Fortior Technology

FU6831/11/18
MCU Embedded and Configurable 3-Phase BLDC/PMSM Motor Controller

Datasheet

Table of Contents

1 System Overview

1.1 Features

- Power Supply：

  FU6831L/Q(1):

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

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

  FU6831N/FU6811N:

  Single Supply HV Mode： VCC=5~24V

  FU6811L:

  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

  FU6818Q:

  Mode1: VCC_MODE=0, VCC= 5~24V，VDRV=7~18V

  Mode2: VCC_MODE=1, VCC=VDD5=3~5.5V，VDRV=7~18V

- Dual Core: 8051 core and ME
- Pipelined instruction architecture; executes 70% of instructions in 1T or 2T
- 16K Byte Flash, with CRC verification, Support self-write and code protection
- 256 Byte IRAM，4K Byte XRAM
- ME：Integrated LPF, PI, SVPWM/SPWM, FOC module
- 1T 16x16 multiplier, 16T 32/32 divider
- 4 priority level interrupt, 16 interrupt source
- 32 GPIO(2)
- Timer

  4 capture timer

  1 advance timer

  1 timer for BLDC motor control

  1 RTC timer
- Support I2C/SPI/UART
- Analog peripheral

  8 channel 12 bit ADC(3). Support burst mode sample and conversion; Selectable voltage reference: Internal or External VREF, or VDD5

  Configurable VREF: 3V, 4V, 4.5V, 5V

Selectable 1/2 VDD5 or 1/2 VREF output

4 operational amplifiers

4 comparators with programmable hysteresis

- Predriver Mode:

  Gate Driver (For FU6811L/N only)

  3P3N Predriver (For FU6831L/Q/N only)

  6N Predriver (For FU6818Q)
- Motor drive modes: BLDC Square Wave (120°, 150°), SVPWM/SPWM、FOC
- Motor position sensing: HALL IC, HALL sensors, BEMF ZCP feedback
- FOC current sampling: Single or dual resistors (4)
- Clock Source

  System Clock Source: Internal clock 24MHz±2%, external 24MHz crystal oscillator or clock input. Dynamic swap supported.

  32768Hz crystal oscillator

  Watch-dog
- Two wire FICE for on chip debugging

(1)FU6831L/Q/N stands for the FU6831LQFP48、FU6831QFN48、FU6831QFN32 respectively，unless specified. FU6811L/Q/N is the same，represent the FU6811LQFP48、FU6811QFN32 respectively.

(2)For FU6831N, the number of GPIO is 18, for FU6811N，the number is 19.

(3)For FU6831N, the channel of 12bit ADC is 6, for FU6811N, the channel is 7.

(4)For FU6831N/FU6811N, as the package's limit, only support the single FOC current sampling.

1.2 Application

FU68xx can be applied for many domestic and industrial products using sensor/sensorless BLDC/PMSM, or induction motor. The typical applications include celling fan, standing fan, cooling fan, exhaust fan, electric tool, electric car, vacuum, pump, compressor and drone.

1.3 Description

FU68xx 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 and is supported with high speed ADC, high speed amplifiers and dividers, CRC, SPI, I2C, UART, timer, PWM, and HV LDO. These are essential for BLDC/PMSM. SVPWM/SPWM and FOC drive implementation.

FU68xx series is formed by 4 types of chip: FU6811, FU6831 and FU6818. These 4 chips are designed for different applications The detailed information can be found in Chapter-35. FU6811 uses Gate Driver output, FU6831 uses 3P3N Predriver output, FU6818 uses 6N Predriver output. By integrating dual core with rich set of peripherals, FU68xx 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 FU6831 Block Diagram

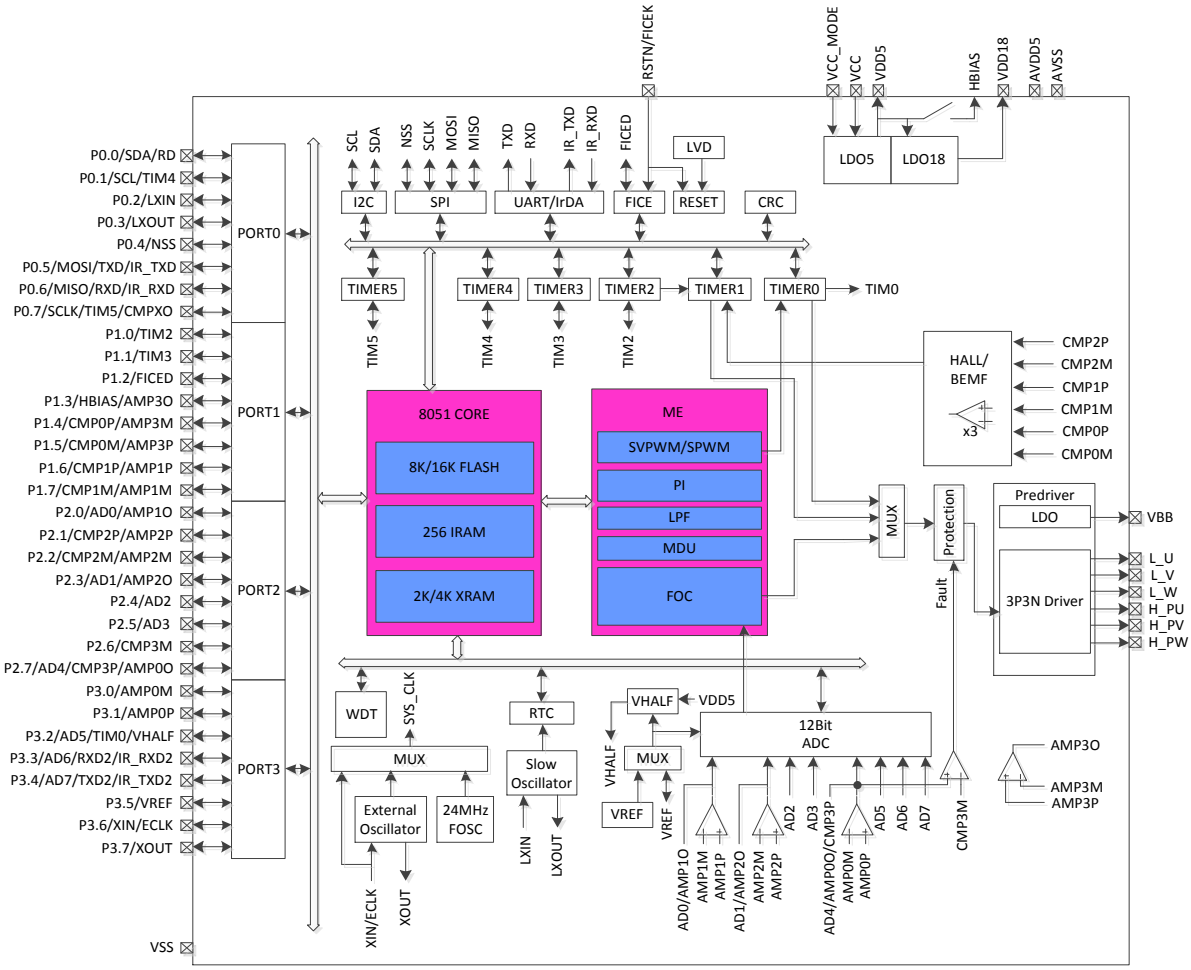


Figure 1-1 FU6831 block diagram

1.4.2 FU6811 Block Diagram

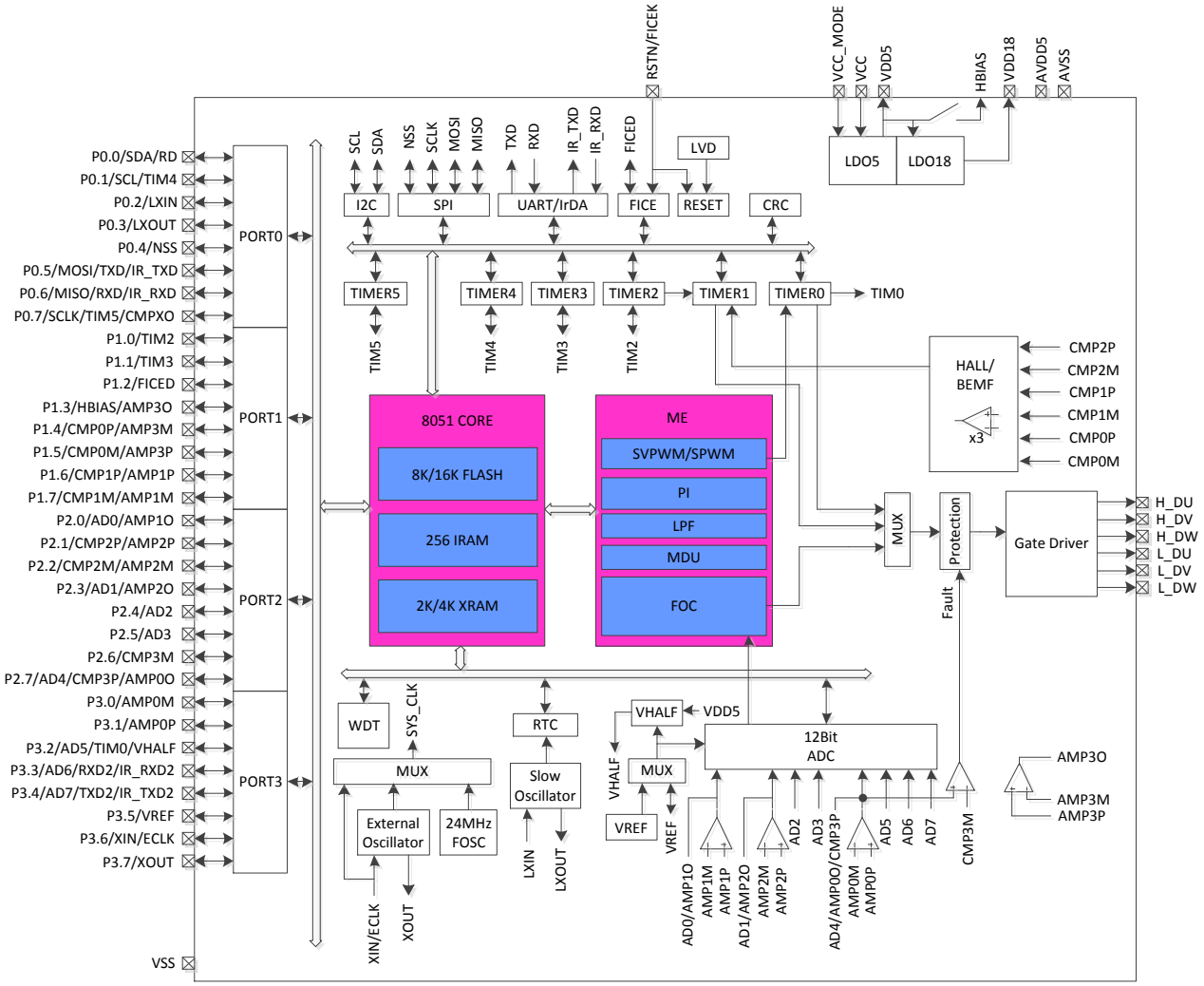


Figure 1-2 FU6811 block diagram

1.4.3 FU6818 Block Diagram

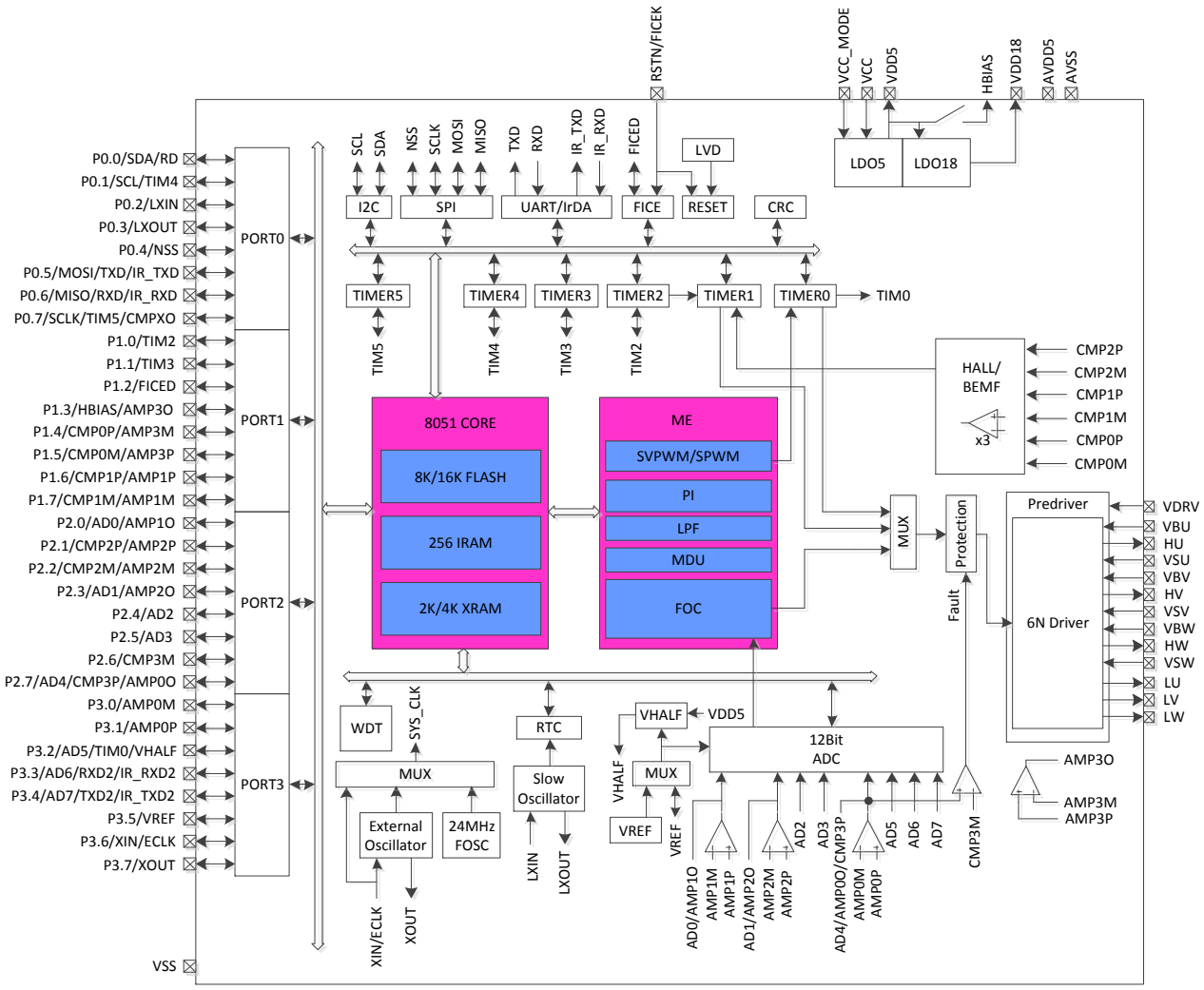


Figure 1-3 FU6818 block diagram

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.

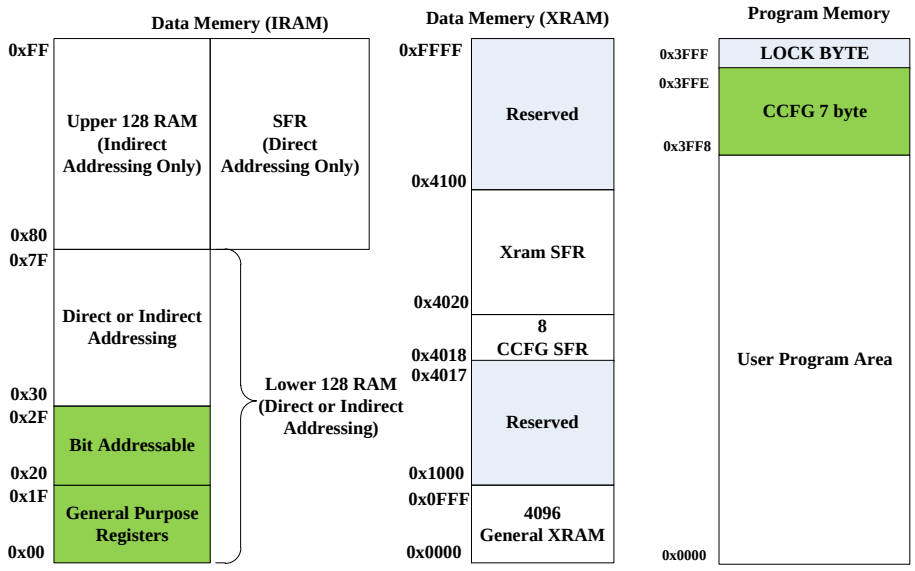


Figure 1-4 On-chip memory map

1.5.1 Program Memory

The FU68xx series implements 16 kB of the program memory space, re-programmable flash memory, organized in a contiguous block from addresses 0x0000 to 0x3FFF.

1.5.2 Data Memory

The FU68xx 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 1-4.

The MOVX instruction in an 8051 device is typically used to access external data memory. On the FU68xx 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 SFR Address Map

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0xF8	DRV_OUT	PL_CR			P0_OE	P1_OE	P2_OE	P3_OE
0xF0	B	OSC_CFG	PI_KIL	PI_KIH	PI_UKMAXL	PI_UKMAXH	PI_UKMINL	PI_UKMINH
0xE8	FOC_SET0	FOC_SET1	PI_EKL	PI_EKH	PI_UKL	PI_UKH	PI_KPL	PI_KPH
0xE0	ACC	SV_CR	SV_USL	SV_USH	SV_ANGL	SV_ANGH	LPF_YL	LPF_YH
0xD8	IP3	EVT_FILT	CMP_CR2	LVSR	LPF_KL	LPF_KH	LPF_XL	LPF_XH
0xD0	PSW	P1IE	P1IF	P2IE	P2IF	CMP_CR0	CMP_CR1	CMP_SR
0xC8	IP2	RSTSRC	MD_MBL	MD_MBH	MD_DB0	MD_DB1	MD_DB2	MD_DB3
0xC0	IP1	MD_MODE	MD_MAL	MD_MAH	MD_DA0	MD_DA1	MD_DA2	MD_DA3
0xB8	IP0	TIM0_IER	TIM0_CCR2L	TIM0_CCR2H	TIM0_CCR3L	TIM0_CCR3H	TIM0_CCR4L	TIM0_CCR4H
0xB0	P3	TIM0_SR	TIM2_CMTRL	TIM2_CMTRH	TIM2_ADTRL	TIM2_ADTRH	TIM0_CCR1L	TIM0_CCR1H
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	TIM5_CR1	TIM4_CNTRL	TIM4_CNTRH	TIM4_DRL	TIM4_DRH	TIM4_ARRL	TIM4_ARRH
0x88	TCON	TIM5_CR0	TIM5_CNTRL	TIM5_CNTRH	TIM5_DRL	TIM5_DRH	TIM5_ARRL	TIM5_ARRH
0x80	P0	SP	DPL	DPH	FLKEY	PSCTL	CFGKEY	PCON

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

1.5.4 XSFR

Table 1-2 XSFR Definition

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0x40f8	FOC_ESQUH	FOC_ESQUL	FOC_UDCFLTH	FOC_UDCFLTL	FOC_CSOH	FOC_CSOL		
0x40f0	FOC_UALPH	FOC_UALPL	FOC_UBETH	FOC_UBETL	FOC_EALPH	FOC_EALPL	FOC_EBETH	FOC_EBETL
0x40e8	FOC_IDH	FOC_IDL	FOC_IQH	FOC_IQL	FOC_VALPH	FOC_VALPL	FOC_VBETH	FOC_VBETL
0x40e0	FOC_IAH	FOC_IAL	FOC_IBH	FOC_IBL	FOC_IBETH	FOC_IBETL	FOC_COMRH	FOC_COMRL
0x40d8	FOC_EK1H	FOC_EK1L	FOC_EK2H	FOC_EK2L	FOC_EK3H	FOC_EK3L	FOC_EK4H	FOC_EK4L
0x40d0	FOC_EKPH	FOC_EKPL	FOC_EKIH	FOC_EKIL	FOC_POWKLPFH	FOC_POWKLPFL	FOC_POWH	FOC_POWL
0x40c8	FOC_EBMFKH	FOC_EBMFKL	FOC_OMEKLPFH	FOC_OMEKLPFL	FOC_FBASEH	FOC_FBASEL	FOC_EOMEH	FOC_EOMEL
0x40c0	FOC_THECORH	FOC_THECORL	FOC_ETHETAH	FOC_ETHETAL	FOC_KSLIDEH	FOC_KSLIDEL	FOC_EKLPFMINH	FOC_EKLPFMINL
0x40b8	FOC_THETAH	FOC_THETAL	FOC_THECOMPH	FOC_THECOMPL	FOC_RTHESTEPH	FOC_RTHESTEPL	FOC_RTHEACCH	FOC_RTHEACCL
0x40b0	FOC_ARRH	FOC_ARRL	FOC_SWDUTYH	FOC_SWDUTYL	FOC_TSMINH	FOC_TSMINL	FOC_TRGDLYH	FOC_TRGDLYL
0x40a8	FOC_UDH	FOC_UDL	FOC_UQH	FOC_UQL	FOC_IDREFH	FOC_IDREFL	FOC_IQREFH	FOC_IDREFL
0x40a0	FOC_QKPH	FOC_QKPL	FOC_QKIH	FOC_QKIL	FOC_QMAXH	FOC_QMAXL	FOC_QMINH	FOC_QMINL
0x4098	FOC_DKPH	FOC_DKPL	FOC_DKIH	FOC_DKIL	FOC_DMAXH	FOC_DMAXL	FOC_DMINH	FOC_DMINL
0x4090	FOC_CR1	FOC_CR2	FOC_CR3	FOC_IER	FOC_SR	FOC_CHC	FOC_PIRAN	FOC_CMR
0x4088	FOC_EFREQACCH	FOC_EFREQACCL	FOC_EFREQMINH	FOC_EFRQMINL	FOC_EFREQHOLDH	FOC_EFREQHOLDL	FOC_RTHECNT	FOC_FDS
0x4080								
0x4078	TIM1_BARRH	TIM1_BARRL	TIM1_BCNTRH	TIM1_BCNTRL	SIN_THETAH	SIN_THETAL	COS_THETAH	COS_THETAL
0x4070	TIM1_DBRH	TIM1_DBRL	TIM1_BCCRH	TIM1_BCCRL	TIM1_RARRH	TIM1_RARRL	TIM1_RCNTRH	TIM1_RCNTRL
0x4068	TIM1_CR0	TIM1_CR1	TIM1_CR2	TIM1_CR3	TIM1_ IER	TIM1_SR	TIM1_DRH	TIM1_DRL
0x4060	TIM0_ARRH/ SV_ARRH	TIM0_ARRL/ SV_ARRL	TIM0_PSCR	TIM0_RCR	TIM0_DTR/ TIM1_DTR/ FOC_DTR	RTC0TMH	RTC0TML	RTC0STA
0x4058	TIM0_CR	TIM0 _EGR	TIM0_CCMR1	TIM0_CCMR2	TIM0_CCER1	TIM0_CCER2	TIM0_CNTRH	TIM0_CNTRL

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0x4050	P1_AN	P2_AN	P3_AN	P0_PU	P1_PU	P2_PU	P3_PU	
0x4048	ADC_SCYC0	ADC_SCYC1			PH_SEL	DRV_CTL	AMP_CR	VREF_CR
0x4040	ADC4_DRH	ADC4_DRL	ADC5_DRH	ADC5_DRL	ADC6_DRH	ADC6_DRL	ADC7_DRH	ADC7_DRL
0x4038	ADC0_DRH	ADC0_DRL	ADC1_DRH	ADC1_DRL	ADC2_DRH	ADC2_DRL	ADC3_DRH	ADC3_DRL
0x4030	SPI_CFG	SPI_CTRL	SPI_SCR	SPI_DATH	SPI_DATL	ADC_CFG	ADC_MASK	ADC_STA
0x4028	I2C_MOD	I2C_ID	I2C_DAT	I2C_STA				
0x4020	TEST	CRC0DIN	CRC0STA	CRC0DAT	CRC0BEG	CRC0CNT	WDT_CSR	WDT_REL
0x4018	CCFG7	CCFG6	CCFG5	CCFG4	CCFG3	CCFG2	CCFG1	CCFG0

2 Pin Configuration and Functions

2.1 FU6831 Pin Definition

Table 2-1 FU6831 Pin Definition

Name	FU6831 QFN48/ LQFP48	IO Type	Descriptiion
P2.2/ CMP2M/ AMP2M	1	DB/ AI/ AI	1. GPIO P2.2 configurable as external interrupt 1 with optional 4.7K pull-up resistance 2. Negative input to comparator CMP2 for HALL2/BEMF2 3. Negative input to Op Amp OP-2
P2.3/ AD1/ AMP2O	2	DB/ AI/ AO	1. GPIO P2.2 configurable as external interrupt 1 2. ADC Channel-1 input, used for sampling of Op Amp OP-2 output. 3. Output of Op Amp OP-2
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/ CMP3M	5	DB/ AI	1. GPIO P2.6 configurable as external interrupt-1 2. Negative input to comparator CMP3 for over-current detection. To be connected to reference value.
P2.7/ AD4/ CMP3P/ AMP0O	6	DB/ AI/ AI/ AO	1. GPIO P2.7 configurable as external interrupt-1 2. ADC Channel-4 input 3. Positive input to comparator CMP3 for over-current detection. To be connected to sampled current value. 4. Output of Op Amp OP-0. Provides amplified value of sampled BUS current.
P3.0/ AMP0M	7	DB/ AI	1. GPIO P3.0 Negative input to Op Amp OP-0. Used for amplification of sampled BUS current.
P3.1/ AMP0P	8	DB/ AI	1. GPIO P3.1 Positive input to Op Amp OP-0. Used for amplification of sampled Bus current.
P3.2/ AD5/ TIM0/ VHALF	9	DB/ AI/ DO/ AO	1. GPIO P3.2 Advanced timer CH4 output 2. ADC Channel-5 input for thermal protection. 3. Advanced timer CH4 output 4. Voltage reference output. Configurable as 1/2 VDD5 or 1/2 VREF. External shunt of 1uF needed.

Name	FU6831 QFN48/	IO Type	Descriptiion
P3.3/ AD6/ RXD2/ IR_RXD2	10	DB/ AI/ DI/ DI	1. GPIO P3.3 2. ADC Channel-6 input 3. UART RxD when UART function transfer is enable 4. IrDA RxD when IrDA function transfer is enable
P3.4/ AD7/ TXD2/ IR_TXD2	11	DB AI/ DO/ DO	1. GPIO P3.4 2. ADC channel-7 input for speed control 3. UART TxD when UART function transfer is enable 4. IrDA TxD when IrDA function transfer is enable
P3.5/ VREF	12	DB/ AI	1. GPIO P3.5 2. Configurable as ADC external reference input or VREF voltage reference output. External shunt of 1uF needed.
AVSS	13	P	Analog ground
AVDD5	14	P	Analog power supply, to be connected externally with VDD5. External shunt of 1uF needed.
P3.6/ XIN/ ECLK	15	DB/ AI/ DI	1. GPIO P3.6 2. XIN of external 24MHz crystal oscillator 3. External clock input
P3.7/ XOUT	16	DB/ AO	1. GPIO P3.7 2. XOUT of external 24MHz crystal oscillator
P0.0/ SDA/ RD	17	DB/ DB/ DO	1. GPIO P0.0 configurable as external interrupt-0 2. I2C SDA with optional 4.7K pull-up resistance
P0.1/ SCL/ FG/TIM4	18	DB/ DB/ DB	1. GPIO P0.1 2. I2C SCL with optional 4.7K pull-up resistance 3. FG pin for motor control, configurable as 1 or 3 pulses per electrical cycle 4. Capture Timer 4 capture input/PWM output
P0.2/ LXIN	19	DB/ AI	1. GPIO P0.2 2. 32768Hz crystal-oscillator input
P0.3/ LXOUT	20	DB/ AI	1. GPIO P0.3 2. 32768Hz crystal-oscillator output
P0.4/ NSS	21	DB/ DB	1. GPIO P0.4 2. SPI NSS
P0.5/ MOSI/ TXD/ IR_TXD	22	DB/ DB/ DO/ DO	1. GPIO P0.5 2. SPI MOSI 3. UART TxD when UART function transfer is disable 4. IrDA TxD when UART function transfer is disable
P0.6/ MISO/ RXD/ IR_RXD	23	DB/ DB/ DI/ DI	1. GPIO P0.6 2. SPI MISO 3. UART RxD when UART function transfer is disable 4. IrDA RxD when UART function transfer is disable

Name	FU6831 QFN48/	IO Type	Descriptiion
P0.7/ SCLK/ TIM5/ CMPXO	24	DB/ DB/ DB/ DO	1. GPIO P0.7 2. SPI clock 3. Timer 5 capture input/PWM output 4. Output of comparator CMPX, for test usage
P1.0/ TIM2	25	DB/ DB	1. GPIO P1.0 configurable as external interrupt-1 2. Timer-2 capture input/PWM output
P1.1/ TIM3	26	DB/ DB	1. GPIO P1.1 configurable as external interrupt-1 2. Timer-3 capture input/PWM output
NC	27		Not connected
H_PU	28	AO	3P3N phase U high side output to PMOS gate. Internal 47k pull-up resistance.
H_PV	29	AO	3P3N phase V high side output to PMOS gate. Internal 47k pull-up resistance.
H_PW	30	AO	3P3N phase W high side output to PMOS gate. Internal 47k pull-up resistance.
L_U	31	DO	3P3N phase U low side output to NMOS gate. Internal 47k pull-down resistance.
L_V	32	DO	3P3N phase V low side output to NMOS gate. Internal 47k pull-down resistance.
L_W	33	DO	3P3N phase W low side output to NMOS gate. Internal 47k pull-down resistance.
VBB	34	P	10V LDO output. External shunt of 1~4.7uF required.
VCC	35	P	Power supply. Input voltage range selectable with VCC_MODE. External shunt of 10uF or more is required. 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, 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 selectable with VCC_MODE. External shunt of 1~4.7uF is required. 2. Please refer to VCC pin's description for description.
VCC_MODE	38	DI	Power supply mode selection. Please refer to VCC pin's description.
RSTN/ FICEK	39	DI/ DI	1. External schmitt reset input with internal pull-up. 2. FICE clock
VDD18	40	P	1.8V LDO output. External shunt of 1~4.7uF required.
P1.2/ FICED	41	DB/ DB	1. GPIO P1.2 configurable as external interrupt-1. 2. FICE data

Name	FU6831 QFN48/	IO Type	Descriptiion
P1.3/ HBIAS/ AMP3O	42	DB/ DO/ AO	1. GPIO P1.3 2. Gated VDD5 output 3. Output of Op Amp OP-3
P1.4/ CMP0P/ AMP3M	43	DB/ AI/ AO	1. GPIO P1.4 configurable as external interrupt-1 with optional 4.7K pull-up resistance 2. Positive input to comparator CMP0 for HALL0/BEMF0 3. Negative input to Op Amp OP-3
P1.5/ CMP0M/ AMP3P	44	DB/ AI/ AI	1. GPIO P1.5 configurable as external interrupt-1 with optional 4.7K pull-up resistance 2. Negative input to comparator CMP0 for HALL0/BEMF0 3. Positive input to Op Amp OP-3
P1.6/ CMP1P AMP1P	45	DB/ AI/ AI	1. GPIO P1.6 configurable as external interrupt-1 with optional 4.7K pull-up resistance. Can be used to detect the HALL1 logical output together with Timer-1. 2. Positive input to comparator CMP1 for HALL1/BEMF1 3. Positive input to Op Amp OP-1. Used for sampling of phase-U current for dual-resistance-sample FOC application.
P1.7/ CMP1M/ AMP1M	46	DB/ AI/ AI	1. GPIO P1.7 with optional 4.7K pull-up resistance. 2. Negative input to comparator CMP1 for HALL1/BEMF1 3. Negative input to Op Amp OP-1.
P2.0/ AD0/ AMP1O	47	DB/ AI/ AO	1. GPIO P2.0 configurable as external interrupt-1. 2. ADC channel-0 input. Used for capturing of sampled phase-U current signal from Op Amp OP-1 3. Output of Op Amp OP-1
P2.1/ CMP2P/ AMP2P	48	DB/ AI/ AI	1. GPIO P2.1 configurable as external interrupt-1 with optional 4.7K pull-up resistance. Can be used to detect the HALL2 logical output together with Timer-1 2. Positive input to comparator CMP1 for HALL1/BEMF1 3. Positive input to Op Amp OP-2. Used for sampling of phase-V current for dual-resistance-sample FOC application.

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 FU6831 LQFP48 Pinout

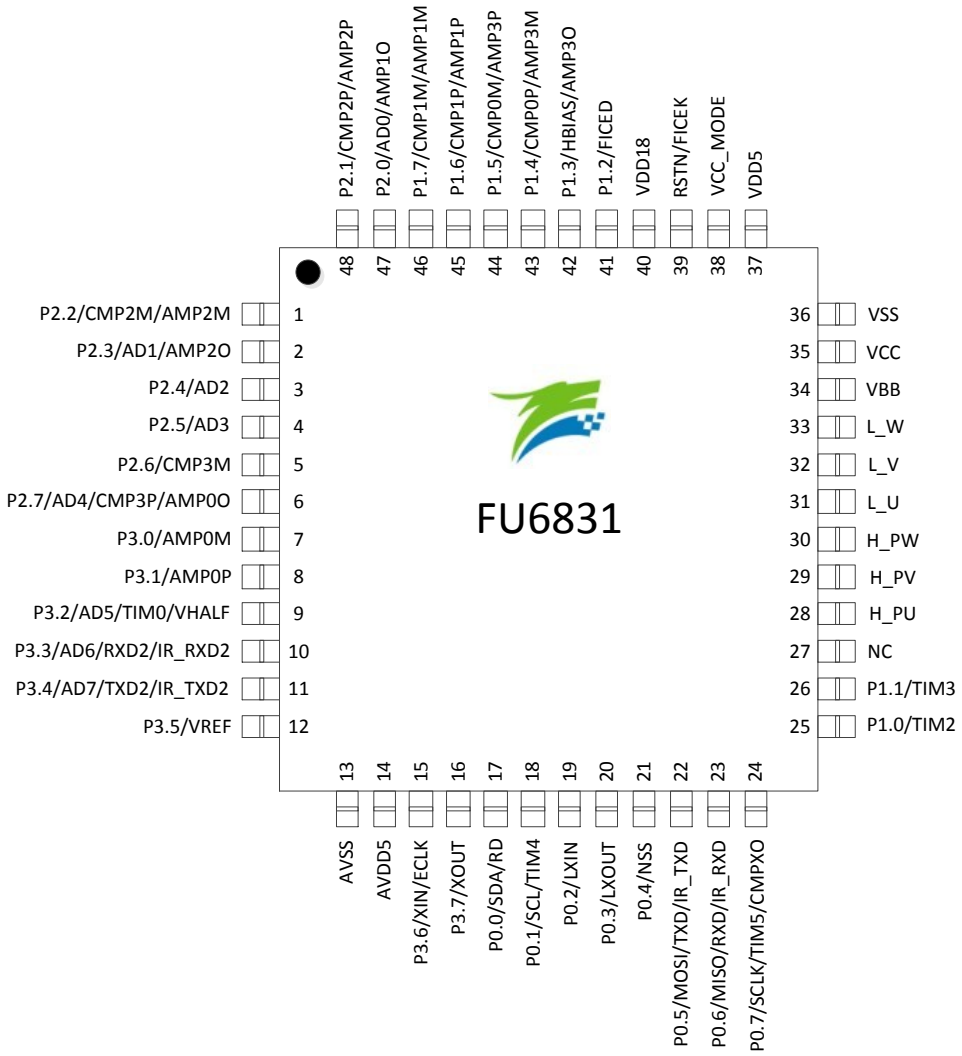


Figure 2-1 FU6831/ LQFP48 Pinout Diagram

2.3 FU6831 QFN48 Pinout

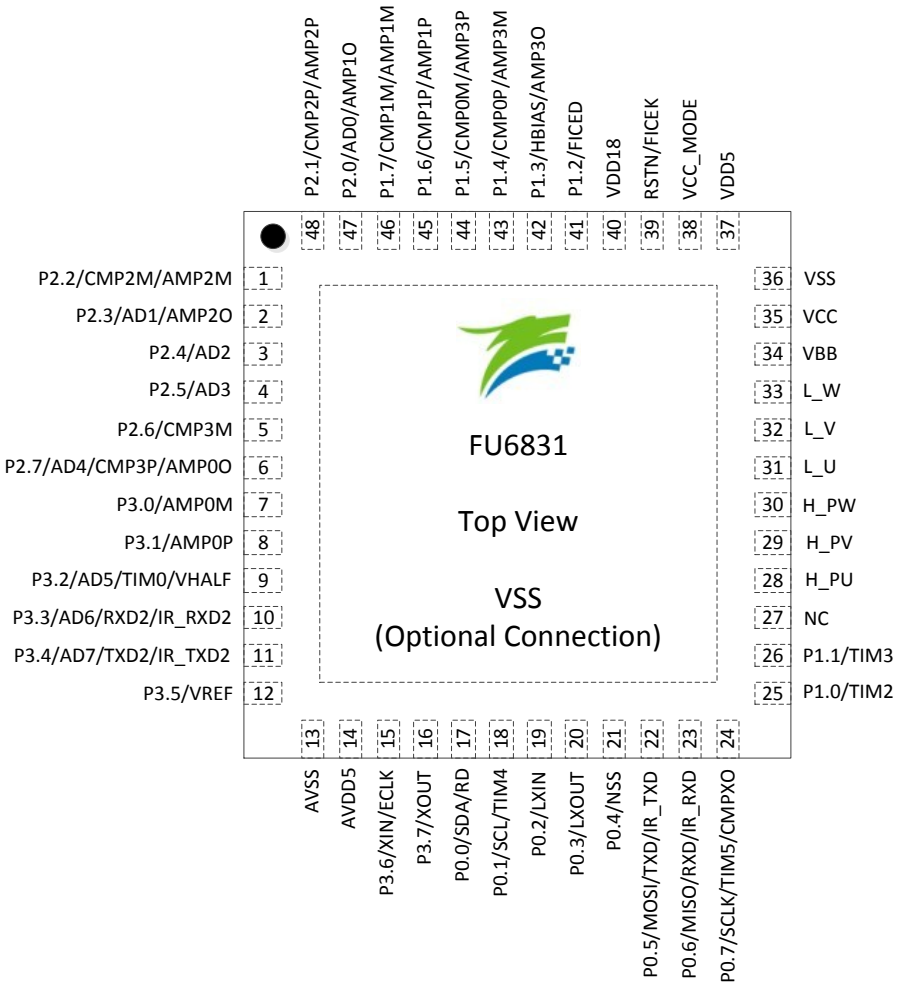


Figure 2-2 FU6831/QFN48 Pinout Diagram

2.4 FU6831 QFN32 Pin Definition

Table 2-2 FU6831 QFN32 Pin Definition

Name	FU6831 QFN32	IO Type	Descriptiion
P2.6/ CMP3M	1	DB/ AI	1. GPIO P2.6 configurable as external interrupt-1 2. Negative input to comparator CMP3 for over-current detection. To be connected to reference value.
P2.7/ AD4/ CMP3P/ AMP0O	2	DB/ AI/ AI/ AO	1. GPIO P2.7 configurable as external interrupt-1 2. ADC Channel-4 input 3. Positive input to comparator CMP3 for over-current detection. To be connected to sampled current value. 4. Output of Op Amp OP-0. Provides amplified value of sampled BUS current.
P3.0/ AMP0M	3	DB/ AI	1. GPIO P3.0 2. Negative input to Op Amp OP-0. Used for amplification of sampled BUS current.
P3.1/ AMP0P	4	DB/ AI	1. GPIO P3.1 2. Positive input to Op Amp OP-0. Used for amplification of sampled Bus current.
P3.2/ AD5/ TIM0/ VHALF	5	DB/ AI/ DO/ AO	1. GPIO P3.2 Advanced timer CH4 output 2. ADC Channel-5 input for thermal protection. 3. Advanced timer CH4 output 4. Voltage reference output. Configurable as 1/2 VDD5 or 1/2 VREF. External shunt of 1uF needed.
P3.4/ AD7/ TXD2/ IR_TXD2	6	DB AI/ DO/ DO	1. GPIO P3.4 2. ADC channel-7 input for speed control 3. UART TxD when UART function transfer is enable 4. IrDA TxD when IrDA function transfer is enable
P3.5/ VREF	7	DB/ AI	1. GPIO P3.5 2. Configurable as ADC external reference input or VREF voltage reference output. External shunt of 1uF needed.
AVSS	8	P	Analog ground
AVDD5	9	P	Analog power supply, to be connected externally with VDD5. External shunt of 1uF needed.
P0.0/ SDA/ RD	10	DB/ DB/ DO	1. GPIO P0.0 configurable as external interrupt-0 2. I2C SDA with optional 4.7K pull-up resistance

Name	FU6831 QFN32	IO Type	Descriptiion
P0.1/ SCL/ FG/TIM4	11	DB/ DB/ DB/	1. GPIO P0.1 2. I2C SCL with optional 4.7K pull-up resistance 3. FG pin for motor control, configurable as 1 or 3 pulses per electrical cycle 4. Capture Timer 4 capture input/PWM output
P1.4/ CMP0P/ P2.4/ AD2	12	DB/ AI/ DB/ AI	1. GPIO P1.4 configurable as external interrupt-1 with optional 4.7K pull-up resistance 2. Positive input to comparator CMP0 for HALL0/BEMF0 3. GPIO P2.4 configurable as external interrupt 1 4. ADC Channel-2 input, used for sampling of power supply voltage.
P0.5/ MOSI/ TXD/ IR_TXD	13	DB/ DB/ DO/ DO	1. GPIO P0.5 2. SPI MOSI 3. UART TxD when UART function transfer is disable 4. IrDA TxD when UART function transfer is disable
P0.6/ MISO/ RXD/ IR_RXD	14	DB/ DB/ DI/ DI	1. GPIO P0.6 2. SPI MISO 3. UART RxD when UART function transfer is disable 4. IrDA RxD when UART function transfer is disable
P0.7/ SCLK/ TIM5/ CMPXO	15	DB/ DB/ DB/ DO	1. GPIO P0.7 2. SPI clock 3. Timer 5 capture input/PWM output 4. Output of comparator CMPX, for test usage
P1.1/ TIM3	16	DB/ DB	1. GPIO P1.1 configurable as external interrupt-1 2. Timer-3 capture input/PWM output
H_PU	17	AO	3P3N phase U high side output to PMOS gate. Internal 47k pull-up resistance.
H_PV	18	AO	3P3N phase V high side output to PMOS gate. Internal 47k pull-up resistance.
H_PW	19	AO	3P3N phase W high side output to PMOS gate. Internal 47k pull-up resistance.
L_U	20	DO	3P3N phase U low side output to NMOS gate. Internal 47k pull-down resistance.
L_V	21	DO	3P3N phase V low side output to NMOS gate. Internal 47k pull-down resistance.
L_W	22	DO	3P3N phase W low side output to NMOS gate. Internal 47k pull-down resistance.
VBB	23	P	10V LDO output. External shunt of 1~4.7uF required.
VCC	24	P	Power supply. VCC input range is 5~24V, VDD5 is driven by internal LDO. External shunt of 10uF or more is required.

Name	FU6831 QFN32	IO Type	Descriptiion
VSS	25	P	Digital ground
VDD5	26	P	VDD5 is driven by internal LDO output. External shunt of 1~4.7uF is required.
RSTN/ FICEK	27	DI/ DI	External schmitt reset input with internal pull-up. FICE clock
VDD18	28	P	1.8V LDO output. External shunt of 1~4.7uF required.
P1.2/ FICED	29	DB/ DB	1. GPIO P1.2 configurable as external interrupt-1. 2. FICE data
P1.3/ HBIAS/	30	DB/ DO/	1. GPIO P1.3 2. Gated VDD5 output
P1.6/ CMP1P P2.0/ AD0	31	DB/ AI/ DB/ AI	1. GPIO P1.6 configurable as external interrupt-1 with optional 4.7K pull-up resistance. Can be used to detect the HALL1 logical output together with Timer-1. 2. Positive input to comparator CMP1 for HALL1/BEMF1 3. GPIO P2.0 configurable as external interrupt-1. 4. ADC channel-0 input. Used for capturing of sampled phase-U current signal
P2.1/ CMP2P/ P2.3/ AD1	32	DB/ AI/ DB/ AI	1. GPIO P2.1 configurable as external interrupt-1 with optional 4.7K pull-up resistance. Can be used to detect the HALL2 logical output together with Timer-1 2. Positive input to comparator CMP1 for HALL1/BEMF1 3. GPIO P2.3 configurable as external interrupt-1 4. ADC channel-1 input. Used for capturing of sampled phase-U current signal

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

Because some PINS were merged and packaged together in FU6831N(QFN32), like the P21/P23，P16/P20，and the P14/P20. The merged IO can be configured individually，but the PINS which merged together can’t set one high and the other low at the same time if they were setted to output mode.

2.5 FU6831 QFN32 Pinout

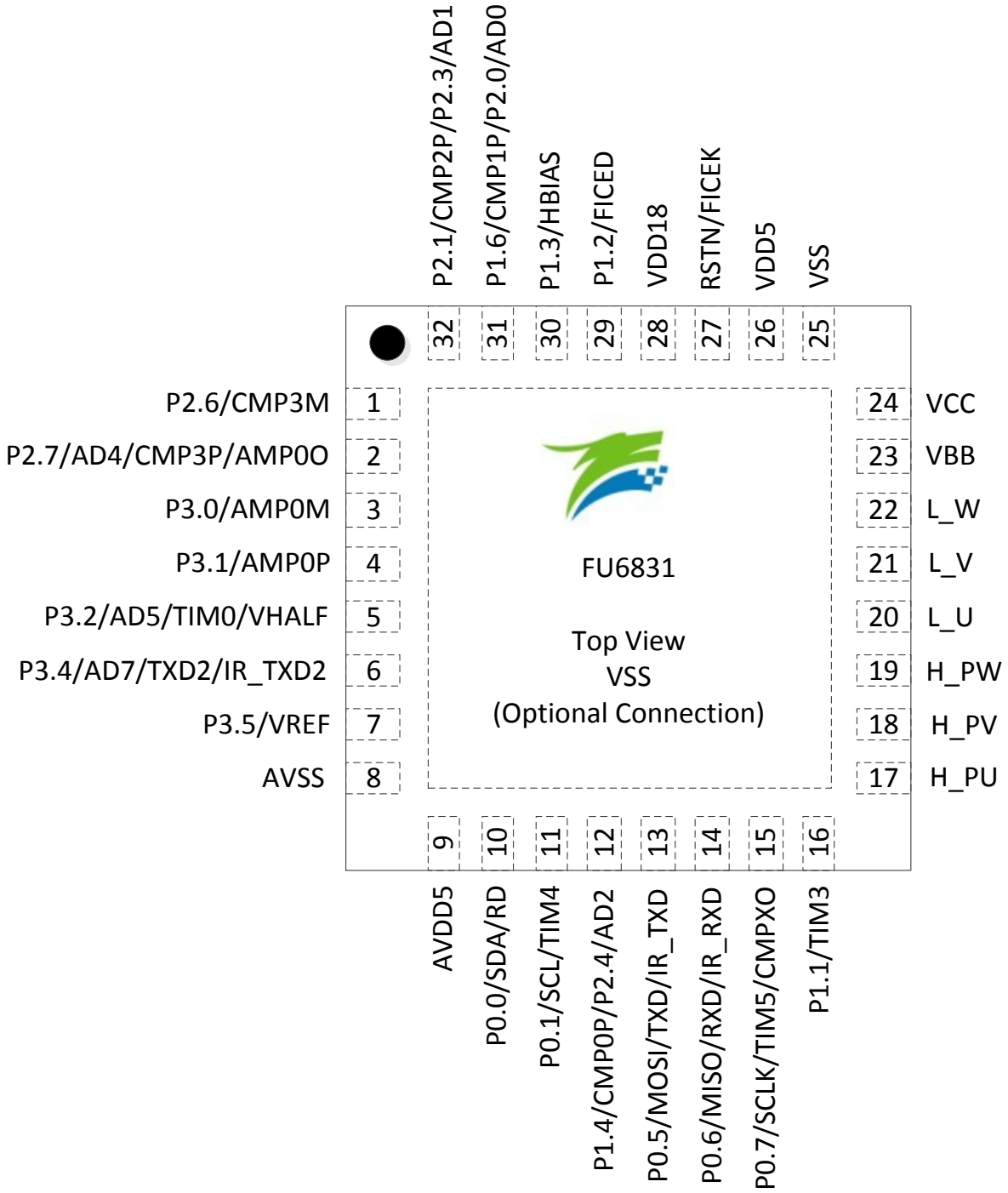


Figure 2-3 FU6831/QFN32 Pinout Diagram

2.6 FU6811 Pin Definition

Table 2-3 FU6811 Pin Definition

Name	FU6811 LQFP48	IO Type	Description
P2.2/ CMP2M/ AMP2M	1	DB/ AI/ AI	1. GPIO P2.2, configurable for external interrupt-1 as input, configurable 4.7K pull-up resistance 2. Using differential, HALL2/BEMF2, negtive input 3. OP-2 negtive input
P2.3/ AD1/ AMP2O	2	DB/ AI/ AO	1. GPIO P2.3, configurable for external interrupt-1 as input 2. ADC Channel-1 input, can be used sampling the output of OP-2. This can be used for sample one of phase current in FOC application. 3. OP-2 output
P2.4/ AD2	3	DB/ AI	1. GPIO P2.4, configurable for external interrupt 1 input 2. ADC Channel 2 input, can be used to sample the BUS voltage
P2.5/ AD3	4	DB/ AI	1. GPIO P2.5, configurable for external interrupt-1 as input 2. ADC Channel-3 input
P2.6/ CMP3M	5	DB/ AI	1. GPIO P2.6, configurable for external interrupt 1 input 2. Comparator-3 negtive input. For over current detection, connect the threshold voltage of over current judgement
P2.7/ AD4/ CMP3P/ AMP0O	6	DB/ AI/ AI/ AO	1. GPIO P2.7, configurable for external interrupt-1 as input 2. COMP-3 positive input. Input the over current signal for over-current protection 3. OP-0 output, for motor control. Sampling the BUS current signal with a sampling resistor, a=nd amplfy the sgnal by using OP-0.
P3.0/ AMP0M	7	DB/ AI	1. GPIO P3.0 2. OP-0 negtive input. Amplify the BUS current signal in motor control.
P3.1/ AMP0P	8	DB/ AI	1. GPIO P3.1 2. Amplifier 0 positive input. Amplifies the signal of BUS current in motor application.
P3.2/ AD5/ TIM0/ VHALF	9	DB/ AI/ DO/ AO	1. GPIO P3.2 2. Input over-temperature signal for protection, input from ADC Channel-5 3. Output of advanced Timer CH4 4. ADC channel-5 input 5. VHALF voltage reference, selectable for 1/2 VDD5, or 1/2 VREF. Shunt a 1uF capacitance to ground

Name	FU6811 LQFP48	IO Type	Description
P3.3/ AD6/ RXD2/ IR_RXD2	10	DB/ AI/ DI/ DI	1. GPIO P3.3 2. ADC Channel-6 input 3. UART RXD when UART function transfer is enable 4. IrDA RxD when IrDA function transfer is enable
P3.4/ AD7/ TXD2/ IR_TXD2	11	DB AI/ DO/ DO	1. GPIO P3.4 2. Analogue speed control input, ADC channel-7 input 3. UART TXD when UART function transfer is enable 4. IrDA TxD when IrDA function transfer is enable
P3.5/ VREF	12	DB/ AI	1. GPIO P3.5 2. VREF voltage reference input or internal VREF output. It is for ADC positive reference voltage. Shunt a 1uF capacitance to ground
AVSS	13	P	Analog ground
AVDD5	14	P	Analog power supply input, connect with VDD5 pin. Shunt a 1uF capacitance to ground
P3.6/ XIN/ ECLK	15	DB/ AI/ DI	1. GPIO P3.6 2. External 24MHz crystal oscillator input for system clock 3. External clock input
P3.7/ XOUT	16	DB/ AO	1. GPIO P3.7 2. External 24MHz crystal oscillator output
P0.0/ SDA/ RD	17	DB/ DB/ DO	1. GPIO P0.0, configurable for external interrupt-0 input 2. Output of locked rotor signal, open drain in output mode, configurable for 4.7K pull-up resistance 3. RD pin for motor control. Open drain in output mode.
P0.1/ SCL/ FG/TIM4	18	DB/ DB/ DB	1. GPIO P0.1 2. I2C SCL, open drain in output mode, configurable 4.7K pull-up resistance 3. FG pin for motor control, selectable for 1 or 3 pulse per electrical cycle. Open drain in output mode.
P0.2/ LXIN	19	DB/ AI	1. GPIO P0.2 2. 32768Hz crystal oscillator input, for RTC timer usage
P0.3/ LXOUT	20	DB/ AI	1. GPIO P0.3 2. 32768Hz crystal oscillator output
P0.4/ NSS	21	DB/ DB	1. GPIO P0.4 2. SPI NSS selection
P0.5/ MOSI/ TXD/ IR_TXD	22	DB/ DB/ DO/ DO	1. GPIO P0.5 2. SPI MOSI, output in master mode, and input in slave mode 3. UART TxD when UART function transfer is disable 4. IrDA TxD when IrDA function transfer is disable

Name	FU6811 LQFP48	IO Type	Description
P0.6/ MISO/ RXD/ IR_RXD	23	DB/ DB/ DI/ DI	1. GPIO P0.6 2. SPI MISO, input in master mode, and output in slave mode 3. UART RxD when UART function transfer is disable 4. IrDA RxD when IrDA function transfer is disable
P0.7/ SCLK/ TIM5/ CMPXO	24	DB/ DB/ DB/ DO	1. GPIO P0.7 2. SPI clock 3. Timer-5 input in capture mode, or output in PWM output mode 4. Comparator output for test usage
P1.0/ TIM2	25	DB/ DB	1. GPIO P1.0, configurable for external interrupt-1 as input 2. Timer-2 input in capture mode, and output in PWM output mode
P1.1/ TIM3	26	DB/ DB	1. GPIO P1.1, configurable for external interrupt-1 as input 2. Timer-3 input in capture mode and output in PWM output mode
NC	27		NC Pin, Keep it floating
NC	28		NC Pin, Keep it floating
L_DU	29	DO	Low side output of phase-U gate-driver
L_DV	30	DO	Low side output of phase-V gate-driver
L_DW	31	DO	Low side output of phase-W gate-driver
H_DU	32	DO	High side output of phase-U gate-driver
H_DV	33	DO	High side output of phase-V gate-driver
H_DW	34	DO	High side output of phase-W gate-driver
VCC	35	P	Power supply pin, voltage range depends on VCC_MODE pin, shunt a 10uF or bigger capacitance to ground. 1. Single supply HV mode: VCC_MODE=0, VCC input 5~24V, VDD5 is generated by internal LDO. 2. Single supply LV mode: VCC_MODE=1 (i.e., connected with VDD5), VDD5 input 3~5.5V and short with VCC 3. Dual supply mode: VCC_MODE=1 (i.e., connected with VDD5), VCC input 5~36V and VDD5 input 5V
VSS	36	P	Digital ground
VDD5	37	P	Power supply input, or internal LDO output, for medium voltage device. Please refer VCC pin's description for VDD5 connection. Shunt a 1~4.7uF capacitance to ground.
VCC_MODE	38	DI	Power supply mode selection. Please refer VCC pin's description.

Name	FU6811 LQFP48	IO Type	Description
RSTN/ FICEK	39	DI/ DI	1. External reset input, building pull-up internal resistor. Schmitt input. 2. FICE interface clock pin
VDD18	40	P	1.8V LDO output. Shunt a 1~4.7uF capacitance to ground.
P1.2/ FICED	41	DB/ DB	1. GPIO P1.2, configurable for external interrupt-1 as input 2. FICE interface data pin
P1.3/ HBIAS/ AMP3O	42	DB/ DO/ AO	1. GPIO P1.3 2. Bias power of Hall sensor. Putput VDD5 by a pass gate 3. OP-3 output
P1.4/ CMP0P/ AMP3M	43	DB/ AI/ AO	1. GPIO P1.4, configurable for external interrupt-1 as input. Configurable 4.7K pull-up resistance. Can be used to detect HALL0 logical output by timer-1. 2. Differential HALL0, or BEMF0, positive input 3. OP-3 negtive input
P1.5/ CMP0M/ AMP3P	44	DB/ AI/ AI	1. GPIO P1.5, configurable for external interrupt-1 as input. Configurable 4.7K pull-up resistance. 2. Differential HALL0, or BEMF0. negtive input 3. OP-3 positive input
P1.6/ CMP1P AMP1P	45	DB/ AI/ AI	1. GPIO P1.6, configurable for external interrupt-1 as input. Configurable 4.7K pull-up resistance. Can be used to detect HALL1 logical output by Timer-1. 2. Differential HALL1, or BEMF1, positive input 3. OP-1 positive input. Sampling phase-U current signal for dual-resistance FOC application.
P1.7/ CMP1M/ AMP1M	46	DB/ AI/ AI	1. GPIO P1.7, configurable 4.7K pull-up resistance. 2. Differential HALL1, or BEMF1, negtive input 3. OP-1 negtive input.
P2.0/ AD0/ AMP1O	47	DB/ AI/ AO	1. GPIO P2.0, configurable for external interrupt-1 input. 2. ADC channel-0 input, can be used to sample phase current U by scaled OP-1 3. OP-1 output
P2.1/ CMP2P/ AMP2P	48	DB/ AI/ AI	1. GPIO P2.1, configurable for external interrupt-1 input. Configurable 4.7K pull-up resistance. Can be used to detect HALL2 logical output by Timer-1. 2. Differential HALL1, or BEMF1, positive input 3. OP-2 positive input. Samplying phase-V current signal for dual-resistance FOC application.

Note:

IO Type:

DI: Digital input.

DO: Digital output

DB: Bidirectional input and output

AI: Analog input

AO: Analog output

P: Power supply or ground pin

2.7 Pinout of FU6811 with LQFP48 packaging

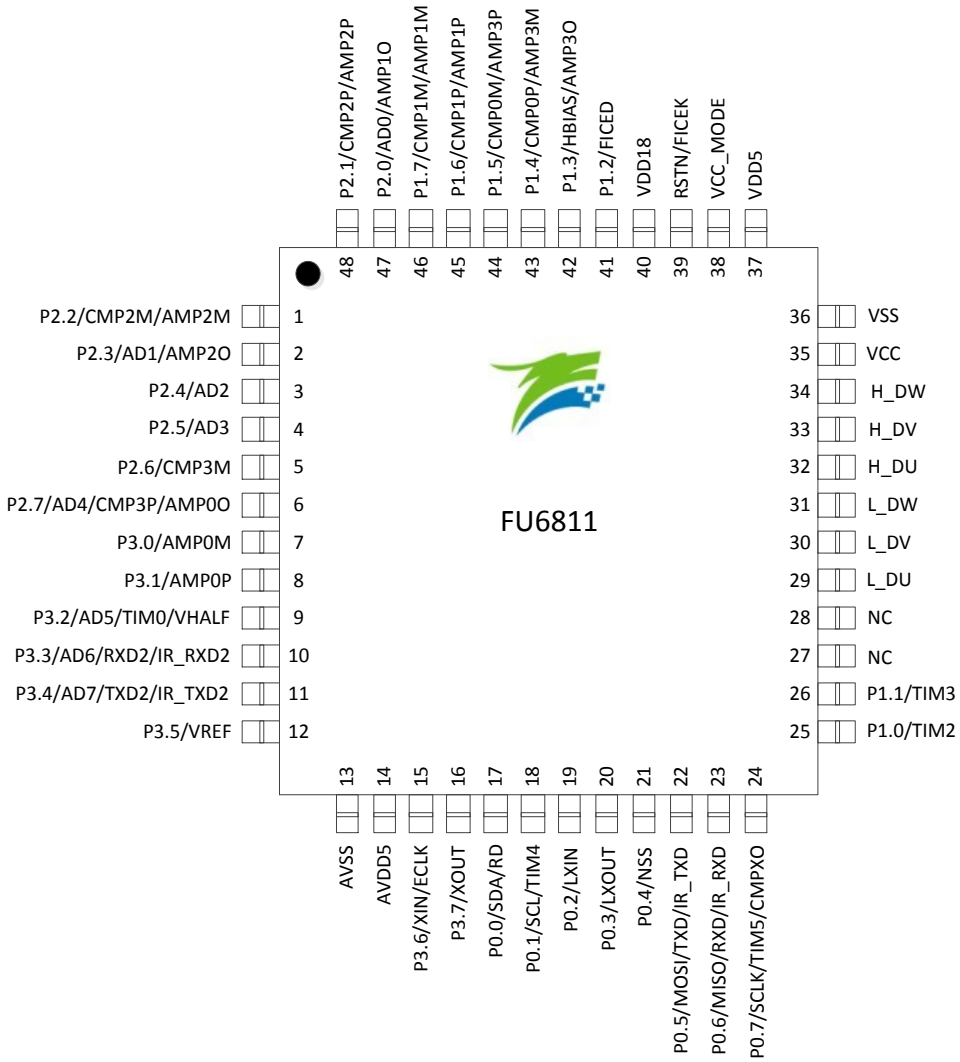


Figure 2-4 Pinout Diagram of FU6811 with LQFP48 packaging

2.8 FU6811 QFN32 Pin Definition

Table 2-4 FU6811 QFN32 Pin Definition

Name	FU6811 QFN32	IO Type	Descriptiion

Name	FU6811 QFN32	IO Type	Descriptiion
P2.7/ AD4/ CMP3P/ AMP0O	1	DB/ AI/ AI/ AO	1. GPIO P2.7 configurable as external interrupt-1 2. ADC Channel-4 input 3. Positive input to comparator CMP3 for over-current detection. To be connected to sampled current value. 4. Output of Op Amp OP-0. Provides amplified value of sampled BUS current.
P3.0/ AMP0M	2	DB/ AI	1. GPIO P3.0 2. Negative input to Op Amp OP-0. Used for amplification of sampled BUS current.
P3.1/ AMP0P	3	DB/ AI	1. GPIO P3.1 2. Positive input to Op Amp OP-0. Used for amplification of sampled Bus current.
P3.2/ AD5/ TIM0/ VHALF	4	DB/ AI/ DO/ AO	1. GPIO P3.2 Advanced timer CH4 output 2. ADC Channel-5 input for thermal protection. 3. Advanced timer CH4 output 4. Voltage reference output. Configurable as 1/2 VDD5 or 1/2 VREF. External shunt of 1uF needed.
P3.3/ AD6/ RXD2/ IR_RXD2	5	DB/ AI/ DI/ DI	1. GPIO P3.3 2. ADC Channel-6 input 3. UART RxD when UART function transfer is enable 4. IrDA RxD when IrDA function transfer is enable
P3.4/ AD7/ TXD2/ IR_TXD2	6	DB AI/ DO/ DO	1. GPIO P3.4 2. ADC channel-7 input for speed control 3. UART TxD when UART function transfer is enable 4. IrDA TxD when IrDA function transfer is enable
P3.5/ VREF	7	DB/ AI	1. GPIO P3.5 2. Configurable as ADC external reference input or VREF voltage reference output. External shunt of 1uF needed.
AVSS	8	P	Analog ground
AVDD5	9	P	Analog power supply, to be connected externally with VDD5. External shunt of 1uF needed.
P0.0/ SDA/ RD	10	DB/ DB/ DO	1. GPIO P0.0 configurable as external interrupt-0 2. I2C SDA with optional 4.7K pull-up resistance

Name	FU6811 QFN32	IO Type	Descriptiion
P0.1/ SCL/ FG/TIM4	11	DB/ DB/ DB/	1. GPIO P0.1 2. I2C SCL with optional 4.7K pull-up resistance 3. FG pin for motor control, configurable as 1 or 3 pulses per electrical cycle 4. Capture Timer 4 capture input/PWM output
P1.4/ CMP0P/ P2.4/ AD2	12	DB/ AI/ DB/ AI	1. GPIO P1.4 configurable as external interrupt-1 with optional 4.7K pull-up resistance 2. Positive input to comparator CMP0 for HALL0/BEMF0 3. GPIO P2.4 configurable as external interrupt 1 4. ADC Channel-2 input, used for sampling of power supply voltage.
P0.5/ MOSI/ TXD/ IR_TXD	13	DB/ DB/ DO/ DO	1. GPIO P0.5 2. SPI MOSI 3. UART TxD when UART function transfer is disable 4. IrDA TxD when UART function transfer is disable
P0.6/ MISO/ RXD/ IR_RXD	14	DB/ DB/ DI/ DI	1. GPIO P0.6 2. SPI MISO 3. UART RxD when UART function transfer is disable 4. IrDA RxD when UART function transfer is disable
P0.7/ SCLK/ TIM5/ CMPXO	15	DB/ DB/ DB/ DO	1. GPIO P0.7 2. SPI clock 3. Timer 5 capture input/PWM output 4. Output of comparator CMPX, for test usage
P1.1/ TIM3	16	DB/ DB	1. GPIO P1.1 configurable as external interrupt-1 2. Timer-3 capture input/PWM output
L_DU	17	DO	Low side output of phase-U gate-driver .
L_DV	18	DO	Low side output of phase-V gate-driver .
L_DW	19	DO	Low side output of phase-W gate-driver .
H_DU	20	AO	High side output of phase-U gate-driver
H_DV	21	AO	High side output of phase-V gate-driver
H_DW	22	AO	High side output of phase-W gate-driver
VCC	23	P	Power supply. VCC input range is 5~24V, VDD5 is driven by internal LDO. External shunt of 10uF or more is required.
VSS	24	P	Digital ground

Name	FU6811 QFN32	IO Type	Descriptiion
VDD5	25	P	VDD5 is driven by internal LDO output. External shunt of 1~4.7uF is required.
RSTN/ FICEK	26	DI/ DI	1. External schmitt reset input with internal pull-up. 2. FICE clock
VDD18	27	P	1.8V LDO output. External shunt of 1~4.7uF required.
P1.2/ FICED	28	DB/ DB	1. GPIO P1.2 configurable as external interrupt-1. 2. FICE data
P1.3/ HBIAS/	29	DB/ DO/	1. GPIO P1.3 2. Gated VDD5 output
P1.6/ CMP1P P2.0/ AD0	30	DB/ AI/ DB/ AI	1. GPIO P1.6 configurable as external interrupt-1 with optional 4.7K pull-up resistance. Can be used to detect the HALL1 logical output together with Timer-1. 2. Positive input to comparator CMP1 for HALL1/BEMF1 3. GPIO P2.0 configurable as external interrupt-1. 4. ADC channel-0 input. Used for capturing of sampled phase-U current signal
P2.1/ CMP2P/ P2.3/ AD1	31	DB/ AI/ DB/ AI	1. GPIO P2.1 configurable as external interrupt-1 with optional 4.7K pull-up resistance. Can be used to detect the HALL2 logical output together with Timer-1 2. Positive input to comparator CMP1 for HALL1/BEMF1 3. GPIO P2.3 configurable as external interrupt-1 4. ADC channel-1 input. Used for capturing of sampled phase-U current signal
P2.6/ CMP3M	32	DB/ AI	1. GPIO P2.6 configurable as external interrupt-1 2. Negative input to comparator CMP3 for over-current detection. To be connected to reference value.

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

Because some PINS were merged and packaged together in FU6811N(QFN32), like the P21/P23，P16/P20，and the P14/P20. The merged IO can be configured individually，but the PINS which merged together can’t set one high and the other low at the same time if they were setted to output mode.

2.9 FU6811 QFN32 Pinout

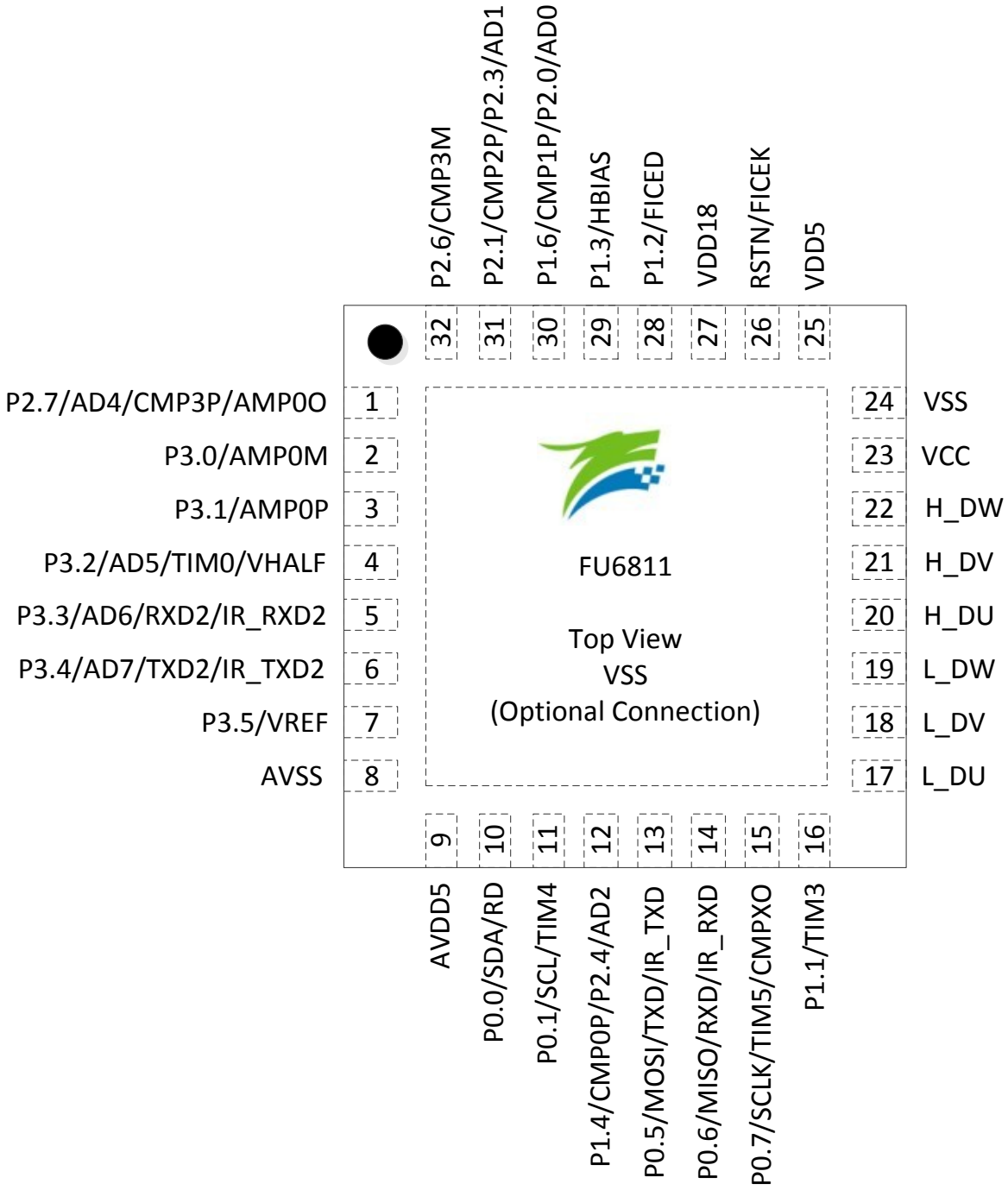


Figure 2-5 FU6811/QFN32 Pinout

2.10 FU6818 Pin Definition

Table 2-5 FU6818 Pin Definition

Name	FU6818 QFN56	IO Type	Description
VSU	1	P	Phase U input for 6N predirver, and for the gronding reference of the high side of phase-U self-boost
HU	2	DO	High side output of Phase-U predriver PWM, 6N control mode, drive NMOS gate.
VBU	3	P	Self-boost power supply for the high side of phase-U 6N predriver.
VSV	4	P	High side output of Phase-V predriver PWM, 6N control mode, drive NMOS gate.
HV	5	DO	High side output of Phase-V predriver PWM, 6N control mode, drive NMOS gate.
VBV	6	P	Self-boost power supply for the high side of phase-V 6N predriver.
VSW	7	P	High side output of Phase-W predriver PWM, 6N control mode, drive NMOS gate.
HW	8	DO	Self-boost power supply for the high side of phase-W 6N predriver.
VBW	9	P	Self-boost power supply for the high side of phase-W 6N predriver.
VCC	10	P	1. Supply pin, and voltage range depends on VCC_MODE pin, shunt a 10uF or more capacitance to ground. 2. Supply Mode 1: VCC_MODE=0, VCC input is 5~24V, VDD5 is generated by internal LDO. VDRV input 7~18V 3. Supply Mode 2: VCC_MODE=1 (i.e. connected with VDD5), VDD5 input is 3~5.5V, and short with VCC. VDRV input is 7~18V
VSS	11	P	Digital ground
VDD5	12	P	Supply input or internal LDO output for medium voltage power supply, Please refer VCC pin's description for VDD5 connection. Shunt a 1~4.7uF capacitance to ground.
VCC_MODE	13	DI	Power supply mode selection. Please refer VCC pin's description.
RSTN/ FICEK	14	DI/ DI	1. External reset input, building pull-up internal resistor. Schmitt input. 2. FICE interface clock pin
VDD18	15	P	1.8V LDO output. Shunt a 1~4.7uF capacitance to ground.
VSS	16	P	Digital ground
P1.2/ FICED	17	DB/ DB	1. GPIO P1.2, configurable for external interrupt-1 as input 2. FICE interface data pin

Name	FU6818 QFN56	IO Type	Description
P1.3/ HBIAS/ AMP3O	18	DB/ DO/ AO	1. GPIO P1.3 HALL 偏置电源，内部通过开关连接 VDD5 运放 3 输出端 2. Send out VDD5 with a pass gate 3. OP-3 output
P1.4/ CMP0P/ AMP3M	19	DB/ AI/ AO	1. GPIO P1.4, configurable for external interrupt-1 as input. Configurable 4.7K pull-up resistance. Can be used to detect the HALL0 logical output with Timer-1. 2. Differential HALL0, or BEMF0, positive input 3. OP-3 negtive input
P1.5/ CMP0M/ AMP3P	20	DB/ AI/ AI	1. GPIO P1.5, configurable for external interrupt-1 as input. Configurable 4.7K pull-up resistance. 2. Differential HALL0, or BEMF0, negtive input 3. OP-3 positive input
P1.6/ CMP1P AMP1P	21	DB/ AI/ AI	1. GPIO P1.6, configurable for external interrupt-1 as input. Configurable 4.7K pull-up resistance. Can be used to detect the HALL1 logical output by timer 1. 2. Differential HALL1, or BEMF1, positive input 3. OP-1 positive input. Connect the phase u current sample signal for dual-resistance-sample FOC application.
P1.7/ CMP1M/ AMP1M	22	DB/ AI/ AI	1. GPIO P1.7, configurable 4.7K pull-up resistance. 2. Differential HALL1, or BEMF1, negtive input 3. OP-1 negtive input.
P2.0/ AD0/ AMP1O	23	DB/ AI/ AO	1. GPIO P2.0, configurable for external interrupt-1 as input. 2. ADC channel-0 input, can be used sample phase current u with scaled OP-1 3. OP-1 output
P2.1/ CMP2P/ AMP2P	24	DB/ AI/ AI	1. GPIO P2.1, configurable for external interrupt-1 as input. Configurable 4.7K pull-up resistance. Can be used detecting the HALL2 logical output with Timer-1. 2. Differential HALL1, or BEMF1, positive input 3. OP-2 positive input. Sampling phase-V current signal for dual-resistance FOC application.
P2.2/ CMP2M/ AMP2M	25	DB/ AI/ AI	1. GPIO P2.2, configurable for external interrupt-1 as input, configurable 4.7K pull-up resistance 2. Differential HALL2/BEMF2 negtive input 3. OP-2 negtive input
P2.3/ AD1/ AMP2O	26	DB/ AI/ AO	1. GPIO P2.3, configurable for external interrupt-1 as input 2. ADC Channel-1 input, can be used to sample the output of OP-2. This can be used for sampling one of phase current in FOC application. 3. OP-2 output

Name	FU6818 QFN56	IO Type	Description
P2.4/ AD2	27	DB/ AI	1. GPIO P2.4, configurable for external interrupt-1 as input 2. ADC Channel-2 input, can be used to sample the BUS voltage
P2.5/ AD3	28	DB/ AI	1. GPIO P2.5, configurable for external interrupt-1 as input 2. ADC Channel-3 input
P2.6/ CMP3M	29	DB/ AI	1. GPIO P2.6, configurable for external interrupt-1 as input 2. COMP-3 negtive input. For over current detection, connect the threshold voltage of over current
P2.7/ AD4/ CMP3P/ AMP0O	30	DB/ AI/ AI/ AO	1. GPIO P2.7, configurable for external interrupt-1 as input 2. COMP-3 positive input. Input the voltage of current for over-current protection 3. OP-0 output, for motor application, convert the BUS current signal to voltage by using a resistor, then amplfy it by using OP-0.
P3.0/ AMP0M	31	DB/ AI	1. GPIO P3.0 2. OP-0 negtive input. Amplifies the signal of BUS current in motor application.
P3.1/ AMP0P	32	DB/ AI	1. GPIO P3.1 2. OP-0 positive input. Amplifies the signal of BUS current in motor application.
P3.2/ AD5/ TIM0/ VHALF	33	DB/ AI/ DO/ AO	1. GPIO P3.2 2. Over temperature signal input with ADC Channel-5 3. Advanced timer CH4 PWM output 4. VHALF voltage reference, selectable for 1/2 VDD5, or 1/2 VREF, shunt a 1uF capacitance to ground
P3.3/ AD6/ RXD2/ IR_RXD2	34	DB/ AI/ DI/ DI	1. GPIO P3.3 2. ADC Channel-6 input 3. UART RxD in UART function transfer mode 4. IrDA RxD in IrDA function transfer mode
P3.4/ AD7/ TXD2/ IR_TXD2	35	DB AI/ DO/ DO	1. GPIO P3.4 2. ADC channel-7 input 3. UART TxD in UART function transfer mode 4. IrDA TxD in IrDA function transfer mode
P3.5/ VREF	36	DB/ AI	1. GPIO P3.5 2. VREF voltage reference input, or internal VREF output. It is for ADC positive reference voltage. Shunt a 1uF capacitance to ground
AVSS	37	P	Analog ground
AVDD5	38	P	Analog supply input, connect with VDD5 pin. Shunt a 1uF capacitance to ground
P3.6/ XIN/ ECLK	39	DB/ AI/ DI	1. GPIO P3.6 2. External 24MHz crystal oscillator input for system clock 3. External clock input

Name	FU6818 QFN56	IO Type	Description
P3.7/ XOUT	40	DB/ AO	1. GPIO P3.7 2. External 24MHz crystal oscillator output
P0.0/ SDA/ RD	41	DB/ DB/ DO	1. GPIO P0.0, configurable for external interrupt-0 as input 2. I2C SDA, open drain in output mode, configurable 4.7K pull-up resistance
P0.1/ SCL/ FG/TIM4	42	DB/ DB/ DB	1. GPIO P0.1 2. I2C SCL, open drain in output mode, configurable 4.7K pull-up resistance 3. FG pin for motor application, selectable 1 or 3 pulse per electrical cycle of rotor. Open drain in output mode.
P0.2/ LXIN	43	DB/ AI	1. GPIO P0.2 2. 32768Hz crystal oscillator input, for RTC timer usage
P0.3/ LXOUT	44	DB/ AI	1. GPIO P0.3 2. 32768Hz crystal oscillator output
P0.4/ NSS	45	DB/ DB	1. GPIO P0.4 2. SPI NSS selection
P0.5/ MOSI/ TXD/ IR_TXD	46	DB/ DB/ DO/ DO	1. GPIO P0.5 2. SPI MOSI, output in master mode, and input in slave mode 3. UART TxD when UART function transfer is disable 4. IrDA TxD when IrDA function transfer is disable
P0.6/ MISO/ RXD/ IR_RXD	47	DB/ DB/ DI/ DI	1. GPIO P0.6 2. SPI MISO, input in master mode, and output in slave mode 3. UART RxD when UART function transfer is disable 4. IrDA RxD when IrDA function transfer is disable
P0.7/ SCLK/ TIM5/ CMPXO	48	DB/ DB/ DB/ DO	1. GPIO P0.7 2. SPI clock 3. Timer-5 input in capture mode, or output in PWM mode 4. Comparator output for test
P1.0/ TIM2	49	DB/ DB	1. GPIO P1.0, configurable for external interrupt-1 as input 2. Timer-2 input in capture mode, and output in PWM mode
P1.1/ TIM3	50	DB/ DB	1. GPIO P1.1, configurable for external interrupt-1 as input 2. Timer-3 input in capture mode, and output in PWM mode
VDRV	51	P	Power supply of 6N predriver, input 7~18V, shunt 1~10uF capacitance to ground
VSS	52	P	Digital ground
NC	53		NC Pin. Keep floating.
LU	54	DO	Low level output of the predriver with 6N control mode for phase-U, driving the NMOS gate.
LV	55	DO	Low level output of the predriver with 6N control mode for phase-V, driving the NMOS gate.
LW	56	DO	Low level output of the predriver with 6N control mode for phase-W, driving the NMOS gate.

Note:

IO: Type Claim.
DI: Digital input.
DO: Digital output
DB: Bidirectional input and output
AI: Analog input
AO: Analog output
P: Power supply or ground pin

2.11 Pinout of FU6818 with QFN56

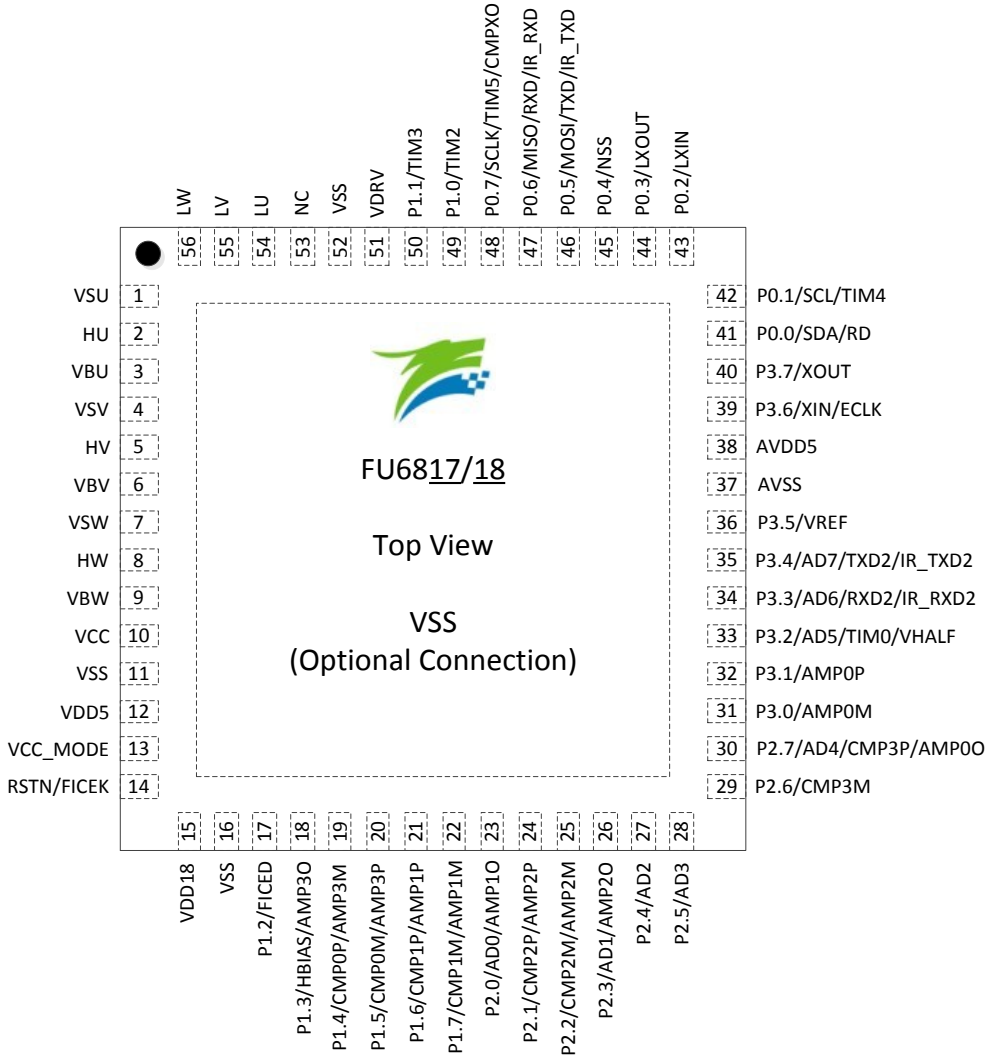


Figure 2-6 FU6818 QFN56 Pinout Diagram

3 Electrical Characteristics

3.1 Absolute Maximum Ratings

Table 3-1Absolute Maximum Ratings

Parameter	Conditions	Min	Typ	Max	Units
Ambient temperatureT_A		-40	—	85	℃
Ambient temperatureT_A	For FU6831,Dual supply mode,VCC_MODE=VDD5=5V,VCC≤24V	-40	—	105	℃
Ambient temperatureT_A	For FU6811,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
VDD relative to VSS		-0.3	—	6.5	V
VDRV relative to VSS	For FU6818	-0.3	—	22	V
VBU, VBV, VBW (floating voltage) relative to VSS	For FU6818	-0.3	—	222	V
VSU to VSV or VSW, and vice versa	For FU6818	-0.3	—	VBU-22, VBV-22, VBW-22	V
RSTN, VCC_MODE and GPIO relative to VSS		-0.3	—	VDD5+0.3	V

Notes:

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 above maximum rating conditions for extended periods may affect device reliability.

3.2 Global Electrical Characteristics

Table 3-2 Global Electrical Characteristics (For FU6831L/Q/N only)(4)

–40 to +85 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
VCC relative to VSS	Single Supply HV Mode VCC_MODE=0	5	—	24	V
	Dual Supply Mode VCC_MODE=1, VCC≥VDD5 (2), (3)	5	—	36	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}	VCC_MODE=0	—	100	250	uA

Notes:

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.
4. FU6831L/Q/N stands for LQFP48/QFN48/QFN32 respectively, unless specified

Table 3-3 Global Electrical Characteristics (For FU6811L/N only)(4)
–40 to +85 °Cunless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
VCC relative to VSS	Single Power Supply HV Mode VCC_MODE=0, VCC_MODE=0	5	—	24	V
	Dual Power Supply Mode, VCC_MODE=1, VCC≥VDD5, (2)	5	—	36	V
	Single Power Supply LV Mode, VCC_MODE=1, (2), (3)	3	—	5.5	V
VDD5 relative to VSS	VCC_MODE=1，Short VCC with VDD5	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}	VCC_MODE=0	—	100	250	uA
	VCC_MODE=1，VCC=VDD5=5V	—	45	100	uA

Notes:

1. Characteristics may vary for different program code
2. Maintain VDD5 withing 4.5~5.5V when erasing or programing of Flash
3. Unless specified, for VCC_MODE=1, VCC_MODE=VDD5
4. FU6811L/N stands for LQFP48/QFN32 respectively, unless specified

Table 3-4 Global Electrical Characteristics (For FU6818)
(–40 to +85 °C, unless specified)

Parameter	Conditions	Min	Typ	Max	Units
VCC relative to VSS	Single power supply HV mode, VCC_MODE=0	5	—	24	V
	Dual Power Supply Mode, VCC_MODE=1, VCC≥VDD5, (2)	5	—	36	V
VDD5 relative to VSS	VCC_MODE=1, with VCC and VDD5 connected, (2), (3)	3	—	5.5	V
VDRV relative to VSS		7	—	18	V
VBU, VBV, VBW (floating voltage) relative to VSS		—	—	200	V
VBU voltage relative to VSU, VBV voltage relative to VSV, VBW voltage relative to VSW		—	—	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}	VCC_MODE=0, VCC=12V, Short VCC and VDRV	—	350	650	uA
	VCC_MODE=1 , VCC=VDD5=5V, VCC=12V, Short VCC and VDRV	—	300	500	uA

Notes:

1. 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.

3.3 GPIO Electrical Characteristics

Table 3-5 GPIO Electrical Characteristics
(–40 to +85 °C, unless specified)

Parameter	Conditions	Min	Typ	Max	Units
Output rising time	50pF load, measure from to 10% to 90%, T_A=25℃	—	15	—	nS
Output falling time	50pF load, measure from to 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}		0.7*VDD5	—	—	V
Input Low Voltage V_{IL}		—	—	0.2*VDD5	V
Pull-up Resistance, excluding P0[1:0], P1[7:4], and P2[2:1]	Vin=0V, T_A=25℃	—	33	—	KΩ
Pull-up Resistance, for P0[1:0], P1[7:4], and P2[2:1]	Vin=0V, T_A=25℃	—	5	—	KΩ

3.4 Electrical Characteristics of Gate Driver IO (for FU6811L/N)

Table 3-6 Electrical Characteristics of Gate Driver IO
(–40 to +85 °C, 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 to 10% to 90%, T_A=25℃	—	7	—	nS
Output falling time	50pF Load, measure from to 90% to 10%, T_A=25℃	—	5	—	nS

3.5 Predriver 3P3N Electrical Characteristics (For FU6831L/Q/N only)

Table 3-7 Predriver 3P3N Electrical Characteristics
(–40 to +85 °C, unless specified)

Parameter	Conditions	Min	Typ	Max	Units
High Side Source Current	T_A=25℃, VCC=24V	—	150	—	mA
High Side Sink Current	T_A=25℃, VCC=24V	—	90	—	mA
Low Side Source Current	T_A=25℃, VCC=24V	—	150	—	mA
Low Side Sink Current	T_A=25℃, VCC=24V	—	180	—	mA
High side output rising time	1nF load, measure from to 10% to 90%, T_A=25℃	—	25	—	nS
High side output falling time	1nF load, measure from to 90% to 10%, T_A=25℃	—	90	—	nS
Low side output rising time	1nF load, measure from to 10% to 90%, T_A=25℃	—	115	—	nS
Low side output falling time	1nF load, measure from to 90% to 10%, T_A=25℃	—	60	—	nS

3.6 Predriver 6N Electrical Characteristics (For FU6818 only)

Table 3-8 Predriver 6N Electrical Characteristics (For FU6818 only)

Parameter	Conditions	Min	Typ	Max	Units
Output high peak current	VDRV=12V	—	1.2	—	A
Output low peak current	VDRV=12V	—	1.4	—	A
Output rising time	VDRV=12V 1nF Load, measure from to 10% to 90%, VDRV=12V, T_A=25℃	—	15	30	nS
Output falling time	VDRV=12V 1nF Load, measure from to 90% to 10%, VDRV=12V, T_A=25℃	—	15	30	nS

3.7 ADC Electrical Characteristics

Table 3-9 ADC Electrical Characteristics

（–40 to +85 °Cunless otherwise 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)	—	6	—	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)	—	500	—	Ω
Input capacitance C_{IN}	(1)	—	30	—	pF
Conversion Time	12 bit mode (1)	—	13	—	ADCLK
	10 bit mode (1)	—	11	—	ADCLK
Sample Time	(1)	3	—	63	ADCLK

Notes:

(1) ADCLK=12MHz

3.8 Electrical Characteristics of Reference Voltage

Table 3-10 VREF& VHALF

(–40 to +85 °C, unless specified)

Parameter	Conditions	Min	Typ	Max	Units
VREF	VREFVSEL=00B	—	3	—	V
	VREFVSEL=01B	—	4	—	V
	VREFVSEL=10B	—	4.5	—	V
	VREFVSEL=11B, VDD5=5.3V	—	5	—	V
VHALF	VHALFM=0	—	VDD5/2	—	V
	VHALFM=1	—	VREF/2	—	V

3.9 Electrical Characteristics of Operation Amplifier

Table 3-11 Electrical Characteristics of Operation Amplifier

(–40 to +85 °C, unless specified)

Parameter	Conditions	Min	Typ	Max	Units
Common Mode Input Range		0	—	VDD5-0.5	V
Input Offset Voltage	T_A = 25℃	—	5	—	mV
A_{OL}, Open Loop Gain	R_L=100KΩ	—	80	—	dB
UGBW, Unit Gain Bandwidth	C_L=40pF	6	10	—	MHz
SR, Slew Rate of OP	C_L=40pF	10	15	—	V/uS

3.10 Electrical Characteristics of Comparator

Table 3-12 Comparator Electrical Characteristics

(–40 to +85 °C, unless specified)

Parameter	Conditions	Min	Typ	Max	Units
V_{ICMR}, Common Mode Input Range		0	—	VDD5	V
V_{HYS}, Input Hysterisys Voltage of COMP	CMP_CR1[5:3]=000	—	0	—	mV
	CMP_CR1[5:3]=100	—	10	—	mV
	CMP_CR1[5:3]=111	—	15	—	mV
V_{OS}, Input Offset Voltage of COMP	T_A = 25℃	—	7	—	mV

3.11 HALL/BEMF Electrical Characteristics

Table 3-13 HALL/BEMF Electrical Characteristics

(–40 to +85 °C, unless specified)

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

3.12 Electrical Characteristics of Oscillator

Table 3-14 Oscillator Electrical Characteristics

(–40 to +85 °C, 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

3.13 Electrical Characteristics of Reset

Table 3-15 Electrical Characteristics of Reset

（T_A =–40 to +85 °C, VCC = 5V~24V,　VCC_MODE=0, unless specified）

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

3.14 Electrical Characteristics of LDO

Table 3-16 LDO Electrical Characteristics
(–40 to +85 °C, 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
VBB Voltage (For FU6831 only)	VCC = 12V~30V	9	10	11	V
VBB Start-up Time (For FU6831 only)	Time to start 0% to 95% of VBB final value. Shunt of 1uF connected.	—	—	1	mS

3.15 Thermal Information

Table 3-17 LQFP48 Thermal Resistance

Parameter	Conditions	Vaule	Units
Θ_{JA}: Junction-to-ambient thermal resistance	(1), (3)	52.4	℃/W
	(2), (3)	72.2	℃/W
Θ_{JC}: Junction-to-case(top) thermal resistance	(2), (3)	17	℃/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 3-18 QFN48 Thermal Resistance

Parameter	Conditions	Vaule	Units
Θ_{JA}: Junction-to-ambient thermal resistance	(1), (3)	36	℃/W
	(2), (3)	60	℃/W
Θ_{JC}: Junction-to-case(top) thermal resistance	(1), (3)	10.5	℃/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 3-19 QFN56 Thermal Resistance

Parameter	Conditions	Vaule	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 3-20 QFN32 Thermal Resistance

Parameter	Conditions	Vaule	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.

4 Reset Sources

Table 4-1 Reset sources control RSTSRC（0xC9）

Bit	7	6	5	4	3	2	1	0
Name	POR	EXTR	LVDR	Reseved	WDTR	FEDR	DBGR	RSTCLR
Type	R	R	R	R	R	R	R	W
Reset	X	X	X	X	X	X	X	0

Bit	Name	Function
[7]	POR	Power-On reset flag 0: Last reset not caused by power-on reset. 1: Last reset caused by power-on reset.
[6]	EXTR	External reset flag 0: Last reset not caused by external reset. 1: Last reset caused by external reset.
[5]	LVDR	Low voltage reset flag 0: Last reset not caused by low voltage reset. 1: Last reset caused by low voltage reset.
[4]	Reserved	Reserved
[3]	WDTR	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]	FEDR	Flash Error detect flag 0: Last reset not caused by flash error detect reset. 1: Last reset caused by flash error detect reset.
[1]	DBGR	Debug reset flag 0: Last reset not caused by debug reset. 1: Last reset caused by debug reset.
[0]	RSTCLR	Reset flag clear Write operation of '1' will clear all reset flags. Read operation is meanless.

4.1 Reset source (RSTSRC)

The chip has 7 reset sources:

(1) Power on reset;
(2) External reset;
(3) Low voltage reset;
(4) Watch-dog reset;
(5) Flash operation error reset
(6) Debug reset.

The last reset flag is saved in a readable register, RSTSRC. RSTSRC can be read to check to

the reset source which caused the last reset. But, bit 0 of RSTSRC, is meanless and can only be set to clear flag the register.

Note: The latest reset will clear the entire register and update it with the latest flag.

4.2 Power-On reset/External reset

Powering on or keeping the PIN RSTN low voltage longer 20us will respectively generate an internal or external device reset. These resets will trigger a boot process, and copies some of the device parameters from flash memory to registers.

4.3 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 PIN VDD5 voltage is lower than voltage gate. Voltage gate level is configurable in configuration registers.

4.4 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.

4.5 FEDR 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. FEDR reset source is always enabled.

5 Interrupt Handler

5.1 Interrupt Register

5.1.1 IE (0xA8)

Table 5-1 IE（0xA8），Interrupt Enable

Bit	7	6	5	4	3	2	1	0
Name	EA	RTCIE	MCDIE	ES0	SPIIE	EX1	TSDIE	EX0
Type	R/W	R/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]	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]	MCDIE	Enable Missing-Clock-Detect(MCD) interrupt 0: Disable MCD interrupt 1: Enable MCD interrupt
[4]	ES0	Enable UART interrupt 0: Disable UART interrupt 1: Enable UART interrupt
[3]	SPIIE	Enable Serial Peripheral Interface (SPI) interrupt 0: Disable SPI interrupt. 1: Enable SPI interrupt
[2]	EX1	Enable External Interrupt 1. 0: Disable external interrupt 1 from INT (P1/P2) inputs. 1: Enable external interrupt 1 from INT (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(P0.0) input. 1: Enable external interrupt 0 from INT0(P0.0) input.

5.1.2 IP0 (0xB8)

Table 5-2 IP0（0xB8）Interrupt Priority-0

Bit	7	6	5	4	3	2	1	0
Name	PFOC		PX1		PX0		PLVW	
Type	R/W	R/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]	PFOC	FOC interrupt priority.
[5:4]	PX1	INT1（External Interrupt 1）interrupt priority.
[3:2]	PX0	INT0（External Interrupt 0）interrupt priority.
[1:0]	PLVW	LVW（Low-Voltage-Warning）interrupt priority.

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

5.1.3 IP1 (0xC0)

Table 5-3 IP1（0xC0）Interrupt Priority-1

Bit	7	6	5	4	3	2	1	0
Name	PCMP		PADC		PTIM1		PTIM0	
Type	R/W	R/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 Comparor Interrupt Priority
[5:4]	PADC	ADC Interrupt Priority
[3:2]	PTIM1	Timer1 Interrupt Priority
[1:0]	PTIM0	Timer0 Interrupt Priority

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

5.1.4 IP2 (0xC8)

Table 5-4 IP2（0xC8）, Interrupt Priority-2

Bit	7	6	5	4	3	2	1	0
Name	PTSD		PTIM45		PTIM23		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]	PTSD	TSD Interrupt Priority
[5:4]	PTIM45	Timer 4/5 Interrupt Priority
[3:2]	PTIM23	Timer 2/3 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

5.1.5 IP3 (0xD8)

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

Bit	7	6	5	4	3	2	1	0
Name	PMCD		PSPI		PI2C		PUART	
Type	R/W	R/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]	PMCD	MCD Interrupt Priority
[5:4]	PSPI	SPI Interrupt Priority
[3:2]	PI2C	I2C Interrupt Priority
[1:0]	PUART	UART Interrupt Priority

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

5.1.6 TCON (0x88)

Table 5-6 TCON（0x88）

Bit	7	6	5	4	3	2	1	0
Name	RSV	MCDIF	TSDIF	IT1		IF0	IT0	
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

Bit	Name	Function
[7]	RSV	Reserved
[6]	MCDIF	MCD interrupt flag 0：No MCD interrupt request. 1：MCD interrupt request. A software write of 0, set the bit to 0.
[5]	TSDIF	TSD interrupt flag: 0：No TSD interrupt request. 1：TSD interrupt request. A software write of 0, set the bit to 0.
[4:3]	IT1[1:0]	INT1 external interrupt-1 senstivity level setting: 2’b00: Active rising edge. 2’b01: Active falling edge. 2’b1x: Voltage level changes.
[2]	IF0	INT0 (External-0 Interrupt 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 senstivity level setting: 2’b00: Active rising edge. 2’b01: Active falling edge. 2’b1x: Voltage level changes.

5.2 Interrupt Summary

Table 5-7 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	Always Highest
LVW	0	0x0003	LVSR[0]	Y	CCFG1[6]	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]
FOC interrupt	3	0x001B	FOC_CR1[1]	Y	FOC_CR1[0]	IP0[7:6]
Advanced timer (TIM0) interrupt	4	0x0023	TIM0_SR1	Y	TIM0_IER	IP1[1:0]
BLDC time (TIM1) Interrupt	5	0x002B	TIM1_BCSR	Y	TIM_BCSR	IP1[3:2]
ADC interrupt	6	0x0033	ADC_STA[0] ADC_CFG[0]	Y	ADC_STA[1] ADC_CFG[1]	IP1[5:4]
CMP interrupt	7	0x003B	CMP_RISR[7:0]	Y	CMP_CR0 CMP_CR1	IP1[7:6]
RTC interrupt	8	0x0043	RTC0STA[6]	Y	IE[6]	IP2[1:0]
Capture Timer (TIM2)/ Capture Timer (TIM3)	9	0x004B	TIM2_CR1[7:5] TIM3_CR1[7:5]	Y	TIM2_CR1[4:3] TIM3_CR1[4:3]	IP2[3:2]
Capture Timer(TIM4)/ Capture Timer(TIM5)	10	0x0053	TIM4_CR1[7:5] TIM5_CR1[7:5]	Y	TIM4_CR1[4:3] TIM5_CR1[4:3]	IP2[5:4]
TSD interrupt	11	0x005B	TCON[5]	Y	IE[1]	IP2[7:6]
UART interrupt	12	0x0063	UT_CR[1:0]	Y	IE[4]	IP3[1:0]
I2C interrupt	13	0x006B	I2C_STA[7:0]	Y	I2C_MOD[5]	IP3[3:2]
SPI interrupt	14	0x0073	SPI_STA[7:0]	Y	SPI_MOD[3],IE[3]	IP3[5:4]
MCD interrupt	15	0x007B	TCON[6]	Y	IE[5]	IP3[7:6]

The FU68XX 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 processsed 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.

5.3 External Interrupt (INT0/INT1)

External interrupt has two interrupt sources INT0 and INT1. Setting P0.0 as digital IO input, P0.0 becomes an external IO interrupt source 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 level sensitive setting is IT0. The main enable register bit of INT1 is EX1, 16 PINs interrupt enable bits are controlled by P1IE and P2IE. Interrupt flags are P1IF/P2IF, interrupt active level setting is IT1.

NOTE:

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

Table 5-8 External Interrupt 1 IO

SFR ADDR.	BIT	NAME	FUNCTION	R/W	RES. VAL.
0xD1	[7:0]	P1IE[7:0]	Port 1.7~1.0 is used as external interrupt sources enable 0 : Disable this port external interrupt request. 1: Enable the port external interrupt.	R/W	0x00
0XD2	[7:0]	P1IF[7:0]	Port 1.7~1.0 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]	Port 2.7~2.0 is used as external interrupt sources enable 0 : Disable this port external interrupt request. 1: Enable the port external interrupt.	R/W	0x00
0XD4	[7:0]	P2IF[7:0]	Port 2.7~2.0 is used as external interrupt sources, interrupt flag 0: No interrupt request from the port. 1: An interrupt request exist from the port. Software clears the interrupt flag. NOTE: software should not clear the other flag.	R/W	0x00

6 I2C

6.1 Operating Instructions

6.1.1 Master Mode

1. configureI2CMS = 1 (I2C_MOD[6]) as master mode；
2. configureI2CSPD(I2C_MOD[2:1]) to select the clock frequency SCL.
3. configureI2CADD (I2C_ID[7:1]) to set target device address.
4. configureDMOD(I2C_STA[6]) to set the direction of the write/read.
5. ConfigureI2CEN = 1 (I2C_MOD[7]) to enable the I2C.
6. ConfigureSTART = 1 (I2C_STA[4]) to send START and address.
7. After sending the address and receiving the ACK/NACK, 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 STR(I2C_STA[2]) = 0 to release SCL after writing I2C_DAT 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, STR(I2C_STA[2]) 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 STR(I2C_STA[2]) = 0 to release SCL after writing I2C_DAT register. At this time, the master begins to receive data. When finished the transmit, STR(I2C_STA[2]) is set to 1by hardware and SCL is pulled down by the master, waiting for setting NACK(I2C_STA[1]) to send ACK/NACK by software.Sending ACK/NACK after setting the STR(I2C_STA[2]) = 0 and releasing SCL. After receiving the next byte, the SCL is forced to pull down by the master.
10. Send STOP condition. If STOP (I2C_STA[3]) is set to 1 during the process of transmit/receive, the master will transmit the STOP after finishing transmitting/receiving the current byte.

6.1.2 Slave Mode

1. Configure 2CMS = 0 (I2C_MOD[6]) as slave mode；
2. Configure 2CADD (I2C_ID[7:1]) to set the slave address, or configure the GC=1 (I2C_ID[0]) to enable the boardcast mode.
3. Configure I2CEN=1 (I2C_MOD[7]) 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. START(I2C_STA[4]) and STR(I2C_STA[2]) is set to 1 by hardware, and then wait for the NACK(I2C_STA[1]) to send ACK/NACK by software. Meanwhile, determine if in slave send mode, then write I2C_DAT register;set STR(I2C_STA[2]) = 0 and release SCL, send data after sending ACK/NACK. After the slave sends the data and receices the ACK/NACK from the master, SCL is forced to be pulled down by slave. Meantime, STR(I2C_STA[2]) is set 1 by hardware.

5. If inRECEIVE mode, the slave set STR(I2C_STA[2]) =0. Then it is ready for receiving the data, release SCL, send data ,waiting the salve reception of the data. After that, STR(I2C_STA[2]) is set to 1 by hardware, and SCL is forced to pull down by the master. Then the system waits for NACK(I2C_STA[1]). After setting STR(I2C_STA[2]) 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 wait for the address data.

6.1.3 I2C Interrupt Resources

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

1. STR(I2C_STA[2]), this interrupt flag is aviliable in both master mode and slave mode. It is used for pull down SCL, and then wait for the next operation.
2. STOP(I2C_STA[3]) 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).

6.2 I2C Register

6.2.1 I2C_MOD (0x4028)

Table 6-1 I2C_MOD（0x4028）

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

Bit	Name	Function
7	I2CEN	I2C enable bit 0: I2C is disabled 1: I2C is enabled. The relate GPIO change to I2C mode with OPEN DRAIN ouput. Whether to turn on the pull-up of I2C depends on the setting of the IO port.
6	I2CMS	I2C mode select 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 bit 0: interrupt is disabled 1: interrupt is enabled with the request generated by I2C_STA.I2CIF

6.2.2 I2C_ID (0x4029)

Table 6-2 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 boardcast call, only valid in the slave mode. 0: the boardcast call is not supported. 1: the boardcast call is supported, which means that address 0x00 can response.

6.2.3 I2C_DAT (0x402A)

Table 6-3 I2C_DAT（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_DAT	I2C data register

6.2.4 I2C_STA (0x402B)

Table 6-4 I2C_STA（0x402B）

Bit	7	6	5	4	3	2	1	0
Name	I2CBSY	DMOD	RSV	START	STOP	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 handware for sending “START” command. This bit is cleared by handware for sending “STOP” command. Slave mode: This bit is set to 1 by handware after receiving “START” command. This bit is cleared by handware after receiving “STOP” command.
6	DMOD	I2C read/white 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> START bit is 1 <b> Change the DMOD by setting the START bit to 1 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)
5	RSV	Reserved

<table>
<tr><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>4</td><td>START</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 START to 1, the process of sending data will be continud till the all the data a=have been sent out, and then start to transmit the START and address data. It is cleared when 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 ingnored untill receiving the next START.
In slave mode, the current data condition is decided by START and STOP:

Table 6-5 I2C state flag
START STOP description
0 0 The current writed/read is the data bytes
0 1 The current receiving is STOP
1 0 The current receiving is START + data bytes
1 1 The current receiving is STOP，and then received one is START + data bytes
Note: START will be cleared automatically by hardware when I2CEN=0</td></tr>
<tr><td>3</td><td>STOP</td><td>Master mode:
The software write 1 validly when I2CBSY=1; then transmit STOP by hardware. It is cleared automatically by hardware at the end of the transmission. If START and STOP are writed at the same time with I2CBSY=1, I2C will transmit 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 comleted. STOP will be forbidden to write till the end of the transmission. It is cleared when 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:
STOP will be cleared by hardware automatically when I2CEN=0.</td></tr>
</table>

Bit	Name	Function
2	STR	I2C event completion flag Being set to1 by hardware and cleared by software. STR will be cleared by hardware automatically when I2CEN=0. Master mode: STR will be set to 1 at the end sending the START and address bytes, or data bytes. At the same time, SCL will be pull down and will be released till STR is cleared. If START and STOP are both logic 1 at the same time, STR will not be set to 1 by hardware at the end STOP transmission. Master 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.

Bit	Name	Function
0	I2CIF	I2C interrupt requesting flag: clearing this bit will allow the I2C to keep the data transmitting state. 0: have a I2C interrupt request 1: have no I2C interrupt request Master mode: I2CIF is logic 1 when STR=1, otherwise it is logic 0. Slave mode: I2CIF is logic 1 when STOP=1 or STR=1, otherwise it is logic 0.

7 SPI

SPI is able to access synchronized serial BUS with full duplex. It may work as host device or slave device with 3-line or 4-line modes, and it may support many host devices and slave devices on a BUS. To avoid data conflict on the BUS, when two or more host deivces transfer data with SPI at the same time, slave selection signal NSS can config as input in order to make SPI works as slave mode, or disable host mode in multi-host environment. It may also config as chip select output (On the Master Mode), or disabled in 3-line mode. In the Master Mode, User can select lots of slave devices through other GPIO pins.

7.1 Operation Declaration

7.1.1 Signal Notice

The four signals used by SPI (SCK, NSS, MOSI, MISO.) are described below.

7.1.1.1 Master Output, Slave Input (MOSI)

MOSI signal is the output of master device, and input of slave device. It may be used to transfer data from master device to slave device. It can be configured as output when SPI works in the master mode. It may also be configured as input when SPI works in the slave mode. The highest bit (MSB) is transferred first. The MOSI is driven by the MSB of the shift register when SPI is configured as master mode.

7.1.1.2 Master Input, Slave Output (MISO)

MISO signal is the input of master device and output of slave device. It may also be used to transfer data from slave device to host device. It can be configured as input when SPI works in the master mode, or configured as output when SPI works in the slave mode. The highest bit (MSB) is transferred first. MISO pin will be in high-impendance state when SPI is disabled, or has not been selected on the 4-line slave mode. The MISO is driven by the MSB of the shift register when SPI works as slave device in the 3-line mode.

7.1.1.3 Serial Clock (SCK)

The signal of SCK is output from host device and input to slave device. It may be used to synchronize serial data transmission on the pin of MOSI, and MISO, between host device and slave device. It occurs when SPI works as host device. However, it will be ignored when the slave device is chosen as 4-line mode (i.e., NSS=1).

7.1.1.4 Slave Select (NSS)

The function of the slave-select (NSS) signal is dependent on the setting of the NSSMD1 and NSSMD0 bits in the SPI0CN register. There are three possible modes that can be selected with

these bits:

1. NSSMD[1:0] = 00: 3-Wire Master or 3-Wire Slave Mode: SPI0 operates in 3-wire mode, and NSS is disabled. When operating as a slave device, SPI0 is always selected in 3-wire mode. Since no select signal is present, SPI0 must be the only slave on the BUS in 3-wire mode. This is intended for point-to-point communication between a master and a single slave.
2. NSSMD[1:0] = 01: 4-line Slave or Multi-Master Mode: SPI0 operates in 4-line mode, and NSS is enabled as an input. When it operats as a slave, NSS selects the SPI0 device. When operating as a master, a 1-to-0 transition of the NSS signal disables the master function of SPI0, and the multiple master devices can thus be used on the same SPI BUS.
3. NSSMD[1:0] = 1x: 4-line Master Mode: SPI0 operates in 4-line mode, and NSS is enabled as an output. The setting of NSSMD0 determines the logic level that NSS pin will output. This configuration should only be used when operating SPI0 as a master device.

Figure 7-1 to Figure7-3 show the typical connection diagrams under various operational modes. Note that the setting of NSSMD bits affects the pinout of the device. When 3-line master or 3-line slave mode is used, the NSS pin will not be mapped by the crossbar. In all other modes, the NSS signal will be mapped to a pin on the device.

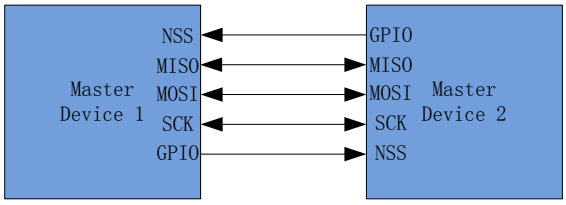


Figure 7-1 Multiple-Master Mode Connection Diagram

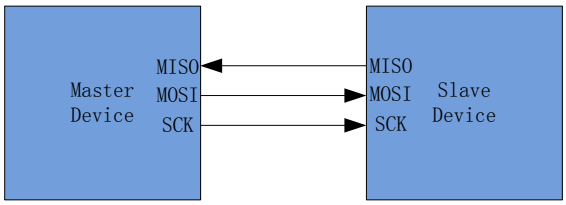


Figure 7-2 3-line Single Master and 3-line Single Slave Mode Connection Diagram

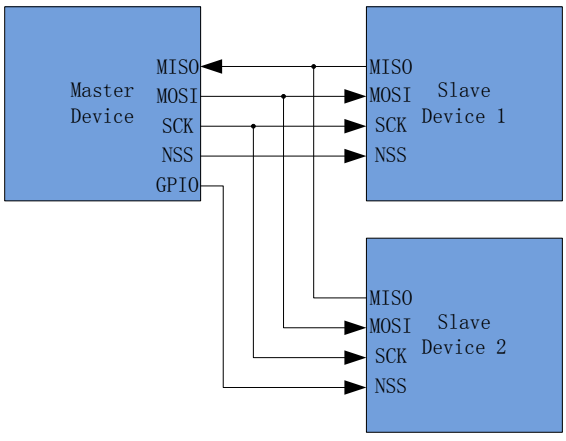


Figure 7-3 4-Wire Single Master Mode and 4-Wire Slave Mode Connection Diagram

7.1.2 SPI Master Mode Operation

Only SPI master can trigger the data transmission. SPI0 is placed in master mode by setting the Master Enable flag (MSTEN). Under master mode, data is written to SPI0 data register (SPI_DAT) buffer. If the SPI shift register is empty, the byte in the transmission buffer is moved to the shift register, and data transmission will be proceeded. The SPI0 master sends out the data immediately to the MOSI line, and also transmits a serial clock signal on SCK. The SPIF (SPIF) flag is set to logic 1 at the end of the transfer. If interrupts are enabled, an interrupt request is generated when the SPIF flag is set. While SPI0 master transfers data to a slave on the MOSI line, the addressed SPI slave device transfers simultaneously the contents of its shift register to the SPI master on the MISO line with a full-duplex operation. Therefore, the SPIF flag serves as both a transmit-completion and receive-data-ready flag. The data byte received from the slave is thus transferred to the master's shift register with MSB-first mode. When a byte is fully shifted to the register, it is moved to the receive buffer where it can be read by using SPI_DAT

Being configured as a master, SPI0 can operate in one of three different modes: multi-master mode, 3-line single-master mode, and 4-line single-master mode. The default, multi-master mode is actived when NSSMD1 (NSSMD1) = 0 and NSSMD0 (NSSMD0) = 1. In this mode, NSS is an input to the device, and is used to disable the master SPI0 when another master is accessing the BUS. When NSS is pulled down, MSTEN and SPIEN are set to 0 to disable the SPI master device, and “Mode Fault” signal is generated (MODF = 1). “Mode Fault” generates an interrupt if it is enabled. Under this circumstance, SPI0 must be manually re-enabled with software. In multi-master systems, devices will be typically defaulted to be slave devices while they are not acting as the system master device. In multi-master mode, slave devices can be addressed individually (if needed) using general-purpose I/O pins. Figure 7-1 shows a connection diagram between two master devices in multiple-master mode.

3-line single-master mode is selected when NSSMD1 = 0 and NSSMD0= 0. Under this mode, NSS is not used, and is not mapped to external port pin. The slave devices being addressed should use general-purpose I/O pins. Figure 7-2 shows a connection diagram between a master device in 3-line master mode and slave device.

4-line single-master mode is selected when NSSMD1 = 1. Under this mode, NSS is configured as an output pin, and can be used as a slave-select signal for a single SPI device. In this mode, the output value of NSS is controlled (in software) with NSSMD0. Additional slave devices being addressed should use general-purpose I/O pins. Figure 7- 3 shows the connection diagram of the master device in 4-line master mode and two slave devices.

7.1.2.1 Master Mode Configuration

1. Configure NSSMID to select 3-line master mode, 4-line single master mode or multi master mode
2. Configure CKPOL(SPI_CFG[4]) to set clock polarity；
3. Configure CKPHA(SPI_CFG[5]) to set clock phase；

4. Configure MSTEN=1(SPI_CFG[6]) to set to master mode；
5. Configure SPI_SCR to set SCK frequency；
6. Configure SPIEN=1(SPI_CTRL[0]) to enable SPI；
7. Configure SPI_DAT to write an operation data during sending and receiving a data.

7.1.3 SPI Slave Operation

When SPI is enabled and not configured as a master, it will operate as a SPI slave. As a slave, bytes are shifted in through the MOSI pin and out through the MISO pin by a master device controlling the SCK signal. A bit counter in the SPI logic counts the SCK edges. When 8 bits have been shifted through the shift register, the SPIF flag is set to logic 1, and the byte is copied into the receive buffer. Data is read from the receive buffer by reading SPI0DAT. A slave device cannot initiate transfers. Data to be transferred to the master device is pre-loaded into the shift register by writing to SPI_DAT. Writes to SPI_DAT are double-buffered, and are placed in the transmit buffer first. If the shift register is empty, the contents of the transmit buffer will immediately be transferred into the shift register. When the shift register already contains data, the SPI will load the shift register with the transmit buffer's contents after the last SCK edge of the nextSPI transfer.

When configured as a slave, SPI can be configured as 4-wire or 3-wire operation. The default, 4-wire slave mode, is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 1. Under 4-wire mode, the NSS signal is routed to a port pin and configured as a digital input. SPI is enabled when NSS is logic 0, and disabled when NSS is logic 1. The bit counter is reset on a falling edge of NSS. Note that the NSS signal must be driven low at least 2 system clocks before the first active edge of SCK for each byte transfer. Figure 7-3 shows a connection diagram between two slave devices in 4-wire slave mode and a master device.

3-wire slave mode is selected when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 0. NSS is not used in this mode, and is not mapped to an external port pin. Since there is no way of uniquely addressing the device in 3-wire slave mode, SPI must be the only slave device present on the BUS. It is important to note that in 3-wire slave mode there is no external means of resetting the bit counter that determines when a full byte has been received. The bit counter can only be reset by disabling and re-enabling SPI0 with the SPIEN bit. Figure 7-2 shows a connection diagram between a slave device in 3- wire slave mode and a master device.

7.1.3.1 Slave Mode Configuration

1. Configure NSSMID to select 3 wire slave mode or 4 wire slave mode
2. Configure CKPOL(SPI_CFG[4]) to set clock polarity；
3. Configure CKPHA(SPI_CFG[5]) to set clock phase；
4. Configure MSTEN=0(SPI_CFG[6]) to set as slave mode；
5. Configure SPIEN=1(SPI_CTRL[0]) to enable SPI；
6. Configure SPI_DAT，write operation data，wait clock signal send by host.

7.1.4 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 zero 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_DAT is attempted in the cases where the transmit buffer has not been emptied to the SPI shift register. When this happens, the write to SPI_DAT will be ignored, and the transmit 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 MSTEN and SPIEN bits in SPICN 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.

7.1.5 Serial Clock Timing Sequence

Four combinations of serial clock phase and polarity can be selected by using the clock control bits in the SPI0 Configuration Register (SPICFG). 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 line relationships for master mode are shown in Figure 7-4. For slave mode, the clock and data relationships are shown in Figure 7-5 and Figure 7-6.

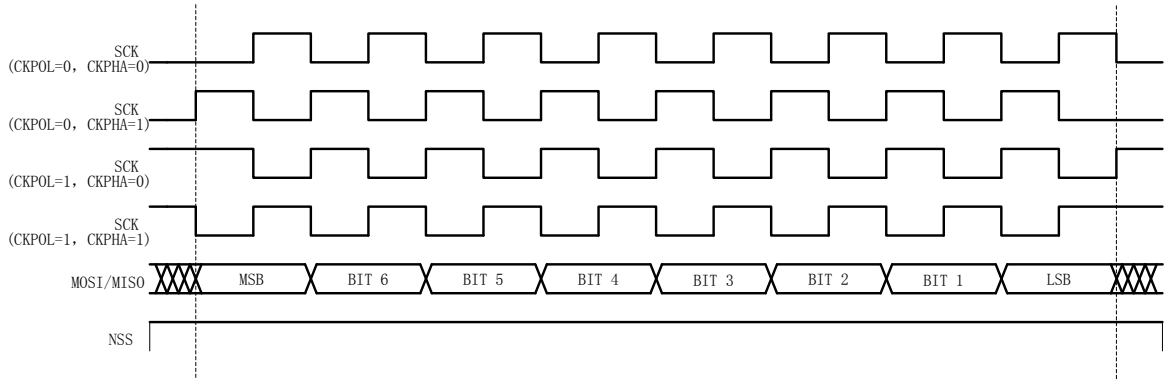


Figure 7-4 Master Mode Data/Clock Timing

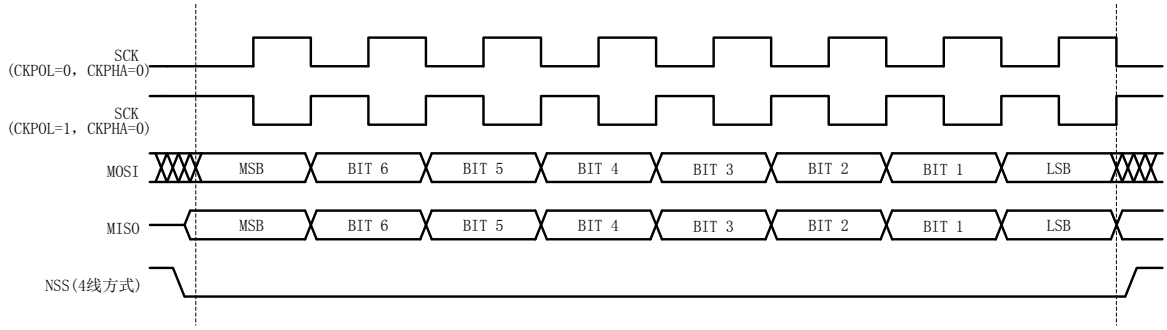


Figure 7-5 Slave Mode Data/Clock Timing (CKPHA = 0)

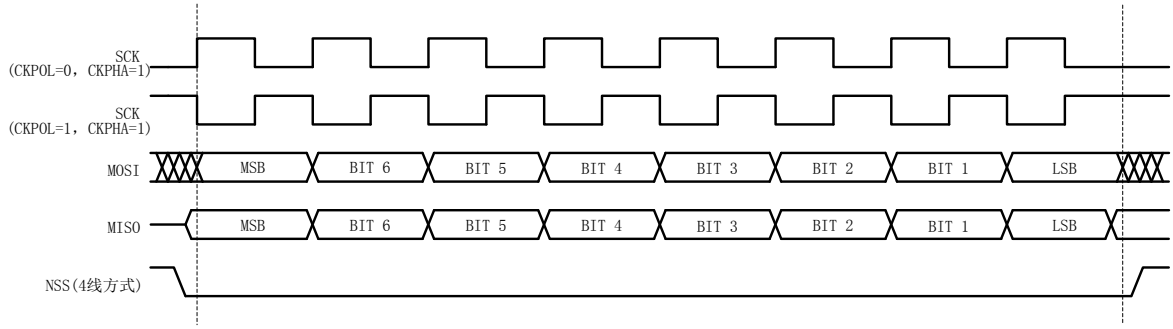


Figure 7-6 Slave Mode Data/Clock Timing (CKPHA = 1)

7.2 SPI Register

7.2.1 SPI_CFG (0x4030)

Table 7-1 SPI_CFG（0x4030）

Bit	7	6	5	4	3	2	1	0
Name	SPIBSY	MSTEN	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]	MSTEN	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 SPI0 is the selected slave. It is cleared to logic 0 when NSS is high (slave not selected). This bit does not indicate the instantaneous value at the NSS pin, but rather a de-glitched version of the pin input.
[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 transmit buffer, or write to the receive buffer. It returns to logic 0 when a data byte is transferred to the shift register from the transmit 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		

7.2.2 SPI_CTRL (0x4031)

Table 7-2 SPI_CTRL（0x4031）

Bit	7	6	5	4	3	2	1	0
Name	SPIF	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]	SPIF	SPI0 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 SPI0DAT is attempted when TXBMT is 0. When this occurs, the write to SPI0DAT 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 SPI0 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 SPI0 shift register. This bit is not automatically cleared by hardware. It must be cleared by software.
[3:2]	NSSMID	Slave Select Mode. 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	Transmit Buffer Empty. This bit will be set to logic 0 when new data has been written to the transmit buffer. This bit will be set to logic 1 when data in the transmit buffer is transferred to the SPI shift register, indicating that it is safe to write a new byte to the transmit buffer.
[0]	SPIEN	SPI0 Enable. 0: SPI disabled. 1: SPI enabled.

7.2.3 SPI_SCR (0x4032)

Table 7-3 SPI_SCR（0x4032）

Bit	7	6	5	4	3	2	1	0
Name	SPI_SCR							
Type	R/W	R/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_SCR	These bits determine the frequency of the SCK output when the SPI module is configured for master mode operation. fsck = sysclk/2x(SPI_SCR[7:0] + 1) for 0<= SPI_SCR <= 255 Example: if sysclk = 24MHz,SPI_SCR=0x04, fsck = 24000000/2x(4+1) = 2400KHz

7.2.4 SPI_DAT (0x4033)

Table 7-4 SPI_DAT（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	Description
[7:0]	SPI_DAT	SPI Transmit 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 transmit buffer and initiates a transfer when in Master Mode. A read of SPI_DAT returns the contents of the receive buffer.

8 UART

8.1 UART Operation Descriptions

8.1.1 Basic Functional Diagram

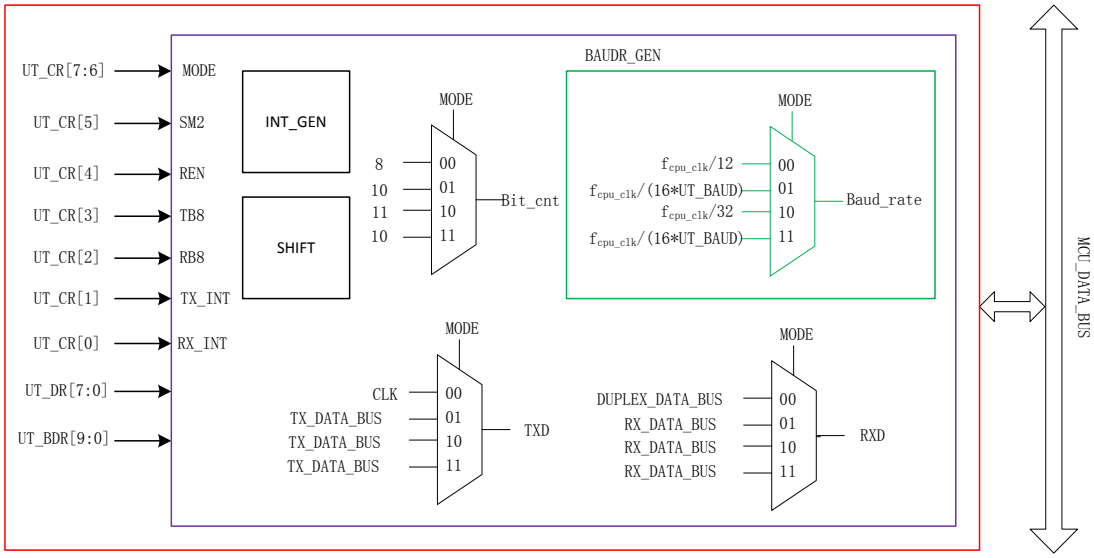


Figure 8-1 Basic Functional Diagram

8.1.2 Mode 0

Synchronous Mode, TXD is clock, RXD is bidirectional data BUS, work in the half-duplex mode with baud rate f_{cpu_clk}/12=2M. LSB will be sent first for 8 bit data, and set UT_CR[REN]=1 to enable receiving operation.

8.1.3 Mode 1

8bit UART Mode. RXD receives data buffer, and TXD sends data to BUS. It supports both full-duplex and half-duplex modes. 10bit protocol: 1bit start (1'b0)+8data bits (LSB first)+1bit stop (1'b1). It will set UT_CR[RB8]=1 after receiving 8 bit data. The baud rate may configured with frequency division according to UT_BAUD, the fomula of baud rate is (f_{cpu_clk}/16)/(UT_BAUD+1), the default value of UT_BAUD is 0x9B, and the default baud rate is 9600 under 24MHz system clock.

8.1.4 Mode 2

9bit UART Mode. It supports also both full-duplex and half-duplex. f_{cpu_clk}/32=750K is the fixed baud rate. 11bit protocol: 1bit start(1'b0)+8data bits(LSB first)+9thbit+1bit stop(1'b1). The value of 9th bit is UT_CR[TB8] in sending data, and the value of 9th bit is UT_CR[RB8] in receiving data.

8.1.5 Mode 3

The basic operation is the similar with mode 2, however, the baud rate configuration is the same as mode 1.

8.1.6 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. Sending nterrupt flag TI will be set to 1 by hardware after UART has completed sending a group of data (8bit for Mode0&Mode1, 9bit for Mode2&Mode3)
2. Receiving interrupt flag RI will be set to 1 by hardware after UART has completed receiving a group of data and stop bit.

8.2 UART Register

8.2.1 UT_CR (0x98)

Table 8-1 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 Tthe baud rate is calculated by the follow formula: Baud rate=f_{cpu_clk}/(16*(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: Baud rate=f_{cpu_clk}/(16*(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 and cleared by software
[3]	TB8	To set the 9th bit of data transmittion 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.

8.2.2 UT_DR (0x99)

Table 8-2 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

8.2.3 UT_BAUD (0x9A,0x9B)

Table 8-3 UT_BAUDH（0x9B）

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

Table 8-1 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
[9:0]	UT_BAUD	Setting of baud rate in the mode1 & mode 3

9 Multiply/DIV Unit (MDU)

Multiply and Division Unit (MDU) is a built-in Multiply and DIV processor, which supports 16bit ×16bit multiplication and 32bit / 32bit division. It supports both unsigned and signed multiplication. The DIV supports unsigned division. MDU mode is set by register MD_MODE.

Multiply mode:

Register (MD_MC3~MD_MC0) = register {MD_MAH、MD_MAL} *

Register {MD_MBH、MD_MBL}

Note:

Registers MD_MC3~2 reuse registers MD_MAH and MD_MAL, while registers MD_MC1~0 reuse registers MD_MBH and MD_MBL. In a multiplication operation, software should write then multiplier to MD_MAH/MD_MAL and then multiplicand to MD_MBH/MD_MBL. The result of the multiplication is read using software, by reading the higher 16 bits from register MD_MAH/MD_MAL, and the lower 16 bits from register MD_MBH/MD_MBL.

One multiplication process consumes single clock cycle time.

Division mode:

MD_DC3~0= MD_DA3~0 / MD_DB3~0.

MD_DD3~0= MD_DA3~0 % MD_DB3~0.

The quotient is saved in register MD_DC3~0, and remainder is saved in register MD_DD3~0.

Note:

Registers MD_DC3~0 reuse registers MD_DA3~0, while register MD_DD3~0 reuse registers MD_DB3~0. In a division operation, Software should write dividend to MD_DA3~0 and divisor to MD_DB3~0. The result of the division is read using software with the quotient from register MD_DC3~0, and remainder from register MD_DD3~0.

One division process consumes about 16 clock cycles. Software can read register DIV_BUSY to check the division status. When DIV_BUSY changes to 0 from 1, it means the division process has end.

9.1 Multiply Using Step:

Step 1: Software sets the register MDSN (MD_MODE[1]) according to the multiplication type. It is set to 0 for unsigned multiplication, and 1 for signed multiplication. Meantime, software sets the register ALIGN(MD_MODE[3:2]) according to the result shift.

Step 2: Software write multiplier to MD_MAH/MD_MAL, and write multiplicand to MD_MBH/MD_MBL.

Step 3: Software reads the higher 16bit of the product from register MD_MAH/MD_MAL, and the lower 16 bits from register MD_MBH/MD_MBL.

9.2 DIV using step:

Step 1: Software writes dividend to MD_DA3~0, and divisor to MD_DB3~0

Step 2: Software sets register DIVSTART(MD_MODE[0]) to 1 for starting the division

Step 3: Software reads register DIVDONE(MD_MODE.7). 1 means division process has been completed, 0 means waiting for new result.

Step 4: Software reads quotient from MD_DA3~0, and remainder from MD_DB3~0.

9.3 DIV NOTE.

NOTE 1:If divisor is set to 0, the DIV will generate an error flag, which is register DIV_ERR(MD_MODE[6]). The flag will be keep to 1, until the MD_DB is not zero in next computing.

NOTE 2:When DIV is busy, the quotion and the remainder is unknown, and it will recover to normal until division process is complished, that is register DIVDONE=1.

NOTE 3:When DIV is busy, changing divisor or division value will not affect the final result, unless set DIVSTART again to restart a new division process.

9.4 MDU Register

9.4.1 MD_MODE (0xC1)

Table 9-1 MDU mode control and status

Bit	7	6	5	4	3	2	1	0
Name	DIVDONE	DIVERR	RSV		ALIGN		MDSN	DIVSTART
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0

Bit	Name	Function
7	DIVDONE	DIV complishment flag 0: DIV is busy 1: DIV process has complished
6	DIVERR	DIV divisor is zero error flag 0: last divison process is wrong (divisor is zero). 1: last divison process is correct (divisor is not zero)
5:4	RSV	Reserved
3:2	ALIGN	The right shift mode setting for multiply process product. 0: not right shift 1: right shift 8bit 2: right shift 12bit 3: right shift 15bit
1	MDSN	Multiply sign setting 0: unsigned computing 1: signed computing
0	DIVSTART	DIV start, software sets this bit to 1 to start DIV computing process. When process is complished, the bit will be cleared by hardware automatically. 0: DIV is not started, last DIV process has been complished. 1: DIV is started and is busy.

9.4.2 MD_MBL (0xCA)

Table 9-2 Multiplier Bit [7:0] (write only), or multiply product Bit [7:0] (read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_MBL							
Type	R/W	R/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]	MD_MBL	Multiplier Bit [7:0](write only) or multiply product [7:0](read only)

9.4.3 MD_MBH (0xCB)

Table 9-3 Multiplier Bit [15:8](write only), or multiply product Bit [15:8](read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_MBH							
Type	R/W	R/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]	MD_MBH	Multiplier Bit [15:8](write only) , or multiply product Bit [15:8] (read only)

9.4.4 MD_MAL (0XC2)

Table 9-4 Multiplicand Bit [7:0] (write only) or multiply product Bit [23:16] (read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_MAL							
Type	R/W	R/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]	MD_MAL	Multiplicand Bit[7:0](write only) or multiply product [23:16](read only)

9.4.5 MD_MAH (0xC3)

Table 9-5 Multiplicand Bit [15:8] (write only) or multiply product [31:24] (read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_MAH							
Type	R/W	R/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]	MD_MAH	Multiplicand Bit [15:8] (write only) or multiply product [31:24] (read only)

9.4.6 MD_DA0 (0xC4)

Table 9-6 Dividend [7:0] (Write only) or quotient [7:0] (Read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_DA0							
Type	R/W	R/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]	MD_DA0	Dividend [7:0] (Write only) or quotient [7:0] (Read only)

9.4.7 MD_DA1 (0xC5)

Table 9-7 Dividend [15:8] (Write only), or quotient [15:8] (Read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_DA1							
Type	R/W	R/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]	MD_DA1	Dividend A[15:8] (Write only), or quotient [15:8] (Read only)

9.4.8 MD_DA2 (0xC6)

Table 9-8 Dividend [23:16] (Write only)or quotient [23:16](Read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_DA2							
Type	R/W	R/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]	MD_DA2	Dividend A[23:16] (Write only),or quotient[23:16] (Read only)

9.4.9 MD_DA3 (0xC7)

Table 9-9 Dividend [31:24] (Write only), or quotient[31:24] (Read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_DA3							
Type	R/W	R/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]	MD_DA3	Dividend A[31:24] (Write only), or quotient[31:24](Read only)

9.4.10 MD_DB0 (0xCC)

Table 9-10 Divisor [7:0] (Write only), or remainder [7:0] (Read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_DB0							
Type	R/W	R/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]	MD_DB0	Divisor [7:0] (Write only), or remainder[7:0] (Read only)

9.4.11 MD_DB1 (0xCD)

Table 9-11 Divisor [15:8] (Write only), or remainder [15:8] (Read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_DB1							
Type	R/W	R/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]	MD_DB1	Divisor [15:8] (Write only), or remainder [15:8] (Read only)

9.4.12 MD_DB2 (0xCE)

Table 9-12 Divisor B[23:16] (Write only),or remainder [23:16] (Read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_DB2							
Type	R/W	R/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]	MD_DB2	Divisor B[23:16] (Write only), or remainder[23:16](Read only)

9.4.13 MD_DB3 (0xCF)

Table 9-13 Divisor [31:24] (Write only), or remainder [31:24] (Read only)

Bit	7	6	5	4	3	2	1	0
Name	MD_DB3							
Type	R/W	R/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]	MD_DB3	Divisor [31:24] (Write only), or remainder[31:24] (Read only)

10 PI

10.1 PI Operation

1、 PI format:

U(K) = U(k-1) + Kp*(E(k) – E(k-1)) + Ki*E(k) ----- (Uk_min < U(k) < Uk_max)

2、 PI controller is started when PISTA bit is set in the PL_CRregister. The operation is completed after 4 MCU clocks, and the result is updated in PI_UK register.

3、 The result of PI controller can truncate to 8~23 bits, with the default truncation set at 12 bits.

4、 The contents U(k-1) and E(k-1) are the previous updates of U(k) and E(k). For example, if U(k) or E(k) are updated, and then PI controller is started, contents of U(k-1), or E(k-1) will that which were updated in U(k) or E(k) before PI controller was started.

MCU has only one PI controller. If the PI controller is used for more applications, initialization should be done before the operation of PI controller.

```
PI_EK = X;                  initialize E(k-1)
SetBit(PL_CR,PISTA,1);      start PI controller
_nop_();
_nop_();
_nop_();
_nop_();                    wait for the completing of PI operation
PI_UK = X;                  initialize U(k-1)
```

10.2 PI Register

10.2.1 PL_CR (0xF9)

Table 10-1 PL_CR（0xF9）

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

Bit	Name	Function
[7:4]	PIRANGE	Truncation of PI output. The default truncation is 12 bit (Input KP, KI can be Q12 format) 0000~1111:8~23
[3:2]	RSV	Reversed
[1]	PISTA	Start PI controller. To be set by software to logic 1. It will be cleared by hardware at next clock. 0：No start 1：Start
[0]	LPFSTA	Start LPF controller. To be set by software to logic 1. It will be cleared by hardware at next clock. 0：No start 1：Start

10.2.2 PI_EK (0xEA , 0xEB)

Table 10-2 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 10-3 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 Range: (-32768, 32767)

10.2.3 PI_UK (0xEC , 0xED)

Table 10-4 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 10-5 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 output of PI controller Range: (-32768,32767)

10.2.4 PI_KP (0xEE , 0xEF)

Table 10-6 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 10-7 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	Proportion coefficient KP Range: (-32768,32767)

10.2.5 PI_KI (0xF2 , 0xF3)

Table 10-8 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 10-9 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	Integral coefficient KI Range: (-32768,32767)

10.2.6 PI_UKMAX (0xF4 , 0xF5)

Table 10-10 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 10-11 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		Maximum limit of UK Range: (-32768,32767)					

10.2.7 PI_UKMIN (0xF6 , 0xF7)

Table 10-12 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 10-13 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	Minimum limit of UK Range: (-32768,32767)

11 Low Pass Filter

11.1 LPF Operation

1、LPF format:

Y(k) = Y(k-1) + LPF_K*(X(k) – Y(k-1))

2、LPF controller is started when LPFSTAbit is set in the PL_CRregister. The operation is completed in 4 MCU clocks, and the result is updated in LPF_Y register.

3、The content of Y(k-1) is the previous update of Y(k). For example, if Y(k) are updated, and then LPF controller is started, contents of Y(k-1) will that which was updated in Y(k) before the LPF controller was started.

11.2 LPF Register

11.2.1 PL_CR (0xF9)

Table 11-1 PL_CR（0xF9）

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

Bit	Name	Function
[7:1]	[7:1] Refer Table 11-1	
[0]	LPFSTA	Start LPF controller. To be set by software to logic 1. It will be cleared by hardware at next clock 0：No start 1：Start

11.2.2 LPF_K (0xDC，0xDD)

Table 11-2 LPF_KH（0xDD）

Bit	7	6	5	4	3	2	1	0
Name	LPF_K[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 11-3 LPF_KL（0xDC）

Bit	7	6	5	4	3	2	1	0
Name	LPF_K[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]	LPF_K	Low-pass filter coefficient Range: (-32768,32767)

11.2.3 LPF_X (0XDE，0XDF)

Table 11-4 LPF_XH（0xDF）

Bit	7	6	5	4	3	2	1	0
Name	LPF_X[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 11-5 LPF_XL（0xDE）

Bit	7	6	5	4	3	2	1	0
Name	LPF_X[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]	LPF_X	Input value Range: (-32768,32767)

11.2.4 LPF_Y (0XE6，0XE7)

Table 11-6 LPF_YH（0xE7）

Bit	7	6	5	4	3	2	1	0
Name	LPF_Y[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 11-7 LPF_YL（0xE6）

Bit	7	6	5	4	3	2	1	0
Name	LPF_Y[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]	LPF_Y	Output Range: (-32768,32767)

12 SVPWM/SPWM/SIN_COS

12.1 SVPWM Operation

The SVPWM module is used under Halll sensored mode. The module is initialized by setting the voltage SV_US, angle SV_ANG, auto-reload value SV_ARR. 5-segment or 7-segment mode can be chosen by using SV5SEG bit in the SV_CRregister. If the SVPWM module is started, the 3-phase drive voltage values are calculated, and the values are automatically stored in the CCR1, CCR2, and CCR3 register. Three PWM signals are output though the configuration of TIMER0.

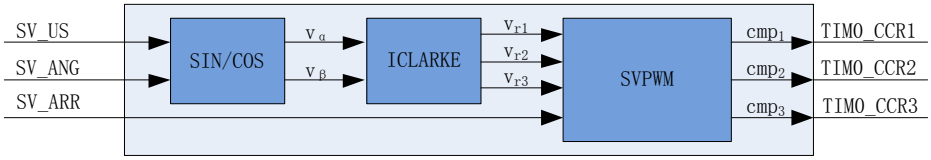


Figure 12-1 SVPWM block diagram

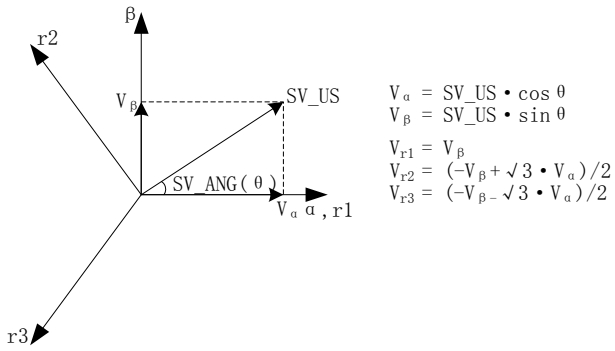


Figure 12-2 Voltage transformation

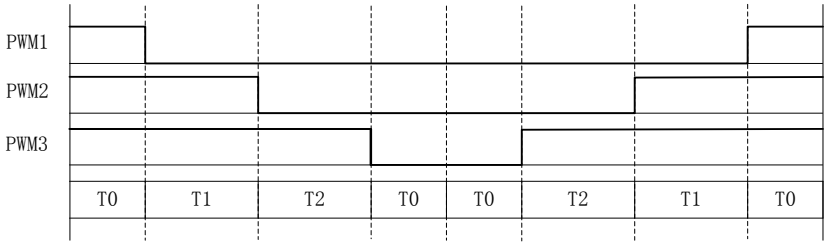


Figure 12-3 7-segment SVPWM output

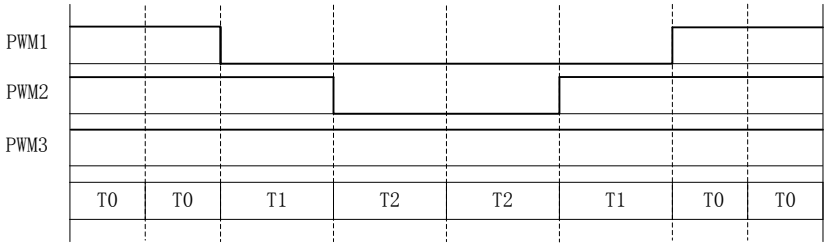


Figure 12-4 5-segment SVPWM output

SVPWM module is started by setting SPWMEN=0, SVPWMEN=1, SVSTA=1 in the

SV_CRregister. When SVPWM module is start, SVBUSY bit will be set in the SV_CRregister. The operation will be completed after 10 MCU clocks, and SVBUSY bit will be set clear, and the result will be updated in CCR1，CCR2，CCR3 register, Vα update in COS_THETA register and Vβ update in SIN_THETA.

12.2 SPWM Operation

The SPWM module is used for single phase induction motor. Set the voltage SV_US, angle SV_ANG, and atuo-reload value SV_ARR; Select unipolar or bipolar by using SPWMSELbit in the SV_CR register. Start SPWM module, the voltage result is equal to SV_ARR*SV_US*SINθ, and the values are automatically stored in the CCR1, CCR2, and CCR3 registers. Three PWM signals are ouputed though the configuration of TIMER0.

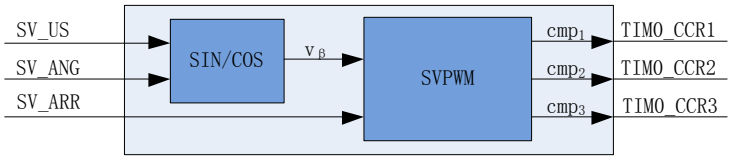


Figure 12-5 SPWM block diagram

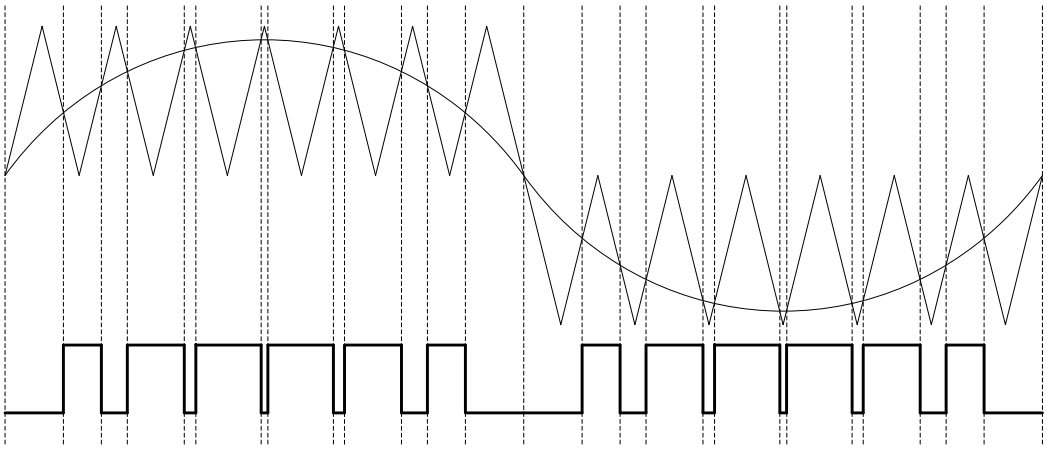


Figure 12-6 SVPW MUnipolar SPWM

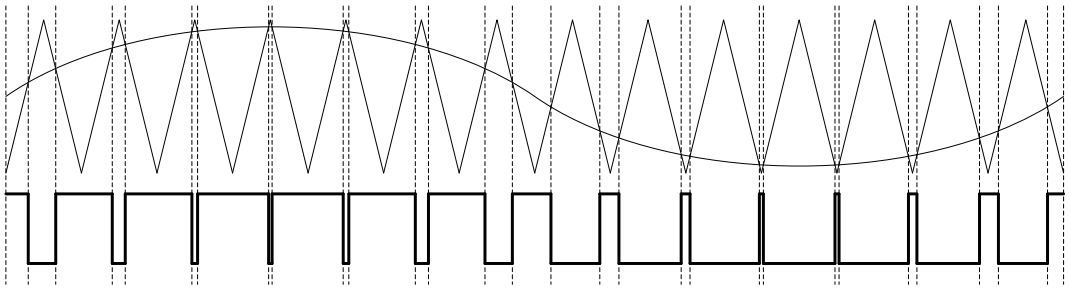


Figure 12-7 SPWM Bipolar SPWM

SPWM module is started by setting SPWMEN=1, SVPWMEN=0, and SVSTA=1 in the SV_CRregister. When it is started, SVBUSY bit will be set in the SV_CRregister, and operation will be completed in 10 MCU clocks. Then SVBUSY bit will be cleared, and the result will be

updated in CCR1，CCR2，CCR3 register, Vα is updated in COS_THETA register and Vβ is updated in SIN_THETA.

12.3 SIN_COS Operation

The SIN_COS operation is used for calculating Vα and Vβ. Set the voltage SV_US = 32760, the result Vα = COSθ and Vβ = SINθ (Q15 format).

SIN_COS module is started by setting SPWMEN=0, SVPWMEN=0, and SVSTA=1 in the SV_CRregister. When SIN_COS module is started, SVBUSY bit will be set in the SV_CRregister, and the operation will be completed in 10 MCU clocks. SVBUSY bit will then be cleared, and the result Vα will be updated in COS_THETA register, and Vβ in SIN_THETA.

12.4 SVPWM/SWPM/SIN_COS Register

12.4.1 SV_CR (0xE1)

Table 12-1 SV_CR（0xE1）

Bit	7	6	5	4	3	2	1	0
Name	SVBUSY	RSV		SPWMSEL	SPWMEN	SV5SEG	SVPWMEN	SVSTA
Type	R/W	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	SVBUSY	SVPWM/SPWM/SIN_COS busy state: 0: No busy 1: Busy
[6:5]	RSV	Reversed
[4]	SPWMSEL	SPWM polarity selection: 0: Unipolar 1: Bipolar
[3]	SPWMEN	SPWM enable SPWMEN/SVPWMEN = 00：SIN_COS SPWMEN/SVPWMEN = 01：SVPWM SPWMEN/SVPWMEN = 10：SPWM SPWMEN/SVPWMEN = 11：SPWM
[2]	SV5SEG	SVPWMmodulation: 0: 7-segment; 1: 5-segment.
[1]	SVPWMEN	SVPWM enable SPWMEN/SVPWMEN = 00：SIN_COS SPWMEN/SVPWMEN = 01：SVPWM SPWMEN/SVPWMEN = 10：SPWM SPWMEN/SVPWMEN = 11：SPWM
[0]	SVSTA	SVPWM/SPWM/SIN_COS is started with a software write of logic 1. It will be cleared by hardware at the next clock 0 : No start 1 : Start

12.4.2 SV_US (0xE2 , 0xE3)

Table 12-2 SV_USH（0xE3）

Bit	7	6	5	4	3	2	1	0
Name	SV_US[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 12-3 SV_USL（0xE2）

Bit	7	6	5	4	3	2	1	0
Name	SV_US[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]	SV_US	Input voltage, range is (0~32760). Modulation ratio in SPWM mode: M=(SV_US+1)/32768*100%

12.4.3 SV_ANG (0xE4 , 0xE5)

Table 12-4 SV_ANGH（0xE5）

Bit	7	6	5	4	3	2	1	0
Name	SV_ANG[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 12-5 SV_ANGL（0xE4）

Bit	7	6	5	4	3	2	1	0
Name	SV_ANG[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	R/W	Reset
[15:0]	SV_ANG	Input angle of SVPWM/SPWM/SIN_COS Range: (0~32767) Angle= SV_ANGLE/32768*360°	RW	0x0

12.4.4 SV_ARR (0x4060 , 0x4061)

Table 12-6 SV_ARRH（0x4060）

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

Table 12-7 SV_ARRL（0x4061）

Bit	7	6	5	4	3	2	1	0
Name	SV_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
[15:0]	SV_ARR	The auto-reload value, and used with TIM0_ARP

12.4.5 SIN_THETA (0x407C , 0x407D)

Table 12-8 SIN_THETAH（0x407C）

Bit	7	6	5	4	3	2	1	0
Name	SIN_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 12-9 SIN_THETAL（0x407D）

Bit	7	6	5	4	3	2	1	0
Name	SIN_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]	SIN_THETA	Sinθ is calculated by using SV_US and SV_ANG

12.4.6 COS_THETA (0x407E , 0x407F)

Table 12-10 COS_THETAH（0x407E）

Bit	7	6	5	4	3	2	1	0
Name	COS_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 12-11 COS_THETAL（0x407F）

Bit	7	6	5	4	3	2	1	0
Name	COS_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]	COS_THETA	Cosθ is calculated by using SV_US and SV_ANG

13 FOC

13.1 FOC Operation

13.1.1 Introduction

As an independent module, the FOC clock stops when the FOC module is not working. It must be enabled by setting the FOCEN bit in the FOC_SET0 register before the operation of the FOC module.

FOC module contains a angle module, a PI controller, a coordinate transform module and a 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 control: the angle estimation module is used for coordinate transforms. Meanwhile, the angular velocity and back-EMF are provided to realize the velocity closed loop and detect the operation starting.

B）HALL-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.

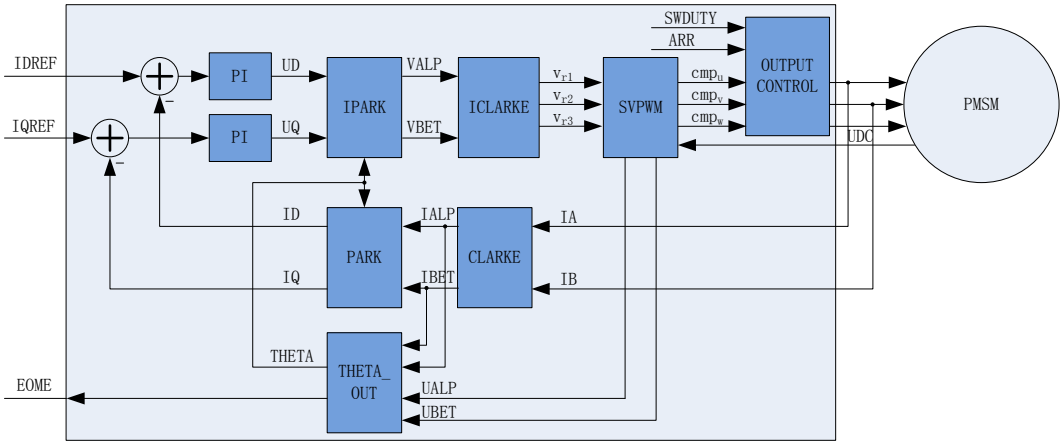


Figure 13-1 FOC functional block diagram

13.1.2 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 velocity signal EOME, together with the MCU or PI module.

13.1.3 PI controller

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

1 Rotor flux control: PI controller of D-axis, reference current IDREF minus feedback current ID as the deviation 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 Rotor torque control: PI controller of Q-axis, reference current IQREF minus feedback

current IQ as the deviation 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 estimate: PI controller of estimator, proportional coefficient EKP and the integral coefficient EKI for adjustment of PI performance. The estimated angle ETHETA is provided.

The DQPIRAN bit in FOC_PIRAN register sets the result truncation bits of both D-axis and Q-axis PI controllers, and the EPIRAN sets the result intercept bit of the estimator alone. This function is equivalent to set the data format of PI controller, where the defalt value is 12 bits, i.e., the data format is Q12. If the truncation bit is set to 15 bits, the data format is Q15, and so on.

13.1.4 Coordinate Transforms

13.1.4.1 Inverse Park transform

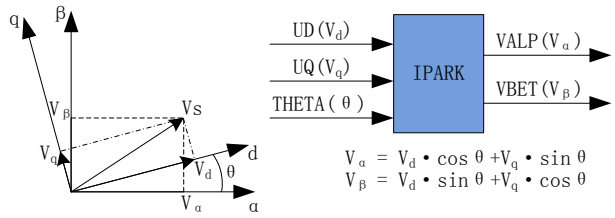


Figure 13-2 Inverse Park transform

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.

13.1.4.2 Inverse Clarke transform

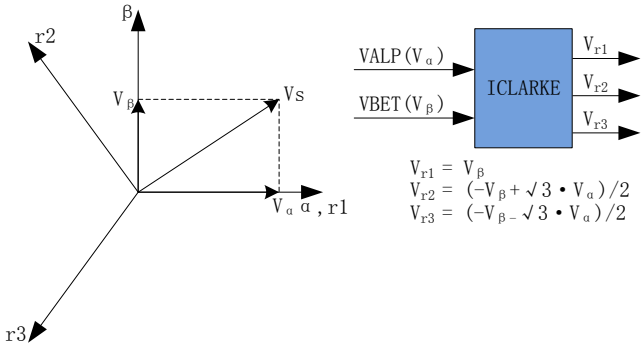


Figure 13-3 Inverse Clarke transform

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

13.1.4.3 Clarke transform

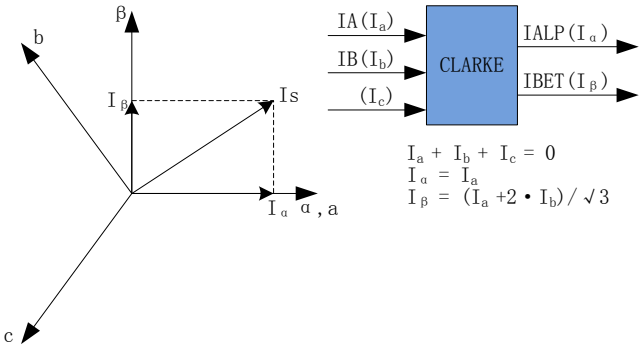


Figure 13-4 Clarke transform

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

13.1.4.4 Park transform

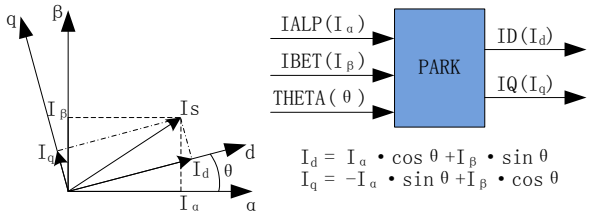


Figure 13-5 park transform

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

13.1.5 SVPWM

Space vector pulse width modulation (SVPWM) algorithm is an important part of FOC control. 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 and the harmonic losses of the motor, reducing the torque ripple and have a higher utilization space.

SVPWM generate pulse-width modulation signals for the 3-phase motor voltage control. The process of generating the pulse width for each of the three phases is reduced to a few simple equations. Each of the three inverter outputs can be in one of two states. The inverter output can be connected to eitherthe plus (+) BUS rail or the minus (-) BUS rail, which allows for $2^3 = 8$ possible states of the output. 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 star. The remaining six states are represented as vectors with 60 degree rotation between each

state.

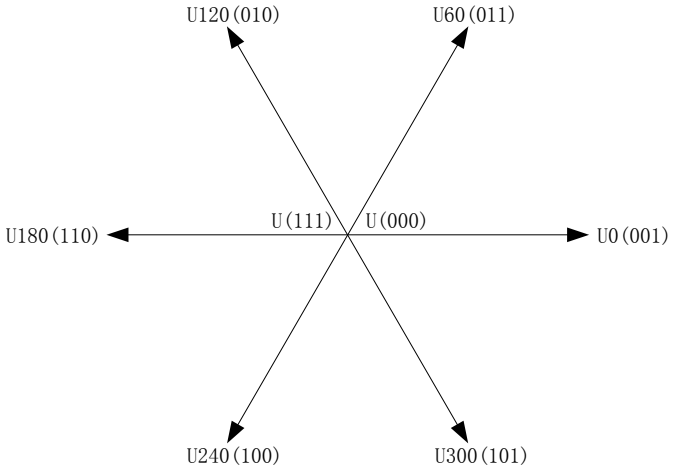


Figure 13-6 SVPWM vector control

The process of SVPWM allows the representation of any resultant vector with the sum of the components of two adjacent vectors. Suppose that UOUT is the desired vector and it lies in the sector between U60 and U0. UOUT is then represented as a time average where during a given PWM period T, U0 is output for T1/T and U60 is output for T2/T. T0 represents a time where no effective voltage is applied into the windings, that is, where a null vector is applied.

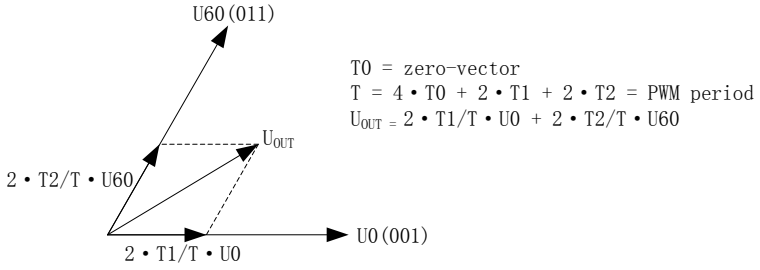


Figure 13-7 AVERAGE SVPWM

Table 13-1 Inverter states of space vector modulation

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)

13.1.5.1 Seven-segment SVPWM

In the single-resistor current sampling mode, FOC algorithm is fixed to seven-segment SVPWM. In the dual-resistors current sampling mode, user can set F5SEG=0 in FOC_CR2 register to choose the seven-segment SVPWM as the output mode.

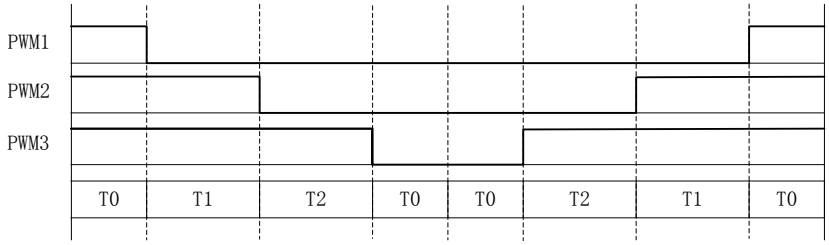


Figure 13-8 Ouput level of the seven-segment SVPWM

13.1.5.2 Five-segment SVPWM

Five-segment SVPWM is only used in the dual-resistors current sampling mode. User is required toset F5SEG=1 in FOC_CR2.

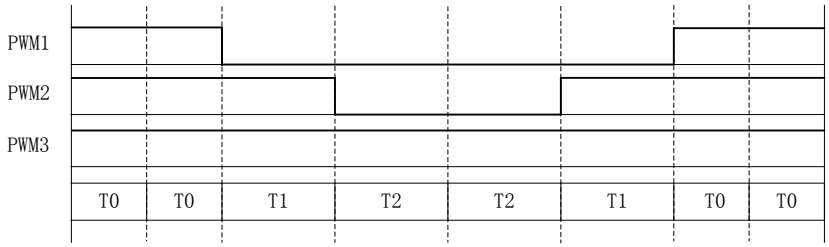


Figure 13-9 Output level of the five-segment SVPWM

13.1.6 Current & voltage sampling

FOC module needs to sample the BUS voltage of the controller and the two of the three phase current, while remaining phase current is obtained by calculation. Before using the FOC module, ADC (set ADCEN=1 in ADC_STA), amplifiers, and related registers must be set. However, there is no need to set the ADC channels and the scanning mode. CSM in FOC_CR1 register must be set to select between single resistor current sampling mode or dual resistors mode. In single resistor current sampling mode, the default sampling channel of the BUS current is channel-4, while in dual resistors mode, channel-0 is the default sampling channel for current i_a and channel-1 is for i_b. UCH bit in FOC_CHC register select the channel for sampling the BUS voltage, but the default is channel-2.

13.1.6.1 Singel resister sampling mode

Set CSM=0 in FOC_CR1 register to choose the single resistor 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 aftering FOC module finishing the operation during the counter with a count-down operation.

13.1.6.2 Dual resisters sampling mode

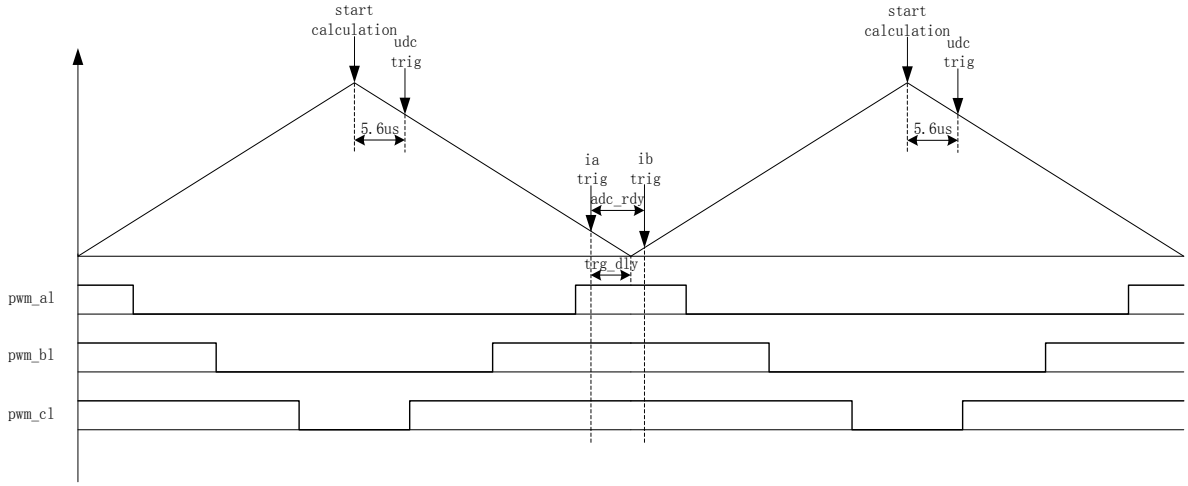


Figure 13-10 Dual resisters sampling mode

Set CSM=1 in FOC_CR1 register, and DSS=0 in FOC_CR2 register, to select dual resistors current sampling mode. In this sampling mode, the sampling time of current i_a (channel-0) can be set by TRG_DLY register, and i_b (channel-1) sampled at the end of the sampling i_a. The BUS voltage can be sampled aftering FOC module has finished the operation during a counter count-down operation. Attention should be paid such that the setting of the current sampling time should make the sampling points of i_a and i_b both in the zero vector space (pwm_al,pwm_bl,pwm_cl=111). If the MCU clock is 24MHz(41.67ns) and TRGDLY = 0x8032, i_a should be sampled before the minima of the counter 41.67*50=2.08us during the counter of the FOC with a conut-down operation. i_b, should be sampled after completing the i_a sampling.

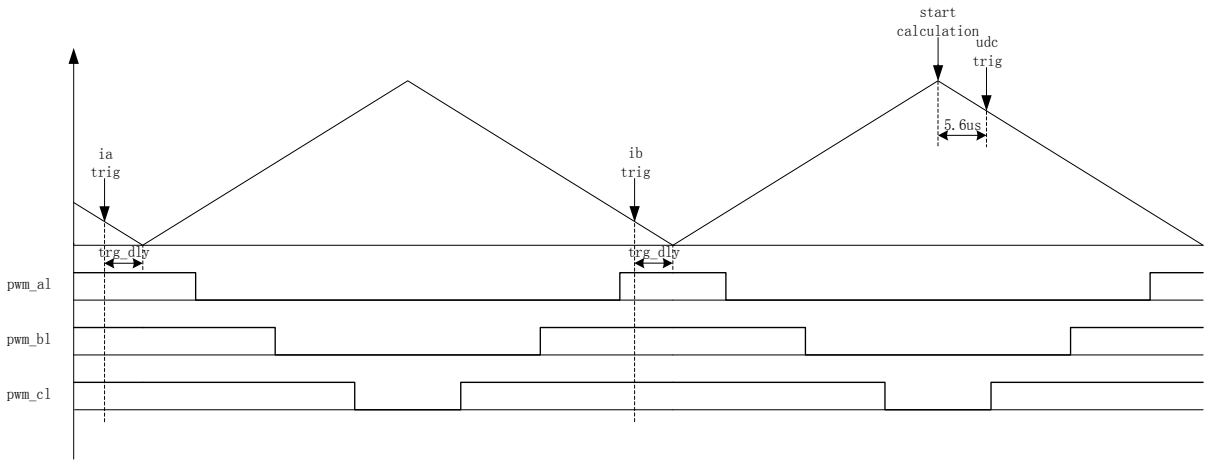


Figure 13-11 Current sampling with dual-resistors mode

Set CSM=1 in FOC_CR1 register and DSS=1 in FOC_CR2 register to select dual resistors current sampling mode. Using this mode, the two carrier periods is considered as one operation

period. Only one phase current is sampled at each carrier period, that is, i_a is sampled in the first carrier period and i_b in the second period. After the completing the sampling of the i_b, before the maxima of the counter, FOC module starts its operation. At the end of the operation, BUS voltage will be sampled.

13.1.6.3 Current sampling reference

The input current sampling range should be doubled before the sampling. This is due to the existence of the positive and negative phase current. So it is required to minus the reference value during an 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 offsets, so usually the user will required to do a a calibration. The calibration procedure is as the following:

When FOC module dosen't work, and there is no current in the 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 reference voltage is 2.5V, FOC_CSO = 2.5/5V*32768 = 16384(0x4000).

1. When FOC_CHC[CSOC]=00/11, write FOC_CSO to modify the reference value of IPRIP.
2. When FOC_CHC[CSOC]=01, write FOC_CSO to modify the reference value of I_A.
3. When FOC_CHC[CSOC]=10, write FOC_CSO to modify the reference value of I_B.

13.1.7 Angle mode

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

1. Ramp force angle
2. Force pull angle
3. Estimator estimation angle
4. Estimator force angle

Table 13-2 Source of the angle module

RFAE	ANGM	EFAE	Source of the angle
1	x	x	Ramp force angle
0	0	x	Force pull angle
0	1	0	Estimation angle of estimator
0	1	1	Estimated Velocity>EFREQMIN: Estimator estimation angle Estimated Velocity<EFREQMIN: Estimator force angle

13.1.7.1 Ramp force angle

Ramp force angle consists of angle THETA, velocity RTHESTEP, acceleration RTHEACC and ramp counter RTHECNT. The ramp formula is:

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

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

Ramp force angle has a higer priority. Set the RFAE = 1 in FOC_CR1 register to enable the ramp function. Ramp module makes a ramp operation in every cycle. When the value of the counter reaches RTHECNT, RFAE is cleared by hardware, and then the ramp is over. Thereafter, if ANGM=1 in the FOC_CR1 register, the angle comes from estimator, while ANGM=0, the angle comes from Force pull angle.

13.1.7.2 Force pull angle

Force pull angle consists of angle THETA and the velocity RTHESTEP. The formula is:
THETA (16bit) = THETA (16bit) + RTHESTEP (high16bit)

There are two states in the force pull angle:

1. Setting RFAE = 1 and ANGM = 0 in FOC_CR1 register, and the strong-pull mode is started after ramp function. The speed RTHESTEP will then be the cumulative result at the end of the ramp. This mode may achieve uniform and force pull without angle feedback function.
2. Setting RFAE = 0 and ANGM = 0 in FOC_CR1 register, and the force pull angle is started without going through the ramp module. The speed RTHESTEP will then be the initial speed writing in the register by software. When RTHESTEP = 0, localization function is realized. When RTHESTEP != 0, HALL FOC control is enabled. (HALL FOC control 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).

13.1.7.3 Estimation angle of the estimator

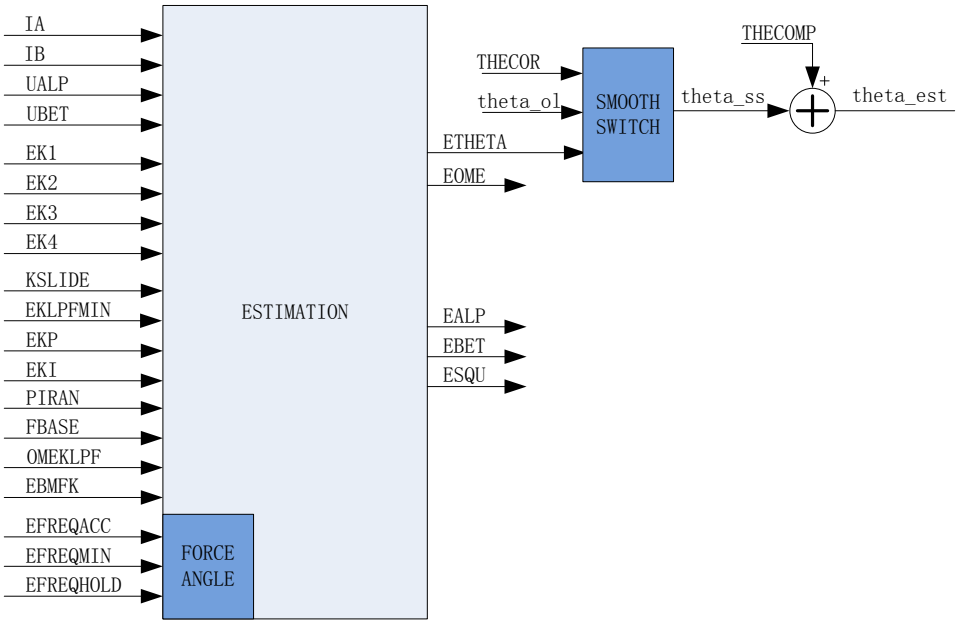


Figure 13-12 Estimator functional block diagram

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

1. Estimation angle of the estimator

The estimator builds the motor model based on the motor parameters and control parameters, and samples the 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 with the PLL mode and sliding mode by setting the ESEL in FOC_CR1 register.

2. Force estimation angle of the estimator

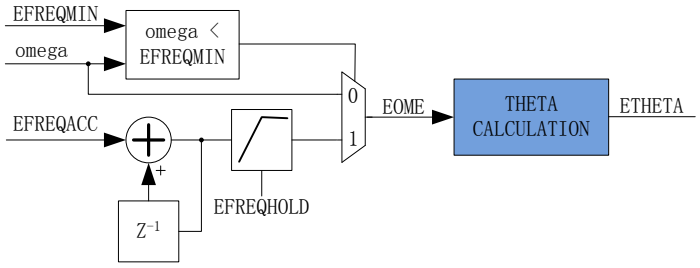


Figure 13-13 Force estimation angle functional block diagram

This function is familiar as the ramp function. Due to the small ouput at motor starting, there can be a huge difference in angle estimation and the speed with the small sampling current. This may result in the launch failure. In this case, the estimator outputs the force pull angle to make the motor start successfully.

Set EFAE=1 in FOC_CR1 register. When the estimated speed from the estimator is less than the minimum value setted, the system will be started compulsorly 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 foced speed. When omega is greater than or equal to EFREQMIN, the final sped EOME is the estimated speed.

3. Smooth switching on the estimating angle

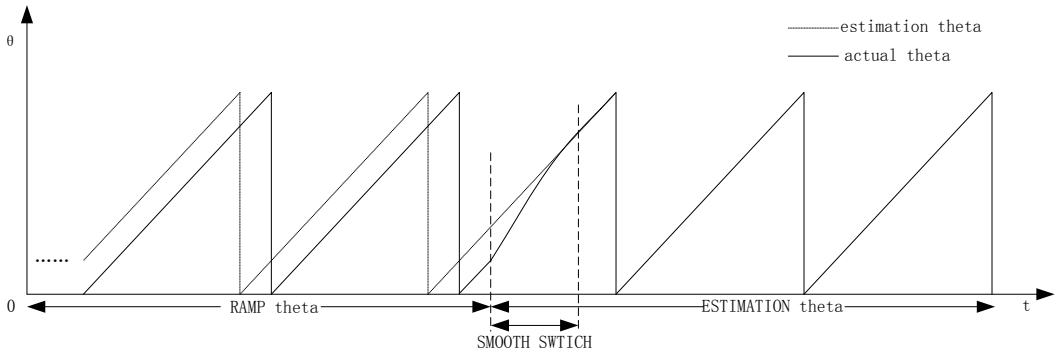


Figure 13-14 Smooth switching curve of the estimating angle

Setting the RFAE = 1 and ANGM = 1 in FOC_CR1 register, the motor operates with the ramp function. At the end of the ramp, it switchs to estimator mode. The estimator also estimates the angle in the ramp, but usually there is a deviation between the estimated angle and the ramp

forced angle. If the angle is switched from ramp forced angle to estimated angle after the ramping, there could be motor jitter due to angle difference, so the smooth switching module of the estimating angle is needed.

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

4. Angle compensation

The value THECOMP is used to compensate the estimated angle. When the first bit is 1, it is the negative compensation; on the contrary, it is the positive compensation.

13.1.8 Output module

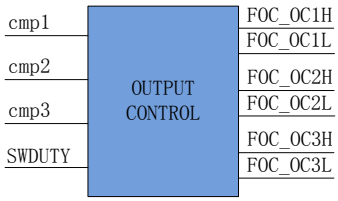


Figure 13-15 Output module diagram

FOC_OC1H/FOC_OC1L, FOC_OC2H/FOC_OC2L, and FOC_OC3H/FOC_OC3L are three pairs of complementary output, and can be used for dead zone insertion. FOC_OCxL is the output of the lower bridge, the same phase as OCxREF. FOC_OCxH is the output of higher bridge, the complementary output of OCxREF. The user may set the CCxEH and CCxEL in FOC_CMR register to choose the output mode as inactive or complementary output with dead zone insertion. CCPH and CCPL in FOC_CMR register are set to configure the polarity of the output.

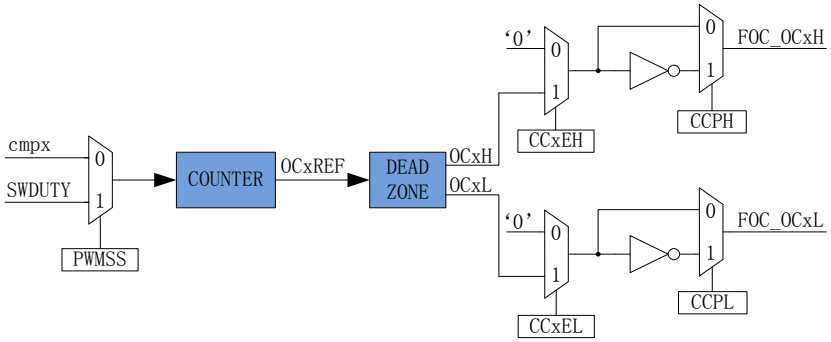


Figure 13-16 FOC ouput mode block diagram

The three complementary outputs duty cycle of FOC are set by PWMSS in FOC_CR1 register. The reference can be derived from CMP1/CMP2/CMP3 within the internal SMPWM module or SWDUTY given by the user. The three PWM signals OCxREF will be obtained after comparing the reference to the counter. When the counter is less than the reference, OCxREF outputs a

high, otherwise it outputs a low.

Set the PWMSS=0 in FOC_CR1 register to set reference values (cmp1, cmp2 and cmp3) of SVPWM module, to compare with counter. The comparison will then generate OC1REF/OC2REF/OC3REF.

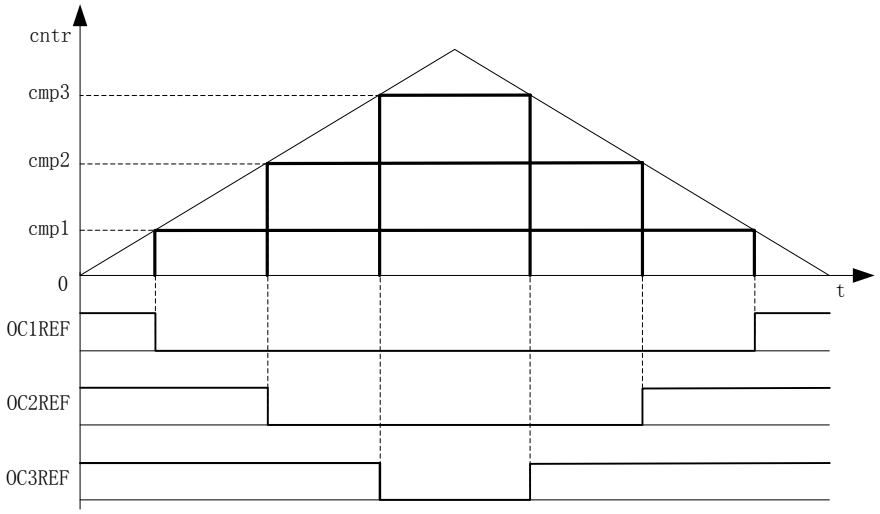


Figure 13-17 PWM spanning graph

Set the PWMSS=1 in FOC_CR1 register to set SWDUTY. OC1REF/OC2REF/OC3REF will then be generated for that duty-cycle.

duty-cycle = SWDUTY / ARR x 100% (suppose that ARR=750, SWDUTY=375, the duty-cycle=50%)

Together with CCxEH and CCxEL in FOC_CMR register, functions such as pre-charge and brake may be achieved. SWDUTY is to control PWM duty-cycle, and CCxEH and CCxEL is to control the six output modes.

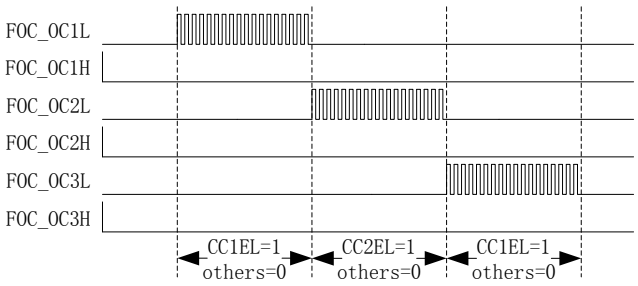


Figure 13-18 Pre-charge graph

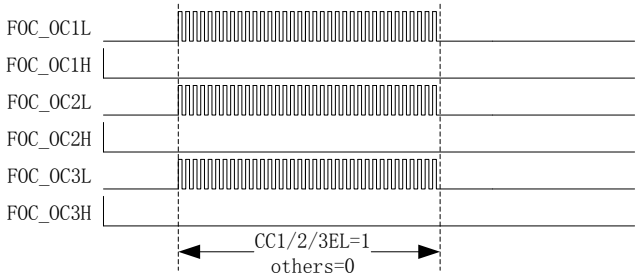


Figure 13-19 Brake graph

For the complementary output, the dead zone insertion is enabled wih FOC_DTR register by setting it not equal to zero. Each channel has an 8-bit dead zone generator. The three channels share the same register (FOC_DTR). When OCxREF at the rising edge, the actual high output at OCxL is delayed by TIM0_DTR, compared to the rising edge of OCxREF. When OCxREF at the falling edge, the actual high output at OCxH delayed by TIM0_DTR, compared to the rising edge of OCxREF. If delay time is greater than the real output pulse width, the related channel pulse width will have no latency and the complementary channel pulse width will not be generated.

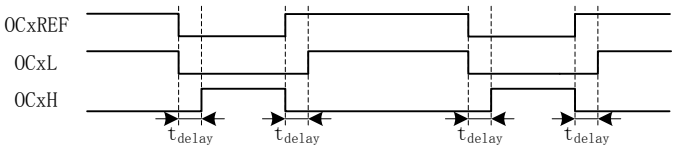


Figure 13-20 Complementary output with dead zone insertion

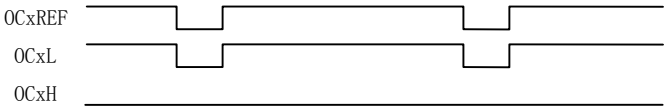


Figure 13-21 Dead zone time is larger than negative level

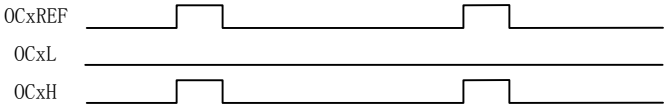


Figure 13-22 Dead zone time is larger than positive level

13.1.9 Motor realtime parameters

FOC module generates the following realime parameters in motor control

1. Angle: THETA
2. Estimated angle: ETHETA; Estimated speed: EOME
3. Voltage in D-axis: UD; Voltage in Q-axis: UQ

4. Current in D-axis: ID; Current in Q-axis: IQ
5. Alpha-axis voltage: VALPI; Beta-axis voltage: VBET
6. UALP and UBET calculated with BUS voltage: UBET
7. Phase current I_A and I_B
8. IALP and IBET obtained with Calrke transform
9. EALP and EBET
10. The squared of the back-emf: ESQU
11. Power: POW

13.1.9.1 Forward and reverse rotation detection

FOC provides initial forward and reverse rotation detection module. Setting ESCMS=1 in FOC_CR2 register and the reference input current IDREF and IQREF to 0, and with the FOC module enabled, 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 swich to estimator angle mode to start motor.

13.1.9.2 Back-EMF detection

Estimator estimates the α-axis Back-emf EALF and the β-axis Back-emf EBET with the input motor parameters, and outputs eα2 +eβ2, 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.

13.1.9.3 Power

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

13.1.10 Interrupt

13.1.10.1 Underflow interrupt

The underflow interrupt event is generated when internal counter in FOC counts down to zero. The interrupt is enabled by setting FUIE in FOC_CR2 register. FIM determines the frequency of the Interrupt generation cycle, that is, interrupt is generated after 1/2/3/4 underflow event. The underflow interrupt flag is set to 1 by hardware and is cleared by software.

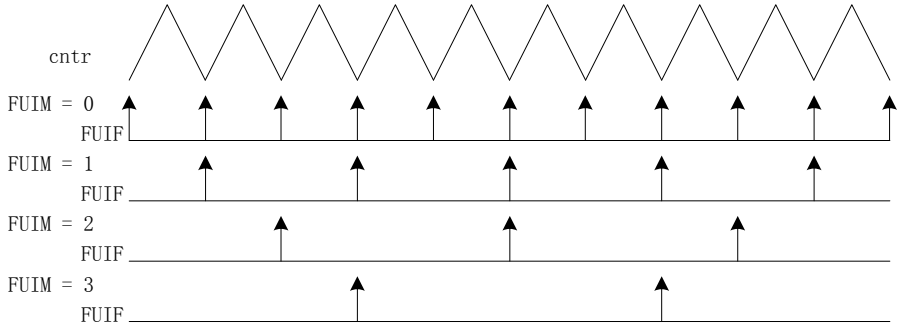


Figure 13-23 FOC underflow interrupt

13.1.10.2 Comparison and matching interrupt

Set FCIM in FOC_CR3 register to configure point of comparison. The reference value is set in FOC_COMR. When the value of the internal counter in FOC is equal to FOC_COMR, interrupt is generated according to the setting of FCIM. The interrupt is enabled by setting FCIE in FOC_CR2 register.

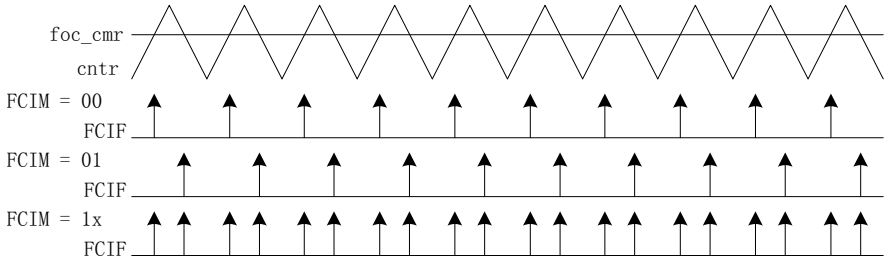


Figure 13-24 FOC comparison and matching interrupt

13.1.11 FOC observer

FOC observer sends the motor realtime parameters to the host-computer with the SPI mode for realtime display. It is enabled by setting DBEN=1 in FOC_CR2 register. The user selects the data to be transmitted in the channels by setting the FOC_FDS register and FOC observer will send the data. It has four channels which divided into two groups, one is channel-1 and channel-2, and another is channel-3 and channel-4.

Table 13-3 FOC observer output variables

FDS12/FDS34	Channel 1/3	Channel 2/4	Register value	Channel 1/3	Channel 2/4
0000	THETA_SS（estimated angle with soft switch）	EOME（estimated speed）	1000	EALP	IALP
0001	THETA（real output angle）	EOME（estimated speed）	1001	ESQU	POW
0010	UD（D-axis voltage）	UQ（Q-axis voltage）	1010	IA	IB

0011	ID（D-axis current）	IQ（Q-axis voltage）	1011	IB	IC
0100	UALP	UBET	1100	UDCFLT（bus voltage ）	RTHESTEP（ramp step）
0101	IALP	IBET	1101	UALP	VALP
0110	EALP	EBET	1110	THETA_SS（estimated angle with soft switch）	ETHETA（real output angle）
0111	VALP	VBET	1111	IALP	IALP_EST（estimated IALPHA current）

13.2 FOC Register

13.2.1 FOC_SET0 (0xE8)

Table 13-4 FOC_SET0（0xE8）

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

Bit	Name	Function
[7]	FOCEN	FOC module enable FOCEN must be set 1 before operation. FOC module will be reset when FOCEN bit is cleared. 0: disable 1: enable
[6:0]	RSV	Reserved

13.2.2 FOC_SET1 (0xE9)

Table 13-5 FOC_SET1（0xE9）

Bit	7:3					2:1		0
Name	RSV					RSV		FOCFR
Type	R	R	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:1]	RSV	Reserved
[0]	FOCFR	FOC direction 0: CW 1: CCW

13.2.3 FOC_FDS (0x408F)

Table 13-6 FOC_FDS（0x408F）

Bit	7	6	5	4	3	2	1	0
Name	FDS34				FDS12			
Type	R/W	R/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]	FDS34	FOC debug module, channel 3,4 output selectable
[3:0]	FDS12	FOC debug module, channel 1,2 output selectable

13.2.4 FOC_CR1 (0x4090)

Table 13-7 FOC_CR1（0x4090）

Bit	7	6	5	4	3	2	1	0
Name	ESEL	EFAE	RFAE	ANGM	CSM	FCE	PWMSS	FOCST
Type	R/W	R/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 selection 0：SMO (Sliding mode) 1：PLL, when ESEL is set to 1, the FOC_KSILDE register is KP in the PLL PI controller, and the FOC_KLPFMIN register is KI in the PLL PI controller.
[6]	EFAE	ESTIMATOR Force angle enable When EFAE set 1, angle is provided by estimator, and auto switch to angle close loop 0: Disable 1: Enable
[5]	RFAE	Force ramp angle enable When RFAE set 1, angle is provided by ramp module. At the end of ramp, RFAE is cleared and switch to estimator mode or force pull mode
[4]	ANGM	Angle mode RFAE=0, the angle is from estimator

		RFAE=1, the estimator is swiched off, or enter force-pull mode 0: Estimator mode 1: Force pull mode
[3]	CSM	Current sampling mode 0: Single resistor sampling 1: Dual resistor sampling
[2]	FCE	FOC counter enable 0: Disable 1: Enable
[1]	PWMSS	PWM SOURCE slection PWM source is from FOC closed loop or software. Setting FOC will allow for current closed-loop. Setting software will allow for implementation of precharging or braking. 0: From FOC close loop 1: From software
[0]	FOCST	FOC START FOCST is set to logic 1 by software. It is cleared by hardware in the next clock 0: No start 1: Start

13.2.5 FOC_CR2 (0x4091)

Table 13-8 FOC_CR2（0x4091）

Bit	7	6	5	4	3	2	1	0
Name	ESCMS	RSV	F5SEG	DSS	RSV			DBEN
Type	R/W	R/W	R/W	R/W	R	R	R	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	ESCMS	ESTIMATION Speed calculate mode SELECT 0: Normal mode 1: Forward or reverse detection mode
[6]	RSV	Reserved
[5]	F5SEG	SVPWM at dual resistor sampling mode 0: Seven-segment mode 1: Five-segment mode
[4]	DSS	Sampling timing at dual resistor sampling mode 0: Concurrent mode 1: Alternating mode
[3:1]	RSV	Reserved
[0]	DBEN	FOC observer enable

		0: disable 1: enable Note: After enabling the FOC debug port, SPI communication port will be occupied.

13.2.6 FOC_CR3 (0x4092)

Table 13-9 FOC_CR3（0x4092）

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

Bit	Name	Function
[7:4]	RSV	Reserved
[3:2]	FCIM	FOC compare interrupt mode Select the counting direction for FCIM. When counter value is equal to FOC_COMR, interrupt flag is set to 1 00: Up counting direction 01: Down counting direction 1x: Up/Down counting direction
[1:0]	FUIM	FOC underflow interrupt mode FOC underflow interrupt trigger for N carrier cycles. 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles

13.2.7 FOC_IER (0x4093)

Table 13-10 FOC_IER（0x4093）

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

Bit	Name	Function
[7:2]	RSV	Reserved
[1]	FCIE	Compare enable 0: Disable 1: Enable
[0]	FUIE	Underflow interrupt enable 0: Disable 1: Enable

13.2.8 FOC_SR (0x4094)

Table 13-11 FOC_SR（0x4094）

Bit	7	6	5	4	3	2	1	0
Name	EUOF	RSV					FCIF	FUIF
Type	R/W0	R	R	R	R	R	R/W0	R/W0
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	EUOF	Error voltage overflow flag Voltage UALP, UBET is calculated by UDC and PWM duty. When UDC and PWM are large, UALP and UBET will overflow. This flag is set by hardware, and cleared by software. 0: No event occurred 1: Error voltage overflow interrupt pending
[6:2]	RSV	Reserved
[1]	FCIF	Compare interrupt flag Select the counting direction by FCIM. When counter value is equal to FOC_COMR, interrupt flag is set to 1 This flag is set by hardware, and cleared by software. 0: No event 1: Compare interrupt pending
[0]	FUIF	Underflow interrupt flag When counter value is equal to 0, the underflow event occurs. This flag is set by hardware, and cleared by software. 0: No event occurred 1: Underflow interrupt pending

13.2.9 FOC_CHC (0x4095)

Table 13-12 FOC_CHC（0x4095）

Bit	7	6	5	4	3	2	1	0
Name	CSOC		ITCH			UCH		
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	0	0	0	1	0

Bit	Name	Function
[7:6]	CSOC	Current sample offset calibrate Set CSOC, write the FOC_CSO register to set the current sample offset 00,11: itrip 01: i_a 10: i_b
[5:3]	ITCH	ITRIP channel select 000: channel 0 001: channel 1 010: channel 2 011: channel 3 100: channel 4 101: channel 5 110: channel 6 111: channel 7
[2:0]	UCH	UDC channel select 000: channel 0 001: channel 1 010: channel 2 011: channel 3 100: channel 4 101: channel 5 110: channel 6 111: channel 7

13.2.10 FOC_PIRAN (0x4096)

Table 13-13 FOC_PIRAN（0x4096）

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

Bit	Name	Function
[7:4]	EPIRAN	Truncation of estimator PI output. The default is 12 bit (The input KP and KI's format is Q12) 0000~1111:8~23
[3:0]	DQPIRAN	Truncation of D/Q axis PI output. The default is 12 bit (The input KP and KI's format is Q12) 0000~1111:8~23

13.2.11 FOC_CMR (0x4097)

Table 13-14 FOC_CMR（0x4097）

Bit	7	6	5	4	3	2	1	0
Name	CCPH	CCPL	CC3EH	CC3EL	CC2EH	CC2EL	CC1EH	CC1EL
Type	R/W	R/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]	CCPH	Selection of the polarity of higher leg of bridge output 0: FOC_OCH active high 0: FOC_OCH active low
[6]	CCPL	Selection of the polarity of lower leg of bridge output 0: FOC_OCL active high 0: FOC_OCL active low
[5]	CC3EH	Upper channel-3 output enable 0: Disable 1: Enable
[4]	CC3EL	Lower channel-3 output enable 0: Disable 1: Enable
[3]	CC2EH	Upper channel-2 output enable 0: Disable 1: Enable
[2]	CC2EL	Lower channel-2 output enable 0: Disable 1: Enable
[1]	CC1EH	Upper channel-1 output enable 0: disable 1: enable
[0]	CC1EL	Lower channel-1 output enable 0: disable 1: enable

13.2.12 FOC_DKP (0x4098 , 0x4099)

Table 13-15 FOC_DKPH（0x4098）

Bit	7	6	5	4	3	2	1	0
Name	FOC_DKP[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 13-16 FOC_DKPL（0x4099）

Bit	7	6	5	4	3	2	1	0
Name	FOC_DKP[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_DKP	Proportional coefficient of D axis PI controller Range: (0,32767)

13.2.13 FOC_DKI (0x409A , 0x409B)

Table 13-17 FOC_DKIH（0x409A）

Bit	7	6	5	4	3	2	1	0
Name	FOC_DKI[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 13-18 FOC_DKIL（0x409B）

Bit	7	6	5	4	3	2	1	0
Name	FOC_DKI[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_DKI	Integral coefficient of D axis PI controller Range: (0,32767)

13.2.14 FOC_DMAX (0x409C , 0x409D)

Table 13-19 FOC_DMAXH（0x409C）

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 13-20 FOC_DMAXL（0x409D）

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 UD Range: (-32768,32767)

13.2.15 FOC_DMIN (0x409E , 0x409F)

Table 13-21 FOC_DMINH（0x409E）

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 13-22 FOC_DMINL（0x409F）

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 UD Range: (-32768,32767)

13.2.16 FOC_QKP (0x40A0 , 0x40A1)

Table 13-23 FOC_QKPH（0x40A0）

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

Table 13-24 FOC_QKPL（0x40A1）

Bit	7	6	5	4	3	2	1	0
Name	FOC_QKP[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_QKP	Proportional coefficient of Q axis PI controller Range: (0,32767)

13.2.17 FOC_QKI (0x40A2 , 0x40A3)

Table 13-25 FOC_QKIH（0x40A2）

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

Table 13-26 FOC_QKIL（0x40A3）

Bit	7	6	5	4	3	2	1	0
Name	FOC_QKI[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_QKI	Integral coefficient of Q axis PI controller Range: (0,32767)

13.2.18 FOC_QMAX (0x40A4 , 0x40A5)

Table 13-27 FOC_QMAXH（0x40A4）

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 13-28 FOC_QMAXL（0x40A5）

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 UQ Range: (-32768,32767)

13.2.19 FOC_QMIN (0x40A6 , 0x40A7)

Table 13-29 FOC_QMINH（0x40A6）

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 13-30 FOC_QMINL（0x40A7）

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 UQ Range: (-32768,32767)

13.2.20 FOC_UD (0x40A8 , 0x40A9)

Table 13-31 FOC_UDH（0x40A8）

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

Table 13-32 FOC_UDL（0x40A9）

Bit	7	6	5	4	3	2	1	0
Name	FOC_UD[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_UD	The output voltage of D axis PI controller Range: (-32768,32767)

13.2.21 FOC_UQ (0x40AA , 0x40AB)

Table 13-33 FOC_UQH（0x40AA）

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

Table 13-34 FOC_UQL（0x40AB）

Bit	7	6	5	4	3	2	1	0
Name	FOC_UQ[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_UQ	The output voltage of Q axis PI controller Range: (-32768,32767)

13.2.22 FOC_IDREF (0x40AC , 0x40AD)

Table 13-35 FOC_IDREFH（0x40AC）

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 13-36 FOC_IDREFL（0x40AD）

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	The reference of d-axis current Range: (-32768,32767)

13.2.23 FOC_IQREF (0x40AE , 0x40AF)

Table 13-37 FOC_IQREFH（0x40AE）

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 13-38 FOC_IQREFL（0x40AF）

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)

13.2.24 FOC_ARR (0x40B0 , 0x40B1)

Table 13-39 FOC_ARRH（0x40B0）

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

Table 13-40 FOC_ARRL（0x40B1）

Bit	7	6	5	4	3	2	1	0
Name	FOC_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
[15:0]	FOC_ARR	Counter auto reload value Range: (0,65535)

13.2.25 FOC_COMR (0x40E6 , 0x40E7)

Table 13-41 FOC_COMRH（0x40E6）

Bit	7	6	5	4	3	2	1	0
Name	FOC_COMR[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 FOC_COMRL（0x40E7）

Bit	7	6	5	4	3	2	1	0
Name	FOC_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
[15:0]	FOC_COMR	Counter compare reference value. When counter value is equal this value, match event occurs. Range: (0,65535)

13.2.26 FOC_SWDUTY (0x40B2 , 0x40B3)

Table 13-43 FOC_SWDUTYH（0x40B2）

Bit	7	6	5	4	3	2	1	0
Name	FOC_SWDUTY[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 13-44 FOC_SWDUTYL（0x40B3）

Bit	7	6	5	4	3	2	1	0
Name	FOC_SWDUTY[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_SWDUTY	PWM duty set by software Range: (0,65535)

13.2.27 FOC_TSMIN (0x40B4 , 0x40B5)

Table 13-45 FOC_TSMINH（0x40B4）

Bit	7	6	5	4	3	2	1	0
Name	FOC_TSMIN[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 13-46 FOC_TSMINL（0x40B5）

Bit	7	6	5	4	3	2	1	0
Name	FOC_TSMIN[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_TSMIN	The minimum window for ADC sampling under single resistor sampling mode Range: (0,65535)
TS = sample window (△T) + deadtime (DT) E.g.: If △T=4us, DT=2us, TS=6us, and Carrier period=62.5us, TS=6/62.5*65536=3146		

13.2.28 FOC_TRGDLY (0x40B6 , 0x40B7)

Table 13-47 FOC_TRGDLYH（0x40B6）

Bit	7	6	5	4	3	2	1	0
Name	FOC_TRGDLY[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 13-48 FOC_TRGDLYL（0x40B7）

Bit	7	6	5	4	3	2	1	0
Name	FOC_TRGDLY[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_TRGDLY	Single resistor samping mode: current sampling triggering is delayed. Range: (-32768,32767) Dual resistor samping mode: current sampling timing TRGDLY[15]=0: counter up counting direction TRGDLY[15]=1: counter down couting direction Range: (0, FOC_ARR[14:0])
Single resistor sampling mode: MCU clock =24MHz(41.67ns), TRGDLY = 5, delay 41.67*5=208ns; TRGDLY = -5, advance 41.67*5=208ns Dual resistor samping mode: MCU clock =24MHz(41.67ns), TRGDLY = 0x8032, sample current in 41.67*50=2.08us before counter minima; TRGDLY = 0x0032, sample current in 41.67*50=2.08us after counter minima		

13.2.29 FOC_THETA (0x40B8 , 0x40B9)

Table 13-49 FOC_THETAH（0x40B8）

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 13-50 FOC_THETAL（0x40B9）

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 force pull angle Software read: FOC module angle Range: (-32768,32767)
Range: (-32768,32767), mapping (-180°,180°) If THETA=8192, the angle=8192/32768*180°= 45°		

13.2.30 FOC_THECOMP (0x40BA , 0x40BB)

Table 13-51 FOC_THECOMPH（0x40BA）

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 13-52 FOC_THECOMPL（0x40BB）

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

13.2.31 FOC_RTHECNT (0x408E)

Table 13-53 FOC_RTHECNT（0x408E）

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	Ramp times = RTHECNT*256 When ramp function is enabled (RFAE is set to 1 in FOC_CR1 register), the ramp function increases by RTHECNT*256 times.

13.2.32 FOC_RTHESTEP (0x40BC , 0x40BD)

Table 13-54 FOC_RTHESTEPH（0x40BC）

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 13-55 FOC_RTHESTEPL（0x40BD）

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
[Field15:0]	FOC_RTHESTEP	Ramp step Software write: Initial step value Software read: Current step value Range: (-32768,32767)
RTHESTEP(32bit)= RTHESTEP(32bit) + RTHEACC (32bit, the high 16bit are zero, and low 16bit are configurable); THETA(16bit) = THETA (16bit) + RTHESTEP (high16bit)		

13.2.33 FOC_RTHEACC (0x40BE , 0x40BF)

Table 13-56 FOC_RTHEACCH（0x40BE）

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 13-57 FOC_RTHEACCL（0x40BF）

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	Ramp acceleration, and the format is 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 high 16bit are zero, and low 16bit are configurable); THETA (16bit) = THETA (16bit) + RTHESTEP (high 16bit)		

13.2.34 FOC_THECOR (0x40C0 , 0x40C1)

Table 13-58 FOC_THECORH（0x40C0）

Bit	7	6	5	4	3	2	1	0
Name	FOC_THECOR[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 13-59 FOC_THECORL（0x40C1）

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
[15:0]	FOC_THECOR	Angle correction When the ramp ends, angle mode switches to estimation mode from ramp mode. There is a deviation between estimation angle and ramp angle, and the angle compensation is necessary. Range: (0,32767), MSB is always 0

13.2.35 FOC_ETHETA (0x40C2 , 0x40C3)

Table 13-60 FOC_ETHETAH（0x40C2）

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

Table 13-61 FOC_ETHETAL（0x40C3）

Bit	7	6	5	4	3	2	1	0
Name	FOC_ETHETA[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_ETHETA	Estimation angle. The format is same as FOC_THETA. Range: (-32768,32767)

13.2.36 FOC_KSLIDE (0x40C4 , 0x40C5)

Table 13-62 FOC_KSLIDEH（0x40C4）

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 13-63 FOC_KSLIDEL（0x40C5）

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 PLL mode (ESEL=1): KP of PLL PI controller Range: (0,32767), MSB is always 0

13.2.37 FOC_EKLPFMIN (0x40C6 , 0x40C7)

Table 13-64 FOC_EKLPFMINH（0x40C6）

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	0	0	0	0	0	0	0	0

Table 13-65 FOC_EKLPFMINL（0x40C7）

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

Bit	Name	Function
[15:0]	FOC_EKLPFMIN / FOC_PLLKI	SMO mode(ESEL=0): the minimum LPF coefficient of estimator PLL mode(ESEL=1): KI of PLL PI controller Range: (0,32767), MSB is always 0

13.2.38 FOC_EBMFK (0x40C8 , 0x40C9)

Table 13-66 FOC_EBMFKH（0x40C8）

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 13-67 FOC_EBMFKL（0x40C9）

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	The coefficient of EKLPF Range: (-32768,32767)
EKLPF = EBMFK * OMEGA EBMFK = 2 * pi * fbase * ΔT		

13.2.39 FOC_OMEKLPF (0x40CA , 0x40CB)

Table 13-68 FOC_OMEKLPFH（0x40CA）

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 13-69 FOC_OMEKLPFL（0x40CB）

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 coefficient for speed calculation Range: (0,32767), MSB is always 0

13.2.40 FOC_FBASE (0x40CC , 0x40CD)

Table 13-70 FOC_FBASEH（0x40CC）

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 13-71 FOC_FBASEL（0x40CD）

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
FBASE = fbase * ΔT * 65536 If fbase = 200HZ , Δ T = 62.5us，and FBASE = 819		

13.2.41 FOC_EOME (0x40CE , 0x40CF)

Table 13-72 FOC_EOMEH（0x40CE）

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

Table 13-73 FOC_EOMEL（0x40CF）

Bit	7	6	5	4	3	2	1	0
Name	FOC_EOME[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_EOME	The estimated speed Range: (-32768,32767)

13.2.42 FOC_EKP (0x40D0 , 0x40D1)

Table 13-74 FOC_EKPH（0x40D0）

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 13-75 FOC_EKPL（0x40D1）

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	Proportional coefficient of estimator PI controller Range: (0,32767)

13.2.43 FOC_EKI (0x40D2 , 0x40D3)

Table 13-76 FOC_EKIH（0x40D2）

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 13-77 FOC_EKIL（0x40D3）

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	integral coefficient of estimator PI controller Range: (0,32767)

13.2.44 FOC_POWKLPF (0x40D4 , 0x40D5)

Table 13-78 FOC_POWKLPFH（0x40D4）

Bit	7	6	5	4	3	2	1	0
Name	FOC_POWKLPF[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 13-79 FOC_POWKLPFL（0x40D5）

Bit	7	6	5	4	3	2	1	0
Name	FOC_POWKLPF[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_POWKLPF	The LPF coefficient of power calculated Range: (0,32767), MSB is always 0

13.2.45 FOC_POW (0x40D6 , 0x40D7)

Table 13-80 FOC_POWH（0x40D6）

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 13-81 FOC_POWL（0x40D7）

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)

13.2.46 FOC_EK1 (0x40D8 , 0x40D9)

Table 13-82 FOC_EK1H（0x40D8）

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 13-83 FOC_EK1L（0x40D9）

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 coefficient of the estimated current Range: (0,32767), MSB is always 0

13.2.47 FOC_EK2 (0x40DA , 0x40DB)

Table 13-84 FOC_EK2H（0x40DA）

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 13-85 FOC_EK2L（0x40DB）

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 coefficient of the estimated current Range: (0,32767), MSB is always 0

13.2.48 FOC_EK3 (0x40DC , 0x40DD)

Table 13-86 FOC_EK3H（0x40DC）

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 13-87 FOC_EK3L（0x40DD）

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 coefficient of the estimated current Range: (0,32767), MSB is always 0

13.2.49 FOC_EK4 (0x40DE , 0x40DF)

Table 13-88 FOC_EK4H（0x40DE）

Bit	7	6	5	4	3	2	1	0
Name	FOC_EK4[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 13-89 FOC_EK4L（0x40DF）

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 coefficient of the estimated current Range: (0,32767), MSB is always 0

13.2.50 FOC_IA (0x40E0 , 0x40E1)

Table 13-90 FOC_IAH（0x40E0）

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 13-91 FOC_IAL（0x40E1）

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)

13.2.51 FOC_IB (0x40E2 , 0x40E3)

Table 13-92 FOC_IBH（0x40E2）

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 13-93 FOC_IBL（0x40E3）

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)

13.2.52 FOC_IBET (0x40E4 , 0x40E5)

Table 13-94 FOC_IBETH（0x40E4）

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 13-95 FOC_IBETL（0x40E5）

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 tranfromation Range: (-32768,32767)

13.2.53 FOC_ID (0x40E8 , 0x40E9)

Table 13-96 FOC_IDH（0x40E8）

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 13-97 FOC_IDL（0x40E9）

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 tranfromation Range: (-32768,32767)

13.2.54 FOC_IQ (0x40EA , 0x40EB)

Table 13-98 FOC_IQH（0x40EA）

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 13-99 FOC_IQL（0x40EB）

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 tranfromation Range: (-32768,32767)

13.2.55 FOC_VALP (0x40EC , 0x40ED)

Table 13-100 FOC_VALPH（0x40EC）

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 13-101 FOC_VALPL（0x40ED）

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	The voltage VALPHA from IPARK tranfromation Range: (-32768,32767)

13.2.56 FOC_VBET (0x40EE , 0x40EF)

Table 13-102 FOC_VBETH（0x40EE）

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 13-103 FOC_VBETL（0x40EF）

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	The VBETA from IPARK tranfromation Range: (-32768,32767)

13.2.57 FOC_UALP (0x40F0 , 0x40F1)

Table 13-104 FOC_UALPH（0x40F0）

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

Table 13-105 FOC_UALPL（0x40F1）

Bit	7	6	5	4	3	2	1	0
Name	FOC_UALP[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_UALP	The voltage UALPHA from voltage calculation module Range: (-32768,32767)

13.2.58 FOC_UBET (0x40F2 , 0x40F3)

Table 13-106 FOC_UBETH（0x40F2）

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

Table 13-107 FOC_UBETL（0x40F3）

Bit	7	6	5	4	3	2	1	0
Name	FOC_UBET[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_UBET	The voltage UBETA from voltage calculation module Range: (-32768,32767)

13.2.59 FOC_EALP (0x40F4 , 0x40F5)

Table 13-108 FOC_EALPH（0x40F4）

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 13-109 FOC_EALPL（0x40F5）

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 alpha axis estimated back-EMF EALPHA Range: (-32768,32767)

13.2.60 FOC_EBET (0x40F6 , 0x40F7)

Table 13-110 FOC_EBETH（0x40F6）

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 13-111 FOC_EBETL（0x40F7）

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 beta axis estimated back-EMF EBETA Range: (-32768,32767)

13.2.61 FOC_ESQU (0x40F8 , 0x40F9)

Table 13-112 FOC_ESQUH（0x40F8）

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

Table 13-113 FOC_ESQUL（0x40F9）

Bit	7	6	5	4	3	2	1	0
Name	FOC_ESQU[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_ESQU	EBETA* EBETA +EALPHA* EALPHA Range: (0,65535)

13.2.62 FOC_UDCFLT (0x40FA，0x40FB)

Table 13-114 FOC_UDCFLTH（0x40FA）

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 13-115 FOC_UDCFLTL（0x40FB）

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 UDC Range: (0,32767)
If BUS voltage is stepped down 6:1 into ADC, and the voltage range of ADC is 0 ~ 5V, the BUS voltage is thus 0~30V; If FOC_UDCFLT is 19661（0x4CCD）, the BUS voltage is 19661/32768*5*6 = 18V. Note: The normal input voltage sent to ADC shouldn't be higher than 3.75V. The higher voltage could induce the overflow in FOC calculation. Voltage higher than 3.75V can be used to trigger overvoltage protection.		

13.2.63 FOC_CSO (0x40FC，0x40FD)

Table 13-116 FOC_CSOH（0x40FC）

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 13-117 FOC_CSOL（0x40FD）

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 sample offset Set CSOC in the FOC_CR1 register, write the FOC_CSO register to calibrate the current sample offset, and the reference of the dual resistor sampling. Range: (0,32767), MSB is always 0
If the range of ADC is 0~5V，and the reference is 2.5V, FOC_CSO = 2.5/5V*32768 = 16384(0x4000)		

13.2.64 FOC_EFREQACC (0x4088 , 0x4089)

Table 13-118 FOC_EFREQACCH（0x4088）

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 13-119 FOC_EFREQACCL（0x4089）

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 force angle mode from the estimator Range: (0,65535)
If fbase = 200HZ and motor pole-pair pp=4，speed_base=60*fbase/pp=3000rpm. Set OMEGA increment be 3rpm, FOC_EFREQACC = 3/speed_base*32768*256 = 8388(0x20c4)		

13.2.65 FOC_EFREQMIN (0x408A , 0x408B)

Table 13-120 FOC_EFREQMINH（0x408A）

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 13-121 FOC_EFREQMINL（0x408B）

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 force angle mode When the estimated speed is smaller then FOC_EFREQMIN, the estimator force angle mode is proper operation Range: (-32768,32767)
If fbase = 200HZ and pole-pair pp=4, speed_base=60*fbase/pp=3000rpm; Set the minimum OMEGA to be 30rpm, FOC_ EFREQMIN = 30/speed_base*32768 = 327(0x147)		

13.2.66 FOC_EFREQHOLD (0x408C , 0x408D)

Table 13-122 FOC_EFREQHOLDH（0x408C）

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 13-123 FOC_EFREQHOLDL（0x408D）

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 force angle mode In the estimator force angle mode, the force speed increases from 0 to FOC_EFREQHOLD Range: (-32768,32767)
If fbase = 200HZ，and motor pole-pair pp=4, speed_base=60*fbase/pp=3000rpm; Set the holding speed of OMEGA be 60rpm, FOC_ EFREQHOLD = 60/speed_base*32768 = 655(0x28f)		

13.2.67 FOC_DTR (0x4064)

Table 13-124 FOC_DTR（0x4064）

Bit	7	6	5	4	3	2	1	0
Name	FOC_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]	FOC_DTR	Deadtime This register defines the duration of the deadtime inserted between the complementary outputs. DT corresponds to this duration. When MCU clock pulse is 24 MHZ(41.67ns) DT= (DTR+1) x 41.67ns NOTE： DTR=0，no deadtime insertion

14 TIMER0 (TIM0)

14.1 Intruduction

Timer0 consists of a 16 bit up-down auto-reload counter, and its counter source is the internal clock. Timer0 includes the following feature:

1、 16-bit up, down, up/down counter auto-reload counter

2、 Repetition counter to update the timer registers only after a given number of cycles of the counter.

3、 8-bit programmable prescaler allowing the counter clock frequency to be divided “on the fly” by any factor between 1 and 255.

4、 4 independent channels that can be configured as:：

a) Output compare

b) PWM generation (edge and center-aligned mode)

c) 6-step PWM generation

d) One-pulse mode output

e) Complementary outputs on three channels with programmable deadtime insertion.

5、 Generation of SVPWM/SPWM with SVPWM/SPWM module

6、 Interrupt generation on the following events:

a) Update: Counter overflow/underflow, counter initialization

b) Trigger event COM

c) Output compare

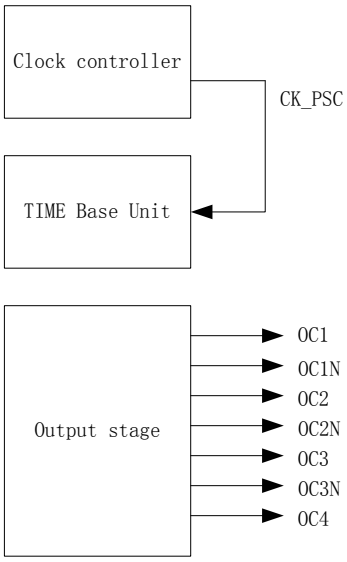


Figure 14-1 TIM0 Diagra

14.1.1 Timer0 Counter

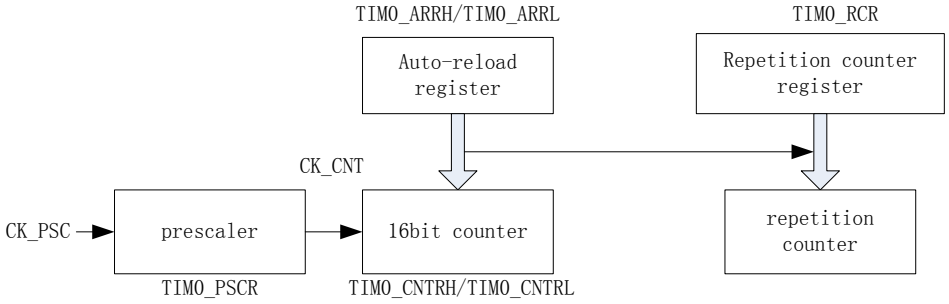


Figure 14-2 Time base unit

The kernel of Timer0 is a 16bit counter (TIM0_CNTR), and the prescaler (TIM0_PSCR) is used to generate a divided clock (fCK_CNT=fCK_PSC/(TIM0_PSCR+1)，and the auto-reload register is used to store the value of counter reload. The repetition counter to update the timer registers only after a given number of cycles of the counter. All above four groups of registers can be read or written by using software.

14.1.1.1 Read Write Sequency for TIM0_ARR

TIM0_ARR includes preload register and shadow register. Two modes are provided.

1、 TIM0_ARR enable preload (TIM0_CR1[ARPE]=1)：In this mode, when data is written to the auto-reload register, it is kept in the preload register and transferred into the shadow register at the next update event (UEV).

2、 TIM0_ARR disable preload (TIM0_CR1[ARPE]=0)：In this mode, when data is written to the auto-reload register, it is transferred into the shadow register immediately.

16-bit values are loaded in the TIM0_ARR 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.

14.1.1.2 Read Write Sequency for TIM0_PSCR

The prescaler TIM0_PSCR value is loaded through a preload register. The shadow register, which contains the current value to be used, is loaded as soon as the LS byte has been written.

To update the 16-bit prescaler, load two bytes in separate write operations starting with the MSB. Do not use the LDW instruction for this purpose as it writes the LSB first.

The new prescaler value is taken into account in the following period (after the next counter update event).

14.1.1.3 TIM0_CNTR: Its writing and counting

There will be no buffering in writing to the counter. Both TIM0_CNTRH and TIM0_CNTRL can be written at any time. It is recommended not to write a new value into the counter while it is running

to avoid loading an incorrect intermediate content.

An 8-bit buffer is implemented for the read. Software must read the MS byte first, and then LS byte value is buffered automatically. This buffered value remains unchanged until the 16-bit read sequence is completed.

1. Up-counting mode

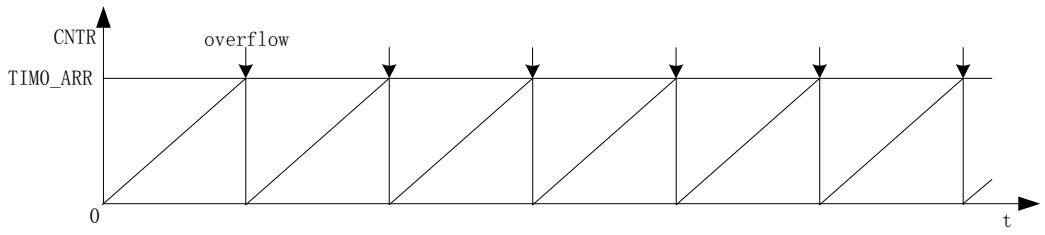


Figure 14-3 Counter in up-counting mode

Set TIM0_CR0[CMS]=00 and TIM0_CR1[DIR]=0 to set the counter in up-counting mode. In this mode, the counter counts from 0 to a user-defined compare value (content ofthe TIM0_ARR register). It then restarts from 0 and generates a counter overflow event. It, however, generates an UEV if TIM0_CR1[UDIS]=0.

2. Down-counting mode

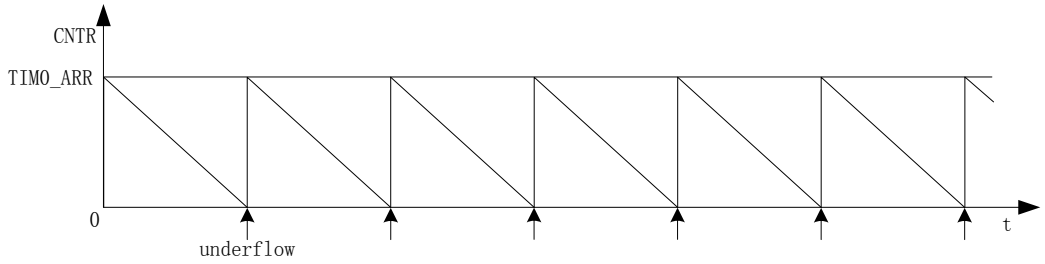


Figure 14-4 Counter in down-counting mode

Set TIM0_CR0[CMS]=00 and TIM0_CR1[DIR]=0 to set the counter in down-counting mode. In this mode, the counter counts from the auto-reload value (content of theTIM0_ARR register) down to 0. It then restarts from the auto-reload value and generates a counter underflow. It, however, generates an UEV, if TIM0_CR1[UDIS]=0.

3. Center-aligned mode (up/down counting)

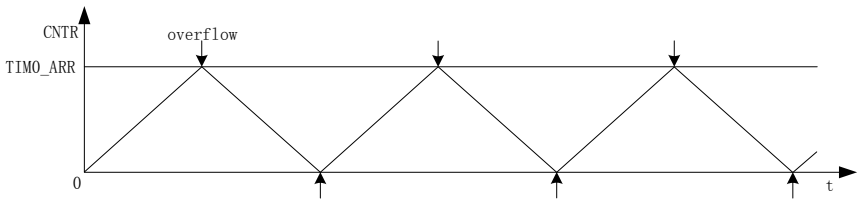


Figure 14-5 Counter in center-aligned mode

Set TIM0_CR0[CMS]=01/10/11 to set the counter in center-aligned mode. In this mode, the counter counts from 0 to the auto-reload value of -1 (content ofthe TIM0_ARR register). This generates a counter overflow event. The counter then counts down to 0 and generates a counter

underflow event. After this, the counter restarts counting from 0.

In this mode, the direction bit (DIR) in the TIM0_CR1 register cannot be written. It is updated by hardware and gives the current direction of the counter.

If the timer has a repetition counter (as in TIM1), the UEV is generated after up and downcounting and repeated for the number specified in repetition counter register (TIM0_RCR). Otherwise, the UEV is generated at each counter overflow and at each counter underflow.

When starting in center-aligned mode, the current up-down configuration is used. It means that the counter starts counting up or down depending on the value written in the DIR bit in the TIM1_CR1 register. The DIR and CMS bits must not be changed at the same time by using the software.

Writing to the counter while running in center-aligned mode is not recommended, as it could lead to unexpected results. The safest way to use center-aligned mode is to generate updated value by using software (setting the UG bit in the TIM1_EGR register) just before starting the counter. Writing to the counter should be avoided while it is running.

4. Repetition down-counter

The repetition down-counter is an auto-reload type, and its repetition rate is maintained as defined by the TIM0_RCR register value. This means that data are transferred from the preload registers to the shadow registers (TIM0_ARR auto-reload register, TIM0_PSCR prescaler register, and TIM0_CCRx capture/compare registers in compare-mode.

The repetition down-counter is decremented at each counter:

a) overflow in up-counting mode

b) underflow in down-counting mode

c) overflow and at each counter underflow in center-aligned mode.

When the UEV is generated by using software (by setting TIM0_EGR[UG]=1), or by using hardware (through theclock/trigger controller), it occurs immediately irrespective of the value of the repetition down-counter. The repetition down-counter is reloaded with the content of the TIM0_RCR register.

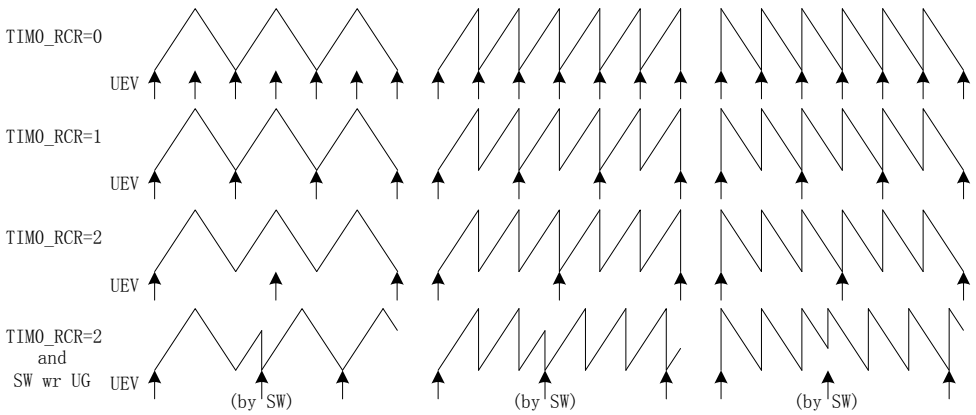


Figure 14-6 Examples of update rate depending on mode and TIM0_RCR register settings

14.1.2 TIM0 clock controller

The TIM0 prescaler is based on a 16-bit counter controlled through a 16-bit register (in TIM0_PSCR register). It can be changed on the fly as this control register is buffered. It can divide the counter clock frequency by any factor between 1 and 256.

The counter clock frequency is calculated as follows:

fCK_CNT=fCK_PSC/(TIM0_PSCR[7:0]+1)

The prescaler value is loaded through a preload register. The shadow register, which contains the current value to be used, is loaded as soon as the LS byte has been written. To update the 16-bit prescaler, load two bytes with separated write operations starting with the MSB.

The new prescaler value is taken into account in the following period (after the next counter update event).

Read operations to the TIM0_PSCR registers access the preload registers.

14.1.3 TIM0 compare channels

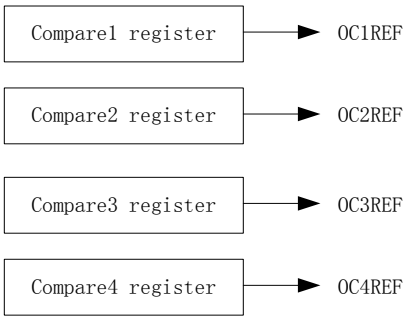


Figure 14-7 Channel output

TIM0 has 4 I/O pins can be configured for output comparing functions. Each Compare-channel is built around a compare-register (including a shadow register) and an output stage (with comparator and output control).

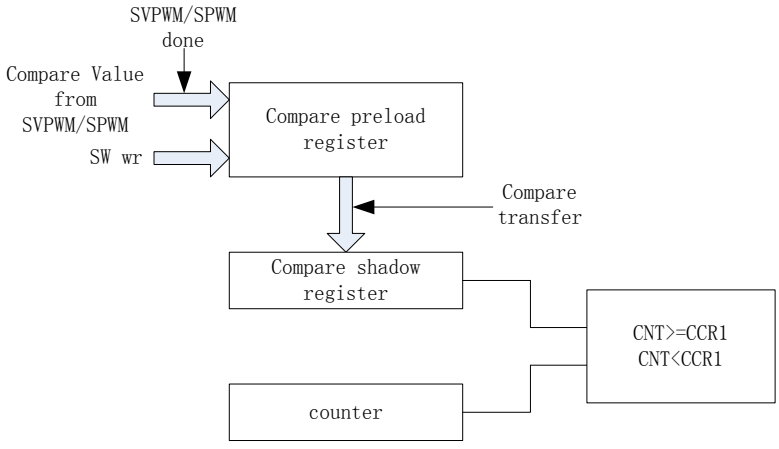


Figure 14-8 Block diagram of channel output stage

14.1.3.1 Compare register

The compare register (TIM0_CCRx) is made of one preload register and one shadow register. The content of the preload register can be obtained from SVPWM/SPWM module or software write. The content of the preload register is copied into the shadow register which is compared to the counter. When the content TIM0_CCRx is equal to the counter, the compare flag TIM0_SR1[CCxIF] is set. If TIM0_IER[CCxE]=1, compare interrupt is generated.

1. TIM_CCRx is updated by SVPWM/SPWM module: When the SVPWM/SPWM module is enabled, the results will be automatically stored in the preload register after the module has completed the operation.
2. Read for TIM_CCRx: Read operations access the preload register.
3. Write sequence for TIM_CCRx:16-bit values are loaded in the TIM1_CCRx registers 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, and stays blocked until the LS byte is written.

14.1.3.2 Output stage

The output stage generates an intermediate waveform called OCiREF (active high) , and then is used for reference.

1. Forced output mode

In output mode (TIM0_CCMRx[OCxM]=100 or 101), each output compare-signal can be forced to high or low level directly by software, which is independent to any comparison between the output compare-register and the counter.

Nevertheless, the comparison between the TIM0_CCRx shadow registers and the counter is still performed and allows the flag to be set. Interrupt requests can be sent accordingly. This will be described in the output compare mode section.

To force an output compare signal to its active level, write 101 in the OCxM bits in the corresponding TIM0_CCMRx registers. OCxREF is forced to be high (OCxREF is always active to high), and the OCx output is forced to be high or low depending on the CCxP polarity bits.

For example, if CCxP = 0 (OCx actives to high) => OCx is forced high.

The OCxREF signal can be forced to low by writing the OCxM bits to 100 in the TIM0_CCMRx registers.

2. Output compare mode

This function is used to control the output waveform or indicate when a period of time has elapsed.

When a match is found between the compare register and the counter:

a) Depending on the output compare mode, the corresponding OCxREF:
   (i) Remains unchanged (OCxM = 000),

(ii) Is set active (OCxM = 001),

(iii) Is set inactive (OCxM = 010)

(iv) Toggles (OCxM = 011)

b) A flag is set in the interrupt status register (CCxIF bits in the TIM0_SR1 register).

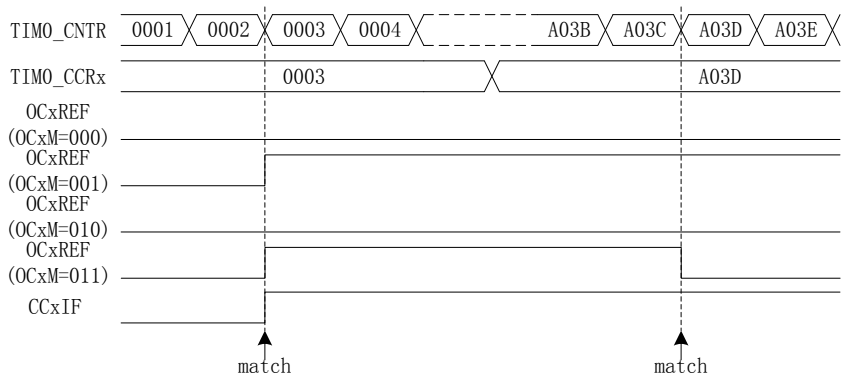


Figure 14-9 Output compare mode

14.1.3.3 PWM mode

Pulse width modulation mode allows user to generate a signal with a frequency determined by the value of the TIM0_ARR register and a duty cycle determined by the value of TIM0_CCRx registers. The PWM mode can be selected independently on each channel (one PWM per OCx output) by writing 110 (PWM mode 1) or 111 (PWM mode 2) in the OCxM bits in the TIM0_CCMRx registers. The corresponding preload register must be enabled by setting the OCxPE bits in the TIM0_CCMRx registers. The auto-reload preload register (in up-counting or center aligned modes) may be enabled by setting the ARPE bit in the TIM0_CR1 register.

As the preload registers are transferred to the shadow registers only when an UEV occurs, all registers have to be initialized by setting the UG bit in the TIM0_EGR register before starting the counter.

OCx polarity is software programmable by using the CCxP bits in the TIM0_CCERx registers. It can be programmed as active high or active low. The OCx output is enabled by the CCxE bits (TIM0_CCERx registers). Refer to the TIM0_CCERx register descriptions for more details.

In PWM mode (1 or 2), TIM0_CNTR and TIM0_CCRx are always compared to determine whether TIM0_CCRx ≤ TIM0_CNTR or TIM0_CNTR ≤ TIM0_CCRx (depending on the direction of the counter).

The timer is able to generate PWM in edge-aligned mode or center-aligned mode depending on the CMS bits in the TIM0_CR1 register.

14.1.3.4 PWM edge-aligned mode

a) Up-counting configuration

Up-counting is selected when the DIR bit in the TIM0_CR1 register is low.

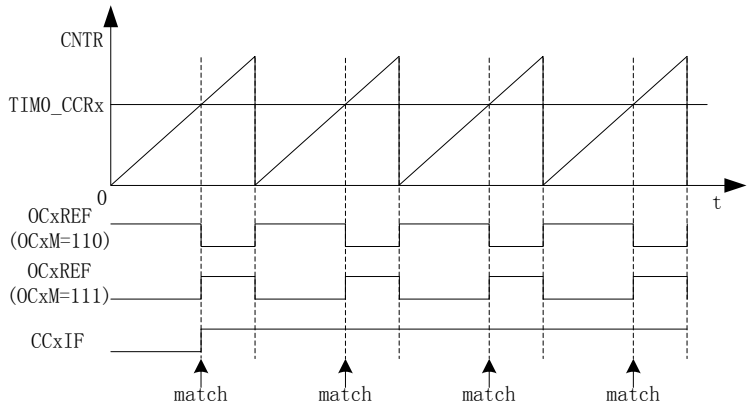


Figure 14-10 Edge-aligned counting mode PWM mode waveforms

b) Down-counting configuration

Down-counting is selected when the DIR bit in the TIM1_CR1 register is high.

14.1.3.5 PWM center-aligned mode

Center-aligned mode is selected when the CMS bits in the TIM0_CR1 register are different from 00 (all the remaining configurations have the same effect on the OCxREF signals).

The compare flag is set when the counter counts up, down, or up and down depending on the CMS bits configuration. The direction bit (DIR) in the TIM0_CR1 register is updated by hardware and is read-only in this mode.

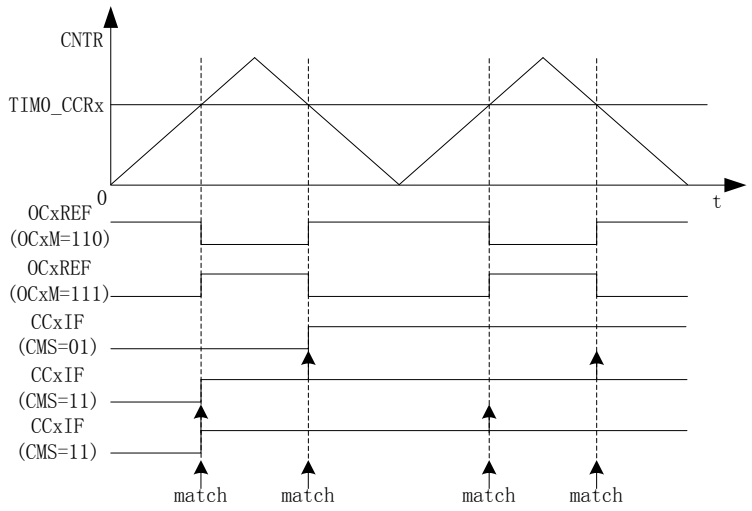


Figure 14-11 The waveform of PWM center-aligned mode

14.1.4 Complementary outputs and outpout of channel-4

TIM1 can output two complementary signals per channel (TIM0_OC1/TIM0_OC1N, TIM0_OC2/TIM0_OC2N, TIM0_OC3/TIM0_OC3N). It also controls the switching-off and

switching-on instants of the outputs. Channel-4 is an independent channel, and no deadtime insertion or zero deadtime. Deadtimes must be adjusted depending on the characteristics of the devices connected to the outputs (for example, intrinsic delays of level shifters, or delays, due to power switches).

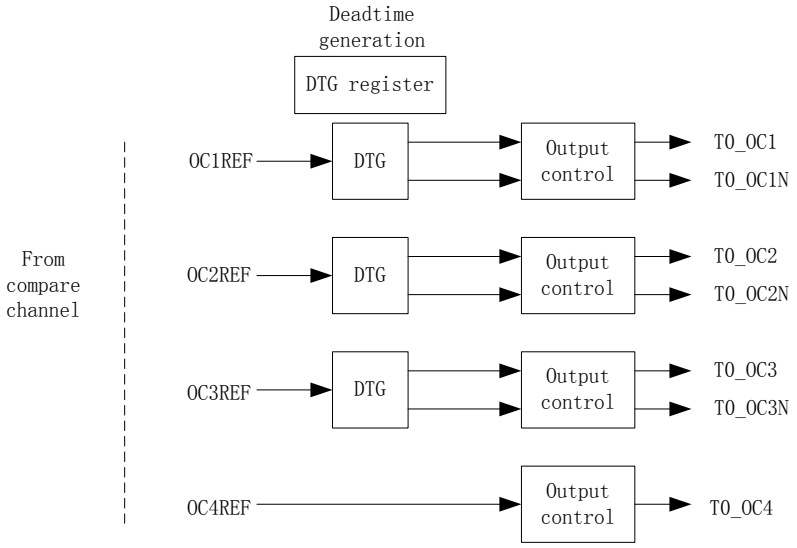


Figure 14-12 channels output block diagram

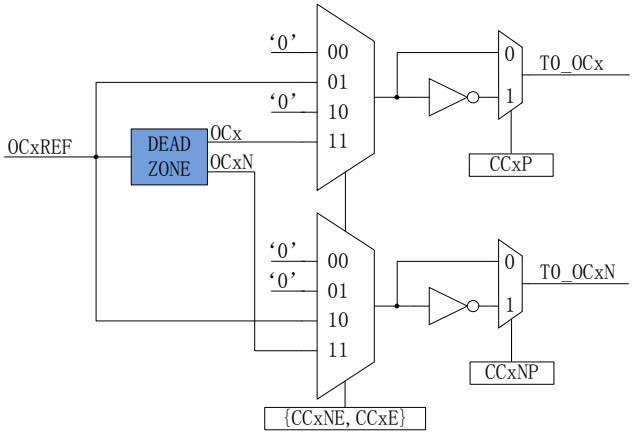


Figure 14-13 complementary output block diagram

The polarity of the outputs can be selected (main output OCx, or complementary OCxN) independently for each output. This is done by writing to the CCxP and CCxNP bits in the TIM0_CCERx registers.

The complementary signals OCx and OCxN are activated by a combination of several control bits: The CCxE and CCxNE bits in the TIM0_CCERx register.

When complementary outputs are implemented on a channel, preload bits are available on OCxM, CCxE and CCxNE bits. The preload bits are transferred to the active bits at the commutation event (COM). This allows the configuration for the next step to be programmed in advance and for configuration of all the channels to be changed at the same time. The COM event can be generated with software by setting the COMG bit in the TIM0_EGRregister. A flag is set when the COM event occurs (COMIF bit in the TIM0_SR register) , which can generate an interrupt (if the

COMIE bit is set in the TIM0_IER register).

Deadtime insertion is enabled by setting the CCxE and CCxNE bits. Each channel embeds an 8-bit deadtime generator. It generates two outputs: OCx and OCxN from a reference waveform, OCxREF. If OCx and OCxN are active high:

a) The OCxoutput signal is the same as the reference signal except for the rising edge, which is delayed relative to the reference rising edge.

b) The OCxN output signal is the opposite of the reference signal except for the rising edge, which is delayed relative to the reference falling edge.

If the delay is greater than the width of the active output (OCxor OCxN), the corresponding pulse is not generated, and the complementary channel will not be delayed.

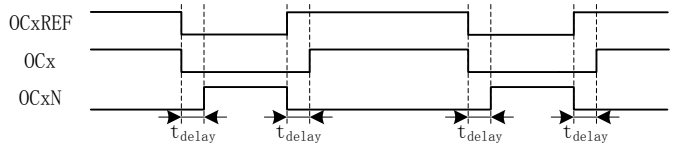


Figure 14-14 Complementary output with deadtime insertion

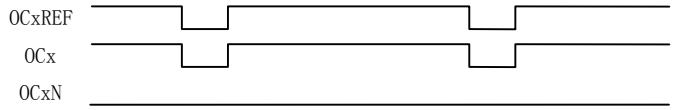


Figure 14-15 Deadtime waveforms with a delay greater than the negative pulse

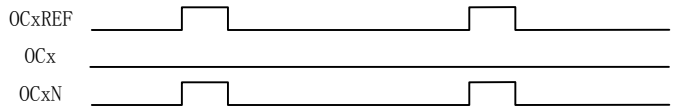


Figure 14-16 Deadtime waveforms with a delay greater than the positive pulse

14.1.5 Timer0 Interrupt

TIM0 has 6 interrupt request sources:

1. Commutation interrupt
2. Compare 4 interrupt
3. Compare 3 interrupt
4. Compare 2 interrupt
5. Compare 1 interrupt
6. Update interrupt (example: overflow, underflow, and counter initialization)

To use the interrupt features for each interrupt channel, set the desired interrupt enable bits in the

TIM0_IER register to enable interrupt requests.

The different interrupt sources can also be generated with software using the corresponding bits in the TIM0_EGR register.

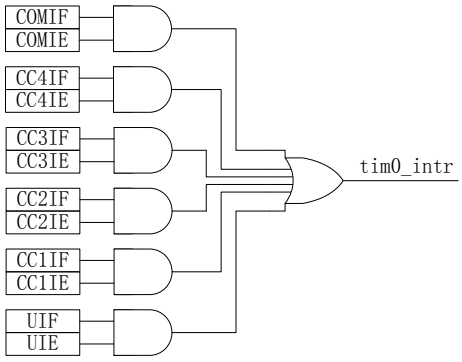


Figure 14-17 TIM0 interrupt request sources

14.2 TIM0 Registers

14.2.1 TIM0_CR (0x4058)

Table 14-1 TIM0_CR（0x4058）

Bit	7	6	5	4	3	2	1	0
Name	ARPE	CMS		DIR	OPM	URS	UDIS	T0CEN
Type	R/W	R/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]	ARPE	Auto-reload preload enable 0：TIM0_ARR TIM0_ARR register is not buffered through a preload register. It can be written directly 1：TIM0_ARR TIM0_ARR register is buffered through a preload register
[6:5]	CMS	Center-aligned mode selection 00: Edge-aligned mode. The counter counts up or down depending on the direction bit (DIR). 01: Center-aligned mode-1. The counter counts up and down. Output compare interrupt flags of channels configured in output (CCxS = 00 in TIM0_CCMRx registers) are triggered only when the counter is counting down. 10: Center-aligned mode-2. The counter counts up and down. Output compare interrupt flags of channels configured in output (CCxS = 00 in CCMRx registers) are triggered only when the counter is counting up. 11: Center-aligned mode-3. The counter counts up and down. Output compare interrupt flags of channels configured in output (CCxS = 00 in TIM0_CCMRx registers) are triggered whenever counter matches compare reference. Note: It is not allowed to switch from edge-aligned mode to center-aligned mode while the counter is enabled (CEN = 1)
[4]	DIR	Direction 0: Counter used as up-counter 1: Counter used as down-counter Note: This bit is read-only when the timer is configured in center-aligned mode.
[3]	OPM	One-pulse mode 0: Counter doesn’t stop at update event 1: Counter stops counting at the next update event (clearing the CEN bit)
[2]	URS	Update request source 0: When it is enabled by the UDIS bit, the UIF bit is set and an update interrupt request is sent when one of the following events occurs: – Registers are updated (counter overflow/underflow) – UG bit is set by software 1: When it is enabled by the UDIS bit, the UIF bit is set and an update interrupt request is sent only when registers are updated (counter overflow/underflow).
[1]	UDIS	Update disable 0: A UEV is generated as soon as a counter overflow occurs, or a software update is generated. Buffered registers are then loaded with their preloaded values. 1: A UEV is not generated and shadow registers keep their value (ARR, PSC, CCRx). The counterand the prescaler are re-initialized if the UG bit is set .
[0]	T0CEN	Counter enable 0: Counter disabled 1: Counter enabled

14.2.2 TIM0_IER (0xB9)

Table 14-2 TIM0_IER（0xB9）

Bit	7	6	5	4	3	2	1	0
Name	RSV		COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
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	R/W	Resetting
[7:6]	RSV	Reversed	R	0x0
[5]	COMIE	Commutation interrupt enable 0: Commutation interrupt disabled 1: Commutation interrupt enabled	RW	0x0
[4]	CC4IE	Compare channel-4 interrupt enable 0: CC4 interrupt disabled 1: CC4 interrupt enabled	RW	0x0
[3]	CC3IE	Compare channel-3 interrupt enable 0: CC3 interrupt disabled 1: CC3 interrupt enabled	RW	0x0
[2]	CC2IE	Compare channel-2 interrupt enable 0: CC2 interrupt disabled 1: CC2 interrupt enabled	RW	0x0
[1]	CC1IE	Compare channel-1 interrupt enable 0: CC1 interrupt disabled 1: CC1 interrupt enabled	RW	0x0
[0]	UIE	Update interrupt enable 0: Update interrupt disabled 1: Update interrupt enabled	RW	0x0

14.2.3 TIM0_SR (0xB1)

Table 14-3 TIM0_SR（0xB1）

Bit	7	6	5	4	3	2	1	0
Name	RSV		COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
Type	R/W	R/W	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	Reversed
[5]	COMIF	Commutation interrupt flag This flag is set by hardware by a COM event (when compare control bits - CCxE, CCxNE, OCxM -have been updated). It is cleared by software. 0: No COM has occurred 1: COM interrupt pending
[4]	CC4IF	compare channel-4 interrupt flag Refer to CC1IF description
[3]	CC3IF	compare channel-3 interrupt flag Refer to CC1IF description
[2]	CC2IF	compare channel-2 interrupt flag Refer to CC1IF description
[1]	CC1IF	compare channel-1 interrupt flag This flag is set by hardware when the counter matches the compare-value, with some exception under center-aligned mode (refer to the CMS bits from TIM0_CR1 register description). It is cleared bysoftware. 0: No match 1: The content of the counter register TIM0_CNT matches the content of the TIM0_CCR1 register Note: When the contents of TIM0_CCRx are greater than the contents of TIM0_ARR, the CCxIF bit goes high on the counter overflow (in up-counting and up/down-counting modes) or underflow (in down-counting mode)
[0]	UIF	Update interrupt flag This bit is set by hardware on an update event. It is cleared by software. 0: No update has occurred 1: Update interrupt pending. This bit is set by hardware when the registers are updated: – At overflow or underflow if UDIS = 0 in the TIM0_CR1 register – When CNT is re-initialized by software using the UG bit in TIM0_EGR register, if URS = 0and UDIS = 0 in the TIM1_CR1 register.

14.2.4 TIM0_EGR (0x4059)

Table 14-4 TIM0_EGR（0x4059）

Bit	7	6	5	4	3	2	1	0
Name	RSV		COMG	CC4G	CC3G	CC2G	CC1G	UG
Type	R/W	R/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]	COMG	Compare control update generation This bit can be set by software and is automatically cleared by hardware. 0: No action 1: When the CCPC bit in the TIM1_CR2 register is set, it allows the CCxE, CCxNE, CCxP, CCxNP, and OCxM bits to be updated. Note: This bit acts only on channels that have a complementary output.
[4]	CC4G	Compare channel-4 generation Refer to CC1G description.
[3]	CC3G	Compare channel-3 generation Refer to CC1G description.
[2]	CC2G	Compare channel-2 generation Refer to CC1G description.
[1]	CC1G	Compare channel-1 generation This bit is set by software to generate an event. It is automatically cleared by hardware. 0: No action 1: A compare event is generated on channel 1: The CC1IF flag is set and the corresponding interrupt request is sent if it is enabled.
[0]	UG	Update generation This bit can be set by software and is automatically cleared by hardware. 0: No action 1: Re-initializes the counter and generates an update of the registers. Note that the prescaler counter is also cleared. The counter is cleared if center-aligned mode is selected, or if DIR = 0 (upcounting). Otherwise, it takes the auto-reload value (TIM0_ARR) if DIR = 1 (down-counting).

14.2.5 TIM0_CCMR1 (0x405A)

Table 14-5 _CCMR1（0x405A）

Bit	7	6	5	4	3	2	1	0
Name	OC2M			OC2PE	OC1M			OC1PE
Type	R/W	R/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]	OC2M	Output compare-2 mode Refer to OC1M description.
[4]	OC2PE	Output compare 2 preload enable Refer to OC1PE description.
[3:1]	OC1M	Output compare 1 mode These bits define the behavior of the output reference signal, OC1REF, and also determine T0_OC1. OC1REF is active high, and OC1 active level depends on the CC1P bit. 000: Frozen. The comparison between the outputs of compare-register TIM0_CCR1 and counter-register TIM0_CNT has no effect on OC1REF. 001: Set channel-1 to active level on match. OC1REF signal is forced to high when the counter-register TIM0_CNT matches the capture/compare register-1 (TIM0_CCR1). 010: Set channel 1 to inactive level on match. OC1REF signal is forced to low when the counter-register TIM0_CNT matches the capture/compare register 1 (TIM0_CCR1). 011: Toggle. OC1REF toggles when TIM0_CNT = TIM0_CCR1 100: Force inactive level: OC1REF is forced to low 101: Force active level: OC1REF is forced to high 110: PWM mode-1. While up-counting, channel 1 is active when TIM0_CNT < TIM0_CCR1. Otherwise, the channel is inactive. In down-counting, channel-1 is inactive (OC1REF = 0) when TIM0_CNT > TIM0_CCR1. Otherwise, the channel is active (OC1REF = 1). 111: PWM mode-2. While up-counting, channel 1 is inactive as long as TIM0_CNT < TIM0_CCR1,otherwise, the channel is active. In down-counting, channel-1 is active when TIM0_CNT >TIM0_CCR1. Otherwise, the channel is inactive. Note: *1. In PWM mode 1 or 2, the OCxREF level changes only when the result of the comparison changes, or the output compare mode switches from "frozen" mode to "PWM" mode.* *2. On the channels having complementary output, the bitfield is preloaded. If the CCPC bit is set in the TIM0_CR2 register, the OCM active bits take the new value from the preload bits only when a COM is generated.*
[0]	OC1PE	Output compare-1 preload enable 0: Preload register on TIM0_CCR1 is disabled. TIM0_CCR1 can be written at anytime. The new value is taken into account immediately. 1: Preload register on TIM0_CCR1 is enabled. Read/write operations access the preload register. TIM0_CCR1 preload value is loaded in the shadow register at each UEV. Note: For correct operation, preload registers must be enabled when the timer is in PWM

14.2.6 TIM0_CCMR2 (0x405B)

Table 14-6 TIM0_CCMR2（0x405B）

Bit	7	6	5	4	3	2	1	0
Name	OC4M			OC4PE	OC3M			OC3PE
Type	R/W	R/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]	OC4M	Output compare-4 mode Refer to OC1M description.
[4]	OC4PE	Output compare -4 preload enable Refer to OC1PE description.
[3:1]	OC3M	Output compare-3 mode Refer to OC1M description.
[0]	OC3PE	Output compare-3 preload enable Refer to OC1PE description.

14.2.7 TIM0_CCER1 (0x405C)

Table 14-7 TIM0_CCER1（0x405C）

Bit	7	6	5	4	3	2	1	0
Name	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
Type	R/W	R/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]	CC2NP	Compare channel-2 complementary output polarity Refer to CC1NP description.
[6]	CC2NE	Compare channel 2 complementary output enable Refer to CC1NE description.
[5]	CC2P	Compare channel 2 output polarity Refer to CC1NP description.
[4]	CC2E	Compare channel 2 output enable Refer to CC1NE description.
[3]	CC1NP	Compare channel-1 complementary output polarity 0: OC1N active high 1: OC1N active low Note: On the channels having complementary output, this bit is preloaded. If the CCPC bit is set in the TIM0_CCER2 register, the CC1NP active bits take the new value from the preload bits.
[2]	CC1NE	Compare channel 1 complementary output enable 0: Off - OC1N is not active. 1: On - OC1N signal is active. Note: On the channels having complementary output, this bit is preloaded. If the CCPC bit is set inthe TIM0_CCER2 register. When both CC1E and CC1NE are set to 1，the outputs of T0_OC1 and T0_OC1N are inserted dead zone.
[1]	CC1P	Compare channel 1 output polarity 0: OC1 active high 1: OC1 active low Note: On the channels having complementary output, this bit is preloaded. If the CCPC bit is set inthe TIM0_CCER2 register, the CC1NP active bits take the new value from the preload bits.
[0]	CC1E	Compare channel-1 output enable 0: Off - OC1 is not active. 1: On - OC1 signal is active. Note: On the channels having complementary output, this bit is preloaded. If the CCPC bit is set in the TIM0_CCER2 register and COM event happens, the CC1E active bits take the new value from the preload bits. When both CC1E and CC1NE are set to 1，the outputs of T0_OC1 and T0_OC1N are inserted dead zone.

14.2.8 TIM0_CCER2 (0x405D)

Table 14-8 TIM0_CCER2（0x405D）

Bit	7	6	5	4	3	2	1	0
Name	RSV	CCPC	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E
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

Bit	Name	Function
[7]	RSV	Reversed
[6]	CCPC	Compare preloaded control 0: The CCxE, CCxNE, CCxP, and CCxNP bits in the TIM0_CCERx registers and the OCxM bit in theTIM0_CCMRx registers, are not preloaded 1: CCxE, CCxNE, CCxP, CCxNP and OCxM bits are preloaded, after the write, they ar eupdated only when COMG bit is set in the TIM0_EGR register. Note: This bit acts only on the channels with complementary outputs.
[5]	CC4P	compare channel 4 output polarity Refer to CC1NP description.
[4]	CC4E	Compare channel-4 output enable Refer to CC1E description.
[3]	CC3NP	Compare channel-3 complementary output polarity Refer to CC1NPdescription.
[2]	CC3NE	Compare channel-3 complementary output enable Refer to CC1E description.
[1]	CC3P	Compare channel-3 output polarity Refer to CC1NPdescription.
[0]	CC3E	Compare channel-3 output enable Refer to CC1E description.

14.2.9 TIM0_CNTR (0x405E，0x405F)

Table 14-9 TIM0_CNTRH（0x405E）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_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 14-10 TIM0_CNTRL（0x405F）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_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]	TIM0_CNTR	Counter value

14.2.10 TIM0_PSCR (0x4062)

Table 14-11 TIM0_PSCR（0x4062）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_PSCR							
Type	R/W	R/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]	TIM0_PSCR	Prescaler value The prescaler value divides the CK_PSC clock frequency. The counter clock frequency fCK_CNT is equal to fCK_PSC / (PSCR[7:0]+1). PSCR contains the value which is loaded in the active prescaler register at each UEV (when the counter is cleared through the UG bit of the TIM0_EGR register). A UEV must be generated so that a new prescaler value can be taken into account.

14.2.11 TIM0_ARR (0x4060 , 0x4061)

Table 14-12 TIM0_ARRH（0x4060）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_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 14-13 TIM0_ARRL（0x4061）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_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]	TIM0_ARR	Auto-reload value ARR is the value to be loaded in the actual auto-reload register. The counter is blocked while the auto-reload value is null.

14.2.12 TIM0_RCR (0x4063)

Table 14-14 TIM0_RCR（0x4063）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_RCR							
Type	R/W	R/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]	TIM0_RCR	Repetition counter value When the preload registers are enabled, these bits allow the user to set up the update rate of the compare-registers (periodic transfers from preload to shadow registers) as well as the updated interrupt generation rate if the update interrupt is enabled (UIE=1). When the REP_CNT related down-counter reaches zero, a UEV is generated and restarts counting from the REP value. As REP_CNT is reloaded with the REP value only at the repetition updated event U_RC, any write to the TIM0_RCR register will not taken into account until the next repetition update event. In PWM mode (REP+1) corresponds to: – The number of PWM periods in edge-aligned mode – The number of half PWM periods in center-aligned mode

14.2.13 TIM0_CCR1 (0xB6，0xB7)

Table 14-15 TIM0_CCR1H（0xB7）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_CCR1H							
Type	R/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 14-16 TIM0_CCR1L（0xB6）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_CCR1L							
Type	R/W	R/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]	TIM0_CCR1	Compare channel-1 value The value of CCR1 is loaded permanently into the actual compare-1 register if the preload feature is enabled (OC1PE bit in TIM0_CCMR1). Otherwise, the preload value is copied in the register when a UEV occurs. The active compare register contains the value which is compared to the counter register, TIM0_CNT, and signalled on the T0_OC1 output. When the SVPWM/SPWM module is enabled, the results will be automatically stored in the preload register after the module completes the operation.

14.2.14 TIM0_CCR2 (0xBA , 0xBB)

Table 14-17 TIM0_CCR2H（0xBB）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_CCR2H							
Type	R/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 TIM0_CCR2L（0xBA）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_CCR2L							
Type	R/W	R/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]	TIM0_CCR2	Compare channel 2 value Refer to TIM0_CCR1 description.

14.2.15 TIM0_CCR3 (0xBC , 0xBD)

Table 14-19 TIM0_CCR3H（0xBD）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_CCR3H							
Type	R/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 TIM0_CCR3L（0xBC）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_CCR3L							
Type	R/W	R/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]	TIM0_CCR3	Compare channel-3 value Refer to TIM0_CCR1 description.

14.2.16 TIM0_CCR4 (0xBE , 0xBF)

Table 14-21 TIM0_CCR4H（0xBF）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_CCR4H							
Type	R/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 TIM0_CCR4L（0xBE）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_CCR4L							
Type	R/W	R/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]	TIM0_CCR4	Compare channel-4 value Refer to TIM0_CCR1 description.

14.2.17 TIM0_DTR (0x4064)

Table 14-23 TIM0_DTR（0x4064）

Bit	7	6	5	4	3	2	1	0
Name	TIM0_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]	DTR[7:0]	Deadtime This bitfield defines the duration of the deadtime inserted between the complementary outputs. DT is the duration. When MCU clock pulse is 24 MHZ(41.67ns) DT= (DTR+1) x 41.67ns NOTE： DTR=0，no deadtime insertion

15 TIMER1 (TIM1)

15.1 Timer1 Operation

Timer1 consists of a 16-bit up basic counter and a 16-bit down reload counter. The counter clock of them come from the internel clock. Timer1 features include:

1. The 16-bit up basic counter used to record the basic timer which starts for position detection or the time of writing.
2. The 16-bit down reload counter used to record time: from position detection to the output of the reload counter time.
3. 4-bit programmable frequency divider for two timer count clock frequency division.
4. Input filtering and sampling
5. Position detection module according to the position detection signal input signal generation
6. Write output timing module for updating the status register
7. 6 channel can be configured to function as following:
    a) PWM generate
    b) multi-step PWM generate
    c) Single pulse mode output
    d) Support three groups of complementary in the dead zone.
8. Interrupt event generated
    a) Basic timer overflow interrupt
    b) Reload timer underflow interrupt
    c) Write timing interrupt
    d) Position detection interrupt
    e) Compare interrupt

Timer-1 mainly used to 6/12 steps of Motor control.It can output according to different needs of different modulation mode PWM waveform.

Timer-1 Internal structure as shown in figure 15-1

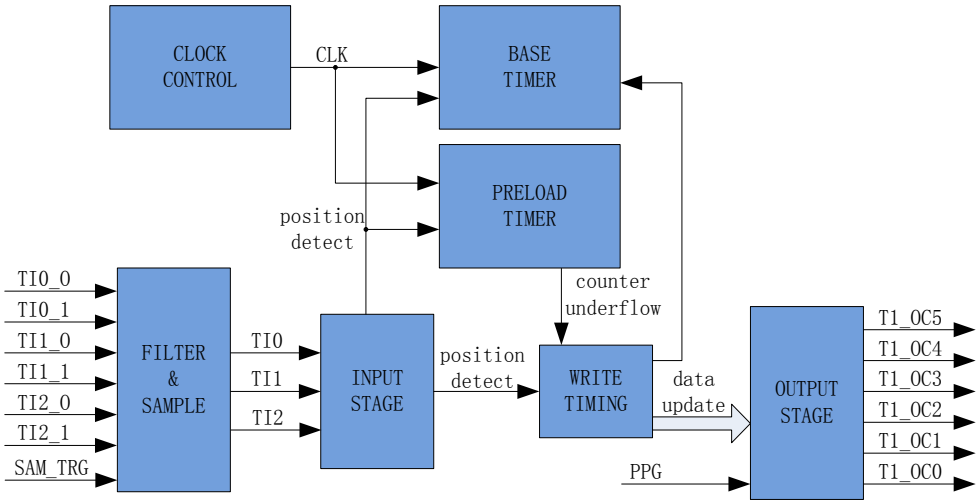


Figure 15-1 Timer1 Internal structure

15.1.1 Timer counter unit

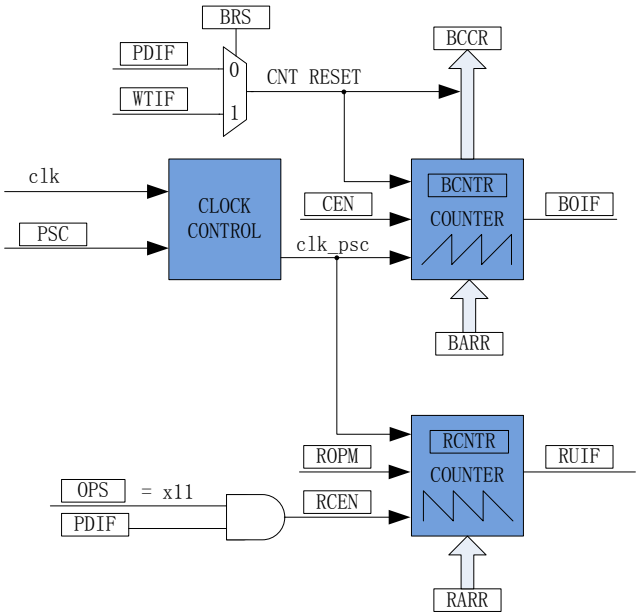


Figure 15-2 Timer1 base unit

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

15.1.1.1 Timer clock controller

Timer clock controller is used as the basic timer and reload timer count clock sources.The frequency divider count the clock divider. It is a 4-bit PSC controller with 12-bit counter, 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 frequently when they are in sleeping state.

The frequency of the counter can be calculated as the following:

If MCU clock is 24MHz(41.67ns), fCK_CNT=fCK_PSC/PSC

Table 15-1 The PSC values of the register corresponding to different clock frequency

PSC	Coffficient (16 bit)	CLK (Hz)	PSC	Coffficient (16 bit)	CLK (Hz)
0000	0x1	24M	1000	0x100	93.75K
0001	0x2	12M	1001	0x200	46.875K
0010	0x4	6M	1010	0x400	23.4375K
0011	0x8	3M	1011	0x600	15.625K
0100	0x10	1.5M	1100	0x800	11.71875K
0101	0x20	750K	1101	0xa00	9.375K
0110	0x40	375K	1110	0xc00	7.8125K
0111	0x80	187.5K	1111	0xe00	6.6964K

15.1.1.2 Basic Timer

The basic timer contains a 16-bit up counter. When the value of TIM1_BCNTR equals to the TIM1_BARR, overflow event generates.The Overflow interrupt flag BOIF of the basic counter is 1. At the same time, TIM1_BCNTR is 0 and counting is restarted.The register of TIM1_CR2 BRS chooses the counter reseting source from position detection events, or writting the sequential events. When the event reset signal is generated, the current count value TIM1_BCNTR is sent to register TIM1_BCCR, and TIM1_BCNTR value is sent to 0, and then the counting is restarted.

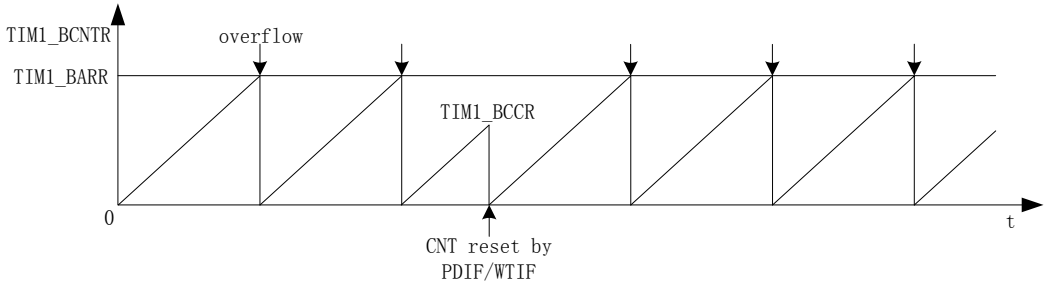


Figure 15-3 The waveform of the basic timer count

The value of register TIM1_BARR is sent to the counter immediately. So the register should be updated when the basic timer stops. Only when the counter of TIM1_BCNTR equals to TIM1_BARR, overflow event can generate. If TIM1_BCNTR>TIM1_BARR, TIM1_BCNTR will go to 0xffff, then restart count from 0. Therefore, the initial value of register TIM1_BCNTR must be smaller than TIM1_BARR.

15.1.1.3 Reload counter

The reload counter contains a 16-bit down counter. When the value of TIM1_RCNTR is 0, underflow event generates, and the underflow interrupt flag of reload counter is 1. At the same time, TIM1_RCNTR value reload the value of register TIM1_RARR. When the register of TIM1_CR1 ROPM is 1, the single mode is opened, but the reload counter starts only one time. When TIM1_RCNTR is 0, underflow event happens. TIM1_RCNTR reload the value of TIM1_RARR of the register. Then the register of TIM1_CR1, i.e. T1RCEN, is reset.

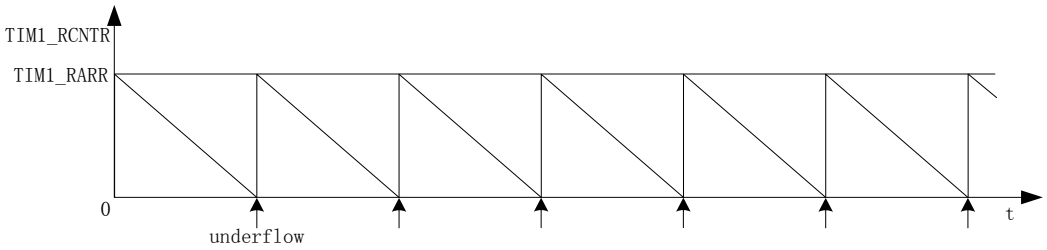


Figure 15-4 The waveform of reload timer count

The register of TIM1_CR1T1RCEN can be written, and configuring the register of TIM1_CR3 OPS is equal to 011/111, when the position detection event happens, T1RCEN will be set to 1.

15.1.2 Input filtering and sampling

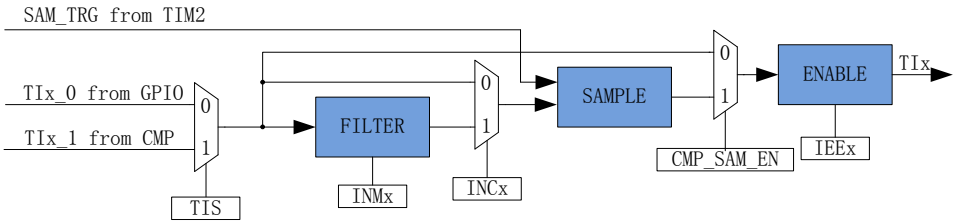


Figure 15-5 the principle diagram of the input signal filtering and sampling

The register of TIM1_CR2 TIS choose input source from the comparator or GPIO. The input noise filtering is optional. The register of TIM1_CR0IEE2/IEE1/IEE0=1 enable input detection. Otherwise The input of TI2/TI1/TI0 constant is invalid.

15.1.2.1 Filtering

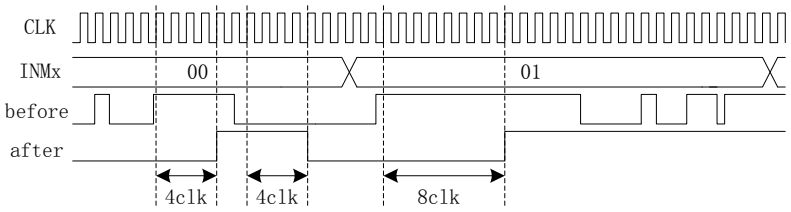


Figure 15-6 Sequence diagram of filter module

Filter circuit according to the register of TIM1_CR2 TIx can choose to filter out pulse width 4/8/16/32 for the clock cycle of input noise.After enable the filter function, the signal after filtering than before signal delay about 4~5/8~9/16~17/32~33clock cycles.

15.1.2.2 Sampling

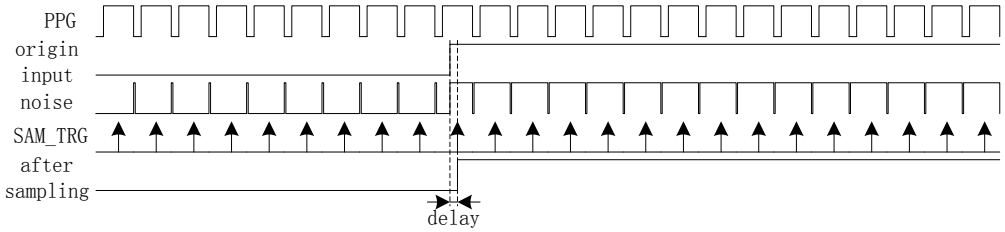


Figure 15-7 Sequence diagram of sampling module

Sampling circuit is mainly applied in the motor control of with sensorless 6/12 steps. In no HALL motor control mode, TI2/TI1/TI0 input is from the comparator. As the output of the comparator is easily interfered by the MOS switching, thus gererates noise in PPG signal. The sampling points of the sampling circuit is from TIMER2. Using the software configuration to register TIM2_CMTR, PPG signal can be sampled at any position of the input signal, and the data filtering can be saved. If the function of enable sampling is used, signal coyld be delayed with uncertain time, and the delay range is from 0 to a whoule sampling period.

15.1.3 Position detection event

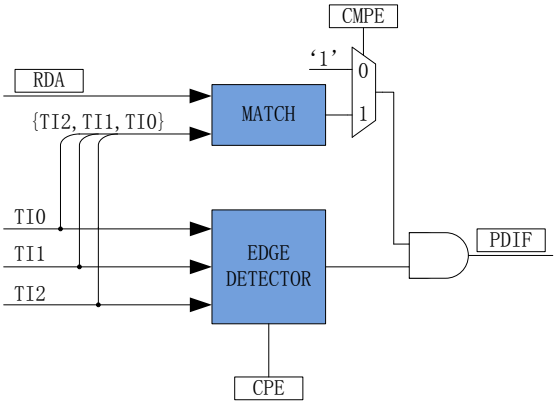


Figure 15-8 Block diagram of Position detection

Position detection by the following two events,

(1) the detection interrupt flag PDIF is set to 1;

(2) input TI2/TI1/TI0 valid along or input TI2/TI1/TI0 and RDA [2：0] due to effective along the trigger input match.

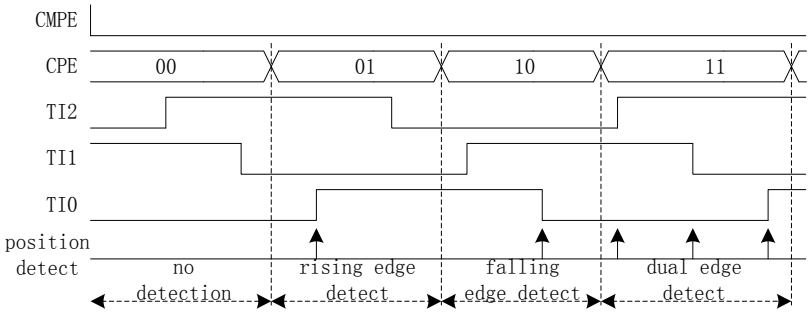


Figure 15-9 The position detection sequence diagram

If the register of TIM1_CR1CPME is set to 0, the CPE of the TIM1_CR0 register determins the effective edge of the input (nor/rising/falling/dual), when the effective edge of the input TI2/TI1/TI0 appears, position detection happens.

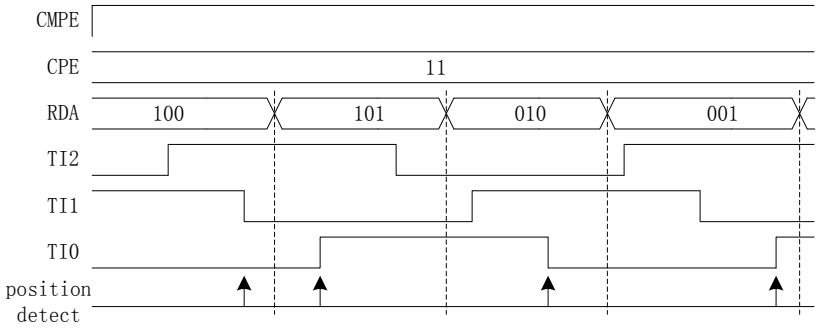


Figure 15-10 The position detection sequence diagram

If the register of TIM1_CR1CPME is set to 1, the CPE of register TIM1_CR0 detemins the effective input edge (nor/rising/falling/dual), when the effective edge of TI2/TI1/TI0 appears, the level of

TI2/TI1/TI0 changes. When it matches the value of RDA[2：0], position detection is started.

15.1.4 Writing sequence events

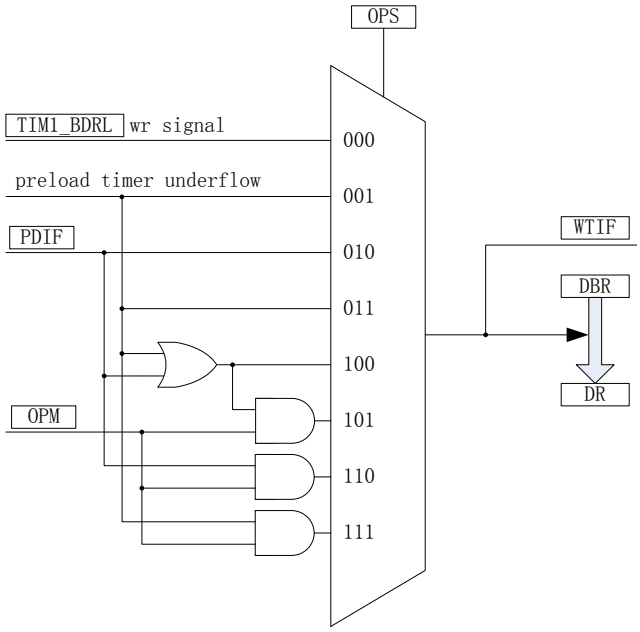


Figure 15-11 Write sequence diagram

Write sequence is determined by the OPS of register of TIM1_CR3. After writing sequence, the timing interrupt mark WTIF is set to 1, and TIM1_DRH/TIM1_DRL is reloaded with the value of TIM1_DBRH/TIM1_DBRL.

15.1.5 Output

TIMER1 has 6 outputs. The three complementary outputs of TIMER1 are TIM1_OC0/TIM1_OC1, TIM1_OC2/TIM1_OC3, and TIM1_OC4/TIM1_OC5. Dead zone interpolation is allowed. The raw output signal PPG is generated by TIMER2. Configuration of the register TIM1_CR1 WTS selectes the data updated, and output the PPG edge signal. This is to prevent short pulse interference. Dead zone module output a set of complementary PPG signals inserted in dead zone.

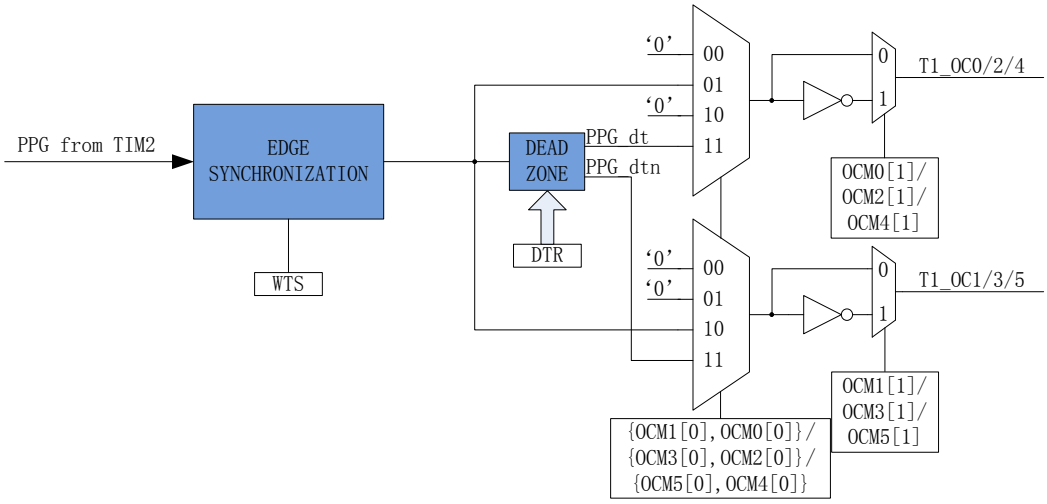


Figure 15-12 The output frame

In output, configuring the OCMx[0] of registerTIM1_DRH/TIM1_DRL can make the output mode to invalid level, or the original output PPG, or the insert complementary output in dead zone. The OCMx[1] of register of TIM1_DRH/TIM1_DRL can configure the polarity of the output. E.g.: Config OCM1[0]=1，OCM0[0]=0，and OCM1[1]=0，then T1_OC1=PPG. It is not a complementary output.

15.1.5.1 PPG edge synchronization

The short pulse output mode changing can be avoided by using PPG signal synchronization. Configuring the register of TIM1_CR1 WTS can select PPG signals rising edge, falling edge, or double the synchronous or asynchronous.

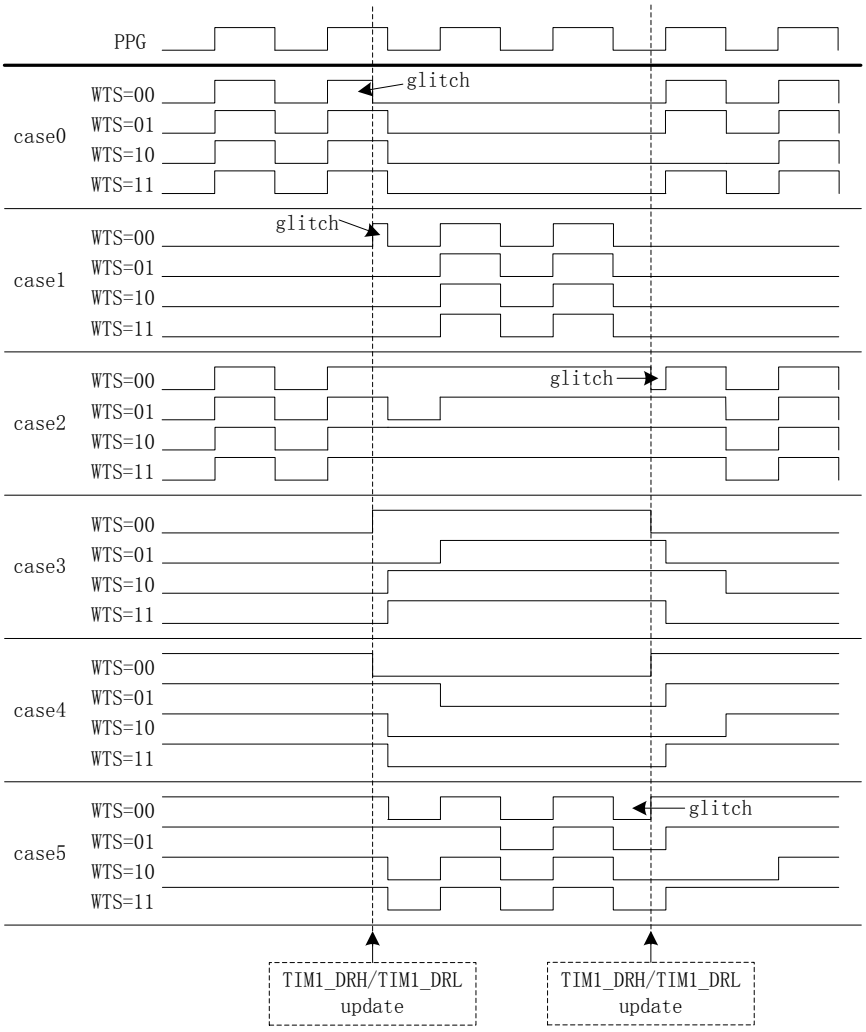


Figure 15-13 PPG Edge synchronization sequence diagram

15.1.5.2 Dead zone output

For complementary PPG signals, if the register of TIM1_DTR is not equal to zero, the dead zone is enabled. When PPG rise along happens, the output of the high level rise along of PPG_dt is higher than the delay of TIM1_DTR set time; When the falling edge of PPG happens, the high level of the actual output falling edge of PPG_dtn is higher than the seting delay time of PPG. If the delay time is longer than the actual output of pulse width, then the pulse width of the corresponding channel is not delayed, and the reverse channel pulse width does not generate.

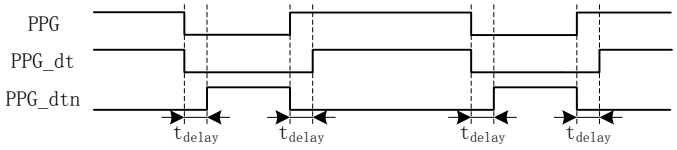


Figure 15-14 Complementary output with dead zone

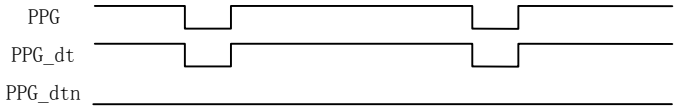


Figure 15-15 The dead zone time is longer than the negative level

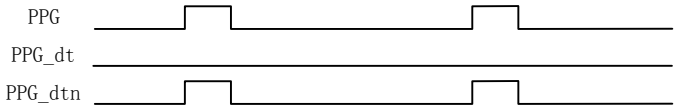


Figure 15-16 The dead zone time is longer than the positive level

15.1.6 Timer1 Interrupt

The timer1 has five interrupt request sources:

1. Basic timer overflow interrupt
2. Reload timer underflow interrupt
3. Position detection interrupt
4. Data updated interrupt
5. Compared interrupt

Configure corresponding interrupt of TIM1_IER can generate the corresponding interrupt request

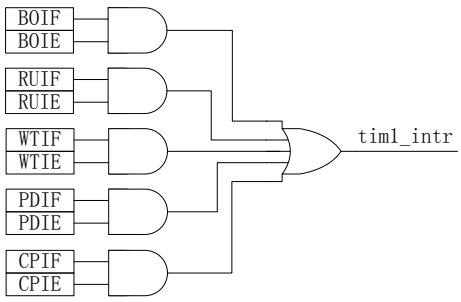


Figure 15-17 TIMER1 interrupt source

15.2 Timer1 Register

15.2.1 TIM1_CR0 (0x4068)

Table 15-2 TIM1_CR0（0x4068）

Bit	7	6	5	4	3	2	1	0
Name	CPE		INC2	INC1	INC0	IEE2	IEE1	IEE0
Type	R/W	R/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]	CPE	TI0/TI1/TI2 Input polarity selection These bits are used to select position detection for the polarity of the input edge. The position detection is triggered by the polarity of the input edge. 00: No detect 01: Rising edge 10: Falling edge 11: Both edges
[5]	INC2	Refer INC0 description
[4]	INC1	Refer INC0 description
[3]	INC0	TI0 input noise filtering enable The pulse width of the filter noise is determined by the TIM1_CR2 of INM0 [1:0] 0: filtering disable 1: filtering enable Note: enable filtering function, the input becomes invalid if the internal clock is stopped,
[2]	IEE2	TI2 input enable Refer IEE0 description
[1]	IEE1	TI1 input enable Refer IEE0 description
[0]	IEE0	TI1 input enable 0: Disable TI0 input 1: enable TI0 input Note: the CMPE of the register of TIM1_CR1 should be set to zero before setting TI0

15.2.2 TIM1_CR1 (0x4069)

Table 15-3 TIM1_CR1（0x4069）

Bit	7	6	5	4	3	2	1	0
Name	T1RCEN	ROPM	WTS		CPD			CMPE
Type	R/W	R/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]	T1RCEN	Reload timer counter is enabled 0: Disable the timer 1: Enable the timer
[6]	ROPM	Reload timer single pulse mode 0: When reloadtimer underflow event happens, counter does not stop 1: When reloadtimer underflow event happens, counter is stopped T1RCEN is removed
[5:4]	WTS	PPG Synchronous edge selection These bits are used to select and write timing synchronization of PPG signals of next sync edge. 00: No synchronous 01: Rising edgeSynchronous 10: Falling edgeSynchronous 11: Both edgeSynchronous Note: If set from TIM2 PPG as set to 0 or 1 compulsarily, WTS should be set to 00. Otherwise, PPG couldn't generate edge changing and and lead the output be fail in forcing to 0 or 1.
[3:1]	CPD	Compare bit These bits are used to compare with the RDA[2:0] of register TIM1_DRH. When the value matches RDA[2:0] , compare interrupt flag CPIF is set to 1.
[0]	CMPE	Position detection is enabled The comparison of the bit is used for enabling position detection: 0: Disable position detection compare 1: Enable position detection compare

15.2.3 TIM1_CR2 (0x406A)

Table 15-4 TIM1_CR2（0x406A）

Bit	7	6	5	4	3	2	1	0
Name	BRS	TIS	INM2		INM1		INM0	
Type	R/W	R/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]	BRS	The basic timer reset source selection 0: Write timing reset 1: The position detection reset
[6]	TIS	Input source（TI0/TI1/TI2） selection 0: The output of the comparator (CMP0/CMP1/CMP2) is selected as input 1: GPIO (P1_4/P1_6/P2_1) is selected as input
[5:4]	INM2	TI2 noise pulse width selection Refer INM0 description
[3:2]	INM1	TI1 noise pulse width selection Refer INM0 description
[1:0]	INM0	TI1 noise pulse width selection. When the noise pulse width is shorter than the setting value, the noise is filtered (If MCU clock is 24MHZ, or 41.67ns) 00: 4 Clock cycles 4 x 47.67ns 01: 8 Clock cycles 8 x 47.67ns 10: 16 Clock cycles 16 x 47.67ns 11: 32 Clock cycles 32 x 47.67ns

15.2.4 TIM1_CR3 (0x406B)

Table 15-5 TIM1_CR3（0x406B）

Bit	7	6	5	4	3	2	1	0
Name	PSC				OPS			T1BCEN
Type	R/W	R/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]	PSC	The timer clock frequency division. These bits are used to N points on MCU clock frequency as the basic and reload timer count clock, timer assumption for MCU clock is 24MHZ(41.67ns) 0000:0x1（24MHZ） 0001:0x2（12MHZ） 0010:0x4 （6MHZ） 0011:0x8（3MHZ） 0100:0x10（1.5MHZ） 0101:0x20（750KHZ） 0110:0x40 （375KHZ） 0111:0x80（187.5KHZ） 1000:0x100（93.75KHZ） 1001:0x200 （46.875KHZ） 1010:0x400（23.4375KHZ） 1011:0x600（15.625KHZ） 1100:0x800（11.71875KHZ） 1101:0xa00 （9.375KHZ） 1110:0xc00（7.8125KHZ） 1111:0xe00（6.6964KHZ）
[3:1]	OPS	Data transmission mode selection These bits are used to select the mode write TIM1_DRH/TIM1_DRL register with TIM1_DBRH/TIM1_DBRL. 000: soft write TIM1_DBRH/TIM1_DBRL 001: 16-bit reload counter underflow 010: Position detection input 011: 16-bit reload counter underflow, and restart 16-bit reload counter with position detection input 100: 16-bit reload counter underflow, or position detection input 101: 16-bit reload counter underflow ,or position detection input single pulse 110: Position detection input single pulse 111: 16-bit reload counter underflow single pulse, and restart 16-bit reload counter with position detection input
[0]	T1BCEN	Basic Time Enable 0: Disable Timer 1: Enable Timer

15.2.5 TIM1_IER (0x406C)

Table 15-6 TIM1_IER（0x406C）

Bit	7	6	5	4	3	2	1	0
Name	RSV			BOIE	RUIE	WTIE	PDIE	CPIE
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:5]	RSV	Reversed
[4]	BOIE	Base timer overflow interrupt enable 0: Disable 1: Enable
[3]	RUIE	Reload timer underflow interrupt enable 0: Disable 1: Enable
[2]	WTIE	Write timing interrupts is enabled 0: Disable 1: Enable
[1]	PDIE	The position detection interrupts are enabled 0: Disable 1: Enable
[0]	CPIE	Compare interrupt enable 0: Disable 1: Enable

15.2.6 TIM1_SR (0x406D)

Table 15-7 TIM1_SR（0x406D）

Bit	7	6	5	4	3	2	1	0
Name	RSV			BOIF	RUIF	WTIF	PDIF	CPIF
Type	R	R	R	R/W0	R/W0	R/W0	R/W0	R/W0
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:5]	RSV	Reverse
[4]	BOIF	Base timer verflow interrupt flag When base timer dose up counting, and the register of TIM1_CNTR equals to TIM1_ARR, overflow event generates. TIM1_CNTR is reset to 0. 0: None 1: Overflow event generates
[3]	RUIF	Reload timer underflow interrupt flag When reload timer does down counting, and the register of TIM1_RCNTR equals to 0, overflow event happens. TIM1_RCNTR reloads TIM1_RARR. RUIF is set 1, and can be reset to 0 with software. 0: None 1: underflow event happens
[2]	WTIF	Write timing interrupt When TIM1_DBRH/TIM1_DBRL is sent to TIM1_DRH/TIM1_DRL, WTIF is set to 1, and can be reset to 0 with software.
[1]	PDIF	Position detection interrupt flag When the register of TIM1_CR1 CMPE = 1, and the value of TI2,TI1,TI0 matches RDA[2:0], then the edge of CPE of TIM1_CR0 determins the arriving of the input edge. When TIM1_CR1 CMPE = 0, the edge of CPE of TIM1_CR0 determins the arriving of the input edge. 0: none 1: position detection event happens
[0]	CPIF	Compare interrupt flag When RDA[2:0] matches CPD[2:0], CPIF is set to 1, otherwise, set to 0.

15.2.7 TIM1_DRH (0x406E)

Table 15-8 TIM1_DRH（0x406E）

Bit	7	6	5	4	3	2	1	0
Name	RSV	RDA			OCM5		OCM4	
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

Bit	Name	Function
[7]	RSV	Reverse
[6:4]	RDA	Position detection match CPD compare Set RDA[2:0], and use it to generate position detection and CPD compare interrupt. When the register of TIM1_CR1 CMPE = 1, and the value of TI2, TI1, TI0 match to RDA[2:0], then according to TIM1_CR0 CPE detect input edge to generate position detection interrupt When RDA[2:0] matches CPD[2:0], interrupt is generated
[3:2]	OCM5	CH5 output (T1_OC5) Refer OCM1 description. T0_OC4 and T0_OC5 are Complementary channel
[1:0]	OCM4	CH4 output (T1_OC4) Refer OCM0 description, T0_OC4 and T0_OC5 are Complementary channel

15.2.8 TIM1_DRL (0x406F)

Table 15-9 TIM1_DRL（0x406F）

Bit	7	6	5	4	3	2	1	0
Name	OCM3		OCM2		OCM1		OCM0	
Type	R/W	R/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]	OCM3	CH3 output (T1_OC3) Refer OCM1 description, T0_OC2 and T0_OC3 are Complementary channel
[5:4]	OCM2	CH2 output (T1_OC2) Refer OCM0 description, T0_OC2 and T0_OC3 are Complementary channel
[3]	OCM1[1]	CH1 outputs (T1_OC1) polarity 0: T0_OC1 ‘H’ level 1: T0_OC1 ‘L’ level
[2]	OCM1[0]	CH1 (T1_OC1) outputs Enable 0:Disable 1:Enable When both OCM0[0] and OCM1[0] are 1, T0_OC0 and T0_OC1 are Complementary output. Then both T0_OC0 and T0_OC1 output inserted dead zone.
[1]	OCM0[1]	CH0 output (T1_OC0) polarity 0: T0_OC1‘H’ level 1: T0_OC1‘L’ level
[0]	OCM0[0]	CH0 (T1_OC0) outputs Enable 0: Disable 1: Enable When both OCM0[0] and OCM1[0] are 1, T0_OC0 and T0_OC1 are Complementary output.Then both T0_OC0 and T0_OC1 output inserted dead zone.

15.2.9 TIM1_DBRH/TIM1_DBRL (0x4070，0x4071)

TIM0_DBRH/TIM0_DBRL is the buff updated to TIM0_DRH/TIM0_DRL. Refer the description of TIM0_DRH/TIM0_DRL.

15.2.10 TIM1_BCCR (0x4072，0x4073)

Table 15-10 TIM1_BCCRH（0x4072）

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 15-11 TIM1_BCCRL（0x4073）

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	Capture the basic timer count In position detection or inhaled reset sequence event, the base timer save the count value to CCR register before the reseting.

15.2.11 TIM1_RARR (0x4074，0x4075)

Table 15-12 TIM1_RARRH（0x4074）

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 15-13 TIM1_RARRL（0x4075）

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	Reload timer reload value automatically. When reload timer underflow interrupt event happens, reload the timer value to RARR

15.2.12 TIM1_RCNTR (0x4076 , 0x4077)

Table 15-14 TIM1_RCNTRH（0x4076）

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 15-15 TIM1_RCNTRL（0x4077）

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	Reload timer value

15.2.13 TIM1_BCNTR (0x407A , 0x407B)

Table 15-16 TIM1_BCNTRH（0x407A）

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 15-17 TIM1_BCNTRL（0x407B）

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	Base timer value

15.2.14 TIM1_BARR (0x4078 , 0x4079)

Table 15-18 TIM1_BARRH（0x4078）

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 15-19 TIM1_BARRL（0x4079）

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	Base timer reload value When base timer value equals to BARR, overflow interrupt generates, and timer value is set to 0.

15.2.15 TIM1_DTR (0x4064)

Table 15-20 TIM1_DTR（0x4064）

Bit	7	6	5	4	3	2	1	0
Name	TIM1_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]	TIM1_DTR	Deadtime DT= (DTR+1)/24MHz Note: when DTR = 0, no deadtime

16 Capture Timer (TIM2/TIM3/TIM4/TIM5)

16.1 Capture Timer Operation

Capture timer includes output, input timer, and input counter modes.

1. output 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
3. input counter mode: detect the time of the PWM number requested

Capture timer include

1. 3-bit Programmable frequency divider for count clock frequency division of the basic counter
2. Base timer of a 16-bit up-counter, clock source is used for clock control output
3. 16-bit up-counter for input counter mode, and clock source is external clock edge of input
4. Input filter module
5. Edge detection module
6. Output mode for generating PWM/one-pulse compare output
7. Interrupt event
8. Timer2 output and used for Timer1 PPG signal. It can output ADC trigger signal and the comparator sampling signal.

16.1.1 Capture Timer clock controller

The clock controller is used to generate the basic timer and reload timer count clock source, and the counting clock frequency is divided by preacaler. Preassigned frequency detector is formed by a 8-bit counter controlled by a PSC with three registers. Eight kinds of frequency coefficients can be selected. Clock source is an internal clock. As no buffer in the control-register, the change of the frequency division coefficient will be updated immediately. Therefore, the frequency coefficient should be updated in the time both the basic timer and reload timers are not working.

The counter frequency:

$f_{CK\_CNT}=f_{CK\_PSC}$/TxPSC

If MCU clock is 24MHz (41.67ns)

Table 16-1 The values of register for different clock frequency

TxPSC	coefficient (HEX)	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

16.1.2 Output Mode

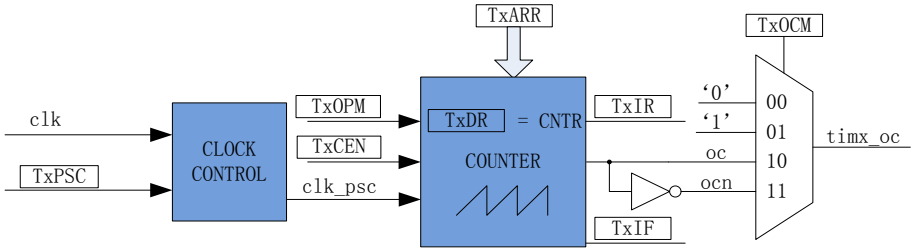


Figure 16-1 Block diagram of the output mode

Output mode outputs signal by the comparing result and the Tx_OCM in the register IMx_CR0, the interrupt is also generated at the same time.

16.1.2.1 Force output mode

Set the register of TIMx_CR0Tx_OCM= 00, the output signal is set compulsively to invalid state, i.e., TIMx_OC is always in “L” level. Set the register of TIMx_CR0Tx_OCM= 01, , the output signal is set compulsively to, i.e., TIMx_OC is always in “H” level. In force output mode, the value of TIMx_DR register compares TIMx_CNTR. When the compare matches, the flag is set and interrupt generates.

16.1.2.2 PWM Mode

Using PWM mode, PWM period is determined by TIMx_ARR. Duty = TIMx_DR/TIMx_ARR x 100%. Set the register of TIMx_CR0Tx_OCM= 10,out is decided by the comparision between TIMx_DR and TIMx_CNTR. If TIMx_CNTR≤TIMx_DR, output is “L”, otherwise is “H”. Set the register of TIMx_CR0Tx_OCM= 11, output is dedermined by the comparision between TIMx_DR and TIMx_CNTR. If TIMx_CNTR≤TIMx_DR , output is “H”, otherwise is “L”.

16.1.2.3 Interrupt event

When TIMx_CNTR = TIMx_DR, compare match event happens. Interrupt flag TIMx_CR1 IR is set to ”1”. The counter starts counting.

When TIMx_CNTR = TIMx_ARR, overflow event happens. Interrupt flag TIMx_CR1 IF is set to “1”. The counter is reseted to ”0”. TxOPM in TIMx_CR0 determins whether the counting sould be continued or not. TxOPM=1, stop; TxOPM=0, restart.

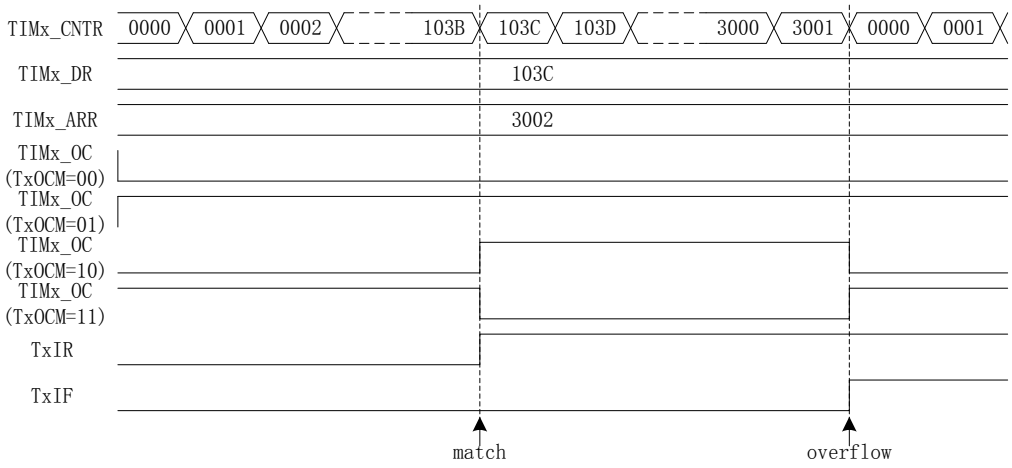


Figure 16-2 Waveform of output mode

16.1.2.4 ADC Trigger and comparator sampling functions of Timer2

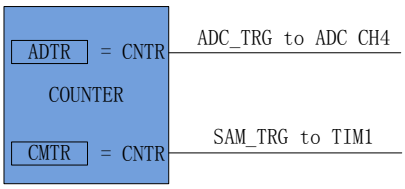


Figure 16-3 ADC Trigger and comparator sampling function of Timer2

The trigger function is to sample CH4 (Bus current) signal automatically in the motor control. Set AD_TRIG_EN of ADC_CFG bit to 1, enable ADC Trigger function. When TIM2_CNTR equals to TIM2_ADTR, trigger signal generate. Then CH4 of ADC4_DR is sampled. Config ADC_CFG AD_TRIG_IE=1, enable ADC trigger interrupt function. In the interrupt event, AD_TRIG_IF is set to ‘1’. It can be reset to ‘0’ with software.

Comparator sampling function is to sample CMP0, CMP1 and CMP2 BEMF in sensorless BLDC motor control. Set CMP_CR2CMPSAME=1, enable camparator sampling function. Set TIM2_CMTR register, when TIM2_CNTR equals to TIM2_CMTR, generates sampling for CMP0, CMP1 and CMP2

16.1.3 Input Signal Filtering and Edge Detection

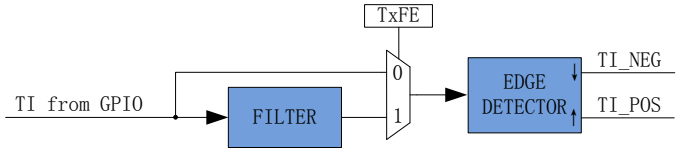


Figure 16-4 Input signal filtering and edge detection

Timer input signal TI comes from GPIO. Input is optional for noise filtering, the input of rising and falling edge of the detection module for next module.

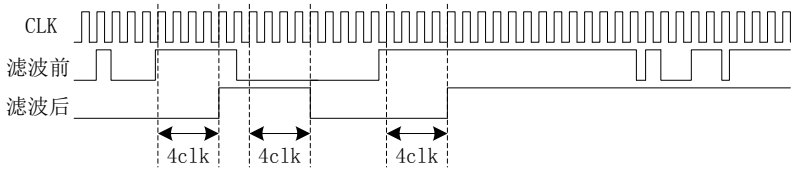


Figure 16-5 Filter module sequence diagram

The input circuit noise of the filter with fixed 4 cycles pulse width. Set TIMx_CR1Tx_FE = 1 to enable the filter function. After filtering, signals will delay 4~5 clock cycles of the one unfiltered.

16.1.4 Input Timer Mode

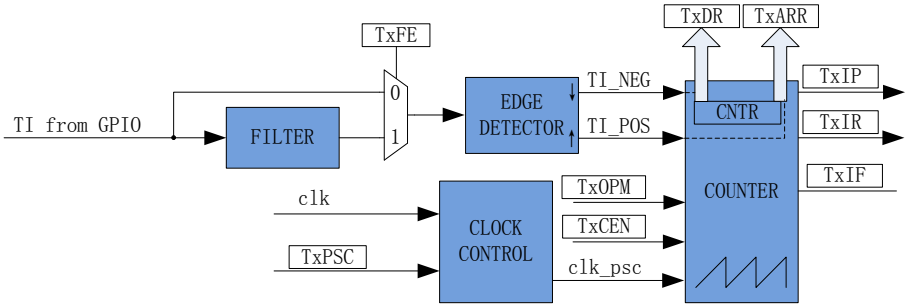


Figure 16-6 input timer mode diagram

Input timer mode detects PWM signal “H” and a period.TIMx_DR and TIMx_ARR record TIMx_CNTR.Input signal can select whether filter or not.

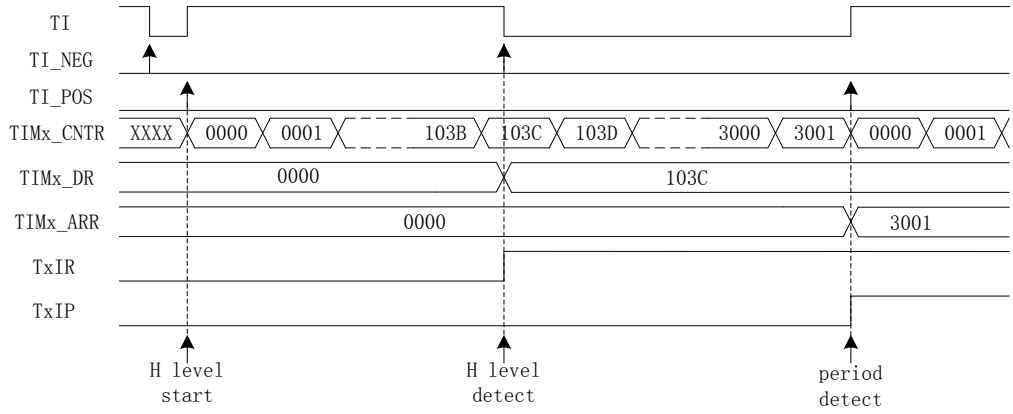


Figure 16-7 Input timer mode sequence diagram

Set TIMx_CR1TxCEN = 1 to enable counter for starting the countng. When first rising edge (falling edge valid) of the inout is detected, TIMx_CNTR is reseted to “0”and then restarts counting.

When falling edge is detected, TIMx_DR records the value of TIMx_CNTR. At the same time, TIMx_CR1 IR is set to 1. TIMx_CNTR then do the up-counting.

When second rising edge is deteced, and the a PWM cycle of the input is detected, TIMx_ARR records the value of TIMx_CNTR. Then interrupt flag TIMx_CR1 IP is set to ‘1’, and TIMx_CNTR is set to ‘0’. The TxOPM of TIMx_CR0 determines whether the counting should be continued, or not. IfTxOPM=1, the counting is stopped, and,TxOPM=0, the counting is restarted.

When the second rising edge cannot be detected, TIMx_CNTR is 0xFFFF, and overflow event is induced. The IF of Interrupt flag TIMx_CR1 is set to ‘1’, and TIMx_CNTR is set to ‘0’. TxOPM of TIMx_CR0 determins whether the counting should be continued, or not. TxOPM=1, stop; TxOPM=0, restart.

16.1.5 Input counter mode

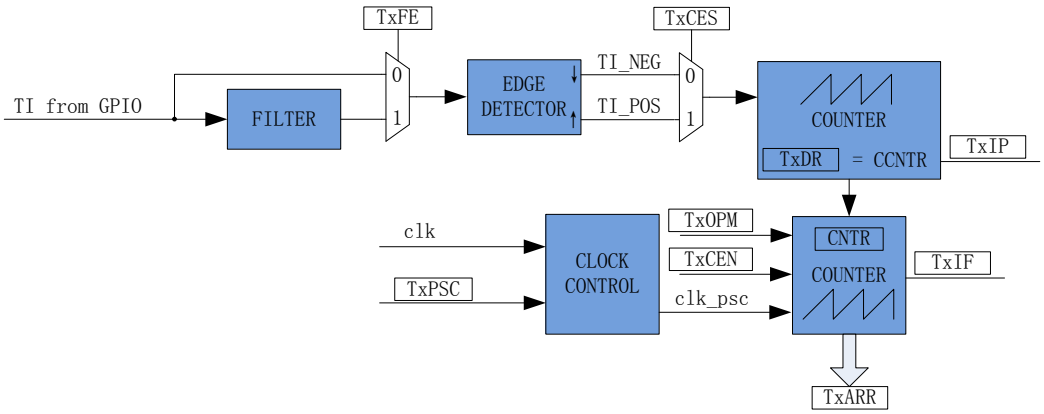


Figure 16-8 Input counter mode

Input count mode detects the time of the the number of PWM inputted. TIMx_ARR is record with the value of TIMx_CNTR. Input signal can be filtered. TIMx_CR0Tx_CES=1, the rasing edge of input signal is used as the counter valid edge; otherwise the falling edge is the valid edge.

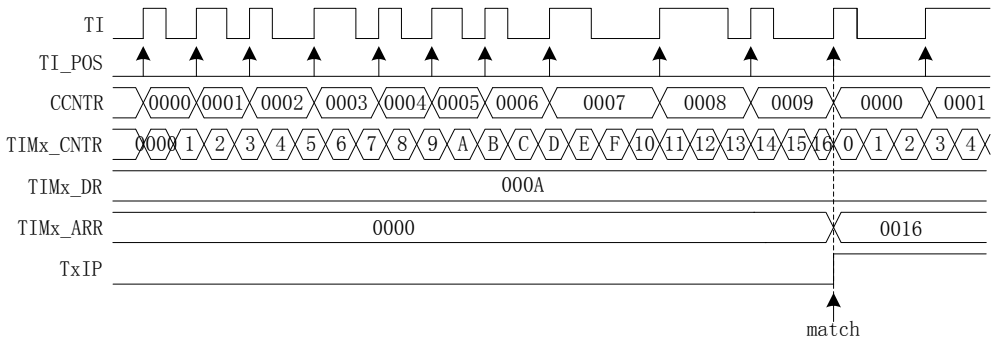


Figure 16-9 Input counter mode

TIMx_CR1TxCEN = 1, the counter is enabled to do up-counting. When timer detects input the first valid edge, TIMx_CNTR is set '0', and counter is restarted.

When timer detects valid edge, CCNTR value is added with 1. TIMx_DR sets the target value of PWM number. TIMx_ARR records the value of TIMx_CNTR when TIMx_CNTR = TIMx_DR. At the same time, TIMx_CR1 IP is set to '1', and TIMx_CNTR and CCNTR are set to '0'. The value of TxOPM in TIMx_CR0 determines whether operation of the counting; TxOPM=1 stop; TxOPM=0. restart.

If the number of input PWM <TIMx_DR, and TIMx_CNTR is 0xFFFF, the overflow event happens. IF in TIMx_CR1 is set to '1', TIMx_CNTR is set to '0', and CCNTR is configured with TxOPM=0. In this state, TIMx_CNTR is set to '0', the counting is restarted, and CCNTR continues its counting. Config TxOPM=1, timer is stopped, TIMx_CNTR ia also stopped, and CCNTR is set to 0.

16.2 Capture Timer Register

16.2.1 TIMx_CR0 (0xA1/0x9C/0x9E/0x89)

Table 16-2 TIMx_CR0（0xA1/0x9C/0x9E/0x89）

Bit	7	6	5	4	3	2	1	0
Name	TxPSC			TxOCM		TxCES	TxCTM	TxOM
Type	R/W	R/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]	TxPSC	The timer clock frequency division These bits are used to N divider on MCU clock frequency and the fractional frequency is used as the basic and reload timer count clockof the timer. E.g., if 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）
[4:3]	TxOCM	Output compare mode selection: 00: force output to 0 01: force output to 1 10: If TIMx_CNTR≤TIMx_DR, output is 0; If TIMx_CNTR >TIMx_DR, output is 1 11: If TIMx_CNTR≤TIMx_DR, output is 1; If TIMx_CNTR >TIMx_DR, output is 0
[2]	TxCES	Input counter mode valid edge selection: 0: falling edge 1: rising edge
[1]	TxCTM	Input mode selection: 0: input timer 1: input counter
[0]	TxOM	Working mode selection: 0: input mode 1: output mode

16.2.2 TIMx_CR1 (0xA9/0x9D/0x9F/0x91)

Table 16-3 TIMx_CR1（0xA9/0x9D/0x9F/0x91）

Bit	7	6	5	4	3	2	1	0
Name	TxIR	TxIP	TxIF	TxIDE	TxIDE	TxIFE	TxFE	TxOPM
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]	TxIR	Output mode : compare matches flag When TIMx_CNTR matches TIMx_DR, the bit is set to ‘1’ wiith hardware, and set to “0” wit software. Input timer mode : Flag of “H” level width When the ”H” level width is detected (from rising edge to falling edge): 0: event happen 1: none

Bit	Name	Function
[6]	TxIP	Input timer mode: PWM period detect flag When Timer detects a PWM period of input (from rising edge to falling edge), the bit is set to '1'. Input counter mode: input PWM count matches the flag When the PWM number of input reaches TIMx_DR, the bit is set to '1'. 0: happen 1: no
[5]	TxIF	Output mode: count overflow flag When TIMx_CNTR matches TIMx_ARR, TIMx_CNTR is set to '0'. TxIF is set to '1'. Input timer mode :count overflow flag If Timer cannot find a PWM period, and TIMx_CNTR is accumulated to 0xFFFF, overflow event happens. In this case, TIMx_CNTR is reset to '0'. TxIF is set to '1'. Input counter mode: basic counter overflow When the number of the PWM period <TIMx_DR, and TIMx_CNTR is accumulated to 0xFFFF, overflow event happens. In this state, TIMx_CR1 IF is set to '1', and TIMx_CNTR is set to '0'.
[4]	TxIDE	Ouput mode : compare matches interrupt enable Input timer mode: PWM period detection enables interrupt Input counter mode : input PWM count matches interrupt enable 0: disable 1: enable
[3]	TxIFE	Output mode: count overflow interrupt enable Input timer mode: count overflow interrupt enable Input timer mode: basic count overflow interrupt enable 0: disable 1: enable
[2]	TxFE	Input noise filter enable If noise width < 4 cycles, the noise is filtered. 0: disable 1: enable
[1]	TxOPM	Single mode Output mode :counter overflow Input timer mode: PWM period is detected, or count overflow event happens Input count mode: input PWM number matches, or basic count overflow
[0]	TxCEN	Basic counter enable 0: disable 1: enable

16.2.3 TIMx_CNTR(0xAA,0xAB/0xA2,0xA3/0x92,0x93/0x8A,0x8B) (x:2-5)

Table 16-4 TIMx_CNTRH（0xAB/0xA3/0x93/0x8B）

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 16-5 TIMx_CNTRL（0xAA/0xA2/0x92/0x8A）

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	Basic counter count value

16.2.4 TIMx_DR(0xAC,0xAD/0XA4,0xA5/0x94,0x95/0x8C,0x8D) (x:2-5)

Table 16-6 TIMx_DRH（0xAD/0xA5/0x95/00x8D）

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 16-7 TIMx_DRL（0xAC/0xA4/0x94/0x8C）

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 match Input timer mode: detect “H” level counter Input counter mode: input PWM number

16.2.5 TIMx_ARR(0xAE,0xAF/0XA6,0XA7/0x96,0x97/0x8E,0x8F)（x:2-5）

Table 16-8 TIMx_ARRH（0xAF/0xA7/0x97/0x8F）

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 16-9 TIMx_ARRL（0xAE/0xA6/0x96/0x8E）

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 (software write) Input timer mode: PWM cycle counted is detected (hardware write) Input counter mode: input PWM count matches the value of basic timer (hardware write)

16.2.6 TIM2_CMTR（Only for TIMER2）（0xB2，0xB3）

Table 16-10 TIM2_CMTRH (Only for TIMER2) (0xB3)

Bit	7	6	5	4	3	2	1	0
Name	TIM2_CMTRH							
Type	R/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 16-11 TIM2_CMTRL（Only for TIMER2）(0xB2)

Bit	7	6	5	4	3	2	1	0
Name	TIM2_CMTRL							
Type	R/W	R/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_CMTR	CMP0，CMP1，CMP2 compare sample point set

16.2.7 TIM2_ADTR (Only for TIMER2) (0xB4 , 0xB5)

Table 16-12 TIM2_ADTRH (Only for TIMER2) (0xB5)

Bit	7	6	5	4	3	2	1	0
Name	TIM2_ADTRH							
Type	R/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 16-13 TIM2_ADTRL (Only for TIMER2) (0xB4）

Bit	7	6	5	4	3	2	1	0
Name	TIM2_ADTRL							
Type	R/W	R/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_ADTR	ADC channel CH4 of ITRIP circuit , trigger point set

17 Watchdog timer (WDT)

17.1 WDT operating instructions

17.1.1 Basic function block diagram

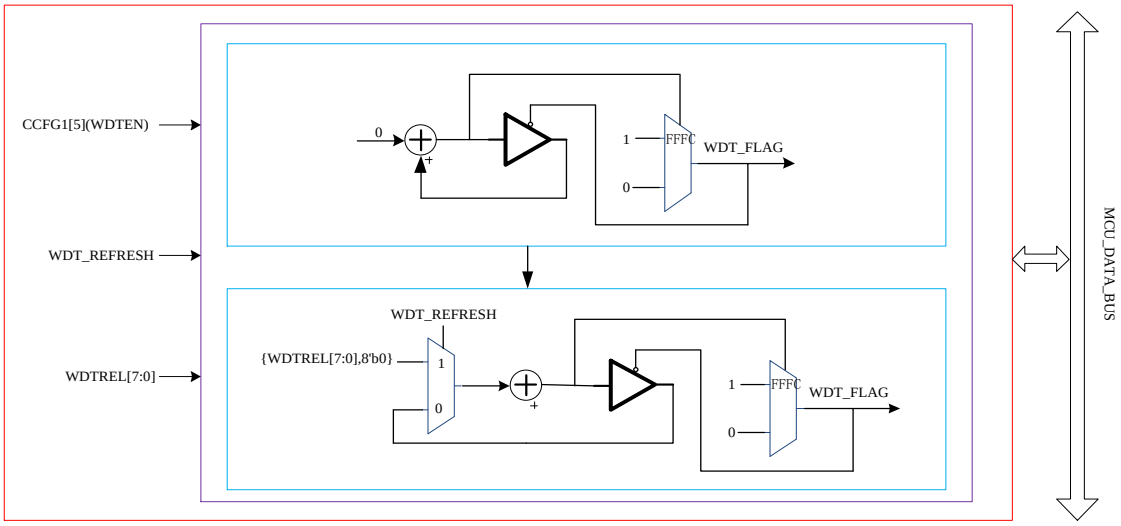


Figure 17-1 WDT basic function block diagram

17.1.2 Basic operating instructions

FU68xx has an internal 16-bit watchdog timer which works under LS_OSC clock.

1. The watchdog is initialized by software after power on.
2. WDT_CSR[WDTEN] is set to logic 1 to start the watchdog timer.
3. Watchdog timer starts counting from {16'h0}, generating the watchdog reset when count reaches FFFC. It takes FFFC~FFFF for the duration of watchdog reset .
4. Then watchdog timer starts counting again form {WDT_REL,8'h0}.
5. Set WDT_CSR[WDTF] to 1 when watchdog timer generates reset event. This flag is set to 1 by hardware and cleared by software.
6. The watchdog timer will be initialized to {WDT_REL，8'h0} When Set WDT_CSR[WDTRF] to 1.
7. The watchdog timer will stop counting as long as the chip turns into the IDE transition condition.
8. The watchdog reset can choose to start boot.

17.2 WDT register

17.2.1 WDT_CSR (0x4026)

Talbe 17-1 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

Bit	Name	Function
[7:2]	RSV	Reserved
[1]	WDTF	Watchdog reset flag
[0]	WDTRF	1: Watchdog timer initialization 0: No initialization

17.2.2 WDT_REL (0x4027)

Table 17-2 WDT_REL（0x4027）

Bit	7	6	5	4	3	2	1	0
Name	WDT_REL							
Type	R/W	R/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]	WDT_REL	This sets the highest 8 bits the watchdog timer after reset.

18 RTC

18.1 Operating instructions

18.1.1 Basic function block diagram

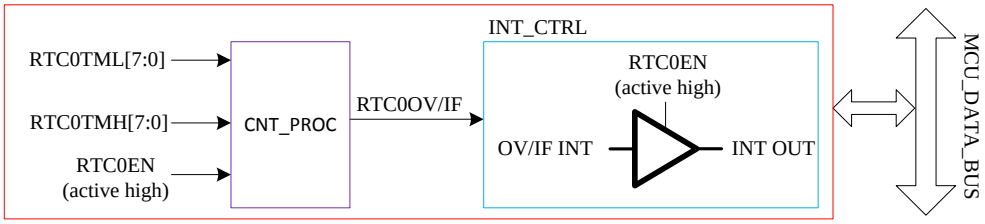


Figure 18-1 the basic function block diagram

18.1.2 Operating instructions

Write RTC0TMH and RTC0TML registers to set the count of RTC.

Set RTC0STA[RTC0EN]=1 to enable RTC.

18.2 RTC register

18.2.1 Count register：RTC0TM (0x4065 , 0x4066)

Table 18-1RTC0TMH（0x4065）

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 18-2RTC0TML（0x4066）

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

Bit	Name	Function
[15:0]	RTC0TM	RTC count register RTC counter counts from 0 to the value that user sets in RTC0TM[15:0] with the clock 32768Hz. When RTC has overflowed and it generates an interrupt. After which RTC reloads to 1 to start counting again.

18.2.2 Conctrol register：RTC0STA (0x4067)

Table 18-3

Bit	7	6	5	4	3	2	1	0
Name	RTC0EN	RTC0OV / RTC0IF	RSV					
Type	R/W	R/W	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	RTC0EN	RTC enable 0: Disabled 1: Enabled
6	RTC0OV/ RTC0IF	RTC counter overflow/interrupt flag An interrupt will be generated when this bit overflows with RTCIE=1. It can be cleared by software. An interrupt will not be generated when this bit overflows with RTCIE=0, but the flag still exist which can be cleared by MCU after read. 0: RTC does not overflow. 1: RTC overflows, cleared by software. Note: this bit will not be cleared if MCU does not clear. It is cleared by software if RTC interrupt occurs.
5:0	RSV	Reserved

19 IO

19.1 IO Operating instructions

1. Each bit in the register P0，P1，P2，P3 connects to the pin P0.0~P0.7，P1.0~P1.7，P2.0~P2.7，P3.0~P3.7 separately
2. Registers P0_OE,P1_OE,P2_OE,P3_OE control the output enable of the pin P0.0~P3.7.
3. Every GPIO pin has a configurable pull-up resistance on-chip. Each pull-up resistance can be enabled separately for the pin P0.0~P3.7 by setting the P0_PU,P1_PU,P2_PU,P3_PU. For the pins P0.0~P0.1,P1.4~P1.7，P2.1~P2.2, the build-in pull-up resistance are about 4.7KΩ. For the reset of the GPIO's, they are about 50KΩ.
4. P1.4~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. For controlling the predriver or gate-driver, there are 6 signals: UH, VH, WH, UL, VL, WL. When is selection bit MOE is "0", they are controlled direct by the register bit OISUH/OISVH/OISWH/OISUL/OISVL/OISWL. When is selection bit MOE is "1", they are controlled by internal signals OCUH/OCVH/OCWH/OCUL/OCVL/OCWL, which output from the timer 0, timer 1 or FOC module by OCS choice it.
6. The output of TIM0 can select between T0_OC4, which is the channel 4 output of 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 take place.
8. IO priority for various modes
   a) As most of the IOs has multiple function, it can't be enabled at the same time. There is a priority for each of them.
   b) P0.1：I2C > TIMER4 > GPIO
   c) P0.5：UART > SPI > GPIO
   d) P0.6：UART > SPI > GPIO
   e) P0.7：CMP > SPI > TIMER5 > GPIO
   f) For the UART's priority, UT1EN is higher prior than UT0EN. When both UT1EN and UT0EN are bit "1", P0.6 takes as RXD of UART, and both P0.5 and P3.4 take as TXD of UART.

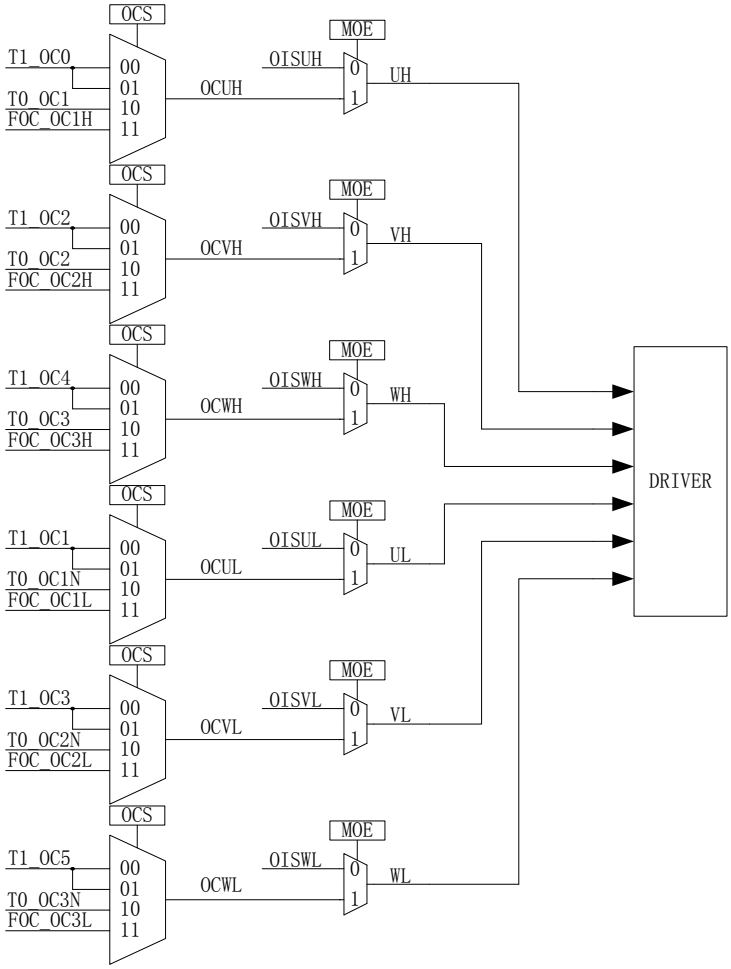


Figure 19-1 U/V/W output configuration

19.2 IO Registers

19.2.1 P0_OE (0xFC)

Table 19-1 P0_OE（0xFC）

Bit	7	6	5	4	3	2	1	0
Name	P0_OE							
Bype	R/W	R/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 1: Enable output 0: Disable output

19.2.2 P1_OE (0xFD)

Table 19-2 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 1: Enable output 0: Disable output

19.2.3 P2_OE (0xFE)

Table 19-3 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 1: Enable output 0: Disable output

19.2.4 P3_OE (0xFF)

Table 19-4 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 1: Enable output 0: Disable output

19.2.5 P1_AN (0x4050)

Table 19-5 P1_AN（0x4050）

Bit	7	6	5	4	3	2	1	0
Name	P1_AN				HBMODE	RSV	ODE1	ODE0
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
[7:4]	P1_AN	P1.7~P1.4 analog mode enable 1: Enable 0: Disable
[3]	HBMODE	P1.3 function mode selection. Combinate the P1_OE.3 to choice the function mode
[2]	RSV	Reserved
[1]	ODE1	P0.1 open drain output enable 1: Enable 0: Disable
[0]	ODE0	P0.0 open drain output enable 1: Enable 0: Disable

HBMODE	P1_OE.3	P1.3 Function Mode
0	0	Digital input
0	1	Digital output
1	0	Analog pin mode
1	1	Digital output with enhanced driving ability when output 1.

19.2.6 P2_AN (0x4051)

Table 19-6 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 1: Enable 0: Disable

19.2.7 P3_AN (0x4052)

Table 19-7 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 1: Enable 0: Disable

19.2.8 P0_PU (0x4053)

Table 19-8 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 1: Enable 0: Disable

19.2.9 P1_PU (0x4054)

Table 19-9 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 1: Enable 0: Disable

19.2.10 P2_PU (0x4055)

Table 19-10 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 1: Enable 0: Disable

19.2.11 P3_PU (0x4056)

Table 19-11 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 1: Enable 0: Disable

19.2.12 DRV_CTL (0x404D)

Table 19-12 DRV_CTL（0x404D）

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

Bit	Name	Function
[7:4]	RSV	Reservec
[3:2]	OCS	Output source selection for the OCUH, OCVH, OCWH, OCUL, OCVL, OCWL 0X: Select time 1 output (T1_OC0, T1_OC1, T1_OC2, T1_OC3, T1_OC4, T1_OC5) 10: Select time 0 output (T0_0C1, T0_OC1N, T0_0C2, T0_OC2N, T0_0C3, T0_OC3N) 11: Select FOC output (FOC_OC1H, FOC_OC1L, FOC_OC2H, FOC_OC2L, FOC_OC3H, FOC_OC3L)
[1]	PDRVEN	Predriver enable 1: Enable 0: Disable Notes: It works only in predriver output mode. In 3P3N pridirver mode, set PDRVEN to enable VBB LDO, wait about 1mS for VBB to stablized. Ignore this bit under gate-driver mode.
[0]	DRVOE	Driver output enable 1: Enable 0: Disable Notes: If in 3P3N pridirver mode, set PDRVEN to enable VBB LDO, a timing of 1mS must lapse before setting of DRVOE. This will avoid unpredictable voltage in the predriver output pins.

19.2.13 DRV_OUT (0xF8)

Table 19-13 DRV_OUT（0xF8）

Bit	7	6	5	4	3	2	1	0
Name	MOE	OIS4	OISWL	OISWH	OISVL	OISVH	OISUL	OISUH
Type	R/W	R/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]	MOE	Driver output source selection. 1: Output from one of these block: timer 0, timer 1 or FOC. If timer 0 is selected, it outputs the t0_OC4, which is channel 4 of timer 0. 0: Output from register DRV_OUT[5:0], and timer 0 outputs OIS4. This bit can be set or cleared by software. When over-current event takes place, it is cleared by hardware and will reset the driver output
[6]	OIS4	Time 0 output level in idle state. When MOE equals “0”, driver output the timer 0’s output.
[5]	OISWL	WL’s idle state, refer OISUH’s description
[4]	OISWH	WH’s idle state, refer OISUH’s description
[3]	OISVL	VL’s idle state, refer OISUH’s description
[2]	OISVH	VH’s idle state, refer OISUH’s description
[1]	OISUL	UL’s idle state, refer OISUH’s description
[0]	OISUH	UH’s idle state, refer OISUH’s description.

19.2.14 PH_SEL (0x404C)

Table 19-14 PH_SEL（0x404C）

Bit	7	6	5	4	3	2	1	0
Name	RSV	UT1EN	UT2EN	T4SEL	T3SEL	T2SEL	T5SEL	T0CH4SEL
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

Bit	Name	Function
[7]	RSV	Reserved
[6]	UT1EN	UART function enable. 1: Enable UART. Pin P0.6 is regarded as RXD; P0.5 is regarded as TXD. 0: Disable UART. Pin P0.6 is regarded as GPIO or other function mode pin. P0.5 is regarded as GPIO or other function mode pin.
[5]	UT2EN	UART function enable in function transfer mode. 1: Enable UART. Pin P3.3 is regarded as RXD; P3.4 is regarded as TXD. 0: Disable UART. Pin P3.3 is regarded as GPIO or other function mode pin. P3.4 is regarded as GPIO or other function mode pin.
[4]	T4SEL	P0.1 in TIM4 mode control bit. 1: P0.1 is regarded as timer 4's capture pin or pwm ouput pin 0: P0.1 is regarded as GPIO or other function mode pin. Note, the I2C mode is prior to TIMER4. P0.1 is regarded as SCL when I2C is enable.
[3]	T3SEL	P1.1 in TIM3 mode control bit. 1: P1.1 is regarded as timer 3's capture pin or pwm ouput pin 0: P1.1 is regarded as GPIO or other function mode pin.
[2]	T2SEL	P1.0 in TIM2 mode control bit. 1: P1.0 is regarded as timer 2's capture pin or pwm ouput pin 0: P1.0 is regarded as GPIO or other function mode pin.
[1]	T5SEL	P0.7 in TIM5 mode control bit. 1: P0.7 is regarded as timer 5's capture pin or pwm ouput pin 0: P0.7 is regarded as GPIO or other function mode pin.
[0]	T0CH4SEL	P3.2 in TIM0 mode control bit. 1: P3.2 is regarded as timer 0 channel 4 pwm ouput pin 0: P3.2 is regarded as GPIO or other function mode pin.

19.2.15 P0（0x80）/P1（0x90）/P2（0xA0）/P3（0xB0）

The pin registers support read-modify-write action. They are P0~P3 regisiters. The read-modify-write (RMW) instuctions are listed at Table 19-16。

Table 19-15 P0/P1/P2/P3

Bit	7	6	5	4	3	2	1	0
Name	Px[7]	Px[6]	Px[5]	Px[4]	Px[3]	Px[2]	Px[1]	Px[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]	端口寄存器 0 Pin register 0	R/W	0x00
P1[7:0]	端口寄存器 1 Pin register 1	R/W	0x00
P2[7:0]	端口寄存器 2 Pin register 2	R/W	0x00
P3[7:0]	端口寄存器 3 Pin register 3	R/W	0x00

Table 19-16 read modify write instructions

Instruction	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

20 Clock and oscillator

There are four clock modules in this chip, the faster internal oscillator, the slower internal oscillator, the faster external oscillator and slower external oscillator. The system clock can be either the faster internal oscillator or faster external oscillator, which is depended by the configuration register and program. The slower internal oscillator is the clock of watch dog so it can be configured to change the overflow time of watch dog. The faster external oscillator is used in the external clock input mode or the external crystal oscillator mode. The slower external oscillator is used in RTC counting.

20.1 the faster external oscillator

20.1.1 operation instruction of faster external oscillator

There are two different modes of the faster external oscillator, crystal input mode and external clock input mode.

20.1.1.1 crystal input mode of the faster external oscillator

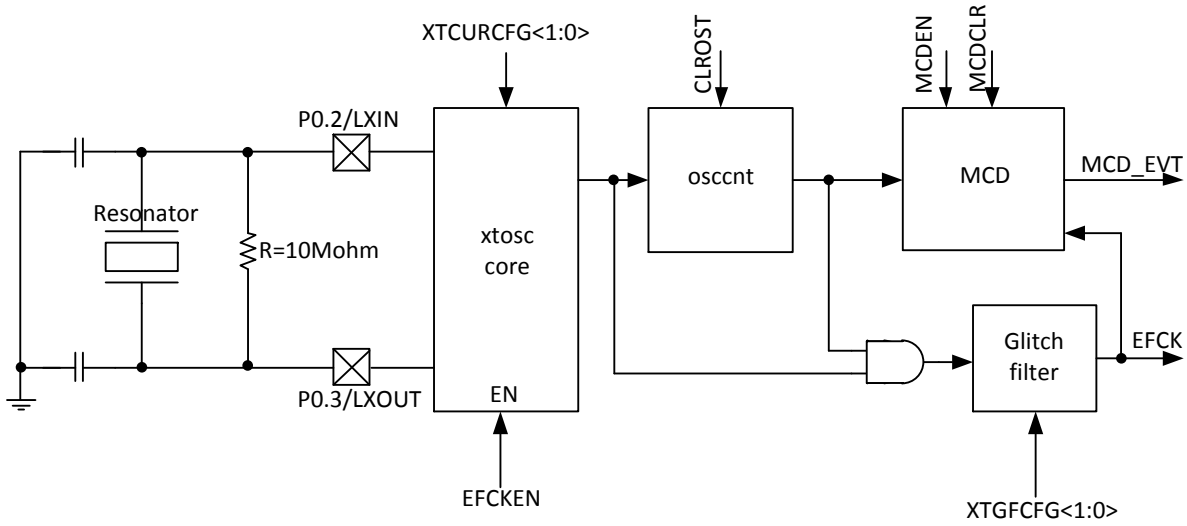


Figure 20-1 the crystal input mode of faster external oscillator module, xtosc

The crystal input mode of faster external oscillator is showed in Figure 20-1. The function of faster external oscillator module is to generate a clock with frequency is exactly 24MHz.

To set the faster external oscillator module xtosc at crystal input mode, the register Flash should be configured: ECMOD=0.

The connection of crystal out of the chip is showed in Figure 20-1.

To enable the xtosc module, the register should be configured: EFCKEN=1.

20.1.1.2 external clock input mode of faster external oscillator

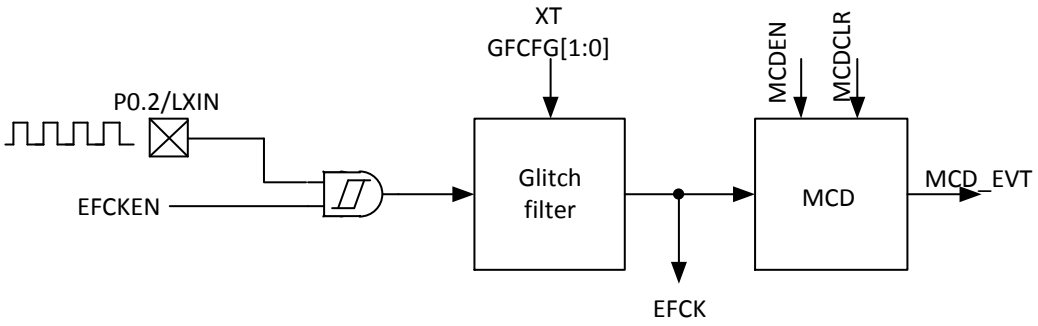


Figure 20-2 faster external oscillator external clock input mode

The external input mode of faster external oscillator is illustrated in Figure 20-2.

To enable the faster external oscillator module xtosc as external clock input mode, the register Flash must be configured as : ECMOD=1.

External clock is input from the LXIN pad.

To enable the faster external oscillator as external clock input mode, the register should be configure as : EFCKEN=1.

To clear the OST counter, set CLR_OST to logic 1. When module is enabled as crystal input mode, the OST counter will first count 1024 cycles, and then output the clock.this can let the clock be stable when it output.

The OST counter doesn't work when the module work in the external clock input mode. OST counter will set to zero by hardware when the clock module is not enabled. When the external clock is absent, the system clock will switch over to internal oscillator automatically. A high pulse should be sent to CLR_OST to clear the counting of OST counter if system clock reverts to external clock by software

To enable crystal input mode, set XTOSC_EN to logic 1.

20.1.2 the register of the faster external oscillator

20.1.2.1 OSC_CFG (SFR: 0xF1)

Table 20-1 faster external oscillator's register

Bit	7	6	5	4	3	2:0
Name	EFCKEN	IFCKEN	CKFLAG /MCDRET	ESOSCAE	ESOSCEN	RSV
Type	R/W	R/W	R/W	R/W	R/W	R
Reset	0	0	0	0	0	0

Bit	Name	Function
7	EFCKEN	The enable signal of faster external oscillator 1, Force enabled of the faster external oscillator; 0, Enable or disable the faster external oscillator depending on the EC_MODE value
6:0		Other 7 bits need refer to the Table 21-1

20.2 faster internal oscillator

20.2.1 the operation instruction of faster internal oscillator

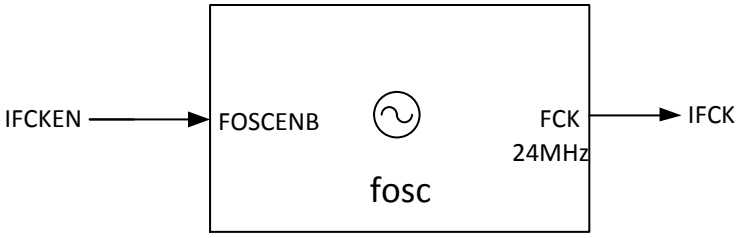


Figure 20-3 input and output signal of faster internal oscillato

The input and output pin of the faster internal oscillator is illustrated in Figure 20-3. The function of faster internal oscillator module is to generate a clock with frequency exactly equal to 24MHz. it is required to configure IFCKEN to logic 0 to enable the faster internal oscillator. When the chip is in sleep mode(PCON.STOP=1), faster internal oscillator doesn't work.

20.2.2 register of faster internal oscillator

20.2.2.1 OSC_CFG (SFR: 0xF1)

Table 20-2 register of the faster internal oscillator

Bit	7	6	5	4	3	2	1	0
Name	EFCKEN	IFCKEN	CKFLAG /MCDRET	ESOSCAE	ESOSCEN	RSV		
Type	R/W	R/W	R/W	R/W	R/W	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
6	IFCKEN	Enable signal of faster internal oscillator 1, Force enable of the faster internal oscillator 0, Enable or disable the faster internal oscillator depending on the configuration of the CKMOD
7,5:0		Other 7 bits need refer to the Table 21-1

20.3 slower external oscillator

20.3.1 operation instruction of the slower external oscillator

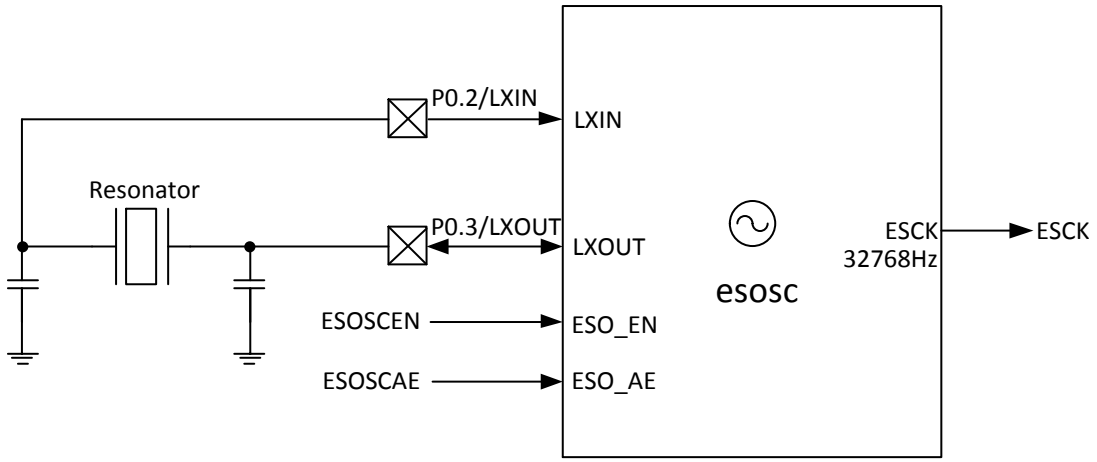


Figure 20-4 input and output signal of slower external oscillator

The Figure 20-4 show input and output signal of slower external oscillator. The function of slower external oscillator is to provide a clock with the frequency is 32768Hz.

It is required to configure: ESOSCEN=1, ESOSCAE=1 to enable slower external oscillator.

20.3.2 register of slower external oscillator

20.3.2.1 OSC_CFG(SFR: 0xF1)

Table 20-3 register of slower external oscillators

Bit	7	6	5	4	3	2	1	0
Name	EFCKEN	IFCKEN	CKFLAG /MCDRET	ESOSCAE	ESOSCEN	RSV		
Type	R/W	R/W	R/W	R/W	R/W	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
4	ESOSCAE	Enable the PIN of slower external oscillator to analog input 0, PIN of slower external oscillator is analog input depending on other IO control bits 1, PIN of slower external oscillator is analog input
3	ESOSCEN	Enable signal of slower external oscillator 0, Disable the slower external oscillator 1, Enable the slower external oscillator
7:5,2:0		Other 6 bits need refer to Table 21-1

20.4 slower internal oscillator

20.4.1 operation instruction of slower internal oscillator

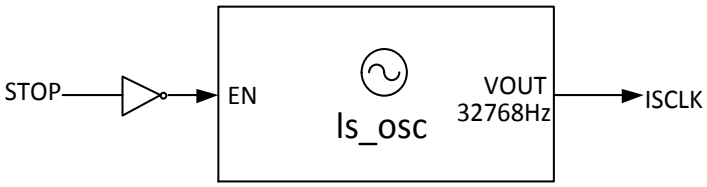


Figure 20-5 input and output signal of slower internal oscillator

The input and output signal of slower internal oscillator is illustrated in Figure 20-5. The function of slower internal oscillator is to provide the clock with frequency is 32768Hz. The slower internal oscillator doesn't work when the chip is in the sleep mode(PCON.STOP=1).

20.4.2 register of the slower internal oscillator

None.

21 Clock control and Missing Clock Dection (MCD)

21.1 Crystal oscillator configuration register. OSC_FIG

Table 21-1OSC_CFG（0xF1）

Bit	7	6	5	4	3	2:0
Name	EFCKEN	IFCKEN	CKFLAG /MCDRET	ESOSCAE	ESOSCEN	RSV
Type	R/W	R/W	R/W	R/W	R/W	R
Reset	0	0	0	0	0	0

Bit	Name	Function
7	EFCKEN	Enable external fast clock. 0: External fast clock is controlled by CKMOD. 1: Force external fast clock enable
6	IFCKEN	Internal fast clock oscillation control bit 0: Internal fast clock oscillation is controlled by CKMOD. 1: Force internal fast clock oscillation enable
5	CKFLAG / MCDRET	This bit has two functions. It acts as CKFLAG under read operation and MCDRET under write operation. Read this bit return the flag indicating system clock mode. 0: Indicate system clock running at internal fast clock. 1: Indicate system clock running at external fast clock. Note. When software set the CKMOD, system clock may be not choicing the external litteraly. It is also decided by dual speed mode and MCD mode. The CKFLAG indicates current system clock state at any time. When MCD_EVT taking place, means missing external clock, MCU will switch the system clock to internal fast clock. MCU will exit MCD_EVT modeand return system clock to external mode by write “1” to MCDRET. No action takes place when write “0” to MCDRET.
4	ESOSCAE	It set the pins for external slow crystal oscillator to analog mode. The external slow crystal oscillator pins: P0.2/LXIN, P0.3/LXOUT. 0: P0.2 and P0.3 are analog mode as decided by related IO control. 1: P0.2 and P0.3 are in analog mode.
3	ESOSCEN	External slow clock oscillation control bit. 0: Disable external slow clock oscillation. 1: Enable external slow clock oscillation.

Bit	Name	Function
2：0	RSV	Reserved

21.2 MCD and Dual-Speed Mode

The MCU has a Missing Clock Detect (MCD) function to monitor the system clock. When the MCU runs at external fast clock mode, if state MCDEN=1, the MCU will detect a MCD event when the external clock has stopped. It will then automatically switch the system clock to internal clock mode and to keep the system running. Hardware will generate an interrupt MCD event is detected Software must clear the MCDIF flag in the interrupt service routine. A read of CKFLAG returns the current system clock status, internal fast clock or external fast clock.

When system clock switches to internal fast clock due to a MCD event, a write of OSC_CFG.CKFLAG=1 switches the system clock back to external fast clock.

Notes, it is strongly recommended setting CCFG3.SPD2 when MCDEN=1.

Setting SPD2 will enable dual-speed mode. If external fast oscillion is not ready when enabled, it switches from the system clock to external clock. This will make the MCU run at internal fast clock mode first, and switch to external fast clock mode after the external fast clock is stable.

22 ADC

22.1 ADC functional block diagram

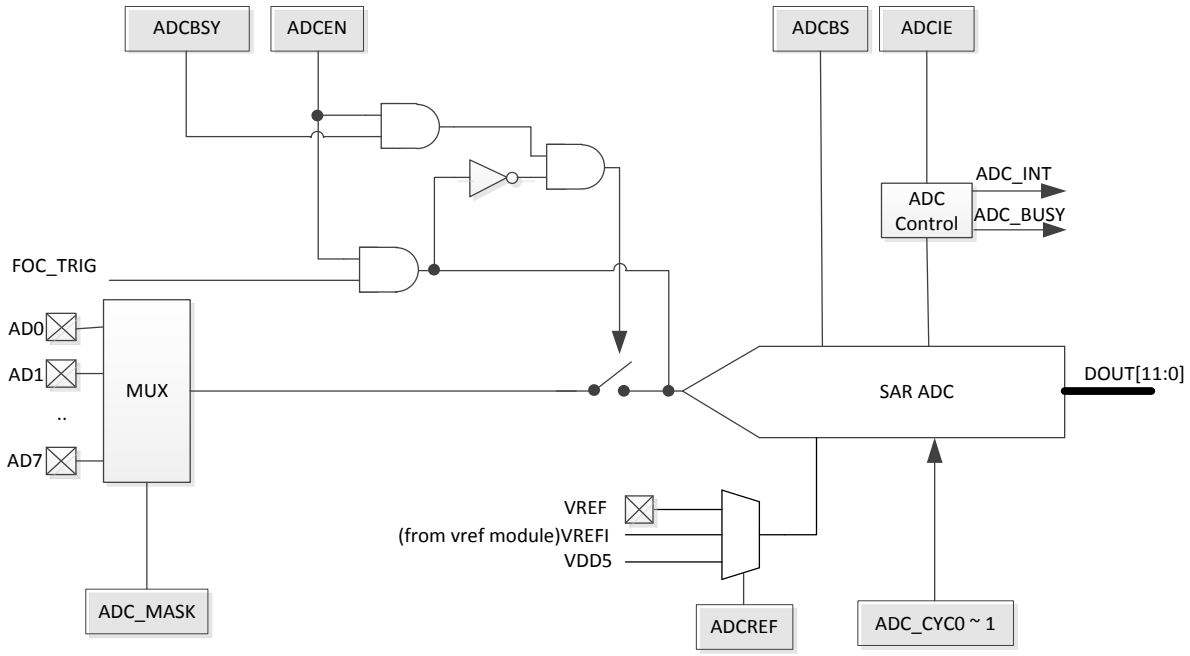


Figure 22-1 ADC functional block diagram

22.2 ADC operation instructions

FU68XX has an integrated 8 channels synchronous ADC with the resolution that can be chosed 10 bit or 12bit, respectively. The ADC_MASK register, as well as the ADCEN bit and ADCBSY bit in ADC_STA register should be set appropriately prior to ADC operation.

In addition, ADC supports the trigger function, with a priority higher than software operation of MCU. The trigger sources can come from the TIMER2 block and FOC block, and as long as FOC block is enabled, the trigger function for ADC will be disabled automatically.

TIM2_ADTR register as well as the ADC trigger function must be set appropriately in order to use the trigger function with TIMER2, namely setting the ADCEN=1 and AD_TRIG_EN=1. Once the trigger conditions are satisfied, it will start ADC automatically and store the value into ADC4_DR register. For more details, please refer to section 16.1.2.

If foc function is launched (set FOC_EN=1 in FPC_SET0 register), it will start ADC automatically and trigger the ADC to sample when needed, meanwhile loading the value of outputs into FOC block as well as the corresponding ADCx_DR register.

NOTE: ADC sampling by the software opration will be interrupted due to higher-priority of ADC trigger function. ADC will interrupt the current operation, such as ADC sampling by the software opration, to do the trigger function when the trigger conditions are satisified. After completing the sampling by trigger function, sampling by the software opration 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. Set the resolation ADCBS.
4. 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 ADC0~7_DR are available to be read by MCU after the end of the ADC conversion.
5. TIMER2 trigger function:
   a) Set TIMER2 as output mode, set TIME2_ADTR, then start TIMER2
   b) Set to enable interrupt ADTRIGIE
   c) Set ADTRIGEN=1
   d) ADC will start to sample the data as long as the trigger conditions are satisfied. After the sampling progress, the value of ADC4_DR will be updated.
6. FOC trigger function:
   a) Launch FOC function
   b) Set to enable interrupt ADTRIGIE
   c) ADC will start to sample the data as long as the trigger conditions are satisfied. After the sampling progress, the related registers will be updated.

22.3 ADC register

22.3.1 ADC_STA (0x4037)

Table 22-1 ADC_STA (0x4037)

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

Bit	Name	Function
[7]	ADCEN	ADC enable 0: Disabled 1: Enabled
[6]	ADCBSY	ADC busy flag With ADCEN enabled, setting this bit to logic 1 will start the conversion process. This bit is cleared automatically by hardware when ADC conversion is complete. Any write of logic 0 is invalid. Reading this bit indicates the ADC state (exclude the TRIG state). When ADC_MASK=0, any write to this bit will be ignored.
[5]	ADCBS	ADC resolution select. 0: 12bit ADC (default) 1: 10bit ADC NOTE: Besides being stored to its corresponding block, All the trigger data are also stored in its corresponding ADCx_DR. The data are also controlled by ADCBS.
[4:2]	RSV	Reserved
[1]	ADCIE	ADC interrupt enable. If enabled, ADTRIGIF can send an interrupt request to MCU (TRIG mode interrupt is not included). 0: Disabled 1: Enabled
[0]	ADCIF	ADC conversion end flag. If ADCIE=1, an interrupt event will be triggered when the conversion has completed. ADC_IE has no effect on this bit. 0: Conversion is not yet completed 1: Conversion is completed This bit uses the interrupt entry with the ADTRIGIF bit, so it is necessary to confirm if it is due to AD_TRIG_IF or ADCIF, before clearing the respective bit.

22.3.2 ADC_CFG (0x4035)

Table 22-2 ADC_CFG（0x4035）

Bit	7:5	4:3	2	1	0
Name	RSV	ADCREF	ADTRIGEN	ADTRIGIE	ADTRIGIF
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

Bit	Name	Function
[7:5]	RSV	Reserved
[4:3]	ADCREF	the reference voltage select for ADC 0: VDD5 available 1: External VREF available 2: Internal VREF available 3: Internal VREF available, put the VREF to PAD simultaneously NOTE: 1: Set the P3.5 as analog mode (P3_AN [5] =1) before selecting the VREF as reference voltage. 2: Set the VREFEN bit in VREF_CR register to 1 when selecting the VREF as reference voltage.
[2]	ADTRIGEN	ADC TRIG function enable 0: Disabled 1: Enabled NOTE: In order to trigger ADC with TIMER2, TIME2_ADTR must also be set to start TIMER2.
[1]	ADTRIGIE	ADC trigger interrupt enable. It is used to control if ADTRIGIF sends an interrupt request to MCU. 0: Disabled 1: Enabled
[0]	ADTRIGIF	ADC trigger mode ending flag. This bit is affected by ADTRIGIE. 1: ADC trigger event occurred and ended 0: No ADC trigger event This bit uses the interrupt entry with the ADCIF bit together, so it is necessary to confirm if it is AD_TRIG_IF or ADCIF, before clearing the respective bit.

22.3.3 ADC_MASK (0x4036)

Table 22-3 ADC_MASK (0x4036)

Bit	7	6	5	4	3	2	1	0
Name	ADC_MASK[7]	ADC_MASK[6]	ADC_MASK[5]	ADC_MASK[4]	ADC_MASK[3]	ADC_MASK[2]	ADC_MASK[1]	ADC_MASK[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]	ADC_MASK[7]	ADC Seventh channel enable
[6]	ADC_MASK[6]	ADC Sixth channel enable
[5]	ADC_MASK[5]	ADC Fifth channel enable
[4]	ADC_MASK[4]	ADC Fourth channel enable
[3]	ADC_MASK[3]	ADC Third channel enable
[2]	ADC_MASK[2]	ADC Second channel enable
[1]	ADC_MASK[1]	ADC First channel enable
[0]	ADC_MASK[0]	ADC Zeroth channel enable

22.3.4 ADC0_DR={ADC0_DRH,ADC0_DRL} (0x4038~0x4039)

Table 22-4 ADC0_DR={ADC0_DRH,ADC0_DRL}（0x4038~0x4039）

ADC0_DRH（0x4038）

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

ADC0_DRL（0x4039）

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/W	R/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]	RSV	Reserved.
[11:8]	DH	In 12bit mode, [11:8] bits are the highest four bits of ADC output data for the zeroth channel. In 10bit mode, [9:8] bits are the highest two bits of ADC output data for the zeroth channel.
[7:0]	DL	The lowest eight bits data for the zeroth channel after completion of ADC conversion.

22.3.5 ADC1_DR={ADC1_DRH,ADC1_DRL} (0x403A~0x403B)

Table 22-5 ADC1_DR={ADC1_DRH，ADC1_DRL}（0x403A~0x403B）

ADC1_DRH (0x403A)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

ADC1_DRL (0x403B)

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/W	R/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]	RSV	Reserved.
[11:8]	DH	In 12bit mode, [11:8] bits are the highest four bits of ADC output data for the zeroth channel. In 10bit mode, [9:8] bits are the highest two bits of ADC output data for the zeroth channel.
[7:0]	DL	The lowest eight bits data for the first channel after completion of ADC conversion.

22.3.6 ADC2_DR={ADC2_DRH,ADC2_DRL} (0x403C~0x403D)

Table 22-6 Table 22-1 ADC2_DR={ADC2_DRH, ADC2_DRL}（0x403C~0x403D）

ADC2_DRH (0x403C)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

ADC2_DRL (0x403D)

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/W	R/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]	RSV	Reserved.
[11:8]	DH	In 12bit mode, [11:8] bits are the highest four bits of ADC output data for the second channel. In 10bit mode, [9:8] bits are the highest two bits of ADC output data for the second channel.
[7:0]	DL	The lowest eight bits data for the second channel after completion of ADC conversion.

22.3.7 ADC3_DR={ADC3_DRH,ADC3_DRL} (0x403E~0x403F)

Table 22-7 ADC3_DR={ADC3_DRH, ADC3_DRL}（0x403E~0x403F）

ADC3_DRH (0x403E)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

ADC3_DRL (0x403F)

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/W	R/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]	RSV	Reserved
[11:8]	DH	In 12bit mode, [11:8] bits are the highest four bits of ADC output data for the third channel. In 10bit mode, [9:8] bits are the highest two bits of ADC output data for the third channel.
[7:0]	DL	The lowest eight bits data for the third channel after completion of ADC conversion.

22.3.8 ADC4_DR={ADC4_DRH,ADC4_DRL} (0x4040~0x4041)

Table 22-8 ADC4_DR={ADC4_DRH，ADC4_DRL}（0x4040~0x4041）

ADC4_DRH (0x4040)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

ADC4_DRL (0x4041)

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/W	R/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]	RSV	Reserved
[11:8]	DH	In 12bit mode, [11:8] bits are the highest four bits of ADC output data for the fourth channel. In 10bit mode, [9:8] bits are the highest two bits of ADC output data for the fourth channel.
[7:0]	DL	The lowest eight bits data for the fourth channel after completion of ADC conversion.

22.3.9 ADC5_DR={ADC5_DRH,ADC5_DRL} (0x4042~0x4043)

Table 22-9 ADC5_DR={ADC5_DRH, ADC5_DRL}（0x4042~0x4043）

ADC5_DRH (0x4042)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

ADC5_DRL (0x4043)

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/W	R/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]	RSV	Reserved
[11:8]	DH	In 12bit mode, [11:8] bits are the highest four bits of ADC output data for the fifth channel. In 10bit mode, [9:8] bits are the highest two bits of ADC output data for the fifth channel.
[7:0]	DL	The lowest eight bits data for the fifth channel after completion of ADC conversion.

22.3.10 ADC6_DR={ADC6_DRH,ADC6_DRL} (0x4044~0x4045)

Table 22-10 ADC6_DR={ADC6_DRH, ADC6_DRL}（0x4044~0x4045）

ADC6_DRH (0x4044)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

ADC6_DRL (0x4045)

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/W	R/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]	RSV	Reserved
[11:8]	DH	In 12bit mode, [11:8] bits are the highest four bits of ADC output data for the sixth channel. In 10bit mode, [9:8] bits are the highest two bits of ADC output data for the sixth channel.
[7:0]	DL	The lowest eight bits data for the sixth channel after completion of ADC conversion.

22.3.11 ADC7_DR={ADC7_DRH,ADC7_DRL} (0x4046~0x4047)

Table 22-11 ADC7_DR={ADC7_DRH，ADC7_DRL}（0x4046~0x4047）

ADC7_DRH (0x4046)

Bit	7:4	3	2	1	0
Name	RSV	DH[3]	DH[2]	DH[1]	DH[0]
Type	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0

ADC7_DRL (0x4047)

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/W	R/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]	RSV	Reserved
[11:8]	DH	In 12bit mode, [11:8] bits are the highest four bits of ADC output data for the seventh channel. In 10bit mode, [9:8] bits are the highest two bits of ADC output data for the seventh channel.
[7:0]	DL	The lowest eight bits data for the seventh channel after completion of ADC conversion.

22.3.12 ADC_SCYC0/ADC_SCYC1 (0x4048~0x4049)

Table 22-12 ADC_SCYC0/ADC_SCYC1（0x4048~0x4049）

ADC_SCYC0 (0x4048)

Bit	7:4	3:0
Name	ADC23_SCYC[3:0]	ADC01_SCYC[3:0]
Type	R/W	R/W
Reset	0011	0011

Bit	Name	Function
[7:4]	ADC23_SCYC [3:0]	ADC sample time set for channel 2 and channel 3, ADC23_SCYC [2:0] bits define the number of ADC12CLK (12MHz clock) cycles in the sampling period. When ADC23_SCYC[3] = 0, ADC23_SCYC[2:0] ADC12CLK cycles When ADC23_SCYC [3] =1, (ADC23_SCYC [2:0]*8 + 7) ADC12CLK cycles.
[3:0]	ADC01_SCYC [3:0]	ADC sample time set for channel 0 and channel 1, ADC01_SCYC [2:0] bits define the number of ADC12CLK (12MHz clock) cycles in the sampling period. When ADC01_SCYC[3] = 0, ADC01_SCYC[2:0] ADC12CLK cycles When ADC01_SCYC [3] =1, (ADC01_SCYC [2:0]*8 + 7) ADC12CLK cycles.

ADC_SCYC1 (0x4049)

Bit	7:4	3:0
Name	ADC67_SCYC[3:0]	ADC45_SCYC[3:0]
Type	R/W	R/W
Reset	0011	0011

Bit	Name	Function
[7:4]	ADC67_SCYC[3:0]	ADC sample time set for channel 6 and channel 7, ADC67_SCYC [2:0] bits define the number of ADC12CLK (12MHz clock) cycles in the sampling period. When ADC67_SCYC[3] = 0, ADC67_SCYC[2:0] ADC12CLK cycles When ADC67_SCYC [3] =1, (ADC67_SCYC [2:0]*8 + 7) ADC12CLK cycles.
[3:0]	ADC45_SCYC[3:0]	ADC sample time set for channel 4 and channel 5, ADC45_SCYC [2:0] bits define the number of ADC12CLK (12MHz clock) cycles in the sampling period. When ADC45_SCYC[3] = 0, ADC45_SCYC[2:0] ADC12CLK cycles When ADC45_SCYC [3] =1, (ADC45_SCYC [2:0]*8 + 7) ADC12CLK cycles.

23 reference voltage

23.1 Reference voltage block operation instructions

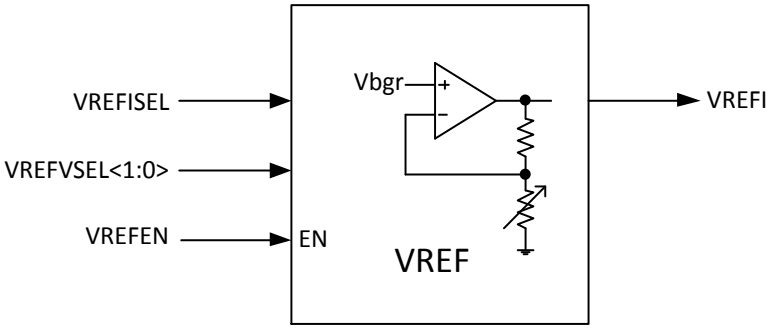


Figure 23-1 VREF block port declarations

FU68XX has a reference voltage block which provides the internal reference voltage for ADC.The port declaration of this block is shown in Figure 23-1.

For proper function of the block, the following bits in VREF_CR register must be set: VREFEN 、VREFISEL and VREFVSEL . For more detail, please refer to Table 23-1.

23.2 REGISTER OF VREF BLOCK

23.2.1 VREF_CR (XRAM: 0x404F)

Table 23-1 VREF_CR（0x404F）

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

Bit	Name	Function
7：6	VREFVSEL	Output reference voltage select 00，output 3V； 01，output 4V； 10，output 4.5V 11，output 5V (it is only valid under the condition that the supply voltage VDD5 is greater than 5.3V in low-voltage mode(VCC_MODE=1) or in dual high voltage mode)
5	VREFISEL	VREF bias current select 0, Normal mode 1, Half current mode,for saving the power
4	VREFEN	VREF enable, used to provide the internal reference voltage for ADC 0, Disabled 1, Enabled
3:2	RSV	Reserved.
1	VHALFM	VHALF block operation mode select 0, Select VDD/2 as the ouput of VHALF block 1, Select VREF/2 as the ouput of VHALF block
0	VHALFEN	VHALF block enable, default vaule:0 0, Disabled 1, Enabled

24 VHALF voltage reference

24.1 VHALF block operation instructions

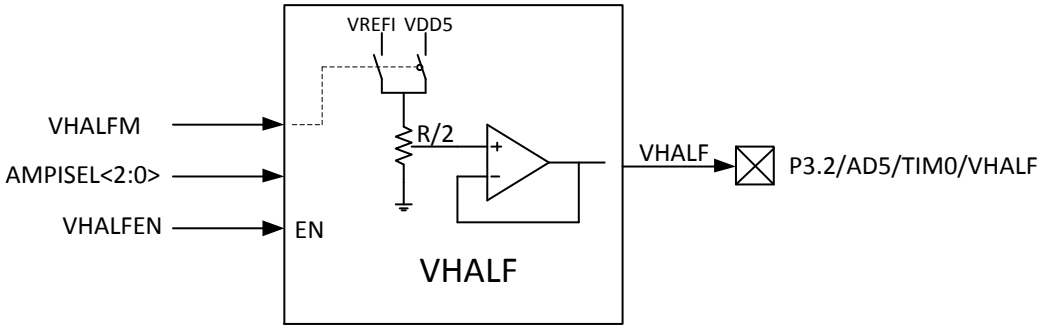


Figure 24-1 VHALF block port declarations

FU68XX has a VHALF block which generates a reference voltage. The port declaration of this block is shown in Figure 24-1.

For proper functioning of the block, the following bits in VREF_CR register must be set: VHALFM and VHALFEN. For more detail, please refer to Table 24-1.

24.2 VHALF register

24.2.1 VREF_CR (XRAM: 0x404F)

Table 24-1 related register of VHALF block (I)

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

Bit	Name	Function
7:2		Please refer to the table 23-1 to get the other six bits
1	VHALFM	VHALF block operation mode select 0, Select VDD/2 as the ouput of VHALF block 1, Select VREF/2 as the ouput of VHALF block
0	VHALFEN	VHALF block enable, default vaule: 0 0, Disabled 1, Enabled

24.2.2 AMP_CR (XRAM: 0x404E)

Table 24-2 related register of VHALF block (II)

Bit	7	6	5	4	3	2	1	0
Name	RSV	AMPISEL			AMP3EN	AMP2EN	AMP1EN	AMP0EN
Type	R	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>7，3：0</td><td></td><td>Please refer to the table 23-1 for further information</td></tr>
<tr><td>[6:4]</td><td>AMPISEL</td><td>The bias current select of AMP3~0 and VHALF. On 2’s complement, the greater the value, the greater the current.
<table>
<tr><th>AMPISEL</th><th>CURRENT SET</th></tr>
<tr><td>100B</td><td>1X</td></tr>
<tr><td>101B</td><td>2X</td></tr>
<tr><td>110B</td><td>3X</td></tr>
<tr><td>111B</td><td>4X</td></tr>
<tr><td>000B</td><td>5X</td></tr>
<tr><td>001</td><td>6X</td></tr>
<tr><td>010B</td><td>7X</td></tr>
<tr><td>011B</td><td>8X</td></tr>
</table></td></tr>
</table>

25 operation amplifier

FU68XX has four integrated high-speed independentoperation amplifiers, namely AMP0、AMP1、AMP2、AMP3,respectively. Every operation amplifiers has their own independent enable port.

25.1 Amplifier operation instructions

25.1.1 (AMP0) bus current operation amplifier

The AMP0 ports connected with pad is shown in the Figure 25-1.

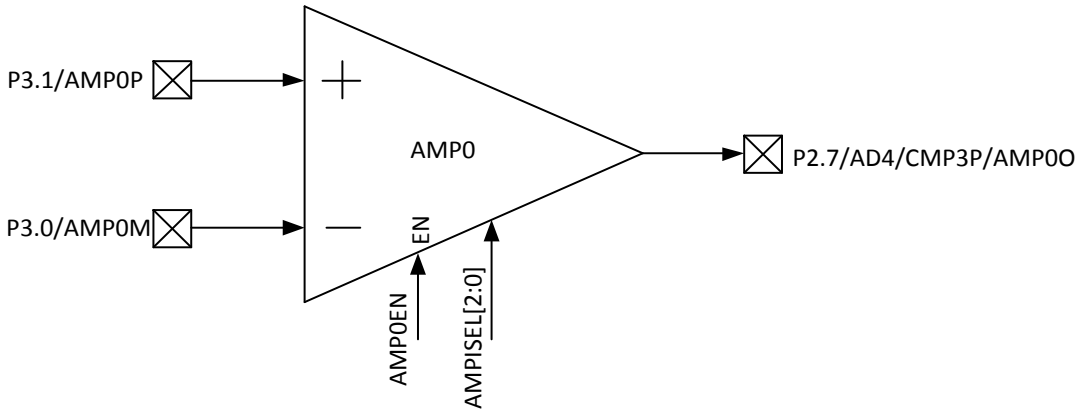


Figure 25-1 The AMP0 ports connected with pad

For proper functioning of the block, the AMP0EN bit must be set to 1.

The AMP0 ports are shown in the Figure 25-1. AMPISEL control the bias current of the four operation amplifiers AMP3~0. Set the P2_AN[7]=1, P3_AN[1:0]=11b to change the GPIO port of P2.7，P3.0，P3.1 to analog mode before enabling AMP0.

25.1.2 Phase current operation amplifier (AMP1/AMP2/AMP3)

25.1.2.1 AMP1

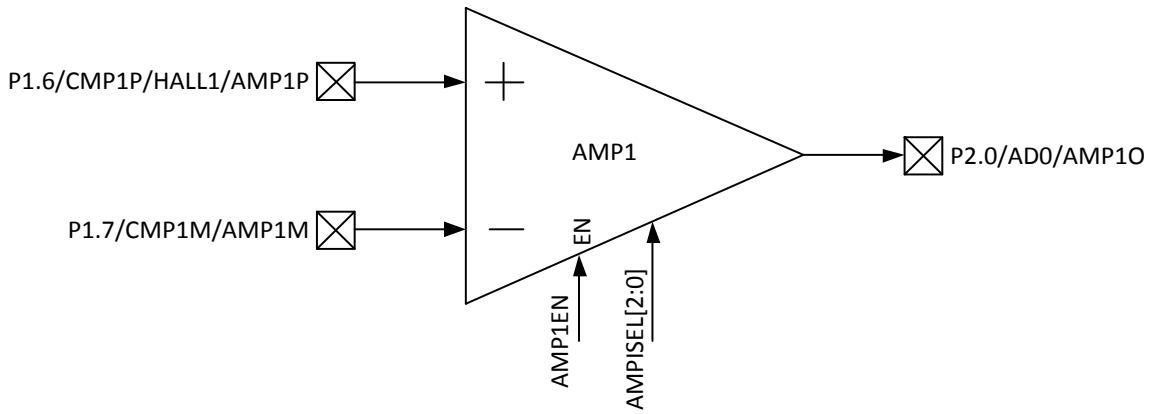


Figure 25-2 The AMP1 ports connected with pad

For proper functioning of the block, the AMP1EN bit must be set to 1.

The AMP0 ports are shown in the Figure 25-2. AMPISEL control the bias current of the four operation amplifiers AMP3~0. Set the P1_AN[7:6]=11b, P2_AN[0]=1b to change the GPIO port of P1.6，P1.7，P2.0 to analog mode before enabling AMP1.

25.1.2.2 AMP2

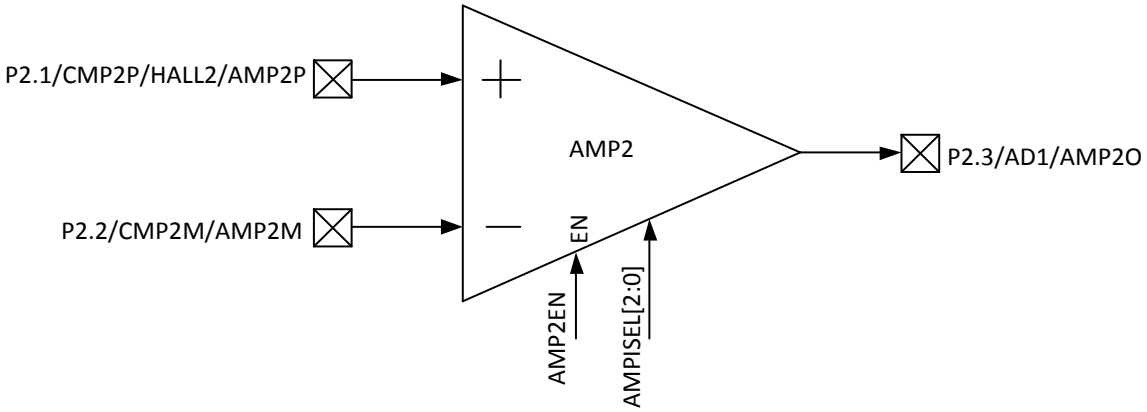


Figure 25-3 The AMP2 ports connected with pad

For proper functioning of the block, the AMP2EN bit must be set to 1.

The AMP2 ports are shown in the Figure 25-3. AMPISEL control the bias current of the four operation amplifiers AMP3~0. Set the P2_AN[3:1]=111b to change the GPIO port of P2.1，P2.2，P2.3 to analog mode before enabling AMP2.

25.1.2.3 AMP3

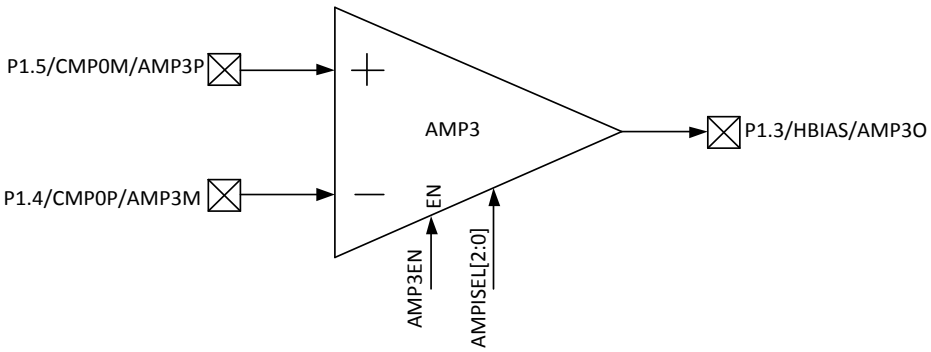


Figure 25-4 The AMP3 ports connected with pad

For proper functioning of the block, the AMP3EN bit must be set to 1.

The AMP3 ports are shown in the Figure 25-4. AMPISEL control the bias current of the four operation amplifiers AMP3~0. Set the P1_AN[5:3]=111b and P1_OE.3=0 to change the GPIO port of P1.3，P1.4，P1.5 to analog mode before enabling AMP3.

25.2 Register of operation amplifier

25.2.1 AMP_CR (0x404E)

Table 25-1 AMP_CR（0x404E）

Bit	7	6	5	4	3	2	1	0
Name	RSV	AMPISEL			AMP3EN	AMP2EN	AMP1EN	AMP0EN
Type	R	R/W			R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	RSV	Reserved.
[6:4]	AMPISEL	The bias current select of AMP3~0 and VHALF. On 2's complement coding, the greater the value, the greater the current. AMPISEL / CURRENT SET 100B / 1X 101B / 2X 110B / 3X 111B / 4X 000B / 5X 001B / 6X 010B / 7X 011B / 8X
[3]	AMP3EN	AMP3 enabled
[2]	AMP2EN	AMP2 enabled
[1]	AMP1EN	AMP1 enabled
[0]	AMP0EN	AMP0 enabled

26 COMPARATOR

26.1 Comparator operation instructions

26.1.1 Comparator CMP3

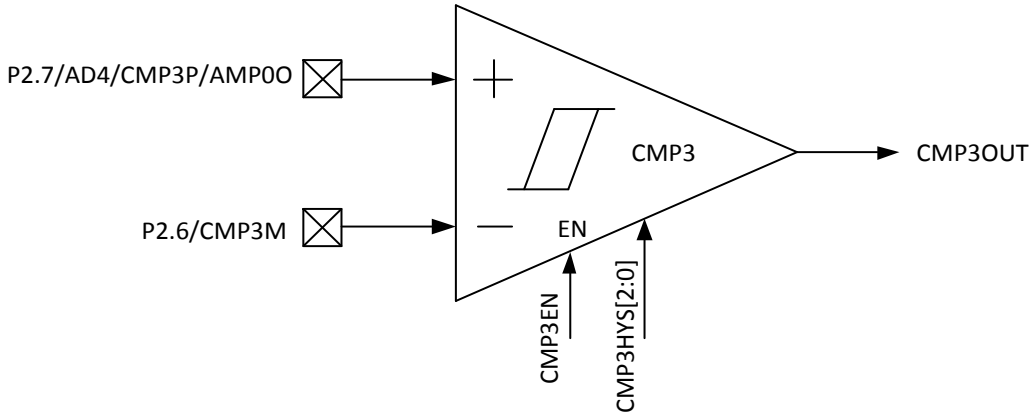


Figure 26-1 CMP3 ports connectedwith related signals

For proper function of CMP3 block, the CMP3EN bit in CMP_CR2 register must be set to 1.

The figure of CMP3 ports are shown in the Figure 26-1. CMP3EN control the hysteresis voltage of CMP3.

26.1.2 BUS current protection

For protection of the chip and motor, BUS current protection wills shutdown the outputs to motor. Set the MCLREN in EVT_FILT register to 1 will produce the protection event. If MCLREN is set to zero, it will not shut down but will trigger an interrupt.

Bus current protection event can be produced by CMP3 when setting the EFSRC in EVT_FILT register, or produced by an external interruption INT0 (P0.0). Assuming that the chip connects to the IPM motor with the FALUT signal connected to P0.0, BUS current protection event will be produced by external interruption INT0 through setting the EFSRC in EVT_FILT register. In this case, the protection interruption is external interrupt INT0. Protection signal can also be produced by comparing the sampling voltage of the BUS, and protection interrupt triggered by CMP3.

The input signal of BUS current protection event can enable the Filtering function by setting the RFEN=1 of EVT_FILT register. Filter width can be set through EFDIV in EVT_FILT register. After enabling the Filtering function, the signal will delay about 4~5/8~9/16~17/24~25 clcok cycles.

26.1.3 Compartor CMP0/CMP1/CMP2

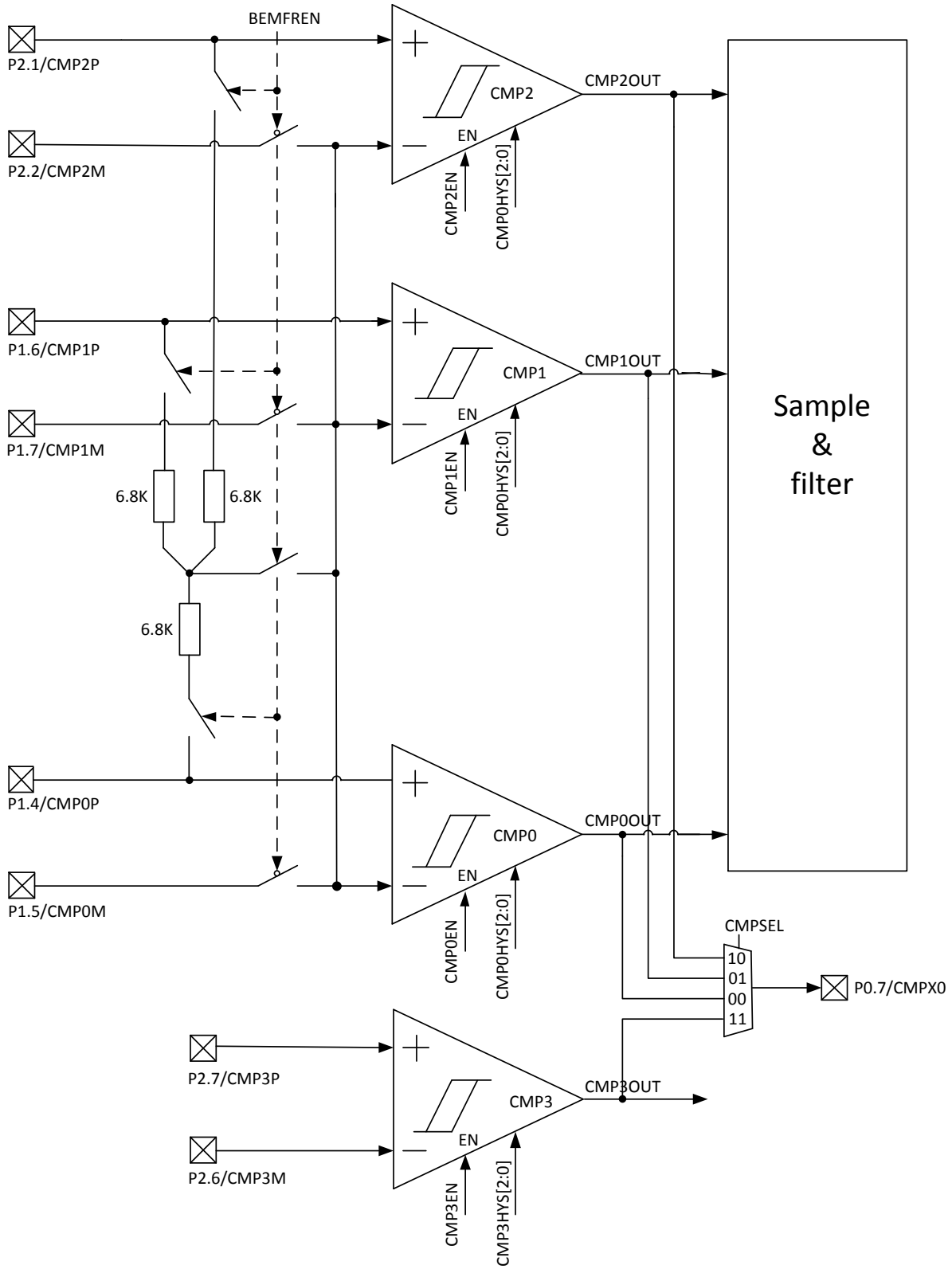


Figure 26-2 comparaor schematic diagram

Besides compartor CMP3, the other three compartors CMP0/CMP1/CMP2 are BEMF/HALL comparators used for postion detection of the motor, as shown in Figure 26-2. In this figure, the dotted line represents a control signal, and the circled switch with a small circle above indicates

that the switch will be turned on when the control signal is low. Conversely, the uncircled switch indicates that the switch will be turned on when the control signal is high.

Under HALL mode, BEMFREN must be set to 0. In this case, the positive input port and negative input port of the three compartors are connected to the pads.

Under BEMF mode, the BEMFREN must be set to 1. In this case, the negative input port of the three comparators connected together to the resistors, and the positive input port are connected to backEMF of UVW phases.

The output signals of the comparator CMP0/CMP1/CMP2 send to the TIMER1 through the sample & filter block.

The comparator CMP0/CMP1/CMP2 is enabled through CMP0EN/CMP1EN/CMP2EN, by the respective bit to logic 1. But hysteresis voltage is controlled by CMP0HYS.

The output of the four compartors are connected connect to a multiplexer. The multiplexer input is selected by setting CMPSEL bit in CMP_CR2 register and its output will be connected the specific pad (P0.7). The Corresponding pad is shown in the figure.

26.1.4 Compartor register

26.1.5 CMP_CR0 (0xD5)

Table 26-1 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 CMPOIM
[5:4]	CMP2IM	Comparator CMP2 interrupt mode Refer to the description of CMPOIM
[3:2]	CMP1IM	Comparator CMP1 interrupt mode Refer to the description of CMPOIM
[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 edges

26.1.6 CMP_CR1 (0xD6)

Table 26-2 CMP_CR1（0xD6）

Bit	7	6	5	4	3	2	1	0
Name	RSV	BEMFREN	CMP3HYS			CMP0HYS		
Type	R	R/W	R/W			R/W		
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function	
[7]	RSV	Reserved	
[6]	BEMFREN	BEMF resister enables. The resistance is 6.8KΩ。 0: Disabled. Under HALL mode this bit must be 0. 1: Enabled. P1.5、P1.7 、P2.2 can used for other purposes.	
[5:3]	CMP3HYS	CMP3 hysteresis voltage select	
		CMP3HYS	hysteresis voltage
		000	no hysteresis
		001	±2.5mV
		010	-5mV
		011	+5mV
		100	±5mV
		101	-7.5mV
		110	+7.5mV
		111	±7.5mV
[2:0]	CMP0HYS	CMP2~0 hysteresis voltage select	
		CMP0HYS	hysteresis voltage
		000	no hysteresis
		001	±2.5mV
		010	-5mV
		011	+5mV
		100	±5mV
		101	-7.5mV
		110	+7.5mV
		111	±7.5mV

26.1.7 CMP_CR2 (0xDA)

Table 26-3 CMP_CR2（0xDA）

Bit	7	6	5	4	3	2	1	0
Name	CMP3EN	CMP2EN	CMP1EN	CMP0EN	CMPSAME	CMPSEL		CMPOE
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]	CMP3EN	Comparator CMP3 enable 0: Disabled 1: Enabled
[6]	CMP2EN	Comparator CMP2 enable 0: Disabled 1: Enabled
[5]	CMP1EN	Comparator CMP1 enable 0: Disabled 1: Enabled
[4]	CMP0EN	Comparator CMP0 enable 0: Disabled 1: Enabled
[3]	CMPSAME	Comparator CMP0,CMP1,CMP2 sampling function enable 0: Disabled 1: Enabled
[2:1]	CMPSEL	Comparator output seclect This bit is used to select one of the comparators outputs 00: select CMP0 output to the port 01: select CMP1 output to the port 10: select CMP2 output to the port 11: select CMP3 output to the port
[0]	CMPOE	Comparator output enable This bit is used to enable the selected comparator outputs 0: disabled 1: enabled

26.1.8 CMP_SR (0xD7)

Table 26-4 CMP_SR (0xD7)

Bit	7	6	5	4	3	2	1	0
Name	CMP3OUT	CMP2OUT	CMP1OUT	CMP0OUT	CMP3INTR	CMP2INTR	CMP1INTR	CMP0INTR
Type	R	R	R	R	R/W0	R/W0	R/W0	R/W0
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	CMP3OUT	Output of comparator CMP3 0: output 0 1: output 1
[6]	CMP2OUT	Output of comparator CMP2 0: output 0 1: output 1
[5]	CMP1OUT	Output of comparator CMP1 0: output 0 1: output 1
[4]	CMP0OUT	Output of comparator CMP0 0: output 0 1: output 1
[3]	CMP3INTR	CMP3 interrupt flag This bit is the CMP3 interrupt flag. It will set to 1 by handware and reset to 0 by software.
[2]	CMP2INTR	CMP2 interrupt flag This bit is the CMP2 interrupt flag. It will set to 1 by handware and reset to 0 by software.
[1]	CMP1INTR	CMP1 interrupt flag This bit is the CMP1 interrupt flag. It will set to 1 by handware and reset to 0 by software.
[0]	CMP0INTR	CMP0 interrupt flag This bit is the CMP3 interrupt flag. It will set to 1 by handware and reset to 0 by software.

26.1.9 EVT_FILT (0xD9)

Table 26-5 EVT_FILT(0xD9)

Bit	7	6	5	4	3	2	1	0
Name	TSDEN	TSDADJ		MCLREN	EFSRC	EFEN	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: disabled 1: enabled.
[6:5]	TSDADJ	Temperature sensor detection adjustment 2'b00: 105℃。 2'b01: 120℃。 2'b10: 135℃。 2'b11: 150℃。
[4]	MCLREN	MOE signal hardware cleared enable Bus current overshoot events in the MOE cleared by hardware when this bit set to enable. 0: Disabled 1: Enabled
[3]	EFSRC	The interrupt source of BUS current protection event 0: Comparator CMP3 1: External interrupt TIN0(P0.0), external interrupt TIN0
[2]	EFEN	This bit is used to enable the filter of the BUS current protection event. 0: Disabled 1: Enabled
[1:0]	EFDIV	This bit is used to set the filter width of the BUS current protection event. 00 ： 4 system clock cycles 01 ： 8 system clock cycles 10 ： 16 system clock cycles 11 ： 24 system clock cycles

27 DIRVER

The output of DRIVER of FU68XX differs when the type of IC is different, the output of FU6811 is Gate Driver, the FU6831 is 3P3N Predriver while the FU6818 is 6N Predriver.

27.1 3P3N Predriver mode(only suitable for FU6831)

27.1.1 the function block diagram and configuration of 3P3N Predriver

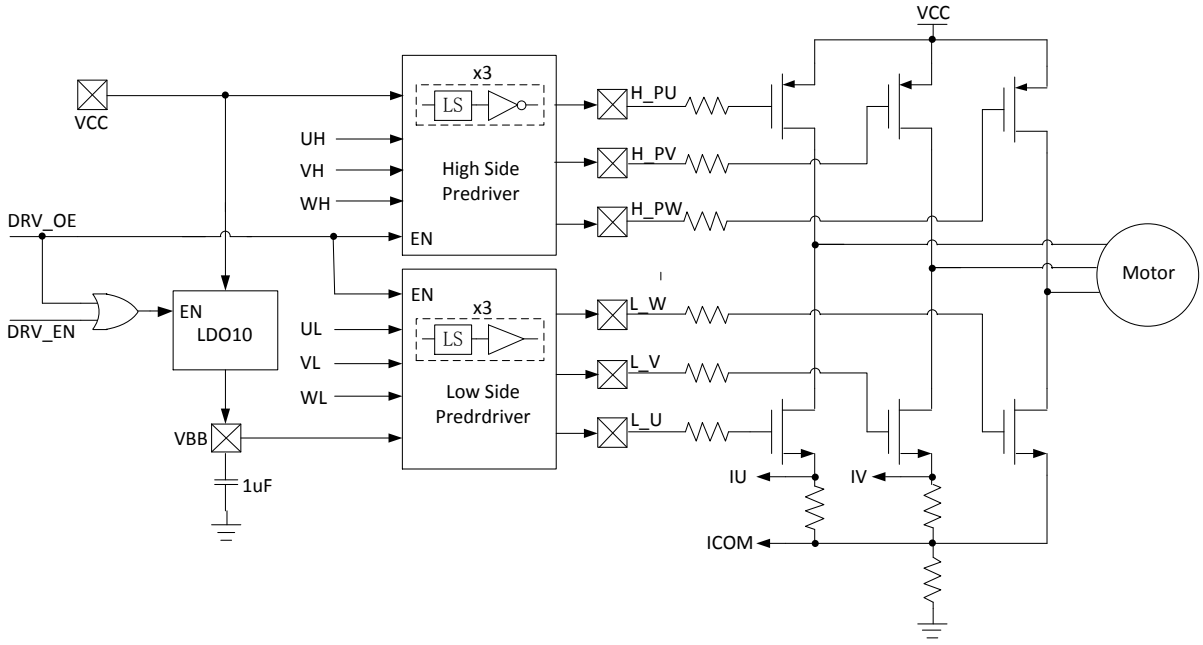


Figure 27-1 the schematic diagram of 3P3N Predriver mode

In the 3P3N Predriver mode, VCC is a input IO, where a 1-4.7uF off-chip capacitor must be connected from VBB to ground. The H_PU/H_PV/H_PW/L_U/L_V/L_W is the output IO of Predriver, H_PU/H_PV/H_PW show inverse logic with respect to internal signal UH/VH/WH. PDRVEN is the enable pin of driver module, DRVOE is driver's output enable in the 3P3N mode as illustrated in the Figure 27-1.

The PDRVEN of register DRV_CTL should be configured to logic high, to enable LDO10, output the voltage of VBB. As it will take some time for LDO to be stable, a 1mS is needed so that the VBB is steady when PDRVEN is high logical level. The register should be configured: DRVOE=1 to enable the Predriver's output, UH/VH/WH will be inverted and connected to H_PU/H_PV/H_PW which are set to drive the gate of PMOS separately. UL/VL/WL are connected to L_U/L_V/L_W to drive the gate of NMOS separately.

27.2 Gate Driver mode (only suitable for FU6811)

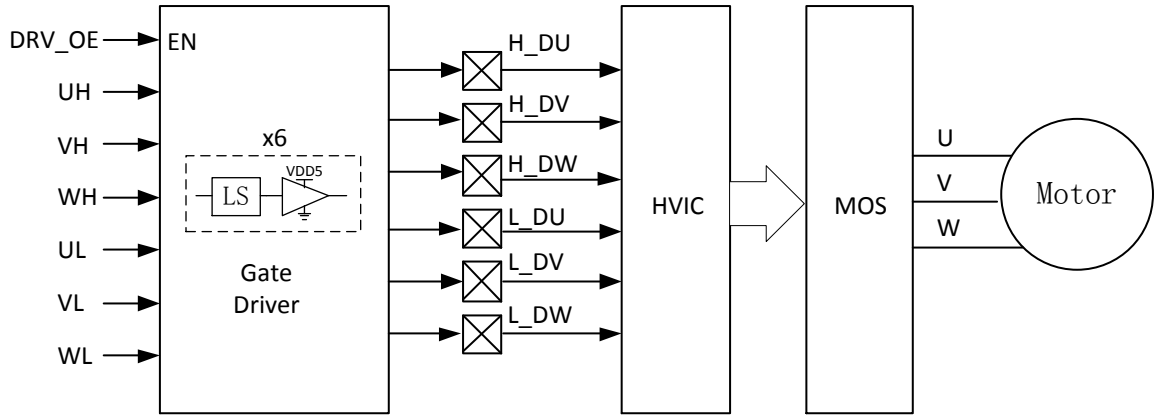


Figure 27-2 schematic diagram of Gate Driver mode

Gate Driver is illustrated in Figure 27-2. In Gate Driver mode, PDRVEN is enable signal of Gate Driver. Unlike the 3P3N Predriver, the outputs of Gate Driver connected to the input of HVIC, while the outputs of HVIC drive the MOSFETs gates.

27.3 6N Predriver mode (only suitable for FU6818)

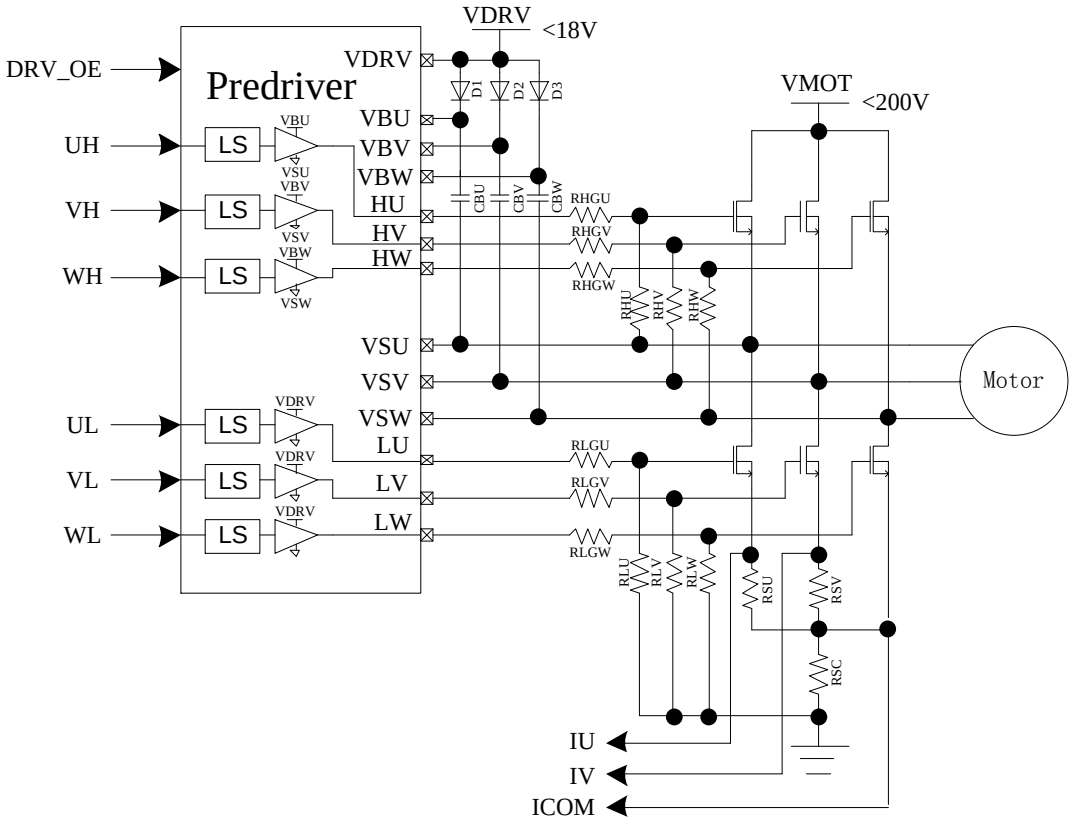


Figure 27-3 schematic diagram of 6N Predriver mode

6N Predriver is illustrated in Figure 27-3. In the 6N Predriver mode, DRVOE is the enable signal of Predriver, the outputs of predriver are connected to six NMOS, which in turn drive the U/V/W phase of motor.

27.4 Register of DRIVER

27.4.1 DRV_CTL (0x404D)

Table 27-1 DRV_CTL(0x404D)

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

Bit	Name	Function
[7:4]	RSV	RSV.
[3:2]	OCS	The source of the OC(U/V/W)H and OC(U/V/W)L's output data 0X：TIMER1 output（T1_OC0，T1_OC1，T1_OC2，T1_OC3，T1_OC4，T1_OC5） 10：TIMER0 output（T0_0C1，T0_OC1N，T0_0C2，T0_OC2N，T0_0C3，T0_OC3N） 11：FOC output（FOC_OC1H，FOC_OC1L，FOC_OC2H，FOC_OC2L，FOC_OC3H，FOC_OC3L）
[1]	PDRVEN	Enable the Driver by setting a logic high. This bit is only valid for Predriver mode. Predriver mode, PDRVEN=1 will enable VBB. A time lapse of 1ms is required for it to stable. This bit is not required under Gate Driver mode.
[0]	DRVOE	The enable of Driver's output. Under Predriver mode, set PDRVEN=1; wait 1mS, then set the DRVOE=1 in order to let output pin level to be stable 0, Disable 1, Enable

28 Supply

28.1 LDO

28.1.1 LDO Operation Instructions

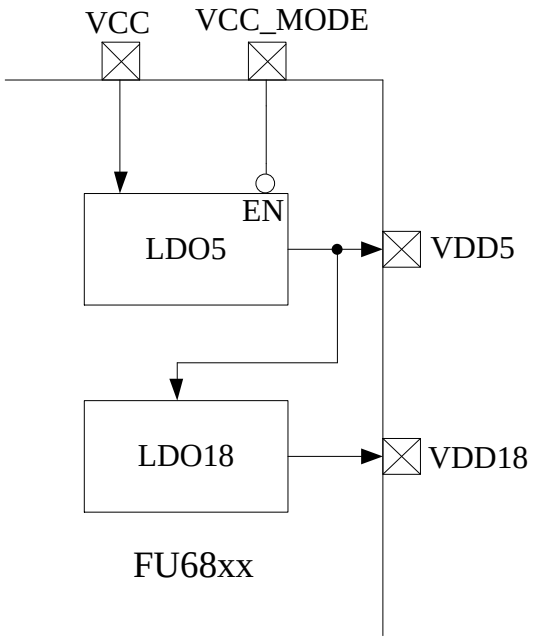


Figure 28-1 Supply Block Diagram

Refer Figure 28-1, it lists the supply and LDO relative pins. MCU builded one 5V LDO module and one 1.8V LDO module.The 5V LDO, LDO5, is enabled by VCC_MODE pin if connecting to ground. VDD5 can ouput 5V supply generated by internal LDO5 or input 5V from external source. VDD18 is for digital core supply.

FU6831:

Single Supply HV Mode (VCC_MODE=0). VCC: 5~24V. Refer Figure 28-2.

Dual Supply Mode (VCC_MODE=1), VCC≥VDD5. VCC= 5~36V, VDD5=5V. Refer Figure 28-3.

FU6811:

Single Supply HV Mode (VCC_MODE=0). VCC: 5~24V. Refer Figure 28-2.

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

Refer Figure 28-3.

Single Supply LV Mode (VCC_MODE=1). VCC=VDD5= 3~5.5V. Refer Figure 28-4.

FU6818:

Mode1: VCC_MODE=0, VCC= 5~24V，VDRV=7~18V

Mode2: VCC_MODE=1, VCC=VDD5=3~5.5V，VDRV=7~18V

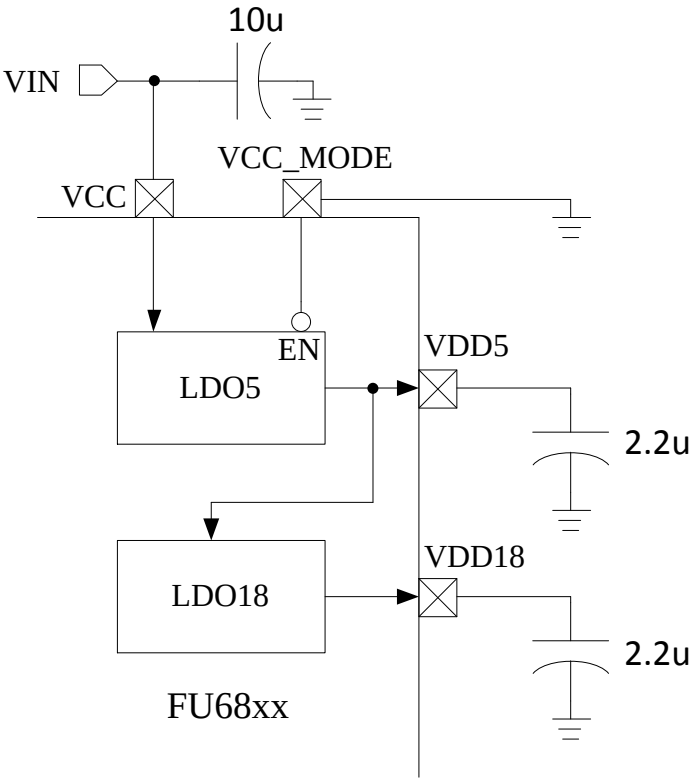


Figure 28-2 Single Supply HV Mode connection

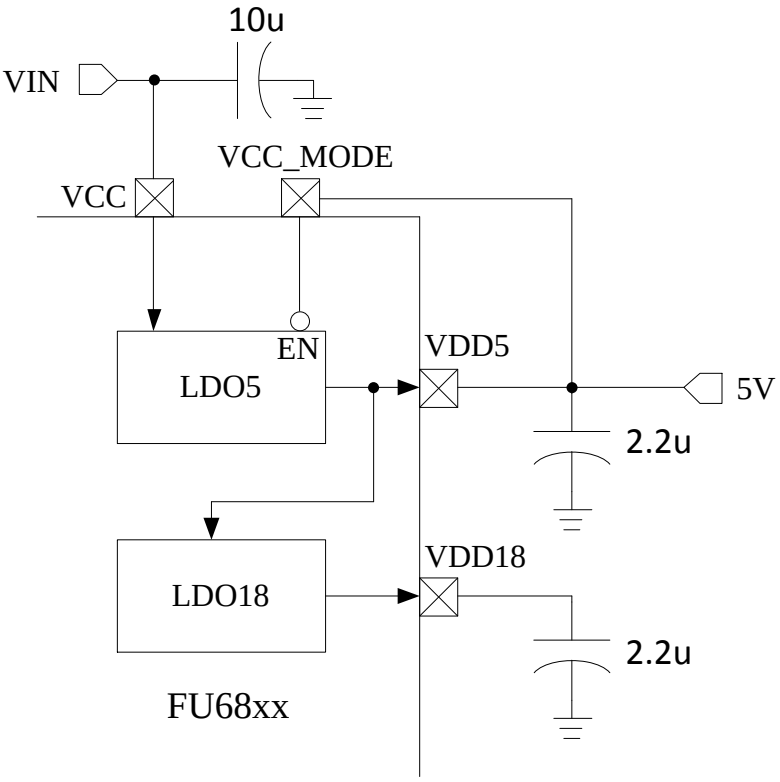


Figure 28-3 Dual Supply Mode connection

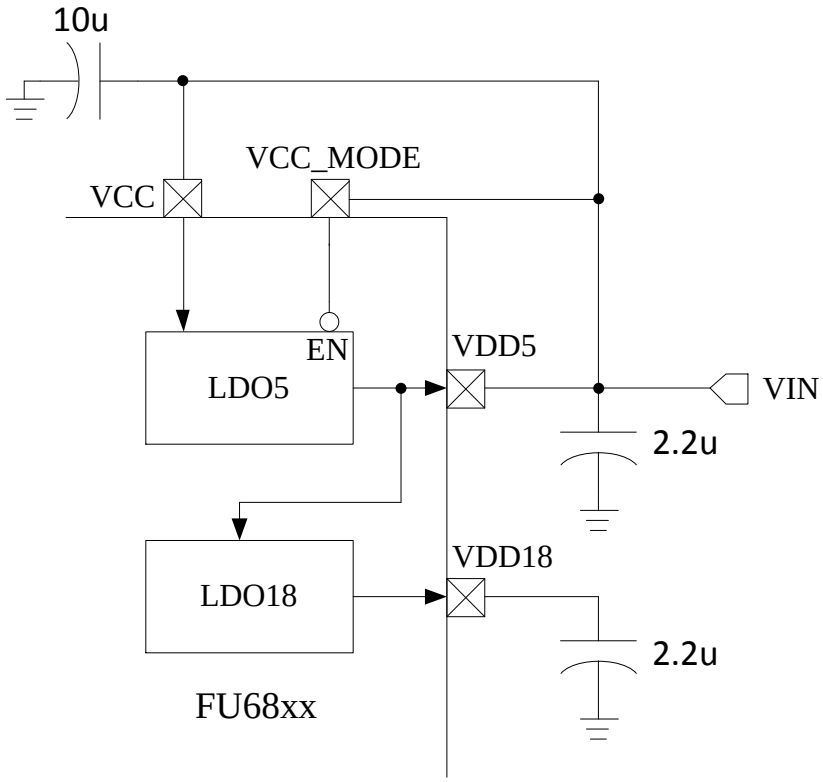


Figure 28-4 Single Supply LV Mode connection

28.2 Low Voltage Dection

28.2.1 Low Voltage Dection Manual

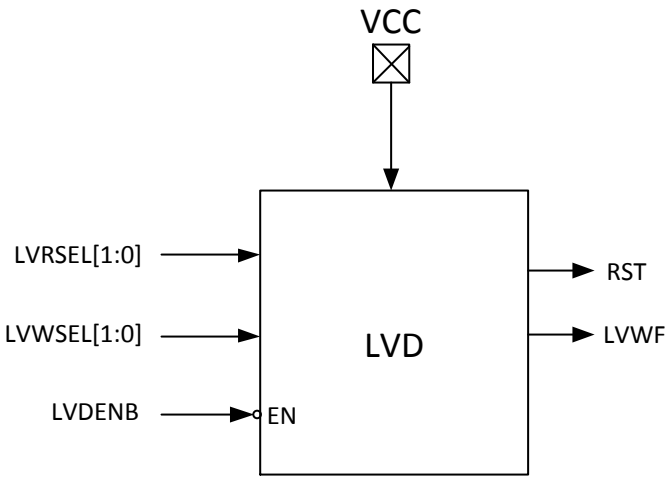


Figure 28-5 Low Voltage Dection Diagram

Clear the LVDENB to enable the low voltage detection function.

28.2.2 CCFG2:RST_MOD (0x401D)

Table 28-1 Register 1for control low voltage dection function

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 VDD5's voltage. 00: The minimum release reset voltage is 2.8V. 01: The minimum release reset voltage is 3.0V 10: The minimum release reset voltage is 3.5V 11: The minimum release reset voltage is 3.8V
5：2		Reserved
1：0	LVWSEL	Warning voltage selection. When VCC's voltage lower than the set value, the hardware set LVWIF. MCU will enter interrupt if LVWIE is "1". 00: The warning voltage threshold on VCC is 7V. 01: The warning voltage threshold on VCC is 8V. 10: The warning voltage threshold on VCC is 9V. 11: The warning voltage threshold on VCC is 10V.

28.2.3 CCFG1:CK_RST_CFG (0x401E)

Table 28-2 Register 2 for control low voltage dection function

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

Bit	Name	Function
7	LVDENB	Low voltage detection enable set bit. 0: Enable low voltage detection. When VDD5 is lower than the threshold, which set by LVRSEL, the MCU will be reset. 1: Disable low voltage detection function. Note. There are two features under low voltage detection. One is low voltage reset with the other low voltage warning. For the low voltage reset, the mcu goes into reset state when VDD5 is lower than the threshold set by LVRSEL. For the low voltage warning, the mcu generate an interrupt flag when VCC is lower than the threshold set by LVWSEL. LVDENB is required to be cleared for both of the function.
6:0		Please refer Table 33-1 for other bits.

28.2.4 LVSR(0xDB)

Table 28-3 LVSR(0xDB)

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

Bit	Name	Function
[7:2]	RSV	Reserved
[1]	LVWF	This bit indicates VCC low than threshold. 0: VCC is higher than the threshold set by LVWSEL. 1: VCC is lower than the threshold set by LVWSEL.
[0]	LVWIF	Interrupt flag for VCC lower than threshold set by LVWSEL. This bit is set by hardware when VCC lower than threshold set by LVWSEL. This bit is cleared by by software. This bit is “0” when LVWIE is “0”. 0: No LVW event. 1: LVW event detected.

29 FLASH Program Interface

29.1 PSCTL：Program Software Control

Table 29-1 PSCTL（0x85）

Bit	7:5	4	3	2	1	0
Name	RSV	FLAVIO	FLAACT	RSV	FLAERS	FLAWEN
Type	R	R	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0

Bit	Name	Function
7:5	RSV	Reserved
4	FLAVIO	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 writing. 1: Write 1 to start FLASH operation (Include program/sector erase)
2	RSV	Reserved
1	FLAERS	Sector erase enable. When set FLAWEN=1 and FLAERS=1, sector erase will be valid. 0: Disable sector erase. 1: Enable sector erase.
0	FLAWEN	Program enable. (if FLAWEN=0,sector erase will be disabled) 0: Disable FLASH program. 1: Enable FLASH program.

29.2 FLKEY：FLASH program/erase KEY

Table 29-2 FLKEY（0x84）

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

Bit	Name	Function
7:0	FLKEY	Write : FLASH sector erase/program KEY Sequential writing 0x5A、0x1F to the register FLKEY will start the Flash Program Interface function. If the writing sequence or writing data is illegal, the function will get to be frozen, until system reset. After unlocking the key, any writing of PSCTL will lock the FLKEY again.
		Read: the lowest 2 bits are status of FLASH sector erase/program, highest 6 bits is reserved. 00: The key locked. 01 : 0x5A has written, waiting for writing of 0xF1. 11: The key is opened. 10: The key is frozen.

29.3 FLASH sector erase/program step

FLASH Sector erase operation step:

Step 1: Software writes register EA to 0 to disable main interrupt request. This is to protect the firmware code.

Step 2: Software writes 0x03 to register PSCTL.

Step 3: Software sequentially writes 0x5A, 0x1F to register FLKEY.

Step 4: Software use instruction “MOVX” to execute erase command to erase the sector, or the area that the user want to write the data.

Step 5: Software set register PSCTL.FLAACT to 1 to start the sector erase process, about 120mS later, the erase process will complete. At the same time FLKEY will be locked again.

For example, FLASH sector erase:

```
mov EA,#00h       ;  //Clear EA to disable all interrupt requests.(Stronglly recommend this
                     // step to avoid interrupt corrupting the FLASH firmware code.
mov PSCTL,#03h ;// Write 0x03 to register PSCTL, prepare for sector erase
mov FLKEY,#05Ah；// Write 0x5A to register FLKEY
mov FLKEY,#01Fh；// Write 0x01F to register FLKEY to open the KEY.
mov   DPH,#03h;     // Select which sector to erase, enter the high_address of the data, which
// is in this secor.
mov   DPL,#0FEh;    // Enter the low_address of the data,which is in this sector.
mov   A,#0FFh;
movx @DPTR,A;     // Movx any data to the address.
```

mov PSCTL,#08h; // Write 0x08 to start the sector erase process, when the process end the
// FLASH key locked again automalically.

Flash byte write operation step:

Step 0: Disable all interrupt requests.

Step 1: Write 0x01 to register PSCTL to enable FLASH data writing function.

Step 2: Consequentially write 0x5A、0x1F to register FLKEY to open FLASH KEY.

Step 3: Using instruction set ”MOVX” to input the data address, where the user wants to Write.

Step 4: Software set register PSCTL.FLAACT to 1 to execute data writing process. After this process, the FLASH key locked again.

Step 5: Check if the writing process is successful.

For example, writing 2 bytes data to FLASH:

```
mov EA,#00h     ;       // Disable all interrupt requests.
mov PSCTL,#01h；  // Write 0x01 to register PSCTL to enable the FLASH byte writing function
mov FLKEY,#05Ah；//
mov FLKEY,#01Fh；// Consequencially write 0x5A 、0x1F to register FLKEY to open FLASH
// key.
mov   DPH,#03h;     //
mov   DPL,#0F0h;   // Enter the FLASH address of the byte, the space where user wants to write.
mov   A,#0Fh;          //
movx @DPTR,A;       // movx data “0x0F” to the FLASH address space “0x03F0”.
mov PSCTL,#08h;   // Write 0x08 to register PSCTL to start the writing process.  After the
// process, FLASH key will lock again automatically.
// First byte data writing is end, start another byte.
mov PSCTL,#01h; // Write 0x01 to register PSCTL to enable the FLASH byte writing function
mov FLKEY,#05Ah; //
mov FLKEY,#01Fh // Consequentially write 0x5A 、0x1F to register FLKEY to open FLASH
// key.
mov   DPH,#03h;   //
mov   DPL,#0F1h;   // Enter the FLASH address of the byte, the area where the user want to write.
mov   A,#0Ah;        //
movx @DPTR,A;     // movx data “0x0A” to the FLASH address space “0x03F1”.
mov PSCTL,#08h;   // Write 0x08 to register PSCTL to start the writing process, after the
// process, FLASH key locked again automatically.
// First byte data writing is end, start another byte.
mov a,PSCTL;       // Check if this writing process is successful.
anl a,#10h,
cjne a,#00h,FLA_OP_ERROR // if PSCTL.FLAVIO is not zero, the process is error.
```

NOTE:

1. To protect user's 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 time sector erase will consume about 120mS, and one byte data writing will consume about 100uS.

30 CRC (Cyclic Redundancy Check Unit)

Table 30-1 CRCCriterion 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

30.1 CRC16 Generator Polynomial

FU68XX CRC system is based CRC16-CCITT Criterion generator polynomial:

$X^{16}+X^{12}+X^{5}+1$。

30.2 CRC16 basic logic diagram

A serial CRC16 circuit diagram as shown in the figure 30-1, FU68XX is based on parallel CRC16, which can be generated witch 1 system clock cycle.

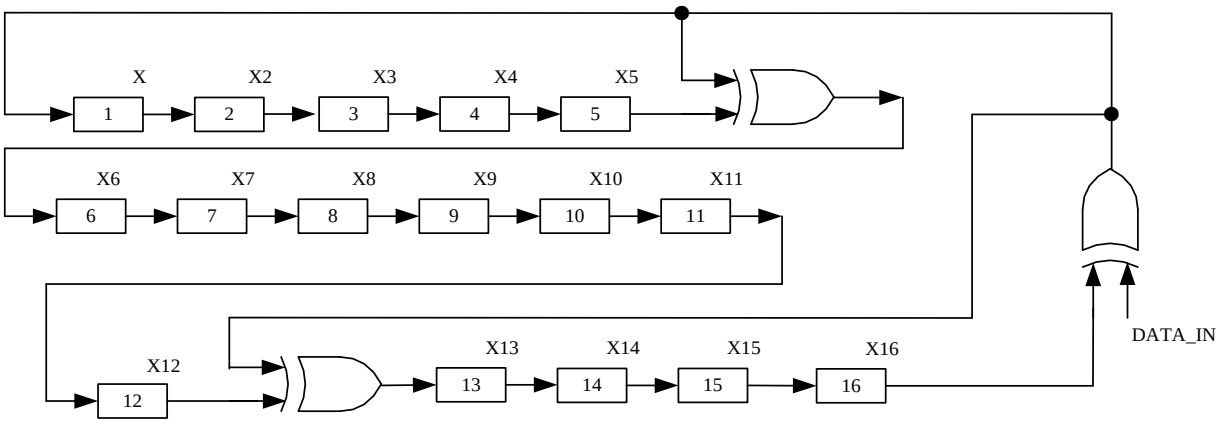


Figure 30-1 CRC16 logic circuit diagram

30.3 Operation step:

30.3.1 Basic function block diagram

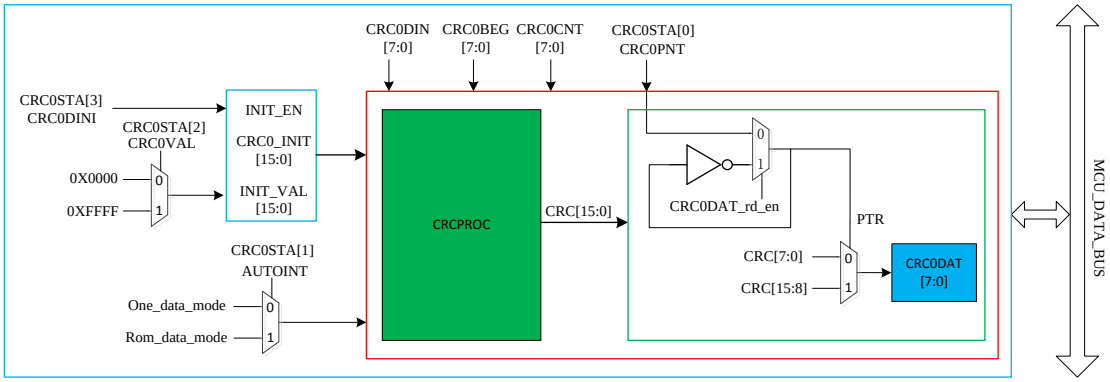


Figure 30-2 Basic function diagram

30.3.2 Computing single byte CRC

To compute single byte CRC, starting as following step:

1. Software clear control register CRC0STA[1] to zero, enter CRC single byte computing mode.
2. Software writes control register CRC0STA to 0x1D. CRC0STA[0]=1 means CRC result pointer points to high byte of the 16bit result. CRC0STA[2]=1 means CRC intial data is 0xFFFF, CRC0STA[3]=1 means CRC result is valid.
3. Software writes object data to CRC0DIN, for example 0x63. After 1 clock cycle, CRC result will be generated, and the result is 0xBD35.
4. Reading CRC result: Software writes CRC0STA[0]=1 and read the CRC result high byte from register CRC0DAT, for example, if the read data is 0xBD; software writes CRC0STA[0]=0 and read the CRC result low byte from register CRC0DAT, and in this example, read data is 0x35.
5. User can also initialize some data to CRC results. The process as follow: firstly, software writes CRC0STA[0]=0 to put the pointer to low byte of the result, and then writes data to CRC0DAT. Secondly, software writes CRC0STA[0]=1 to put the pointer to high byte of the result, and then writes data to CRC0DAT. Thirdly, software writes some data to CRC0DIN, after the next clock cycle, a new CRC result generated.

   For example, set CRC0STA[0]=0, CRC0DAT=0xCF; and then set CRC0STA[0]=1, CRC0DAT=0xD4, and then write CRC0DIN=0xD1, after one clock cycle, new CRC result generated 0xD4CF. Finally, software writes CRC0STA[0]=0 to read low byte of the result CRC0DAT, and writes CRC0STA[0]=1 to read high byte of the result CRC0DAT, the new CRC result is 0x9FA5.

30.3.3 Batch computing CRC of ROM data

To computing many sectors of FLASH ROM area CRC result, do steps as following:

Step 1: Software writes register CRC0STA[1] to 0, stop all computing.

Step 2: Software writes other bit of CRC0STA proper value, set the result pointer, set initial value, and enable initialization.

Step 3: Software writes proper value to register CRC0BEG to set start address of FLASH ROM.

Step 4: Software writes proper value to register CRC0CNT to set how many sectors of FLASH ROM to compute CRC result.

Step 5: Software set register CRC0STA[1] to 1 to start CRC computing.

Step 6: Software can read CRC0STA[4] to check if the computing process has finished.A read of 1 means finished.

Step 7: Software set CRC0STA[0] to 0 and read register CRC0DAT, CRC result low byte, then set CRC0STA[0] to 1 and read register CRC0DAT, CRC result high byte

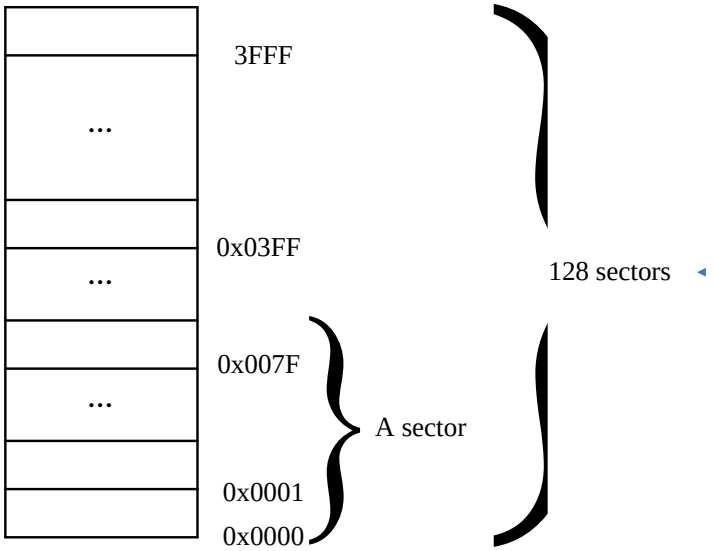


Figure 30-3 FLASH ROM Sector frame

As shown in up figure, all FLASH ROM is 16K bytes, which includes 128 secors, from sector0 to sector 127, each sector includes 128 bytes. Software can set register CRC0BEG to any value from 0x00 to 0x7F. But should consider CRC0BEG value when set CRC0CNT, for example, if set CRC0BEG to 0x7F, then should set CRC0CNT to 0x00, it means computing 1 sector FLASH ROM data CRC.

30.4 CRC Registers

30.4.1 CRC0STA

Table 30-2 CRC0STA（0x4022）

Bit	7:5	4	3	2	1	0
Name	RSV	CRCDONE	CRC0DINI	CRC0VAL	AUTOINT	CRC0PNT
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	CRCDONE Automatic CRC calculation completed 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]	CRC0DINI	CRC result initialization bit. Writing a 1 to this bit will initialize the entire CRC result based on CRC0VAL.
[2]	CRC0VAL	CRC set value initialization bit. 0: CRC result is set to 0x0000 on write of 1 to CRC0DINI. 1: CRC result is set to 0xFFFF on write of 1 to CRC0INI.
[1]	AUTOINT	Automatic CRC calculation enable When this bit is set to 1, any write to CRC0CNT will initiate an automatic CRC starting at FLASH sector CRC0BEG and continuing for CRC0CNT sectors. CRC calculation include 2 modes, one of single CRC alculation, 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]	CRC0PNT	CRC0 result pointer. Specify the byte of the CRC result to be read/written on the next access to CRC0DAT. The value of these bits will auto-increment upon each read or write. 0: CRC0DAT accesses bits 7-0 of the 16-bit CRC result. 1: CRC0DAT accesses bits 15-8 of the 16-bit CRC result.

30.4.2 CRC0DIN: CRC input data

Table 30-3 CRC0DIN（0x4021）

Bit	7	6	5	4	3	2	1	0
Name	CRC0DIN							
Type	W							
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:0]	CRC0DIN	CRC0 data input. Each write to CRC0DIN results in the written data being computed into the CRC algorithm.

30.4.3 CRC0DAT: CRC Calculation Result

Table 30-4 CRC0DAT（0x4023）

Bit	7	6	5	4	3	2	1	0
Name	CRC0DAT							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7:0]	CRC0DAT	CRC Data output. Each read or write performed on CRC0DAT targets the CRC result bits pointed to by the CRC0 Result Pointer(CRC0PNT bits in CRC0STA)

30.4.4 CRC0BEG

Table 30-5 CRC0BEG（0x4024）

Bit	7	6	5	4	3	2	1	0
Name	RSV	CRC0BEG						
Type	R	R/W						
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
[7]	RSV	Reserved
[6:0]	CRC0BEG	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 CRC0BEG x sector size.

30.4.5 CRC0CNT

Table 30-6 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]	CRC0CNT	Automatic CRC calculation sector count. These bits specify the number of FLASH sectors to include in an automatic CRC calculation. The last address of the last FLASH sector included in the automatic CRC calculation is $(CRC0BEG + CRC0CNT) \times SectorSize$ Notes: The sector size is 128 bytes.

31 Power Management

31.1 PCON(0x87)

Table 31-1 PCON（0x87）

Bit	7	6	5	4	3	2	1	0
Name	RSV		GF3	GF2	GF1	GF0	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	GF0	General Purpose Flag 0
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

31.2 Power Consumption Mode

RTC(Real Time Clock) is an individual function based on external critical. It is not affected by power consumption mode setting, WDT(watch-dog timer) is also an individual function based on internal critical. When system is in idle mode or deep sleep mode, WDT timer will be hold.

The device has 3 power consumption mode: normal mode, idle mode and deep sleep mode. Below table is the station in every mode

Table 31-2 Power Consumption Mode

Mode	Function	Wake-up Sources	Power Consumption Performance
Normal	All function is working, except some modules user software disabled.	NA	Power consumption is highest, but action fastest.
Idle	CPU clock is gated, other modules is disabled or working, decided by user's software. Watch-dog clock is gated.	Any interrupt request External reset Debug reset	Power consumption is low, action fast.
Deep Sleep	System Deep-Sleep。 Fast clock critical circuit will be closed, and if ADC or FOC or motor control circuit is running before fast clock critical circuit closed, fast clock will running until those running circuits have ended Watch-dog clock is closed	External(INT0/INT1) interrupt request RTC interrupt request External reset Debug reset	Power consumption is lowest, action slow.

32 Code Protection

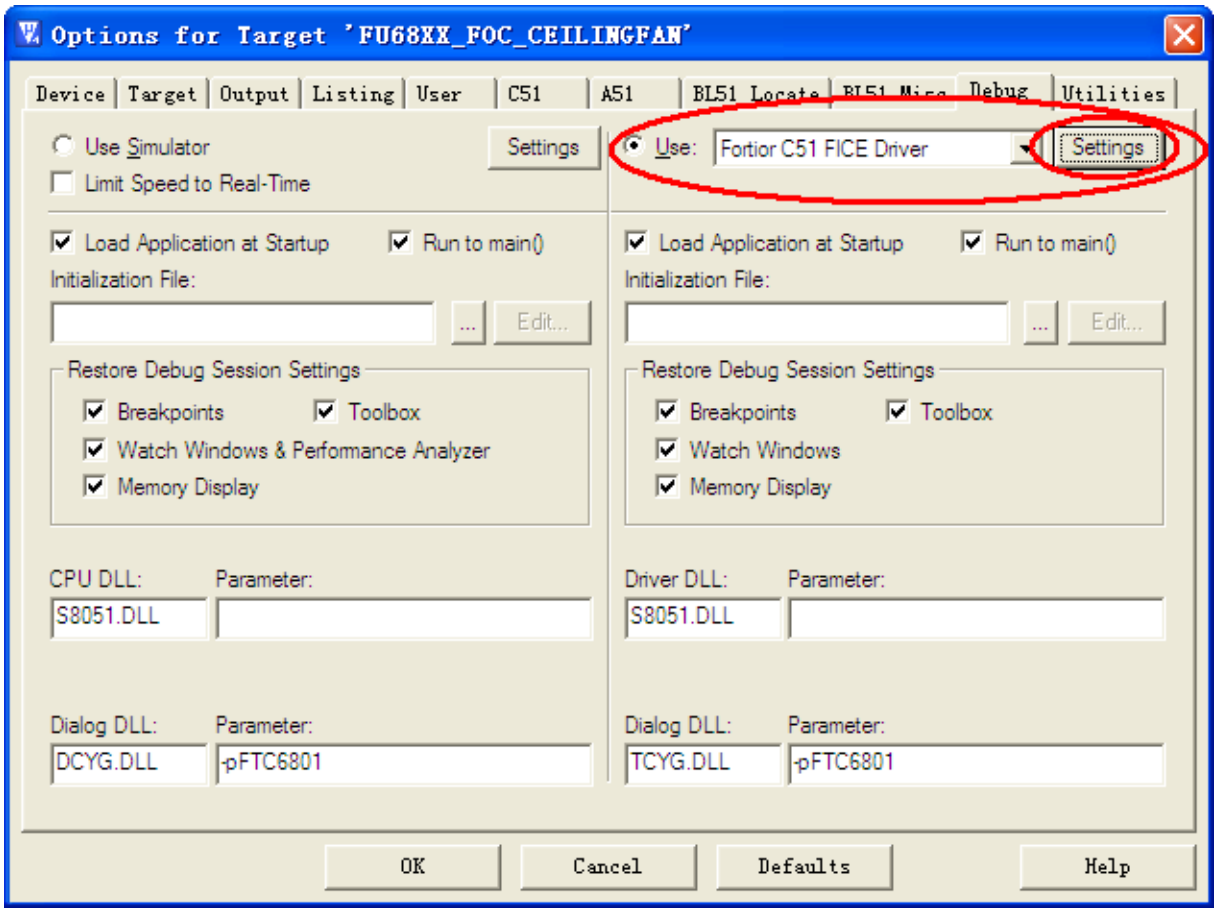


Figure 32-1

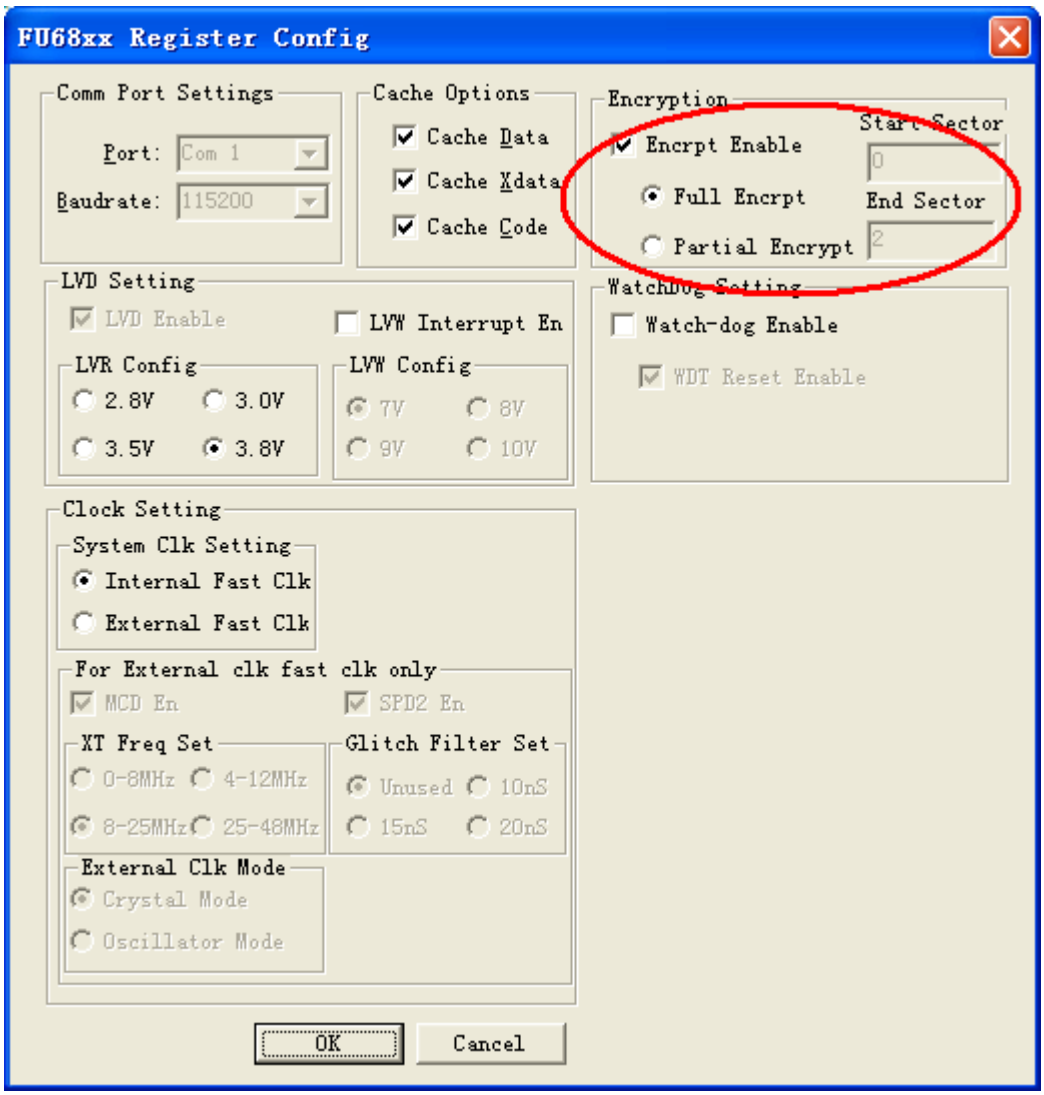


Figure 32-2

FU68XX Supports user FLASH ROM code IPR(Intellectual Property Protection), mothed and step as below:

Step 1: Open 8051 integrated design tool, enter “Target Options” , select “Debug” button, and click button settings as figure 32-1.

Step 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.

33 Config Register

33.1 CCFG, User Config Register

33.1.1 CCFG1: CK_RST_CFG

Table 33-1 CCFG1（0x401E）

Bit	7	6	5	4:1	0
Name	LVDENB	LVWIE	WDTEN	RSV	CKMOD
Type	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0

Note:

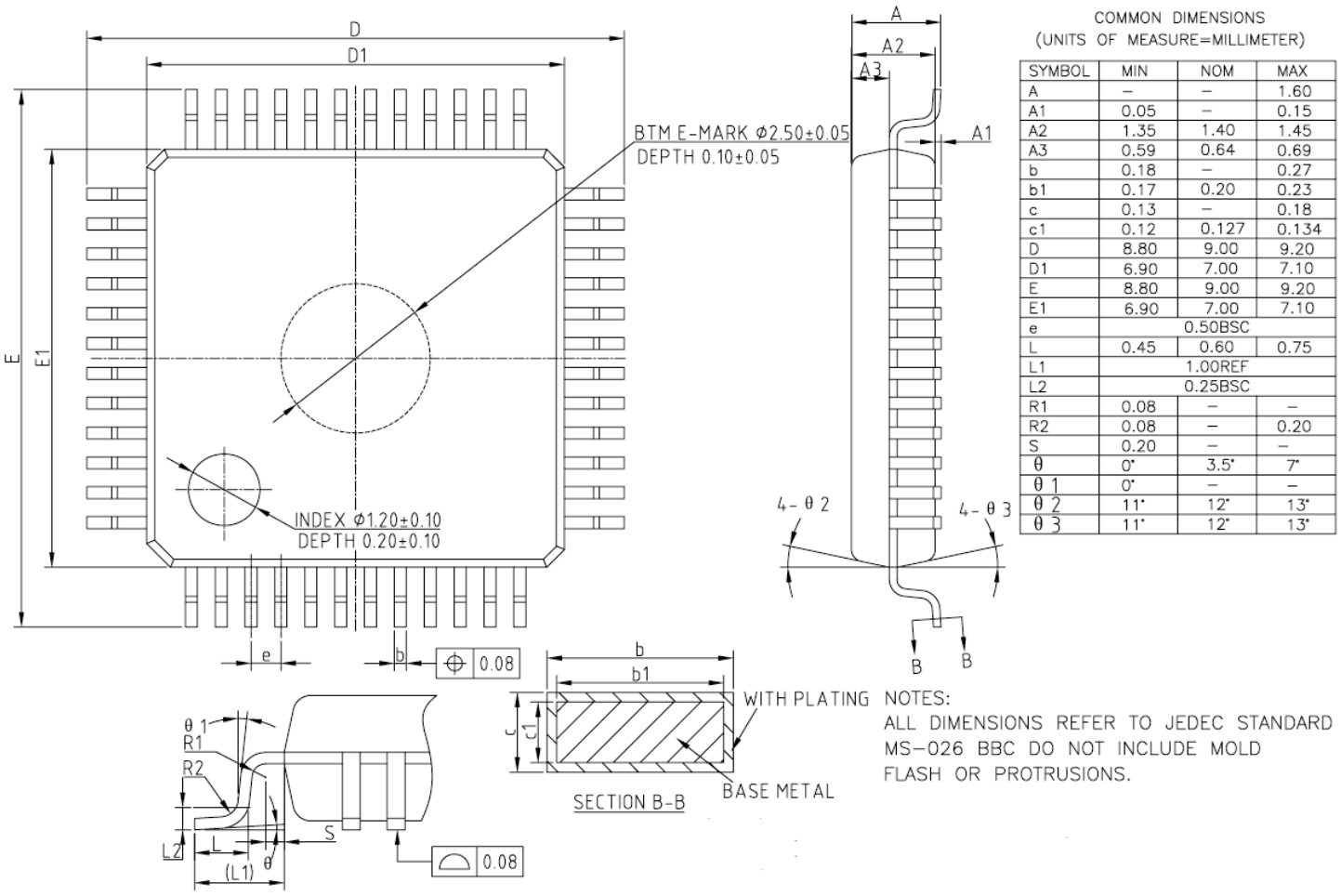
*: indicates default is 0 after reset.

Bit	Name	Function
7	LVDENB	Low Voltage Reset Detect Enable 0：Enable：MCU will reset when VDD5 is less than threshold voltage set by LVRSEL 1：Disable Note： It includs two parts：one is low voltage reset detect, MCU will reset when VDD5 is less than threshold voltage setted by LVRSEL; another one is low voltage alarm function, where MCU will trigger interrupt of low voltage alarm when VCC is less than threshold voltage setted by LVWSEL. For any of above functions to work，LVDENB must be set to zero first (this also means this bit must be enabled). However, the low voltage alarm function will also be controlled by bit of LVWIE .
6	LVWIE	VCC low voltage alarm interrupt enable, the threshold voltage of VCC will be setted by LVWSEL. 0：Disable 1：Enable，if you need enable low voltage interrupt function, you must enable low voltage reset detect function (by setting LVDENB bit to zero)
5	WDTEN	Watch-dog Enable。 0：Disable 1：Enable
4:1	RSV	Reserved

Bit	Name	Function
0	CKMOD	System clock selection： 0: it will be as internal fast clock 1: it will be as external clock Note：when system boot, if this bit in the flash is 1, the EFOSC_AE that supply to the analog circuits will be set to 1 after system boot has finished, The two pins of external fast clock crystal will be forced as analog input, the program cann't modify it simultaneously. However, if this bit in the flash is 0, the EFOSC_AE will be cleared to 0 after system boot. If the system enter to the external fast clock crystal mode, IC will leak electricity in virtue of two crystal pin floating when system sleep, Thus, user have to set this bit to 1 in the flash before they want to use external crystal fast clock.

34 Package

34.1 LQFP48_7X7

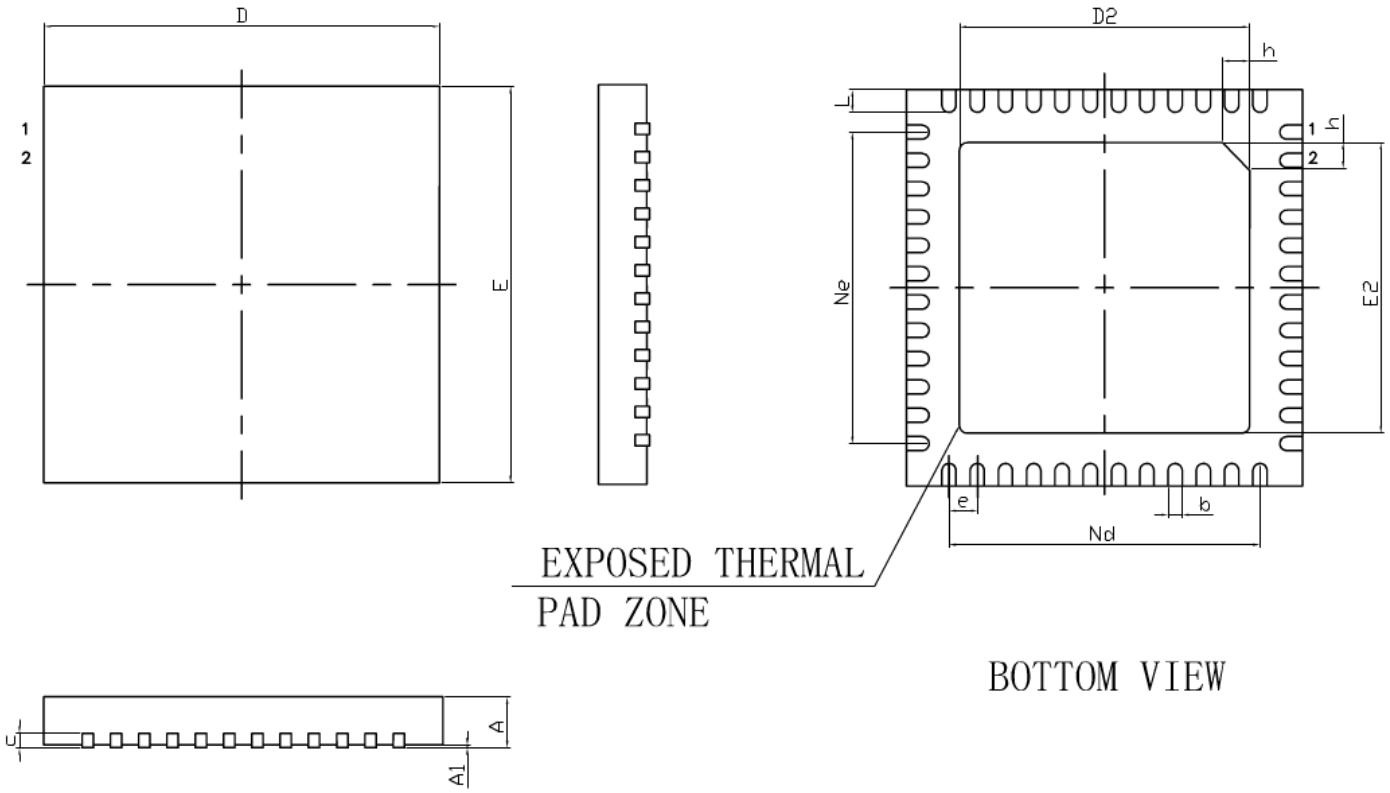


SYMBOL	MIN	NOM	MAX
A	–	–	1.60
A1	0.05	–	0.15
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.18	–	0.27
b1	0.17	0.20	0.23
c	0.13	–	0.18
c1	0.12	0.127	0.134
D	8.80	9.00	9.20
D1	6.90	7.00	7.10
E	8.80	9.00	9.20
E1	6.90	7.00	7.10
e		0.50BSC	
L	0.45	0.60	0.75
L1		1.00REF	
L2		0.25BSC	
R1	0.08	–	–
R2	0.08	–	0.20
S	0.20	–	–
θ	0°	3.5°	7°
θ1	0°	–	–
θ2	11°	12°	13°
θ3	11°	12°	13°

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

Figure 34-1 LQFP48_7X7 Package Diagram

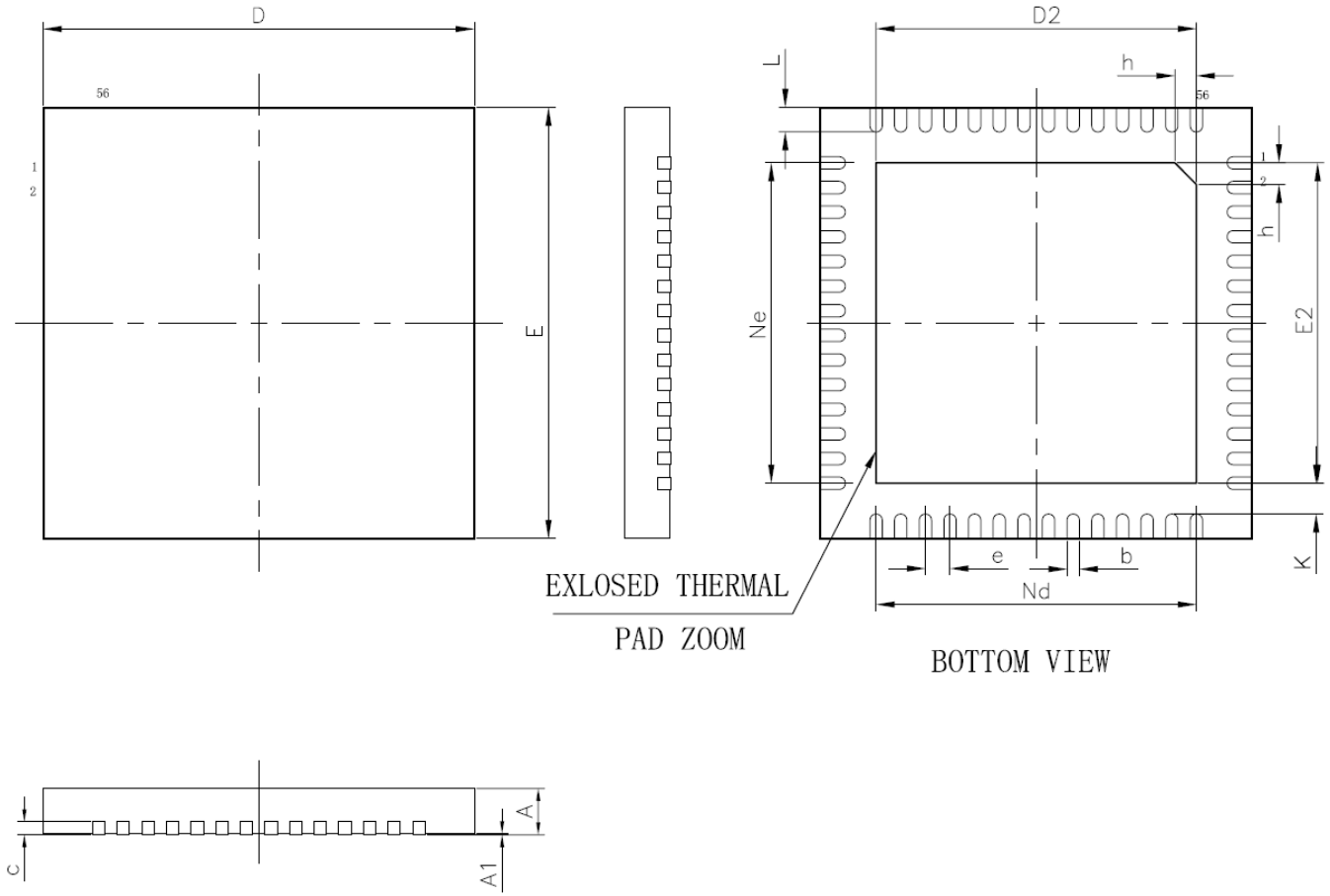
34.2 QFN48_6X6



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0. 70	0.75	0. 80
A1	—	0.02	0. 05
b	0. 15	0.20	0. 25
c	0. 18	0.20	0. 23
D	5. 90	6. 00	6. 10
D2	4. 10	4.20	4. 30
e	0. 40BSC		
Ne	4. 40BSC		
Nd	4. 40BSC		
E	5. 90	6. 00	6. 10
E2	4. 10	4.20	4. 30
L	0. 35	0.40	0. 45
h	0. 30	0. 35	0. 40
L/F载体尺寸 (MIL)	177*177		

Figure 34-2 QFN48_6X6 Package Diagram

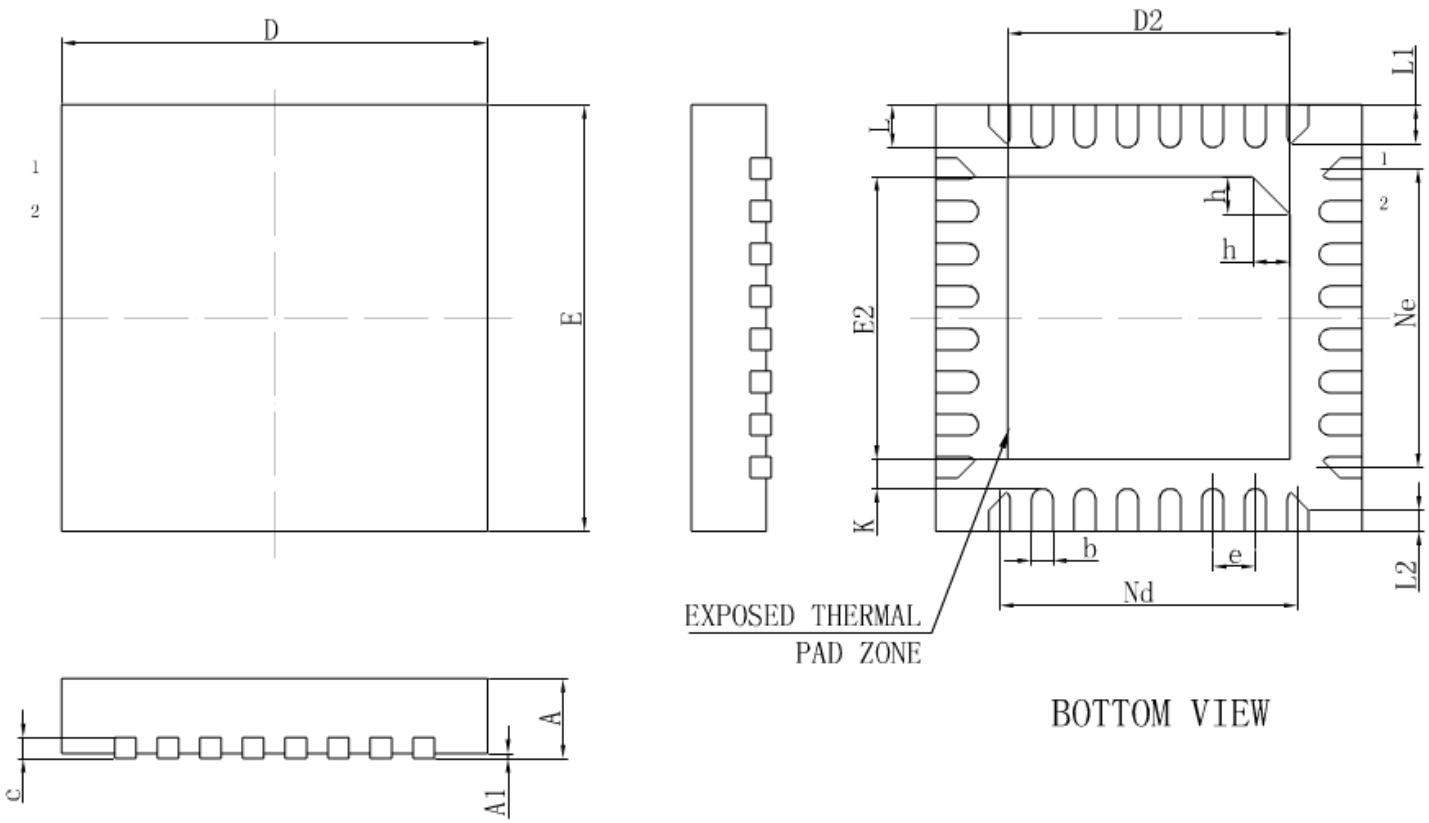
34.3 QFN56_7X7



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.70	0.75	0.80
A1	—	0.02	0.05
b	0.15	0.20	0.25
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
L/F载体尺寸 (mil)	217*217		

Figure 34-3 QFN56_7X7 Package Diagram

34.4 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
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
载体尺寸 (Mil)	112*112		

Figure 34-4 QFN32_4x4 Package Diagram

35 Ordering Information

Table 35-1 Product Selection Guide

Type	MIPS(Peak)	FLASH(KB)	XRAM(KB)	Oscillation Mode				Predriver Interface			Driving Control			Series Interface			GPIO	Timer	Analog Peripheral						Leadfree	Package
																			ADC			VREF	Amplifier	Comperator		
				Inernal Fast Clock	External Fast Clock	Internal Slow Clock	External Slow Clock	6N Predriver	3P3N Predriver	Gate Driver	BLDC	SVPWM	FOC	I2C	SPI	UART			Number	Channel	Bits					
FU6831L	24	16	4	√	√	√	√	—	√	—	√	√	√	√	√	√	32	6	1	8	12	√	4	4	√	LQFP48 (7x7 mm)
FU6831Q	24	16	4	√	√	√	√	—	√	—	√	√	√	√	√	√	32	6	1	8	12	√	4	4	√	QFN48 (6x6 mm)
FU6831N	24	16	4	√	—	√	—	—	√	—	√	√	√	√	√	√	18	6	1	6	12	√	1	4	√	QFN32 (4x4 mm)
FU6811L	24	16	4	√	√	√	√	—	—	√	√	√	√	√	√	√	32	6	1	8	12	√	4	4	√	LQFP48 (7x7 mm)
FU6811N	24	16	4	√	—	√	—	—	—	√	√	√	√	√	√	√	19	6	1	7	12	√	1	4	√	QFN32 (4x4 mm)
FU6818Q	24	16	4	√	√	√	√	√	—	—	√	√	√	√	√	√	32	6	1	8	12	√	4	4	√	QFN56 (7x7 mm)

Copyright Notice

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

Contained herein