTEP (32bit) + RTHEACC (32bit, the higher 16 bit are zero, and lower 16 bit are configurable); THETA (16bit) = THETA (16bit) + RTHESTEP (High 16bit)		

14.2.10 FOC__RTHECNT (0x40AC)

Table 14-15 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	Function
[7:0]	FOC__RTHECNT	Ramping times = RTHECNT × 256 When ramping function is enabled (FOC_CR1[RFAE] is set to 1), the ramping function increases by RTHECNT × 256 times.

14.2.11 FOC__THECOR (0x40AD), Shared with BLDC Mode

Table 14-16 FOC__THECOR (0x40AD)

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

Bit	Name	Function
[7:0]	FOC__THECOR	Angle switching correction When the ramping ends, angle mode switches to estimator mode from ramping mode. There is a deviation between estimated angle and ramping angle, and the angle compensation is necessary. The format is the same as FOC__THETA. Range: (0,255)

14.2.12 FOC__THECOMP (0x40AE, 0x40AF)

Table 14-17 FOC__THECOMPH (0x40AE)

Bit	7	6	5	4	3	2	1	0
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

Table 14-18 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	Function
[15:0]	FOC__THECOMP	Angle compensation value

		The estimated angle plus the compensation value is the output angle of estimator Range: (-32768,32767)

14.2.13 FOC_DMAX (0x40B0, 0x40B1)

Table 14-19 FOC_DMAXH (0x40B0)

Bit	7	6	5	4	3	2	1	0
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

Table 14-20 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	Function
[15:0]	FOC_DMAX	Maximum limit of d-axis PI controller output UD Range: (-32768,32767)

14.2.14 FOC_DMIN 0x40B2, 0x40B3)

Table 14-21 FOC_DMINH (0x40B2)

Bit	7	6	5	4	3	2	1	0
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

Table 14-22 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	Function
[15:0]	FOC_DMIN	Minimum limit of d-axis PI controller output UD Range: (-32768,32767)

14.2.15 FOC_QMAX (0x40B4, 0x40B5)

Table 14-23 FOC_QMAXH (0x40B4)

Bit	7	6	5	4	3	2	1	0

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

Table 14-24 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	Function
[15:0]	FOC_QMAX	Maximum limit of q-axis PI controller output UQ Range: (-32768,32767)

14.2.16 FOC_QMIN (0x40B6, 0x40B7)

Table 14-25 FOC_QMINH (0x40B6)

Bit	7	6	5	4	3	2	1	0
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

Table 14-26 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	Function
[15:0]	FOC_QMIN	Minimum limit of q-axis PI controller output UQ Range: (-32768,32767)

14.2.17 FOC__UD (0x40B8, 0x40B9)

Table 14-27 FOC__UDH (0x40B8)

Bit	7	6	5	4	3	2	1	0
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

Table 14-28 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	Function
[15:0]	FOC__UD	The Output d-axis PI controller Range: (-32768,32767)

14.2.18 FOC__UQ (0x40BA, 0x40BB)

Table 14-29 FOC__UQH (0x40BA)

Bit	7	6	5	4	3	2	1	0
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

Table 14-30 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	Function
[15:0]	FOC__UQ	The Output q-axis PI controller Range: (-32768,32767)

14.2.19 FOC__ID (0x40BC, 0x40BD)

Table 14-31 FOC__IDH (0x40BC)

Bit	7	6	5	4	3	2	1	0
Name	FOC__ID[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-32 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	Function

[15:0]	FOC__ID	The current ID from PARK transformation Range: (-32768,32767)

14.2.20 FOC__IQ (0x40BE, 0x40BF)

Table 14-33 FOC__IQH (0x40BE)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IQ[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-34 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	Function
[15:0]	FOC__IQ	The current IQ from PARK transformation Range: (-32768,32767)

14.2.21 FOC__IBET (0x40C0, 0x40C1)

Table 14-35 FOC__IBETH (0x40C0)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IBET [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-36 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	Function
[15:0]	FOC__IBET	The current IBETA from CLARKE transformation Range: (-32768,32767)

14.2.22 FOC__VBET (0x40C2, 0x40C3)

Table 14-37 FOC__VBETH (0x40C2)

Bit	7	6	5	4	3	2	1	0
Name	FOC__VBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-38 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	Function
[15:0]	FOC__VBET	VBETA from inverse PARK transformation Range: (-32768,32767)

14.2.23 FOC__VALP (0x40C4, 0x40C5)

Table 14-39 FOC__VALPH (0x40C4)

Bit	7	6	5	4	3	2	1	0
Name	FOC__VALP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-40 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	Function
[15:0]	FOC__VALP	VALPHA from inverse PARK transformation Range: (-32768,32767)

14.2.24 FOC_UDCPS (0x40C2, 0x40C3)

Table 14-41 FOC_UDCPSH (0x40C2)

Bit	7	6	5	4	3	2	1	0
Name	FOC_UDCPS [15:8]							

Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Table 14-42 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	Function
[15:0]	FOC_UDCPS	d-axis voltage compensation value (write only) The result of d-axis PI controller plus the compensation value is the Output d-axis PI controller Range: (-32768, 32767)

14.2.25 FOC_UQCPS (0x40C4, 0x40C5)

Table 14-43 FOC_UQCPSH (0x40C4)

Bit	7	6	5	4	3	2	1	0
Name	FOC_UQCPS [15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Table 14-44 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	Function
[15:0]	FOC_UQCPS	q-axis voltage compensation value (write only) The result of q-axis PI controller plus the compensation value is the Output q-axis PI controller Range: (-32768, 32767)

14.2.26 FOC__IC (0x40C6, 0x40C7)

Table 14-45 FOC__ICH (0x40C6)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IC[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-46 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	Function
[15:0]	FOC__IC	The phase current IC Range: (-32768, 32767)

14.2.27 FOC__IB (0x40C8, 0x40C9)

Table 14-47 FOC__IBH (0x40C8)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IB[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-48 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	Function
[15:0]	FOC__IB	The phase current IB Range: (-32768, 32767)

14.2.28 FOC__IA (0x40CA, 0x40CB)

Table 14-49 FOC__IAH (0x40CA)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IA[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-50 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	Function

[15:0]	FOC__IA	The phase current IA Range: (-32768, 32767)

14.2.29 FOC__THETA (0x40CC, 0x40CD)

Table 14-51 FOC__THETAH (0x40CC)

Bit	7	6	5	4	3	2	1	0
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

Table 14-52 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	Function
[15:0]	FOC__THETA	Angle Software write: the forced pulling angle Software read: FOC module angle Range: (-32768,32767)
Range: (-32768,32767), mapping (-180°,180°) If THETA = 8192, the angle =180° × 8192 / 32768 = 45°		

14.2.30 FOC__ETHETA (0x40CE, 0x40CF)

Table 14-53 FOC__ETHETAH (0x40CE)

Bit	7	6	5	4	3	2	1	0
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

Table 14-54 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	Function
[15:0]	FOC__ETHETA	Software read: The angle estimated by estimator (angle before compensation). The format is the same as FOC__THETA.

		Software write: The initial angle in estimator. Range: (-32768,32767)

14.2.31 FOC__EALP (0x40D0, 0x40D1)

Table 14-55 FOC__EALPH (0x40D0)

Bit	7	6	5	4	3	2	1	0
Name	FOC__EALP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-56 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	Function
[15:0]	FOC__EALP	The α-axis estimated BEMF EALPHA Range: (-32768,32767)

14.2.32 FOC__EBET (0x40D2, 0x40D3)

Table 14-57 FOC__EBETH (0x40D2)

Bit	7	6	5	4	3	2	1	0
Name	FOC__EBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-58 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	Function
[15:0]	FOC__EBET	The β-axis estimated BEMF EBETA Range: (-32768,32767)

14.2.33 FOC__EOME (0x40D4, 0x40D5)

Table 14-59 FOC__EOMEH (0x40D4)

Bit	7	6	5	4	3	2	1	0
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

Table 14-60 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	Function
[15:0]	FOC__EOME	The estimated speed OMEGA Range: (-32768,32767)

14.2.34 FOC__UQEX (0x40D6, 0x40D7)

Table 14-61 FOC__UQEXH (0x40D6)

Bit	7	6	5	4	3	2	1	0
Name	FOC__UQEX [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-62 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	Function
[15:0]	FOC__UQEX	Overflow value of q-axis PI controller. FOC__UQEX = FOC__UQ – FOC_QMAX When FOC__UQ > FOC_QMAX, FOC__UQEX is positive; When FOC__UQ < FOC_QMAX, FOC__UQEX is negative; FOC__UQEX is used to as the input of field weakening controller. For details, please refer to the example. Value range (-32768, 32768)

14.2.35 FOC__POW (0x40D8, 0x40D9)

Table 14-63 FOC__POWH (0x40D8)

Bit	7	6	5	4	3	2	1	0
Name	FOC__POW[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-64 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	Function
[15:0]	FOC__POW	Power Range: (-32768, 32767)

14.2.36 FOC__IAMAX (0x40DA, 0x40DB)

Table 14-65 FOC__IAMAXH (0x40DA)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IAMAX [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-66 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	Function
[15:0]	FOC__IAMAX	A phase current maximum value This value is the maximum value of A phase current during a full electrical period. The current maximum value will not be cleared automatically. Set FOC_CR2[ICLR] to 1 to clear the current maximum value. Range: (-32768, 32767)

14.2.37 FOC__IBMAX (0x40DC, 0x40DD)

Table 14-67 FOC__IBMAXH (0x40DC)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IBMAX [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-68 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	Function
[15:0]	FOC__IBMAX	B phase current maximum value This value is the maximum value of B phase current during a full electrical period. The current maximum value will not be cleared automatically. Set FOC_CR2[ICLR] to 1 to clear the current maximum value. Range: (-32768, 32767)

14.2.38 FOC__ICMAX (0x40DE, 0x40DF)

Table 14-69 FOC__ICMAXH (0x40DE)

Bit	7	6	5	4	3	2	1	0
Name	FOC__ICMAX [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-70 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	Function
[15:0]	FOC__ICMAX	C phase current maximum value This value is the maximum value of C phase current during a full electrical period. The current maximum value will not be cleared automatically. Set FOC_CR2[ICLR] to 1 to clear the current maximum value.

		Range: (-32768, 32767)

14.2.39 FOC_EKP (0x4074, 0x4075), Shared with BLDC Mode

Table 14-71 FOC_EKPH (0x4074)

Bit	7	6	5	4	3	2	1	0
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

Table 14-72 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	Function
[15:0]	FOC_EKP	KP factor of estimator PI controller Range: (0,32767). The data format is Q12.

14.2.40 FOC_EKI (0x4076, 0x4077), Shared with BLDC Mode

Table 14-73 FOC_EKIH (0x4076)

Bit	7	6	5	4	3	2	1	0
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

Table 14-74 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	Function
[15:0]	FOC_EKI	KI factor of estimator PI controller Range: (0,32767). The data format is Q15.

14.2.41 FOC_EBMFK (0x407C, 0x407D), Shared with BLDC Mode

Table 14-75 FOC_EBMFKH (0x407C)

Bit	7	6	5	4	3	2	1	0

Name	FOC_EBMFK[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

Table 14-76 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	Function
[15:0]	FOC_EBMFK	EKLPF factor of BEMF low pass filter in angle estimator Range: (-32768, 32767). The data format is Q15.
EKLPF = EBMFK × OMEGA EBMFK = 2 × pi × fbase × △T		

14.2.42 FOC_KSLIDE (0x4078, 0x4079), Shared with BLDC Mode

Table 14-77 FOC_KSLIDEH (0x4078)

Bit	7	6	5	4	3	2	1	0
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

Table 14-78 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	Function
[15:0]	FOC_KSLIDE /FOC_PLLKP	SMO mode (ESEL = 0): KSLIDE, Q15 format. PLL mode (ESEL = 1): KP of PLL PI controller, Q12 format. Range: (0,32767)

14.2.43 FOC_EKLPFMIN (0x407A, 0x407B), Shared with BLDC Mode

Table 14-79 FOC_EKLPFMINH (0x407A)

Bit	7	6	5	4	3	2	1	0
Name	FOC_EKLPFMIN/ FOC_PLLKPI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset Value	0	0	0	0	0	0	0	0

Table 14-80 FOC_EKLPFMINL (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 Value	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:0]	FOC_EKLPFMIN / FOC_PLLKI	SMO mode (ESEL = 0): The minimum value of BEMF low pass filter factor EKLPF. The feedback value of angle estimator will be forced to this value when it is lower than this value. Q15 format. PLL mode (ESEL = 1): KI of PLL PI controller, Q15 format. Range: (0,32767)

14.2.44 FOC_OMEKLPF (0x407E, 0x407F)

Table 14-81 FOC_OMEKLPFH (0x407E)

Bit	7	6	5	4	3	2	1	0
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

Table 14-82 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	Function
[15:0]	FOC_OMEKLPF	The LPF factor for speed calculation Range: (0,32767), Q15 format.

14.2.45 FOC_FBASE (0x4080, 0x4081)

Table 14-83 FOC_FBASEH (0x4080)

Bit	7	6	5	4	3	2	1	0
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

Table 14-84 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	Function
[15:0]	FOC_FBASE	Base value of frequency. It is used to calculate DELTA THETA.
FBASE = fbase × △T × 65536 If fbase = 200Hz, △T = 62.5μs, then FBASE = 819		

14.2.46 FOC_EFREQACC (0x4082, 0x4083), Shared with BLDC Mode

Table 14-85 FOC_EFREQACCH (0x4082)

Bit	7	6	5	4	3	2	1	0
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

Table 14-86 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	Function
[15:0]	FOC_EFREQACC	Omega increment of the forced angle mode in angle estimator Range: (0,65535)
If fbase = 200Hz and motor pole-pairs = 4, speed_base = 60 × fbase / pp = 3000rpm. Set OMEGA increment to be 3rpm, FOC_EFREQACC = 32768 × 256 × 3 / speed_base = 8388(0x20c4)		

14.2.47 FOC_EFREQMIN (0x4084, 0x4085), Shared with BLDC Mode

Table 14-87 FOC_EFREQMINH (0x4084)

Bit	7	6	5	4	3	2	1	0
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

Table 14-88 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	Function
[15:0]	FOC_EFREQMIN	The minimum speed of estimator forced angle mode When the estimated speed is smaller than FOC_EFREQMIN, the estimator forced angle mode is proper operation Range: (-32768,32767)
If fbase = 200Hz and motor pole-pairs = 4, speed_base = 60 × fbase / pp = 3000rpm. Set the minimum OMEGA to be 30rpm, FOC_ EFREQMIN = 32768 × 30 / speed_base= 327(0x147)		

14.2.48 FOC_EFREQHOLD (0x4086, 0x4087), Shared with BLDC Mode

Table 14-89 FOC_EFREQHOLDH (0x4086)

Bit	7	6	5	4	3	2	1	0
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

Table 14-90 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	Function
[15:0]	FOC_EFREQHOLD	The holding speed of estimated forced angle mode In the estimator forced angle mode, the forced speed increases from 0 to FOC_EFREQHOLD Range: (-32768,32767)
If fbase = 200Hz，and motor pole-pairs = 4, speed_base=60 × fbase / pp=3000rpm. Set the holding speed of OMEGA to be 60rpm, FOC_ EFREQHOLD = 32768 × 60 / speed_base = 655(0x28f)		

14.2.49 FOC_EK3 (0x4088, 0x4089)

Table 14-91 FOC_EK3H (0x4088)

Bit	7	6	5	4	3	2	1	0

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

Table 14-92 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	Function
[15:0]	FOC_EK3	The 3rd factor of the current model in estimator Range: (0,32767), Q15 format.

14.2.50 FOC_EK4 (0x408A,0x408B)

Table 14-93 FOC_EK4H (0x408A)

Bit	7	6	5	4	3	2	1	0
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

Table 14-94 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	Function
[15:0]	FOC_EK4	The 4th factor of the current model in estimator Range: (-32768,32767), Q15 format.

14.2.51 FOC_EK1 (0x408C, 0x408D)

Table 14-95 FOC_EK1H (0x408C)

Bit	7	6	5	4	3	2	1	0
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

Table 14-96 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	Function
[15:0]	FOC_EK1	The 1st factor of the current model in estimator Range: (0,32767), Q15 format.

14.2.52 FOC_EK2 (0x408E, 0x408F)

Table 14-97 FOC_EK2H (0x408E)

Bit	7	6	5	4	3	2	1	0
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

Table 14-98 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	Function
[15:0]	FOC_EK2	The 2nd factor of the current model in estimator Range: (0,32767), Q15 format.

14.2.53 FOC__IDREF (0x4090, 0x4091), Shared with BLDC Mode

Table 14-99 FOC__IDREFH (0x4090)

Bit	7	6	5	4	3	2	1	0
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

Table 14-100 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	Function
[15:0]	FOC__IDREF	Reference of d-axis current Range: (-32768,32767)

14.2.54 FOC__IQREF (0x4092, 0x4093), Shared with BLDC Mode

Table 14-101 FOC__IQREFH (0x4092)

Bit	7	6	5	4	3	2	1	0
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

Table 14-102 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	Function
[15:0]	FOC__IQREF	The reference of q-axis current Range: (-32768,32767)

14.2.55 FOC_DQKP (0x4094, 0x4095), Shared with BLDC Mode

Table 14-103 FOC_DQKPH (0x4094)

Bit	7	6	5	4	3	2	1	0
Name	FOC_DQKP[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

Table 14-104 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	Function
[15:0]	FOC_DQKP	KP factor of d/q-axis PI controllers Range: (0,32767), Q12 format

14.2.56 FOC_DQKI (0x4096, 0x4097), Shared with BLDC Mode

Table 14-105 FOC_DQKIH (0x4096)

Bit	7	6	5	4	3	2	1	0
Name	FOC_DQKI[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

Table 14-106 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	Function
[15:0]	FOC_DQKI	KI factor of d/q-axis PI controllers Range: (0,32767), Q15 format

14.2.57 FOC__UDCFLT (0x4098, 0x4099)

Table 14-107 FOC__UDCFLTH (0x4098)

Bit	7	6	5	4	3	2	1	0
Name	FOC__UDCFLT[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 14-108 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	Function
[15:0]	FOC__UDCFLT	Filtered BUS voltage The FOC module samples the BUS voltage and filters it for motor control. ADC Channel-2 is used by default. Value range (0,32767)
Providing the sampled bus voltage is divided by 6 and ADC voltage range is 0 ~ 5V, the range of the BUS voltage is 0 ~ 30V. If FOC__UDCFLT is 19661(0x4CCD), then the BUS voltage = 5 × 6 × 19661 / 32768 = 18V.		

15 SPWM

15.1 SPWM Operating Descriptions

15.1.1 Introductions

FOC/SVPWM module is used in the applications of single-phase motor and stepper motor. As an independent module, the clock stops when the SPWM module is not working. It must be enabled by setting the DRV_CR[FOC_EN] to 1 before the operation of the SPWM module.

FOC module contains an angle module, a PI controller, a coordinate transformation module and an output module. The internal in SPWM module contains a closed loop current control, which can output eight channel PWM signals to drive the motor. The reference value ID and IQ will be provided by the user, and together with the ADC, which samples the current signal, closed loop current control can be implemented. U/V phase output is β-axis voltage, and W/X phase output is α-axis voltage

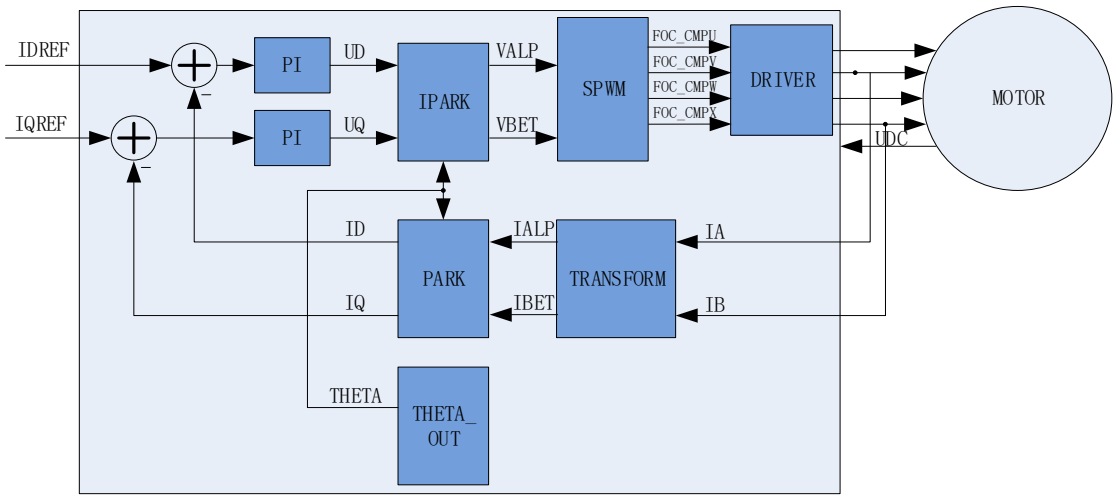


Figure 15-1 Functional Block Diagram of SPWM

15.1.2 Reference Input

SPWM module has closed loop current control, and d-axis current and q-axis current are adopted as the reference input. Set FOC__THETA to 0, α-axis current and β-axis current are controlled by the same module.

15.1.3 PI Controller

SPWM module contains 2 PI controllers, and they are applied to:

1 Flux control: PI controller of d-axis, reference current IDREF minus feedback current ID as the error input. Proportional coefficient DKP and the integral coefficient DKI for adjustment of PI performance. DMAX and DMIN for limiting of the output amplitude. The output, voltage of d-axis UD, is provided.

2 Torque control: PI controller of q-axis, reference current IQREF minus feedback current IQ as the error input. Proportional coefficient QKP and the integral coefficient QKI for adjustment of PI performance. QMAX and QMIN for limiting of output amplitude. The output, voltage of q-axis UQ, is provided.

15.1.4 Coordinate Transformation

15.1.4.1 Inverse PARK Transformation

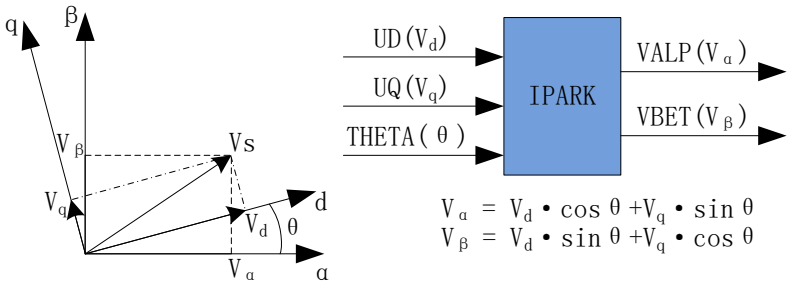


Figure 15-2 Inverse PARK Transformation

After the PI iteration, the two components of voltage vector on the rotating d-q axis can be obtained. From d-q axis, the transforms to α-β axis can thus be obtained with the inverse Park transform.

15.1.4.2 PARK Transformation

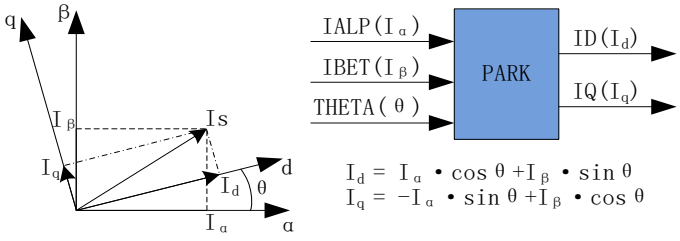


Figure 15-3 PARK Transformation

Park transformation converts the current components from a 2D coordinate system with the axis called α-β, into another 2D system called d-q, which is rotating with the rotor flux.

15.1.5 SPWM

15.1.5.1 Unipolar SPWM

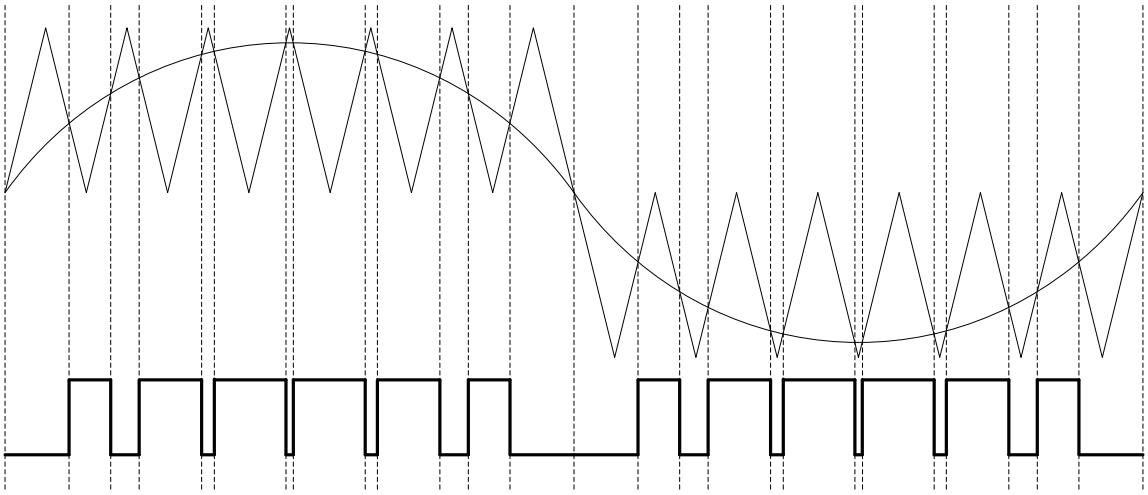


Figure 15-4 Unipolar SPWM

Setting FOC_CR1[SPWMSEL] to 0, SPWM outputs voltage in unipolar mode.

15.1.5.2 Bipolar SPWM

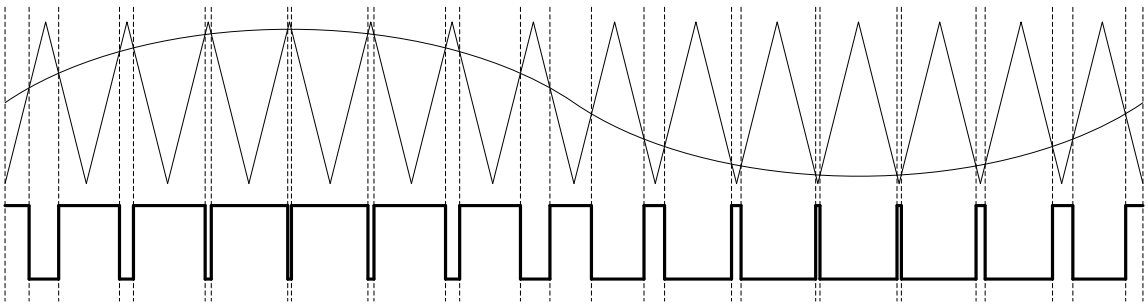


Figure 15-5 Bipolar SPWM

Setting FOC_CR1[SPWMSEL] to 1, SPWM outputs voltage in bipolar mode.

15.1.6 Current and Voltage Sampling

SPWM module needs to sample the BUS voltage of the inverter and phase current. Before using the SPWM module, users should enable ADC (set ADC_STA[ADCEN] to 1), Op Amp, and related registers must be configured. However, there is no need to configure ADC channels and scanning mode. Set FOC_CR1[CSM] to 10 to enable the current sampling. In the application of stepper motor, the default sampling channel of IALPHA is Channel-0 (corresponding to FOC__IA) and IBETA is Channel-1 (corresponding to FOC__IB). In the application of single-phase motor, Channel-0 (corresponding to FOC__IA) or Channel-1 (corresponding to FOC__IB) can be used to sample BUS current.

15.1.6.1 Current Sampling Offset

The input current sampling range should be doubled before the sampling due to the existence of the positive and negative phase current. So it is required to minus an offset during the operation, which the default value is 0x4000. However, there is a deviation between the default value and the real value, owing to ADC and hardware implementation. Therefore, it is necessary to calibrate an offset. The calibration procedure is as the following: When SPWM module doesn't work and there is no current in three phases, MCU starts to sample the corresponding channel and do a write to the FOC_CSO register after averaging all the sampled value. If the voltage range is 0~5V and the offset voltage is 2.5V, FOC_CSO = 32768 × 2.5V / 5V = 16384(0x4000).

1. When FOC_CHC[CSOC] = 01, write FOC_CSO to modify the offset value of IA.
2. When FOC_CHC[CSOC] = 10, write FOC_CSO to modify the offset value of IB.

15.1.7 Angle Mode

The angle mode can be selected as climbing mode and forced pulling mode。

15.1.7.1 Forced Ramping angle

Forced ramping angle consists of angle THETA, velocity RTHESTEP, acceleration RTHEACC and ramping counter RTHECNT. The formula is:

RTHESTEP (32bit) = RTHESTEP (32bit) + RTHEACC (32bit, the higher 16 bit are fixed to zero and the lower 16 bit are configurable).

THETA (16bit) = THETA (16bit) + RTHESTEP (higher 16 bit)

Forced ramping angle has a highest priority. Set FOC_CR1[RFAE] to 1 to enable the ramping function. Ramping module makes a ramping operation in every cycle. When the value of the counter reaches RTHECNT, RFAE is cleared by hardware, and then the ramping is over. Thereafter, the angle comes from forced pulling angle.

15.1.7.2 Forced Pulling angle

Forced pulling angle consists of angle THETA and velocity RTHESTEP. The formula is:

THETA (16bit) = THETA (16bit) + RTHESTEP (higher16 bit)

There are two states in the forced pulling angle:

1. Setting FOC_CR1[RFAE] to 1 and FOC_CR1[ANGM] to 0, forced pulling mode is started after ramping function. The speed RTHESTEP will be the cumulative result at the end of the ramping. This

mode may achieve uniform and forced pulling without angle feedback function.

2. Setting FOC_CR1[RFAE] to 0 and FOC_CR1[ANGM] to 0, forced pulling angle is started without going by the ramping module. The speed RTHESTEP will be the initial speed writing in the register by software. When RTHESTEP is set to 0, localization function is realized. When RTHESTEP is not set to 0, SPWM based on position sensor is enabled. MCU calculate the existing angle and the speed when the position signal is received, and writes to the THETA and RTHESTEP to make modifications.

15.1.8 Motor Realtime Parameters

SPWM module can generate the following realime parameters in motor control

1. Angle: THETA
2. d-Axis voltage: UD. q-Axis voltage: UQ
3. d-Axis voltage: ID. q-Axis voltage: IQ
4. α-Axis voltage: VALP. β-Axis voltage: VBET
5. BUS voltage: UDCFLT
6. IALP and IBET obtained with Calrke transformation

15.2 SPWM Register

15.2.1 FOC_CR1 (0x40A0)

Table 15-1 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	Function
[7]	OVMDL	Inverse IALPHA enable: 0: Disable 1: Enable
[6]	EFAE	Inverse IBETA enable: 0: Disable 1: Enable
[5]	RFAE	Forced ramping angle enable When RFAE set 1, angle is provided by ramping module. At the end of ramping, RFAE is cleared by hardware and switch to estimator mode or forced pulling mode 0: Disable 1: Enable

[4]	ANGM	In SPWM mode, this bit must be 0
[3:2]	CSM	In SPWM mode, this bit must be 01
[1]	SPWMSEL	SPWM output mode： 0: Unipolar 1: Bipolar
[0]	SVPWMEN	Selection in SVPWM/SPWM mode 1: SVPWM 0: SPWM

15.2.2 FOC_CR2 (0x40A1)

Table 15-2 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	Function
[7]	ESEL	IALPHA/IBETA automatic inversion 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: Concurrent sampling mode 1: Alternating sampling mode
[3:2]	CSOC	Current sampling offset calibration Write this bit to set the current sample offset. In SPWM mode, write 01 to set IALPHA offset and write 10 to set IBETA offset. 00,11: Invalid 01: IALPHA 10: IBETA
[1]	UQD	Disable q-axis PI controller. When UQD = 1, FOC__UQ is not updated by PI controller 0: Enable q-axis PI controller 1: Disable q-axis PI controller
[0]	UDD	Disable d-axis PI controller. When UDD = 1, FOC__UD is not updated by PI controller 0: Enable d-axis PI controller 1: Disable d-axis PI controller

15.2.3 FOC_TRGDLY (0x40A5)

Table 15-3 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	Function
[7:0]	FOC_TRGDLY	SPWM mode: current sampling timing TRGDLY[7] = 0: counter up counting direction TRGDLY[7] = 1: counter down counting direction Range: (0, DRV_ARR[6:0])
SPWM mode: MCU clock = 24MHz (41.67ns), TRGDLY = 0x85, sample current in 41.67ns × 5 = 208.3ns before counter minima; TRGDLY = 0x05, sample current in 41.67ns × 5 = 208.3ns after counter minima.		

15.2.4 FOC_CSO (0x40A6, 0x40A7)

Table 15-4 FOC_CSOH (0x40A6)

Bit	7	6	5	4	3	2	1	0
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

Table 15-5 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	Function
[15:0]	FOC_CSO	Current sampling offset Configure FOC_CR1[CSOC] to select IALPHA and IBETA, write the offset to this register. Range: (0,32767)
If the range of ADC is 0~5V and the offset is 2.5V, FOC_CSO = 32768 × 2.5V / 5V = 16384 (0x4000)		

15.2.5 FOC__RTHESTEP (0x40A8, 0x40A9)

Table 15-6 FOC__RTHESTEPH (0x40A8)

Bit	7	6	5	4	3	2	1	0
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

Table 15-7 FOC__RTHESTEPL (0x40A8)

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	Function
[15:0]	FOC__RTHESTEP	Ramping step Software write: Initial step value Software read: Current step value Range: (-32768,32767)
RTHESTEP (32bit) = RTHESTEP (32bit) + RTHEACC (32bit, the higher 16 bit are zero, and lower 16 bit are configurable); THETA (16bit) = THETA (16bit) + RTHESTEP (high16bit)		

15.2.6 FOC_RTHEACC (0x40AA, 0x40AB)

Table 15-8 FOC_RTHEACCH (0x40AA)

Bit	7	6	5	4	3	2	1	0
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

Table 15-9 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	Function
[15:0]	FOC_RTHEACC	Ramping acceleration, and format is the same as FOC__THETA Range: (-32768,32767) Note: FOC_RTHETACC is internally represented as 32bit, and the 1st bit is sign bit.
RTHESTEP (32bit) = RTHESTEP (32bit) + RTHEACC (32bit, the higher 16 bit are zero, and lower 16 bit are configurable); THETA (16bit) = THETA (16bit) + RTHESTEP (High 16bit)		

15.2.7 FOC__RTHECNT (0x40AC)

Table 15-10 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	Function
[7:0]	FOC__RTHECNT	Ramping times = RTHECNT × 256 When ramping function is enabled (FOC_CR1[RFAE] is set to 1), the ramping function increases by RTHECNT × 256 times.

15.2.8 FOC_DMAX (0x40B0, 0x40B1)

Table 15-11 FOC_DMAXH (0x40B0)

Bit	7	6	5	4	3	2	1	0
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

Table 15-12 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	Function
[15:0]	FOC_DMAX	Maximum limit of d-axis PI controller output UD Range: (-32768,32767)

15.2.9 FOC_DMIN (0x40B2, 0x40B3)

Table 15-13 FOC_DMINH (0x40B2)

Bit	7	6	5	4	3	2	1	0
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

Table 15-14 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	Function
[15:0]	FOC_DMIN	Minimum limit of d-axis PI controller output UD Range: (-32768,32767)

15.2.10 FOC_QMAX (0x40B4, 0x40B5)

Table 15-15 FOC_QMAXH (0x40B4)

Bit	7	6	5	4	3	2	1	0
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

Table 15-16 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	Function
[15:0]	FOC_QMAX	Maximum limit of q-axis PI controller output UQ Range: (-32768,32767)

15.2.11 FOC_QMIN (0x40B6, 0x40B7)

Table 15-17 FOC_QMINH (0x40B6)

Bit	7	6	5	4	3	2	1	0
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

Table 15-18 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	Function

[15:0]	FOC_QMIN	Minimum limit of q-axis PI controller output UQ Range: (-32768,32767)

15.2.12 FOC__UD (0x40B8, 0x40B9)

Table 15-19 FOC__UDH (0x40B8)

Bit	7	6	5	4	3	2	1	0
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

Table 15-20 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	Function
[15:0]	FOC__UD	The Output d-axis PI controller Range: (-32768,32767)

15.2.13 FOC__UQ (0x40BA, 0x40BB)

Table 15-21 FOC__UQH (0x40BA)

Bit	7	6	5	4	3	2	1	0
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

Table 15-22 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	Function
[15:0]	FOC__UQ	The Output q-axis PI controller Range: (-32768,32767)

15.2.14 FOC__ID (0x40BC, 0x40BD)

Table 15-23 FOC__IDH (0x40BC)

Bit	7	6	5	4	3	2	1	0
Name	FOC__ID[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 15-24 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	Function
[15:0]	FOC__ID	The current ID from PARK transformation Range: (-32768,32767)

15.2.15 FOC__IQ (0x40BE, 0x40BF)

Table 15-25 FOC__IQH (0x40BE)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IQ[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 15-26 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	Function
[15:0]	FOC__IQ	The current IQ from PARK transformation Range: (-32768,32767)

15.2.16 FOC__IBET (0x40C0, 0x40C1)

Table 15-27 FOC__IBETH (0x40C0)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IBET [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 15-28 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	Function
[15:0]	FOC__IBET	The current IBETA from CLARKE transformation Range: (-32768,32767)

15.2.17 FOC__VBET (0x40C2, 0x40C3)

Table 15-29 FOC__VBETH (0x40C2)

Bit	7	6	5	4	3	2	1	0
Name	FOC__VBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 15-30 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	Function
[15:0]	FOC__VBET	VBETA from inverse PARK transformation Range: (-32768,32767)

15.2.18 FOC__VALP (0x40C4, 0x40C5)

Table 15-31 FOC__VALPH (0x40C4)

Bit	7	6	5	4	3	2	1	0
Name	FOC__VALP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 15-32 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	Function
[15:0]	FOC__VALP	VALPHA from inverse PARK transformation Range: (-32768,32767)

15.2.19 FOC_UDCPS (0x40C2, 0x40C3)

Table 15-33 FOC_UDCPSH (0x40C2)

Bit	7	6	5	4	3	2	1	0
Name	FOC_UDCPS [15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Table 15-34 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	Function
[15:0]	FOC_UDCPS	d-axis voltage compensation value (write only) The result of d-axis PI controller plus the compensation value is the Output d-axis PI controller Range: (-32768, 32767)

15.2.20 FOC_UQCPS (0x40C4, 0x40C5)

Table 15-35 FOC_UQCPSH (0x40C4)

Bit	7	6	5	4	3	2	1	0
Name	FOC_UQCPS [15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Table 15-36 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	Function
[15:0]	FOC_UQCPS	q-axis voltage compensation value (write only) The result of q-axis PI controller plus the compensation value is the Output q-axis PI controller Range: (-32768, 32767)

15.2.21 FOC__IB (0x40C8, 0x40C9)

Table 15-37 FOC__IBH (0x40C8)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IB[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 15-38 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	Function
[15:0]	FOC__IB	The β-phase current IBETA Range: (-32768, 32767)

15.2.22 FOC__IA (0x40CA,0x40CB)

Table 15-39 FOC__IAH (0x40CA)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IA[15:8]							
Type	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0

Table 15-40 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 Value	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:0]	FOC__IA	The α-phase current IALPHA Range: (-32768, 32767)

15.2.23 FOC__THETA (0x40CC, 0x40CD)

Table 15-41 FOC__THETAH (0x40CC)

Bit	7	6	5	4	3	2	1	0
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

Table 15-42 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	Function
[15:0]	FOC__THETA	Angle Software write: the forced pulling angle Software read: SPWM module angle Range: (-32768,32767)
Range: (-32768,32767), mapping (-180°,180°) If THETA = 8192, the angle =180° × 8192 / 32768 = 45°		

15.2.24 FOC__IAMAX (0x40DA, 0x40DB)

Table 15-43 FOC__IAMAXH (0x40DA)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IAMAX [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 15-44 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	Function
[15:0]	FOC__IAMAX	α-phase current maximum value This value is the maximum value of α-phase current during a full electrical period. The current maximum value will not be cleared automatically. Set FOC_CR2[ICLR] to 1 to clear the current maximum value. Range: (-32768, 32767)

15.2.25 FOC__IBMAX (0x40DC, 0x40DD)

Table 15-45 FOC__IBMAXH (0x40DC)

Bit	7	6	5	4	3	2	1	0
Name	FOC__IBMAX [15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 15-46 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	Function
[15:0]	FOC__IBMAX	β-phase current maximum value This value is the maximum value ofβ-phase current during a full electrical period. The current maximum value will not be cleared automatically. Set FOC_CR2[ICLR] to 1 to clear the current maximum value. Range: (-32768, 32767)

15.2.26 FOC__IDREF (0x4090, 0x4091), Shared with BLDC Mode

Table 15-47 FOC__IDREFH (0x4090)

Bit	7	6	5	4	3	2	1	0
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

Table 15-48 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	Function
[15:0]	FOC__IDREF	Reference of d-axis current Range: (-32768,32767)

15.2.27 FOC__IQREF (0x4092, 0x4093), Shared with BLDC Mode

Table 15-49 FOC__IQREFH (0x4092)

Bit	7	6	5	4	3	2	1	0

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

Table 15-50 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	Function
[15:0]	FOC__IQREF	The reference of q-axis current Range: (-32768,32767)

15.2.28 FOC_DQKP (0x4094, 0x4095), Shared with BLDC Mode

Table 15-51 FOC_DQKPH (0x4094)

Bit	7	6	5	4	3	2	1	0
Name	FOC_DQKP[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

Table 15-52 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	Function
[15:0]	FOC_DQKP	KP factor of d/q-axis PI controllers Range: (0,32767), Q12 format

15.2.29 FOC_DQKI (0x4096, 0x4097), Shared with BLDC Mode

Table 15-53 FOC_DQKIH (0x4096)

Bit	7	6	5	4	3	2	1	0
Name	FOC_DQKI[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

Table 15-54 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	Function
[15:0]	FOC_DQKI	KI factor of d/q-axis PI controllers Range: (0,32767), Q15 format

15.2.30 FOC__UDCFLT (0x4098, 0x4099)

Table 15-55 FOC__UDCFLTH (0x4098)

Bit	7	6	5	4	3	2	1	0
Name	FOC__UDCFLT[15:8]							
Type	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0

Table 15-56 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 Value	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:0]	FOC__UDCFLT	Filtered BUS voltage The SPWM module samples the BUS voltage and filters it for motor control. ADC Channel-2 is used by default. Value range (0,32767)
Providing the sampled bus voltage is divided by 6 and ADC voltage range is 0 ~ 5V, the range of the BUS voltage is 0 ~ 30V. If FOC__UDCFLT is 19661(0x4CCD), then the BUS voltage = 5 × 6 × 19661 / 32768 = 18V.		

16 TIM1

16.1 Timer1 Operating Descriptions

Timer1 consists of a 16 bit up basic counter and a 16 bit up reload counter. The counter clock of them come from the internal clock. Timer 1 is mainly used in the applications of square wave current supplied BLDC and processing HALL signal. Timer1 features include:

1. The 16 bit up basic counter is used to record the count of basic timer between two position detections/the time of writing (i.e., count for 60 degree commutation).
2. The 16 bit up reload counter is used to record time: from position detection to the overflow of the reload counter (i.e., count for diode freewheeling time and the time of zcp to commutation).
3. 3 bit programmable frequency divider for two timer count clock frequency division.
4. Input filtering and sampling
5. Position signal generation of detection module according to the position detection signal input
6. Writing sequence module for updating the output states register
7. 7 state register control comparator and output
8. Interrupt event generated
   a) Basic timer overflow interrupt
   b) Reload timer overflow interrupt
   c) Writing sequence interrupt
   d) Position detection interrupt
   e) Diode freewheeling end interrupt

Timer 1 Internal structure as shown in Figure 16-1.

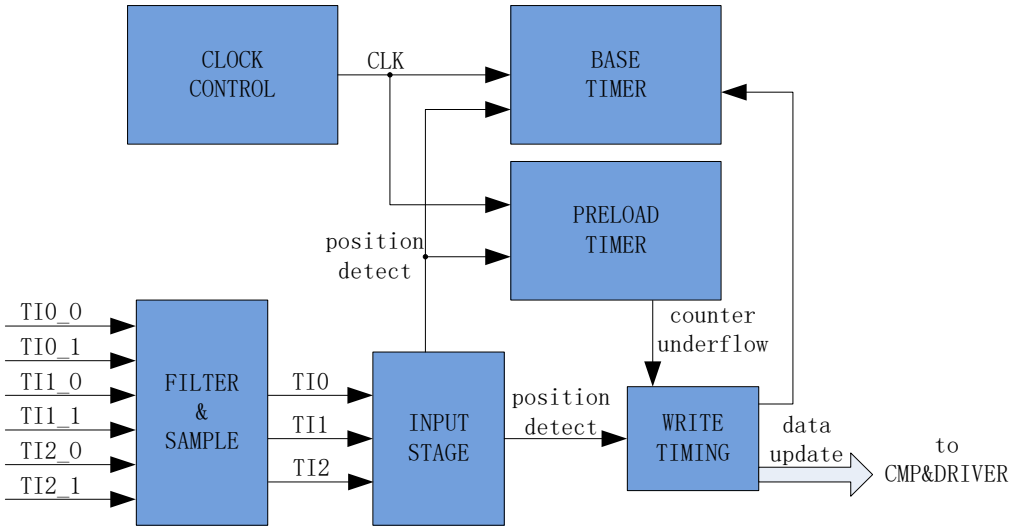


Figure 16-1 Internal Structure of Timer 1.

16.1.1 Timer Counter Unit

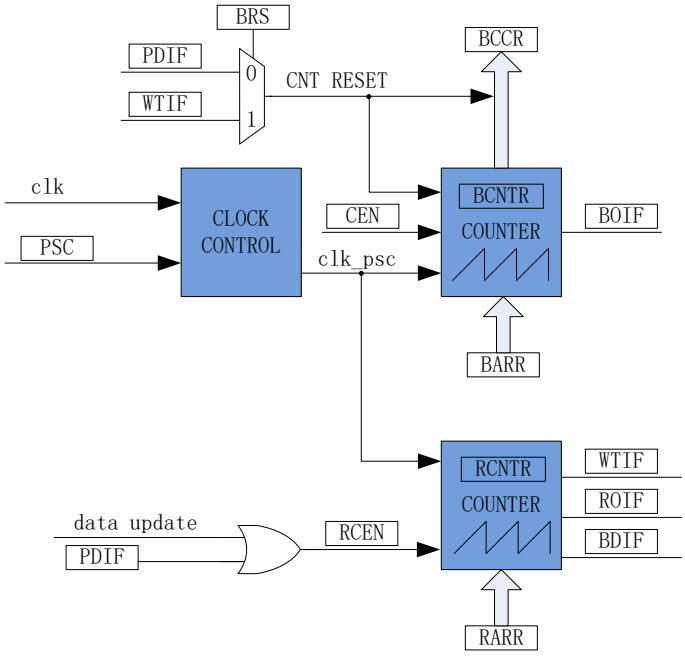


Figure 16-2 Timer 1 Base Unit

Timer1 consists of a frequency divider, a 16-bit up basic counter and a 16-bit down reload counter.

16.1.1.1 Timer Clock Controller

Timer clock controller is used to generate the basic timer and reload timer count clock sources. The count clock source frequency is divided by prescaler. It is a 12-bit counter with 4 bit PSC controller, and can select 16 kinds of frequency division coefficients. The clock source come from the internal clock. As there is no buffer in the control register, frequency division coefficient can be updated immediately, so the coefficients in the basic timer and reload timers should be updated when they are in sleeping state.

The frequency of the counter can be calculated as:

If MCU clock is 24MHz (41.67ns), $f_{CK\_CNT} = f_{CK\_PSC} / PSC$

Table 16-1 The Clock Frequency Corresponding to Different PSC Value

PSC	Coefficient	CLK(Hz)	PSC	Coefficient	CLK(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

16.1.1.2 Basic Timer

The basic timer contains a 16 bit up counter. When the value of TIM1__BCNTR equals to

TIM1__BARR, overflow event is triggered. The Overflow interrupt flag BOIF of the basic counter is set to 1. At the same time, TIM1__BCNTR is cleared and counting is restarted. Setting TIM1_CR2[BRS] chooses the counter resetting source from position detection event, or writing sequence event. When the event reset signal is generated, the current count value TIM1__BCNTR is sent to register TIM1__BCCR. TIM1__BCNTR value is set to 0, and then the counting is restarted.

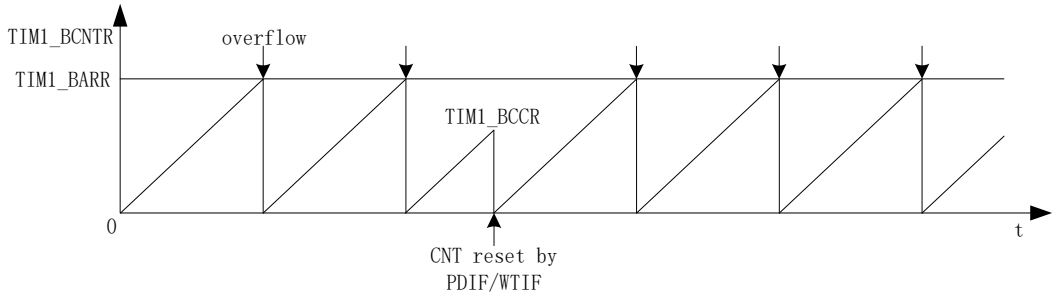


Figure 16-3 Waveform of the Basic Timer Count

The value of TIM1__BARR is sent to the counter immediately. So the register should be updated when the basic timer stops. Only when the count of TIM1__BCNTR equals to TIM1__BARR, overflow event generate. If TIM1__BCNTR > TIM1__BARR, TIM1__BCNTR will go to 0xFFFF, then restart count from 0. Therefore, the initial value of TIM1__BCNTR must be smaller than TIM1__BARR.

16.1.1.3 Reload Timer

The reload timer contains a 16 bit up counter. When the value of TIM1__BCNTR equals to TIM1__BARR, overflow event is generated. The Overflow interrupt flag ROIF of the road counter is set to 1. At the same time, TIM1__BCNTR and RCEN are cleared and counting will not be restarted until RCEN is set to 1.

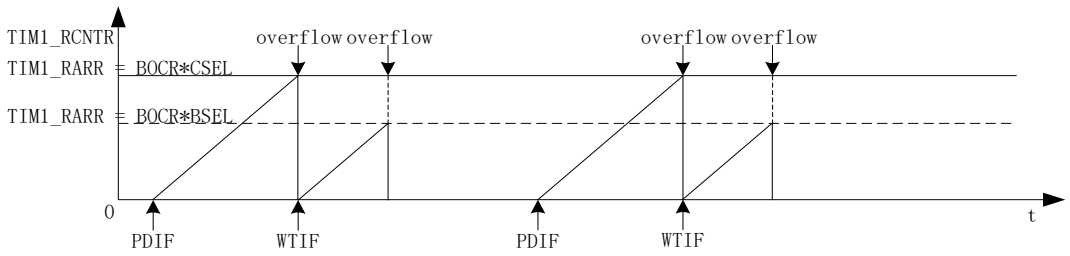


Figure 16-4 Waveform of Reload Timer Count

T1RCEN can be enable by position detection event and writing sequence event. When reload counter overflow event is triggered, T1RCEN is cleared by hardware.

16.1.2 Input Filtering and Sampling

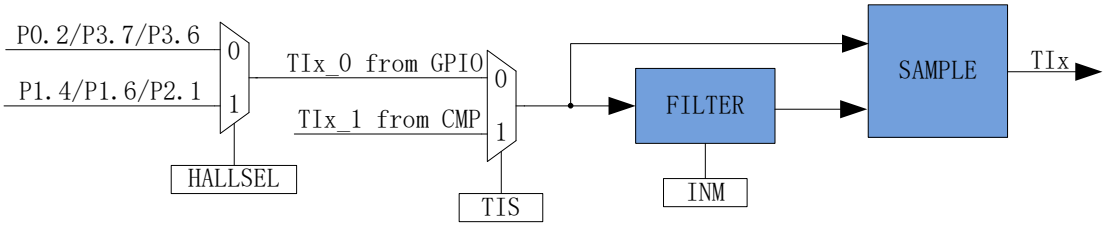


Figure 16-5 Schematic Diagram of Input Signal Filtering and Sampling

The value of TIM1_CR2[TIS] select the input source from comparator or GPIO. The GPIO pin can be selected as P1.4, P1.6 and P2.1 or P0.2, P3.7 and P3.6 by the configuration of CMP_CR1[HALLSEL]. Input noise filtering could be enabled by setting TIM1_CR3[INM]. The sampling mode of comparator is configurable by writing CMP_CR3[SAMSEL].

16.1.2.1 Filtering

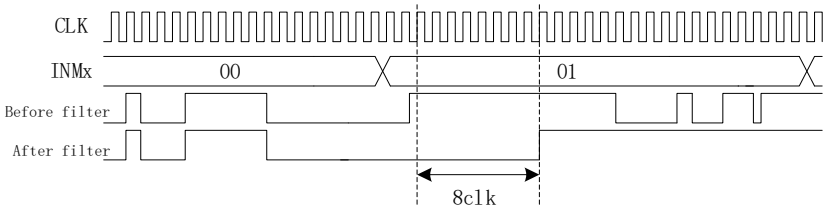


Figure 16-6 Sequence Diagram of Filter Module

According to TIM1_CR2[TIx], the filtering out pulse width of input noise can choose as 8/32/64periods. After enable the filtering, the signal will be delayed about 8~9/32~33/64~65 periods.

16.1.2.2 Sampling

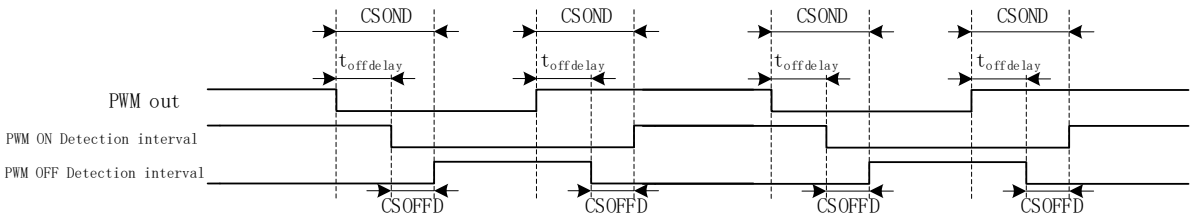


Figure 16-7 Diagram of Sampling comparator output time

For square wave current control based BLDC mode, the input source of TI2/TI1/TI0 is comparator. Since the Output comparator is distorted by the commutation of power devices. The sampling mode is configurable by writing CMP_CR3[SAMSEL]. The sampling interval can be configured by CSOFFD and

CSOND.

There is a delay from the PWM output effect to comparator. And the time is mainly decided by the followings: the resistance of driver, turn on delay time and turn off delay time of power device, input delay of comparator and hysteresis configuration. The sampling of comparator will be disabled due to this effect, and the time is configurable by CSOFFD and CSOND. The disable time of CMP0, CMP1 and CMP2 will be delayed for off delay = CSOND – CSOFFD.

e.g., providing the delay from PWM output effect to comparator is 2μs and the time of effect is 1μs, the CSOFFD and CSOND can be configured as

CSOFFD > 1μs = 1000ns / (41.67ns × 8) = 3

CSOND > (2 + 1) μs = 3000 / (41.67ns × 8) = 9

To measure the delay from PWM output effect to the comparator, the following steps could be implemented. Setting CMP_CR3[SAMSEL] = 00 disable sampling delay. Configuration of CMP_CR3[CMPSEL] output the corresponding comparator. PWM output and comparator is enabled. Users rotate motor to update the comparator output and the delay between PWM output and comparator output is obtained.

To measuring the width of comparator output distortion, the above method could be used.

16.1.3 Position Detection Event

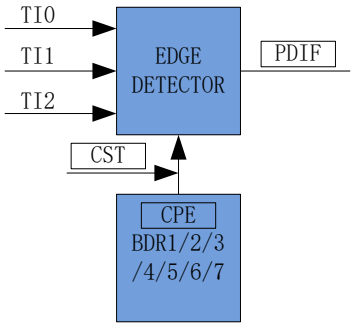


Figure 16-8 Block Diagram of Position Detection

The active input edge of positive detection event is configurable by TIM1_DBR1~7[CPE]. When active input edge (TI2/TI1/TI0) has been sampled, position detection event is generated. The CPE in corresponding register work based on TIM1_CR4[CST].

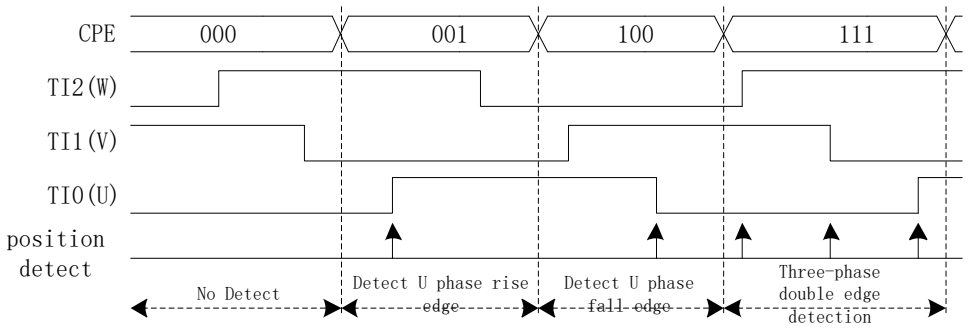


Figure 16-9 Sequence Diagram of Position Detection

The relation between active input edge and TIM1_DBR1~7[CPE] is as follows:

CPE	Description	CPE	Description
000	0	100	Falling edge of U phase, U phase corresponding comparator is enabled
001	Rising edge of U phase, U phase corresponding comparator is enabled	101	Rising edge of W phase, W phase corresponding comparator is enabled
010	Falling edge of W phase, W phase corresponding comparator is enabled	110	Falling edge of V phase, V phase corresponding comparator is enabled
011	Rising edge of V phase, V phase corresponding comparator is enabled	111	3-phase both edge, 3-phase corresponding comparator is enabled

16.1.4 Writing sequence Event

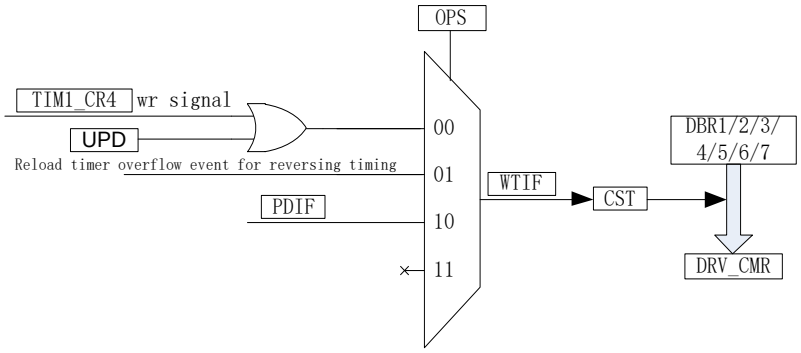


Figure 16-10 Block Diagram of Writing Sequence Event

Writing sequence event source is determined by TIM1_CR0[OPS]. When writing sequence happens, the interrupt flag WTIF is set to 1. Moreover, if TIM1_CR4[CST] is in 001~110, CST will plus 1 automatically and DRV_CMR is updated with the value of TIM1_DBR1~7.

16.1.5 Timer 1 Interrupt

Timer 1 has 5 interrupt sources:

1. Basic timer overflow interrupt
2. Reload timer overflow interrupt
3. Writing sequence interrupt
4. Position detection interrupt
5. Diode freewheeling end interrupt

Configuration of TIM1_IER can enable the corresponding interrupt source.

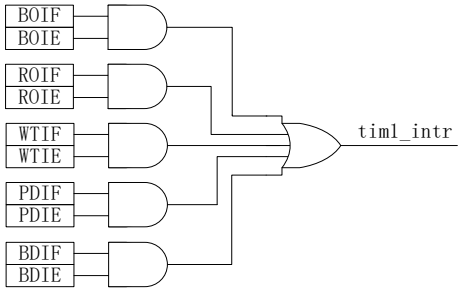


Figure 16-11 Timer 1 Interrupt Source

16.2 Square Wave Current Control based BLDC

For the application of square wave current control based BLDC, Timer 1, with comparator and Driver module, has the following functions:

1. Record the reference time, which can be filtered, of 60 degree commutation.
2. Forced commutation will be implemented when there is not any position signal.
3. Disable sampling comparator output during diode freewheeling.
4. Record the time from detection position signal to commutation. Commutate output direction automatically.
5. Replace CMP_CR2[CMP0_SEL] to control comparator 0.
6. Sampling comparator Output signal at PWM ON/OFF state. The signal can be filtered.
7. Replace DRV_CMR to control PWM output.

16.2.1 BLDC with Six-Step Commutation

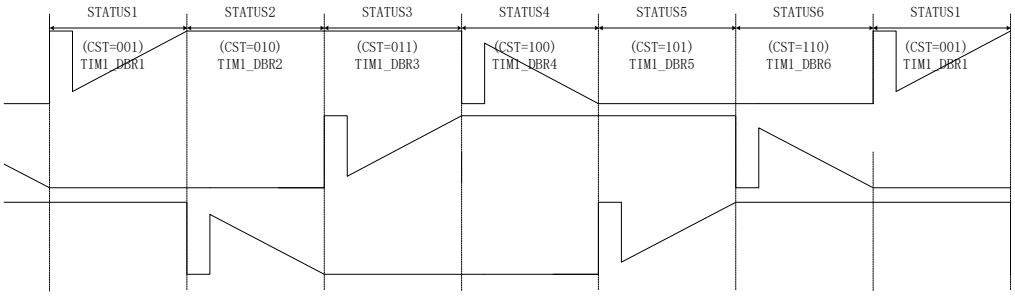


Figure 16-12 Diagram of BLDC Six-Step Commutation

TIM1_CR4[CST] is the predefined state, which maps to TIM1_DBR1~7. State 0 is used to turn off output. State 7 can be predefined to realize braking, pre-charging, localization, start or other functions. State 1~6 is used for six-step commutation. The state will plus 1 after commutation.

States 1~7 maps to TIM1_DBR 1~7. When writing sequence event happens, the corresponding TIM1_DBRx will updated to DRV_CMR and [CMP_CR2]CMP0_SEL.

16.2.2 Operating Principle of BLDC Control Based on Timer 1

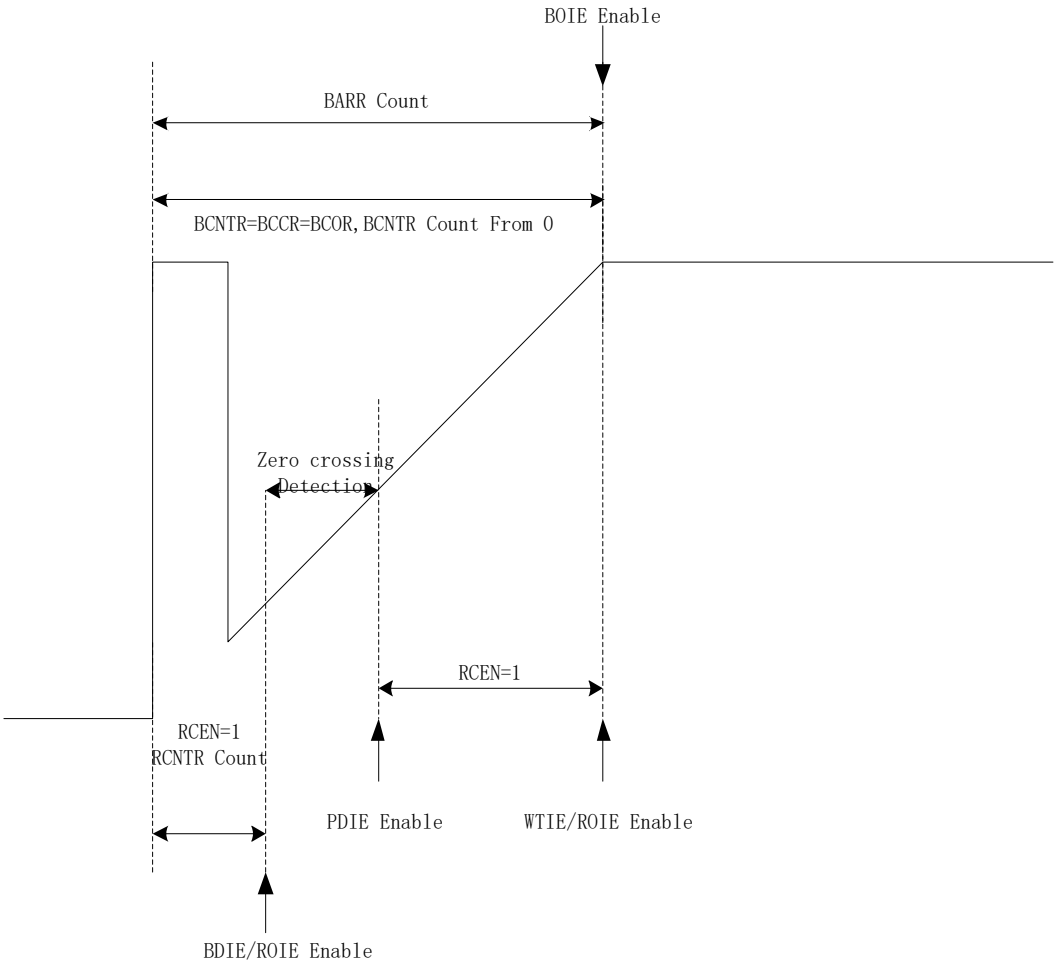


Figure 16-13 Operating Principle of BLDC Control Based on Timer 1

16.2.2.1 60 Degree Reference

TIM1__BCCR records the last 60 degree count. Setting TIM1_CR2[BRS] to 0 or 1 select the counting method, which is between two commutations or two BEMF zero-crossing points.

The value of TIM1_BCOR is already filtered, and regard as 60 degree reference. TIM1_BCOR is 1/2/4/8 periods average value of TIM1__BCCR, select by TIM1_CR0[CFLT].

60 degree reference is used to calculate diode freewheeling angle, angle from zcp to commutation and forced commutation angle.

16.2.2.2 60 Degree Forced Commutation

When the motor operates steadily, it is general that zero-crossing point is about 30 degree lagging commutation. If there is no zcp signal within 60 degree, based on 60 degree reference, of commutation, forced commutation is necessary. Setting TIM1_CR0[FORC] to 1 enable forced commutation function. Commutation will implemented by hardware when there is no zcp signal within 60 degree of commutation, and basic timer overflow interrupt flag BOIF is set to 1 (Note: When FORC = 1, zcp signal is detected within 60 degree of commutation, if TIMA1_BCNTR > TIM1__BARR, BOIF will not set to 0). Setting TIM1_CR0[FORC] to 0, if TIM1__BCNTR > TIM1__BARR, BOIF is set to 1. Users can commutate by software based on BOIF and PDIF.

16.2.2.3 Diode Freewheeling

After commutation, since the previous conducting phase becomes a floating phase, the energy in inductance of this phase is released to power supply or ground through a freewheeling diode. During this processing, the comparator output signal will be distorted, and Timer 1 will not sample these signal. After diode freewheeling, BDIF is set to 1.

Diode freewheeling time is set by TIM1_CR1[BSEL]. Diode freewheeling angle = 60 × BSEL / 128.

16.2.2.4 Angle between zcp and Commutation

When zcp is detected after commutation, the next commutation will be implemented by hardware based on the predefined time from zcp and commutation, and the Writing Sequence interrupt flag WTIF is set to 1.

The time from zcp to commutation can be set by TIM1_CR2[CSEL]. Commutation angle = 60 × CSEL / 128.

16.2.2.5 Cycle-by-cycle Current Limiting

Refer to Section 30.1.1.2.

16.3 Timer 1 Register

16.3.1 TIM1_CR0 (0x4068)

Table 16-2 TIM1_CR0 (0x4068)

Bit	7	6	5	4	3	2	1	0
Name	T1RWEN	T1CFLT		T1FORC	T1OPS		T1BCEN	T1RCEN
Type	R/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	Function
[7]	T1RWEN	T1RCEN is enable to be written under T1RWEN = 1. The read of this bit is always 0
[6:5]	T1CFLT	60 degree reference filtering selection: The average of X times 60 degree is used as the reference. 00: 1 times 60 degree 01: 2 times 60 degree 10: 4 times 60 degree 11: 8 times 60 degree
[4]	T1FORC	60 degree forced commutation enable Note: If zero crossing has been detected, forced commutation will not be implemented regardless of this bit. 0: Disable 1: Enable
[3:2]	T1OPS	Data transmission method selection These bits are used to select the transmission method of TIM1_DBRx to the DRV_CMR. 00: Writing 1 to UPD by software or writing TIM1_CR4 01: The overflow event of reload timer for commutation 10: Position detection input 11: The overflow event of reload timer for commutation
[1]	T1BCEN	Basic timer counter enable 0: Disable 1: Enable
[0]	T1RCEN	Reload timer counter enable T1RCEN is enable to be written under T1RWEN = 1. The T1RCEN is set 1 by hardware when position detection event or writing sequence event is triggered. After overflow event of reload timer, T1RCEN is cleared by hardware. 0: Disable 1: Enable

16.3.2 TIM1_CR1 (0x4069)

Table 16-3 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	Function
[7]	T1BAPE	TIM1__BARR auto-load enable After enabling, when the basic timer is reset due to a position detection event or a writing sequence event, write 60 degree reference value to TIM1__BARR. (Used to implementing forced 60 degree commutation when zcp is not detected.) 0: Disable 1: Enable
[6:0]	BSEL	Diode freewheeling angle selection Angle (time) of diode freewheeling after commutation. During this predefined time, the comparator output is not detected. Diode freewheeling angle = 60 × BSEL / 128

16.3.3 TIM1_CR2 (0x406A)

Table 16-4 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	Function
[7]	T1BRS	Reset source of basic timer selection 0: Writing sequence event and 60 degree forced commutation event 1: Position detection event
[6:0]	CSEL	Commutation angle selection After position detection event triggered, the angle of commutation is delayed by the value of CSEL. delay angle = 60 × CSEL / 128

16.3.4 TIM1_CR3 (0x406B)

Table 16-5 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	R/W
Reset	0	0	0	0	0	1	0	0

Bit	Name	Function
[7]	RSV	Reserved
[6:4]	T1PSC	Timer 1 clock division selection. The basic and reload timer count clock is divided from MCU clock. These bits are used as divisor for the frequency division and MCU clock is 24MHZ(41.67ns) 000:0x1(24MHz) 001:0x2(12MHz) 010:0x4 (6MHz) 011:0x8(3MHz) 100:0x10(1.5MHz) 101:0x20(750kHz) 110:0x40 (375kHz) 111:0x80(187.5kHz)
[3:2]	T1TIS	Input source (TI0 / TI1 / TI2) selection 00: The input is selected as (P1.4 / P1.6 / P2.1) or (P0.2 / P3.7 / P3.6) according to CMP_CR1 [7] 01: The input is Output comparator (CMP0/CMP1/CMP2) 1x: Reserved
[1:0]	T1INM	Filtering width of TI0/TI1/TI2 input selection The data will be not sampled when it is less than the value set by these bits. 00: No filtering 01: 8 system clock cycles 10: 32 system clock cycles 11: 64 system clock cycles

16.3.5 TIM1_CR4 (0x406C)

Table 16-6 TIM1_CR4 (0x406C)

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

Bit	Name	Function
[7:3]	RSV	Reserved
[2:0]	T1CST	Predefined commutation state T1CST is the predefined state, which maps to TIM1_DBR1~7. When T1CST is 001~111 states, timer1 will enable comparator 0/1/2 reasonably according to the CPE in the corresponding state. T1CST will plus 1 when writing sequence event is generated.

CST	TIM1_DBRx	CST	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)

Table 16-7 TIM1_IER (0x406D)

Bit	7	6	5	4	3	2	1	0
Name	T1UPD	RSV	RSV	T1BOIE	T1RUIE	T1WTIE	T1PDIE	T1BDIE
Type	W	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	T1UPD	When OPS = 00, writing 1 to T1UPD trigger data transmission. This bit is write-only, and will be cleared by hardware after writing.
[6]	RSV	Reserved
[5]	RSV	Reserved
[4]	T1BOIE	Basic timer overflow interrupt enable 0: Disable 1: Enable
[3]	T1ROIE	Reload timer overflow interrupt enable: 0: Disable 1: Enable
[2]	T1WTIE	Writing sequence interrupt enable: 0: Disable 1: Enable
[1]	T1PDIE	Position detection interrupt enable: 0: Disable 1: Enable
[0]	T1BDIE	Diode freewheeling end interrupt enable 0: Disable 1: Enable

16.3.7 TIM1_SR (0x406E)

Table 16-8 TIM1_SR (0x406E)

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

Bit	Name	Function
[7:6]	RSV	Reserved
[5]	RSV	Reserved
[4]	T1BOIF	Basic timer overflow interrupt flag When the value of TIM1__BCNTR equals to TIM1__BARR, overflow event is generated and TIM1_VCNTR is cleared. This bit is set to 1 by hardware and cleared by software.

		Note: To clear TIM1__BCNTR in the interrupt, writing UPD or TIM1_CR4 when TIM1_CR2[T1BRS] = 0. 0: No event 1: An overflow event
[3]	T1ROIF	Reload timer overflow interrupt flag When the value of TIM1__BCNTR equals to TIM1__BARR, overflow event is triggered and TIM1__RCNTR is cleared. This bit is set to 1 by hardware and cleared by software. 0: No event 1: An overflow event
[2]	T1WTIF	Writing sequence interrupt flag When the value of TIM1_DBRH / TIM1_DBRL is updated to TIM1_DRH / TIM1_DRL. This bit is set to 1 by hardware and cleared by software. Note: When OPS = 00, writing WTIF to 1 by software will generate a writing sequence event. 0: No event 1: Writing sequence event
[1]	T1PDIF	Position detection interrupt flag When the inputs (TI2, TI1, TI0) are the same as TIM1_DBRx [CPE], a position detection interrupt event is triggered. This bit is set to 1 by hardware and cleared to 0 by software. 0: No event 1: Position detection event
[0]	T1BDIF	Diode freewheeling end interrupt flag After commutation, the sampling of comparator output will be disabled during diode freewheeling. This bit is set to 1 by hardware and cleared to 0 by software at the end of diode freewheeling. 0: No event 1: Diode freewheeling event

16.3.8 TIM1_BCOR (0x4070, 0x4071)

Table 16-9 TIM1_BCORH (0x4070)

Bit	7	6	5	4	3	2	1	0
Name	TIM1_BCORH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 16-10 TIM1_BCORL (0x4071)

Bit	7	6	5	4	3	2	1	0
Name	TIM1_BCORL							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:0]	TIM1_BCOR	Filtering value of TIM1__BCCR (i.e., 60 degree reference)

16.3.9 TIM1_DBRx (x=1 ~ 7) (0x4074 + 2 × x, 0x4075 + 2 × x)

TIM1_DBRx (x =1~7) correspond to the data when CST = 1/2/3/4/5/6/7. The following uses TIM1_DBR1 as an example.

Table 16-11 TIM1_DBR1H (0x4074)

Bit	7	6	5	4	3	2	1	0
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
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

Table 16-12 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

<table>
<tr><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>[15]</td><td>RSV</td><td>Reserved</td></tr>
<tr><td>[14:12]</td><td>T1CPE</td><td>TI0/TI1/TI2 input edge polarity selection and comparator enable
These bits are used to select the polarity of input edge for position detection and enable the corresponding comparator. Position detection is triggered based on the polarity of input edges configured by these bits.
<table>
<tr><th>CPE</th><th>Description</th><th>CPE</th><th>Description</th></tr>
<tr><td>000</td><td>0</td><td>100</td><td>Detect the falling edge of U phase, and enable the comparator corresponding to U phase</td></tr>
<tr><td>001</td><td>Detect the rising edge of U phase, and enable the comparator corresponding to U phase</td><td>101</td><td>Detect the rising edge of W phase, and enable the comparator corresponding to W phase</td></tr>
<tr><td>010</td><td>Detect the falling edge of W phase, and enable the comparator corresponding to W phase</td><td>110</td><td>Detect the falling edge of V phase, and enable the comparator corresponding to V phase</td></tr>
<tr><td>011</td><td>Detect the rising edge of V phase, and enable the comparator corresponding</td><td>111</td><td>Detect three-phase both edge, and enable three-phase comparator</td></tr>
</table></td></tr>
</table>

		to V phase
[11]	T1WHP	W phase high side output polarity 0: High voltage level drive 1: Low voltage level drive
[10]	T1WLP	W phase low side output polarity 0: High voltage level drive 1: Low voltage level drive
[9]	T1VHP	V phase high side output polarity 0: High voltage level drive 1: Low voltage level drive
[8]	T1VLP	V phase low side output polarity 0: High voltage level drive 1: Low voltage level drive
[7]	T1UHP	U phase high side output polarity 0: High voltage level drive 1: Low voltage level drive
[6]	T1ULP	U phase low side output polarity 0: High voltage level drive 1: Low voltage level drive
[5]	T1WHE	W phase high side output enable 0: Disable 1: Enable Note: When high side and low side of W phase are enabled, deadtime will be added to the output automatically.
[4]	T1WLE	W phase low side output enable 0: Disable 1: Enable
[3]	T1VHE	V phase high side output enable 0: Disable 1: Enable Note: When high side and low side of V phase are enabled, deadtime will be added to the output automatically.
[2]	T1VLE	V phase low side output enable 0: Disable 1: Enable
[1]	T1UHE	U phase high side output enable 0: Disable 1: Enable Note: When high side and low side of U phase are enabled, deadtime will be added to the output automatically.
[0]	T1ULE	U phase low side output enable 0: Disable 1: Enable

16.3.10 TIM1__BCNTR (0x4082, 0x4083)

Table 16-13 TIM1__BCNTRH (0x4082)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__BCNTRH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 16-14 TIM1__BCNTRL (0x4083)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__BCNTRL							
Type	R/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	Function
[15:0]	TIM1__BCNTR	The count of basic timer, is used to count 60 degree commutation time. Note: TIM1__BCNTR selects the reset source based on TIM1_CR2[[T1BRS]. TIM1__BCNTR will not be cleared when an overflow event is triggered.

16.3.11 TIM1__BCCR (0x4084, 0x4085)

Table 16-15 TIM1__BCCRH (0x4084)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__BCCRH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 16-16 TIM1__BCCRL (0x4085)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__BCCRL							
Type	R/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	Function
[15:0]	TIM1__BCCR	The count of capture basic timer When a position detection event or a writing sequence event is generated, the count of basic timer is sent to BCCR and cleared by hardware.

16.3.12 TIM1__BARR (0x4086, 0x4087)

Table 16-17 TIM1__BARRH (0x4086)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__BARRH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 16-18 TIM1__BARRL (0x4087)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__BARRL							
Type	R/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	Function
[15:0]	TIM1__BARR	Auto-reload value of basic timer When the count of basic timer is equal to BARR, an overflow interrupt event is generated and the value of counter is cleared.

16.3.13 TIM1__RARR (0x4088, 0x4089)

Table 16-19 TIM1__RARRH (0x4088)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__RARRH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 16-20 TIM1__RARRL (0x4089)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__RARRL							
Type	R/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	Function
[15:0]	TIM1__RARR	Auto-reload value of reload timer When the count of reload timer is equal to RARR, an overflow interrupt event is generated and the value of counter is cleared.

16.3.14 TIM1__RCNTR (0x408A, 0x408B)

Table 16-21 TIM1__RCNTRH (0x408A)

Bit	7	6	5	4	3	2	1	0

Name	TIM1__RCNTRH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 16-22 TIM1__RCNTRL (0x408B)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__RCNTRL							
Type	R/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	Function
[15:0]	TIM1__RCNTR	Count of reload timer, used for counting time of diode freewheeling and zcp to commutation.

16.3.15 TIM1__ITRIP (0x4098, 0x4099)

Table 16-23 TIM1__ITRIPH (0x4098)

Bit	7	6	5	4	3	2	1	0
Name	TIM1__ITRIP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Table 16-24 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	Function
[15:0]	TIM1__ITRIP	Filtered BUS current BUS current is sampled by hard core, and TIM1__ITRIP is the filtered value. ADC channel-4 is used by default. Range: (0,32767)

17 TIM2

17.1 TIM2 Operating Descriptions

Timer 2 includes output, input timer, input counter and QEP&RSD modes:

1. Output mode: Generate PWM output waveform
2. Input timer mode: Detect input PWM “H” or “L” level time, calculate the duty cycle of PWM
3. Input counter mode: Detect the time of a predefined PWM number
4. QEP&RSD mode: Quadrature encoder mode. Forward and reverse rotation detection mode

Timer 2 includes:

1. 3 bit programmable frequency divider for count clock frequency division of the basic counter
2. A 16 bit up basic counter. Clock source is the clock controller output
3. A 16 bit up-down counter dedicated to input counter mode and QEP&RSD mode. Clock source is effective edge of external input
4. Input filter module
5. Edge detection module
6. Output mode for generating PWM
7. Interrupt event

17.1.1 Timer 2 Clock Controller

The clock controller is used to generate the basic timer count clock source, and the count clock source frequency is divided by prescaler. It is an 8bit counter with 3 bit PSC controller, and can select 8 kinds of frequency division coefficients. The clock source come from the internal clock. As there is no buffer in the control register, frequency division coefficient can be updated immediately. Therefore, the coefficients in the basic timer should be updated when they are in sleeping state.

The frequency of the counter can be calculated as:

If MCU clock is 24MHz (41.67ns), $f_{CK\_CNT} = f_{CK\_PSC} / T2PSC$

Table 17-1 The Clock Frequency Corresponding to Different T2PSC Value

T2PSC	Coefficient	CLK(Hz)
000	0x1	24M
001	0x2	12M
010	0x4	6M
011	0x8	3M
100	0x10	1.5M
101	0x20	750K
110	0x40	375K
111	0x80	187.5K

17.1.2 TIM2__CNTR: Its Reading, Writing and Counting

Timer 2 counter works only when T2CEN = 1. There is no buffer in writing to the counter and the value of TIM2__CNTR will be updated immediately. Therefore, it is recommended to write a new value into the counter while it is disabled. For reading, software must read the MS byte first, and then LS byte value is buffered by hardware. This buffered value remains unchanged until the read of 16 bit sequence is completed.

17.1.3 Output Mode

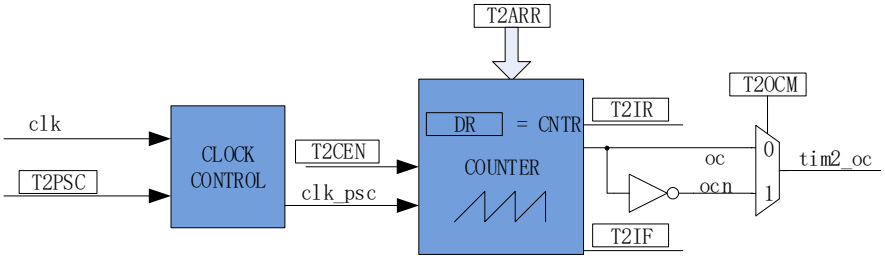


Figure 17-1 Block Diagram of Output Mode

Output mode is configurable by TIM2_CR0[T2_OCM]. Output signal and corresponding interrupt are generated by the comparing result between TIM2__CNTR and TIM2__DR.

17.1.3.1 TIM2__ARR/TIM2__DR: Its Reading and Writing

In output mode, TIM2__ARR/TIM2__DR includes preload register and shadow register. When data is written to TIM2__ARR/TIM2__DR, it is kept in the preload register. Data will be transferred into shadow register when an overflow event T2IF is triggered or counter is disabled (T2CEN = 0).

16 bit values are loaded in the TIM2__ARR/TIM2__DR register through preload registers. This must be performed by two write instructions, one for each byte. The MS byte must be written first. The shadow register update is blocked as soon as the MS byte has been written, until the LS byte has been written.

17.1.3.2 High/Low Voltage Level Output Mode

Setting TIM2_CR0[T2OCM] to 0 and TIM2__DR = TIM2__ARR, output signal TIM2_OC is low voltage level. Setting TIM2_CR0[T2OCM] to 1 and TIM2__DR = TIM2__ARR, output signal TIM2_OC is high voltage level.

It should be noted that configuring TIM2__DR = 0 will generate a clock period pulse.

17.1.3.3 PWM Mode

The period of PWM output is configurable by TIM2__ARR. Duty cycle is calculated by TIM2__DR, duty cycle = 100% × TIM2__DR / TIM2__ARR. If TIM2_CR0[T2_OCM] is set to 0, when TIM2__CNTR ≤ TIM2__DR, PWM output is low voltage level, when TIM2__CNTR > TIM2__DR, PWM output is high voltage level. If TIM2_CR0[T2_OCM] is set to 1, when TIM2__CNTR ≤ TIM2__DR, PWM output is high voltage level, when TIM2__CNTR > TIM2__DR, PWM output is low voltage level.

17.1.3.4 Interrupt Event

a) When TIM2__CNTR = TIM2__DR, a compare event is triggered and interrupt flag TIM2_CR1[T2IR] is set to 1. The count of counter continues going on.

b) When TIM2__CNTR = TIM2__ARR, an overflow event is triggered and interrupt flag TIM2_CR1[T2IF] is set to 1. The value of counter is cleared. If TIM2_CR0[T2OPM] = 1, the counter is stopped and TIM2_CR0[T2OPM] = 0, the counter is restarted.

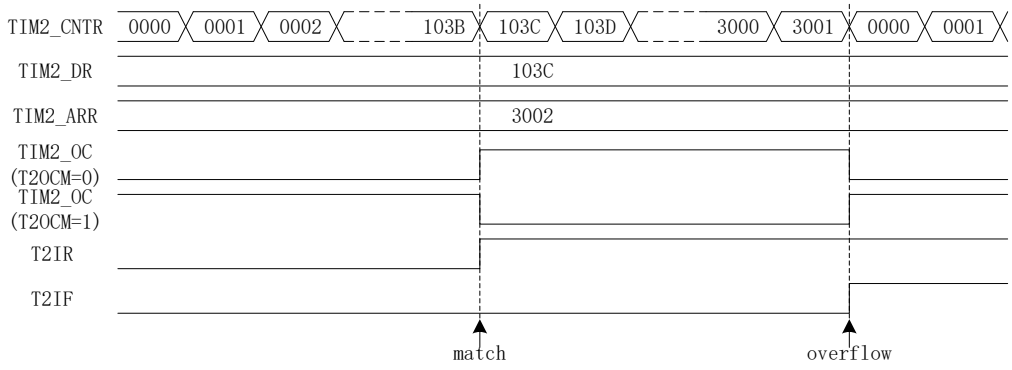


Figure 17-2 Output Waveform of Output Mode

17.1.4 Filtering and Edge Detection of Input Signals

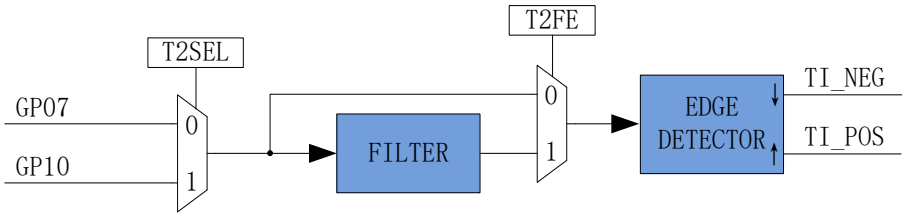


Figure 17-3 Block Diagram of Input Signal Filtering and Edge Detection

Timer input signal TI comes from P0.7 or P1.0. Input is optional for noise filtering by the

configuration of PH_SEL[T2SEL]. The rising and falling edge of input are detected and used for next module.

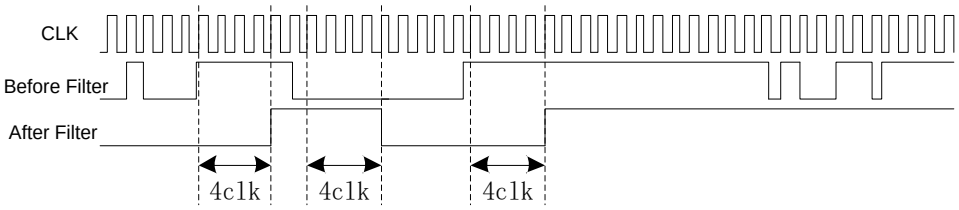


Figure 17-4 Filter Module Sequence Diagram

The filtering width of input noise filter is 4 clock periods. Set TIM2_CR1[T2_FE] to 1 to enable the filter function. After filtering, signals will delay 4~5 clock cycles of the one unfiltered.

17.1.5 Input Timer Mode

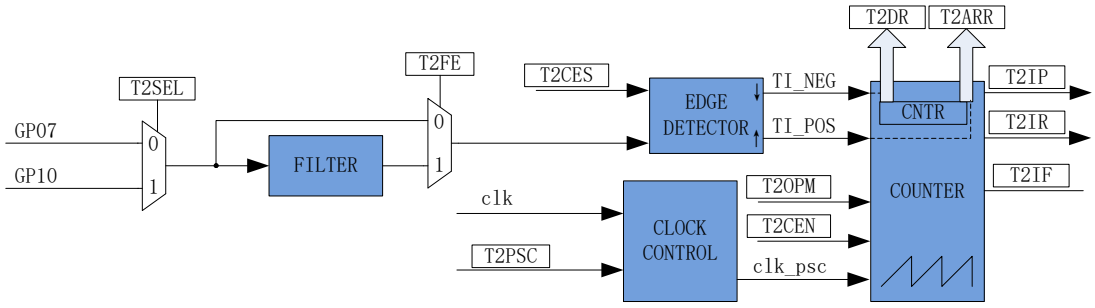


Figure 17-5 Block Diagram of Input Timer Mode

Input timer mode detects duty cycle and period of PWM signal. If T2CES is set to 0, two adjacent rising edges are selected as one period and pulse width is from rising edge to falling edge. If T2CES is set to 1, two adjacent falling edges are selected as one period and pulse width is from falling edge to rising edge. The count value TIM2__CNTR is recorded in TIM2 _ARR and TIM2__DR. The filtering of input signal is optional.

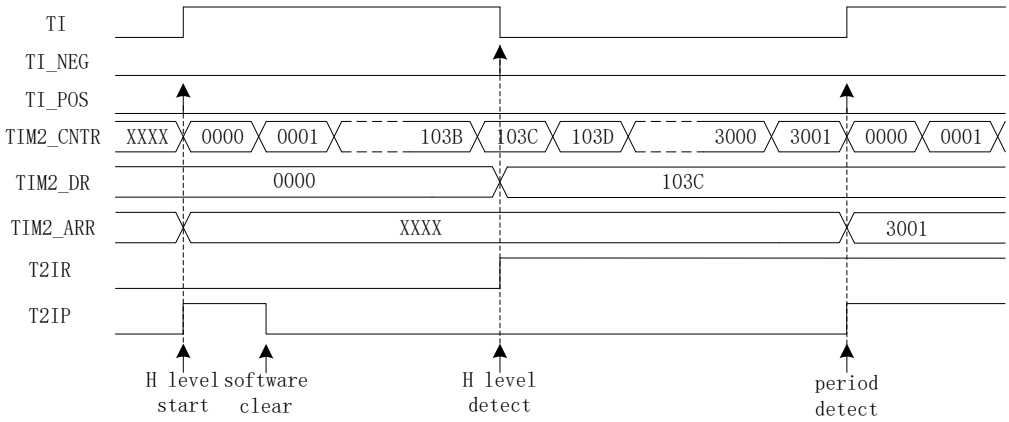


Figure 17-6 Input Timer Mode Sequence Diagram (T2CES = 0)

Providing T2CES = 0 as an example. Setting TIM2_CR1[T2CEN] = 1 enable counter for starting the counting. When first rising edge (falling edge invalid) of the input is detected, TIM2__CNTR is cleared and then restarts counting.

When falling edge is detected, TIM2__DR records the value of TIM2__CNTR. At the same time, TIM2_CR1[T2IR] is set to 1. TIM2__CNTR then continues the up-counting.

When second rising edge is detected, and a PWM period of input is detected. TIMx__ARR records the value of TIM2__CNTR. Then interrupt flag TIM2_CR1[T2IP] is set to 1, and TIM2__CNTR is cleared. TIM2_CR0[T2OPM] determines whether the counting should be continued, or not. If T2OPM = 1, the counting is stopped, and T2OPM = 0, the counting is restarted.

When TIM2__CNTR is 0xFFFF and the second rising edge has not been detected, an overflow event is induced. The Interrupt flag TIM2_CR1[T2IF] is set to 1. TIM2__CNTR is cleared and recount.

17.1.6 Input Counter Mode

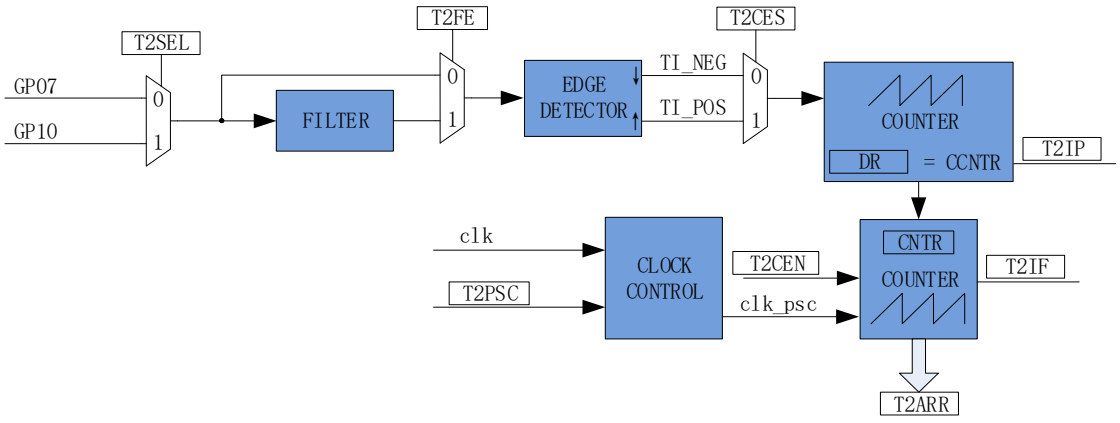


Figure 17-7 Block diagram of Input Counter Mode

In input counter mode, TIM2__DR includes preload register and shadow register. When data is written to TIM2__DR, it is kept in the preload register. Data will be transferred into shadow register when a compare event T2IP/an overflow event T2IF is triggered or counter is disabled (T2CEN = 0).16 bit values are loaded in the TIM2__DR register through preload registers. This must be performed by two write instructions, one for each byte. The MS byte must be written first. The shadow register update is blocked as soon as the MS byte has been written, until the LS byte has been written.

Input count mode detects the time of a predefined number of PWM inputted. TIM2__ARR is recorded with the value of TIM2__CNTR. Input signal can be filtered. Setting TIM2_CR0[T2_CES] = 1, the rising edge of input signal is used as the effective edge and TIM2_CR0[T2_CES] = 0, the falling edge of input signal is used as the effective edge.

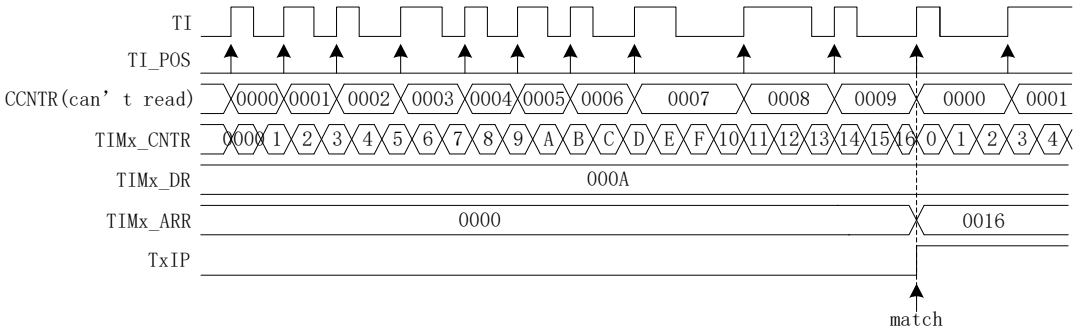


Figure 17-8 Sequence Diagram of Input Counter Mode

Configuring TIM2_CR1[T2CEN] = 1 enable the counter to do up-counting. When timer detects the first effective edge of input, TIM2__CNTR is cleared and the counter continues going on.

When timer detects effective edge, CCNTR value is added with 1. TIM2__DR sets the target value of PWM number. TIM2__ARR records the value of TIM2__CNTR when CCNTR = TIM2__DR. At the same time, TIM2_CR1[T2IP] is set to 1. The value of TIM2__CNTR and CCNTR are cleared and recount.

If the number of detected PWM is less than TIM2__DR and TIM2__CNTR is 0xFFFF, the overflow event is generated. TIM2_CR1[T2IF] is set to 1. TIM2__CNTR recount with the initial value of 0 and CCNTR continues the previous count.

17.1.7 QEP & RSD Mode

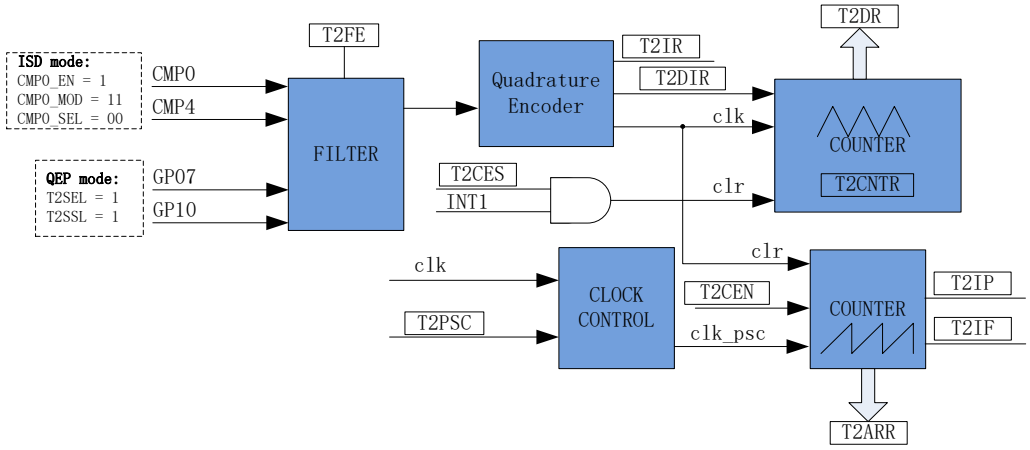


Figure 17-9 Block Diagram of QEP & RSD Mode

QEP&RSD mode obtains the relative position, direction and velocity information by detecting the quadrature inputs. P0.7 and P1.0 (QEP mode) or CMP0 and CMP4 (RSD mode) are used as inputs and sent to the quadrature decoder module, after filtering, to obtain count and direction T2DIR. A change of direction will generate the interrupt flag T2IR.

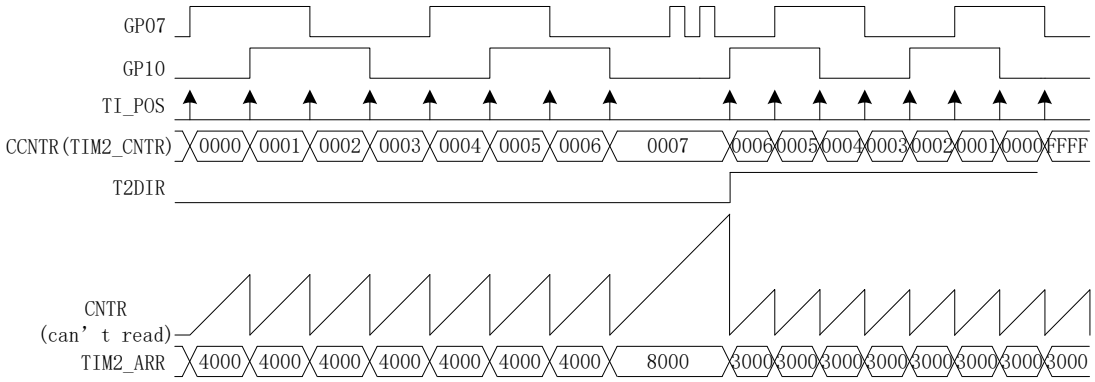


Figure 17-10 Sequence Diagram of QEP & RSD Mode

The dedicated counter is an up-down counter, and the count clock is the effective count edge of quadrature encoder module output. Providing T2DIR = 0, the direction is forward. The counter plus 1 when the effective edge comes. Providing T2DIR = 1, the direction is reverse. The counter minus 1 when the effective edge comes. Dedicated counter can be cleared by external interrupt INT1. For this function, connect the Z signal of encoder to any port of external interrupt 1, enable external interrupt 1 and set T2CES to 1. When external interrupt 1 is triggered, the count of dedicated counter is recorded in TIM2__DR and cleared. The dedicated counter is cleared to 0 when it counts from 0 to 65535 and set to 65535 after decreasing from 65535 to 0. Software read of TIM2__CNTR can obtain the value of the dedicated counter.

Basic counter is an up-counter, and the count clock can be divided. Basic counter is used to record the

time of two effective edges. When an effective edge comes, the current count of basic counter is updated to TIM2__ARR and cleared. Interrupt flag T2IP is generated. When the value of basic counter is 0xFFFF, an overflows event and the interrupt flag T2IF are generated.

17.1.7.1 Comparator Sampling in RSD Mode

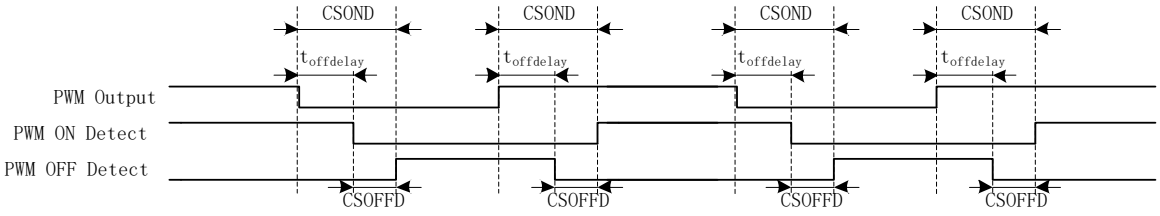


Figure 17-11 Diagram of Sampling Time

For RSD mode, the Output two comparators are used to determine motor direction. Since the Output comparator is distorted by the commutation of power devices. The sampling mode is configurable by writing CMP_CR3[SAMSEL]. The sampling interval can be configured by CSOFFD and CSOND.

There is a delay from the PWM output effect to comparator. And the time is mainly decided by the followings: the resistance of driver, turn on delay time and turn off delay time of power device, input delay of comparator and hysteresis configuration. The sampling of comparator will be disabled due to this effect, and the time is configurable by CSOFFD and CSOND. The disable time of CMP0, CMP1 and CMP2 will be delayed for off delay = CSOND – CSOFFD.

e.g., providing the delay from PWM output effect to comparator is 2μs and the time of effect is 1μs, the CSOFFD and CSOND can be configured as

CSOFFD > 1μs = 1000ns / (41.67ns × 8) = 3

CSOND > (2 + 1) μs = 3000 / (41.67ns × 8) = 9

To measure the delay from PWM output effect to the comparator, the following steps could be implemented. Setting CMP_CR3[SAMSEL] = 00 disable sampling delay. Configure CMP_CR3[CMPSEL] to output the corresponding comparator. PWM output and comparator is enabled. Users rotate motor to update the comparator output and the delay between PWM output and comparator output is obtained.

To measuring the width of comparator output distortion, the above method could be used.

17.1.8 Stepper Mode

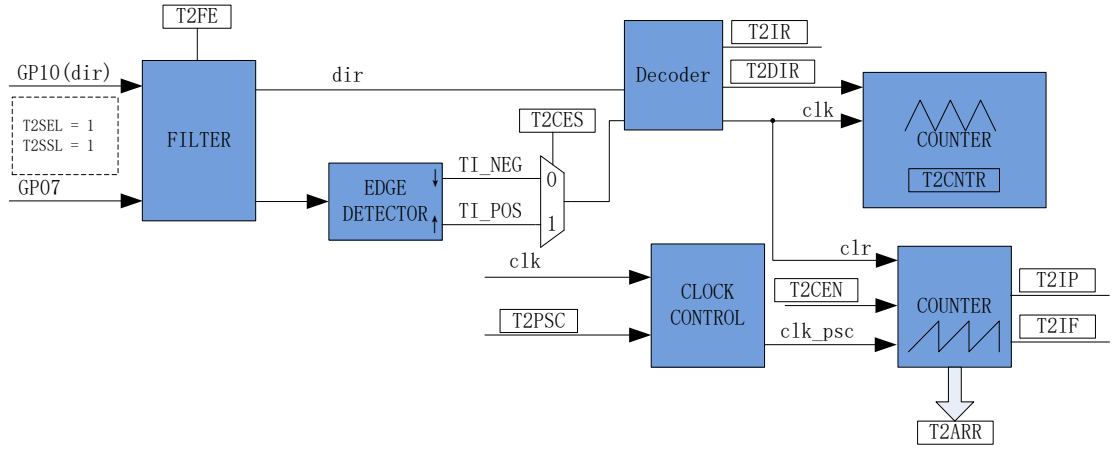


Figure 17-12 Block Diagram of Stepper Mode

For stepper mode, P1.0 is the input as direction signal and P0.7 is the input of count signal. Stepper mode obtains the relative position, direction and velocity information by detecting the above inputs. The rising or falling edge of P0.7 can be selected as an effective edge. RSD module obtains the count and direction by the filtered input. An interrupt flag T2IR will be triggered when the direction is changed. (Note: When P1.0 changes, T2DIR and T2IR will not act until the effective edge of P0.7 comes. T2DIR and T2IR will act immediately when external interrupt 1 is used.)

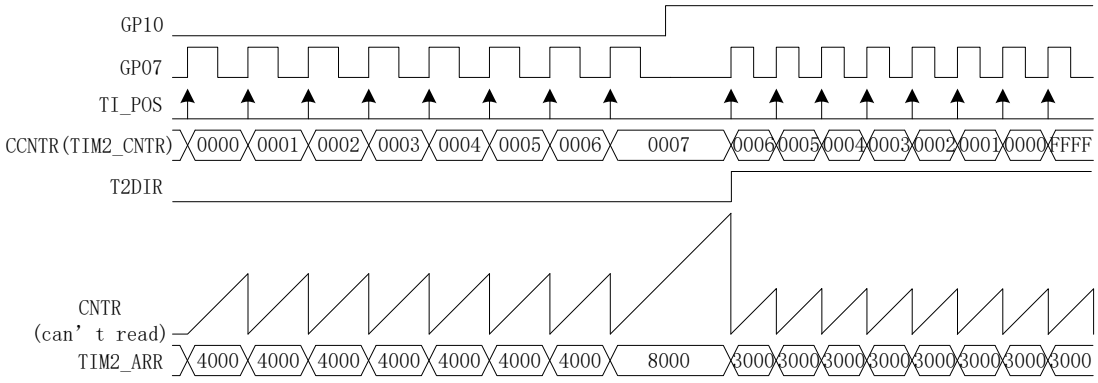


Figure 17-13 Sequence Diagram of Step Mode

The dedicated counter is an up-down counter, and the count clock is the effective count edge of RSD module output. Providing P1.0 is low voltage level (forward direction), T2DIR = 0 and the counter plus 1 when the effective edge comes. Providing P1.0 is high voltage level (reverse direction), T2DIR = 1 and the counter minus 1 when the effective edge comes. The dedicated counter is cleared to 0 when it counts from 0 to 65535 and set to 65535 after decreasing from 65535 to 0. Software read of TIM2__CNTR can obtain the value of the dedicated counter.

Basic counter is an up-counter, and the count clock can be divided. Basic counter is used to record the

time of two effective edges. When an effective edge comes, the current count of basic counter is updated to TIM2__ARR and cleared. Interrupt flag T2IP is generated. When the value of basic counter is 0xFFFF, an overflows event and the interrupt flag T2IF are generated.

17.2 TIM2 Register

17.2.1 TIM2_CR0 (0xA1)

Table 17-2 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	Function
[7:5]	T2PSC	Timer 2 clock division selection. The basic timer count clock is divided from MCU clock. These bits are used as divisor for the frequency division and MCU clock is 24MHZ (41.67ns). 000:0x1(24MHz) 001:0x2(12MHz) 010:0x4 (6MHz) 011:0x8(3MHz) 100:0x10(1.5MHz) 101:0x20(750kHz) 110:0x40 (375kHz) 111:0x80(187.5kHz)
[4]	T2OCM	Output mode: Compare mode selection 0: When TIM2__CNTR ≤ TIM2__DR, output 0; TIM2__CNTR > TIM2__DR, output 1. 1: When TIM2__CNTR ≤ TIM2__DR, output 1; TIM2__CNTR > TIM2__DR, output 0. Input counter mode: Invalid Input timer mode: Invalid QEP & RSD mode & stepper mode: Mode selection 0: QEP & RSD mode 1: Stepper mode
[3]	T2IRE	Output mode: Comparing interrupt enable Input timer mode: Pulse width detection interrupt enable Input counter mode: Invalid QEP & RSD mode & stepper mode: Direction change interrupt enable 0: Disable 1: Enable
[2]	T2CES	Input timer mode: Period and duty cycle edge selection 0: Two adjacent rising edges are one period, and the pulse width is from rising edge to falling edge (high-level pulse width) 1: Two adjacent falling edges are one period, and the pulse width is

		from falling edge to rising edge (low-level pulse width) Input counter mode & stepper mode: Effective edge selection 0: Falling edge 1: Rising edge QEP & RSD mode: External interrupt 1 (Z signal) clear pulse counter enable 0: Disable 1: Enable
[1:0]	T2MOD	Mode selection 00: Input timer mode 01: Output mode 10: Input counter mode 11: QEP & RSD mode & stepper mode

17.2.2 TIM2_CR1 (0xA9)

Table 17-3 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/W0	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	T2IR	Output mode: Compare flag This bit is set to 1 by hardware when TIM2__CNTR is equal to TIM2__DR. It is cleared by software. Input timer mode: Pulse width detection flag When timer detects a pulse width (i.e., duty cycle), this bit is set to 1 by hardware. It is cleared by software. Input counter mode: Invalid QEP & RSD mode & stepper mode: Direction change flag 0: No event 1: Event is generated
[6]	T2IP	Output mode: Invalid Input timer mode: PWM period detection flag When timer detects an entire PWM period, this bit is set to 1 by hardware. It is cleared by software. Input counter mode: Input PWM count matching flag When the number of input PWM is equal to the value of TIM2__DR, this bit is set to 1 by hardware. It is cleared by software. QEP & RSD mode & stepper mode: effective edge interrupt flag When the input edge is an effective edge, this bit is set to 1 by hardware. It is cleared by software. 0: No event

		1: Event is generated
[5]	T2IF	Output mode: Counter overflow flag When TIM2__CNTR is equal to TIM2__ARR, TIM2__CNTR is cleared and this bit is set to 1 by hardware. It is cleared by software. Input timer mode: Counter overflow flag When the timer has not detected a PWM period and the value of TIM2__CNTR is accumulated to 0xFFFF, an overflow event is triggered, and TIM2__CNTR is cleared. This bit is set to 1 by hardware and cleared by software. Input counter mode: Basic counter overflow flag When the number of input PWM has not reached TIM2__DR and TIM2__CNTR is accumulated to 0xFFFF, an overflow event is generated, and TIM2__CNTR is cleared. This bit is set to 1 by hardware and cleared by software. QEP & RSD mode & stepper mode: Basic counter overflow flag When the basic counter accumulates to 0xFFFF, an overflow event is generated and the basic counter is cleared to 0. This bit is set to 1 by hardware and cleared by software. 0: No event 1: Event is generated
[4]	T2IPE	Output mode: Invalid Input timer mode: PWM period detection interrupt enable Input counter mode: PWM count matching interrupt enable QEP & RSD mode & stepper mode: Effective edge interrupt enable 0: Disable 1: Enable
[3]	T2IFE	Output mode: Counter overflow interrupt enable Input timer mode: Counter overflow interrupt enable Input counter mode: Basic counter overflow interrupt enable QEP & RSD mode & stepper mode: Basic counter overflow enable 0: Disable 1: Enable
[2]	T2FE	Input noise filtering enable When the pulse width of input is less than 4 clock cycles, it will be filtered. If the MCU clock is 24MHZ (41.67ns), the filtering pulse width is 166.67ns. 0: Disable 1: Enable
[1]	T2DIR	Dedicated to QEP & RSD & stepper mode: Motor Direction 0: Forward 1: Reverse
[0]	T2CEN	Counter enable 0: Disable 1: Enable

17.2.3 PI_CR (0xF9)

Table 17-4 PI_CR (0xF9)

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

Bit	Name	Function
[7]	T2SS	TIM2 stepper motor mode input selection 0: P0.1 is the direction, P07 is the pulse count 1: P0.1 is the reverse pulse count, P07 is the positive pulse count
[6:3]	RSV	Reserved
[2:0]		Refer to Table 13-1

17.2.4 TIM2__CNTR (0xAA, 0xAB)

Table 17-5 TIM2__CNTRH (0xAB)

Bit	7	6	5	4	3	2	1	0
Name	TIM2__CNTRH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 17-6 TIM2__CNTRL (0xAA)

Bit	7	6	5	4	3	2	1	0
Name	TIM2__CNTRL							
Type	R/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	Function
[15:0]	TIM2__CNTR	Output mode/input timer mode/input counter mode: the value of the basic counter QEP&RSD mode & stepper mode: the value of the dedicated counter

17.2.5 TIM2__DR (0xAC, 0xAD)

Table 17-7 TIM2__DRH (0xAD)

Bit	7	6	5	4	3	2	1	0
Name	TIM2__DRH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 17-8 TIM2__DRL (0xAC)

Bit	7	6	5	4	3	2	1	0
Name	TIM2__DRL							
Type	R/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	Function
[15:0]	TIM2__DR	Output mode: compare matching value (by software) Input timer mode: Count of detected pulse width (by hardware) Input counter mode: The number of PWM (by software) QEP & RSD mode: The value of the dedicated counter when T2CES = 1 and external interrupt 1 (Z signal) arrives Stepping mode: Invalid

17.2.6 TIM2__ARR (0xAE, 0xAF)

Table 17-9 TIM2__ARRH (0xAF)

Bit	7	6	5	4	3	2	1	0
Name	TIM2__ARRH							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 17-10 TIM2__ARRL (0xAE)

Bit	7	6	5	4	3	2	1	0
Name	TIM2__ARRL							
Type	R/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	Function
[15:0]	TIM2__ARR	Output mode: reload value (by software) Input timer mode: Count of one PWM period (by hardware) Input counter mode: The value of basic counter when the input PWM count matches (by hardware) QEP & RSD mode & stepper mode: The value of basic counter when the input edge is detected as an effective edge (by hardware)

18 TIM3/TIM4

18.1 TIM3/TIM4 Operating Descriptions

Timer 3/ Timer 4 includes output, input timer modes.

1. Out mode: Generate output waveform (PWM, one pulse mode)
2. Input timer mode: Detect input PWM “H” or “L” level time, calculate the duty cycle of PWM

Timer 3/Timer 4 include

1. 3 bit Programmable frequency divider for count clock frequency division of the basic counter
2. A 16 bit up basic counter. Clock source is the clock controller output
3. Input filter module
4. Edge detection module
5. Output mode for generating PWM/one-pulse compare output
6. Interrupt event

18.1.1 Clock Controller

The clock controller is used to generate the basic timer count clock source, and the count clock source frequency is divided by prescaler. It is an 8 bit counter with 3 bit PSC controller, and can select 8 kinds of frequency division coefficients. The clock source come from the internal clock. As there is no buffer in the control register, frequency division coefficient can be updated immediately. Therefore, the coefficients in the basic timer should be updated when they are in sleeping state.

The frequency of the counter can be calculated as:

If MCU clock is 24MHz (41.67ns), $f_{CK\_CNT} = f_{CK\_PSC} / TxPSC$

Table 18-1 The Clock Frequency Corresponding to Different TxPSC Value

TxPSC	Coefficient	CLK(Hz)
000	0x1	24M
001	0x2	12M
010	0x4	6M
011	0x8	3M
100	0x10	1.5M
101	0x20	750K
110	0x40	375K
111	0x80	187.5K

18.1.2 TIMx__CNTR: Its Reading Writing and Counting

Timer x counter works only when TxCEN = 1. There is no buffer in writing to the counter and the value of TIMx__CNTR will be updated immediately. Therefore, it is recommended to write a new value into the counter while it is disabled. For reading, the LS byte value will not be buffered by hardware and

the reading of TIMx__CNTR should operate when counter is disabled.

18.1.3 Output Mode

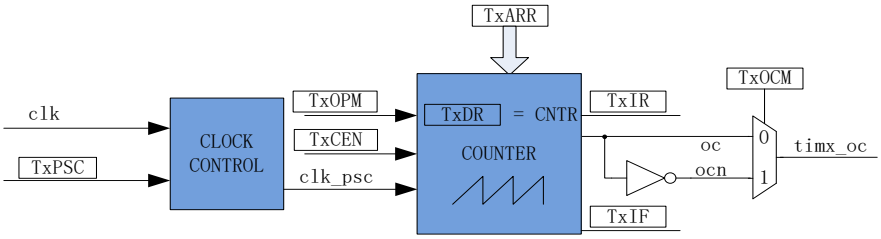


Figure 18-1 Block diagram of Output Mode

Output mode is configurable by TIMx_CR0[Tx_OCM]. Output signal and corresponding interrupt are generated by the comparing result between TIMx__CNTR and TIMx__DR.

18.1.3.1 High/Low Voltage Level Output Mode

Setting TIMx_CR0[TxOCM] to 0 and TIMx__DR = TIMx__ARR, output signal TIMx_OC is low voltage level. Setting TIMx_CR0[TxOCM] to 1 and TIMx__DR = TIMx__ARR, output signal TIMx_OC is high voltage level.

It should be noted that configuring TIMx__DR = 0 will generate a clock period pulse.

18.1.3.2 PWM Mode

The period of PWM output is configurable by TIMx__ARR. Duty cycle is calculated by TIMx__DR, duty cycle = 100% × TIMx__DR / TIMx__ARR. If TIMx_CR0[Tx_OCM] is set to 0 and TIMx__CNTR ≤ TIMx__DR, PWM output is low voltage level, and TIMx__CNTR > TIMx__DR, PWM output is high voltage level. If TIMx_CR0[Tx_OCM] is set to 1 and TIMx__CNTR ≤ TIMx__DR, PWM output is high voltage level, and TIMx__CNTR >TIMx__DR, PWM output is low voltage level.

18.1.3.3 Interrupt Event

a) When TIMx__CNTR = TIMx__DR, a compare event is triggered and interrupt flag TIMx_CR1[TxIR] is set to 1. The count of counter continues going on.

b) When TIMx__CNTR = TIMx__ARR, an overflow event is triggered and interrupt flag TIMx_CR1[TxIF] is set to 1. The value of counter is cleared. If TIMx_CR0[TxOPM] = 1, the counter is stopped; TIMx_CR0[TxOPM] = 0, the counter is restarted.

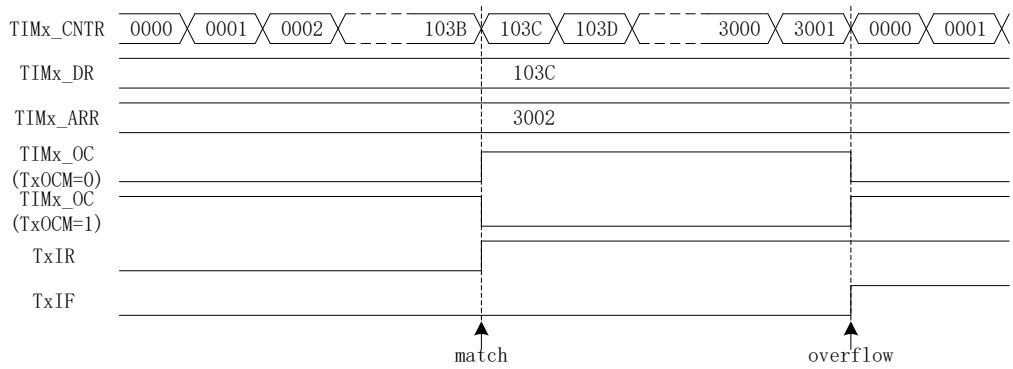


Figure 18-2 Output Waveform of Output Mode

18.1.4 Filtering and Edge Detection of Input Signals

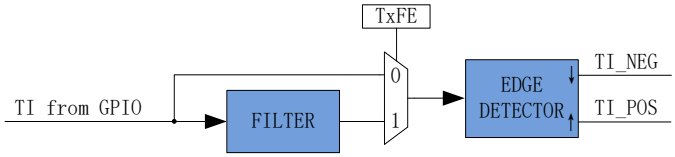


Figure 18-3 Block Diagram of Input Signal Filtering and Edge Detection

Timer 3/Timer 4 input signal TI comes from P1.1 or P0.1. Input is optional for noise filtering. The rising and falling edge of input are detected and used for next module.

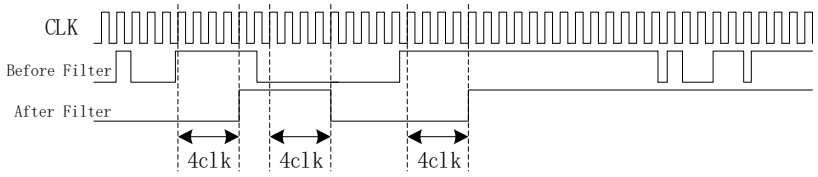


Figure 18-4 Sequence Diagram of the Filter Module

The filtering width of input noise filter is 4 clock periods. Set TIMx_CR1[Tx_FE] to 1 to enable the filter function. After filtering, signals will delay 4~5 clock cycles of the one unfiltered.

18.1.5 Input Timer Mode

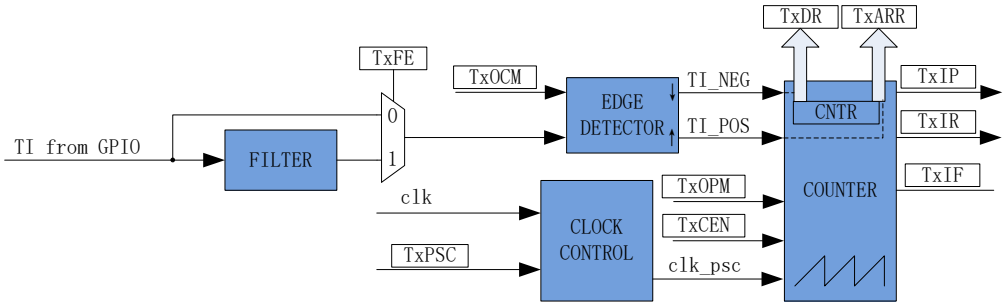


Figure 18-5 Block Diagram of Input Timer Mode

Input timer mode detects duty cycle and period of PWM signal. If TxOCM is set to 0, two adjacent rising edges are selected as one period and pulse width is from rising edge to falling edge. If TxOCM is set to 1, two adjacent falling edges are selected as one period and pulse width is from falling edge to rising edge. The count value TIMx__CNTR is recorded in TIMx _ARR and TIMx__DR. The filtering of input signal is optional.

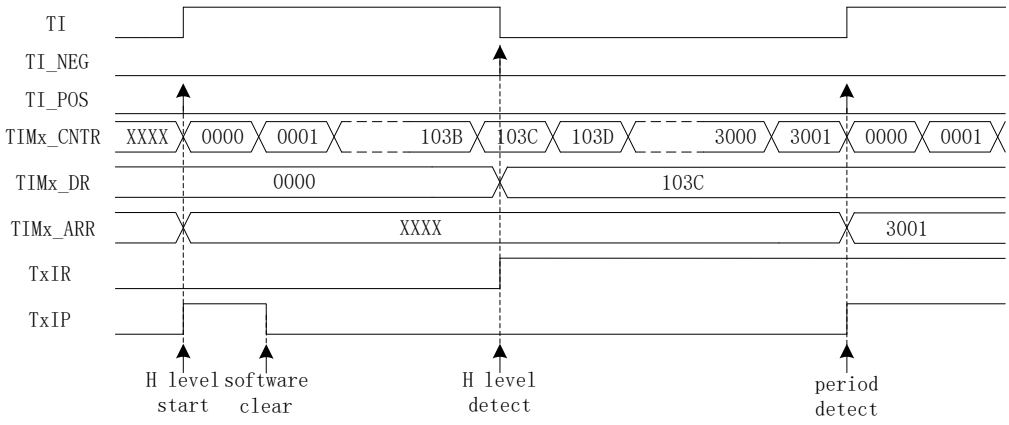


Figure 18-6 Timing diagram of input timer mode (TxOCM = 0)

Providing TxOCM = 0 as an example. Setting TIMx_CR1[TxCEN] = 1 enable counter for starting the counting.

When falling edge is detected, TIMx__DR records the value of TIMx__CNTR. At the same time, TIMx_CR1[TxIR] is set to 1. TIMx__CNTR then continues the up-counting.

When second rising edge is detected, and a PWM period of input is detected. TIMx__ARR records the value of TIMx__CNTR. Then interrupt flag TIMx_CR1[T2IP] is set to 1, and TIMx__CNTR is cleared. TIMx_CR0[TxOPM] determines whether the counting should be continued, or not. If TxOPM = 1, the counting is stopped; TxOPM = 0, the counting is restarted.

When TIMx__CNTR is 0xFFFF and the second rising edge has not been detected, an overflow event is induced. The Interrupt flag TIMx_CR1[T2IF] is set to 1. TIMx__CNTR is cleared and recount or not based on the configuration of TIMx_CR0[TxOPM]. If TxOPM = 1, the counting is stopped; TxOPM = 0, the counting is restarted.

18.2 TIM3 / TIM4 Register

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

Table 18-2 TIMx_CR0 (0x9C/0x9E)

Bit	7	6	5	4	3	2	1	0
Name	TPSC			TOCM	TIRE	RSV	TOPM	TMODE
Type	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:5]	TPSC	Timer 3/Timer 4 clock division selection. The basic timer count clock is divided from MCU clock. These bits are used as divisor for the frequency division and MCU clock is 24MHZ (41.67ns). 000:0x1(24MHz) 001:0x2(12MHz) 010:0x4 (6MHz) 011:0x8(3MHz) 100:0x10(1.5MHz) 101:0x20(750kHz) 110:0x40 (375kHz) 111:0x80(187.5kHz)
[4]	TOCM	Output mode: Compare mode selection 0: When TIMx__CNTR ≤ TIMx__DR, output 0; TIMx__CNTR > TIMx__DR, output 1. 1: When TIMx__CNTR ≤ TIMx__DR, output 1; TIMx__CNTR > TIMx__DR, output 0. Input timer mode: Period and duty cycle edge selection 0: Two adjacent rising edges are one period, and the pulse width is from rising edge to falling edge (high-level pulse width) 1: Two adjacent falling edges are one period, and the pulse width is from falling edge to rising edge (low-level pulse width)
[3]	TIRE	Output mode: Comparing interrupt enable Input timer mode: Pulse width detection interrupt enable 0: Disable 1: Enable
[2]	RSV	Reserved
[1]	TOPM	Single pulse width mode selection Output mode: Counter overflow event Input timer mode: PWM period detection or Counter overflow event 0: The counter will not stop when an event is generated.

		1: The counter will stop (clears TxCEN) when an event is generated.
[0]	TMODE	Mode selection 0: Input timer mode 1: Output mode

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

Table 18-3 TIMx_CR1 (0x9D/0x9F)

Bit	7	6	5	4	3	2	1	0
Name	TIR	TIP	TIF	TIDE	TIFE	TINM		TCEN
Type	R/W0	R/W0	R/W0	R/W0	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	TIR	Output mode: Compare flag This bit is set to 1 by hardware when TIMx__CNTR is equal to TIMx__DR. It is cleared by software. Input timer mode: Pulse width detection flag When timer detects a pulse width (i.e., duty cycle), this bit is set to 1 by hardware. It is cleared by software. 0: No event 1: Event is generated
[6]	TIP	Output mode: Invalid Input timer mode: PWM period detection flag When timer detects an entire PWM period, this bit is set to 1 by hardware. It is cleared by software. 0: No event 1: Event is generated
[5]	TIF	Output mode: Counter overflow flag When TIMx__CNTR is equal to TIMx__ARR, TIMx__CNTR is cleared and this bit is set to 1 by hardware. It is cleared by software. Input timer mode: Counter overflow flag When the timer has not detected a PWM period and the value of TIMx__CNTR is accumulated to 0xFFFF, an overflow event is triggered, and TIMx__CNTR is cleared. This bit is set to 1 by hardware and cleared by software. 0: No event 1: Event is generated
[4]	TIPE	Output mode: Invalid Input timer mode: PWM period detection interrupt enable 0: Disable 1: Enable
[3]	TIFE	Output mode: Counter overflow interrupt enable Input timer mode: Counter overflow interrupt enable 0: Disable

		1: Enable
[2:1]	TINM	Input filtering width selection. When the pulse width of input is less than the predefined value, it will be filtered. If the MCU clock is 24MHZ (41.67ns) 00: No filtering 01: 4 clock periods, 4 × 41.67ns 10: 8 clock periods, 8 × 41.67ns 11: 16 clock periods, 16 × 41.67ns
[0]	TCEN	Counter enable 0: Disable 1: Enable

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

Table 18-4 TIMx__CNTRH (0xA3/0x93)

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

Table 18-5 TIMx__CNTRL (0xA2/0x92)

Bit	7	6	5	4	3	2	1	0
Name	TIMx__CNTRL							
Type	R/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	Function
[15:0]	TIMx__CNTR	The value of the basic counter

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

Table 18-6 TIMx__DRH (0xA5/0x95)

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

Table 18-7 TIMx__DRL (0xA4/0x94)

Bit	7	6	5	4	3	2	1	0
Name	TIMx__DRL							
Type	R/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	Function
[15:0]	TIMx__DR	Output mode: compare matching value (by software) Input timer mode: Count of detected pulse width (by hardware)

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

Table 18-8 TIMx__ARRH (0xA7/0x97)

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

Table 18-9 TIMx__ARRL (0xA6/0x96)

Bit	7	6	5	4	3	2	1	0
Name	TIMx__ARRL							
Type	R/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	Function
[15:0]	TIMx__ARR	Output mode: reload value (by software) Input timer mode: Count of one PWM period (by hardware)

19 SYS_TICK

19.1 Operating Descriptions

SYS_TICK module is used to generate a fixed time interrupt. The interrupt period is configurable by setting the value of SYS_ARR. Set DRV_SR[STIE] to 1 to enable SYS_TICK. The interrupt entry is 10, shared with Timer 4.

19.2 Register

19.2.1 DRV_SR (0x4061)

Table 19-1DRV_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	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	SYSTIF	SYS TICK interrupt flag This bit is set to 1 by hardware and cleared by software. 0: No event 1: SYS TICK interrupt event
[6]	SYSTIE	SYS TICK interrupt enable 0: Disable 1: Enable
[5:0]		Please refer to Table 20-3

19.2.2 SYST_ARR (0x4064, 0x4065)

Table 19-2 SYS_ARRH (0x4064)

Bit	7	6	5	4	3	2	1	0
Name	SYS_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

Table 19-3 SYS_ARRL (0x4065)

Bit	7	6	5	4	3	2	1	0
Name	SYS_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	Function

[15:0]	SYS_ARR	Periodic value of SYS_TICK Setting this value determines the period of SYS_TICK interrupt. The default value is 1ms. SYS_TICK frequency = 24M / (SYS_ARR + 1) Range: (0,65535)

20 Driver

20.1 Operating Descriptions

20.1.1 Introduction

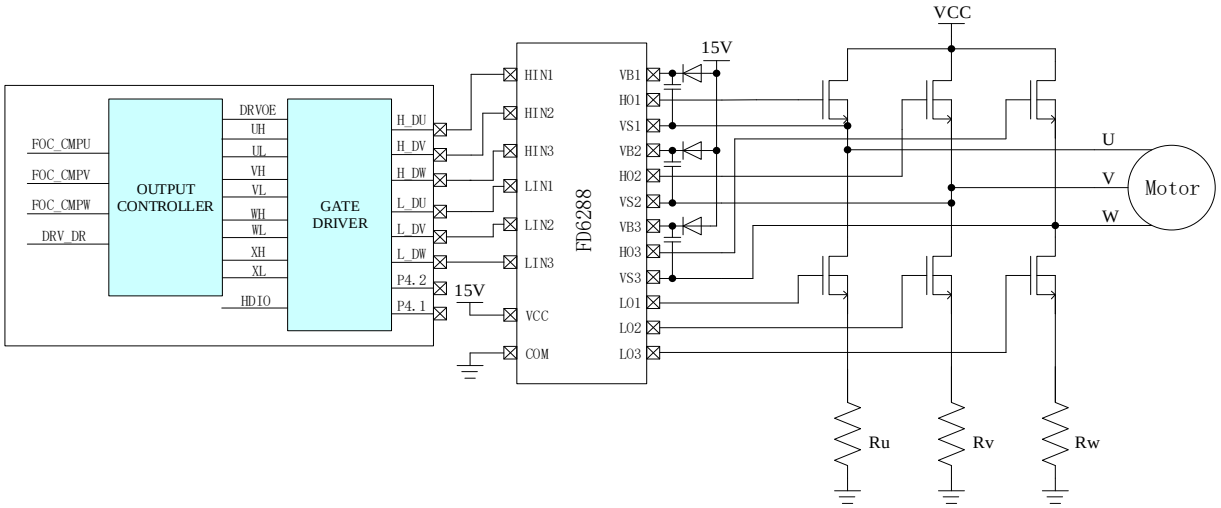


Figure 20-1 Block Diagram of FU6813 Driver

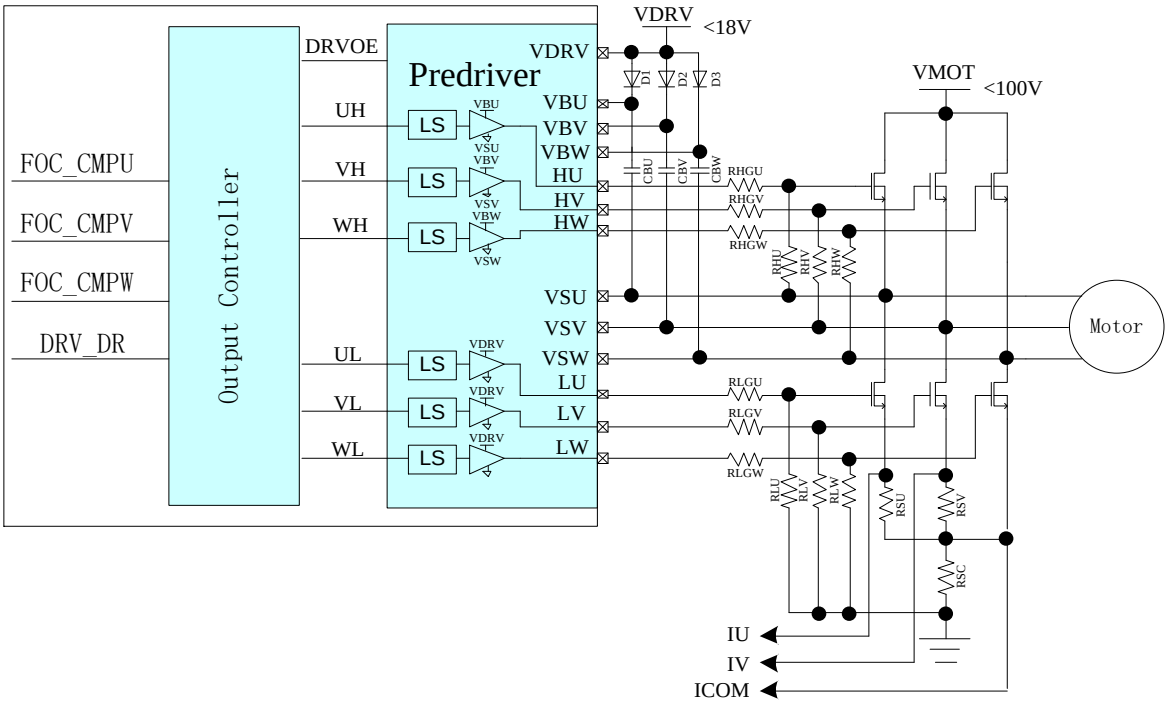


Figure 20-2 Block Diagram of FU6863 Driver

As shown above, FOC_CMPU/V/W is the 3-phase output by the FOC module and DRV_DR is the compare value set by software. Compared with these value, output control module output 4 pairs signal to Gate Driver (FU6813) or 3-phase signal U/V/W to Pre-driver (FU6863). Where, U/V/W are used for the

control of BLDC and U/V/W/X are used in the application of stepper motor.

20.1.2 Output Control Module

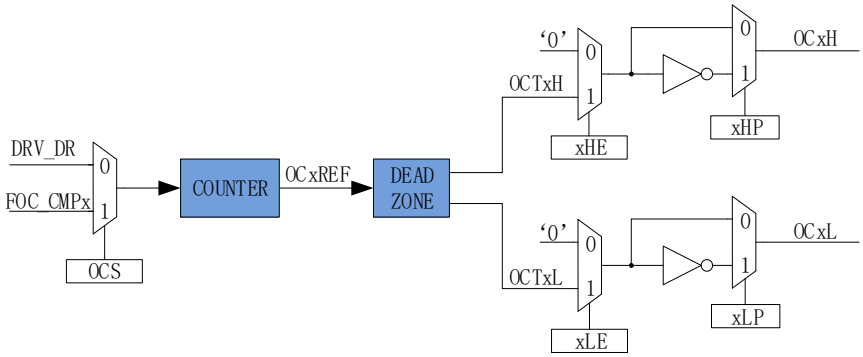


Figure 20-3 Front-End Block Diagram of Output Control Module

Before configuring the Driver module, DRV_CR[MESEL] must be set to 1 (i.e., the motor engine works in FOC/SVPWM/SPWM mode). Otherwise, the motor engine works in BLDC mode.

When OCS is set to 0, the compare value of PWM comes from DRV_DR. OCTxH is the reference polarity of PWM signal. When OCxH and OCxL output at the same time, the Output polarity of OCTxL is opposite with OCTxH. When OCS is set to 1, the compare value comes from FOC module. OCTxL is the reference polarity. When OCxH and OCxL output at the same time, the Output polarity of OCTxH is opposite with OCTxL.

20.1.2.1 Count Compare Module

The compare value is selected as FOC_CMPU/V/W or DRV_DR by the configuration of DRV_CR[OCS]. Compared with the value of counter, 4 pairs PWM signal OCxREF is generated. DRV_DR is used to realize pre-charging, braking and the control of BLDC. When CNTR is less than the compare value, OCxREF output high voltage level, and CNTR is more than the compare value, OCxREF output low voltage level.

Setting DRV_CR[OCS] to 1, FOC_CMPU/V/W is selected as compare value to generate PWM signal OC1REF/OC2REF/OC3REF.

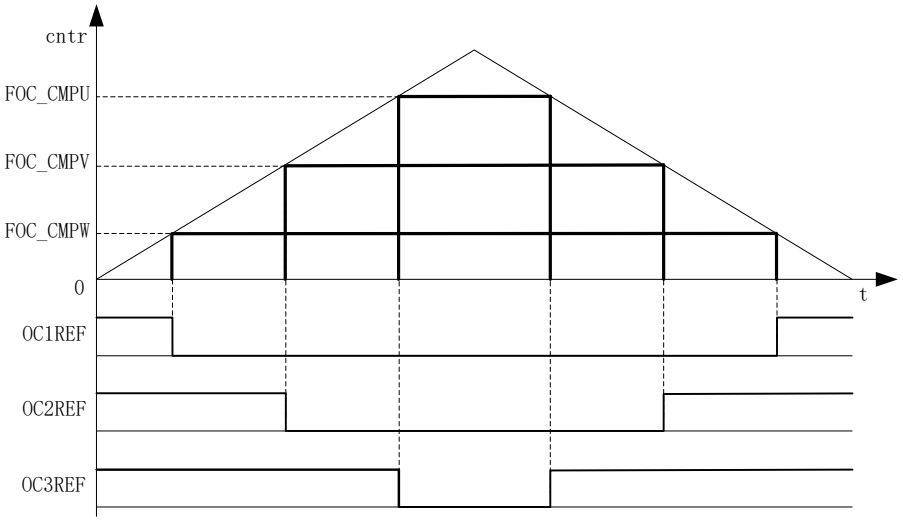


Figure 20-4 Diagram of PWM signal

Setting DRV_CR[OCS] to 0, DRV_DR is selected as compare value to generate 3 pairs same PWM signal OC1REF/OC2REF/OC3REF.

Duty cycle = 100% × DRV_DR / DRV_ARR (Providing DRV_ARR = 750, DRV_DR =375, and duty cycle = 50%)

20.1.2.2 Deadtime module

Driver module can output complementary OCxl and OCxH (i.e., with deadtime insertion) by hardware. The deadtime insertion is enable when FOC_DTR is not equal to 0. Each channel has a 8 bit deadtime generator, which can be configured by DRV_DTR. When the rising edge of OCxREF comes, the rising edge of OCxL is delayed by a predefined time, which is set by DRV_DTR. When the falling edge of OCxREF comes, the rising edge of OCxH is delayed by the same time.

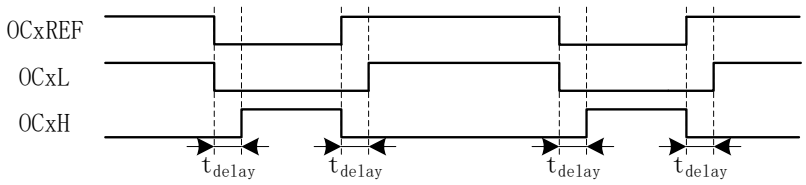


Figure 20-5 Complementary Output with Deadtime Insertion

20.1.2.3 Output Enable and Polarity

The output mode can be selected as invalid voltage level or complementary output with deadtime

insertion by the configuration of DRV_CMR[xHE, xLE]. Configure DRV_CMR[xHP, xLP] to select the polarity of output. In the application of BLDC control, DRV_CMR can be controlled by Timer 1 automatically to implement the commutation of inverter by hardware. Setting DRV_CR[MESEL] to 0, the motor engine works in BLDC mode. When a commutation event is generated, the corresponding TIM_DBRx is updated to DRV_CMR and CMP_CR2[4:3].

By configuring the xHE and xLE of the DRV_CMR register, you can select the output mode as an invalid level or insert the complementary Output deadzone. Configure the polarity of the output by configuring xHP and xLP in the DRV_CMR register. DRV_CMR is generally configured by software. When applied to BLDC square wave control, you can configure TIMER1 to automatically control DRV_CMR to achieve the function of automatic commutation; configure MESEL = 0 of the DRV_CR register to let the motor engine ME select the BLDC control mode. At this time, when TIM1 generates When a commutation event occurs, the corresponding TIM1_DBRx is updated to DRV_CMR and CMP_CR2 [4: 3] of the comparator.

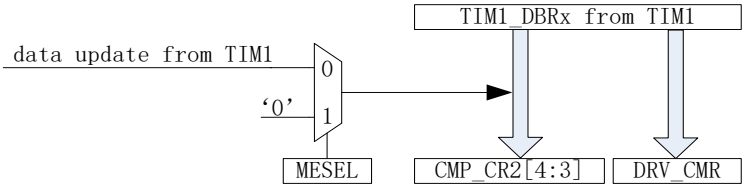


Figure 20-6 DRV_CMR and CMP_CR2[4: 3] Controlled by Timer 1

The functions, such as pre-changing and braking, can be implemented by the configuration of DRV_CMR[xHE, xLE] with DRV_DR and DRV_ARR. The duty cycle is controlled by DRV_DR and DRV_ARR. The output mode is controlled by DRV_CMR[xHE, xLE].

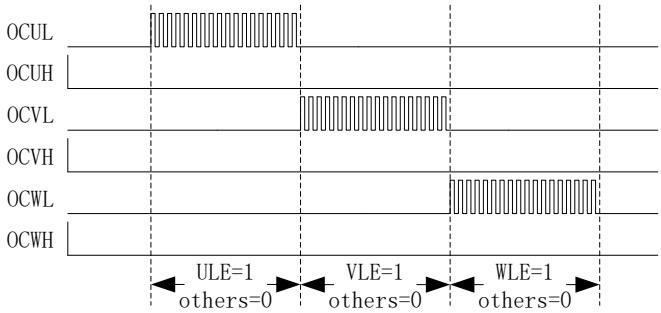


Figure 20-7 Diagram of Pre-Charging

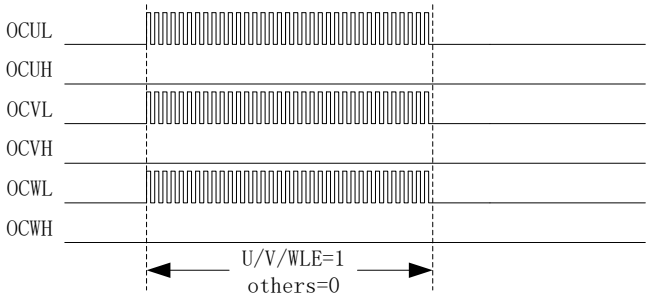


Figure 20-8 Diagram of Braking

20.1.2.4 Main Output Enable

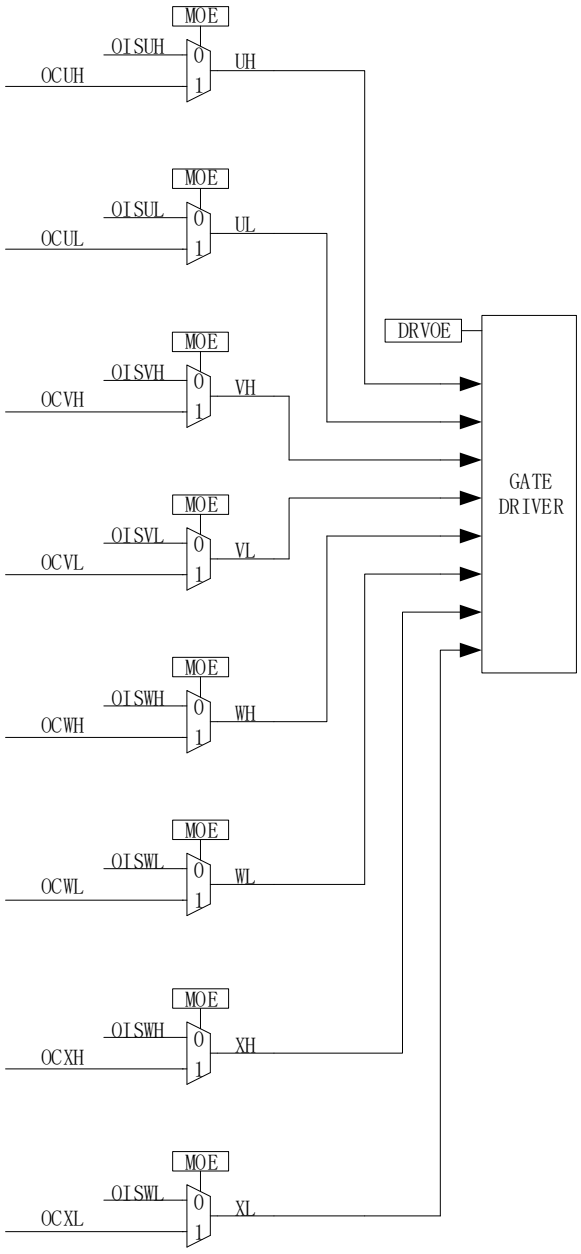


Figure 20-9 Back-End Block Diagram of the Output Control Module

When MOE is enabled, the Driver module Outputs PWM signals, which are used to control motor. When MOE is disabled, the Driver outputs the predefined voltage level, used for the standby mode.

20.1.2.5 Interrupt

20.1.2.5.1 Comparing Interrupt

Configure DRV_SR[DCIM] to select the compare mode, comparing event triggered in up-counting process or down-counting process. When the value of counter is equal to the compare value DRV_COMR in corresponding process, an interrupt event is generated and flag DCIF is set to 1. Software writing 0 to DCIF can clear the flag, and writing 1 is invalid.

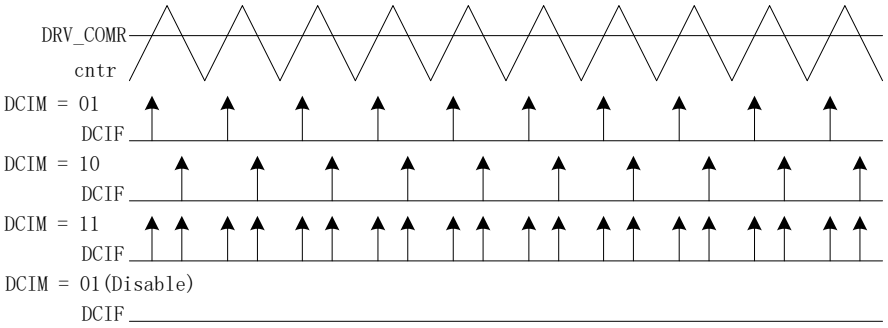


Figure 20-10 Driver Comparing Interrupt

20.1.2.5.2 FG Interrupt

Set DRV_SR[FGIE] to 1 to enable the FG interrupt. At this time, each rotation of the motor (electrical angle) generates an interrupt event.

20.1.3 Gate Driver Mode (Only for FU6813)

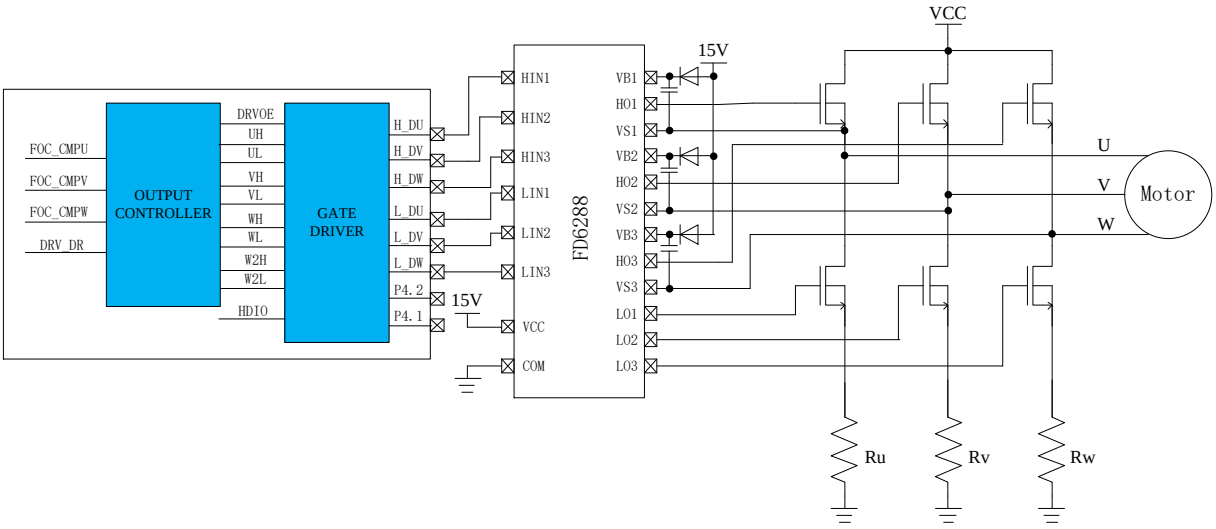


Figure 20-11 Schematic Diagram of Gate Driver Mode

Gate Driver is the Output FU6813. As shown in Figure 20-11, DRVOE is the enable signal of Gate Driver. Different from FU6863 6N Pre-driver, the Output Gate Driver is connected to HVIC, and the gate of MOS is driven by HVIC.

20.1.4 6N Pre-driver Mode (for FU6863)

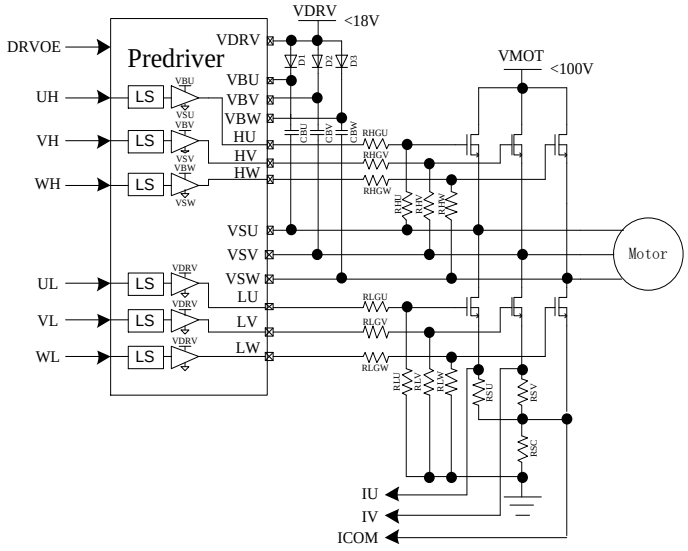


Figure 20-12 Schematic Diagram of 6N Pre-driver Mode

6N Pre-driver is shown in Figure 20-12. DRVOE is the enable signal of Pre-driver. The Output Pre-driver is connected to 6 NMOS to drive motor。

Table 20-1 FU6863Q Truth Table of internal Pre-driver Signal

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 Registers

20.2.1 DRV_CR (0x4062)

Table 20-2 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	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	DRVEN	Counter enable 0: Disable 1: Enable
[6]	DDIR	Output direction (forward and reverse) This bit is used to control the direction of motor, both effective in BLDC mode and FOC mode. For BLDC mode, some Timer 1 register is needed to configure. 0: Forward 1: Reverse
[5]	FOCEN	FOC/SVPWM/SPWM Module enable 0: Disable 1: Enable
[4]	DRPE	DRV_DR preload enable When preload is enabled, software writing DRV_DR will be updated after an underflow event. Otherwise, it is updated immediately 0: Disable 1: Enable
[3]	OCS	Counter compare value selection 0: DRV_DR 1: FOC/SVPWM/SPWM module
[2]	MESEL	ME mode selection 0: BLDC mode 1: FOC/SVPWM/SPWM mode
[1]	RSV	Reserved
[0]	DRVOE	Driver output enable 0: Disable 1: Enable

20.2.2 DRV_SR (0x4061)

Table 20-3 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	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	SYSTIF	SYS TICK interrupt flag This bit is set to 1 by hardware and cleared by software. 0: No event 1: SYS TICK interrupt event
[6]	SYSTIE	SYS TICK interrupt flag This bit is set to 1 by hardware and cleared by software. 0: No event 1: SYS TICK interrupt event
[5]	FGIF	FG interrupt flag Every rotation of motor (electrical cycle), the FG interrupt is generated This bit is set to 1 by hardware and cleared by software. 0: No event 1: FG interrupt event
[4]	DCIF	DRV comparing interrupt flag When the count value is equal to DRV_COMR, an interrupt flag is generated at corresponding counting process, configured by DCIM. This bit is set to 1 by hardware and cleared by software. 0: No event 1: Comparing interrupt event
[3]	FGIE	FG interrupt enable When the interrupt is enabled, A FG interrupt will be generated every rotation (electrical cycle). 0: Disable 1: Enable
[2]	DCIP	Generates an interrupt in every x periods 0: 1 periods 1: 2 periods
[1:0]	DCIM	DRV comparing interrupt mode This bit set the process, where could generate a DRV comparing interrupt. 00: No interrupt is generated 01: Up-counting process 10: Down-counting process

		11: Up/Down-counting process

20.2.3 DRV_OUT (0xF8)

Table 20-4 DRV_OUT (0xF8)

Bit	7	6	5	4	3	2	1	0
Name	MOE	RSV	OISWXL	OISWXH	OISVL	OISVH	OISUL	OISUH
Type	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	MOE	Main output enable This bit is used to select the source of the U/V/W/X phase outputs. This bit can be set to 1 and cleared by software. When bus current protection is triggered (refer to 30.1.1.1), this bit is cleared by hardware and output is disabled. 0: Disable. Outputs the predefined voltage level OISUH/ OISVH/ OISWH and OISUL/ OISVL/ OISWL. 1: Enable. Outputs PWM signal.
[6]	RSV	Reserved
[5]	OISWXL	Default voltage level of output for WL and XL Refer to the description of OISUH
[4]	OISWXH	Default voltage level of output for WH and XH Refer to the description of OISUH
[3]	OISVL	Default voltage level of output for VL Refer to the description of OISUH
[2]	OISVH	Default voltage level of output for VH Refer to the description of OISUH
[1]	OISUL	Default voltage level of output for UL Refer to the description of OISUH
[0]	OISUH	Default voltage level of output for UH This bit sets the default voltage level of UH. When MOE = 0, UH outputs the predefined voltage level. 0: Low voltage level 1: High voltage level

20.2.4 DRV_CMR (0x405C, 0x405D)

Note: For BLDC control mode, DRV_CMR is controlled by Timer 1.

Table 20-5 DRV_CMRH (0x405C)

Bit	7	6	5	4	3	2	1	0

Name	XHP	XLP	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

Table 20-6 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	Function
[15]	XHP	Phase X high side output polarity 0: High voltage level active 1: Low voltage level active
[14]	XLP	Phase X low side output polarity 0: High voltage level active 1: Low voltage level active
[13]	XHE	Phase X high side output enable 0: Disable 1: Enable Note: When XLE and XHE is set to 1 at the same time, phase X outputs complementary PWM signals with deadtime insertion. Low side output is the reference polarity. When both side signal output at the same time, the Output polarity of high side is opposite with low side.
[12]	XLE	Phase X low side output enable 0: Disable 1: Enable
[11]	WHP	Phase W high side output polarity 0: High voltage level active 1: Low voltage level active
[10]	WLP	Phase W low side output polarity 0: High voltage level active 1: Low voltage level active
[9]	VHP	Phase V high side output polarity 0: High voltage level active 1: Low voltage level active
[8]	VLP	Phase V low side output polarity 0: High voltage level active 1: Low voltage level active
[7]	UHP	Phase U high side output polarity 0: High voltage level active 1: Low voltage level active
[6]	ULP	Phase U low side output polarity

		0: High voltage level active 1: Low voltage level active 1: Active low
[5]	WHE	Phase W high side output enable 0: Disable 1: Enable Note: When WLE and WHE is set to 1 at the same time, phase X outputs complementary PWM signals with deadtime insertion. Low side output is the reference polarity. When both side signal output at the same time, the Output polarity of high side is opposite with low side.
[4]	WLE	Phase W low side output enable 0: Disable 1: Enable
[3]	VHE	Phase V high side output enable 0: Disable 1: Enable Note: When VLE and VHE is set to 1 at the same time, phase X outputs complementary PWM signals with deadtime insertion. Low side output is the reference polarity. When both side signal output at the same time, the Output polarity of high side is opposite with low side.
[2]	VLE	Phase V low side output enable 0: Disable 1: Enable
[1]	UHE	Phase U high side output enable 0: Disable 1: Enable Note: When ULE and UHE is set to 1 at the same time, phase X outputs complementary PWM signals with deadtime insertion. Low side output is the reference polarity. When both side signal output at the same time, the output polarity of high side is opposite with low side.
[0]	ULE	Phase U low side output enable 0: Disable 1: Enable

20.2.5 DRV_ARR (0x405E, 0x405F)

Table 20-7 DRV_ARRH (0x405E)

Bit	7	6	5	4	3	2	1	0
Name	DRV_ARR[11:8]							
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 20-8 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	Function
[11:0]	DRV_ARR	The reload value of Driver counter, which configures the carrier period and operation period. When the Driver counter count from 0 to DRV_ARR, it generates an overflow event. And then count down to 0. $f_{Carrier} = f_{mcu} / (2 \times (DRV\_ARR + 1))$ Range: (0,4095)

20.2.6 DRV_COMR (0x405A, 0x405B)

Table 20-9 DRV_COMRH (0x405A)

Bit	7	6	5	4	3	2	1	0
Name	DRV_COMR[11:8]							
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 20-10 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	Function
[11:0]	DRV_COMR	The compare match value of the counter. When the value of Driver counter is equal to DRV_COMR, a comparing event will be generated at corresponding counting process. Range: (0,4095)

20.2.7 DRV_DR (0x4058, 0x4059)

Table 20-11 DRV_DRH (0x4058)

Bit	7	6	5	4	3	2	1	0
Name	DRV_DR[11:8]							
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Table 20-12 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	Function
[11:0]	DRV_DR	PWM duty cycle by software When DRV_CR[OCS] = 0, the output voltage level is determined by comparing DRV_CNTR and DRV_DR. When DRV_CNTR < DRV_DR, output high voltage level, and DRV_CNTR ≥ DRV_DR, output high voltage level. Note: When this register is used as the compare reference, the high side output is the reference polarity, and the low side output voltage level is opposite with high side. Range: (0,4095)

20.2.8 DRV_DTR (0x4060)

Table 20-13 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	Function
[7:0]	DRV_DTR	Deadtime DTR is the deadtime inserted between high and low side complementary outputs. Providing MCU clock is 24MHz (41.67ns), Deadtime = (DTR + 1) × 41.67ns Note: When DTR = 0, there is no deadtime.

21 WDT

The watchdog module is a timer in LS_OSC clock. It is used to monitor the operation of program and guard against crash. The user software must regularly reset the watchdog counter within a certain time frame. Otherwise, an overflow event is triggered and watchdog will generate a reset to the processor.

The watchdog counter count from 0 after enabled. When the value of counter is counting to 0xFFFC, watchdog module outputs a signal, with the length of 4 LS_OSC periods, to reset FU6813/63. When software regularly reset the watchdog counter within a certain time frame, watchdog counter will recount from 0 and not reset the processor.

21.1 Functional Block Diagram of Watchdog Module

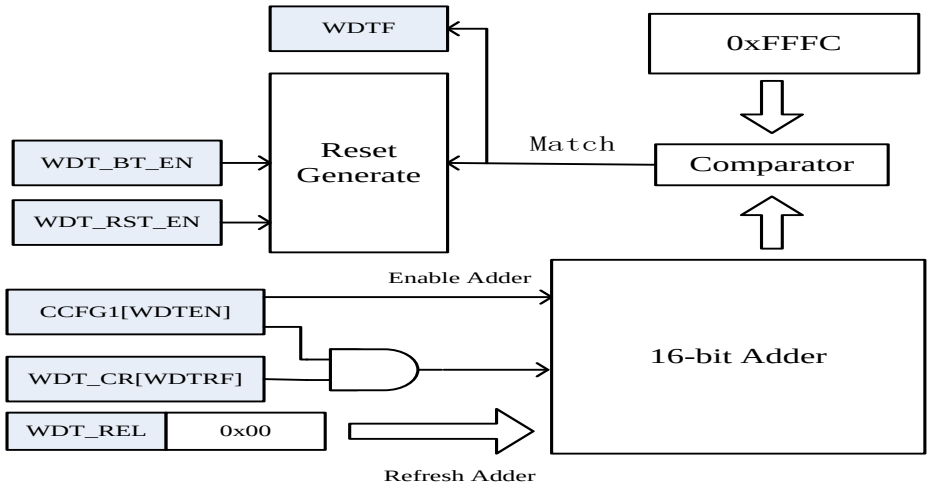


Figure 21-1 Functional Block Diagram of Watchdog Module

21.2 Attentions

1. When the processor works in standby mode or sleep mode, WDT will stop counting and the count value will not be cleared.
2. WDT is disabled during simulation.
3. When the processor is reset by WDT, RST_SR[RSTWDT] is set to 1.

21.3 WDT Operating Instructions

1. Set CCFG1[WDTEN] to 1 to start watchdog counting.
2. Configure WDT_ARR, this register can be written before enabling WDT.
3. Set WDT_CR[WDTRF] to 1 to reset watchdog counter.

21.4 WDT Register

21.4.1 WDT_CR (0x4026)

Table 21-1 WDT_CSR (0x4026)

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

Field	Name	Description
[7:2]	RSV	Reserved
[1]	WDTF	Watchdog reset flag
[0]	WDTRF	1: Reset watchdog counter; 0: Invalid

21.4.2 WDT_ARR (0x4027)

Table 21-2 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

Field	Name	Description
[7:0]	WDT_ARR	This register is the value of watchdog counter when it is reset.

22 RTC and Clock Calibration

22.1 RTC Descriptions

22.1.1 Functional Block Diagram

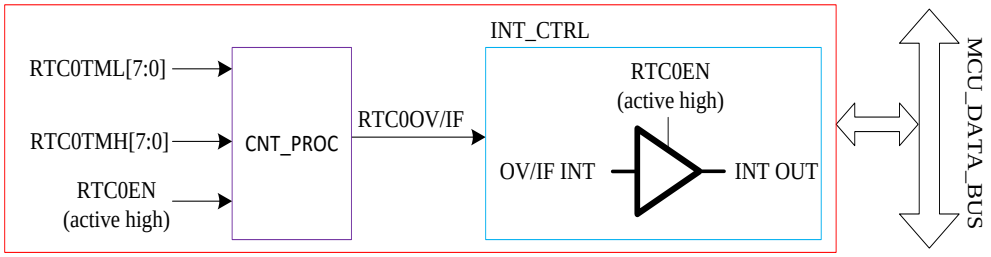


Figure 22-1 Functional Block diagram of RTC

22.1.2 Operating Instructions

Software write RTC0TMH and RTC0TML to set the reload value of the RTC counter;

Set RTC0STA[RTC0EN] to 1 to enable RTC counting.

22.2 RTC Register

22.2.1 RTC_TM (0x402C, 0x402D)

Table 22-1RTC_TMH (0x402C)

Bit	7	6	5	4	3	2	1	0
Name	RTC0TMH							
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

Table 22-2 RTC_TML (0x402D)

Bit	7	6	5	4	3	2	1	0
Name	RTC0TML							
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

Field	Name	Description
[15:0]	RTC_TM	RTC count value register Writing: RTC counter counts with the initial value of 0 at 32768Hz frequency. When the value of counter is equal to RTC_TM, an interrupt event is generated. The counter is cleared and continues re-count. Reading: The transient value of counter.

22.2.2 RTC_STA (0x402E)

Table 22-3 RTC_STA (0x402E)

Bit	7	6	5	4	3	2	1：0
Name	RTC_EN	RTC0OV / RTC0IF	SCK_SEL	ISOSCEN	ESOSCAE	ESOSCEN	RSV
Type	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset	0	0	0	0	0	0	0

Field	Name	Description
7	RTC_EN	RTC enable 0: Disable 1: Enable
6	RTC_OV/ RTC_IF	RTC counting overflow interrupt/overflow flag. When RTCIE is 1 and RTC counter count to RTC_TM, an interrupt will be generated. This bit can be cleared by software. When RTCIE is 0 and RTC counter count to RTC_TM, no interrupt but an overflow flag will be generated. Software can read this flag and clear it to 0. 0: Not overflowed. 1: Overflowed. Note: This bit will not be cleared by hardware. Therefore, software clear after an overflow event is necessary.
5	SCK_SEL	RTC slow clock source 0: Internal slow clock 1: External slow clock Note: The slow clock is also used for clock calibration. For details, please refer to the corresponding chapters.
4	ISOSCEN	Internal slow clock enable 0: Disable 1: Enable
3	ESOSCAE	External slow clock crystal PIN enable 0: Whether the external slow clock crystal PIN configured as an analog input is determined by other IO registers. 1: External slow clock crystal PIN is analog input.
2	ESOSCEN	External slow clock enable 0: Disable. 1: Enable
1:0	RSV	Reserved

22.3 Clock Calibration

The clock calibration module use a slow clock to calibrate fast clock. The reference slow clock can be

selected as internal or external slow clock. The main idea is: Using a 12 bit counter count 4 slow clock periods time, reading the value of counter can evaluate the accuracy of fast clock.

Set CAL_STA to 1 to start clock calibration. When CAL_BSY = 0, the clock calibration is finished and CAL_ARR is the value of counter, counting 4 slow clock periods time.

22.3.1 Register: CAL_CR0, CAL_CR1 (0x4044, 0x4045)

Table 22-4 CAL_CR0 (0x4044)

Bit	7	6:4	3	2	1	0
Name	CAL_STA/CAL_BSY	RSV	CAL_ARR[11:8]			
Type	R/W	R	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0

Table 22-5 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

Field	Name	Description
[15]	CAL_STA/ CAL_BUSY	Writing 1: Start clock calibration. Reading 0: The clock calibration is completed. Reading 1: The clock calibration is in progress.
[14:12]	RSV	Reserved
[11:0]	CAL_ARR	Count value of calibration counter Calibration counter count 4 slow clock periods at fast clock frequency. Note: When this value is 0, there is no corresponding slow clock input. When this value is 0xFFF, calibration counter overflows.

23 IO

23.1 IO Operating Descriptions

1. Each bit in the register P0，P1，P2，P3, P4 connects to the pin P0.0~P0.7, P1.0~P1.7, P2.0~P2.7, P3.0~P3.7, P4.0~P4.2 respectively.
2. Registers P0_OE, P1_OE, P2_OE, P3_OE, P4_OE control the output enable of the pin P0.0~P4.2.
3. Every GPIO pin has a configurable pull-up resistance on-chip. Each pull-up resistance can be enabled respectively for the pin P0.0~P4.2 by setting the P0_PU, P1_PU, P2_PU, P3_PU, P4_PU. For the pins P0.0~P0.2, P1.3~P1.6，P2.1, P3.7~P3.6 the build-in pull-up resistance are about 5kΩ. For the rest of the GPIOs, they are about 33kΩ.
4. P1.3~P1.7, P2.0~P2.7, P3.0~P3.5 can be set as analog IO pins. Setting the registers P1_AN, P2_AN, P3_AN will set the pin to analog mode. All the digital function in the IO will be temporary disabled under this setting. Any read to the respective bits in the registers P1，P2，P3 will return 0.
5. The source of U/V/W phase output signal OCUH/OCVH/OCWH and OCUL/OCVL/OCWL is selected as Timer 0, Timer 1 or FOC module, configured by DRV_CTL[OCS]. When MOE = 0, they are controlled directly by the predefined voltage level OISUH/OISVH/OISWH and OISUL/OISVL/OISWL in DRV_OUT. When MOE = 1, they are controlled by OCUH/OCVH/OCWH and OCUL/OCVL/OCWL.
6. The Output Timer 0 can select between T0_OC4, which is the channel-4 Output Timer 0, or OIS4, by setting the bit DRV_OUT[MOE].
7. The bit DRV_OUT[MOE] can be written 0 or 1 by software. It will be cleared by hardware when over-current event takes place.
8. IO priority
    a) For all multiplexing IOs, GPIO has the lowest priority.
    b) P0.1：I2C > Timer4 > DBG_SIG > GPIO
    c) P0.5：SPI > UART > GPIO
    d) P0.6：SPI > UART > GPIO
    e) P0.7：Timer 2 > CMP > SPI > GPIO

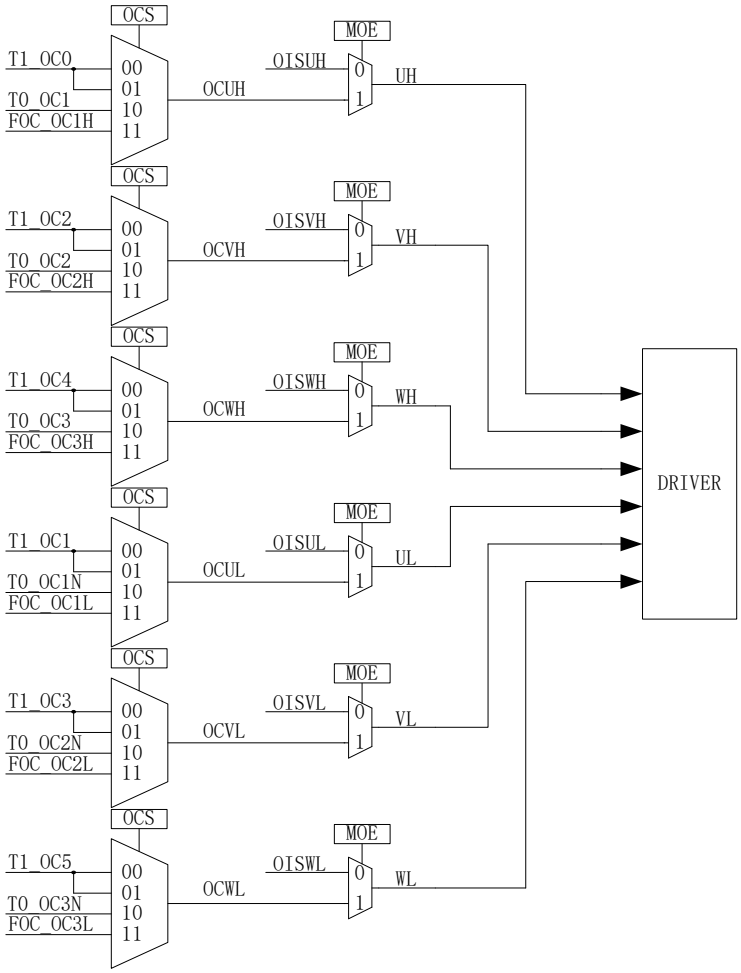


Figure 23-1 Configuration of U/V/W Phase Output

23.2 IO Registers

23.2.1 P0_OE (0xFC)

Table 23-1 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	Function
[7:0]	P0_OE	P0.0~P0.7 output enable in digital mode 0: Disable 1: Enable

23.2.2 P1_OE (0xFD)

Table 23-2 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	Function
[7:0]	P1_OE	P1.0~P1.7 output enable in digital mode 0: Disable 1: Enable

23.2.3 P2_OE (0xFE)

Table 23-3 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	Function
[7:0]	P2_OE	P2.0~P2.7 output enable in digital mode 0: Disable 1: Enable

23.2.4 P3_OE (0xFF)

Table 23-4 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	Function
[7:0]	P3_OE	P3.0~P3.7 output enable in digital mode 0: Disable 1: Enable

23.2.5 P4_OE (0xE9)

Table 23-5 P4_OE (0xE9)

Bit	7	6	5	4	3	2	1	0

Name	RSV	P4_OE[2]	P4_OE[1]	P4_OE[0]
Type	R	R/W	R/W	R/W
Reset	0	0	0	0

Bit	Name	Function
[7:3]	RSV	Reserved
[2:0]	P4_OE	P4.0~P4.2 output enable in digital mode 0: Disable 1: Enable

23.2.6 P1_AN (0x4050)

Table 23-6 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	Function
[7:4]	P1_AN	P1.7~P1.4 analog mode enable 0: Disable 1: Enable
[3]	HBMOD	P1.3 function mode selection. Combine the P1_OE[3] to select the function mode. HBMODE / P1_OE[3] / P1.3Mode 0 / 0 / Digital input 0 / 1 / Digital output 1 / 0 / Analog mode 1 / 1 / Digital output with enhanced driving ability when output 1.
[2]	HDIO	Gate Driver IO output driving ability selection (only for L_DU, L_DV, L_DW, H_DU, H_DV, H_DW of FU6813) 0: Normal driving ability 1: Enhanced driving ability
[1]	ODE1	P0.1 open drain output enable 0: Disable 1: Enable
[0]	ODE0	P0.0 open drain output enable 0: Disable 1: Enable

23.2.7 P2_AN (0x4051)

Table 23-7 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	Function
[7:0]	P2_AN	P2.7~P2.0 analog mode enable 0: Disable 1: Enable

23.2.8 P3_AN (0x4052)

Table 23-8 P3_AN (0x4052)

Bit	7	6	5	4	3	2	1	0
Name	RSV		P3_AN					
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:6]	RSV	Reserved
[5:0]	P3_AN	P3.5~P3.0 analog mode enable 0: Disable 1: Enable

23.2.9 P0_PU (0x4053)

Table 23-9 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	Function
[7:0]	P0_PU	P0.7~P0.0 pull-up resistance enable 0: Disable 1: Enable

23.2.10 P1_PU (0x4054)

Table 23-10 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	Function
[7:0]	P1_PU	P1.7~P1.0 pull-up resistance enable 0: Disable 1: Enable

23.2.11 P2_PU (0x4055)

Table 23-11 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	Function
[7:0]	P2_PU	P2.7~P2.0 pull-up resistance enable 0: Disable 1: Enable

23.2.12 P3_PU (0x4056)

Table 23-12 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	Function
[7:0]	P3_PU	P3.7~P3.0 pull-up resistance enable 0: Disable 1: Enable

23.2.13 P4_PU (0x4057)

Table 23-13 P4_PU (0x4057)

Bit	7	6	5	4	3	2	1	0

Name	RSV					P4_PU[2]	P4_PU[1]	P4_PU[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	Function
[7:3]	RSV	Reserved
[2:0]	P4_PU	P4.2~P4.0 pull-up resistance enable 0: Disable 1: Enable

23.2.14 PH_SEL (0x404C)

Table 23-14 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	Function
[7]	SPITMOD	This bit determines the MISO pin state after SPI transmission, when the device works in slave mode. 0: Output state 1: High-impedance state
[6]	UART1EN	UART1 function enable 0: Disable 1: Enable
[5]	UART2EN	UART2 function enable 0: Disable 1: Enable
[4]	T4SEL	P0.1 in Timer 4 mode control bit. 0: P0.1 is regarded as GPIO or other function mode pin. 1: P0.1 is regarded as Timer 4 input or output. Note: the I2C mode is prior to TIMER4. P0.1 is regarded as SCL when I2C is enable.
[3]	T3SEL	P1.1 in Timer 3 mode control bit. 0: P1.1 is regarded as GPIO or other function mode pin. 1: P1.1 is regarded as Timer 3 input or output.
[2]	T2SEL	P1.0 in Timer 2 mode control bit. 0: P1.0 is regarded as GPIO or other function mode pin. 1: P1.0 is regarded as Timer 2 input or output.
[1]	T2SSEL	P0.7 in Timer 2 mode control bit. 0: P0.7 is regarded as GPIO or other function mode pin. 1: P0.7 is regarded as Timer 2 input or output. Note: Timer 2 has the highest priority, after it comparator output, and

		then the MISO of SPI.
[0]	XOE	P4.2/P4.1 in Driver mode control bit 0: P4.2/P4.1 is regarded as GPIO or other function mode pin. 1: P4.2/P4.1 is regarded as Output XH/XL, PWM signal or predefined voltage level based on the configuration of DRVOE[MOE].

23.2.15 P0 (0x80)/P1 (0x90)/P2 (0xA0)/P3 (0xB0)/P4 (0xE8)

The pin registers support read-modify-write action. They are P0~P4 registers. The read-modify-write (RMW) instructions as shown in the following Table.

Table 23-15 P0/P1/P2/P3/P4

Bit	7	6	5	4	3	2	1	0
Name	GPx[7]	GPx[6]	GPx[5]	GPx[4]	GPx[3]	GPx[2]	GPx[1]	GPx[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

Name	Description	R/W	Reset
P0[7:0]	Pin register 0	R/W	0x00
P1[7:0]	Pin register 1	R/W	0x00
P2[7:0]	Pin register 2	R/W	0x00
P3[7:0]	Pin register 3	R/W	0x00
P4[2:0]	Pin register 4	R/W	0x00

Note: P4 has a total of 3 Pins. The corresponding P4 output register is P4[2: 0]。

Table 23-16 Read-Modify-Write Instructions

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

24 ADC

24.1 Functional Block Diagram of ADC

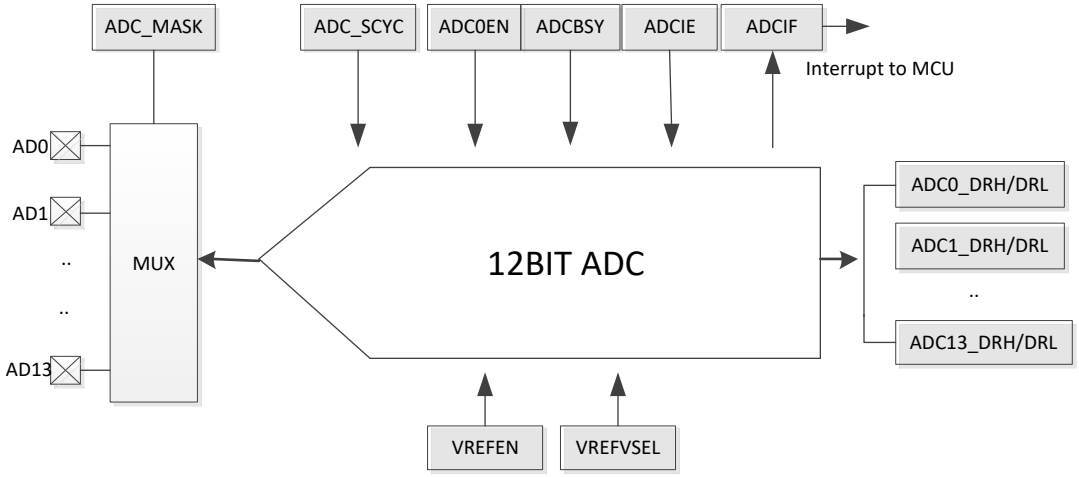


Figure 24-1 Functional Block Diagram of ADC

24.2 ADC Operating Instructions

The FU6813/63 ADC module is a 12-bit analog-to-digital converter with 14 channels. ADC module supports trigger mode. Software writing ADC_CR corresponding start bit enable sequenced mode. When FOC function is enabled, ADC works in trigger mode to sample system states in the appropriate timing. After that, ADC will be back to sequenced mode.

24.2.1 Sequential Scan Mode

The FU6813/63 has an integrated 14 channels synchronous ADC with the resolution 12-bit. Before ADC operation, the sampling channel is selected by ADC_MASK and the sampling delay time is configured (the minimum value is 3). Set ADC_CR[ADCEN, ADCBSY] to start ADC sampling.

In addition, ADC supports the trigger function, with a priority higher than software operation of MCU. The trigger source comes from the FOC module

If FOC function is enabled (set FOC_SET0[FOC_EN] to 1), it will start ADC automatically and trigger the ADC to sample system states in the appropriate timing. The sampling value will be updated to FOC module automatically.

NOTE: ADC sampling by the software operation will be interrupted due to higher-priority of ADC trigger function. ADC will interrupt the current operation, such as ADC sampling by the software operation, to do the trigger function when the trigger conditions are satisfied. After completing the sampling by trigger function, sampling by the software operation will be restored immediately.

The specific steps of ADC as follows:

1. Set the appropriate sample clock cycles ADC_SCYC. This value should be set according to the application, with a minimum value of 3.
2. Set the appropriate ADC reference voltage ADCREF. Note, the VDD5 must be larger than 5.3V if the internal VREF with VREF = 5V has been chosen to be the reference voltage. Namely that under the high-voltage mode (VDD_MODE = 0). VRF = 5V is not supported.
3. MCU software operation:

a) Set the requisite channel number ADC_MASK.

b) Set to enable interrupt ADCIE.

c) Set ADCEN=1 and ADCBSY=1 sequentially to start ADC.

d) The values of ADC0~13_DR are available to be read by MCU after the end of the ADC conversion.

24.2.2 ADC Trigger Mode

When FOC module is enabled, ADC can be triggered to sample system states in the appropriate timing. The ADC module can work in trigger mode or sequenced mode simultaneously, controlled by internal circuit. However, it is not recommended.

24.3 ADC Register

24.3.1 ADC_CR (0x4039)

Table 24-1 ADC_CR (0x4039)

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

Bit	Name	Function
[7]	ADCEN	ADC enable 0: Disable 1: Enable
[6]	ADCBSY	ADC busy flag With ADCEN enabled, setting this bit to logical 1 will start the conversion process. This bit is cleared automatically by hardware when ADC conversion is complete. When this bit is logical 1, writing 1 is invalid. And write of logical 0 is invalid. Reading this bit indicates the ADC state. When ADC_MASK=0, any write to this bit will be ignored.
[5:3]	RSV	Reserved
[2]	ADCALIGN	ADC data alignment mode selection 0: ADC data is right aligned.

		1: ADC data is aligned to the second highest bit. (exclude the trigger mode)
[1]	ADCIE	ADC interrupt enable. If enabled, ADCIF can send an interrupt request to MCU (TRIG mode interrupt is not included). 0: Disable 1: Enable
[0]	ADCIF	ADC conversion end flag. When the conversion has completed, ADCIE = 1 and an interrupt event will be triggered. ADCIE has no effect on this bit. 0: Not completed 1: Completed This bit uses the interrupt entry with the ADTRIGIF, so it is necessary to confirm the interrupt flag, ADTRIGIF or ADCIF, before clearing the corresponding bit.

24.3.2 ADC_MASK (0x4036, 0x4037)

Table 24-2 ADC_MASKH (0x4036)

Bit	7：6	5	4	3	2	1	0
Name	RSV	CH13EN	CH12EN	CH11EN	CH10EN	CH9EN	CH8EN
Type	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0

Table 24-3 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	Function
[14:15]	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

24.3.3 ADC_SCYC (0x4035[5:2], 0x4038)

Table 24-4 ADC_SCYCL (0x4038)

Bit	7:4	3:0
Name	ADC_SCYCL [7:4]	ADC_SCYCL [3:0]
Type	R/W	R/W
Reset	0011	0011

Bit	Name	Function
[11:8]	ADC_SCYCH [3:0]	ADC sampling time set for channel-8/9/10/11/12/13. When ADC_SCYC[11] is 0, the sampling period is ADC_SCYC[10:8] times ADC clock cycles. When ADC_SCYC[11] is 0, the sampling period is (ADC_SCYC[10:8] × 8 + 7) times ADC clock cycles.
[7:4]	ADC_SCYCL [7:4]	ADC sampling time set for channel-5/6/7. When ADC_SCYC[7] is 0, the sampling period is ADC_SCYC[6:4] times ADC clock cycles. When ADC_SCYC[7] is 0, the sampling period is (ADC_SCYC[6:4] × 8 + 7) times ADC clock cycles.
[3:0]	ADC_SCYCL [3:0]	ADC sampling time set for channel-0/1/2/3/4. When ADC_SCYC[3] is 0, the sampling period is ADC_SCYC[2:0] times ADC clock cycles. When ADC_SCYC[3] is 0, the sampling period is (ADC_SCYC[2:0] × 8 + 7) times ADC clock cycles.

24.3.4 ADC0_DR (0x0600, 0x0601)

Table 24-5 ADC0_DRH (0x0600)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-6 ADC0_DRL (0x0601)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-0 after completion of ADC conversion.

[7:0]	DL	The lowest 8 bits of ADC output data for the channel-0 after completion of ADC conversion.

24.3.5 ADC1_DR (0x0602, 0x0603)

Table 24-7 ADC1_DRH (0x0602)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-8 ADC1_DRL (0x0603)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-1 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-1 after completion of ADC conversion.

24.3.6 ADC2_DR (0x0604, 0x0605)

Table 24-9 ADC2_DRH (0x0604)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-10 ADC2_DRL (0x0605)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-2 after completion of

		ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-2 after completion of ADC conversion.

24.3.7 ADC3_DR (0x0606, 0x0607)

Table 24-11 ADC3_DRH (0x0606))

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-12 ADC3_DRL (0x0607)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-3 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-3 after completion of ADC conversion.

24.3.8 ADC4_DR (0x0608, 0x0609)

Table 24-13 ADC4_DRH (0x0608)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-14 ADC4_DRL (0x0609)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-4 after completion of ADC conversion.

[7:0]	DL	The lowest 8 bits of ADC output data for the channel-4 after completion of ADC conversion.

24.3.9 ADC5_DR (0x060A, 0x060B)

Table 24-15 ADC5_DRH (0x060A)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-16 ADC5_DRL (0x060B)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-5 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-5 after completion of ADC conversion.

24.3.10 ADC6_DR (0x060C, 0x060D)

Table 24-17 ADC6_DRH (0x060C)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-18 ADC6_DRL (0x060D)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-6 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-6 after completion of ADC conversion.

24.3.11 ADC7_DR (0x060E, 0x060F)

Table 24-19 ADC7_DRH (0x060E)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-20 ADC7_DRL (0x060F)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-7 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-7 after completion of ADC conversion.

24.3.12 ADC8_DR (0x0610, 0x0611)

Table 24-21 ADC8_DRH (0x0610)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-22 ADC8_DRL (0x0611)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-8 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-8 after completion of ADC conversion.

24.3.13 ADC9_DR (0x0612, 0x0613)

Table 24-23 ADC9_DRH (0x0612)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-24 ADC9_DRL (0x0613)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-9 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-9 after completion of ADC conversion.

24.3.14 ADC10_DR (0x0614, 0x0615)

Table 24-25 ADC10_DRH (0x0614)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-26 ADC10_DRL (0x0615)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-10 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-10 after completion of ADC conversion.

24.3.15 ADC11_DR (0x0616, 0x0617)

Table 24-27 ADC11_DRH (0x0616)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-28 ADC11_DRL (0x0617)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-11 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-11 after completion of ADC conversion.

24.3.16 ADC12_DR (0x0618, 0x0619)

Table 24-29 ADC12_DRH (0x0618)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-30 ADC12_DRL (0x0619)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-12 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-12 after completion of ADC conversion.

24.3.17 ADC13_DR (0x061A, 0x061B)

Table 24-31 ADC13_DRH (0x061A)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R	R	R	R
Reset	0	0	0	0	0

Table 24-32 ADC13_DRL (0x061B)

Bit	7	6	5	4	3	2	1	0
Name	DL[7]	DL[6]	DL[5]	DL[4]	DL[3]	DL[2]	DL[1]	DL[0]
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[15:12]	RSV	Reserved
[11:8]	DH	The higher 4 bits of ADC output data for the channel-13 after completion of ADC conversion.
[7:0]	DL	The lowest 8 bits of ADC output data for the channel-13 after completion of ADC conversion.

25 DAC

25.1 Functional Block Diagram of DAC0

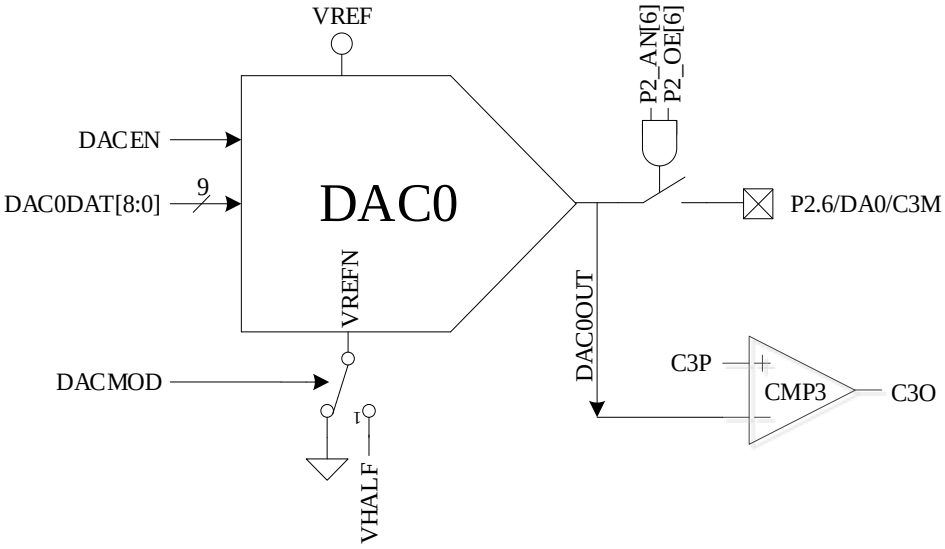


Figure 25-1 Functional Block Diagram of DAC0

Note：

1. The DAC0 output has no current driving ability, therefore, it must connect to a capacitive load. If a resistive load is connected, the voltage follower is necessary for output.
2. Set P2_AN[6] = 1 and P2_OE[6] = 1 to output DAC0 to P2.6
3. DAC0 uses VRFE as the reference voltage. Configuring VREFEN = 1, DACEN = 1 enable DAC0 operation.

25.1.1 DAC_CR (0x4035)

Table 25-1 DAC_CR (0x4035)

Bit	7	6	5	4	3	2	1	0
Name	DACEN	DACMOD	ADC_SCYC[11:8]				DAC2EN	RSV
Type	R/W	R/W	R/W				R	R
Reset	0	0	0011				0	0

Bit	Name	Function
[7]	DACEN	DAC0 and 1 enable 0: Disable 1: Enable
[6]	DACMOD	DAC mode selection 0: Normal mode. DAC output voltage range is 0 to VREF. 1: Half voltage mode. DAC output voltage range is VHALF to VREF.
[5:2]	ADC_SCYC[11:8]	Please refer to the description of ADC_SCYC
[1]	DAC2EN	DAC2 enable

		0: Disable 1: Enable
[0]	RSV	Reserved

25.1.2 DAC0_DR (0x404B)

Table 25-2 DAC0_DR (0x404B)

Bit	7:0
Name	DAC0DAT[8:1]
Type	R/W
Reset	0

Bit	Name	Function
[7:0]	DAC0DAT[8:1]	Higher 8 bits data of input for DAC0 Controller

25.2 Functional Block Diagram of DAC1

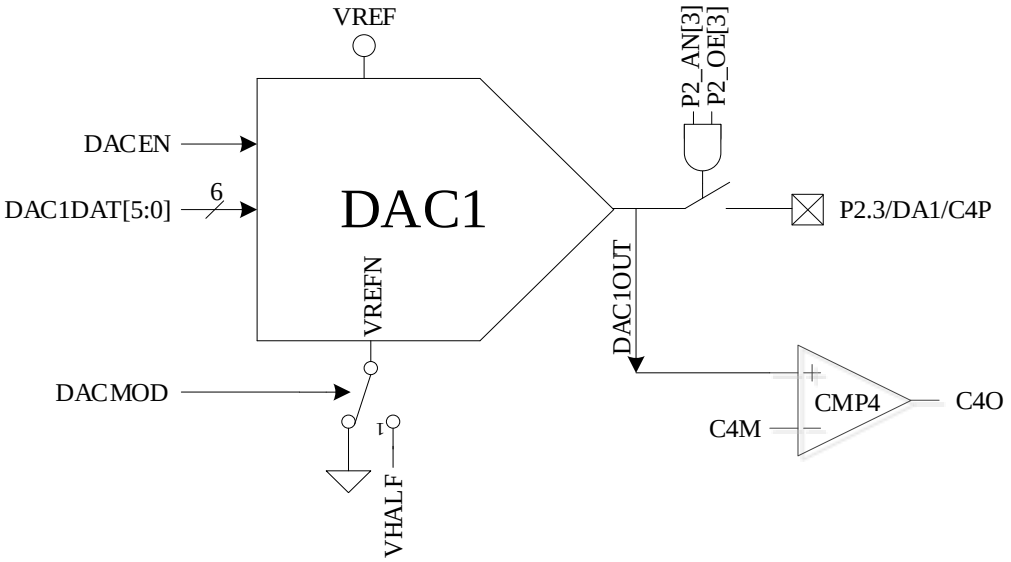


Figure 25-2 Functional Block Diagram of DAC1

Note：

1. The DAC1 output has no current driving ability, therefore, it must connect to a capacitive load. If a resistive load is connected, the voltage follower is necessary for output.
2. Set P2_AN[3] = 1 and P2_OE[3] = 1 to output DAC1 to P2.3
3. DAC1 uses VRFE as the reference voltage. Configuring VREFEN = 1, DACEN = 1 enable DAC1 operation.

25.2.1 DAC1_DR (0x404A)

Table 25-3 DAC1_DR (0x404A)

Bit	7	6	5:0
Name	DAC0DAT[0]	RSV	DAC1DAT
Type	R/W	R/W	R/W
Reset	0	0	0

Bit	Name	Function
[7]	DAC0DAT[0]	LSB of input for DAC0 Controller
[6]	RSV	Reserved
[5:0]	DAC1DAT	6 bits data input for DAC1 controller

25.3 Functional Block Diagram of DAC2

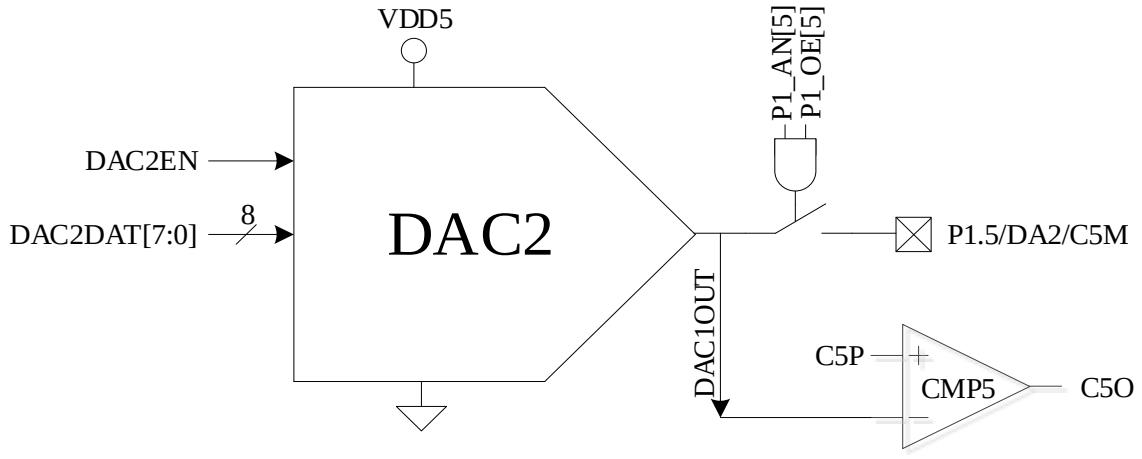


Figure 25-3 Functional Block Diagram of DAC2

Note:

1. The DAC2 output has no current driving ability, therefore, it must connect to a capacitive load. If a resistive load is connected, the voltage follower is necessary for output.
2. Set P1_AN[5] = 1 and P1_OE[5] = 1 to output DAC2 to P1.5.

25.3.1 DAC2_DR (0x4049)

Table 25-4DAC2_DR (0x4049)

Bit	7:0
Name	DAC2DAT
Type	R/W
Reset	0

Bit	Name	Function
[7:0]	DAC2DAT	8 bits data input for DAC2 controller

26 DMA

26.1 DMA Descriptions

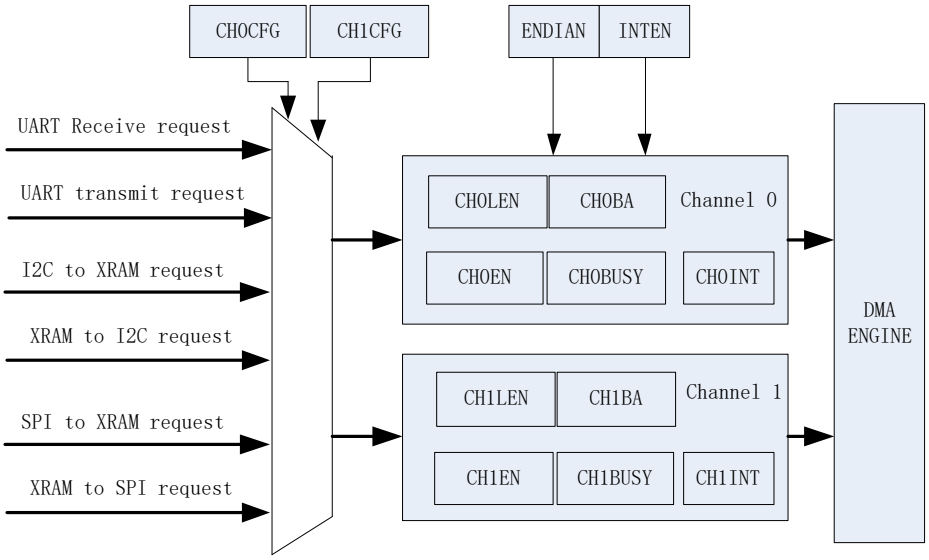


Figure 26-1 Functional Block Diagram of DMA

The DMA module contains a dual-channel DMA controller. It provides a hardware method of transferring data between peripherals (SPI, UART, I2C) and XRAM. During the transfer, the operation of DMA to XRAM will not disturb MCU writing and reading of XRAM. The length and address of data can is configurable. The transferring sequence direction can be selected and DMA has an interrupt event.

The software operation of DMA as following steps:

1. The configuration and enabling of peripherals.
2. Configuring DMAx_CR0[CFG] to select the peripherals input and output channel controlled by DMA.
3. The configuration of interrupt, transmission sequence direction, length and address of data.
4. Software write 1 to DMAx_CR0[EN] and DMAx_CR0[BSY] to enable DMA.
5. Clear DMA interrupt flag after the completion of transmitting.
6. Set DMAx_CR0[BSY] to 1 to restart DMA.

26.2 DMA Registers

ADDR	XSFR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0x403A	DMA0_CR0	DAMEN	DMABSY	DMACFG[2:0]			DMA_IE	ENDIAN	DMAIF
0x403B	DMA1_CR0	DMAEN	DMABSY	DMACFG[2:0]			DBG_SW	DBG_EN	DMAIF
0x403C	DMA0_CR1	CH0LEN[5:0]							
0x403E							CH0BA[10:8]		
0x403F		CH0BA[7:0]							
0x403D	DMA1_CR1	CH1LEN[5:0]							

0x4040			CH1BA[10:8]
0x4041		CH1BA[7:0]	

26.2.1 DMA0_CR0 (0x403A)

Table 26-1 DMA0_CR0 (0x403A)

Bit	7	6	5:3	2	1	0
Name	DMAEN	DMABSY	DMACFG	DMAIE	ENDIAN	DMAIF
Type	R/W	R/W1	R/W	R/W	R/W	R/W
Reset	0	0	000	0	0	0

Bit	Name	Function
[7]	DMAEN	DMA channel-0 enable
[6]	DMABSY	DMA channel-0 busy/start Reading: 0: The transmission of channel-0 has been completed or there is no transmission. 1: There is a transmitting uncompleted.. Writing: 0: Meaningless. 1: Start the transmitting of channel-0.
[5:3]	DMACFG [2:0]	Channel-0 peripheral selection 000: From UART to XRAM 001: From XRAM to UART. 010: From I2C to XRAM 011: From XRAM to I2C 100: From SPI to XRAM 101: From XRAM to SPI 110: From UART2 to XRAM 111: From XRAM to UART2. This bit cannot be configured when channel-0 is busy.
[2]	DMAIE	DMA interrupt enable. 0: Disable 1: Enable. An interrupt is generated by DMA when CH0INT or CH1INT is set 1.
[1]	ENDIAN	DMA data transferring mode. 0: Higher byte have a higher priority of receiving or transmitting. 1: Lower byte have a higher priority of receiving or transmitting. The setting of this bit is for 16 bit data mode. This bit should be set to 0 in 8 bit data mode. This bit cannot be configured when channel-0/1 is busy.
[0]	DMAIF	DMA channel-0 transmitting completed interrupt flag. This bit is set to 1 by hardware and cleared by software. 0: No event 1: Channel-0 transmitting completed. (When INTEN = 1, an interrupt event is triggered. Otherwise, only a flag is generated)

26.2.2 DMA1_CR0 (0x403B)

Table 26-2 DMA0_CR1 (0x403B)

Bit	7	6	5:3	2	1	0
Name	DMAEN	DMABSY	DMACFG	DBGSW	DBGEN	DMAIF
Type	R/W	R/W1	R/W	R/W	R/W	R/W
Reset	0	0	000	0	0	0

Bit	Name	Function
[7]	DMAEN	DMA channel-1 enable
[6]	DMABSY	DMA channel-1 busy/start Reading: 0: The transmission of channel-1 has been completed or there is no transmission. 1: There is a transmitting uncompleted.. Writing: 0: Meaningless. 1: Start the transmitting of channel-1
[5:3]	DMACFG [2:0]	Channel-1 peripheral selection 000: From UART to XRAM 001: From XRAM to UART. 010: From I2C to XRAM 011: From XRAM to I2C 100: From SPI to XRAM 101: From XRAM to SPI 110: From UART2 to XRAM 111: From XRAM to UART2. This bit cannot be configured when channel-1 is busy.
[2]	DBGSW	DEBUG mode address selection 0: DEBUG mode address is XSFR (0x4020~0x40FF) 1: DEBUG mode address is XRAM (0x0000~0x0317)
[1]	DBGEN	DEBUG mode enable 0: Normal mode 1: DEBUG mode When CH1CFG = 101 and DBG_EN = 1, DMA enables DEBUG mode. Enable SPI, and it is configured as 3-wire master mode, which only sends data and MISO pin is not used. DMA module send data, in DBG_SW address, by SPI every FOC interrupt. The DBG_SW can be configured by setting CHABA/CH1LEN. NSS is low voltage level during the transmission process. When data is already sent, NSS is turned to high voltage level and prepare for the next transmission.
[0]	DMAIF	DMA channel-1 transmitting completed interrupt flag. This bit is set to 1 by hardware and cleared by software.

		0: No event 1: Channel-1 transmitting completed. (When INTEN = 1, an interrupt event is triggered. Otherwise, only a flag is generated)

26.2.3 DMA0_LEN (0x403C)

Table 26-3 DMA0_LEN (0x403C)

Bit	7:6	5:0
Name	Reserved	DMA0_LEN
Type	R	R/W
Reset	00	00000

Bit	Name	Function
[7:6]	RSV	Reserved
[5:0]	DMA0_LEN	Writing: XRAM data length of DMA channel-0 These bits cannot be written when DMA channel-0 is busy. It is recommended that to set DMA0_LEN to an odd number when ENDIAN = 1. Reading: The number of byte, which is transmitting by DMA channel-0 (0 represents the first byte).

26.2.4 DMA0_BARR (0x403E, 0x403F)

Table 26-4 DMA0_BARRH (0x403E)

Bit	7:3	2:0
Name	Reserved	DMA0_BARR[10:8]
Type	R	R/W
Reset	0	0

Table 26-5 DMA0_BARRL (0x403F)

Bit	7:0
Name	DMA0_BARR[7:0]
Type	R/W
Reset	8'h00

Bit	Name	Function
[10:0]	DMA0_BARR	XRAM first address of DMA channel-0. It cannot be configured when channel-0 is busy. Note: XRAM address area of DMA channel-0 is: DMA0_BARR [10:0]~(DMA0_BARR[10:0] + DMA0_LEN[5:0]).

26.2.5 DMA1_LEN (0x403D)

Table 26-6 DMA1_LEN (0x403D)

Bit	7:6	5:0
Name	Reserved	DMA1_LEN
Type	R/W	R/W
Reset	00	00000

Field	Name	Description
[7:6]	RSB	Reserved
[5:0]	DMA1_LEN [5:0]	Writing: XRAM data length of DMA channel-1 These bits cannot be written when DMA channel-1 is busy. It is recommended that to set DMA1_LEN to an odd number when ENDIAN = 1. Reading: The number of byte, which is transmitting by DMA channel-1 (0 represents the first byte).

26.2.6 DMA1_BARR (0x4040, 0x4041)

Table 26-7 DMA1_BARRH (0x4040)

Bit	7:3	2:0
Name	Reserved	DMA1_BARR[10:8]
Type	R	R/W
Reset	0	0

Table 26-8 DMA1_BARRL (0x4041)

Bit	7:0
Name	DMA1_BARR[7:0]
Type	R/W
Reset	0x00

Field	Name	Description
[10:0]	DMA1_BARR	XRAM first address of DMA channel-1. It cannot be configured when channel-1 is busy. Note: XRAM address area of DMA channel-1 is: DMA1_BARR [10:0]~(DMA1_BARR[10:0] + DMA1_LEN[5:0]).

When DMA peripheral is selected as I2C (including from I2C to XRAM and XRAM to I2C), STR is cleared by software. When I2C work as slave mode and receives a STOP signal, I2C_SR[STOP] is cleared by software to restart DMA transmission.

27 VREF

27.1 VREF Module Operating Instructions

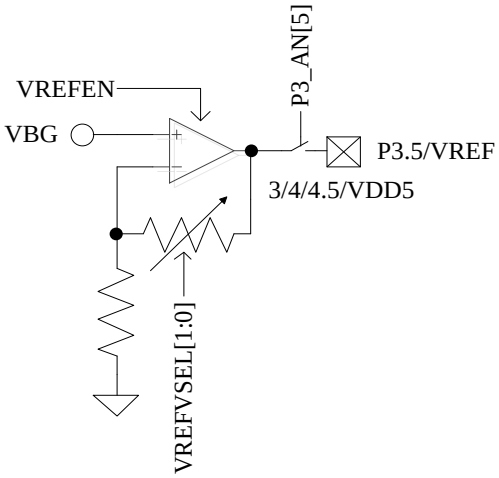


Figure 27-1 VREF Module Port Declarations

FU6813/63 has a reference voltage module. The port declaration of this module is shown in Figure 27-1.

For proper function of the module, Setting register as follows: VREFEN = 1 and VREFVSEL, which configures output voltage. For more detail, please refer to Table 27-1. To output VREF, set P3_AN[5] to 1. The VREF is used as the internal voltage reference of ADC and DAC.

27.2 VREF Module Registers

27.2.1 VREF_VHALF_CR (0x404F)

Table 27-1 VREF_VHALF_CR (0x404F)

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

Field	Name	Description
7:6	VREFVSEL	VREF module voltage selection 00: 4.5V 01: VDD5 10: 3V 11: 4V
5	RSV	Reserved
4	VREFEN	VREF module enable

		0: Disable. When P3_AN[5] = 1, the external reference input from P3.5. 1: Enable. When P3_AN [5] = 1, P3.5 output VREF. Shunt a 0.1~1μF capacitance to ground to stabilize voltage.
3:1	RSV	Reserved
0	VHALFEN	VHALF enable 0: Disable 1: Enable

28 VHALF

28.1 VHALF Module Operating Instruction

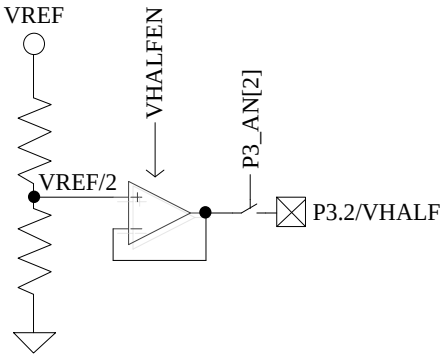


Figure 28-1 VHALF Module Port Declarations

The port declaration of this module is shown in Figure 28-1.

For proper function of the module, Setting register as follows: VHALFEN = 1 and P3_AN[2] = 1. P3.2 outputs VHALF.

28.2 VHALF Module Register

For VHALF module register, please refer to Table 27-1.

29 Operation Amplifier

FU6813/63 has 4 integrated high-speed independent OP Amps, namely AMP0, AMP1, AMP2 and AMP3, respectively. Each Op Amp has independent enable port. FU6813N has 2 Op Amps.

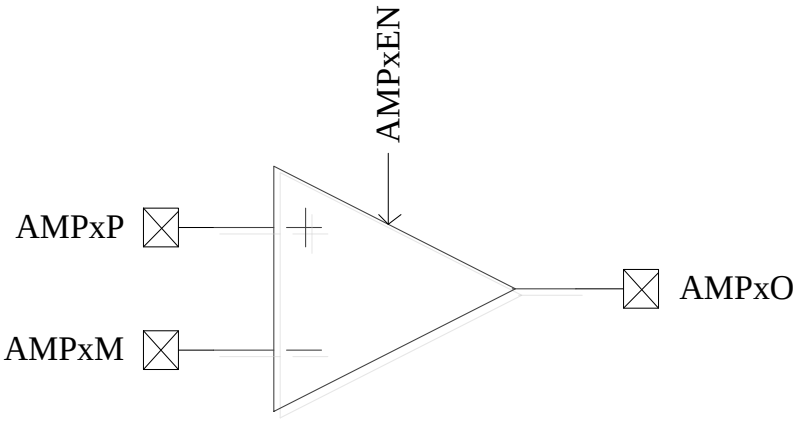


Figure 29-1 Diagram of Op Amp Module

29.1 Op Amps Operation Descriptions

29.1.1 Bus Current Op Amp (AMP0)

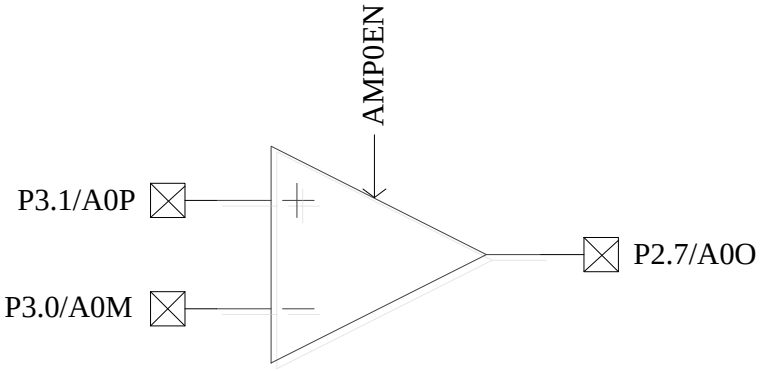


Figure 29-2 AMP0 Ports Connected with Pad

For proper functioning of the block, the AMP0EN bit must be set to 1.

The AMP0 ports connected with pad are shown in Figure 29-2. The GPIO connected to AMP0 should be set as analog mode before enable AMP0 (i.e., Setting P2_AN[7] = 1, P3_AN[0] = 1 and P3_AN[1] = 1).

29.1.2 Phase Current Op Amps (AMP1/AMP2)

29.1.2.1 AMP1

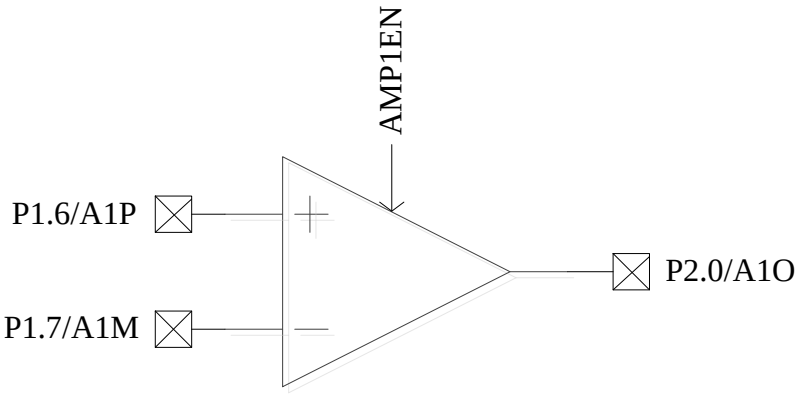


Figure 29-3 AMP1 Ports Connected with Pad

For proper functioning of the block, the AMP1EN bit must be set to 1.

The AMP1 ports connected with pad are shown in Figure 29-3. The GPIO connected to AMP1 should be set as analog mode before enable AMP1 (i.e., Setting P1_AN[6] = 1, P1_AN[7] = 1 and P2_AN[0] = 1).

29.1.2.2 AMP2

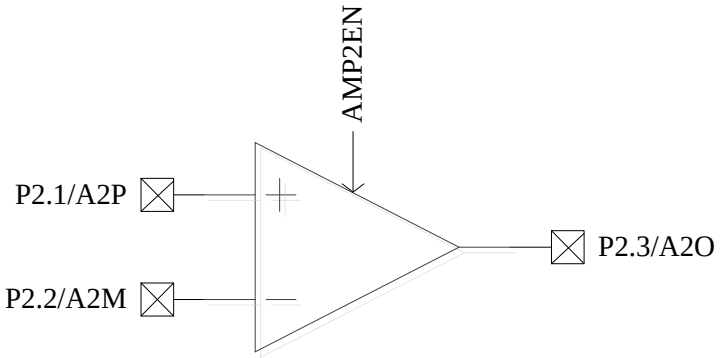


Figure 29-4 AMP2 Ports Connected with Pad

For proper functioning of the block, the AMP2EN bit must be set to 1.

The AMP2 ports connected with pad are shown in Figure 29-4. The GPIO connected to AMP2 should be set as analog mode before enable AMP2 (i.e., Setting P2_AN[1] = 1, P2_AN[2] = 1 and P2_AN[3] = 1).

29.1.3 AMP3

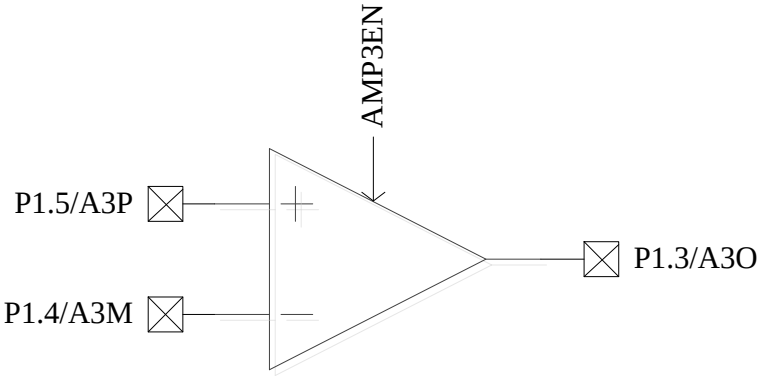


Figure 29-5 AMP3 Ports Connected with Pad

For proper functioning of the block, the AMP3EN bit must be set to 1.

The AMP3 ports connected with pad are shown in Figure 29-4. The GPIO connected to AMP3 should be set as analog mode before enable AMP3 (i.e., Setting P1_AN[3] = 1, P1_AN[4] = 1 and P1_AN[5] = 1).

29.2 Op Amp Registers

29.2.1 AMP_CR (0x404E)

Table 29-1 AMP_CR (0x404E)

Bit	7	6	5	4	3	2	1	0
Name	RSV				AMP3EN	AMP2EN	AMP1EN	AMP0EN
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Field	Name	Description
[7:4]	RSV	Reserved
[3]	AMP3EN	AMP3 enable
[2]	AMP2EN	AMP2 enable
[1]	AMP1EN	AMP1 enable
[0]	AMP0EN	AMP0 enable

30 Comparator

30.1 Comparator Operation Instructions

30.1.1 Comparator CMP3

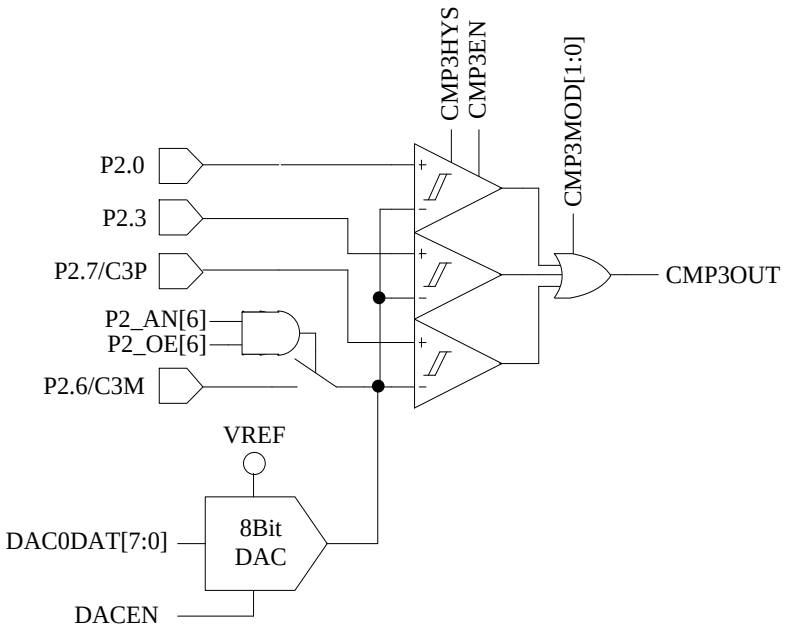


Figure 30-1 Schematic Diagram of CMP3

The CMP3 ports connected with pad is shown in Figure 30-1.

To enable CMP3, configure the following steps:

1. Configure negative input reference voltage. The reference could be the Output DAV or external supply.
2. Configure CMP3MOD to select single/dual/triple-input modes
3. Configure CMP3HYS to enable/disable the hysteresis function.
4. Configure CMP_CR2[CMP3EN] to enable CMP3.

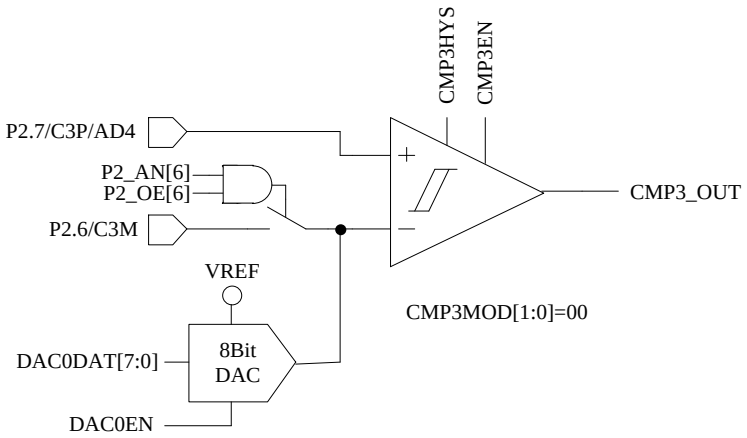


Figure 30-2 Set CMP3MOD[1:0] to 00 to Select Single-Input Mode

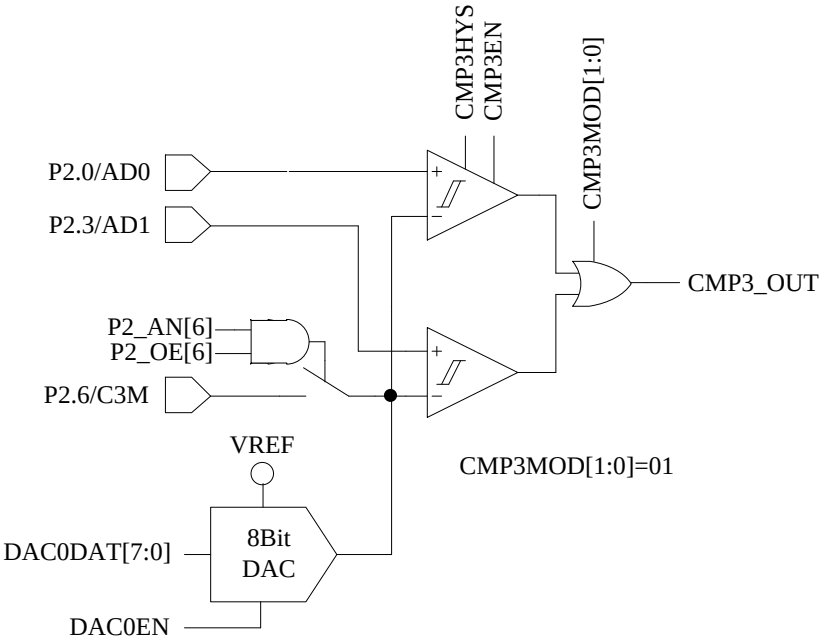


Figure 30-3 Set CMP3MOD[1:0] to 01 to Select Dual-Input Mode

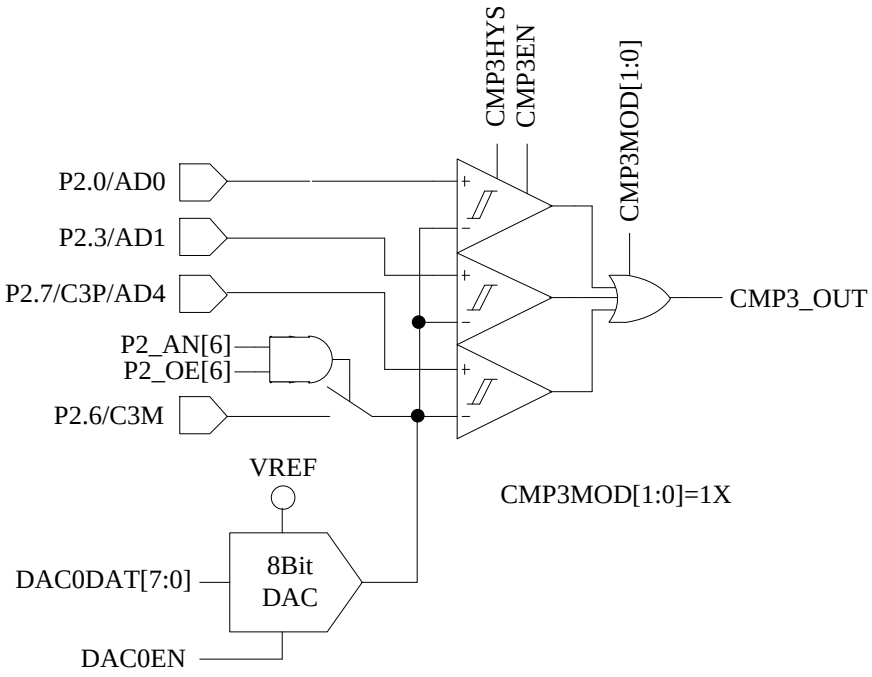


Figure 30-4 Set CMP3MOD[1:0] to 1X to Select Triple-Input Mode

30.1.1.1 Bus Current Protection

For protection of the chip and motor, BUS current protection will disable the outputs to motor based on the generation of BUS current protection signal. Set EVT_FILT[MOEMD] to 01 will produce the

protection event. If EVT_FILT[MOEMD] is set to 00, it will not disable outputs but trigger an interrupt.

Bus current protection event can be produced by CMP3 or an external interrupt INT0 (P0.0) by the configuration of EVT_FILT[EFSRC]. Assuming that the chip connects to an IPM module with the FALUT signal connected to P0.0, BUS current protection event will be produced by external interrupt INT0 by setting EVT_FILT[EFSRC] to 1. In this case, the protection interrupt is external interrupt INT0. Protection signal can also be produced by comparing the sampling voltage of the BUS to a predefined voltage, and protection interrupt triggered by CMP3.

The input signal of BUS current protection can be filtered by the configuration of EVT_FILT[EFDIV]. Filter width is selected as 4/8/16. After enabling the Filtering function, the signal will delay about 4~5/8~9/16~17 clock cycles.

30.1.1.2 Cycle-by-Cycle Current Limiting

The cycle-by-cycle current limiting is used in the application of square wave current based BLDC control. A current protection event will disable the outputs of Driver. When EVT_FILT[MOEMD] = 10, MOE is enabled by an DRV counter overflow event. When EVT_FILT[MOEMD] = 10, MOE is enabled by an DRV counter overflow/underflow event. Moreover, MOE will be enabled automatically every 5μs.

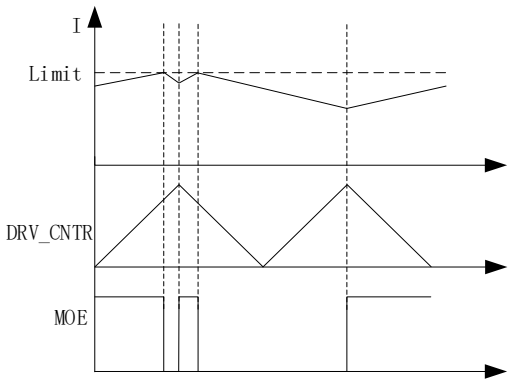


Figure 30-5 Cycle-by-Cycle Current Limiting When MOEMD = 10

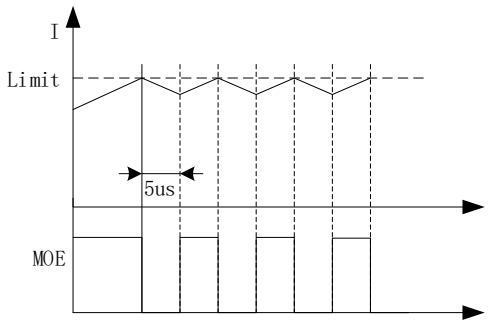


Figure 30-6 Cycle-by-Cycle Current Limiting When MOEMD = 11

30.1.2 Comparator CMP4

CMP4 is a hysteresis comparator as shown in Figure 30-7. CMP4OUT cannot be read by software directly. The optional method is using external interrupt to monitor the Output CMP4. When CMP4 is enabled, CMP3MOD[1:0] must not be 01. Generally, CMP4 is used with CMP3 in the application of cycle-by-cycle current limiting in BLDC control.

To enable CMP4, configure the following steps:

1. Set P2_AN[3] and P2_AN[7] to 1 to enable P2.3, P2.7 analog mode..
2. Set CMP4EN to 1 to enable CMP4.
3. Clear external interrupt flag to enable external interrupt INT0.
4. When the Output CMP4 turn over from 0 to 1, external interrupt is triggered.

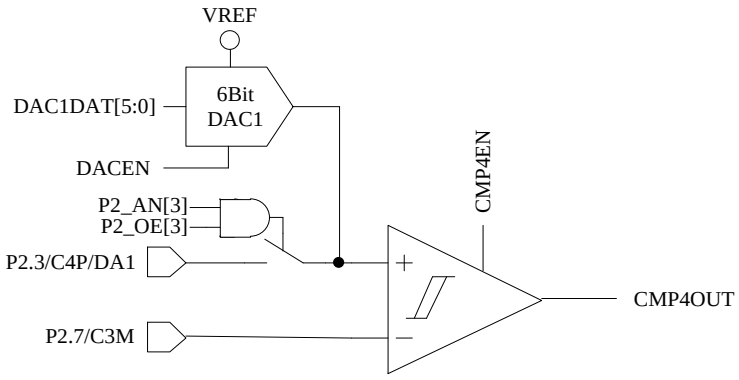


Figure 30-7 Schematic Diagram of CMP4

30.1.3 Comparator CMP5

CMP5 is a hysteresis comparator as shown in Figure 30-8 Schematic Diagram of CMP5. CMP5OUT can be read by software and is used in the application of cycle-by-cycle current limiting and over current protection of PFC sampling current.

Setting P1_AN[3] = 1, the positive input port of CMP5 is connected to P1.3.

The negative input port of CMP5 is selected as P1.5 or the Output DAC2.

1. Setting P1_AN[5] = 1 and P1_OE[5] =1, P1.5 is the negative input of CMP5.
2. To select the Output DAC2, enable DAC2 and set DAC2DAT to configure DAC output voltage.

Set CMP5EN to1 to enable CMP5.

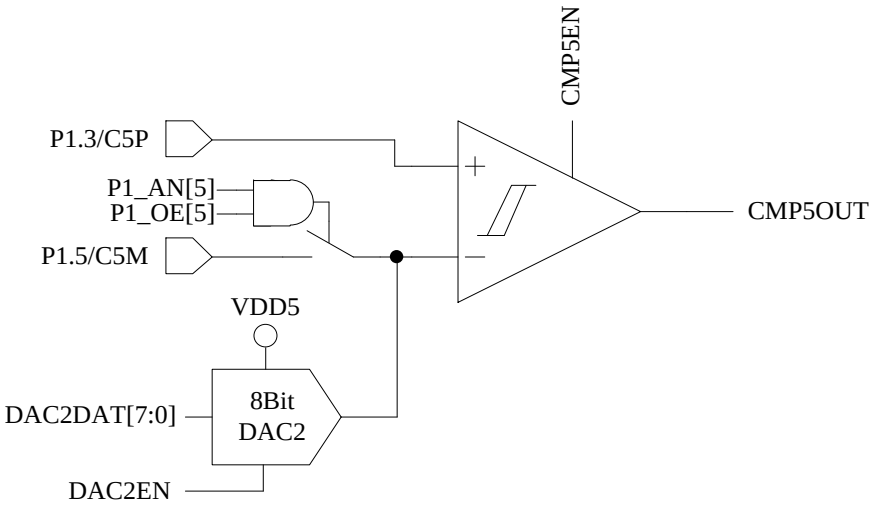


Figure 30-8 Schematic Diagram of CMP5

30.1.4 Comparator CMP0

CMP0 has several compare mode can is used in the application of motor position and speed detection.

When CMP0MOD[1:0] = 00, CMP0 works in none internal resistance based triple-comparator mode. As shown in Figure 30-9, this mode is used for BEMF detection of motors with off-chip virtual center point. The negative input is connected to P1.5 and the positive inputs are connected to P1.4, P1.6 and P2 .1. CMP0OUT, CMP1OUT and CMP2OUT update the corresponding outputs.

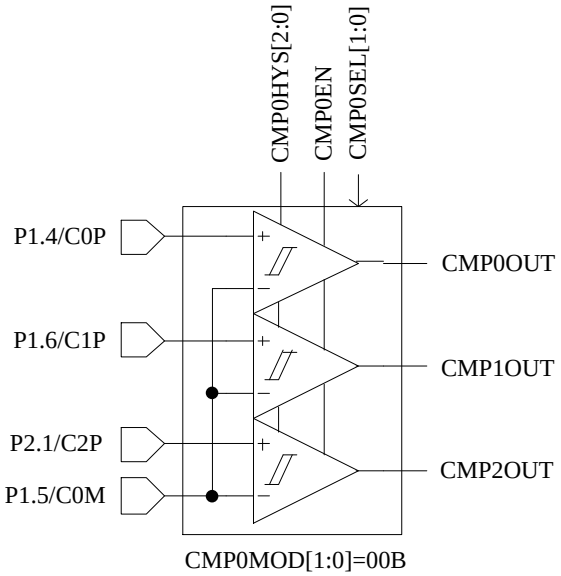


Figure 30-9 CMP0MOD[1:0] = 00, None Internal Resistance Based Triple-Comparator Mode.

When CMP0MOD[1:0] = 01, CMP0 works in internal resistance based triple-comparator mode. This mode is used for BEMF detection of motors with on-chip virtual center point. The input sources are configured by CMP0FS. When CMP0FS = 0, the port declaration is shown in Figure 30-10. When CMP0FS = 1, the port declaration is shown in Figure 30-11.

When CMP0MOD[1:0] = 01 and CMP0FS = 0, The negative input is connected to internal resistance center point and the positive inputs are connected to P1.4, P1.6 and P2 .1. CMP0OUT, CMP1OUT and CMP2OUT update the corresponding outputs.

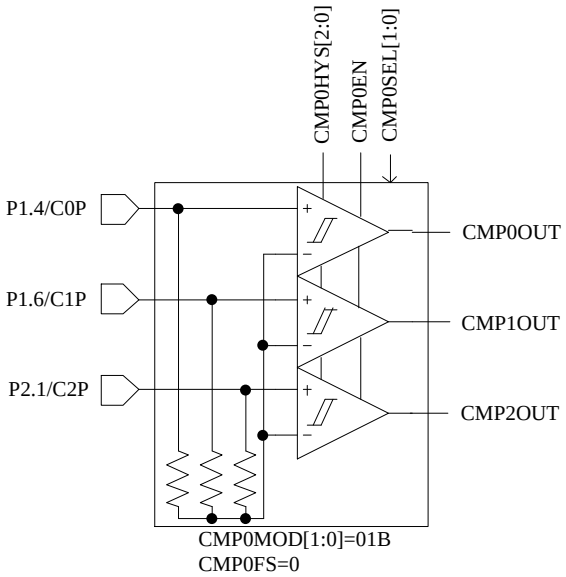


Figure 30-10 CMP0MOD[1:0] = 01 and CMP0FS = 0, Internal Resistance Based Triple-Comparator Mode.

.

When CMP0MOD[1:0] = 01 and CMP0FS = 1, The negative input is connected to internal resistance center point and the positive inputs are connected to P1.4, P1.3 and P1.5. CMP0OUT, CMP1OUT and CMP2OUT update the corresponding outputs.

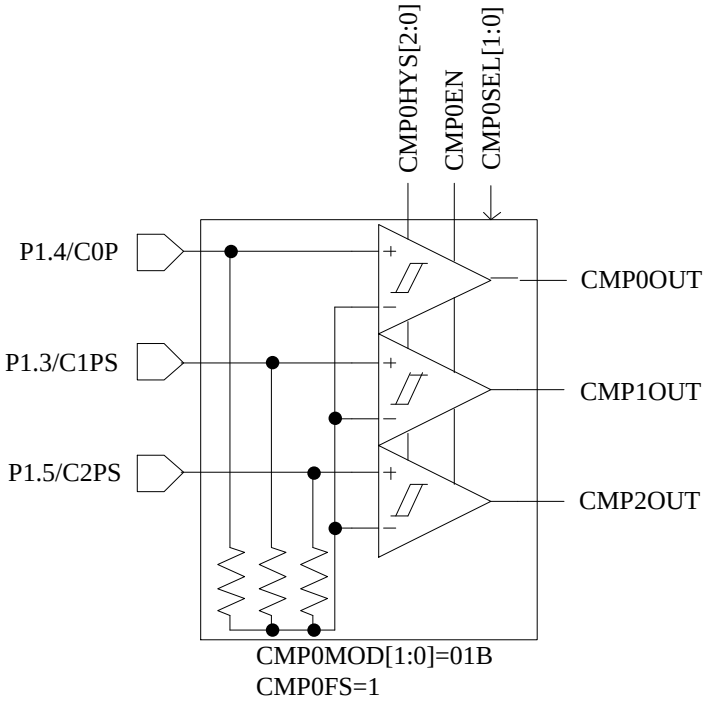


Figure 30-11 CMP0MOD[1:0] = 01 and CMP0FS = 1, Internal Resistance Based Triple-Comparator

Mode.

When CMP0MOD[1:0] = 10, CMP0 works in differential triple-comparator mode as shown in Figure 30-12. This mode is used in the application of position detection with differential HALL input. The negative inputs are connected to P1.5, P1.7 and P2.2, and the positive inputs are connected to P1.4, P1.6 and P2.1. CMP0OUT, CMP1OUT and CMP2OUT update the corresponding outputs.

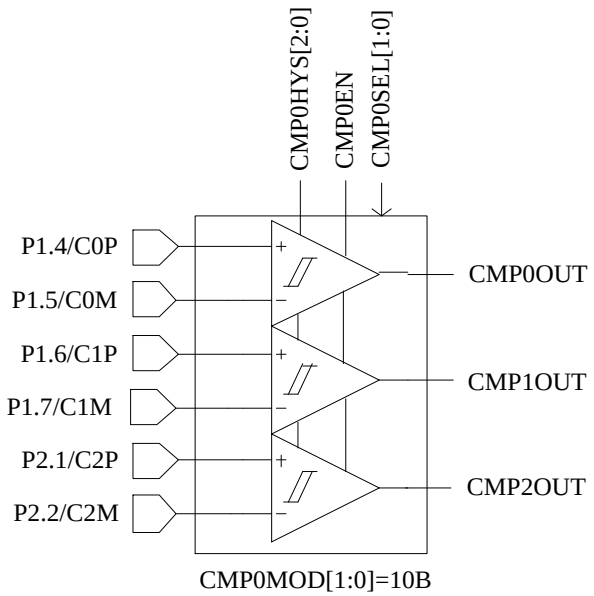


Figure 30-12 CMP0MOD[1:0] = 10, Differential Triple-Comparator Mode.

When CMP0MOD[1:0] = 11, CMP0 works in dual-comparator mode as shown in Figure 30-13. The negative input is connected to P1.5 and the positive inputs are connected to P1.4 and P1.3. CMP0OUT and CMP1OUT update the corresponding outputs.

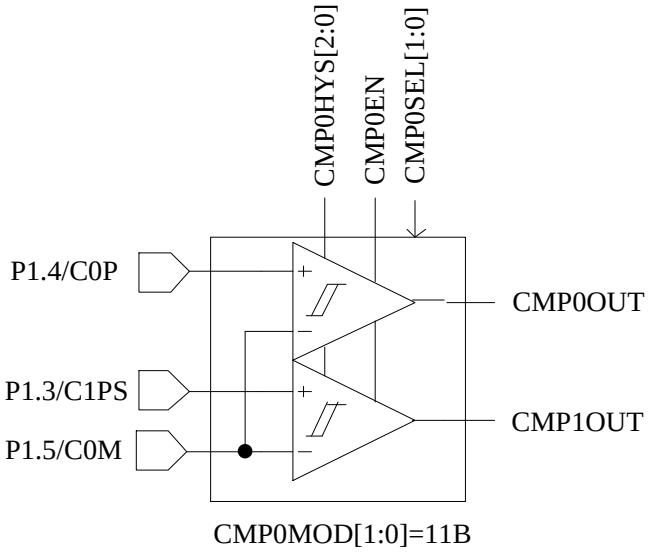


Figure 30-13 CMP0MOD[1:0] = 11, Dual-Comparator Mode.

To work in the HALL mode. Configure BEMFREN = 0. The positive and negative inputs of 3 comparators is connected to GPIO.

To work in the BEMF mode. Configure BEMFREN = 1. The negative inputs of 3 comparators are connected with resistances. The BEMF signal of phase U/V/M are connected with the positive inputs of CMP0/CMP1/CMP2. The negative input corresponding GPIO is disconnected and can be used for other functions.

The outputs of CMP0/CPM1/CPM2 is sent to TIMER1 through sample and filter module.

CMP0/CPM1/CPM2 share unique enable bit CMP0EN and unique hysteresis voltage selection bit CMP0HYS.

30.1.5 Comparator Sampling

The sampling time configuration of comparator is mainly used for BLDC control and RSD, which is used to sample comparator output without the disturbance of power device commutation. Please refer to section 16.1.2.2 for BLDC control. And section 17.1.7.1 for RSD.

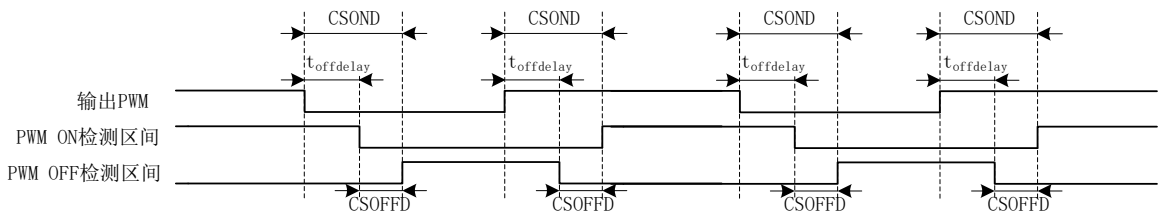


Figure 30-14 Diagram of Sampling Time

There is a delay from the PWM output effect to comparator. And the time is mainly decided by the

followings: the resistance of driver, turn on delay time and turn off delay time of power device, input delay of comparator and hysteresis configuration. The sampling of comparator will be disabled due to this effect, and the time is configurable by CSOFFD and CSOND. The disable time of CMP0, CMP1 and CMP2 will be delayed for off delay = CSOND – CSOFFD.

e.g., providing the delay from PWM output effect to comparator is 2μs and the time of effect is 1μs, the CSOFFD and CSOND can be configured as

CSOFFD > 1μs = 1000ns / (41.67ns × 8) = 3

CSOND > (2 + 1) μs = 3000 / (41.67ns × 8) = 9

To measure the delay from PWM output effect to the comparator, the following steps could be implemented. Setting CMP_CR3[SAMSEL] = 00 disable sampling delay. Configure CMP_CR3[CMPSEL] to output the corresponding comparator. PWM output and comparator is enabled. Users rotate motor to update the comparator output and the delay between PWM output and comparator output is obtained.

To measure the width of comparator output distortion, the above method could be used.

30.1.6 Comparator Output

The outputs of 4 comparators are connected to the multiplexer. Users can configure CMP_CR2[CMPSEL] to select the Output a comparator to the corresponding GPIO.

30.2 Comparator Register

30.2.1 CMP_CR0 (0xD5)

Table 30-1 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	
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:6]	CMP3IM	Comparator CMP3 Interrupt Mode Refer to the description of CMP0IM
[5:4]	CMP2IM	Comparator CMP2 Interrupt Mode Refer to the description of CMP0IM
[3:2]	CMP1IM	Comparator CMP1 Interrupt Mode Refer to the description of CMP0IM
[1:0]	CMP0IM	Comparator CMP0 Interrupt Mode 00: No interrupt generation 01: Interrupt generation at rising edge 10: Interrupt generation at falling edge 11: Interrupt generation at both rising and falling edge

30.2.2 CMP_CR1 (0xD6)

Table 30-2 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		
Reset	0	0		0	0	0	0	0

Bit	Name	Function
[7]	HALLSEL	HALL signal input selection 0: P0.2, P3.7, P3.6 1: P1.4, P1.6, P2.1
[6:5]	CMP3MOD	CMP3 positive input selection as in Figure 30-1. The negative input is connected to P2.6 or DAC output. 00: Single-input mode as shown in Figure 30-2 01: Dual-input mode as shown in Figure 30-3 1X: Triple-input mode as shown in Figure 30-4
[4]	CMP3EN	CMP3 enable 0: Disable 1: Enable
[3]	CMP3HYS	CMP3 hysteresis enable 0: Disable 1: Enable
[2:0]	CMP0HYS	CMP0 hysteresis voltage selection

CMP0HYS	Hysteresis voltage
000	No hysteresis
001	±2.5mV
010	-5mV
011	+5mV
100	±5mV
101	-10mV
110	+10mV
111	±10mV

30.2.3 CMP_CR2 (0xDA)

Table 30-3 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		R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	CMP4EN	CMP4 enable

<table>
<tr><td></td><td></td><td>0: Disable
1: Enable</td></tr>
<tr><td>[6:5]</td><td>CMP0MOD</td><td>CMP0/1/2 mode selection
<table>
<tr><td>CMP0MOD</td><td>Mode</td></tr>
<tr><td>00</td><td>None internal resistance based triple-comparator mode as shown in Figure 30-9.</td></tr>
<tr><td>01</td><td>Internal resistance based triple-comparator mode.
When CMP0FS = 0, please refer to Figure 30-10
When CMP0FS = 1, please refer to Figure 30-11</td></tr>
<tr><td>10</td><td>Differential triple-comparator mode as shown in Figure 30-12</td></tr>
<tr><td>11</td><td>Dual-comparator mode, and CMP2 is disabled. Please refer to Figure 30-13</td></tr>
</table></td></tr>
<tr><td>[4:3]</td><td>CMP0SEL</td><td>CMP0 port selection, used with CMP0MOD. The default value is 00. In BLDC mode, Timer 1 will control CMP0SEL automatically. Please refer to section 20.1.2.3
<table>
<tr><td>CMP0MOD</td><td>CMP0SEL</td><td>Function Description</td></tr>
<tr><td rowspan="4">00</td><td>00</td><td>All CMP0/1/2 are enabled. The negative inputs of 3 comparator are connected to C0M. C0P, C1P and C2P are compared with C0M by hardware and the outputs are updated in CMP0OUT, CMP1OUT and CMP2OUT.</td></tr>
<tr><td>01</td><td>CMP0 is enabled. The positive input is connected to C0P. The negative input is connected to C0M and the output is updated in CMP0OUT.</td></tr>
<tr><td>10</td><td>CMP1 is enabled. The positive input is connected to C1P. The negative input is connected to C0M and the output is updated in CMP1OUT.</td></tr>
<tr><td>11</td><td>CMP2 is enabled. The positive input is connected to C2P. The negative input is connected to C0M and the output is updated in CMP2OUT.</td></tr>
<tr><td>01</td><td>00</td><td>All CMP0/1/2 are enabled. The negative inputs of 3 comparator are connected to internal resistance center point. When CMP0FS = 0, C0P, C1P and C2P are compared with C0M by hardware. When</td></tr>
</table></td></tr>
</table>

					CMP0FS = 1, C0P, C1PS and C2PS are compared with C0M. The outputs are updated in CMP0OUT, CMP1OUT and CMP2OUT.
				01	The positive input of CMP0 is connected to C0P and the negative input is connected to internal resistance center point. The output is updated to CMP0OUT.
				10	The positive input of CMP0 is connected to C1P (CMP0FS = 0) or C1PS (CMP0FS = 1). The negative input is connected to internal resistance center point and the output is updated to CMP1OUT.
				11	The positive input of CMP0 is connected to C2P (CMP0FS = 0) or C2PS (CMP0FS = 1). The negative input is connected to internal resistance center point and the output is updated to CMP2OUT.
			10	00	All CMP0/1/2 are enabled. The negative inputs of 3 comparator are connected to C0P, C1P and C2P. The negative inputs are connected to C0M, C1M and C2M. The outputs are updated in CMP0OUT, CMP1OUT and CMP2OUT.
				01	The positive input of CMP0 is connected to C0P and the negative input is connected to C0M. The output is updated to CMP0OUT.
				10	The positive input of CMP0 is connected to C1P and the negative input is connected to C1M. The output is updated to CMP1OUT.
				11	The positive input of CMP0 is connected to C2P and the negative input is connected to C2M. The output is updated to CMP2OUT.
			11	00	CMP0/1 are enabled. The negative inputs of 2 comparator are connected to C0P, C1PS. The negative inputs are connected to C0M. The outputs are updated in CMP0OUT, CMP1OUT.
				01	The positive input of CMP0 is connected to C0P and the negative input is connected to C0M. The output is updated to CMP0OUT.
				10	The positive input of CMP0 is connected to C1PS and the negative input is connected to C0M. The output is updated to CMP1OUT.

			11	Reserved
[2:1]	RSV	Reserved		
[0]	CMP0EN	CMP0 enable 0: Disable 1: Enable		

30.2.4 CMP_CR3 (0xDC)

Table 30-4 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		
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	CMPDTEN	Comparator deadtime sampling enable 0: Disable 1: Enable
[6:5]	DBGSEL	DEBUG signal selection Select a DEBUG signal to P0.1 00: Disable DEBUG 01: Diode freewheeling end signal and zero crossing signal in the application of square wave current based BLDC control 10: ADC trigger signal 11: Comparator sampling interval
[4:3]	SAMSEL	CMP0, CMP1, CMP2 and ADC sampling timing selection 00: Sampling in the ON and OFF state, without sampling delay. 01: Sampling in the OFF state, with a predefined sampling delay configured by CMP_SAMR. 10: Sampling in the ON state, with a predefined sampling delay configured by CMP_SAMR. 11: Sampling in the ON and OFF state，with a predefined sampling delay configured by CMP_SAMR.
[2:0]	CMPSEL	Comparator output selection Select a comparator signal output to the port. 000：Disable 001：CMP0 010：CMP1 011：CMP2 100：CMP3 101：CMP4 110：CMP5 111：OMEGA start flag

30.2.5 CMP_CR4 (0xE1)

Table 30-5 CMP_CR4 (0xE1)

Bit	7	6	5	4	3	2	1	0
Name	CMP4OUT	CMP5OUT	RSV				CMP0FS	CMP5EN
Type	R	R	R				R/W	R/W
Reset	1	0	0				0	0

Bit	Name	Function
[7]	CMP4OUT	CMP4 compare result
[6]	CMP5OUT	CMP5 compare result
[5:2]	RSV	Reserved
[1]	CMP0FS	CMP1, CMP2 function transfer enable: 0: Disable 1: Enable. Only valid when CMP0_MOD = 01. Otherwise, this bit is invalid.
[0]	CMP5EN	CMP5 enable 0: Disable 1: Enable

30.2.6 CMP_SAMR (0x40AD)

Table 30-6 CMP_SAMR (0x40AD)

Bit	7	6	5	4	3	2	1	0
Name	CMP_SAMR							
Type	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	Function
[7:4]	CSOND	CMP0, CMP1 and CMP2 sampling start delay time. The commutation of power device will disturb the Output comparator. Set CSOND to configure a predefined sampling start delay time to avoid the effect of power device and the delay time generated by driver circuit should also be considered. If MCU clock is 24MHz (41.67ns) Sampling start delay time = CSOND × 41.67ns × 8 Note: CSOND must be larger than CSOFFD. Please refer to section 16.1.2.2 for BLDC control and section 17.1.7.1 for RSD.
[3:0]	CSOFFD	CMP0, CMP1 and CMP2 sampling disabled time. The commutation of power device will disturb the Output comparator. Set CSOFFD to configure a predefined sampling disabled time to

		avoid the effect of power device. If MCU clock is 24MHz (41.67ns) Sampling disabled time = CSOFFD× 41.67ns × 8 Note: CSOND must be larger than CSOFFD. Please refer to section 16.1.2.2 for BLDC driver and section 17.1.7.1 for RSD.

30.2.7 CMP_SR (0xD7)

Table 30-7 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	Function
[7]	CMP3IF	CMP3 interrupt flag CMP3 interrupt event is generated. This bit is set to 1 by hardware and cleared by software. 0: No event 1: A CMP3 Interrupt event
[6]	CMP2IF	CMP2 interrupt flag CMP2 interrupt event is generated. This bit is set to 1 by hardware and cleared by software. 0: No event 1: A CMP2 Interrupt event
[5]	CMP1IF	CMP1 interrupt flag CMP1 interrupt event is generated. This bit is set to 1 by hardware and cleared by software. 0: No event 1: A CMP1 Interrupt event
[4]	CMP0IF	CMP0 interrupt flag CMP0 interrupt event is generated. This bit is set to 1 by hardware and cleared by software. 0: No event 1: A CMP0 Interrupt event
[3]	CMP3OUT	CMP3 compare result
[2]	CMP2OUT	CMP2 compare result
[1]	CMP1OUT	CMP1 compare result
[0]	CMP0OUT	CMP0 compare result

30.2.8 EVT_FILT (0xD9)

Table 30-8 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	
Reset	0	1	1	0	0	0	0	0

Bit	Name	Function
[7]	TSDEN	Temperature sensor detection enable. 0: Disable 1: Enable
[6:5]	TSDADJ	Temperature sensor detection adjustment. 00: 105℃ 01: 120℃ 10: 135℃ 11: 150℃
[4:3]	MOEMD	MOE signal mode These bit select the clear and enable mode of MOE when a BUS overcurrent event is generated. 00: MOE is not cleared by hardware. 01: MOE is cleared by hardware. 10: MOE is cleared by hardware and enabled in an overflow event of DRV counter (used for square wave current based BLDC control). 11: MOE is cleared by hardware and enabled in an overflow/underflow event of DRV counter. MOE will also be enabled every 5μs (used for square wave current based BLDC control).
[2]	EFSRC	Filter module input source for BUS current protection 0: CMP3 1: External interrupt 0 (P0.0)
[1:0]	EFDIV	Filtering width of Bus current protection 00: No filtering 01: 4 system clock periods 10: 8 system clock periods 11: 16 system clock periods

31 Supply

31.1 LDO

31.1.1 LDO Operation Instructions

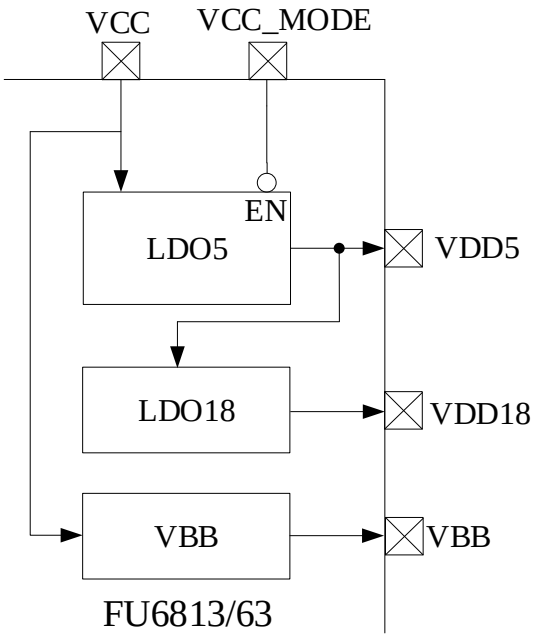


Figure 31-1 Block Diagram of Supply

As shown in Figure 31-1, it lists the supply and LDO relative pins. MCU has a 5V LDO (supply analog module) and a 1.8V LDO (supply digital module). VDD5 is select as internal LDO5 or external supply by the configuration of VCC_MODE. VBB can be disabled by the IDE tool as shown in Figure 31-2 and is enabled by default, which could supply Pre-driver. Providing VBB is enabled, when VCC > 15V, VBB = 15V, and when VCC ≤ 15V, VBB = VCC.

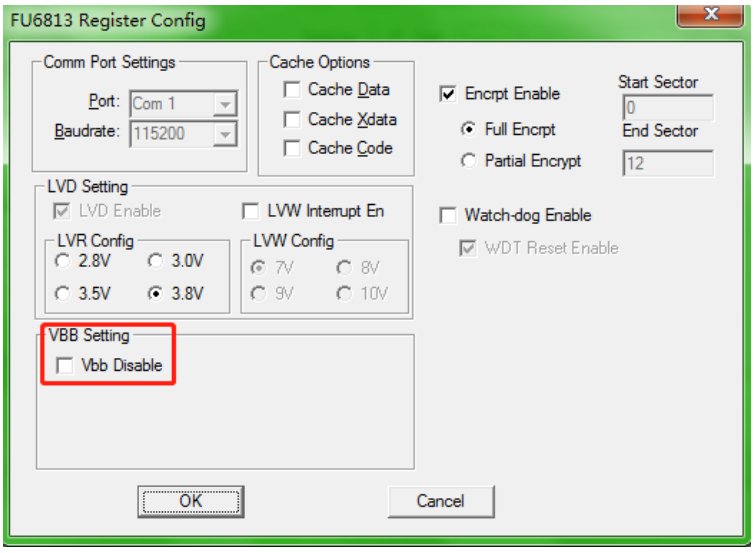


Figure 31-2 VBB Configuration

FU6813:

Single Supply HV Mode (VCC_MODE = 0). VCC = 5~24V. Please refer to Figure 31-3.

Dual Supply Mode (VCC_MODE = 1), VCC ≥ VDD5. VCC = 5~36V, VDD5 = 5V. Please refer to Figure 31-4.

Single Supply LV Mode (VCC_MODE = 1). VCC = VDD5 = 3~5.5V. Please refer to Figure 31-5.

FU6863:

Mode 1: VCC_MODE = 0, VCC = 5~24V, VDRV = 7~18V

Mode 2: VCC_MODE = 1, VCC = VDD5 = 3~5.5V, VDRV = 7~18V

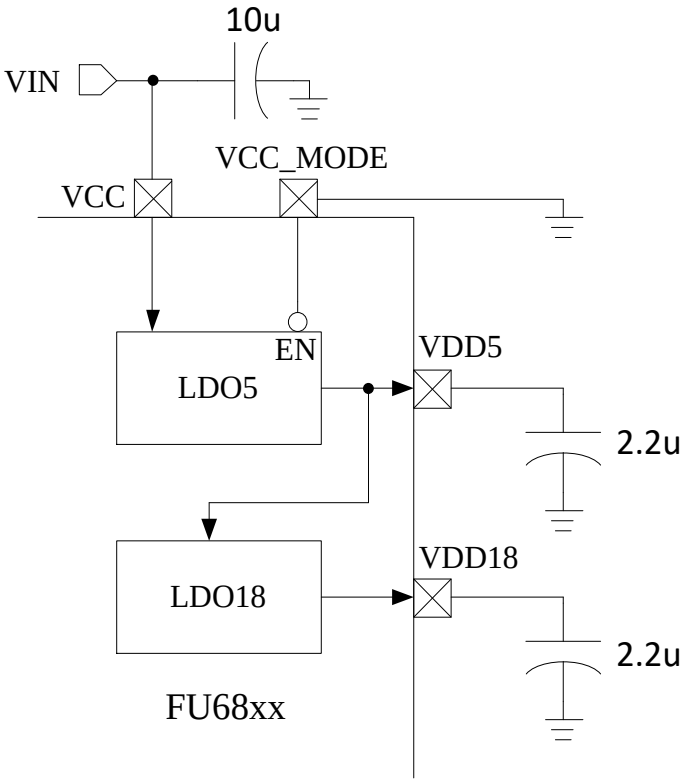


Figure 31-3 Single Supply HV Mode Connection

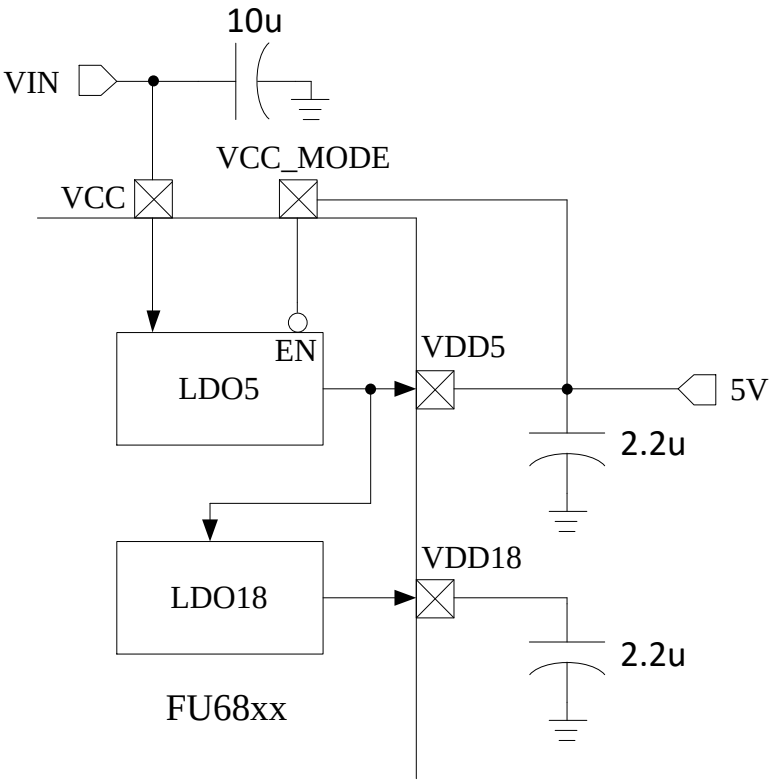


Figure 31-4 Dual Supply Mode Connection

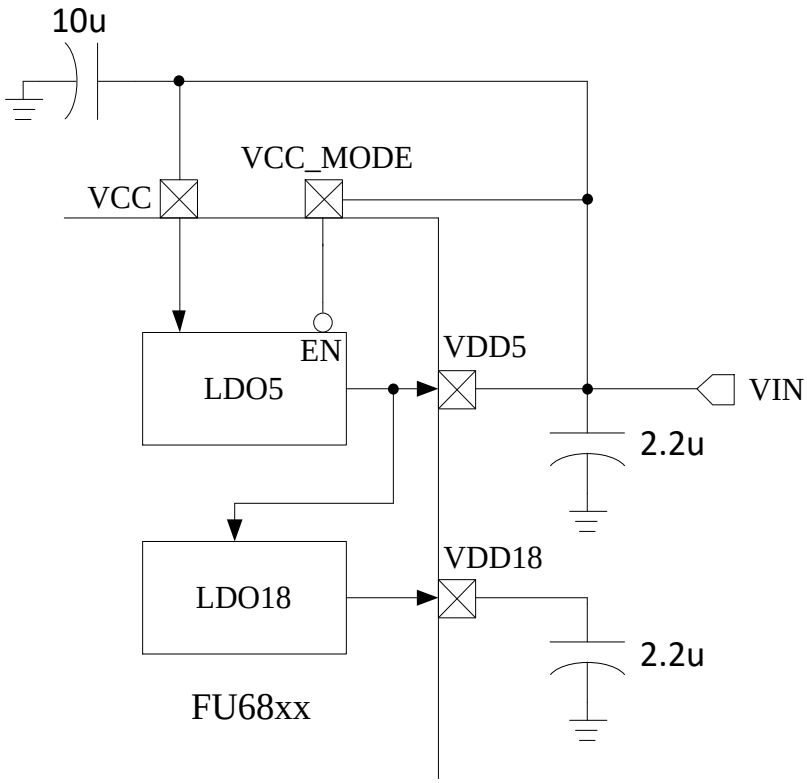


Figure 31-5 Single Supply LV Mode Connection

31.2 Low Voltage Detection

31.2.1 Low Voltage Detection Operation Instructions

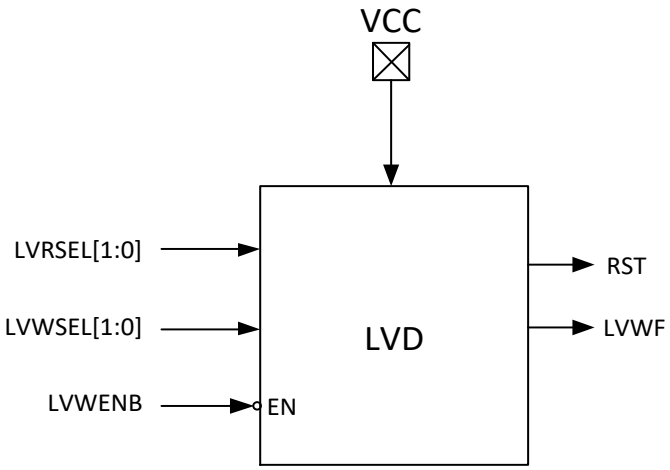


Figure 31-6 Low Voltage Detection Diagram

To enable low voltage detection module, set LVWENB to 0. Users can configure the forewarning voltage by LVWSEL. The low voltage reset module is enabled by default and the reset voltage can be configured by LVRSEL.

31.2.2 CCFG2: RST_MOD (0x401D)

Table 31-1 CCFG2: RST_MOD (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	
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7：6	LVRSEL	Reset voltage selection. It detects the voltage of VDD5. 00: 2.8V 01: 3.0V 10: 3.5V 11: 3.8V
5：2	RSV	Reserved
1：0	LVWSEL	Warning voltage selection. When the voltage of VCC voltage lower than the set value, the hardware set LVWIF. MCU will enter interrupt if LVWIE is 1. 00: 7V. 01: 8V. 10: 9V. 11: 10V.

31.2.3 CCFG1:CK_RST_CFG (0x401E)

Table 31-2 CCFG1:CK_RST_CFG (0x401E)

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

Bit	Name	Function
7	LVWENB	Low voltage detection enable 0: Enable 1: Disable
6:0		

31.2.4 LVSR (0xDB)

Table 31-3 LVSR (0xDB)

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

Bit	Name	Function
[7:6]	RSV	Reserved
[5:3]	EXT0CFG	External interrupt 0 port configuration 000: Configure P0.0 as the external interrupt 0 input 001: Configure P0.1 as the external interrupt 0 input 010: Configure P0.2 as the external interrupt 0 input 011: Configure P0.3 as the external interrupt 0 input 100: Configure P0.4 as the external interrupt 0 input 101: Configure P0.5 as the external interrupt 0 input 110: Configure P0.6 as the external interrupt 0 input 111: Configure Output CMP4 output as external interrupt 0 input
[2]	TSDF	Over-temperature flag 0: The current temperature is not beyond the pre-set temperature. 1: The current temperature exceeds the pre-set temperature. This bit is often used with the temperature protection interrupt flag TCON[TSDIF], which is the dynamic over-temperature flag.
[1]	LVWF	VCC low voltage flag This bit indicates VCC lower than threshold. 0: VCC is higher than the threshold set by LVWSEL. 1: VCC is lower than the threshold set by LVWSEL.
[0]	LVWIF	VCC low voltage interrupt flag This bit is set by hardware when VCC lower than threshold set by

		LVWSEL and cleared by software. This bit is 0 when LVWIE is 0. 0: No event 1: A LVW event

32 FLASH

32.1 FLA_CR (0x85)

Table 32-1 FLA_CR (0x85)

Bit	7:5	4	3	2	1	0
Name	RSV	FLAERR	FLAACT	FLAPRE	FLAERS	FLAWEN
Type	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0

Bit	Name	Function
7:5	RSV	Reserved
4	FLAERR	Program error flag (read only) 0: Operation is successful when software programs or erases sector. 1: Operation error when software is programming or erasing sector
3	FLAACT	Flash operation (sector erase or byte writing) act 0: Invalid 1: Write 1 to start FLASH operation (Include program and sector erase)
2	FLA_SEC_PRE	Sector pre-programming enable (this sector must be pre-programmed before sector erase) 0: Disable 1: Enable Note: If FLAWEN=0, this bit is invalid
1	FLAERS	Sector erase enable 0: Disable 1: Enable Note: If FLAWEN=0, sector erase will be disabled
0	FLAWEN	Flash program enable 0: Disable 1: Enable Note: If FLAWEN=0, sector erase will be disabled

32.2 FLA_KEY (0x84)

Table 32-2 FLA_KEY (0x84)

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

Bit	Name	Function
7:0	FLA_KEY	Writing : FLASH sector erase/program key Sequential writing 0x5A, 0x1F to this register will start the Flash Program Interface function. If the writing sequence or writing data is illegal, the function will be frozen until system reset. After unlocking the key, any writing of FLA_CR will lock the FLA_KEY again.
		Reading: the lowest 2 bits are status of FLASH sector erase/program, highest 6 bits is reserved. 00: The key is locked. 01: 0x5A has been written, waiting for writing of 0x1F.

		11: The key is opened. 10: The key is frozen.

32.3 FLASH Sector Erase/Program Operation Instructions

Since the MOVX instruction in interrupt processing will error operate FLASH, for the security of the FLASH operation, all interrupt events should be disabled before self-programming.

Notes:

1. To protect firmware code, before FLASH operation, (including Sector erase and Byte program), all interrupt requests are strongly recommended to be disabled。
2. FLASH operation will requires time, decided by internal circuitry. One sector erasing will consume about 120~150ms.
3. Every sector has 256 bytes. The last byte (address: 0x7F00~0x7FFF) will not be erased at any time. When an instruction in unprotected sector, including the operation of reading, writing and erasing) accesses the protected sector, MCU will be reset.

33 CRC

Table 33-1 CRC Criterion and Generator Polynomial

Number	CRC Criterion	Generator Polynomial	Hexadecimal
1	CRC12	x^12 + x^11 + x^3 + x^2 + x + 1	80F
2	CRC16	x^16 + x^15 + x^2 + 1	8005
3	CRC16-CCITT	x^16 + x^12 + x^5 + 1	1021
4	CRC32	x^32 + x^26 + x^23 + x^22 + x^16 + x^12 + x^11 + x^10 + x^8 + x^9 + x^5 + x^4 + x + 1	04C11DB7

33.1 CRC16 Generator Polynomial

FU6813/63 CRC system is based CRC16-CCITT Criterion generator polynomial: $X^{16} + X^{12} + X^5 + 1$

33.2 Basic Logic Diagram of CRC16

A serial CRC16 circuit diagram as shown in Figure 33-1, FU6813/63 is based on parallel CRC16, which can be generated within one system clock cycle.

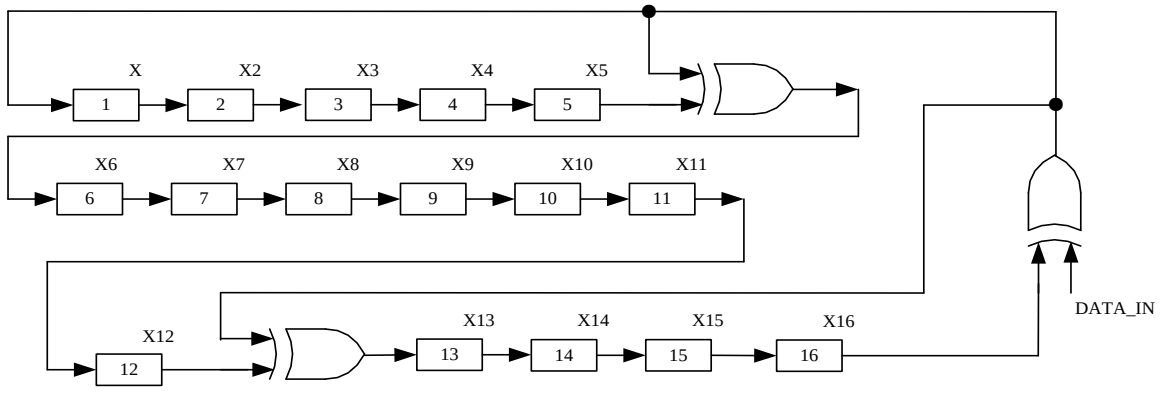


Figure 33-1 Logic Circuit Diagram of CRC16

33.3 Operation Steps

33.3.1 Computing single byte CRC

To compute single byte CRC, starting as following steps:

1. Based on the initial value, there two ways to initialize CRC_DR. If the initial value is 0x0000 or 0xFFFF, configure CRC_CR[CRCVAL] and set CRC_CR[CRCDINI] to 1. Otherwise, configure CRC_CR[CRCPNT] and CRC_DR to initialize CRC.
2. Software writes object data to CRC_DIN, such as 0x63. After 1 clock cycle, CRC result will be generated, and the result is 0xBD35.

3. Reading CRC result: Software writes CRC_CR[CRCPNT] = 1 and read the CRC result high byte from CRC_DR. Writes CRC_CR[CRCPNT] = 0 and read the CRC result low byte from CRC_DR. Combined result is the complete data.

33.3.2 Batch Computing CRC of ROM Data

To computing many sectors of FLASH ROM area CRC result, do steps as the followings:

1. Initialize CRC_DR. The method is the same as single byte mode.
2. Software writes value to CRC_BEG to set start address of FLASH ROM.
3. Software writes value to CRC_CNT to set sectors number of FLASH ROM to compute CRC result.
4. Software writes 1 to CRC_CR[AUTOINT] (keep other bits unchanged) to start calculation process.
5. Reading of CRC is the same as single byte mode.

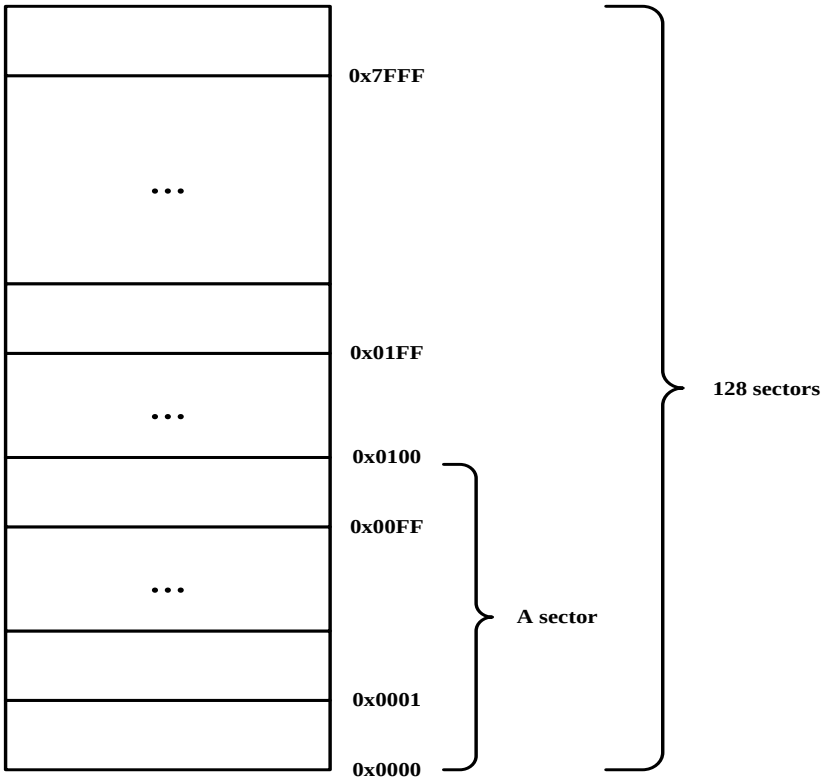


Figure 33-2 FLASH ROM Sector Frame

As shown in Figure 33-2, all FLASH ROM is 32K bytes, which includes 128 sectors, from sector0 to sector 127, each sector includes 256 bytes. Software can set register CRC_BEG to any value from 0x00 to 0x7F. But should consider CRC_BEG value when set CRC_CNT, for example, if set CRC_BEG to 0x7F, then should set CRC_CNT to 0x00, it means computing 1 sector FLASH ROM data CRC.

33.4 CRC Register

33.4.1 CRC_CR (0x4022)

Table 33-2 CRC_CR (0x4022)

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

Bit	Name	Function
[7:5]	RSV	Reserved
[4]	CRCDONE	Automatic CRC calculation completed flag Set to 0 when a CRC calculation is in progress. Code execution will stopped during a CRC calculation. Therefore, reads from firmware will always return 1.
[3]	CRCDINI	CRC result initialization bit Writing 1 to this bit will initialize CRC result based on CRC0VAL. Software reading of this bit is always 0.
[2]	CRCVAL	CRC set value initialization selection 0: CRC result is set to 0x0000 on writing of 1 to CRC0DINI. 1: CRC result is set to 0xFFFF on writing of 1 to CRC0DINI.
[1]	AUTOINT	Automatic CRC calculation enable When this bit is set to 1, any write to CRC_CNT will initiate an automatic CRC starting at FLASH sector CRC_BEG and continuing for CRC_CNT sectors. CRC calculation include 2 modes, one of single CRC calculation, another being batch calculation of FLASH ROM. Under single mode, software should set this bit to 0, and automatic batch calculation should set to 1 to start.
[0]	CRCPNT	CRC0 result pointer. Specify the byte of the CRC result to be read/write on the next access to CRC_DR. 0: CRC_DR accesses bits 7~0 of the 16 bit CRC result. 1: CRC_DR accesses bits 15~8 of the 16 bit CRC result.

33.4.2 CRC_DIN (0x4021)

Table 33-3 CRC_DIN (0x4021)

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

Bit	Name	Function

[7:0]	CRC_DIN	CRC data input. Each write to CRC_DIN results in the written data being computed into the CRC algorithm. Note: This register is a virtual register, and the written data will not be saved. Reading this register returns 0x00.

33.4.3 CRC_DR (0x4032)

Table 33-4 CRC_DR (0x4023)

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

Bit	Name	Function
[7:0]	CRC_DR	CRC Data output. Each read or write performed on CRC_DR targets the CRC result bits pointed to by CRC_CR[CRCPNT].

33.4.4 CRC_BEG (0x4024)

Table 33-5 CRC_BEG (0x4024)

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

Bit	Name	Function
[7]	RSV	Reserved
[6:0]	CRC_BEG	Automatic CRC Calculation starting sector These bits specify the FLASH sector to start the automatic CRC calculation. The starting address of the first FLASH sector included in the automatic CRC calculation is CRC_BEG × sector size (256).

33.4.5 CRC_CNT (0x4025)

Table 33-6 CRC0CNT (0x4025)

Bit	7	6	5	4	3	2	1	0
Name	RSV	CRC0CNT						
Type	R	R/W						
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	RSV	Reserved
[6:0]	CRC_CNT	Automatic CRC calculation sector count These bits specify the number of FLASH sectors to include in an automatic CRC calculation. The address of the last FLASH sector included in the automatic CRC calculation can be calculated by this register.

34 Power Management

34.1 PCON (0x87)

Table 34-1 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	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:6]	RSV	Reserved
5	GF3	General purpose flag 3
4	GF2	General purpose flag 2
3	GF1	General purpose flag 1
2	LDOM	LDO5 Deep Sleep Selection 0: LDO5 works normally in sleep mode. 1: LDO5 works in low power mode in sleep mode.
1	STOP	Software set 1 to cause device to enter into sleep mode. Cleared by hardware automatically when device wakes up.
0	IDLE	Software set 1 to make device enter idle mode. Cleared by hardware automatically when device wakes up. Power consumption mode: {STOP, IDLE} = 1x, system in deep sleep mode {STOP, IDLE} = 01, system in idle mode {STOP, IDLE} = 00, system in normal working mode

34.2 Power Mode

The device has 3 power consumption mode: normal mode, idle mode and deep sleep mode. Below table is the station in every mode:

Table 34-2 Power mode

Mode	Function	Wake-up Sources	Power Consumption performance
Normal	All functions are working, except some modules disabled by users.	NA	Power consumption is highest, but action fastest.
Idle	CPU clock is gated, other modules is disabled or working, decided by software. Watch-dog clock is gated.	Any interrupt request External reset Debug reset	Low power consumption. Flexible performance.
Deep Sleep	System deep-sleep Fast clock critical circuit will be closed, and if ADC or FOC or motor control circuit is	External(INT0/INT1) interrupt request External reset	Power consumption is lowest, action slow.

	running before fast clock critical circuit closed, fast clock will continue running until those running circuits have ended Watch-dog clock is closed	Debug reset	

35 Code Protection

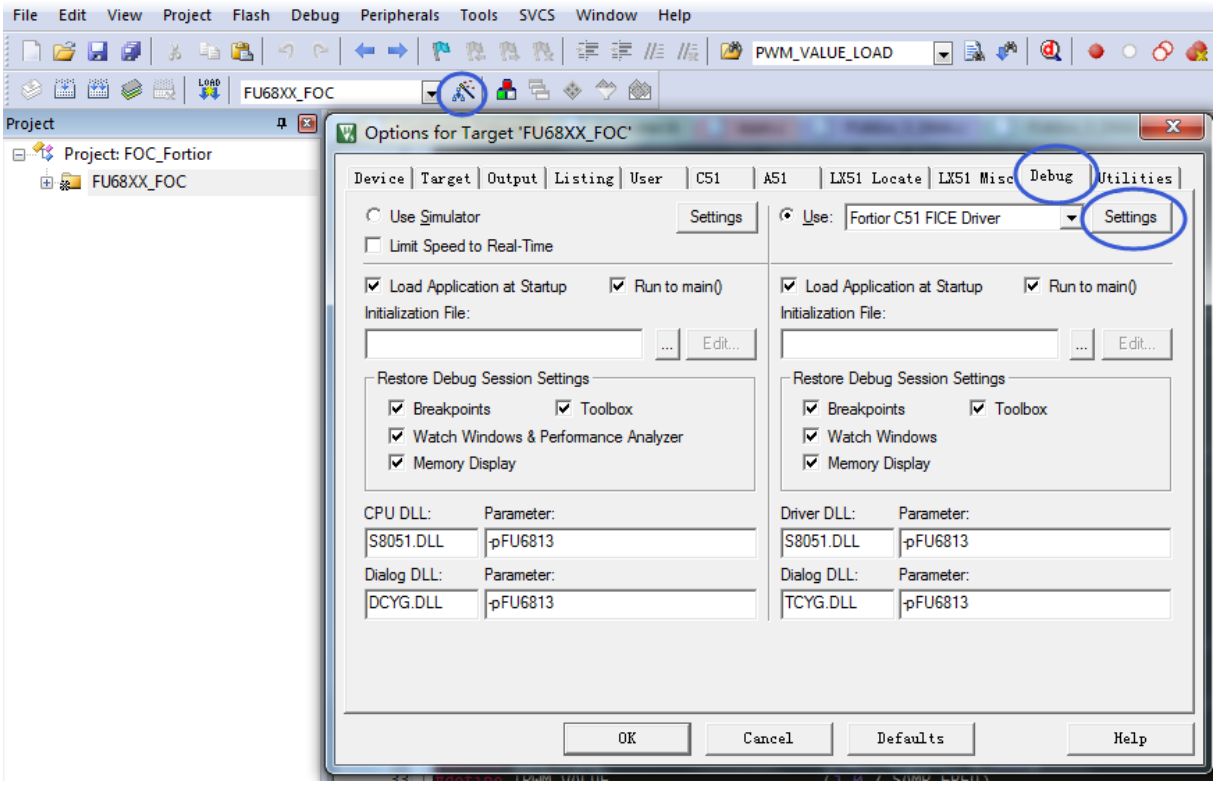


Figure 35-1 First Step of Code Protection

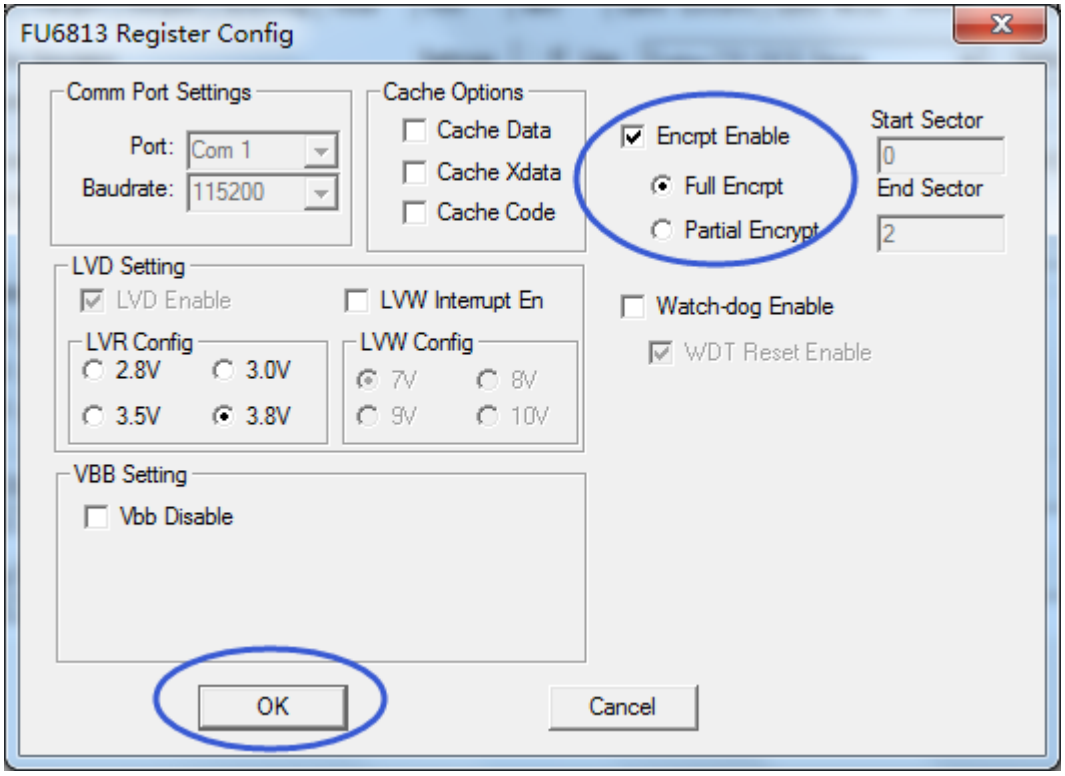


Figure 35-2 Full Protection Mode

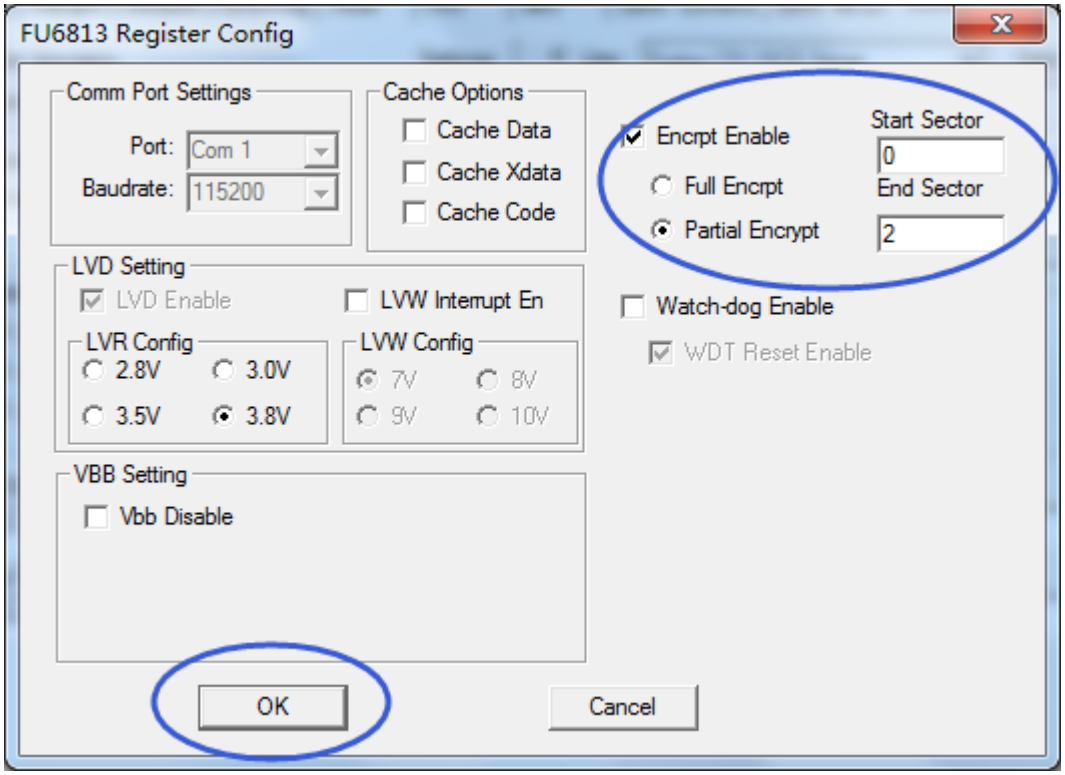


Figure 35-3 Partial Protection Mode

FU6813/63 supports user FLASH ROM code IPR(Intellectual Property Protection), method and step as below:

1. Open 8051 integrated design tool, enter “Target Options”, select “Debug” button, and click button settings as figure 32-1.
2. Select as figure 32-2, and click “OK” button. Then, compiling the project will generate .BIN file. Burning it to FLASH will finished the IPR step.

There are 2 code protection modes: full protection mode and partial protection mode. For full protection mode as shown in Figure 35-2, every sector of FLASH will be protected. For partial protection mode as shown in Figure 35-3, the sector from 0 to END SECTOR will be protected. The last sector will always be protected and each sector size is 256 bytes.

Copyright Notice

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

Fortior Technology(Shenzhen) Co.,Ltd.

Room203, 2/F, Building No.11,Keji Central Road2,
SoftwarePark, High-Tech Industrial Park, Shenzhen, P.R. China 518057
Tel: 0755-26867710
Fax: 0755-26867715
URL: http://www.fortiortech.com