CGF022A
Data Sheet

8Bit Single-Chip Microcontroller
Embedded 40V 3-Phase Gate-Driver

Version: V0.1

Features

Motor Driving Engine (MDE)

- Estimated Angle Phase Lock Loop (PLL)
- Field Oriental Control (FOC) Sine-Wave Solutions
- Slide Mode Rotor Position Estimated (SMO)
- Space Vector PWM (SVPWM)
- Supports Digital OCP and Analog OCP (Over Current Protection)
- Supports Initial Position Detection (IPD)
- Programmable Dead-Time
- Independent PI Controller
- Independent General Low Pass Filter
- Frequency Generator

Gate Driver

- Integrated 40V 3-phase P/N MOSFET Pre-driver
- Shoot-through protection
- Built-in 5V LDO
-

Embedded MCU

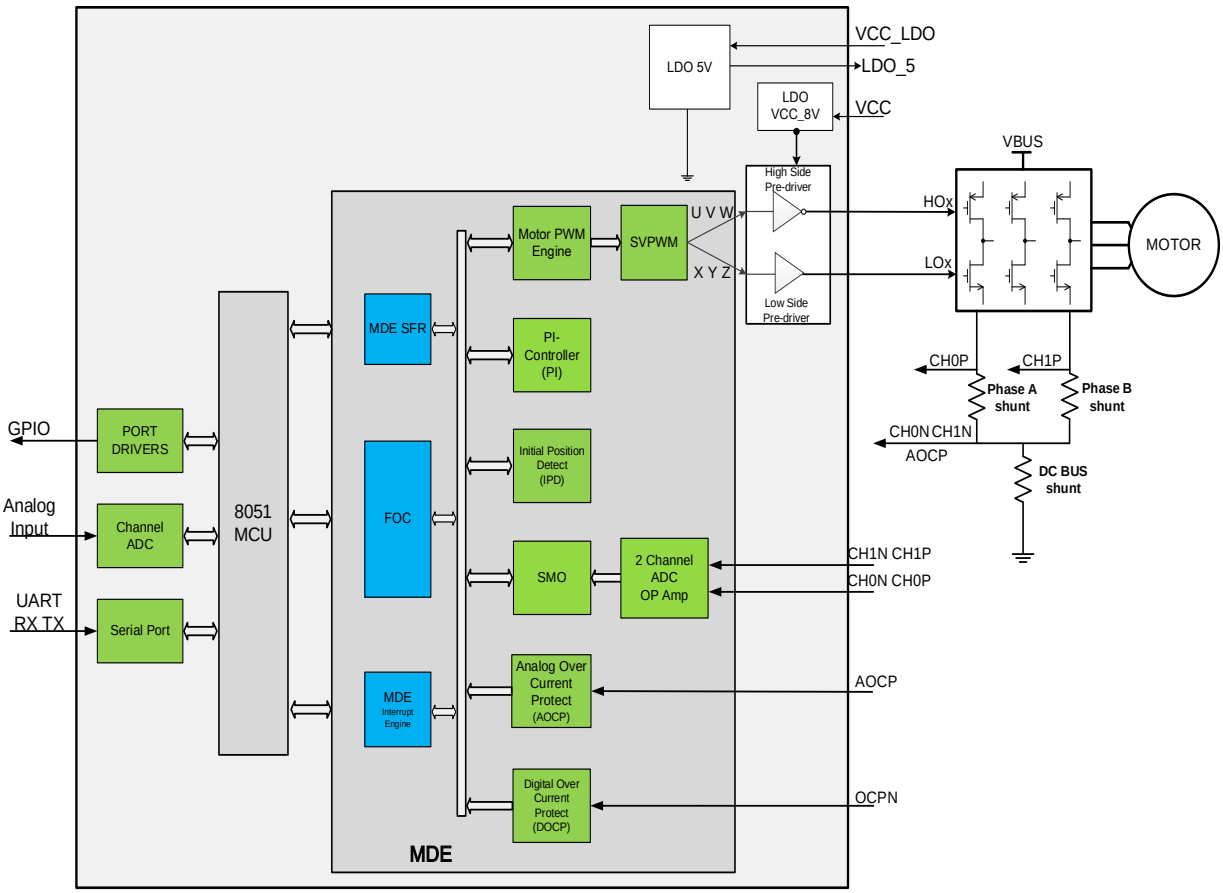
- MCS®-51 Compatible
- 1T 8052 Central Processing Unit
- 4.5V to 5.5V Operation Range
- 4 Level Priority Interrupt
- 13 Interrupt Sources
- 1 External Interrupts (INT1N)
- 2 External Interrupts (INT0N, INT1N)
- 2 External OCP Interrupts (AOCP, OCP)
- Memory Size:
- 16KB Flash Program Memory
- 256 x 8-bit IRAM
- 512 x 8-bit XRAM
- 256 Byte EEPROM
- Up to 25 General-Purpose Input / Output (GPIO) Pins
- Three 16-bit Timer/Counters
- Watchdog (WD) Timer
  8CH 10-bit ADC & 1CH 10-bit DAC
- Full Duplex UART Serial Channel
- IIC Interface (Master/Slave Mode)
- One Wire RF/IR Receiver Output Signal Decode
- CRC16-CCITT Function
- Independent General PWM
- External Capture
- Internal Capture
- Fast Multiplication-Division Unit (MDU): 16*16,32/16, 16/16, 32-bit L/R shifting and 32-bitnormalization

List of Contents

1. General Description

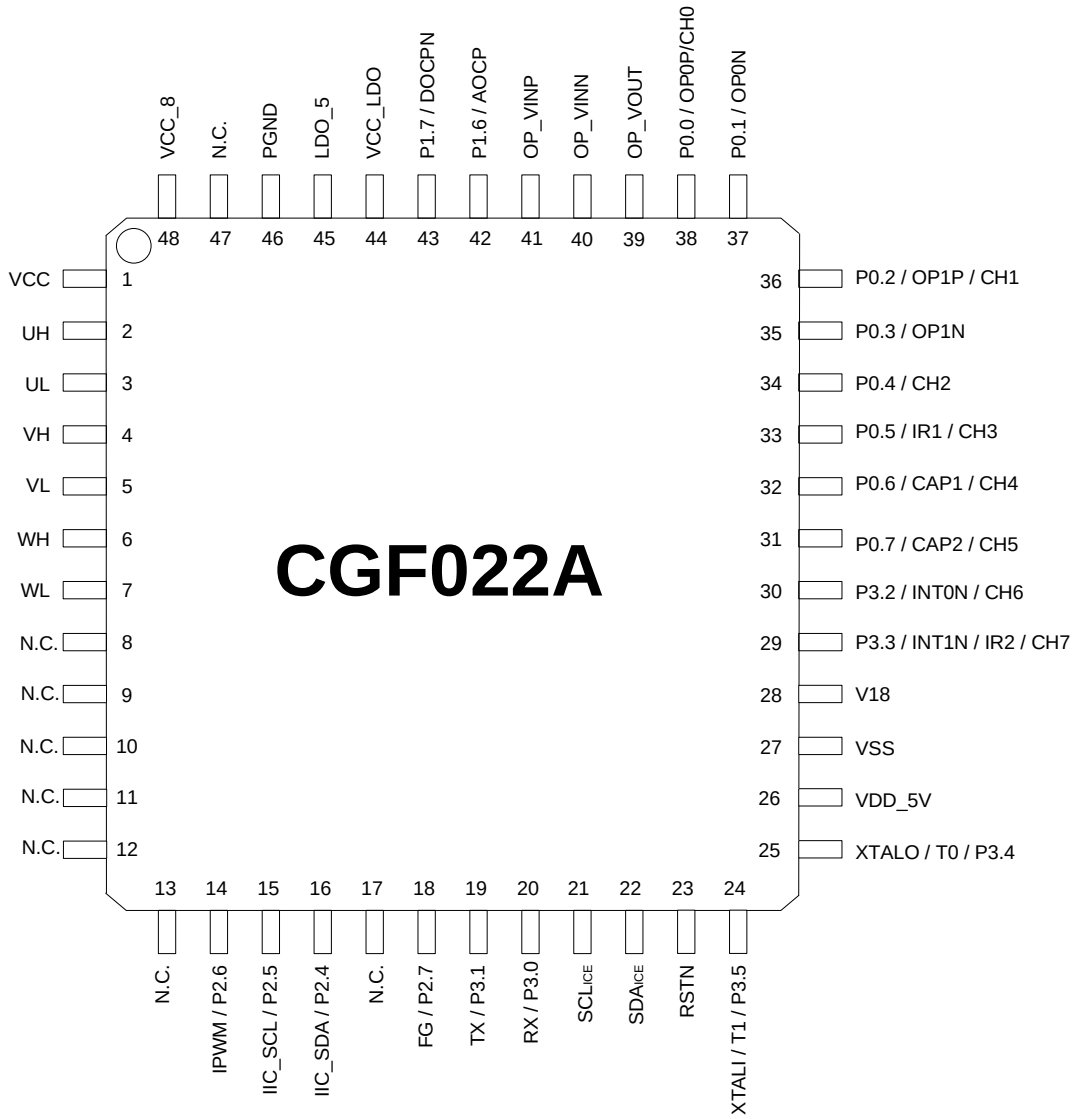
The CGF022A is a highly integrated motor drive controller. The CGF022A is composed of FOC sensor-less MCU and 40V 3-phase P/N Gate-Driver that suit for under DC 40V and medium motor system, for example household fan、water pump、industry fan…etc.

2. Block Diagram



3. Pin Configurations

3.1. Package Instruction LQFP7x7- 48



3.2. Pin Description

Table 3.1. Pin Description

Pin #	Name	Type	Description
1	VCC	Power	Supply voltage input.
2	UH	O	High side phase-U PMOS driver.
3	UL	O	Low side phase-U NMOS driver.
4	VH	O	High side phase-V PMOS driver.
5	VL	O	Low side phase-V NMOS driver.
6	WH	O	High side phase-W PMOS driver.
7	WL	O	Low side phase-W NMOS driver.
8	NC		
9	NC		
10	NC		
11	NC		
12	NC		
13	NC		
14	P2.6	I/O	Bit6 of Port 2.
	IPWM	O	Independent User PWM Output
15	P2.5	I/O	Bit5 of Port 2.
	IIC_SCL	O	IIC clock
16	P2.4	I/O	Bit4 of Port 2.
	IIC_SDA	O	IIC data
17	NC		
18	P2.7	I/O	Bit7 of Port 2.
	FG	O	Function Generate Output
19	P3.1	I/O	Bit1 of Port 3.
	TX	O	Serial Data Receive (UART)
20	P3.0	I/O	Bit0 of Port 3.
	RX	I	Serial Data Transmit (UART)
21	SCL_{ICE}		For ICE.
22	SDA_{ICE}		For ICE.
23	RSTN	I	System Reset.
24	P3.5	I/O	Bit 5 of Port 3
	XTALI	I	Crystal input pin. **Connect the crystal 12MHz between this pin and XTALO and a 22pF capacitor to VSS**
	T1	I	TIMER1 External Input
25	P3.4	I/O	Bit 4 of Port 3
	XTALO	O	Crystal output pin. **Connect the crystal 12MHz between this pin and XTALI and a 22pF capacitor to VSS**
	T0	I	TIMER0 External Input
26	VDD5	Power	5.0V Voltage Input. A 0.1uF and 10uF (minimum) capacitor should be connected between this pin and VSS.
27	VSS	Ground	Power Ground.
28	V18	O	1.8V Voltage Output. A 0.1uF and 1uF (minimum) capacitor should be connected between this pin and VSS.
29	P3.3	I/O	Bit 3 of Port 3
	CH7	I	Analog Input Ch7

	INT1N	I	External Interrupt 1. Low level trigger or falling edge trigger
	IR2	I	IR receiver signal input 2
30	P3.2	I/O	Bit2 of Port 3
	CH6	I	Analog Input Ch6
	INT0N	I	External Interrupt 0. Low level trigger or falling edge trigger
31	P0.7	I/O	Bit7 of Port 0
	CH5	I	Analog Input Ch5
	CAP2	I	Capture Input 2
32	P0.6	I/O	Bit6 of Port 0
	CH4	I	Analog Input Ch4
	CAP1	I	Capture Input 1
33	P0.5	I/O	Bit5 of Port 0
	CH3	I	Analog Input Ch3
	IR1	I	IR receiver signal input 1
34	P0.4	I/O	Bit4 of Port 0.
	CH2	I	Analog Input Ch2.
35	P0.3	I/O	Bit3 of Port 0.
	OP1N	0	OP1-Amp N- Input
36	P0.2	I/O	Bit2 of Port 0.
	CH1	I	Analog Input Ch1. (Current feedback)
	OP1P	I	OP1-Amp P-Input.
37	P0.1	I/O	Bit1 of Port 0.
	OP0N	I	OP0-Amp N-Input.
38	P0.0	I/O	Bit0 of Port 0.
	CH0	I	Analog Input Ch0. (Current feedback)
	OP0P	I	OP0-Amp P- Input.
39	OP_VOUT	O	OPA output
40	OP_VINN	I	OPA N-Input
41	OP_VINP	I	OPA P-Input
42	P1.6	I/O	Bit6 of Port 1.
	AOCP	I	Analog OCP Control.
43	P1.7	I/O	Bit7 of Port 1.
	DOCPN	I	Digital OCP Control.
44	VCC_LDO	Power	LDO 5V power supply.
45	LDO_5	Power	5V output of LDO.
46	PGND	Ground	Power ground.
47	NC		
48	VCC_8	O	LDO VCC_8 output

4. Absolute Maximum Ratings

Item	Min.	Typ.	Max.
VDD5 Supply Voltage	V_{SS}-0.3V		V_{SS}+6.0V
VDD5 Input Voltage	V_{SS}-0.3V		V_{DD}+0.3V
Operating Temperature with LDO_5V 30mA	-40℃		85℃
Operating Temperature w/o LDO_5V	-40℃		105℃
Gate Driver Source peak current		80mA	
Gate Driver Sink peak current		50mA	
Storage Temperature	-50℃		150℃
Operating Temperature	-20℃		105℃
I_{OH} Total		-80mA	
I_{OL} Total		80mA	
Total Power Dissipation		500mW	
Electrostatic Discharge Capability – Human Body Mode		TBD (KV)	
Electrostatic Discharge Capability – Machine Mode		TBD (V)	

5. D.C. Characteristics

Ta=25℃

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V	Conditions				
VCC	DC Voltage Input			10	–	40	V
$V_{IH\_U/V/WH}$	Logic "1" Input Voltage	Vcc = 10V to 40V		2.4	–	–	V
$V_{IL\_U/W/WH}$	Logic "0" Input Voltage	Vcc = 10V to 40V		–	–	0.8	
$I_{O+\_U/V/WH}$	Source Peak Current	CL = 1nF			80		mA
$I_{O-\_U/V/WH}$	Sink Peak Current	CL = 1nF			50		mA
I_{QCC}	Quiescent V_{CC} supply current	V_{IN} = 0V or 5V				800	μA
R_{IN}	HI_U/V/W, LI_U/V/W Pin Pull Low Resistor				100		KΩ
Rh_{OUT}	HO_U/V/W Pin Pull High Resistor				100		KΩ
RI_{OUT}	LO_U/V/W Pin Pull Low Resistor				100		KΩ
VCC_8 LDO Characteristics							
VCC_8	VCC_8 Regulator Output Voltage	VCC = 24V, Io = 6mA			16		V
5V LDO Characteristics							
V_{LDO}	Regulator Output Voltage	VCC = 24V, Io = 30mA			5.0		V
V_{DD5}	Operating Voltage	—	fsys=48MHz	4.5	5	5.5	V
V_{18}	V_{18} Output Range	—	Load Current <30mA	1.71	1.8	1.81	V

I_{DD}	Operating Current	5V	No load, fsys=48Mhz,ADC off, MOC off	—	9	12	mA
V_{IL}	Input Low Voltage for I/O Ports.	—	—	0	—	0.3 V_{DD5}	V
V_{IH}	Input High Voltage for I/O Ports.	—	—	0.75 V_{DD5}	—	V_{DD5}	V
V_{LVD}	LVD Voltage Level	—	—				
V_{OL}	Output Low Voltage for I/O Ports.	VDD5	I_{OL}=20mA	—	—	0.5	V
V_{OH}	Output High Voltage for I/O Ports.	VDD5	I_{OH}=-7.4mA	4.5	—	—	V
R_{PU}	Pull-up Resistance for I/O Ports	VDD5	—	10	30	50	KΩ
R_{PD}	Pull-down Resistance for I/O Ports	VDD5	—	10	30	50	KΩ

6. A.C. Characteristics

Ta=25℃

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		VDD5	Conditions				
f_{sys}	System Frequency	4.5V~5.5V	Ta=-40℃ to 125℃	TBD	48.0	TBD	MHz
			Ta=-20℃ to 85℃	TBD	48.0	TBD	MHz
			Ta=25℃	-1%	48.0	+1%	MHz
f_{TIMER}	Timer Input Pin Frequency	—	—	—	—	4	f_{sys}
t_{INT}	Interrupt Pulse Width	—	—	1	5	10	t_{sys}
t_{LVD}	Low Voltage Width to interrupt	—	—	120	240	480	us
t_{V18}	V18 Stable Time	—	—	60	120	240	us
t_{RSDT}	System Reset Delay Time(Power On Reset)	—	—	25	50	100	ms

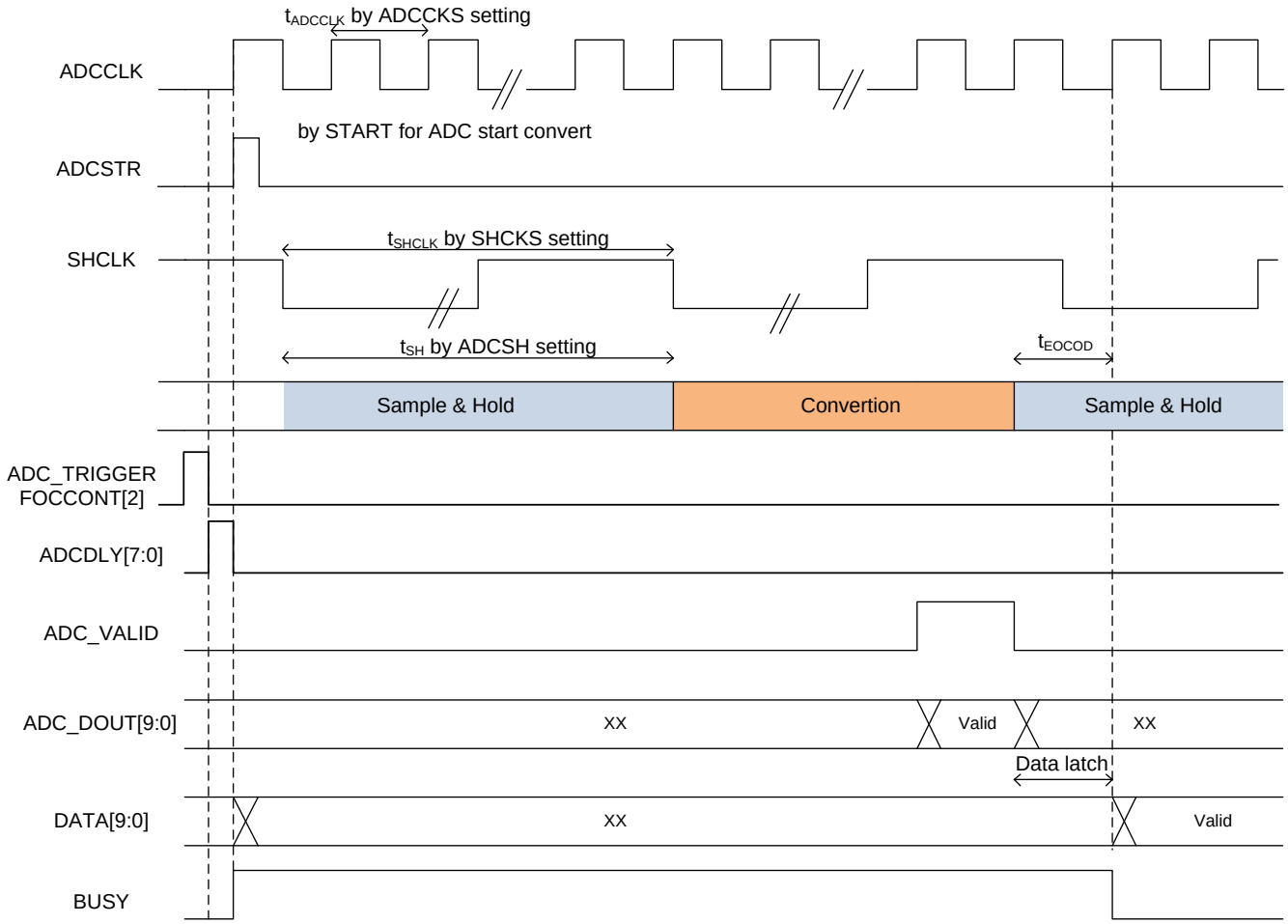
7. OPA Characteristics

Ta=25℃, V_{DD}= 5V, V_{SS}=GND

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
V_{CRM}	Common Mode Input Range		V_{SS}-0.3	—	V_{DD}+0.3	V
V_{OS}	Input Offset Voltage	V_{CM}=V_{SS}	-4.5		4.5	mV
A_{OL}	DC Open-Loop Gain	V_{OUT}=0.3V~V_{DD}-0.3V V_{CM}=V_{SS}	88	112		dB
GBWP	Gain Bandwidth Product	R_L=10KΩ C_L=60 pF		1		MHz
SR	Slew Rate	C_L=60 pF		0.6		V/us

Note: OPA default is power down. If you want to use OPA, you need to set the OPAPD control bit to "1" first. (see Table25.10.1 AOCPCONT SFR bit[5])

8. A/D Converter Characteristics



Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		VDD5	Conditions				
R_I	Input Impedance	—	—	—		—	MΩ
I_{AD}	Additional Power Consumption if A/D Converter is Used	5V	—	—	6	—	mA
I_{ADSTB}	A/D Converter Standby Current	—	Load Current < 10mA	—	—	4	uA
t_{ADCCLK}	A/D Converter Clock Time	—	24MHz	—	41.7	—	ns
		—	12MHz	—	83.3	—	ns
t_{CONV}	A/D Conversion Time	—	24MHz	—	0.875	—	us
		—	12MHz	—	1.7	—	us
t_{SH}	A/D Sample and Hold Time	—	6MHz	0.17	—	0.68	us
		—	3MHz	0.34	—	1.36	us
		—	2.4MHz	0.42	—	1.68	us
		—	2MHz	0.5	—	2	us
DNL	Differential Non-linearity	4.5V	No load, t_{CONV}=2.5us	-1	—	+3	LSB
		5.5V		-1	—	+3	LSB
		4.5V	No load, t_{CONV}=5us	-1	—	+3	LSB
		5.5V		-1	—	+3	LSB
INL	Integral Non-linearity	4.5V	No load, t_{CONV}=2.5us	-4	—	+4	LSB
		5.5V		-4	—	+4	LSB
		4.5V	No load, t_{CONV}=5us	-4	—	+4	LSB
		5.5V		-4	—	+4	LSB
G_{ERR}	Gain Error	—	—	-10	—	+10	LSB
t_{EOCOD}	End of Conversion Output Delay	5V	—	—		—	ns

9. Special Function Registers (SFR)

SFRs Memory Map

	8	9	A	B	C	D	E	F	
F8	PINCONG1	PINCONG2	PINCONG3	PINCONG4	PINCONG5	PINCONG6	RSTS	TAKEY	FF
F0	B	PINSET1	PINSET2	PINSET3	PINSET4	PINSET5	PINSET6	PINSET7	F7
E8	IICS	IICCTL	IICA1	IICA2	SFR_PAGE	SP_CYC	AOCPCONT	OCPNCONT	EF
E0	ACC	MPWMDATL	MPWMDATH	MPWMCONT1	MPWMINV	MPWMDT	PI_GAIN	MPWMCONT2	E7
D8	VDCCONT	CAPT_L	CAPT_H	CAPH_L	CAPH_H	CAPCONT	PI_KT_L	PI_KT_H	DF
D0	PSW	PFCON	-----	-----	FOC_D_L	FOC_D_H	FOCCONT	PI_TMSR	D7
C8	T2CON	ADCOS_L	ADCOS_H	-----	-----	-----	-----	-----	CF
C0	IRCON1	ADCSTR	ADCCONT	ADCD1	ADCD2	ADCDLY	-----	PINCONG7	C7
B8	IEN1	IP1	IICRWD	SMO_D1_L	SMO_D1_H	SMO_D2_L	SMO_D2_H	MPWMCPSF	BF
B0	P3	-----	PI_MIN_LMT_L	PI_MIN_LMT_H	TL2	TH2	WDTC	WDTK	B7
A8	IEN0	IP0	PI_TR_L	PI_TR_H	PI_OUT_L	PI_OUT_H	PI_FB_L	PI_FB_H	AF
A0	P2	USER_LPF_L	USER_LPF_H	EEPROM	INI_ANG_DAT	INI_ANG_CTRL	PI_CMD_L	PI_CMD_H	A7
98	SCON	SBUF	SRELL	SRELH	IICEBT	PI_UI_L	PI_UI_H	MOTOR_CONT2	9F
90	P1	PI_KI_L	PI_KI_H	PI_KP_L	PI_KP_H	PI_MAX_LMT_L	PI_MAX_LMT_H	MOTOR_CONT1	97
88	TCON	TMOD	TL0	TL1	TH0	TH1	AUX	SYNC	8F
80	P0	SP	DP0L	DP0H	DP1L	DP1H	RCON	PCON	87
	0	1	2	3	4	5	6	7	

SYMBOL	DIRECT ADDRESS		SFR_PAGE						
	[L,H]		0	1	2	3	4	5	6
MPWM									
MPWMDATA [L,H]	E1H	E2H	CYCLE	DUTY_U	DUTY_V	DUTY_W			
PI-Controller									
PI_KI[L,H]	91H	92H	IQ_KI	ID_KI	SPD_KI	PLL_KI	USER_KI		
PI_KP [L,H]	93H	94H	IQ_KP	ID_KP	SPD_KP	PLL_KP	USER_KP		
PI_KT [L,H]	DEH	DFH	IQ_KT	ID_KT	SPD_KT	PLL_KT	USER_KT		
PI_TR [L,H]	AAH	ABH	IQ_TR	ID_TR	SPD_TR	PLL_TR			
PI_MAX_LMT [L,H]	95H	96H	IQ_MAX	ID_MAX	SPD_MAX	PLL_MAX	USER_MAX		
PI_MIN_LMT [L,H]	B2H	B3H	IQ_MIN	ID_MIN	SPD_MIN	PLL_MIN	USER_MIN		
PI_CMD[L,H]	A6H	A7H	IQ_CMD	ID_CMD	SPD_CMD	PLL_CMD	USER_CMD		
PI_UI [L,H]	9DH	9EH	IQ_UI	ID_UI	SPD-UI	PLL-UI	USER_UI		
PI_OUT [L,H]	ACH	ADH	IQ_OUT	ID_OUT	SPD-OUT	PLL-OUT	USER_OUT		
PI_FB [L,H]	AEH	AFH	IQ_FB	ID_FB	SPD-FB	PLL-FB	USER_FB		
Slide Mode Controller									
SMO_D1 [L,H]	BBH	BCH	GS	SMO-gain	Angle base	Z-Correct	SMO-angle	BanBan-gain	
SMO_D2 [L,H]	BDH	BEH	FS	SMO-filter	----	MAXSMC-Err	SPEEDER		
Field Oriented Controller									
FOC_D[L,H]	D4H	D5H	VQ_OFSET	VD_OFSET		AS	CPU_ANG	FOC-angle	SVPWM-Amp
Independent General Low Pass Filter									
GEN_LPF[L,H]	A1H	A2H	LPF_GAIN	LPF_IN	LPF_OUT	LPF_PROUT			
ADC Offset Value									

ADCOS [L,H]	C9H	CAH	ADC1OS	ADC2OS					
Initial Angle Estimated Controller									
INI_ANG_CTRL	A4H		Pattern 10	Pattern 32	Pattern 54				
EEPROM									
EEPROM	A3H		EE_ADDR	EE_DATA	EE_CMD				
CAPTURE									
CAPCONT	DDH		E_CAPCONT	I_CAPCONT					
CAPT[L,H]	D9H	DAH	EXT_CAPT	INT_CAPT					
CAPH[L,H]	DBH	DCH	EXT_CAPH	INT_CAPH					

10. CGF022A SFRs and Reset Value

SYMBOL	DESCRIPTION	DIRECT ADDRESS	RESET VALUE
ACC	Accumulator	E0H	00H
ADCSTR	ADC Start Convert and Setting Register	C1H	00H
ADCCONT	ADC Control Register	C2H	83H
ADCD1	ADC Data Register 1	C3H	00H
ADCD2	ADC Data Register 2	C4H	00H
ADCDLY	ADC Sample Delay	C5H	33H
ADCOS_L	ADC Data Offset Low byte	C9H	00H
ADCOS_H	ADC Data Offset High byte	CAH	02H
AUX	Auxiliary	8EH	11H
AOCPCONT	Analog OCP Control Register	EEH	C7H
B	B Register	F0H	00H
CAPCONT	Capture Control Register	DDH	03H
CAPT_L	Capture Total Count Low	D9H	00H
CAPT_H	Capture Total Count High	DAH	00H
CAPH_L	Capture High-level Count Low	DBH	00H
CAPH_H	Capture High-level Count High	DCH	00H
DPTR0:	Data Pointer 0 (2 bytes)	DCH	FFH
DP0L	Data Pointer 0 Low	82H	00H
DP0H	Data Pointer 0 High	83H	00H
DPTR1	Data Pointer 1 (2 bytes)	E7H	00H
DP1L	Data Pointer 1 Low	84H	00H
DP1H	Data Pointer 1 High	85H	00H
EEPROM	D&Q -Axis Voltage Offset Low byte	A3H	00H
FOCCONT	Field Oriented Control Register	D6H	00H
FOC_D_L	Field Oriented Control Data Low byte	D4H	00H
FOC_D_H	Field Oriented Control Data High byte	D5H	00H
IEN0	Interrupt Enable Register 0	A8H	00H
IEN1	Interrupt Enable Register 1	B8H	00H
IP0	Interrupt Priority Register 0	A9H	00H
IP1	Interrupt Priority Register 1	B9H	00H
IRCON1	Interrupt Request Register 1	C0H	00H
IICS	IIC Status Register	E8H	00H
IICCTRL	IIC Control Register	E9H	04H
IICA1	IIC Address 1 Register	EAH	A0H
IICA2	IIC Address 2 Register	EBH	60H
IICRWD	IIC Read Write Register	BAH	00H
IICEBT	IIC Enable Bus Transaction Register	9CH	00H
INI_ANG_DAT	Initial Angle Estimated Data Register	A4H	EBH
INI_ANG_CTRL	Initial Angle Estimated Control Register	A5H	18H
MOTOR_CONT1	Motor Control Register 1	97H	00H
MOTOR_CONT2	Motor Control Register 2	9FH	A4H
MPWMCONT1	MPWM Control Register 1	E3H	00H
MPWMDATL	MPWM Data Low	E1H	00H
MPWMDATH	MPWM Data High	E2H	00H
MPWMDT	Motor PWM Dead-Time Register	E5H	78H
MPWMINV	MPWM Inverse Selection Register	E4H	00H
MPWMCONT2	MPWM Control Register 2	E7H	00H
MPWMCPSF	Motor PWM Compensation Factor Register	BFH	00H
OCPNCONT	OCP Control Register	EFH	85H
P0	Port 0	80H	FFH
P1	Port 1	90H	FFH
P2	Port 2	A0H	FFH
P3	Port 3	B0H	FFH

PCON	Power Control Register	87H	00H
PSW	Program Status Word Register	D0H	00H
PFCON	Peripheral Frequency Control Register	D1H	00H
PINCONG1	Pin Configure Register 1	F8H	AAH
PINCONG2	Pin Configure Register 2	F9H	AAH
PINCONG3	Pin Configure Register 3	FAH	A0H
PINCONG4	Pin Configure Register 4	FBH	AAH
PINCONG5	Pin Configure Register 5	FCH	AAH
PINCONG6	Pin Configure Register 6	FDH	A0H
PINCONG7	Pin Configure Register 7	C7H	0AH
PINSET1	Pin I/O Setting Register 1	F1H	AAH
PINSET2	Pin I/O Setting Register 2	F2H	AAH
PINSET3	Pin I/O Setting Register 3	F3H	0AH
PINSET4	Pin I/O Setting Register 4	F4H	00H
PINSET5	Pin I/O Setting Register 5	F5H	80H
PINSET6	Pin I/O Setting Register 6	F6H	0AH
PINSET7	Pin I/O Setting Register 7	F7H	B0H
PI_GAIN	PI Control KPx16 Select Register	E6H	F7H
PI_KT_L	PI-control KT Data Low byte	DEH	00H
PI_KT_H	PI-control KT Data High byte	DFH	00H
PI_TMSR	PI-control Tracking Mode Select Register	D7H	00H
PI_MIN_LMT_L	PI-control Minimal Limit Data Low byte	B2H	01H
PI_MIN_LMT_H	PI-control Minimal Limit Data High byte	B3H	80H
PI_TR_L	PI-control TR Data Low byte	AAH	00H
PI_TR_H	PI-control TR Data High byte	ABH	00H
PI_OUT_L	PI-control Output Data Low byte	ACH	00H
PI_OUT_H	PI-control Output Data High byte	ADH	00H
PI_FB_L	PI-control Feedback Data Low byte	AEH	00H
PI_FB_H	PI-control Feedback Data High byte	AFH	00H
PI_CMD_L	PI-control Command Data Low byte	A6H	00H
PI_CMD_H	PI-control Command Data High byte	A7H	00H
PI_UI_L	PI-control Integral Data Low byte	9DH	00H
PI_UI_H	PI-control Integral Data High byte	9EH	00H
PI_KI_L	PI-control KI Data Low byte	91H	00H
PI_KI_H	PI-control KI Data High byte	92H	00H
PI_KP_L	PI-control KP Data Low byte	93H	00H
PI_KP_H	PI-control KP Data High byte	94H	00H
PI_MAX_LMT_L	PI-control Minimal Limit Data Low byte	95H	FFH
PI_MAX_LMT_H	PI-control Minimal Limit Data High byte	96H	7FH
RCON	Internal RAM Control Register	86H	F0H
RSTS	Reset Source Register	FEH	0AH
SFR_PAGE	Special Function Registers Page	ECH	00H
SP_CYC	Speed Control Loop Cycle	EDH	26H
SCON	Serial Port Control Register	98H	00H
SBUF	Serial Port Data Buffer	99H	00H
SRELL	Serial Port Reload Register High	9AH	00H
SRELH	Serial Port Reload Register High	9BH	00H
SP	Stack Pointer	81H	07H
SMO_D1_L	Sliding Mode Observer Data1 Low Byte	BBH	----
SMO_D1_H	Sliding Mode Observer Data1 High Byte	BCH	----
SMO_D2_L	Sliding Mode Observer Data2 Low Byte	BDH	00H
SMO_D2_H	Sliding Mode Observer Data2 High Byte	BEH	00H
SYNC	MOC Sync Register	8FH	00H
T2CON	Timer 2 Control Register	C8H	00H
TAKEY	Time Access Key Register	FFH	00H
TCON	Timer 0/1 Control Register	88H	00H
TH0	Timer 0 High byte	8CH	00H
TH1	Timer 1 High byte	8DH	00H

TH2	Timer 2 High byte	B5H	00H
TL0	Timer 0 Low byte	8AH	00H
TL1	Timer 1 Low byte	8BH	00H
TL2	Timer 2 Low byte	B4H	00H
TMOD	Timer 0/1 Mode Register	89H	00H
GEN_LPF_L	User Low Pass Filter Data Low byte	A1H	00H
GEN_LPF _H	User Low Pass Filter Data High byte	A2H	00H
WDTC	Watchdog Timer Control Register	B6H	04H
WDTK	Watchdog Timer Refresh Key	B7H	00H

11. External Special Function Registers (XSFR)

XSFRs Memory Map

	8	9	A	B	C	D	E	F	
1078									107F
1070	ZALPHA_L	ZALPHA_H	ES_IALPHA_L	ES_IALPHA_H	ES_EALPHA_L	ES_EALPHA_H			1077
1068	ZBETA_L	ZBETA_H	ES_IBETA_L	ES_IBETA_H	ES_EBETA_L	ES_EBETA_H			106F
1060	IA_L	IB_H	IALPHA_L	IALPHA_H	VALPHA_L	VALPHA_H	VA_L(R)	VA_H(R)	1067
1058	IB_L	IB_H	IBETA_L	IBETA_H	VBETA_L	VBETA_H	VB_L(R)	VB_H(R)	105F
1050									1057
1048									104F
1040	IR_DOUT0	IR_DOUT1	IR_DOUT2	IR_DOUT3	IR_DOUT4	IR_DOUT5			1047
1038	IR_DEC_SET	IR_DEC_CTRL	IR_HEADER_Z1_L	IR_HEADER_Z1_H	IR_HEADER_Z2_L	IR_HEADER_Z2_H	IR_STOP_Z_L	IR_STOP_Z_H	103F
1030	MD_MODE	MD_CONT	MD0	MD1	MD2	MD3	MD4	MD5	1037
1028	SOFT_RST_KEY	SOFT_RST_EN							102F
1020	IPWM_CYC_L	IPWM_CYC_H	IPWM_DUTY_L	IPWM_DUTY_H	IPWM_CTRL			FGCTRL	1027
1018	CRC_CTRL	CRC_DIN	CRC_DOUT_L	CRC_DOUT_H	CRC_STR_BANK	CRC_END_BANK			101F
	0	1	2	3	4	5	6	7	

12. CGF022A XSFRs and Reset Value

SYMBOL	DESCRIPTION	DIRECT ADDRESS	RESET VALUE
CRC_CTRL	CRC Control Register	1018H	00H
CRC_DIN	CRC Input Data Register	1019H	00H
CRC_DOUT_L	CRC Output Remainder Data Register Low Byte	101AH	00H
CRC_DOUT_H	CRC Output Remainder Data Register High Byte	101BH	00H
CRC_STR_BANK	Start Bank Index for Flash Bank CRC Computation	101CH	00H
CRC_END_BANK	End Bank Index for Flash Bank CRC Computation	101DH	00H
FGCTRL	Function Generator Control Register	1027H	00H
IR_DOUT0	IR Decode Output Data Byte0	1040H	00H
IR_DOUT1	IR Decode Output Data Byte1	1041H	00H
IR_DOUT2	IR Decode Output Data Byte2	1042H	00H
IR_DOUT3	IR Decode Output Data Byte3	1043H	00H
IR_DOUT4	IR Decode Output Data Byte4	1044H	00H
IR_DOUT5	IR Decode Output Data Byte5	1045H	00H
IR_DEC_SET	IR Data Decode Setting Register	1038H	00H
IR_DEC_CTRL	IR Data Decode Control Register	1039H	00H
IR_HEADER_Z1_L	IR Data HEADER Zone1 Cycle Register Low Byte	103AH	80H
IR_HEADER_Z1_H	IR Data HEADER Zone1 Cycle Register High Byte	103BH	BBH
IR_HEADER_Z2_L	IR Data HEADER Zone2 Cycle Register Low Byte	103CH	00H
IR_HEADER_Z2_H	IR Data HEADER Zone2 Cycle Register High Byte	103DH	7DH
IR_STOP_Z_L	IR Data STOP Zone Cycle Register Low Byte	103EH	80H
IR_STOP_Z_H	IR Data STOP Zone Cycle Register High Byte	103FH	BBH
IPWM_CYC_L	Independent General PWM Cycle Low Byte	1020H	02H
IPWM_ CYC _H	Independent General PWM Cycle High Byte	1021H	00H
IPWM_DUTY_L	Independent General PWM Duty Low Byte	1022H	FFH
IPWM_ DUTY _H	Independent General PWM Duty High Byte	1023H	FFH
IPWM_CTRL	Independent General PWM Control Register	1024H	00H

MD_MODE	MDU Mode Control Register	1030H	10H
MD_CTRL	MDU Control Register	1031H	00H
MD0	Multiplication Division Register 0	1032H	00H
MD1	Multiplication Division Register 1	1033H	00H
MD2	Multiplication Division Register 2	1034H	00H
MD3	Multiplication Division Register 3	1035H	00H
MD4	Multiplication Division Register 4	1036H	00H
MD5	Multiplication Division Register 5	1037H	00H
SOFT_RST_KEY	Software Reset Key Register	1028H	00H
SOFT_RST_EN	Software Reset Enable Register	1029H	00H
ES_IALPHA_L(Read)	Estimate Current Iα Data Register Low Byte	1072H	00H
ES_IALPHA_H(Read)	Estimate Current Iα Data Register High Byte	1073H	00H
ES_EALPHA_L(Read)	Estimate EEMF Eα Data Register Low Byte	1074H	00H
ES_EALPHA_H(Read)	Estimate EEMF Eα Data Register High Byte	1075H	00H
ES_IBETA_L(Read)	Estimate Current Iβ Data Register Low Byte	106AH	00H
ES_IBETA_H(Read)	Estimate Current Iβ Data Register High Byte	106BH	00H
ES_EBETA_L(Read)	Estimate EEMF Eβ Data Register Low Byte	106CH	00H
ES_EBETA_H(Read)	Estimate EEMF Eβ Data Register High Byte	106DH	00H
IA_L(Read)	ADC Output of phase A current (Ia) Low Byte	1060H	00H
IA_H(Read)	ADC Output of phase A current (Ia) High Byte	1061H	00H
IALPHA_L(Read)	α-axis current (iα) Low Byte of CLARKE Transform	1062H	00H
IALPHA_H(Read)	α-axis current (iα) High Byte of CLARKE Transform	1063H	00H
IB_L(Read)	ADC Output of phase B current (Ib) Low Byte	1058H	00H
IB_H(Read)	ADC Output of phase B current (Ib) High Byte	1059H	00H
IBETA_L(Read)	β-axis current (iβ) Low Byte of CLARKE Transform	105AH	00H
IBETA_H(Read)	β-axis current (iβ) High Byte of CLARKE Transform	105BH	00H
VALPHA_L(Read)	α-axis Stator Voltage Data Register Low Byte	1064H	00H
VALPHA_H(Read)	α-axis Stator Voltage Data Register High Byte	1065H	00H
VA_L(Read)	Phase A Drive Voltage Data Register Low Byte	1066H	00H
VA_H(Read)	Phase A Drive Voltage Data Register High Byte	1067H	00H
VBETA_L(Read)	β-axis Stator Voltage Data Register Low Byte	105CH	00H
VBETA_H(Read)	β-axis Stator Voltage Data Register High Byte	105DH	00H
VB_L(Read)	Phase B Drive Voltage Data Register Low Byte	105EH	00H
VB_H(Read)	Phase B Drive Voltage Data Register High Byte	105FH	00H
ZALPHA_L(Read)	The Z gain of EEMF Eα Data Register Low Byte	1070H	00H
ZALPHA_H(Read)	The Z gain of EEMF Eα Data Register High Byte	1071H	00H
ZBETA_L(Read)	The Z gain of EEMF Eβ Data Register Low Byte	1068H	00H
ZBETA_H(Read)	The Z gain of EEMF Eβ Data Register High Byte	1069H	00H

13. Memory

The **CGF022A** memory structure follows the general 8052 structure.
There are three memory areas: Program Memory (Flash), External Data Memory (XRAM) and Internal Data Memory (IRAM). In addition, **CGF022A** integrates 16Kbytes Flash, 256bytes IRAM and 512bytes XRAM.

13.1. Program Memory

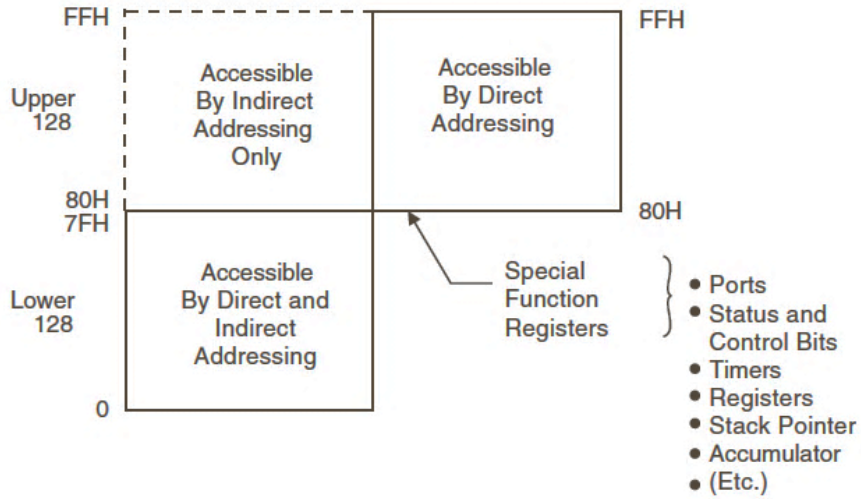
The **CGF022A** contains 16Kbytes of on-chip Flash memory for program storage.

13.2. Data Memory

The **CGF022A** contains 256bytes of general internal data memory (IRAM) and 512 bytes of external data memory (XRAM).

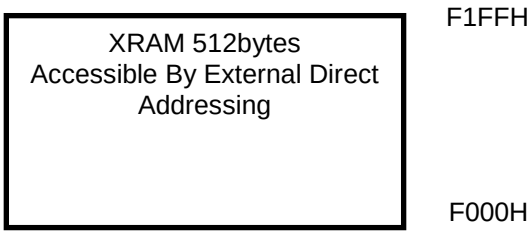
13.2.1 Data Memory (IRAM)(00H~FFH)

The lower 128 bytes of IRAM may be accessed through both direct and indirect addressing. The upper 128 bytes of IRAM and the 128 bytes of SFR registers share the same address space. The upper 128 bytes of data memory may only be accessed using indirect addressing. The SFR registers can only be accessed through direct addressing. The lowest 32 bytes (00H -1FH) of data memory are grouped into 4 banks of 8 registers each. The **RS0** and **RS1** bits (**PSW**.3 and**PSW**.4) select which register bank is in use. Instructions using register addressing will only access the currently specified bank.



13.2.2 Data Memory (XRAM)(F000H~F1FFH)

External addresses F000H to F1FFh contain the on-chip expanded SRAM. This memory can be accessed via external direct addressing mode (with **MOVX** instructions). The address space of instruction **MOVX** @Ri,A (i=0,1) is determined by **RCON** [7:0] of SFR 86H **RCON**(internal RAM control register). The default setting of **RCON** [7:0] is F0h (page0). One page of XRAM is512 bytes.



14. Instruction Set

The **CGF022A** is fully binary compatible with the MCS-51 instruction set.

Arithmetic operations	Description	Bytes	Cycles	Hex Code
ADD A,Rn	Add register to accumulator	1	1	0x28-0x2F
ADD A,direct	Add directly addressed data to accumulator	2	2	0x25
ADD A,@Ri	Add indirectly addressed data to accumulator	1	2	0x26-0x27
ADD A,#data	Add immediate data to accumulator	2	2	0x24
ADDC A,Rn	Add register to accumulator with carry	1	1	0x38-0x3F
ADDC A,direct	Add directly addressed data to accumulator with carry	2	2	0x35
ADDC A,@Ri	Add indirectly addressed data to accumulator with carry	1	2	0x36-0x37
ADDC A,#data	Add immediate data to accumulator with carry	2	2	0x34
SUBB A,Rn	Subtract register from accumulator with borrow	1	1	0x98-0x9F
SUBB A,direct	Subtract directly addressed data from accumulator with borrow	2	2	0x95
SUBB A,@Ri	Subtract indirectly addressed data from accumulator with borrow	1	2	0x96-0x97
SUBB A,#data	Subtract immediate data from accumulator with borrow	2	2	0x94
INC A	Increment accumulator	1	1	0x04
INC Rn	Increment register	1	2	0x08-0x0F
INC direct	Increment directly addressed location	2	3	0x05
INC @Ri	Increment indirectly addressed location	1	3	0x06-0x07
INC DPTR	Increment data pointer	1	1	0xA3
DEC A	Decrement accumulator	1	1	0x14
DEC Rn	Decrement register	1	2	0x18-0x1F
DEC direct	Decrement directly addressed location	2	3	0x15
DEC @Ri	Decrement indirectly addressed location	1	3	0x16-0x17
MUL AB	Multiply A and B	1	5	0xA4
DIV	Divide A by B	1	5	0x84
DA A	Decimally adjust accumulator	1	1	0xD4

Logic operations	Description	Bytes	Cycles	Hex Code
ANL A,Rn	AND register to accumulator	1	1	0x58-0x5F
ANL A,direct	AND directly addressed data to accumulator	2	2	0x55
ANL A,@Ri	AND indirectly addressed data to accumulator	1	2	0x56-0x57
ANL A,#data	AND immediate data to accumulator	2	2	0x54
ANL direct,A	AND accumulator to directly addressed location	2	3	0x52
ANL direct,#data	AND immediate data to directly addressed location	3	4	0x53
ORL A,Rn	OR register to accumulator	1	1	0x48-0x4F
ORL A,direct	OR directly addressed data to accumulator	2	2	0x45
ORL A,@Ri	OR indirectly addressed data to accumulator	1	2	0x46-0x47
ORL A,#data	OR immediate data to accumulator	2	2	0x44
ORL direct,A	OR accumulator to directly addressed location	2	3	0x42
ORL direct,#data	OR immediate data to directly addressed location	3	4	0x43
XRL A,Rn	Exclusive OR register to accumulator	1	1	0x68-0x6F
XRL A,direct	Exclusive OR directly addressed data to accumulator	2	2	0x65
XRL A,@Ri	Exclusive OR indirectly addressed data to accumulator	1	2	0x66-0x67
XRL A,#data	Exclusive OR immediate data to accumulator	2	2	0x64
XRL direct,A	Exclusive OR accumulator to directly addressed location	2	3	0x62
XRL direct,#data	Exclusive OR immediate data to directly addressed location	3	4	0x63
CLR A	Clear accumulator	1	1	0xE4
CPL A	Complement accumulator	1	1	0xF4
RL A	Rotate accumulator left	1	1	0x23
RLC A	Rotate accumulator left through carry	1	1	0x33
RR A	Rotate accumulator right	1	1	0x03
RRC A	Rotate accumulator right through carry	1	1	0x13
SWAP A	Swap nibbles within the accumulator	1	1	0xC4

Data transfer operations	Description	Bytes	Cycles	Hex Code
MOV A,Rn	Move register to accumulator	1	1	0xE8-0xEF
MOV A,direct	Move directly addressed data to accumulator	2	2	0xE5
MOV A,@Ri	Move indirectly addressed data to accumulator	1	2	0xE6-0xE7
MOV A,#data	Move immediate data to accumulator	2	2	0x74
MOV Rn,A	Move accumulator to register	1	2	0xF8-0xFF
MOV Rn,direct	Move directly addressed data to register	2	4	0xA8-0xAF
MOV Rn,#data	Move immediate data to register	2	2	0x78-0x7F
MOV direct,A	Move accumulator to direct	2	3	0xF5
MOV direct,Rn	Move register to direct	2	3	0x88-0x8F
MOV direct1,direct2	Move directly addressed data to directly addressed location	3	4	0x85
MOV direct,@Ri	Move indirectly addressed data to directly addressed location	2	4	0x86-0x87
MOV direct,#data	Move immediate data to directly addressed location	3	3	0x75
MOV @Ri,A	Move accumulator to indirectly addressed location	1	3	0xF6-0xF7
MOV @Ri,direct	Move directly addressed data to indirectly addressed location	2	5	0xA6-0xA7
MOV @Ri,#data	Move immediate data to in directly addressed location	2	3	0x76-0x77
MOV DPTR,#data16	Load data pointer with a 16-bit immediate	3	3	0x90
MOVC A,@A+DPTR	Load accumulator with a code byte relative to DPTR	1	3	0x93
MOVC A,@A+PC	Load accumulator with a code byte relative to PC	1	3	0x83
MOVX A,@Ri	Move external RAM (8-bit addr.) to accumulator	1	3	0xE2-0xE3
MOVX A,@DPTR	Move external RAM (16-bit addr.) to accumulator	1	3	0xE0
MOVX @Ri,A	Move accumulator to external RAM (8-bit addr.)	1	4	0xF2-0xF3
MOVX @DPTR,A	Move accumulator to external RAM (16-bit addr.)	1	4	0xF0
PUSH direct	Push directly addressed data onto stack	2	4	0xC0
POP direct	Pop directly addressed location from stack	2	3	0xD0
XCH A,Rn	Exchange register with accumulator	1	2	0xC8-0xCF
XCH A,direct	Exchange directly addressed location with accumulator	2	3	0xC5
XCH A,@Ri	Exchange indirect RAM with accumulator	1	3	0xC6-0xC7
XCHD A,@Ri	Exchange low-order nibbles of indirect and accumulator	1	3	0xD6-0xD7

Program branches	Description	Bytes	Cycles	Hex Code
ACALL addr11	Absolute subroutine call	2	6	xxx10001b
LCALL addr16	Long subroutine call	3	6	0x12
RET	Return from subroutine	1	4	0x22
RETI	Return from interrupt	1	4	0x32
AJMP addr11	Absolute jump	2	3	xxx00001b
LJMP addr16	Long jump	3	4	0x02
SJMP rel	Short jump (relative address)	2	3	0x80
JMP @A+DPTR	Jump indirect relative to the DPTR	1	2	0x73
JZ rel	Jump if accumulator is zero	2	3	0x60
JNZ rel	Jump if accumulator is not zero	2	3	0x70
JC rel	Jump if carry flag is set	2	3	0x40
JNC	Jump if carry flag is not set	2	3	0x50
JB bit,rel	Jump if directly addressed bit is set	3	4	0x20
JNB bit,rel	Jump if directly addressed bit is not set	3	4	0x30
JBC bit,rel	Jump if directly addressed bit is set and clear bit	3	4	0x10
CJNE A,direct,rel	Compare directly addressed data to accumulator and jump if not equal	3	4	0xB5
CJNE A,#data,rel	Compare immediate data to accumulator and jump if not equal	3	4	0xB4
CJNE Rn,#data,rel	Compare immediate data to register and jump if not equal	3	4	0xB8-0xBF
CJNE @Ri,#data,rel	Compare immed. to ind. and jump if not equal	3	4	0xB6-0xB7
DJNZ Rn,rel	Decrement register and jump if not zero	2	3	0xD8-0xDF
DJNZ direct,rel	Decrement directly addressed location and jump if not zero	3	4	0xD5
NOP	No operation	1	1	0

Boolean manipulation	Description	Bytes	Cycles	Hex Code
CLR C	Clear carry flag	1	1	0xC3
CLR bit	Clear directly addressed bit	2	3	0xC2
SETB C	Set carry flag	1	1	0xD3
SETB bit	Set directly addressed bit	2	3	0xD2
CPL C	Complement carry flag	1	1	0xB3
CPL bit	Complement directly addressed bit	2	3	0xB2
ANL C,bit	AND directly addressed bit to carry flag	2	2	0x82
ANL C,/bit	AND complement of directly addressed bit to carry	2	2	0xB0
ORL C,bit	OR directly addressed bit to carry flag	2	2	0x72
ORL C,/bit	OR complement of directly addressed bit to carry	2	2	0xA0
MOV C,bit	Move directly addressed bit to carry flag	2	2	0xA2
MOV bit,C	Move carry flag to directly addressed bit	2	3	0x92

15. MCU

15.1 8051 Engine

SFR	Description	Address	Reset value
ACC	Accumulator	E0H	00H
B	B Register	F0H	00H
PSW	Program Status Word Register	D0H	00H
SP	Stack Pointer	81H	07H
DP0H	Data Pointer 0 High	83H	00H
DP0L	Data Pointer 0 Low	82H	00H
DP1H	Data Pointer 1 High	85H	00H
DP1L	Data Pointer 1 Low	84H	00H
AUX	Auxiliary	8EH	11H
RCON	Internal RAM Control Register	86H	F0H

15.1.1 ACC (Accumulator)

The most important of all special function registers, that's the first comment about Accumulator which is also known as ACC or A. The Accumulator (sometimes referred to as Register A also) holds the result of most of arithmetic and logic operations.

ACC Address = E0H Reset Value = 0x00H

Accumulator

Bit	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

15.1.2 B (B Register)

The B register is used during multiplying and division instructions. It can also be used as a scratch-pad register to hold temporary data.

B Address = F0H Reset Value = 0x00H

B Register

Bit	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

15.1.3 PSW (Program Status Word Register)

The PSW register contains status bits that reflect the current state of the CPU. Note that the Parity bit can only be modified by hardware upon the state of ACC register.

PSW Address = D0H Reset Value = 0x00H

Program Status Word Register

Bit	CY	AC	F0	RS1	RS0	OV	F1	P
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

CY	Carry flag : Carry bit in arithmetic operations and accumulator for Boolean operations.
AC	Auxiliary Carry flag : Set if there is a carry-out from third bit of Accumulator in BCD Operations.
F0	General purpose Flag0 : General purpose flag available for user.
RS1	Register bank select control bit 1, used to select working register bank.
RS0	Register bank select control bit 0, used to select working register bank.
OV	Overflow flag : Set in case of overflow in Accumulator during arithmetic operations.
F1	General purpose Flag 1 : General purpose flag available for user.
P	Parity flag : Reflects the number of ‘1’s in the Accumulator. P = ‘1’ if Accumulator contains an odd number of ‘1’s P = ‘0’ if Accumulator contains an even number of ‘1’s

The state of RS1 and RS0 bits selects the working register bank as follows:

RS1	RS0	Selected Register Bank	Location
0	0	Bank 0	00H – 07H
0	1	Bank 1	08H – 0FH
1	0	Bank 2	10H – 17H
1	1	Bank 3	18H – 1FH

15.1.4 SP (Stack Pointer)

This register points to the top of stack in internal data memory space. It is used to store the return address of program before executing interrupt routine or subprograms. The SP is incremented before executing PUSH or CALL instruction and it is decremented after executing POP or RET(I) instruction (it always points the top of stack). A reset initializes the stack pointer to location 0x07. Therefore, the first value pushed on the stack is placed at location 0x08.

SP — Address = 81H — Reset Value = 0x07H

Stack Pointer

Bit	SP.7	SP.6	SP.5	SP.4	SP.3	SP.2	SP.1	SP.0
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

15.1.5 DP0 (Data Pointer 0)

These registers are intended to hold 16-bit address in the indirect addressing mode used by MOVX (move external memory), MOVC (move program memory) or JMP (computed branch) instructions. They may be manipulated as 16-bit register or as two separate 8-bit registers. DP0H holds higher byte and DP0L holds lower byte of indirect address.

It is generally used to access external code or data space, e.g.:

MOVCA,@A+DPTR (code space)

MOVA,@DPTR (data space)

DP0L — Address = 82H — Reset Value = 0x00H

Data Pointer 0 Low Byte

Bit	DP0L[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

DP0H — Address = 83H — Reset Value = 0x00H

Data Pointer 0 High Byte

Bit	DP0H[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

15.1.6 DP1 (Data Pointer 1)

The dual data pointer accelerates the movement of block data. The standard DPTR is a 16-bit register that is used to address external memory, or peripherals. The standard data pointer is called DPTR0 and the second data pointer is called DPTR1. The data pointer select bit chooses the active pointer. The data pointer select bit (DPS) is located in the LSB of AUX register (AUX.1).

The user switches between DPTR0 and DPTR1 by toggling the DPS bit. All DPTR-related instructions use the currently selected DPTR for any activity.

DP1L — Address = 84H — Reset Value = 0x00H

Data Pointer 1Low Byte

Bit	DP1L[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

DP1H — Address = 85H — Reset Value = 0x00H

Data Pointer 1 High Byte

Bit	DP1H[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

15.1.7 AUX(Auxiliary Register)

AUX Auxiliary Register — Address = 8EH — Reset Value = 0x11H

Bit	LVD_EN	LVD	-----	ITS	SMOD	BRS	DPS	CP
	7	6	5	4	3	2	1	0
Type	R/W	R	X	R/W	R/W	R/W	R/W	R

LVD_EN	Low voltage detect enable : 1: Enable
LVD	Low voltage detect status. : 1:Low voltage occur
ITS	MCU instruction timing select. : 0:1T 1:2T
SMOD	Serial Port (UART) baud rate select.
BRS	Serial Port (UART) baud rate generator select.
DPS	Data pointer register select : 0 : Select DPTR Register DP0H, DP0L 1 : Select DPTR Register DP1H, DP1L
CP	Code protect : 0 : Non-protect 1: Protect

15.1.8 RCON (Internal RAM Control Register)

256 bytes of on-chip expanded RAM are provided and can be accessed by external memory addressing method only (instruction MOVX). The address space of instruction MOVX @Ri,A (i= 0,1) is determined by RCON [7:0] of RCON. The default setting of RCON [7:0] is F0H.

RCON Internal RAM Control Register — Address = 86H — Reset Value = 0xF0H

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

15.2 GPIO

Four I/O ports are available: Port0, Port1, Port2, and Port3.
All 32port pins on CGF022A can configure to one of four modes : quasi-bidirectional (standard 8051 port outputs),push-pull output, open drain output, or input-only. All port pins default to input-only mode after reset. Two configuration registers (PINSETx, PINCONFGx) for each port select the output mode for each port pin.

SFR	Description	address	Reset value
P0	Port 0	80H	FFH
P1	Port 1	90H	FFH
P2	Port 2	A0H	FFH
P3	Port 3	B0H	FFH
PINCONG1	Pin Configure Register 1	F8H	AAH
PINCONG2	Pin Configure Register 2	F9H	AAH
PINCONG3	Pin Configure Register 3	FAH	A0H
PINCONG4	Pin Configure Register 4	FBH	AAH
PINCONG5	Pin Configure Register 5	FCH	AAH
PINCONG6	Pin Configure Register 6	FDH	A0H
PINCONG7	Pin Configure Register 7	C7H	0AH
PINSET1	Pin I/O Setting Register 1	F1H	AAH
PINSET2	Pin I/O Setting Register 2	F2H	AAH
PINSET3	Pin I/O Setting Register 3	F3H	0AH
PINSET4	Pin I/O Setting Register 4	F4H	00H
PINSET5	Pin I/O Setting Register 5	F5H	80H
PINSET6	Pin I/O Setting Register 6	F6H	0AH
PINSET7	Pin I/O Setting Register 7	F7H	B0H

15.2.1 Port

P0 Port 0	Address = 80H						Reset Value = 0XFFH	
Bit	-----	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
	7	6	5	4	3	2	1	0
Type	X	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P1 Port 1	Address = 90H						Reset Value = 0XFFH	
Bit	-----	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
	7	6	5	4	3	2	1	0
Type	X	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P2 Port 2	Address = A0H						Reset Value = 0XFFH	
Bit	P2.7	P2.6	P2.5	P2.4	P2.3	-----	-----	-----
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	X	X	X

P3 Port 3	Address = B0H						Reset Value = 0XFFH	
Bit	-----	-----	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

15.2.2 PINCONG (Pin Configure Register)

PINCONG1 Address = F8H Reset Value = 0xAAH

Pin Configure Register 1

Bit	CH4CONG[1:0]		CH5CONG[1:0]		CH6CONG[1:0]		CH7CONG[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : Quasi-bidirectional(standard 8051 port outputs)

01 : Push-pull output

10 : Input-only (High impedance)

11 : Open drain output

PINCONG2 Address = F9H Reset Value = 0xAAH

Pin Configure Register 2

Bit	CH0P CONG[1:0]		CH1P CONG[1:0]		CH2CONG[1:0]		CH3CONG[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : Quasi-bidirectional(standard 8051 port outputs)

01 : Push-pull output

10 : Input-only (High impedance)

11 : Open drain output

PINCONG3 Address = FAH Reset Value = 0xA0H

Pin Configure Register 3

Bit	XCONG[1:0]		UCONG[1:0]		XTALO CONG[1:0]		XTALI CONG[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : Quasi-bidirectional(standard 8051 port outputs)

01 : Push-pull output

10 : Input-only (High impedance)

11 : Open drain output

PINCONG4 Address = FBH Reset Value = 0xAAH

Pin Configure Register 4

Bit	ZCONG[1:0]		WCONG[1:0]		YCONG[1:0]		VCONG[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : Quasi-bidirectional(standard 8051 port outputs)

01 : Push-pull output

10 : Input-only (High impedance)

11 : Open drain output

PINCONG5 Address = FCH Reset Value = 0xAAH

Pin Configure Register 5

Bit	OCPNCONG[1:0]		CH1N CONG[1:0]		CH0N CONG [1:0]		AOCPCONG [1:0]	
	7	6	5	4	3	2	1	0
Ty	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : Quasi-bidirectional(standard 8051 port outputs)

01 : Push-pull output

10 : Input-only (High impedance)

11 : Open drain output

PINCONG6 Address = FDH Reset Value = 0xA0H

Pin Configure Register 6

Bit	IIC_SCLCONG [1:0]		IIC_SDACONG [1:0]		RXCONG[1:0]		TXCONG[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : Quasi-bidirectional(standard 8051 port outputs)

01 : Push-pull output

10 : Input-only (High impedance)

11 : Open drain output

PINCONG7 Address = C7H Reset Value = 0x0AH

Pin Configure Register 7

Bit	-----		-----		FGCONG [1:0]		IPWMCONG [1:0]	
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

00 : Quasi-bidirectional(standard 8051 port outputs)

01 : Push-pull output

10 : Input-only (High impedance)

11 : Open drain output

15.2.3 PINSET (Pin I/O Setting Register)

PINSET1 Address = F1H Reset Value = 0xAAH

Pin I/O Setting Register 1

Bit	CH4SET[1:0]		CH5SET[1:0]		CH6SET[1:0]		CH7SET[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : No pull

01 : Pull down

10 : Pull up

11 : No pull

PINSET2 Address = F2H Reset Value = 0xAAH

Pin I/O Setting Register 2

Bit	CH0PSET[1:0]		CH1P SET[1:0]		CH2SET[1:0]		CH3SET[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : No pull

01 : Pull down

10 : Pull up

11 : No pull

PINSET3 Address = F3H Reset Value = 0x0AH

Pin I/O Setting Register 3

Bit	XSET[1:0]		USET[1:0]		XTALO SET[1:0]		XTALI SET[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : No pull

01 : Pull down

10 : Pull up

11 : No pull

PINSET4 Address = F4H Reset Value = 0x00H

Pin I/O Setting Register 4

Bit	ZSET[1:0]		WSET[1:0]		YSET[1:0]		VSET[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : No pull

01 : Pull down

10 : Pull up

11 : No pull

PINSET5 Address = F5H Reset Value = 0x80H

Pin I/O Setting Register 5

Bit	OCPNSET[1:0]		CH1N SET[1:0]		CH0N SET [1:0]		AOCPSET[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : No pull

01 : Pull down

10 : Pull up

11 : No pull

PINSET6 Address = F6H Reset Value = 0x0AH

Pin I/O Setting Register 6

Bit	IIC_SCLSET[1:0]		IIC_SDASET[1:0]		RXSET[1:0]		TXSET [1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

00 : No pull

01 : Pull down

10 : Pull up

11 : No pull

PINSET7 Address = F7H Reset Value = 0xB0H

Pin I/O Setting Register 7

Bit	MDES	----	OCPNDBT[1:0]		FGSET4[1:0]		IPWMSET[1:0]	
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MDES

0 : U、V、W、X、Y、Z is GPIO Mode.

1: U、V、W、X、Y、Z is SVPWM Mode.(ADC CH0 & CH1 Auto Converter)

MDES=1, Pin set must to Push-pull output mode. (PINCONG3 [7:4] and PINCONG4 [7:0])

DOCPNDBT (De-bounce time for digital OCPN)

00 : 0nS

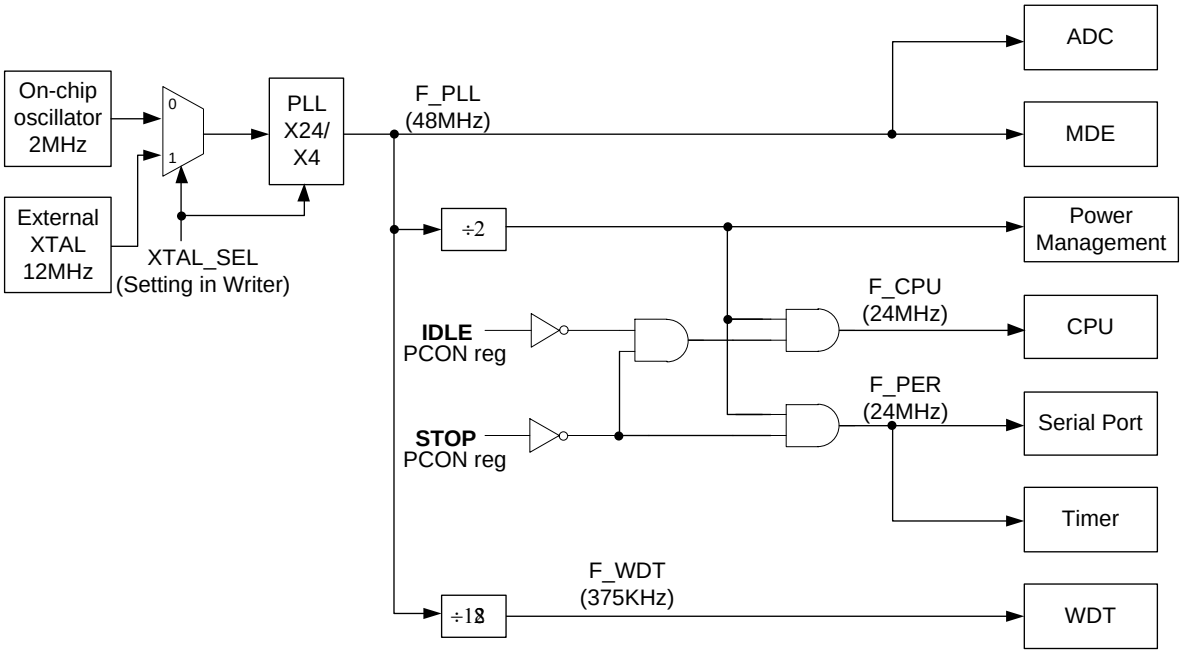
01 : 250nS

10 : 500nS

11 : 1000nS

15.3 Clock Structure

The clock source of the device may be either external, or internal. The external crystal (12MHz) is connect to pins XTALI and XTALO. The internal clock source (on-chip oscillator) is run at 2MHz. The choice of internal, or external, clock source is setting by Writer.



15.4 Timer

The CGF022A has three 16-bit timer/counter registers: Timer0, Timer1 and Timer2. All can be configured for counter, or timer, operations.
In addition to the “timer” or “counter” selection, Timer0 and Timer1 have four operating modes from which to select which are selected by bit-pairs (M1, M0) in TMOD. Modes 0, 1,and 2 are the same for both timer/counters. Mode 3 is different.

	Timer0	Timer1	Timer2
Mode 0	13-bit timer/counter	13-bit timer/counter	13-bit timer/counter
Mode 1	16-bit timer/counter	16-bit timer/counter	16-bit timer/counter
Mode 2	8-bit auto-reload timer/counter	8-bit auto-reload timer/counter	8-bit auto-reload timer/counter
Mode 3	two independent 8-bit timers/counters	stop	8-bit timers/counters

Two Special Function registers (TMOD and TCON) are used to select the appropriate mode.

SFR	Description	address	Reset value
PFCON	Peripheral Frequency Control Register	D1H	00H
TMOD	Timer 0/1 Mode Register	89H	00H
TCON	Timer 0/1 Control Register	88H	00H
T2CON	Timer2 Control Register	C8H	00H
TH0	Timer0 High byte	8CH	00H
TL0	Timer0 Low byte	8AH	00H
TH1	Timer1 High byte	8DH	00H
TL1	Timer1 Low byte	8BH	00H
TH2	Timer2 High byte	B5H	00H
TL2	Timer2 Low byte	B4H	00H

15.4.1 PFCON (Peripheral Frequency Control Register)

PFCON Address = D1H Reset Value = 0x00H

Peripheral Frequency Control Register

Bit	-----	-----	SRELPS[1:0]		T1PS[1:0]		T0PS[1:0]	
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

SRELPS[5:4]	Serial port (UART) Prescaler select : 00 :F_PER/64 01 :F_PER/32 10 :F_PER/16 11 :F_PER/8
T1PS[3:2]	Timer1(T1) Prescaler select : 00 :F_PER/12 01 :F_PER 10 :F_PER/96 11 :-----
T0PS[1:0]	Timer0(T0) Prescaler select : 00 :F_PER/12 01 :F_PER 10 :F_PER/96 11 :-----

15.4.2 TMOD (Timer 0/1 Mode Register)

TMOD register is used in configuration of MCUTimer0 and Timer1.

TMOD Address = 89H Reset Value = 0x00H

Timer 0/1 Mode Register

Bit	GATE1	C/T1	T1M1	T1M0	GATE0	C/T0	T0M1	T0M0
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GATE1	Timer1 gate control :
[7]	If set, enables external gate control (pin INT1N) for Counter1. When INT1N is high, and TR1 bit is set, the Counter1 is incremented every falling edge on INT1N input pin
C/T1	Timer1 counter/timer select :
[6]	0 : Timer
	1 : Counter
GATE0	Timer 0 gate control :
[3]	If set, enables external gate control (pin INT0N) for Counter0. When INT0N is high, and TR0 bit is set, the Counter0 is incremented every falling edge onINT0N input pin
C/T0	Timer0 counter/timer select :
[2]	0 : Timer
	1 : Counter

T1M1 /T0M1	T1M0 /T0M0	Mode	Function
0	0	Mode0	13-bit Counter/Timer, with 5 lower bits in TL0 (TL1) register and 8 bits in TH0 (TH1) register (for Timer0 or Timer1, respectively). The 3 high-order bits of TL0 (TL1) are zeroed whenever Mode 0 is enabled. (Not auto-reload)
0	1	Mode1	16-bit Counter/Timer. (Not auto-reload)
1	0	Mode2	8 -bit auto-reload Counter/Timer. The reload value is kept in TH0 (TH1), while TL0 (TL1) is incremented every clock cycle. Reloaded from TH0 (TH1) at overflow.
1	1	Mode3	For Timer1: Timer1 is stopped. For Timer0: Timer0 acts as two independent 8 bit Timers / Counters – TL0, TH0. (Not auto-reload)

15.4.3 TCON (Timer 0/1Control Register)

TCON register is used to control operation of these modules.CGF022A includes two external digital interrupt sources INT0N and INT1N), with dedicated interrupt sources. INT0N and INT1N are configurable as falling edge or low level. The IT0 and IT1 bits in TCON select level- or edge-sensitive.IE0 and IE1 in the TCON register serve as the interrupt-pending flags for the INT0N and INT1N external interrupts, respectively.

TCON Address = 88H Reset Value = 0x00H

Timer 0/1 Control Register

Bit	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TF1	Timer1 overflow flag : Bit set by hardware when Timer1 overflows. This flag can be cleared by software and is automatically cleared when interrupt is processed.
TR1	Timer1 run control : 0 : Stop 1 : Run
TF0	Timer0 overflow flag : Bit set by hardware when Timer1 overflows. This flag can be cleared by software and is automatically cleared when interrupt is processed.
TR0	Timer0run control : 0 : Stop 1 : Run
IE1	External interrupt 1 flag : Set by hardware, when External interrupt (INT1N) is observed. Cleared by hardware when interrupt is processed.
IT1	External interrupt 1 type control : 0 : External interrupt 1 is activated at low level on input pin 1 : External interrupt 1 is activated at falling edge on input pin
IE0	External interrupt 0 flag : Set by hardware, when External interrupt (INT0N) is observed. Cleared by hardware when interrupt is processed.
IT0	External interrupt 0 type control : 0 : External interrupt 0 is activated at low level on input pin 1 : External interrupt 0 is activated at falling edge on input pin

The TF0, TF1 (Timer0 andTimer1 overflow flags), IE0 and IE1 (External interrupt 0 and 1 flags) will be automatically cleared by hardware when the corresponding service routine is called.

15.4.4 T2CON (Timer2 Control Register)

T2CON is used to control Timer2 run/stop, mode, prescaler.

T2CON Address = C8H Reset Value = 0x00H

Timer2 Control Register

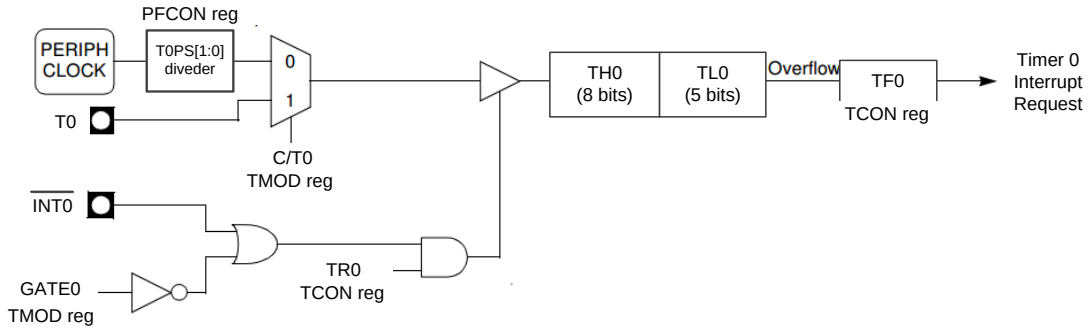
Bit	-----	-----	TF2	TR2	T2M1	T2M0	T2PS1	T2PS0
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

TF2	Timer2 overflow flag :

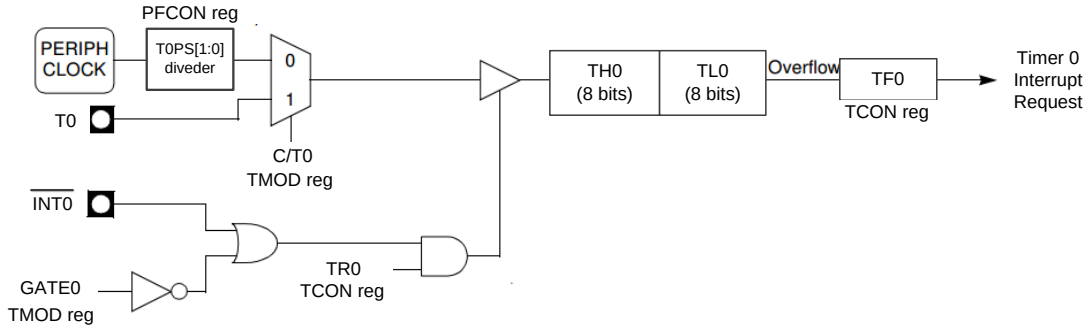
	Bit set by hardware when Timer2 overflows. This flag can be cleared by software and is automatically cleared when interrupt is processed.
TR2	Timer2run control : 0 : Stop 1 : Run
T2PS[1:0]	Timer2(T2) Prescaler select : 00 :F_PER/12 01 : F_PER 10 : F_PER/96 11 :-----

T2M1	T2M0	Mode	Function
0	0	Mode0	13-bit Timer, with 5 lower bits in TL2 register and 8 bits in TH2 register.(Not auto-reload)
0	1	Mode1	16-bit Timer. (Not auto-reload)
1	0	Mode2	8 -bit auto-reload Timer. The reload value is kept in TH2, while TL2 is incremented every clock cycle. Reloaded from TH2 at overflow.
1	1	Mode3	8 bit Timers. (Not auto-reload)

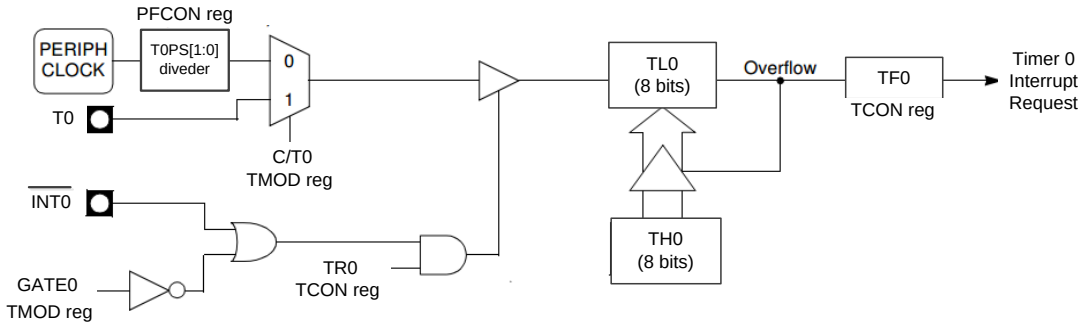
15.4.5 Timer0 Mode 0



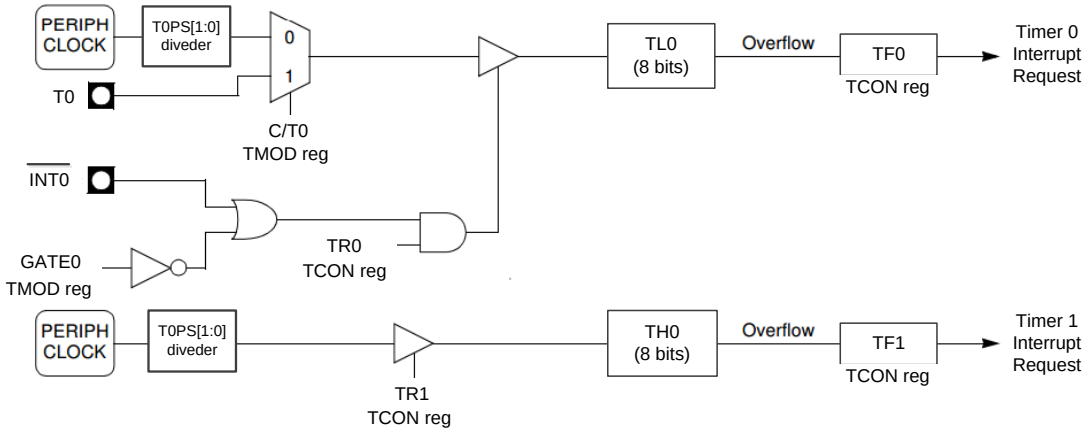
15.4.6 Timer0 Mode 1



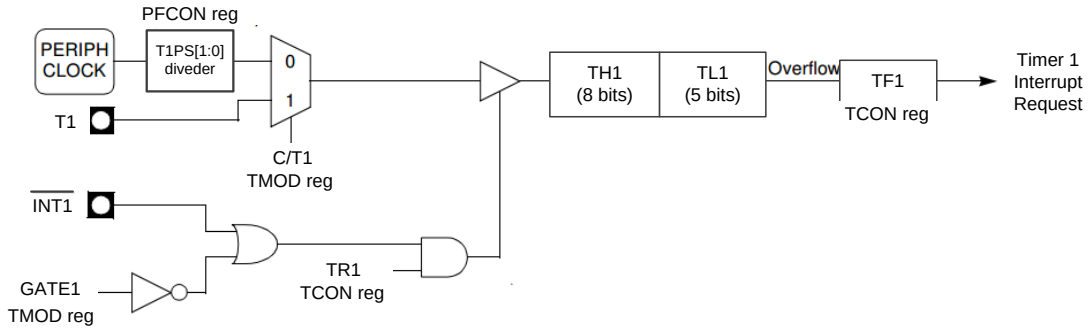
15.4.7 Timer0 Mode 2



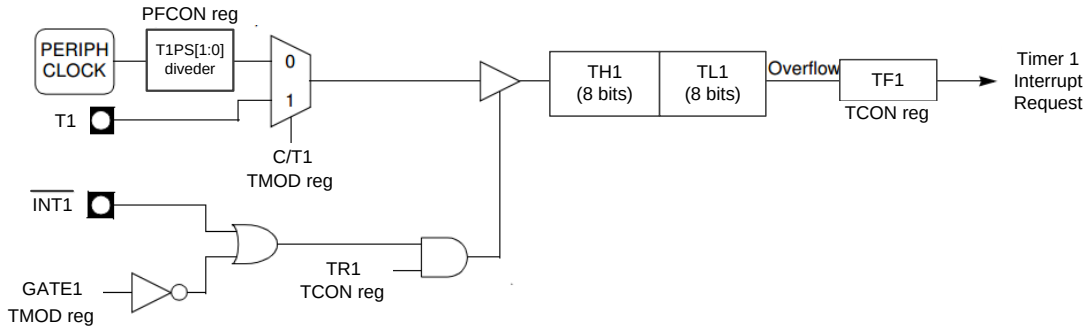
15.4.8 Timer0 Mode 3



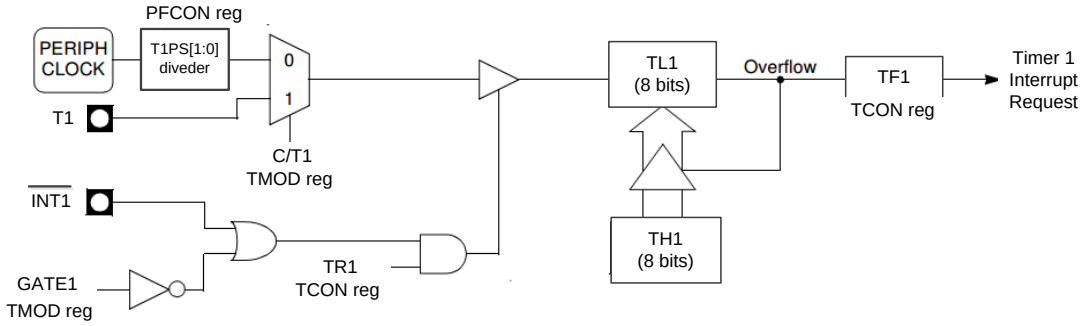
15.4.9 Timer1 Mode 0



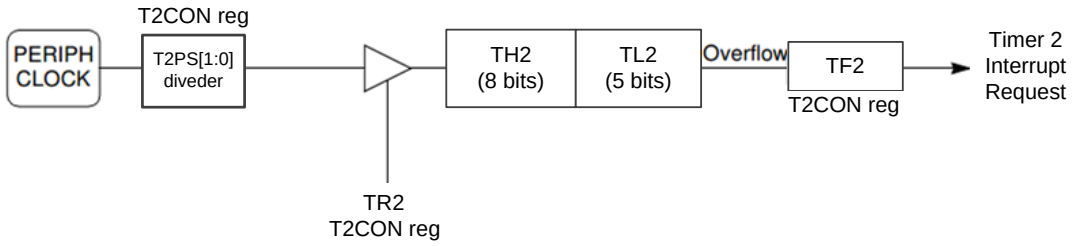
15.4.10 Timer1 Mode 1



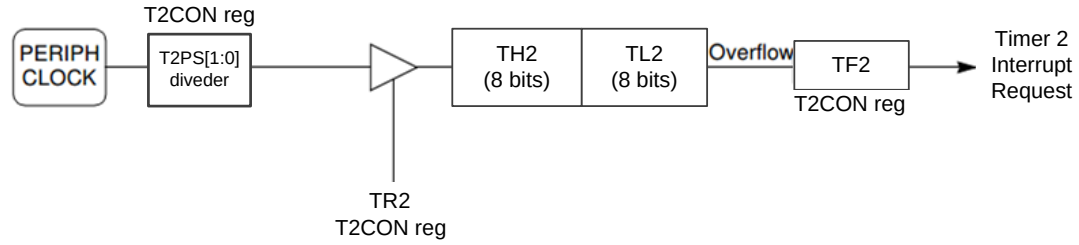
15.4.11 Timer1 Mode 2



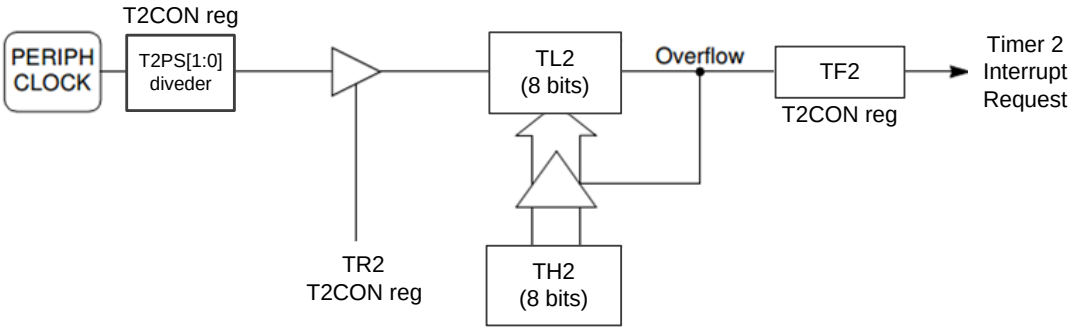
15.4.12 Timer2 Mode 0



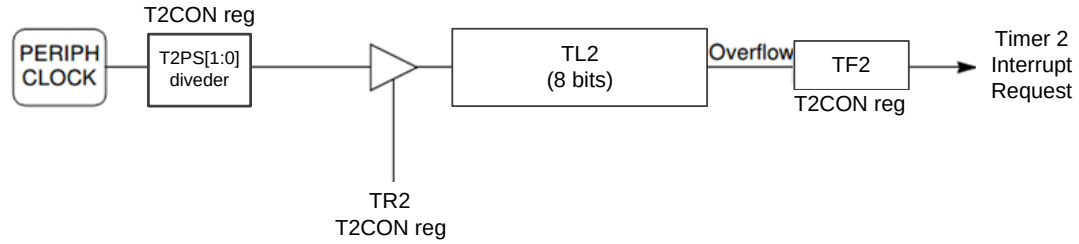
15.4.13 Timer2 Mode 1



15.4.14 Timer2 Mode 2



15.4.15 Timer2 Mode 3



15.5 Watchdog Timer

The Watchdog Timer (WDT) is a8-bit free-running counter that generates a reset signal or interrupt (WDTC.6) if it overflows. It can help the application software to recover from an abnormal condition. The WDT is independent from Timer0, Timer1, or Timer2. The F_WDT is 375KHz, it is from on-chip RC oscillator.

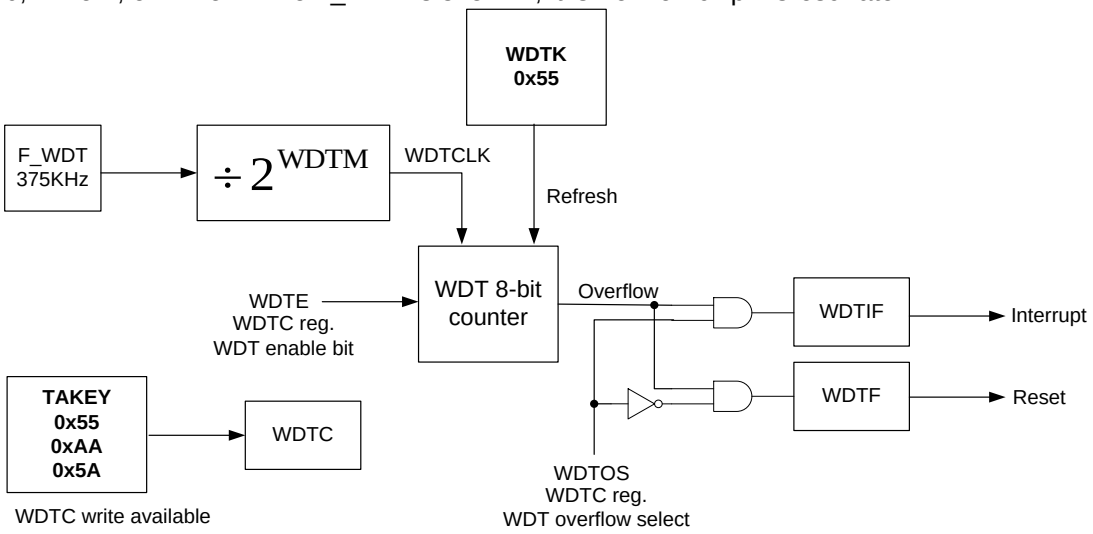


Figure 12.5.1 WDT block diagram

WDTCLK = F_WDT x 1 / 2^{WDTM}

WDT (8-bit counter) overflow time = 256 / WDTCLK

SFR	Description	address	Reset value
RSTS	Reset Source Register	FEH	0AH
TAKEY	Time Access Key Register	FFH	00H
WDTC	Watchdog Timer Control Register	B6H	04H
WDTK	Watchdog Timer Refresh Key	B7H	00H

RSTS Address = FEH Reset Value = 0x0AH

Reset Source Register

Bit	-----	-----	-----	WDTRF	PINRF[1:0]		PORF[1:0]	
	7	6	5	4	3	2	1	0
Type	X	X	X	R/W	R/W	R/W	R/W	R/W

WDTRF	Watchdog timer reset flag. This flag is set to 1 if the Watchdog Timer Reset caused the reset.
PINRF[1:0]	RSTN pin reset flag. This flag is set to 10b if the RSTN pin caused the reset. Clear by firmware.
PORF[1:0]	POR reset flag. This flag is set to 10b if the POR caused the reset. Clear by firmware.

15.5.1 WDTC (Watchdog Timer Control Register)

WDTC Address = B6H Reset Value = 0x04H

Watchdog Timer Control Register

Bit	-----	WDTOS	WDTE	-----	WDTM[3:0]			
	7	6	5	4	3	2	1	0
Type	X	R/W	R/W	X	R/W	R/W	R/W	R/W

WDTOS	Watchdog timer overflow select : 0 : When WDT overflow, enable WDT reset. 1 : When WDT overflow, enable WDT interrupt.
WDTE	Watchdog timer enable : 0 : Disable WDT. 1 : Enable WDT.
WDTM[3:0]	WDT clock divider : WDTCLK = 375KHz x 1 / 2^{WDTM} (default is 375K / 16)

15.5.2 TAKEY (Time Access Key Register)

TAKEY Address = FFH Reset Value = 0x00H

Time Access Key Register

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

WDTC default is read only, must write three specific values 55H, AAH and 5AH to the TAKEY enable the WDTC write available.
The sequence is:
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah

15.5.3 WDTK (Watchdog Timer Refresh Key)

WDTK Address = B7H Reset Value = 0x00H

Watchdog Timer Refresh Key

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

The application must write 0x55 into the WDTK register, for the Watchdog timer to be cleared.

For example, enable the watchdog with a time-out reset period of 5.461ms.

Following write sequence:
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah ; WDTC write is available.
MOV WDTC, #23h ; WDTM [3:0] = 0111b. WDTE =1 to enable the WDT.
MOV WDTK, #55h ;Refresh WDT.

15.6 Serial Port (UART)

The Serial Port provides a flexible full-duplex synchronous/asynchronous receiver/transmitter, called UART. The communication rate can be set by configuring the baud rate in SFRs. The two serial buffers consist of two separate registers, a transmit buffer and a receive buffer. Writing data to the SFR SBUF, transfers the data to the serial output buffer and starts the transmission. Reading from the SBUF, reads data from the serial receive buffer. The serial port can simultaneously transmit and receive data. It can also buffer 1 byte at receive, which prevents the receive data from being lost if the CPU reads the second byte before the transmission of the first byte is completed.

SFR	Description	address	Reset value
AUX	Auxiliary	8EH	11H
PFCON	Peripheral Frequency Control Register	D1H	00H
SCON	Serial Port Control Register	98H	00H
SBUF	Serial Port Data Buffer	99H	00H
SRELH	Serial Port Reload Register High	9BH	00H
SRELL	Serial Port Reload Register Low	9AH	00H

AUX Address = 8EH Reset Value = 0x11H

Auxiliary Register

Bit	LVD_EN	LVD	-----	ITS	SMOD	BRS	DPS	CP
	7	6	5	4	3	2	1	0
Type	R/W	R	X	R/W	R/W	R/W	R/W	R

LVD_EN	Low voltage detect enable : 1: Enable
LVD	Low voltage detect status. : 1: Low voltage occur
ITS	MCU instruction timing select. : 0:1T 1:2T
SMOD	Serial Port (UART) baud rate select.
BRS	Serial Port (UART) baud rate generator select.
DPS	Data pointer register select : 0 : Select DPTR Register DP0H, DP0L 1 : Select DPTR Register DP1H, DP1L
CP	Code protect : 0 : Non-protect 1: Protect

PFCON Address = D1H Reset Value = 0x00H

Peripheral Frequency Control Register

Bit	-----	-----	SRELPS[1:0]		T1PS[1:0]		T0PS[1:0]	
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

SRELPS[1:0]	Serial port (UART) Prescaler select : 00 :F_PER/64 01 :F_PER/32 10 :F_PER/16 11 :F_PER/8
T1PS[1:0]	Timer1(T1) Prescaler select : 00 :F_PER/12 01 :F_PER 10 :F_PER/96 11 :-----
T0PS[1:0]	Timer0(T0) Prescaler select : 00 : F_PER/12 01 : F_PER 10 : F_PER/96 11 :-----

15.6.1 SCON (Serial Port Control Register)

The SCON register controls the function of Serial Port (UART).

SCON Address = 98H Reset Value = 0x00H

Serial Port Control Register

Bit	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SM0	SM1	Mode	Description	Baud Rate	
0	0	Mode 0	Shift register	F_PER/12	
0	1	Mode 1	8bit UART	Variable	
1	0	Mode 2	9bit UART	Depends on SMOD (AUX.3)	
				SMOD	Baud Rate
				0	F_PER/64
				1	F_PER/32
1	1	Mode 3	9bit UART	Variable	

SM2	Multiprocessor communication enable
REN	Serial reception enable : 0 : Serial reception at Serial Port is disabled. 1 :Serial reception at Serial Port is enabled.
TB8	Transmitter bit 8 : This bit is used while transmitting data through Serial Port in Modes 2 and 3. The state of this bit corresponds with the state of the 9th transmitted bit (e.g. parity check or multiprocessor communication). It is controlled by software.
RB8	Received bit 8 : This bit is used while receiving data through Serial Port 0 in Modes 2 and 3. It reflects the state of the 9th received bit. In Mode 1, if multiprocessor communication is enabled (sm2 = 0), this bit is the stop bit that was received. In Mode 0 this bit is not used.
TI	Transmit interrupt flag : (completion of a serial transmission) It is set by hardware at the end of bit 8 in mode 0 or at the beginning of a stop bit in other modes. It must be cleared by software.
RI	Receive interrupt flag : (It must be cleared by software.) It is set by hardware after completion of a serial reception at Serial Port 0. It is set by hardware at the end of bit 8 in mode 0 or in the middle of a stop bit in other modes.

Serial Port working in modes 1 or mode 3:

When BRS = 0 (AUX.2)

TIPS[1:0] = 00b

$$\text{Baud rate} = \frac{2^{SMOD}}{32\times(256\text{-TH1})} \times \frac{F\_PER}{12}$$

TIPS[1:0] = 01b

$$\text{baud rate} = \frac{2^{SMOD}}{32\times(256\text{-TH1})} \times F\_PER$$

TIPS[1:0] = 10b

$$\text{baud rate} = \frac{2^{SMOD}}{32\times(256\text{-TH1})} \times \frac{F\_PER}{96}$$

When BRS = 1 (AUX.2)

SRELPS[1:0] = 00b

$$\text{baud rate} = \frac{2^{SMOD}}{2^{10}-SREL[H,L]} \times \frac{F\_PER}{64}$$

SRELPS [1:0] = 01b

$$\text{baud rate} = \frac{2^{SMOD}}{2^{10}-SREL[H,L]} \times \frac{F\_PER}{32}$$

SRELPS [1:0] = 10b

$$\text{baud rate} = \frac{2^{SMOD}}{2^{10}-SREL[H,L]} \times \frac{F\_PER}{16}$$

SRELPS [1:0] = 11b

$$\text{baud rate} = \frac{2^{SMOD}}{2^{10}-SREL[H,L]} \times \frac{F\_PER}{8}$$

15.6.2 SBUF (Serial Port Data Buffer)

Writing data to this register sets data in serial output buffer and starts the transmission through Serial Port. Reading from the SBUF, reads data from the serial receive buffer.

SBUF	Address = 99H					Reset Value = 0x00H		
Serial Port Data Buffer								
Bit	SBUF[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

15.6.3 SREL (Serial Port Reload Register)

Serial Port Reload Register is used for Serial Port baud rate generation. Only 10 bits are used, where 8 bits from the SRELL as lower bits and 2 bits from the SRELH (SRELH.1, SRELH.0) as higher bits.

SRELL	Address = 9AH					Reset Value = 0x00H		
Serial Port Reload Register Low								
Bit	SREL[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SRELH	Address = 9BH					Reset Value = 0x00H		
Serial Port Reload Register High								
Bit	-----	-----	-----	-----	-----	-----	SREL.9	SREL.8
	7	6	5	4	3	2	1	0
Type	X	X	X	X	X	X	R/W	R/W

15.7 Power Management

The Power Control Register (PCON) is used to control the CGF022A STOP and IDLE power management modes.

PCON	Address = 87H					Reset Value = 0x00H		
Power Control Register								
Bit	-----	-----	-----	-----	-----	-----	STOP	IDLE
	7	6	5	4	3	2	1	0
Type	X	X	X	X	X	X	R/W	R/W

STOP	Stop mode bit.
[1]	Setting this bit activates STOP operation. (read as 0)
IDLE	Idle mode bit.
[0]	Setting this bit activates IDLE mode operation. (read as 0)

15.7.1 STOP MODE

Setting the STOP Mode Select bit (PCON.1) causes the controller core to enter STOP mode as soon as the instruction that sets the bit completes execution. In STOP mode the CPU, GPIO, UART, and Timers are stopped, but the ADC, MDE, and WDT is still work.

STOP mode can terminate by an internal or external reset. On reset, the device performs the normal reset sequence and begins program execution at address 0x0000.

The assertion of an enabled interrupt will cause the STOP Mode Selection bit (PCON.1) to be cleared and the CPU to resume operation.

15.7.2 IDLE MODE

Setting the IDLE Mode Select bit (PCON.0) causes the hardware to halt the CPU and enter IDLE mode as soon as the instruction that sets the bit completes execution.

In IDLE mode only the CPU is stop. All internal registers and memory maintain their original data.

IDLE mode can terminate by an internal or external reset. On reset, the device performs the normal reset sequence and begins program execution at address 0x0000.

The assertion of an enabled interrupt will cause the IDLE Mode Selection bit (PCON.0) to be cleared and the CPU to resume operation.

15.8 Reset

The reset logic is used to place the device into a known state.

CGF022A provides Power-on Reset flag, External Reset RSTN flag and Watchdog timer Reset flag to monitor reset status. The source of the reset can be monitor.

15.8.1 RSTS (Reset Source Register)

RSTS Address = FEH Reset Value = 0x0AH

Reset Source Register

Bit	-----	-----	-----	WDTRF	PINRF[1:0]		PORF[1:0]	
	7	6	5	4	3	2	1	0
Type	X	X	X	R/W	R/W	R/W	R/W	R/W

WDTRF	Watchdog timer reset flag.
[4]	This flag is set to 1 if the Watchdog Timer Reset caused the reset.
PINRF	RSTN pin reset flag.
[3:2]	This flag is set to 10b if the RSTN pin caused the reset. Clear by firmware.
PORF	POR reset flag.
[1:0]	This flag is set to 10b if the POR caused the reset. Clear by firmware.

15.9 Interrupt Controller

The ISR - Interrupt Service Routine unit, is a subcomponent responsible for interrupt handling. It receives up to 14 interrupt requests. Each interrupt source has its own request flag that is located in devices which is a source of interrupt. No interrupt request flags are located directly in ISR. All interrupts are requested by high level on correspondent inputs to ISR. Each of the interrupt sources can be individually enabled or disabled by corresponding enable flag in IEN0, IEN1SFR registers. Additionally all interrupts can be globally enabled or disabled by the —EA flag in the IEN0 SFR. All interrupt sources are divided into 6 interrupts groups. Each of the interrupt groups can have one of four interrupt priority levels assigned. The interrupt priority level is defined by flags located in the IP0 and IP1 SFR registers.

Interrupt Number (use Keil C Tool)	Interrupt Vector Address	Interrupt Request Flags
0	0003H	IE0 – External interrupt 0
1	000BH	TF0 – Timer0 interrupt
2	0013H	IE1 – External interrupt 1
3	001BH	TF1 – Timer1 interrupt
4	0023H	SPIF(TI, RI)– Serial port interrupt
5	002BH	TF2 – Timer2 interrupt
6	0033H	-----
7	003BH	-----
8	0043H	OCPSIF – OCP Short interrupt
9	004BH	ADCIF –ADC interrupt
10	0053H	MPWMMINIF–MPWM MIN interrupt
11	005BH	MPWMMAXIF–MPWM MAX interrupt
12	0063H	IICIF – IIC interrupt
13	006BH	LVDIF – Low voltage detect interrupt
14	0073H	WDTIF – Watchdog timer interrupt
15	007BH	CAPTUREIF –CAPTURE interrupt

Table15.9.1 Interrupt vectors

Group priority	Interrupt Group	Highest priority in group		Lowest priority in group
Highest	Group0	LVDIF	IE0	-----
	Group1	WDTIF	TF0	-----
	Group2	OCPSIF	ADCIF	IE1
	Group3	MPWMMINIF	MPWMMAXIF	TF1
	Group4	IICIF	SPIF(TI, RI)	-----
Lowest	Group5	CAPIF	TF2	-----

Table12.9.2 Interrupt Priority Groups

SFR	Description	address	Reset value
IEN0	Interrupt Enable Register 0	A8H	00H
IEN1	Interrupt Enable Register 1	B8H	00H
IRCON1	Interrupt Request Register 1	C0H	00H
IP0	Interrupt Priority Register 0	A9H	00H
IP1	Interrupt Priority Register 1	B9H	00H

15.9.1 IEN0 (Interrupt Enable Register 0)

IEN0 Address = A8H Reset Value = 0x00H

Interrupt Enable Register 0

Bit	EA	-----	ET2	ESP	ET1	EX1	ET0	EX0
	7	6	5	4	3	2	1	0
Type	R/W	X	R/W	R/W	R/W	R/W	R/W	R/W

EA	Interrupts enable :
[7]	0 : Disable All interrupts.
	1 : Enable interrupt.
ET2	Timer2 interrupt enable:
[5]	0 : Disable Timer2 overflow interrupt.
	1 : When EA = 1, enable Timer2 overflow interrupt.
ESP	Serial port interrupt enable:
[4]	0 : Disable Serial port interrupt.
	1 : When EA = 1, enable Serial port interrupt.
ET1	Timer1 interrupt enable:
[3]	0 : Disable Timer1 overflow interrupt.
	1 : When EA = 1, enable Timer1 overflow interrupt.
EX1	External interrupt 1 enable:
[2]	0 : Disable External interrupt 1.
	1 : When EA = 1, enable External interrupt 1.
ET0	Timer0 interrupt enable:
[1]	0 : Disable Timer0 overflow interrupt.
	1 : When EA = 1, enable Timer0 overflow interrupt.
EX0	External interrupt 0 enable:
[0]	0 : Disable External interrupt 0.
	1 : When EA = 1, enable External interrupt 0.

15.9.2 IEN1 (Interrupt Enable Register 1)

IEN1 Address = B8H Reset Value = 0x00H

Interrupt Enable Register 1

Bit	CAPIE	WDTIE	LVDIE	IICIE	MPWMMAXIE	MPWMMINIE	ADCIE	OCPSIE
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CAPIE [7]	Capture interrupt enable: 0 : Disable CAPTURE interrupt. 1 : When EA = 1, enable CAPTURE interrupt.
WDTIE [6]	Watchdog timer interrupts enable : 0 : Disable WDT interrupt. 1 : When EA = 1 and WDTOS = 1, enable WDT overflow interrupt.
LVDIE [5]	LVD (Low voltage detect) interrupt enable: 0 : Disable LVD interrupt. 1 : When EA = 1, enable LVD interrupt.
IICIE [3]	IIC interrupt enable: 0 : Disable IIC interrupt. 1 : When EA = 1, enable IIC interrupt.
MPWMMAXIE [3]	MPWM maximum interrupt enable: 0 : Disable MPWM maximum interrupt. 1 : When EA = 1, enable MPWM maximum interrupt.
MPWMMINIE [2]	MPWM minimum interrupt enable: 0 : Disable MPWM minimum interrupt . 1 : When EA = 1, enable MPWM minimum interrupt.
ADCIE [1]	ADC interrupt enable: 0 : Disable ADC interrupt. 1 : When EA = 1, enable ADC interrupt.
OCPSIE [0]	OCP (Over current protect) Short interrupt enable: 0 : Disable OCP Short interrupt . 1 : When EA = 1, enable OCP Short interrupt.

15.9.3 IRCON1 (Interrupt Request Register 1)

IRCON1 Address = C0H Reset Value = 0x00H

Interrupt Request Register 1

Bit	CAPTUREIF	WDTIF	LVDIF	IICIF	MPWMMAXIF	MPWMMINIF	ADCIF	OCPSIF
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CAPTIF[7]	Capture interrupt (Counter Over flow and Rising trigger) flag.
WDTIF[6]	Watchdog timer interrupts flag.
LVDIF[5]	LVD (Low voltage detect) interrupt flag.
IICIF[4]	IIC interrupt flag.
MPWMMAXIF[3]	MPWM maximum interrupt flag.
MPWMMINIF[2]	MPWM minimum interrupt flag.
ADCIF[1]	ADC interrupt flag.
OCPSIF[0]	OCP Short interrupt flag.(AOCP and DOCPN)

15.9.4 IP (Interrupt Priority Register)

The 13 interrupt sources are grouped into 6 priority groups. For each of the groups, one of four priority levels can be selected. It is achieved by setting appropriate values in IP0 and IP1 registers. The contents of the Interrupt Priority Registers define the priority levels for each interrupt source according to the tables below.

IP0 Address = A9H Reset Value = 0x10H

Interrupt Priority Register 0

Bit	-----	-----	G5IP0	G4IP0	G3IP0	G2IP0	G1IP0	G0IP0
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

G5IP0[5]	Group5 interrupt priority bit 0
G4IP0[4]	Group4 interrupt priority bit 0
G3IP0[3]	Group3 interrupt priority bit 0
G2IP0[2]	Group2 interrupt priority bit 0
G1IP0[1]	Group1 interrupt priority bit 0
G0IP0[0]	Group0 interrupt priority bit 0

IP1 Address = B9H Reset Value = 0x00H

Interrupt Priority Register 1

Bit	-----	-----	G5IP1	G4IP1	G3IP1	G2IP1	G1IP1	G0IP1
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

G5IP1[5]	Group5 interrupt priority bit 1
G4IP1[4]	Group4 interrupt priority bit 1
G3IP1[3]	Group3 interrupt priority bit 1
G2IP1[2]	Group2 interrupt priority bit 1
G1IP1[1]	Group1 interrupt priority bit 1
G0IP1[0]	Group0 interrupt priority bit 1

		Group x	
Level	Priority	IP1[GxIP1]	IP0 [GxIP0]
Level 0	Lowest	0	0
Level 1		0	1
Level 2		1	0
Level 3	Highest	1	1
x : 0~5			

16. 10-bit Analog-to-Digital Converter (ADC)

CGF022A provides 8 channels 10-bit ADC. The result of the conversion is provided at ADCD [9:0]. The ADC channel 0 and channel 1 are automatic conversion when IC is regarded as sensor-less FOC motor controller, then the ADC auto conversion The time point is related to the temporary register FOCCONT[2]. Under normal use, the timing diagram of ADC conversion is shown in Figure 16.1. and ADC SFR List Please refer to Table16.1.

SFR	Description	address	Reset value
ADCCONT	ADC Control Register	C2H	83H
ADCDLY	ADC Sample Delay (For CH0&CH1)	C5H	33H
ADCSTR	ADC Start Convert and Setting Register	C1H	00H
ADCD1	ADC Data Register 1	C3H	00H
ADCD2	ADC Data Register 2	C4H	00H
ADCOS_L	ADC1 Offset Value Register Low Byte	C9H	00H
ADCOS_H	ADC1 Offset Value Register High Byte	CAH	02H

Table16.1 : ADC SFR List

16.1 ADCCONT (ADC Control Register)

When the ADC is used under normal circumstances, the conversion timing diagram is shown in Figure 16.1.1, in which the ADC input channel, ADC clock, ADC sample and hold time, and the 10-bit output data after the ADC conversion is completed (see Table 16.4.1) The order settings of the high and low bits are set in ADCCONT SFR (see Figure 16.1.1 and Table 16.1.1);
In addition, the ADC sample and hold clock is set in bit[7:6] (SHCKS) of the ADCSTR register (see Table 16.2.1). When all ADC parameters are set, fill in "1" to bit[0] of the ADCSTR register, and the ADC will start sampling and conversion.

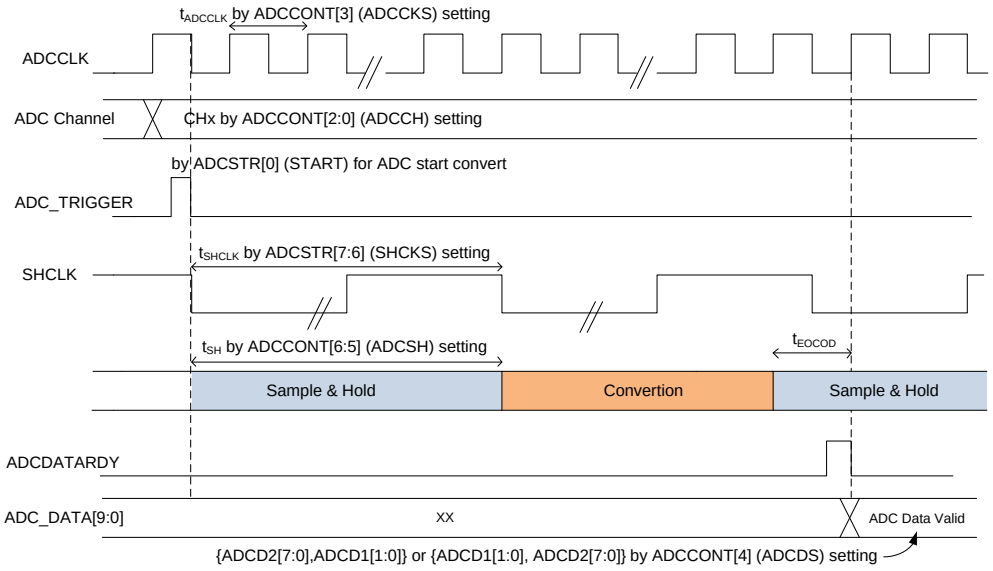


Figure 16.1.1 ADC conversion timing

ADCCONT — Address = C2H — Reset Value = 0x80H
ADC Control Register

Bit	ADCPD	ADCSH[1:0]		ADCDS	ADCCKS	ADDCH[2:0]		
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ADCPD [7] — ADC power down control register :
0 : Normal
1 : Power down

ADCSH [6:5] — ADC sample and hold time : (base on SHCLK)
00 : 1 clock
01 : 2 clock
10 : 3 clock
11 : 4 clock

ADCDS [4] — ADC data select :

	MSB	10 bit result		LSB
0 :	ADCD2[7:0]		ADCD1.1	ADCD1.0
1 :	ADCD1.1	ADCD1.0	ADCD2[7:0]	

ADCCKS [3] — ADC conversion clock select : (ADCCLK)
0 : 24MHz
1 : 12MHz

ADCCH [2:0] — ADC conversion channel select :

000 :CH0	100 :CH4
001 :CH1	101 :CH5
010 :CH2	110 :CH6
011 : CH3	111 : CH7

Table 16.1.1 ADC Control Register

16.2 ADCSTR (ADC Start Convert and Setting Register)

As mentioned in the previous section, under normal use of ADC, when all ADC parameters are set and "1" is filled in bit[0] of the ADCSTR register, the ADC will start sampling and conversion. Set in bit[5:4] (OPAGAIN, see Table16.2.1) of ADCSTR that only CH0 and CH1 can be amplified when used as a motor driver, and CH0 and CH1 are automatically converted into phase currents in sequence by the MDE hardware; ADC's CH2~CH7 can only be used after CH0 and CH1 are converted, and the OPAGAIN at this time is fixed at 1 times. The ADC CH0 and CH1 current sampling block diagram when used as a motor driver is shown in Figure 16.2.1.

ADCSTR — Address = C1H — Reset Value = 0x00H

ADC Start Convert and Setting Register

Bit	SHCKS[1:0]		OPAGAIN		BUSY	-----		START
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R	X	X	W

SHCKS [7:6]	ADC sample and hold clock select : (SHCLK)	
	00 : 6MHz	10 : 2.4MHz
	01 : 3MHz	11 : 2MHz
OPAGAIN [5:4]	OPAmp Gain	
	00 : 1	10 : 5
	01 : 2.5	11 : 10
BUSY [3]	ADC conversion busy flag :	
	0 : ADC conversion finish	
	1 : ADC conversion busy	
START [0]	ADC start conversion register : (write 1 only)	
	1 : ADC start conversion	

Table 16.2.1 ADC Start Convert and Setting Register

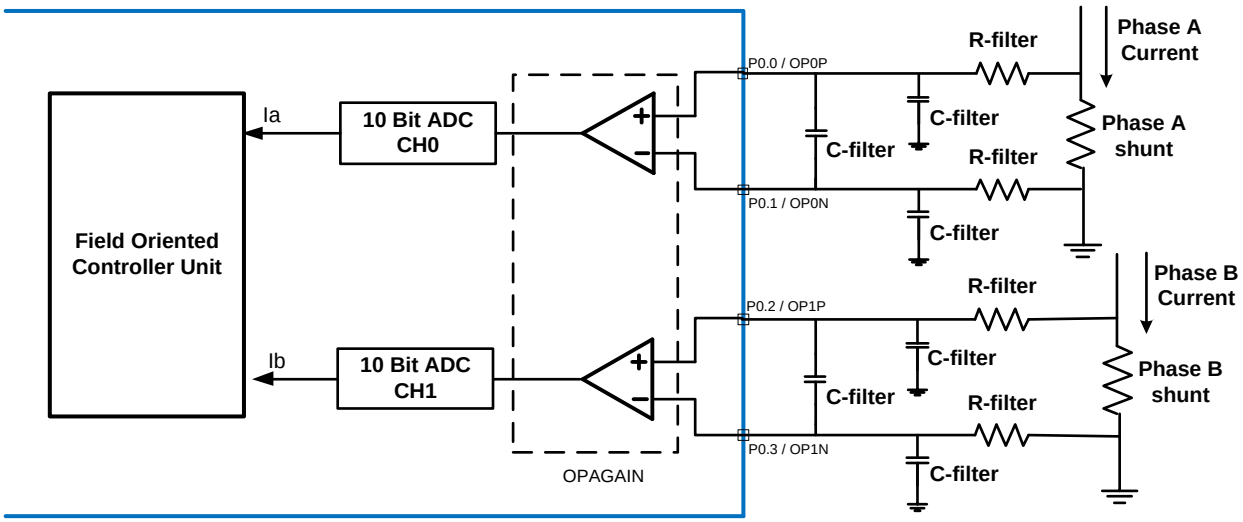


Figure 16.2.1 The block diagram of ADC CH0 and CH1 for Sensor-less FOC motor driving

16.3 ADCDLY (ADC Sample Delay Register)

The setting of the ADCDLY register is used for CH0 and CH1 of the motor drive. When switching the Power device, a spike current will occur. After setting the delay time in the ADCDLY SFR, the current sampling will be started when the current is stable. Therefore The setting of ADCDLY is the same as OPAGAIN and is only suitable for motor drive current sampling.

In addition, since the motor drive current sampling is automatically performed by MDE, the sampling time point can be set by bit[2] (ADCTRIG, see Table 16.3.2) of the FOCCONT register to be in the PWM counter The maximum value or 0 starts sampling after the setting time of ADCDLY.

After each current sampling of CH0 and CH1, MDE will issue an ADC interrupt. Using the ADC interrupt vector, the user can perform ADC conversion of other channels. The timing diagram of the entire ADC sampling is shown in Figure 16.3.1.

ADCDLY — Address = C5H — Reset Value =0x33H

ADC Sample Delay

Bit	ADCDLY[7:0]							
	7	6	5	4	3	2	1	0
Type	W	W	W	W	W	W	W	W

$$\text{ADC Sample delay time} = \frac{\text{ADCDLY}}{\text{PWM Frequency}}$$

Table 16.3.1 ADC Sample Delay Register

FOCCONT — Address = D6H — Reset Value = 0x00H

Field Oriented Control Register

Bit	PI CLEAR	ESTCR	INV ADCD	ADCTRIG	PLLEN	SPEEDEN
				2		
Type				R/W		

ADCTRIG [2]	ADC Trigger (Phase A and Phase B): 0: PWM counter max 1: PWM counter min

Table16.3.2 : FOCCONT. ADCTRIG Description

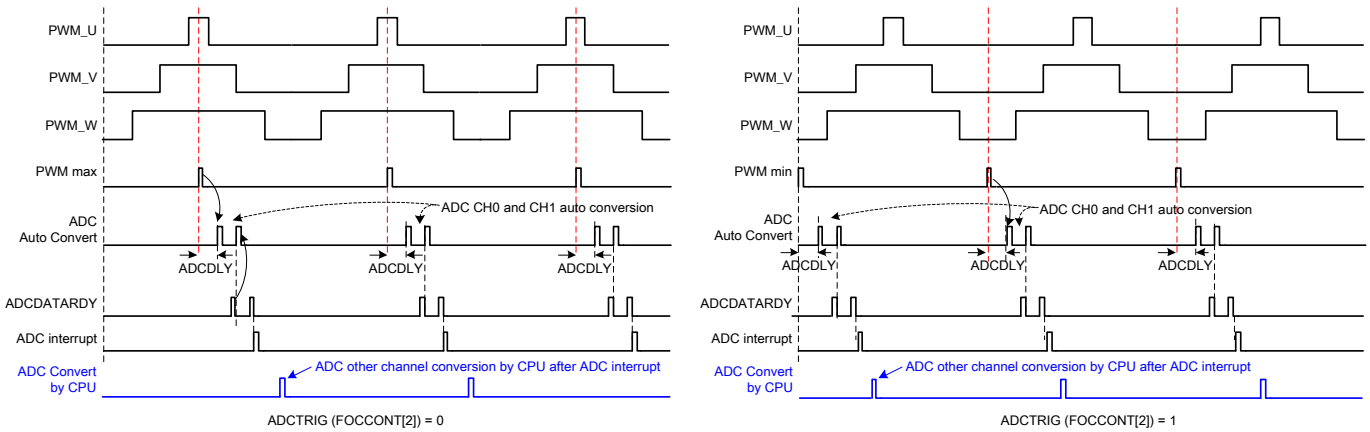


Figure 16.3.1 : The Timing Diagram of ADC Convert by MDE and CUP

16.4 ADC Data Register

The 10-bit data converted by the CH2~CH7 ADC will be placed in the ADCD1 and ADCD2 SFRs of Table 16.4.1 respectively. The arrangement is based on the setting of ADCCONT[4] (ADCDS, see Table 16.1.1). When ADCDS = 0, then ADC_DATA[9:0] = {ADCD2[7:0], ADCD1[1:0]}; when ADCDS = 1, then ADC_DATA[9:0] = { ADCD1[1:0], ADCD2[7:0]}.

ADCD1 Address = C3H Reset Value = 0x00H
ADC Data Register1

Bit	ADCD1 [1:0]							
	---	---	---	---	---	---	1	0
Type	R	R	R	R	R	R	R	R

ADCD2 Address = C4H Reset Value = 0x00H
ADC Data Register 2

Bit	ADCD2 [7:0]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

Table 16.4.1 ADC Data Register ADCD1 and ADCD2

16.5 ADC1 and ADC2 Offset Value Register

When sampling the motor driving current, MDE will automatically sample the two-phase current (ia and ib). If there is a fixed offset error between the sampled i_a and i_b and the actual measured current, CGF022A provides ia and ib each 10-bit offset correction (Table 16.5.1). Therefore, the actual ia current in the FOC operation is the CH0 sampling value + ADC1OS (Table 16.5.2); the actual i_b current is the CH1 sampling value + ADC2OS (Table 16.5.3).

SFR	Description		Address
ADCOS_L	ADC Offset Value Register Low Byte		C9H
ADCOS_H	ADC Offset Value Register High Byte		CAH
	Parameters	**Description**	**Reset Value**
SFR_PAGE = 0	ADCOS1[H,L]	ADC1 Offset Value	0x0200H
SFR_PAGE = 1	ADCOS2[H,L]	ADC2 Offset Value	0x0200H

Table 16.5.1 ADC Offset Register

ADC1OS_L Address = C9H (SFR_PAGE = 0) Reset Value =0x00H

ADC1Offset Value Register Low Byte

Bit	ADC1OS[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ADC1OS_H Address = CAH (SFR_PAGE = 0) Reset Value = 0x02H

ADC1Offset Value Register High Byte

Bit	-----						ADC1OS [9:8]	
	7	6	5	4	3	2	1	0
Type	-----	-----	-----	-----	-----	-----	R/W	R/W

Table 16.5.2 Phase A Current (i_a) Offset Register

ADC2OS_L Address = C9H (SFR_PAGE = 1) Reset Value = 0x00H

ADC2Offset Value Register Low Byte

Bit	ADC2OS[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ADC2OS_H Address = CAH (SFR_PAGE = 1) Reset Value = 0x02H

ADC2 Offset Value Register High Byte

Bit	-----						ADC2OS [9:8]	
	7	6	5	4	3	2	1	0
Type	-----	-----	-----	-----	-----	-----	R/W	R/W

Table 16.5.3 Phase B Current (i_b) Offset Register

17. EEPROM

CGF022A provides internal flash memory control signals that can perform byte read/write and page erase of the last 256 bytes of flash memory to simulate EEPROM as parameter storage. The read/write address, data, and command are control by EE_ADDR, EE_DATA, and EE_CMD register respectively which in the page of EEPROM SFR (Table 17.1). When performing an EEPROM erase, read or write command, you can first read the flag of the relevant command to confirm that the previous command has been completed before proceeding to the next command. The control flow of each command is shown in Figure 17.1. Please refer to Table17.1.1, Table17.2.1, and Table17.3.1 for each register.

SFR	Description		Address
EEPROM	EEPROM Function Register		A3H
	Parameters	**Description**	**Reset Value**
SFR_PAGE = 0	EE_ADDR	EEPROM read/write address	0x00H
SFR_PAGE = 1	EE_DATA	EEPROM read/write data	0x00H
SFR_PAGE = 2	EE_CMD	EEPROM command	0x00H

Table 17.1 EEPROM Register

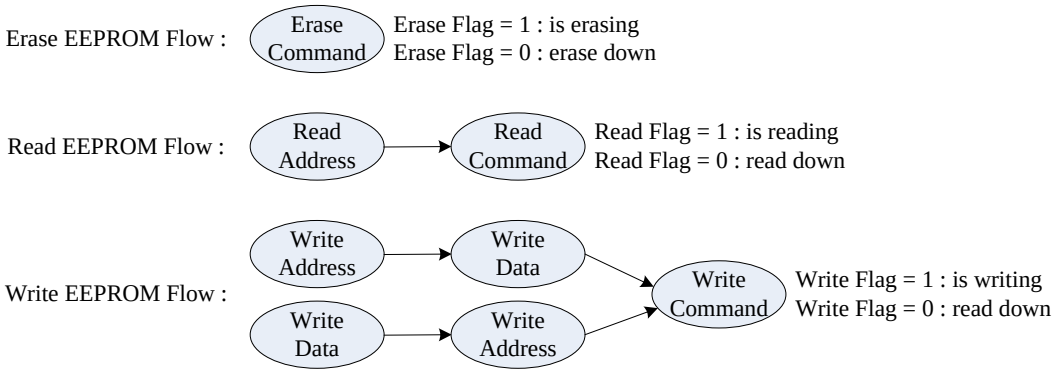


Figure 17.1 : EEPROM Erase, Read, and Write Control Flow

17.1 EE_ADDR (EEPROM Read/Write Address Register)

EE_ADDR	Address = A3H (SFR_PAGE = 0)						Reset Value = 0x00H	
EEPROM read/write address								
Bit	EE_ADDR [7:0]							
Type	7	6	5	4	3	2	1	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.1.1 EEPROM Read/Write Address Register

17.2 EE_DATA (EEPROM Read/Write Data Register)

EE_DATA	Address = A3H (SFR_PAGE = 1)						Reset Value = 0x00H	
EEPROM read/write data								
Bit	EE_DATA [7:0]							
Type	7	6	5	4	3	2	1	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.2.1 EEPROM Read/Write Data Register

17.3 EE_CMD (EEPROM Command Register)

EE_CMD	Address = A3H (SFR_PAGE = 2)						Reset Value = 0x00H	
EEPROM command								
Bit	EE_CMD [7:0]							
	ERS_FLAG	---	WR_FLAG	RD_FLAG	ERASE	---	WR_CMD	RD_CMD
Bit	7	6	5	4	3	2	1	0
Type	R/W	---	R/W	R/W	R/W	---	R/W	R/W

ERS_FLAG [7]	EEPROM ERASE Flag: 1 : EEPROM Erase progressing
WR_FLAG [5]	EEPROM WRITE Flag: 1 : EEPROM Write progressing
RD_FLAG [4]	EEPROM READ Flag: 1 : EEPROM Read progressing
ERASE [3]	EERPOM ERASE: 1 : ERASE command
WR_CMD [1]	EERPOM WRITE command: 1 : WRITE command
RD_CMD [0]	EERPOM READ command: 1 : READ command

Table 17.3.1 EEPROM Command Register

18. IIC Emulation

The IIC module uses the SCL (clock) and the SDA (data) line to communicate with external IIC interface. Its speed can be selected to 400Kbps (maximum) by software setting the IICBR [2:0] control bit. The IIC module provided 2 interrupts (RXIF, TXIF). It will generate START, repeated START and STOP signals automatically in master mode and can detects START, repeated START and STOP signals in slave mode. The maximum communication length and the number of devices that can be connected are limited by a maximum bus capacitance of 400pF. The IIC interrupt vector is 63h. IIC SFR List Please refer to Table18.1.

SFR	Description	Address	Reset value
IICS	IIC Status Register	E8H	00H
IICCTRL	IIC Control Register	E9H	04H
IICA1	IIC Address 1 Register	EAH	A0H
IICA2	IIC Address 2 Register	EBH	60H
IICRWD	IIC Read Write Register	BAH	00H
IICEBT	IIC Enable Bus Transaction Register	9CH	00H

Table 18.1 : IIC SFR List

18.1 IICCTL (IIC Control Register)

IICCTL Address = E9H Reset Value = 0x04H

IIC Control Register

Bit	IICEN	MSS	MAS	AB_EN	BF_EN	IICBR[2:0]		
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IICEN	Enable IIC module
[7]	0 : Disable
	1 : Enable
MSS	Master or slave mode select
[6]	0 : slave mode
	1 : master mode *The software must set this bit before setting others register.
MAS	Master address select (master mode only)
[5]	0 : Master address is to use IICA1
	1 : Master address is to use IICA2
AB_EN	Arbitration lost enable bit. (Master mode only)
[4]	If set AB_EN bit, the hardware will check arbitration lost. Once arbitration lost occurred, hardware will return to IDLE state. If this bit is cleared, hardware will not care arbitration lost condition. Set this bit when multi-master and slave connection. Clear this bit when single master to single slave.
BF_EN	Bus busy enable bit. (Master mode only)
[3]	If set BF_EN bit, hardware will not generate a start condition to bus until BF=0. Clear this bit will always generate a start condition to bus when MStart is set. Set this bit when multi-master and slave connection. Clear this bit when single master to single slave.
IICBR	Baud rate selection (master mode only) , where Fosc is 12MHz
[2:0]	000 : Fosc/32 100 : Fosc/512
	001 : Fosc/64 101 : Fosc/1024
	010 : Fosc/128 110 : Fosc/2048
	011 : Fosc/256 111 : Fosc/4096

18.2 IICS (IIC Status Register)

IICS Address = E8H Reset Value = 0x00H

IIC Status Register

Bit	---	MPIF	LAIF	RXIF	TXIF	RXAK	TXAK	RW or BB
	7	6	5	4	3	2	1	0
Type	---	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MPIF	The Stop condition Interrupt Flag
[6]	The stop condition occurred and this bit will be set. Software need to clear this bit.
LAIF	Arbitration lost bit. (Master mode only)
[5]	The Arbitration Interrupt Flag, the bus arbitration lost occurred and this bit will be set. Software need to clear this bit.
RXIF	The data Receive Interrupt Flag
[4]	RXIF is set after the IICRWD (IIC Read Write Data Buffer) is loaded with a newly receive data.
TXIF	The data Transmit Interrupt Flag
[3]	TXIF is set when the data of the IICRWD (IIC Read Write Data Buffer) is downloaded to the shift register.
RXAK	The Acknowledge Status indicate bit
[2]	When clear, it means an acknowledge signal has been received after the complete 8 bits data transmit on the bus.
TXAK	The Acknowledge status transmit bit (Figure 18.2.1)
[1]	When received complete 8 bits data, this bit will set (NoAck) or clear (Ack) and transmit to master to indicate the receive status.
RW or BB	Bus busy or slave mode read/write on the IIC bus
[0]	**Master Mode:** BB : Bus busy bit If detect scl=0 or sda=0 or bus start, this bit will be set. If detect stop, this bit will be cleared. This bit can be cleared by software to return ready state. **Slave Mode:** RW : The slave mode read (received) or wrote (transmit) on the IIC bus. When this bit is clear, the slave module received data on the IIC bus (SDA).(Slave mode only).

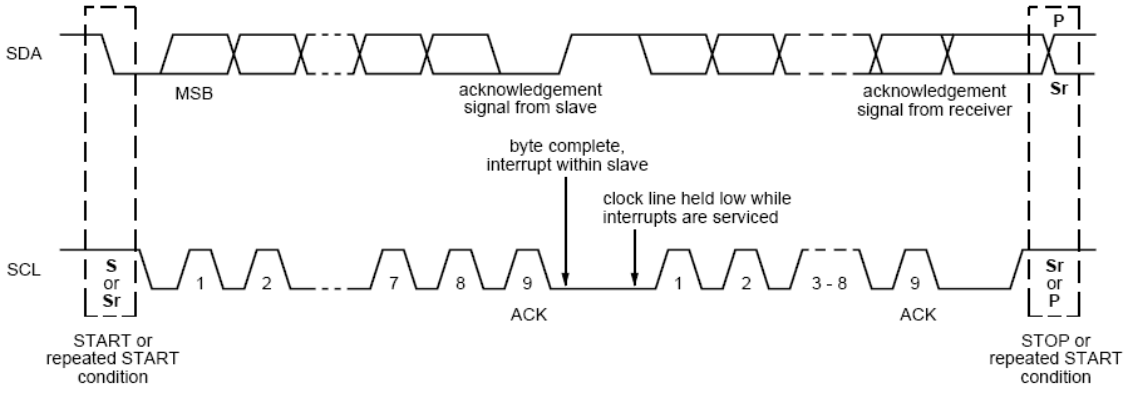


Figure 18.2.1 : Acknowledgement bit in the 9th bit of a byte transmission

18.3 IICA1 (IIC Address1 Register)

IICA1 Address = EAH Reset Value = 0xA0H

IIC Address1 Register

Bit	IICA1							Match1 or RW1
	7	6	5	4	3	2	1	0
Type	R/W							R or R/W

Slave mode	
IICA1	IIC Address1 registers
[7:1]	This is the first 7-bit address for this slave module. It will be checked when an address (from master) is received.
Match1	IICA1 match bit (Read only)
[0]	When IICA1 matches with the received address from the master side, this bit will set to 1 by hardware. When IIC bus gets first data, this bit will clear.
Master mode	
IICA1	IIC Address1 registers
[7:1]	This 7-bit address indicates the slave with which it wants to communicate.
RW1	Master read/write mode indicate
[0]	This bit will be sent out as RW of the slave side if the module has set the MStart or RStart bit. It appears at the 8th bit after the IIC address as below figure (Figure18.3.1). It is used to tell the salve the direction of the following communication. If it is 1, the module is in master receive mode. If 0, the module is in master transmit mode. The Fig. as shown as follow. RW1=1, master receive mode RW1=0, master transmit mode

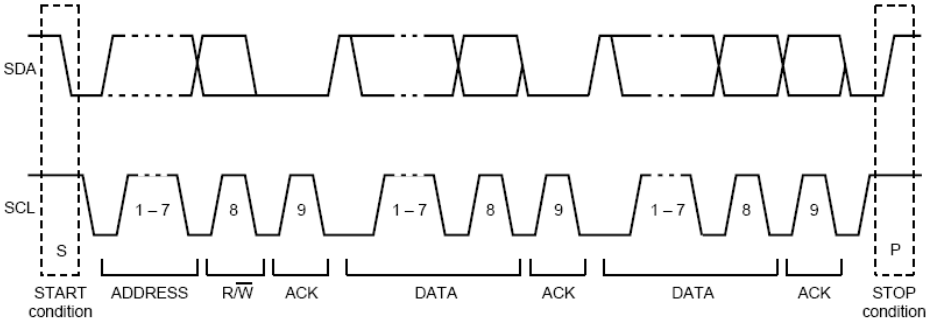


Figure 18.3.1 **:** RW bit in the 8th bit after IIC address

18.4 IICA2 (IIC Address2 Register)

IICA2 Address = EBH Reset Value = 0x60H

IIC Address 2 Register

Bit	7	6	5	4	3	2	1	0
	IICA2							Match2 or RW2
Type	R/W							R or R/W

Slave mode

IICA2	IIC Address2 registers
[7:1]	This is the second 7-bit address for this slave module. It will be checked when an address (from master) is received.
Match2	IICA2 match bit (Read only)
[0]	When IICA2 matches with the received address from the master side, this bit will set to 1 by hardware. When IIC bus gets first data, this bit will clear.

Master mode

IICA2	IIC Address2 registers
[7:1]	This 7-bit address indicates the slave with which it wants to communicate.
RW2	Master read/write mode indicate
[0]	This bit will be sent out as RW of the slave side if the module has set the MStart or RStart bit. It is used to tell the salve the direction of the following communication. If it is 1, the module is in master receive mode. If 0, the module is in master transmit mode. RW2=1, master receive mode RW2=0, master transmit mode

18.5 IICRWD (IIC Read Write Data Register)

IICRWD Address = BAH Reset Value = 0x00H

IIC Read Write Register

Bit	7	6	5	4	3	2	1	0
	IICRWD							
Type	R/W							

IICRWD	IIC read write data buffer
[7:0]	In receiving (read) mode, the received byte is stored here. In transmitting mode, the byte to be shifted out through SDA stays here.

18.6 IICEBT (IIC Enable Transaction Register)

IICEBT Address = 9CH Reset Value = 0x00H

IIC Enable Transaction Register

Bit	7	6	5	4	3	2	1	0
	FU_EN		---	---	---	---	---	---
Type	R/W		---	---	---	---	---	---

Master mode

[7:6] Function Enable

00 : reserved

01 : IIC bus module will enable read/write data transfer on SDA and SCL

10 : IIC bus module generate a start condition on the SDA/SCL, then send out address which is stored in the IICA1/IICA2(selected by MAS control bit)

11 : IIC bus module generates a stop condition on the SDA/SCL.

Notice : 1. FU_EN[7:6] will be auto-clear by hardware, so setting FU_EN[7:6] repeatedly is necessary.

Slave mode

[7:6] Function Enable

01 : FU_EN[7:6] should be set as 01 only. The other value is inhibited.

Notice :
1. FU_EN[7:6] should be set as 01 before read/write data transfer for bus release; otherwise, SCL will be locked(pull low).
2. FU_EN[7:6] should be set as 01 after read/write data transfer for receiving a stop condition from bus master.
3. In transmit data mode(slave mode), the output data should be filled into IICRWD before setting FU_EN[7:6] as 01.
4. FU_EN[7:6] will be auto-clear by hardware, so setting FU_EN[7:6] repeatedly is necessary.

19. Capture

CGF022A provides external and internal signal capture function that can count the high level length (CAPH_H, CAPH_L) and the period length (CAPT_H, CAPT_L) of the external and internal signal (Table 19.1). This interrupt is a combination of external signal capture interrupt and internal signal capture interrupt, which will be issued after the rising edge of the external or internal capture signal. User must to monitor the external capture OK (ECAP_OK) and internal capture OK (ICAP_OK) in the capture interrupt routine to distinguish which counter can access, and it is cleared by user when set the clear capture OK flag (ECAP_OK_CLR or ICAP_OK_CLR) bit "1". The external and internal capture counter are share same SFR but separate in different page (Table 19.2).

SFR	Description	Address	Reset value
CAPCONT	Capture Control Register	DDH	03H
Capture Total Count:			
CAPT_L	Capture Total Count Low	D9H	00H
CAPT_H	Capture Total Count High	DAH	00H
Capture Count:			
CAPH_L	Capture High-level Count Low	DBH	00H
CAPH_H	Capture High-level Count High	DCH	00H

Table 19.1 : Capture SFR List

SFR	**Description**		**Address**
CAPCONT	Capture Control Register		DDH
	Parameters	**Description**	**Reset Value**
SFR_PAGE = 0	EXT_CAPCONT	External Control Register	0x03H
SFR_PAGE = 1	INT_CAPCONT	Internal Control Register	0x03H
SFR	**Description**		**Address**
CAPT_L	Capture Total Count Register Low Byte		D9H
CAPT_H	Capture Total Count Register High Byte		DAH
	Parameters	**Description**	**Reset Value**
SFR_PAGE = 0	EXT_CAPT[H,L]	External Capture Total Count Register	0x0000H
SFR_PAGE = 1	INT_CAPT[H,L]	Internal Capture Total Count Register	0x0000H
SFR	**Description**		**Address**
CAPH_L	Capture High-level Count Register Low Byte		DBH
CAPH_H	Capture High-level Count Register High Byte		DCH
	Parameters	**Description**	**Reset Value**
SFR_PAGE = 0	EXT_CAPH[H,L]	External Capture High-level Count Register	0x0000H
SFR_PAGE = 1	INT_CAPH[H,L]	Internal Capture High-level Count Register	0x0000H

Table 19.2 : Capture Counter SFR

19.1 External Capture

External capture input can be selected from CAP1 or CAP2, and the setting method is as shown in bit [4] (CAPPINSEL) in the table below. In addition, the clock frequency for counting the capture signal can be selected according to the cycle length of the input signal (set by ECAPCKS in Table 19.1.1), which can avoid overflow in the internal counter and cause distortion of the high level length and cycle length of the capture signal. When the capture of the external signal is completed, an interrupt will be issued and the ECAP_OK flag will be set to "1". The user must write "1" to ECAP_OK_CLR to clear the ECAP_OK flag to "0" (Figure 19.1.1).

19.1.1 EXT_CAPCONT (External Capture Control Register)

EXT_CAPCONT Address = DDH (SFR_PAGE = 0) Reset Value = 0x03H

External Capture Control Register

	ECAP_EN	ECAP_OK	-----	CAPPINSEL	ECAP_OK_CLR	ECAPCKS[2:0]		
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	X	R/W	W	R/W	R/W	R/W

ECAP_EN [7]	External Capture Enable : 0 : External Capture Disable 1: External Capture Enable
ECAP_OK [6]	External Capture OK : 0 : External signal is capturing 1: External signal capture down, clear by write “1” in **ECAP_OK_CLR**
CAPPINSEL [4]	Capture input pin select : 0 : CAP2(P0.7) 1: CAP1(P0.6)
ECAP_OK_CLK [3]	External Capture OK Clear : Write “1” to clear the **ECAP_OK** bit
ECAPCKS [2:0]	External Capture Clock Select : 000 : 48MHz/4 100 : 48MHz/64 001 : 48MHz/8 101 : 48MHz/128 010 : 48MHz/16 110 : 48MHz/256 011 : 48MHz/32 111 : 48MHz/512

Table 19.1.1 : External Capture Control Register

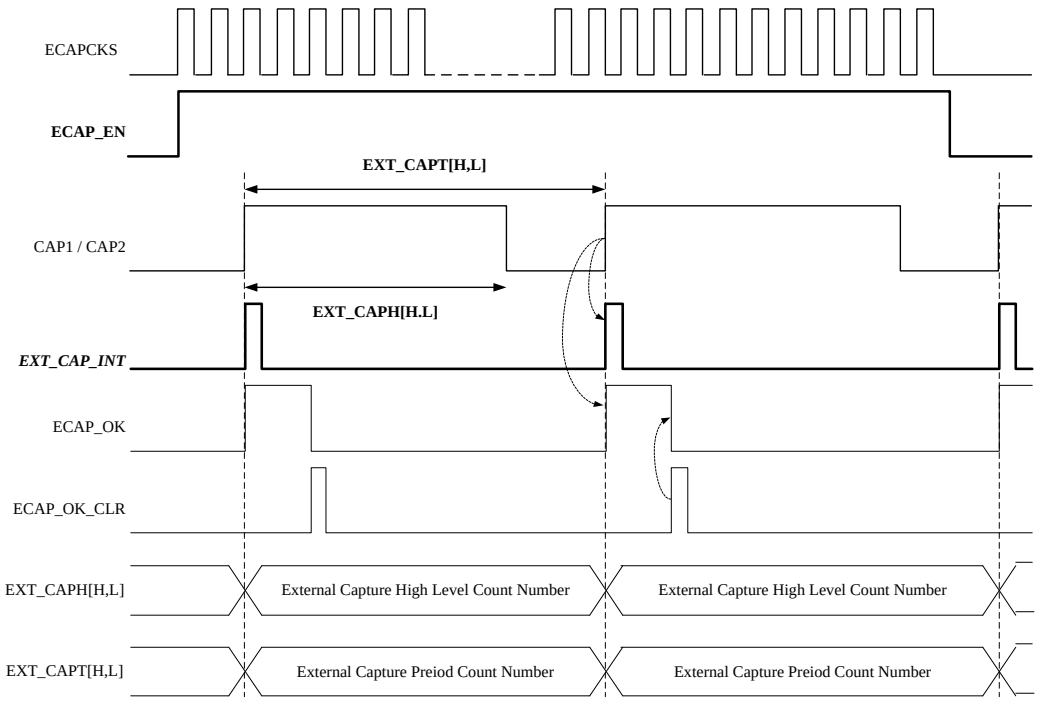


Figure 19.1.1 : The Waveform Sequence of External Capture

19.1.2 EXT_CAPT (External Capture Total Count)

When an external capture interrupt occurs, in addition to the External Capture Total Count being updated to the latest capture count value, it will also re-count the new capture cycle length. The timing is shown in Figure 19.1.1.

EXT_CAPT_L Address = D9H (SFR_PAGE=0) Reset Value = 0x00H

External Capture Total Count Low

Bit	EXT_CAPT[7:0]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

EXT_CAPT_H Address = DAH (SFR_PAGE=0) Reset Value = 0x00H

External Capture Total Count High

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

19.1.3 EXT_CAPH (External Capture High-Level Count)

The value of External Capture High-Level Count will be temporarily stored when the capture signal changes from "High" to "Low". When the capture interrupt occurs, it will be updated to the latest value at the same time as the External Capture Total Count. The timing is shown in Figure 19.1.1.

EXT_CAPH_L Address = DBH (SFR_PAGE=0) Reset Value = 0x00H

External Capture High-level Count Low

Bit	EXT_CAPH[7:0]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

EXT_CAPH_H Address = DCH (SFR_PAGE=0) Reset Value = 0x00H

External Capture High-level Count High

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

19.2 Internal Capture

Internal capture input data is from INT_CAPCONT (Table 19.2.1), and the setting method is as shown in bit [4] (IN_CAP_IN) in the table below. In addition, the clock frequency for counting the capture signal can be selected according to the cycle length of the input signal (set by ICAPCKS in Table 19.2.1), which can avoid overflow in the internal counter and cause distortion of the high level length and cycle length of the capture signal. When the capture of the internal signal is completed, an interrupt will be issued and the ICAP_OK flag will be set to "1". The user must write "1" to ICAP_OK_CLR to clear the ICAP_OK flag to "0" (Figure 19.2.1).

19.2.1 INT_CAPCONT (Internal Capture Control Register)

INT_CAPCONT Address = DDH (SFR_PAGE = 1) Reset Value = 0x03H

Internal Capture Control Register

	ICAP_EN	ICAP_OK	-----	INT_CAP_IN	ICAP_OK_CLR	ICAPCKS[2:0]		
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	X	R/W	W	R/W	R/W	R/W

ICAP_EN [7]	Internal Capture Enable : 0 : Internal Capture Disable 1: Internal Capture Enable
ICAP_OK [6]	Internal Capture OK : 0 : Internal signal is capturing 1: Internal signal capture down, clear by write “1” in **ICAP_OK_CLR**
INT_CAP_IN [4]	Internal Capture input Data : 0 : Data “0” input 1: Data “1” input
ICAP_OK_CLK [3]	Internal Capture OK Clear : Write “1” to clear the **ICAP_OK** bit
ICAPCKS [2:0]	Internal Capture Clock Select : 000 : 48MHz/4 100 : 48MHz/64 001 : 48MHz/8 101 : 48MHz/128 010 : 48MHz/16 110 : 48MHz/256 011 : 48MHz/32 111 : 48MHz/512

Table 19.2.1 : Internal Capture Control Register

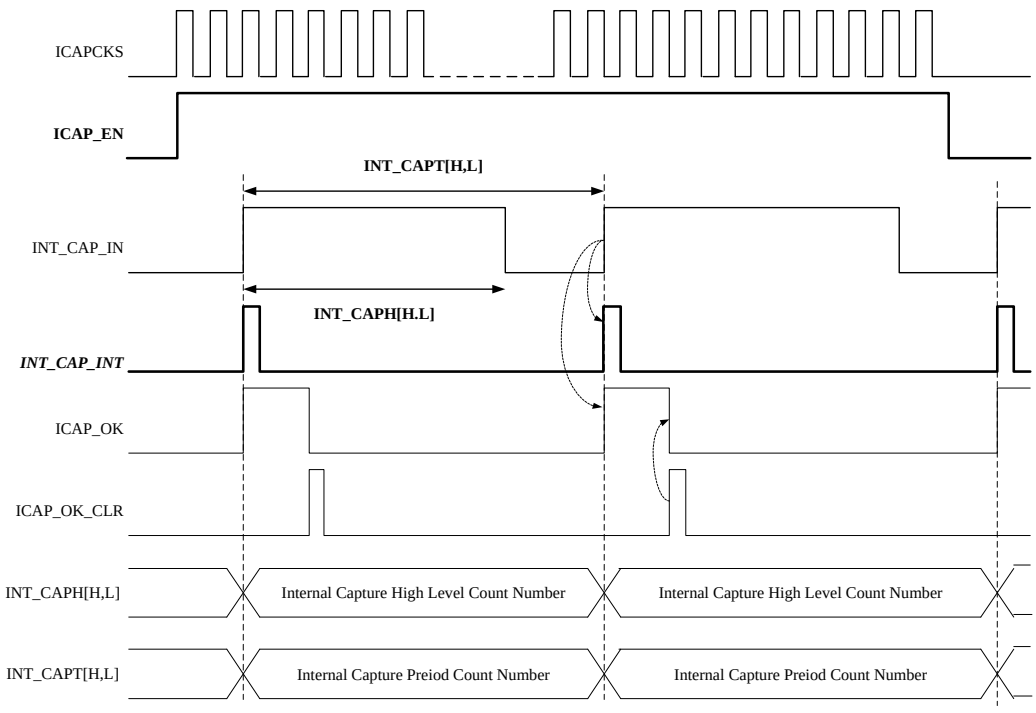


Figure 19.2.1 : The Waveform Sequence of Internal Capture

19.2.2 INT_CAPT (Internal Capture Total Count)

When an internal capture interrupt occurs, in addition to the Internal Capture Total Count being updated to the latest capture count value, it will also re-count the new capture cycle length. The timing is shown in Figure 19.2.1.

INT_CAPT_L Address = D9H (SFR_PAGE=1) Reset Value = 0x00H

Internal Capture Total Count Low

Bit	INT_CAPT[7:0]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

INT_CAPT_H Address = DAH (SFR_PAGE=1) Reset Value = 0x00H

Internal Capture Total Count High

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

19.2.3 INT_CAPH (Internal Capture High-Level Count)

The value of Internal Capture High-Level Count will be temporarily stored when the capture signal changes from "High" to "Low". When the capture interrupt occurs, it will be updated to the latest value at the same time as the Internal Capture Total Count. The timing is shown in Figure 19.2.1.

INT_CAPH_L Address = DBH (SFR_PAGE=1) Reset Value = 0x00H

Internal Capture High-level Count Low

Bit	INT_CAPH[7:0]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

INT_CAPH_H Address = DCH (SFR_PAGE=1) Reset Value = 0x00H

Internal Capture High-level Count High

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

20. Multiplication and Division Unit (MDU)

The MDU is an on-chip arithmetic co-processor which enables the CGF022A to perform additional extended arithmetic operations. All operations are singed/unsigned integer operations. Operands and results are stored in MD0–MD5 registers. The module is controlled by the MD_MODE and MD_CONT register. Any calculation of the MDU overwrites its operands. The MDU support five operations: Division 32-bit/16-bit, Division 16-bit/16-bit, Multiplication, Shift and Normalize.

SFR	Description	address	Reset value
MD_CONT	MDU Control Register	ADH	00H
MD_MODE	MDU Mode Control Register	ACH	10H
MD0	Multiplication Division Register 0	AEH	00H
MD1	Multiplication Division Register 1	AFH	00H
MD2	Multiplication Division Register 2	B1H	00H
MD3	Multiplication Division Register 3	B2H	00H
MD4	Multiplication Division Register 4	B3H	00H
MD5	Multiplication Division Register 5	BAH	00H

20.1 MD_MODE (MDU Control Mode)

MD_MODE Address = ACH Reset Value = 0x10H

MDU Mode Control Register

Bit	-----	-----	-----	MDUF	-----	-----	MDUS	----
	7	6	5	4	3	2	1	0
Type	X	X	X	R	X	X	R/W	X

MDUF [4]	MDU finish flag : 0 : MDU busy. 1 : MDU calculation finished.
MDUS [1]	MDU Signed select : 0 :Signed calculation. 1 :Unsigned calculation.

20.2 MD_CONT (MDU Control Register)

MD_CONT Address = ADH Reset Value = 0x10H

MDU Control Register

Bit	MDEF	MDOV	SLR	SC[4:0]				
	7	6	5	4	3	2	1	0
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W

MDEF	MDU Error flag :
[7]	Indicates an improperly performed operation (when one of the arithmetic operations has been restarted or interrupted by a new operation).
MDOV	MDU Overflow flag :
[6]	Overflow occurrence in the MDU operation.
SLR	Shift direction :
[5]	0 : shift left operation 1 : shift right operation
SC	Shift counter :
[4:0]	When set to all ‘0‘s, normalize operation is selected. After normalization, the SC[4:0] contain the number of normalizing shifts performed. When at least one of these bit is set high shift operation is selected. The number of shifts performed is determined by the number written to SC[4:0], where SC.4 is the MSB.

20.2.1 MDEF

The MDEF error flag indicates an improperly performed operation (when one of the arithmetic operations has been restarted or interrupted by a new operation). The error flag mechanism is automatically enabled with the first write operation to MD0and disabled with the final read instruction from MD3(multiplication or shift/norm) or MD5 (division) in phase three.

The error flag is set when:

There is a write access to MDx registers (any of MD0-MD5 and MD_CONT) during phase two of MDU operation (restart or calculations interrupting) There is a read access to one of MDx registers during phase two of MDU operation when the error flag mechanism is enabled. In such condition error flag is set but the calculation is not interrupted. The error flag is reset only after read access to MD_CONT register. The error flag is read only.

20.2.2 MDOV

The MDOV overflow flag is set when one of the following conditions occurs: Division by zero
Multiplication with a result greater than FFFFH
Start of normalizing if the ('MD3.7' ='1') most significant bit of MD3 is set
Any operation of the MDU that does not match the above conditions clears the overflow flag. Note that the overflow flag is exclusively controlled by hardware. It cannot be written.

20.3 MD0 – MD5 (Multiplication Division Register)

MD0 Address = AEH Reset Value = 0x00H
Multiplication Division Register 0

Bit	MD0[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MD1 Address = AFH Reset Value = 0x00H
Multiplication Division Register 1

Bit	MD1[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MD2 Address = B1H Reset Value = 0x00H
Multiplication Division Register 2

Bit	MD2[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MD3 Address = B2H Reset Value = 0x00H
Multiplication Division Register 3

Bit	MD3[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MD4 Address = B3H Reset Value = 0x00H
Multiplication Division Register 4

Bit	MD4[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MD5 Address = BAH Reset Value = 0x00H
Multiplication Division Register 5

Bit	MD5[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.4 MDU Operation Description

The operation of the MDU consists of three phases:

20.4.1 Loading the MDx registers

The type of calculation the MDU has to perform is selected by the order in which the MDx registers are written to. A write to MD0 is the first transfer to be done in any case. Next writes must be done as shown in the table below to determine the MDU operation. The last write will start the selected operation.

Operation	32-bit/16-bit	16-bit/-16bit	16-bit x 16-bit	Shift/ normalizing
First write	MD0 Dividend Low	MD0 Dividend Low	MD0 Multiplicand Low	MD0 LSB
	MD1 Dividend	MD1 Dividend High	MD4 Multiplicator Low	MD1
	MD2 Dividend		MD1 Multiplicand High	MD2
	MD3 Dividend High			MD3 MSB
	MD4 Divisor Low	MD4 Divisor Low		
Last write	MD5 Divisor High	MD5 Divisor High	MD5 MultiplicatorHigh	MD_CONT start conversion

20.4.2 Executing calculation

During the calculation period, the MDU works in parallel to the CPU. When the calculation is complete, the hardware will set the MDUF bit to one (MDUF = '1'). The flag will be cleared at the next calculation.

The following table provides the execution time for each mathematical operation.

Operation	Number of clock cycles	
Division 32-bit/16-bit	17 clock cycles	
Division 16-bit/16-bit	9 clock cycles	
Multiplication	11 clock cycles	
Shift	Min 3 clock cycles (SC = 01H)	Max 18 clock cycles (SC = 1FH)
Normalize	Min 4 clock cycles (SC <= 01H)	Max 19 clock cycles (SC = 1FH)

20.4.3 Reading the result from the MDx registers

The Read-out sequence of the first "MDx" registers is not critical but the last read determines the end of a whole calculation.

Operation	32-bit/16-bit	16-bit/16-bit	16-bit x 16-bit	Shift/ normalizing
First read	MD0 Quotient Low	MD0 Quotient Low	MD0 Product Low	MD0 LSB
	MD1 Quotient	MD1 Quotient High	MD1 Product	MD1
	MD2 Quotient		MD2 Product	MD2
	MD3 Quotient High			
	MD4 Remainder Low	MD4 Remainder Low		
Last read	MD5 Remainder High	MD5 Remainder High	MD3 Product High	MD3 MSB

20.4.4 Shifting

In shift operation, 32-bit integer variable stored in MD0 to MD3 registers (the latter contains the most significant byte) is shifted left or right by a specified number of bits. The SLR bit (MD_CONT.5) defines the shift direction, and bits SC[4:0] (MD_CONT.4 –MD_CONT.0) specifies the shift count (which must not be 0). During shift operation, zeroes come into the left end of MD3 for shifting right or right end of the MD0 for shifting left.

20.4.5 Normalizing

All leading zeroes of 32-bit integer variable stored in MD0 to MD3 registers, the latter contains the most significant byte are removed by shift left operations. The whole operation is completed when the MSB (most significant bit) of MD3 register contains a ‘1‘. After normalizing, bits SC[4:0] (MD_CONT.4 –MD_CONT.0) contain the number of shift left operations, which were done.

21. Cyclic Redundant Check (CRC)

The CRC check function in CGF022A is based on CRC16-CCITT polynomial X16 + X12 + X5 + 1 (Figure 21.1). The general architecture of CRC computation is using LSFR and XOR by serial. The CRC function can support byte data and flash bank (128-byte per bank) CRC computation, the block diagram is shown as Figure 21.2 and the example of bank assign for bank CRC computation is shown as Figure 21.3. The data selection and the control are setting by the CRC XSFRs (Table 21.1).

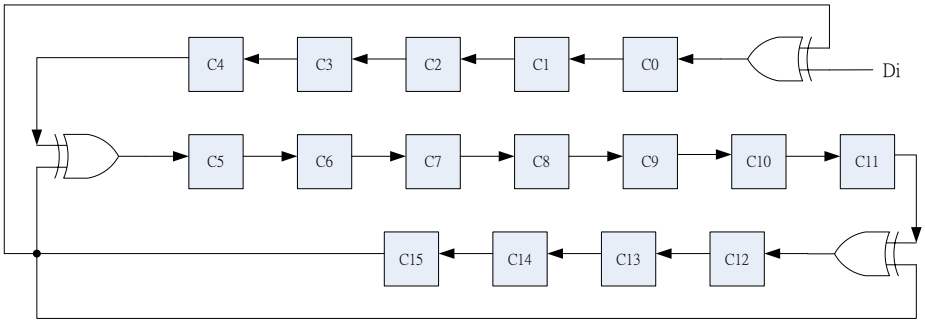


Figure 21.1 : The general LSFR architecture of CRC16-CCITT

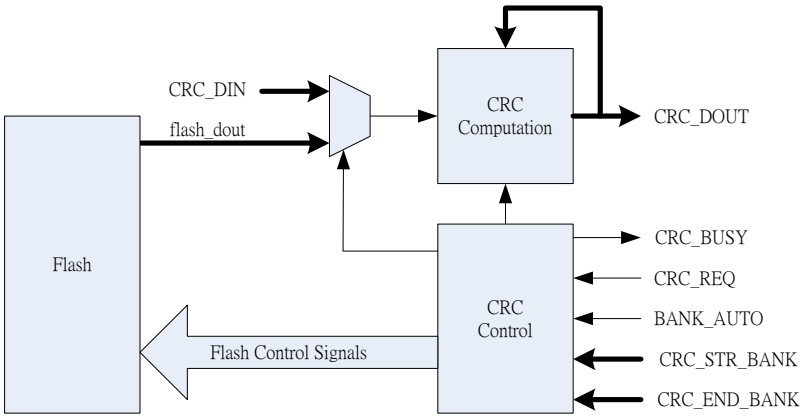
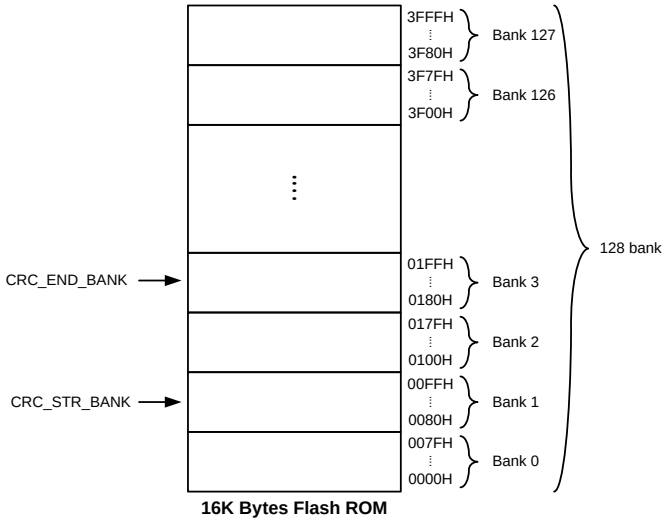


Figure 21.2 : The byte and flash bank CRC computation architecture



EX:
CRC_STR_BANK = 0x01
CRC_END_BANK = 0x03
The bank range for CRC computation : (0x01 ~ 0x03) bank

Figure 12.3 : The example of bank assign for bank CRC computation

XSFR	Description	Address	Reset value
CRC_CTRL	CRC Control Register	1018H	00H
CRC_DIN	CRC Input Data Register	1019H	00H
CRC_DOUT_L	CRC Output Remainder Low Byte Data Register	101AH	00H
CRC_DOUT_H	CRC Output Remainder High Byte Data Register	101BH	00H
CRC_STR_BANK	Start Bank Index for Flash Bank CRC Computation	101CH	00H
CRC_END_BANK	End Bank Index for Flash Bank CRC Computation	101DH	00H

Table 21.1 : CRC XSFR List

21.1 CRC_CTRL (CRC Control Register)

CRC_CTRL Address = 1018H Reset Value = 0x00H

CRC Control Register

Bit	7	6	5	4	3	2	1	0
	CRC_REQ	CRC_BUSY	-----	-----	-----	-----	-----	BANK_AUTO
Type	W	W	-----	-----	-----	-----	-----	R/W

CRC_REQ [7]	CRC Computation Request: 1 : Write “1” for CRC Computation Request
CRC_BUSY [6]	CRC Computation Busy: 0 : CRC Computation Down 1 : CRC Computation Busy
BANK_AUTO [0]	Enable Bank CRC Computation: 0 : Disable (CRC_DIN 1-byte CRC Computation). 1 : Enable (Bank multi-byte CRC Computation)

21.2 CRC_DIN (CRC Data Register)

CRC_DIN Address = 1019H Reset Value = 0x00H

CRC Input Data Register

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

21.3 CRC_DOUT (CRC Output Remainder Data Register)

CRC_DOUT_L Address = 101AH Reset Value = 0x00H

CRC Output Remainder Data Register Low Byte

Bit	CRC_DOUT [7:0]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

CRC_DOUT_H Address = 101BH Reset Value = 0x00H

CRC Output Remainder Data Register High Byte

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

21.4 CRC_STR_BANK (Start Bank Index Register for Bank CRC Computation)

CRC_STR_BANK Address = 101CH Reset Value = 0x00H

Start Bank Index for Flash Bank CRC Computation

Bit	CRC_STR_BANK							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CRC_STR_BANK [7:0]	Start Bank Index for Flash Bank CRC Computation: 0~127

21.5 CRC_END_BANK (End Bank Index Register for Bank CRC Computation)

CRC_END_BANK Address = 101DH Reset Value = 0x00H

End Bank Index for Flash Bank CRC Computation

Bit	CRC_END_BANK							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CRC_END _BANK [7:0]	End Bank Index for Flash Bank CRC Computation: 0~127

22. Independent General PWM

The CGF022A support one16-bit independent general PWM output for other duty control application (Figure 22.1). The IPWM control and setting XSFRs is shown as Table 22.1.

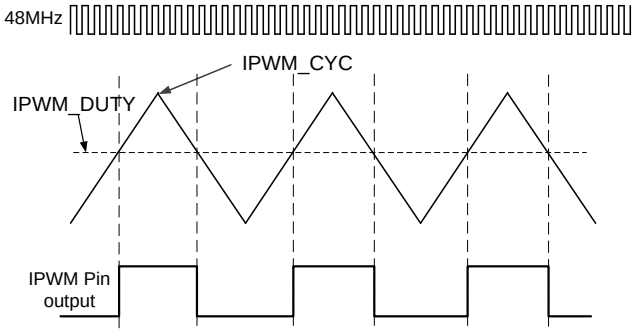


Figure 22.1 : Independent general PWM generator

XSFR	Description	Address	Reset value
IPWM_CYC_L	Independent General PWM Cycle Low Byte	1020H	02H
IPWM_CYC _H	Independent General PWM Cycle High Byte	1021H	00H
IPWM_DUTY_L	Independent General PWM Duty Low Byte	1022H	FFH
IPWM_DUTY_H	Independent General PWM Duty High Byte	1023H	FFH
IPWM_CTRL	Independent General PWM Control Register	1024H	00H

Table 22.1 : IPWM XSFR List

21.1 IPWM_CTRL (Independent General PWM Register)

IPWM_CTRL Address = 1024H Reset Value = 0x00H

Independent General PWM Control Register

	----	----	----	----	----	IPWM_MODE		IPWM_EN
Bit	7	6	5	4	3	2	1	0
Type	----	----	----	----	----	R/W		R/W

IPWM_MODE [2:1]	Independent General PWM mode select: 00 : Force Low 01 : Force High 10 : Active High 11 : Active Low
IPWM_EN [0]	Independent General PWM enable: 0 : Disable 1 : Enable.

21.2 IPWM_CYC (Independent General PWM Cycle Register)

IPWM_CYC_L Address = 1020H Reset Value = 0x02H

Independent General PWM Cycle Register Low Byte

	IPWM_ MAX[7:0]							
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IPWM_CYC_H Address = 1021H Reset Value = 0x00H

Independent General PWM Cycle Register High Byte

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

21.3 IPWM_DUTY (Independent General PWM Duty Register)

IPWM_DUTY_L Address = 1022H Reset Value = 0xFFH

Independent General PWM Duty Low Byte

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

IPWM_DUTY_H Address = 1023H Reset Value = 0xFFH

Independent General PWM Duty High Byte

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

23. One Wire RF/IR Signal Decode

The one wire RF/IR signal decode function can decode one wire data automatically by hardware that after setting the data format and type by IR decode XSFR. The maximum decode data is 48 bits. It can decode the output data of RF receiver module or control board. The block diagram is shown as Figure 23.1. The one wire RF/IR signal decode function related XSFRs are shown in Table 23.1.

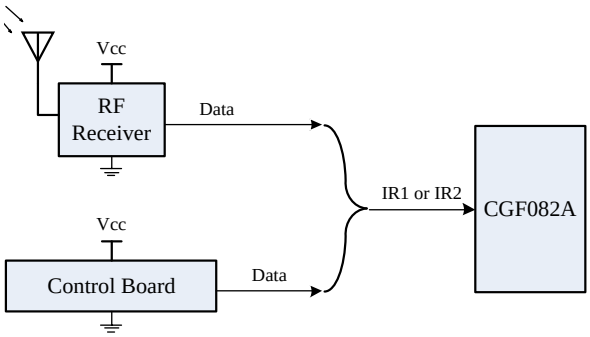


Figure 23.1 : One wire signal decode by CGF022A

XSFR	Description	Address	Reset value
IR_DEC_SET	IR Data Decode Setting Register	1038H	00H
IR_DEC_CTRL	IR Data Decode Control Register	1039H	00H
IR_HEADER_Z1_L	IR Data HEADER Zone1 Cycle Register Low Byte	103AH	80H
IR_HEADER_Z1_H	IR Data HEADER Zone1 Cycle Register High Byte	103BH	BBH
IR_HEADER_Z2_L	IR Data HEADER Zone2 Cycle Register Low Byte	103CH	00H
IR_HEADER_Z2_H	IR Data HEADER Zone2 Cycle Register High Byte	103DH	7DH
IR_STOP_Z_L	IR Data STOP Zone Cycle Register Low Byte	103EH	80H
IR_STOP_Z_H	IR Data STOP Zone Cycle Register High Byte	103FH	BBH
IR_DOUT0	IR Decode Output Data Byte0	1040H	00H
IR_DOUT1	IR Decode Output Data Byte1	1041H	00H
IR_DOUT2	IR Decode Output Data Byte2	1042H	00H
IR_DOUT3	IR Decode Output Data Byte3	1043H	00H
IR_DOUT4	IR Decode Output Data Byte4	1044H	00H
IR_DOUT5	IR Decode Output Data Byte5	1045H	00H

Table 23.1 : One wire RF/IR signal decode XSFR List

23.1 IR_DEC_SET (IR Data Decoder Setting Register)

Since one wire RF/IR signal has various types, we provide the IR_DEC_SET setting register for users to set the one wire RF/IR data type they want to receive (Table 23.1.1). Including whether the received data has a header (IR_DEC_SET[7], Figure 23.1.1), the decoding pattern of the received data bits (IR_DEC_SET[5], Figure 23.1.2), and the first bit received is the MSB or LSB (IR_DEC_SET[6], Figure 23.1.3), etc. can be set in the IR_DEC_SET register.

If the received signal includes HEADER, you must also set the HEADER high-level time length (Table 23.3.1) and low-level time length (Table 23.4.1); for STOP, you only need to set a time length (Table 23.4.1). In addition, the clock frequency for counting the signal can be selected according to the HEADER or STOP length of the input signal (set by DEC_CLK_SEL in Table 23.1.1), which can avoid overflow in the internal counter and cause distortion of the one wire signal.

IR_DEC_SET Address = 1038H Reset Value = 0x00H

IR Data Decoder Setting Register

	HEADER_EN	DOUT_REV	DIN_TYP	DIN_DB		DEC_CLK_SEL		
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W		R/W		

HEADER_EN [7]	Input Data with HEADER: 0 : No HEADER 1 : With HEADER
DOUT_REV [6]	IR Decode Output Data Reverse: 0 : First input data is LSB 1 : First input data is MSB
DIN_TYP [5]	Input Data Type : 0 : Data Type1: Falling to Rising edge: Duty > 50% → Din = 0; Duty < 50% → Din = 1 1 : Data Type2: Falling to Rising edge: Duty > 50% → Din = 1; Duty < 50% → Din = 0
DIN_DB_SEL [4:3]	Input Data De-bounce Time Select : 00 : 0 ns 01 : 250 ns 10 : 500 ns 11 : 1000 ns
DEC_CLK_SEL [2:0]	Data Decode Clock Frequency Select : 000 : 24 MHz 001 : 16 MHz 010 : 8 MHz 011 : 6 MHz 100 : 3 MHz 101: 2 MHz

Table 23.1.1 : IR Data Decoder Setting Register

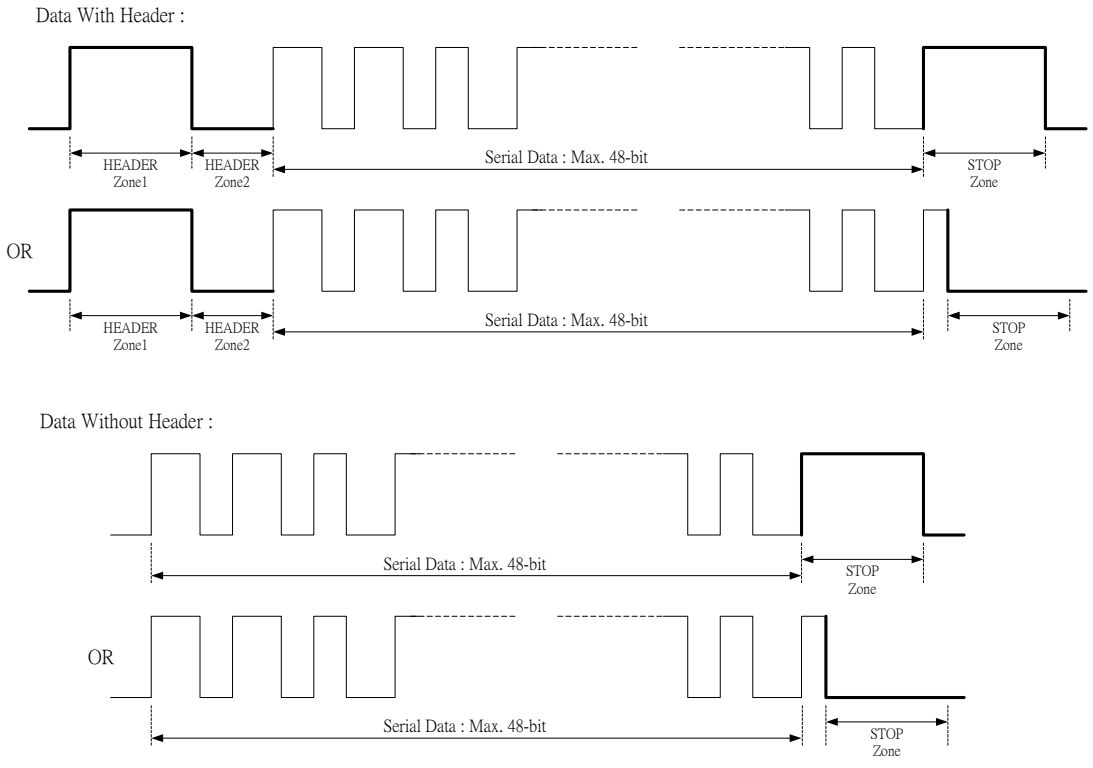


Figure 23.1.1 : The header and stop signal express

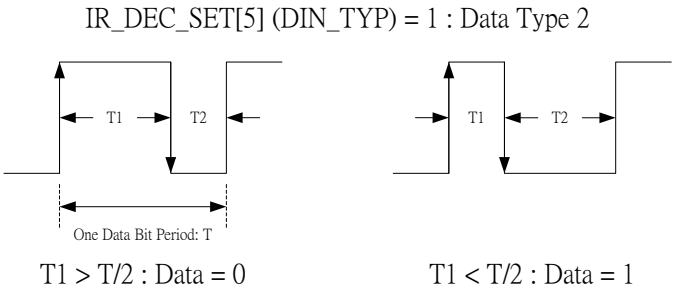


Figure 23.1.2 : The input data type select

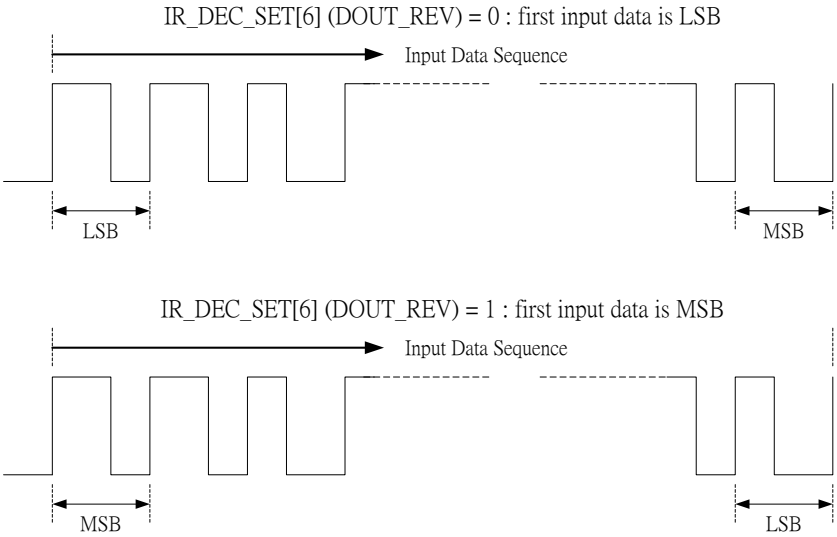


Figure 23.1.3 : The output data reverse select

23.2 IR_DEC_CTRL (IR Data Decoder Control Register)

One wire signal input can be selected from IR1 or IR2, and the setting method is as shown in bit[6] of IR_DEC_CTRL register (IRIN_SEL) in the Table 23.2.1. When IR_DEC_EN (Table 23.2.1 bit[7]) is set to "1", decoding of the input signal will begin. When the one wire signal decoding is completed, the IRDEC_OK flag will be set to "1", and the decoded data will be placed in the IR Decode output register (Table 23.6 .1). The user must write "1" to IRDEC_OK_CLR (Table 23.2.1 bit[0]) to clear the IRDEC_OK flag to "0".

IRDEC_CTRL Address = 1039H Reset Value = 0x00H

IR Data Decoder Control Register

	IR_DEC_EN	IRIN_SEL	---	IRDEC_OK	---	---	---	IRDEC_OK_CLR
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	---	R/W	---	---	---	W

IRDEC_EN [7]	IR Decode Enable: 1 : Enable IR Decode.
IRIN_SEL [6]	IR Decode Data Input Select: 0 : IR decode data input from IR1 (PIN CH3) 1 : IR decode data input from IR2 (PIN CH7)
IRDEC_OK [4]	IR Data Decode OK Flag: 1 : IR Input Decode Finish. Clear by write “1” at IRDEC_OK_CLR bit.
IRDEC_OK_CLR [0]	Clear IR Data Decode OK Flag: 1 : Write “1” to Clear IRDEC_OK Flag .

Table 23.2.1 : IR Data Decoder Control Register

23.3 IR_HEADER_Z1 (IR Data Header Zone1 Cycle Register)

IR_HEADER_Z1_L Address = 103AH Reset Value = 0x80H

IR Data HEADER Zone1 Cycle Register Low Byte

	IR_HEADER_Z1[7:0]							
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IR_HEADER_Z1_H Address = 103BH Reset Value = 0xBBH

IR Data HEADER Zone1 Cycle Register High Byte

	IR_HEADER_Z1[15:8]							
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.3.1 : IR Data Header Zone1 Cycle Register

23.4 IR_HEADER_Z2 (IR Data Header Zone2 Cycle Register)

IR_HEADER_Z2_L Address = 103CH Reset Value = 0x00H

IR Data HEADER Zone2 Cycle Register Low Byte

Bit	IR_HEADER_Z2[7:0]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IR_HEADER_Z2_H Address = 103DH Reset Value = 0x7DH

IR Data HEADER Zone2 Cycle Register High Byte

Bit	IR_HEADER_Z2[15:8]							
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.4.1 : IR Data Header Zone2 Cycle Register

23.5 IR_STOP_Z (IR Data STOP Zone Cycle Register)

IR_STOP_Z_L Address = 103EH Reset Value = 0x80H

IR Data STOP Zone Cycle Register Low Byte

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

IR_STOP_Z_H Address = 103FH Reset Value = 0xBBH

IR Data STOP Zone Cycle Register High Byte

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

Table 23.5.1 : IR Data STOP Zone Cycle Register

23.6 IR_DOUT0~IR_DOUT5 (IR Decode Output Register Byte0 ~ Byte5)

IR_DOUT0 Address = 1040H Reset Value = 0x00H
IR Decode Output Data Byte0

Bit	IR_DOUT [7:0]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

IR_DOUT1 Address = 1041H Reset Value = 0x00H
IR Decode Output Data Byte1

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

IR_DOUT2 Address = 1042H Reset Value = 0x00H
IR Decode Output Data Byte2

Bit	IR_DOUT [23:16]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

IR_DOUT3 Address = 1043H Reset Value = 0x00H
IR Decode Output Data Byte3

Bit	IR_DOUT [31:24]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

IR_DOUT4 Address = 1044H Reset Value = 0x00H
IR Decode Output Data Byte4

Bit	IR_DOUT [23:16]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

IR_DOUT5 Address = 1045H Reset Value = 0x00H
IR Decode Output Data Byte5

Bit	IR_DOUT [31:24]							
	7	6	5	4	3	2	1	0
Type	R	R	R	R	R	R	R	R

Table 23.6.1 : IR Decode Output Data Register Byte0~Byte5

24. Software Reset

The CGF022A support software reset function for reset system by user. It is must release the software reset key before software reset enable. The software reset XSFRs is shown as Table 24.1.

XSFR	Description	Address	Reset value
SOFT_RST_KEY	Software Reset Key Register	1028H	00H
SOFT_RST_EN	Software Reset Enable Register	1029H	00H

Table 24.1 : IPWM XSFR List

24.1 SOFT_RST_KEY (Software Reset Key Register)

SOFT_RST_KEY Address = 1028H Reset Value = 0x00H

Software Reset Key Register

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

Software reset key, must write three specific values AAH, 55H and A5H to the SOFT_RST_KEY for enable the SOFT_RST_EN write available.

The sequence is :

MOV SOFT_RST_KEY, #AAh
MOV SOFT_RST_KEY, #55h
MOV SOFT_RST_KEY, #A5h

24.2 SOFT_RST_EN (Software Reset Enable Register)

SOFT_RST_EN Address = 1029H Reset Value = 0x00H

Software Reset Enable Register

Bit	SRST_EN	-----	-----	-----	-----	-----	-----	-----
	7	6	5	4	3	2	1	0
Type	W	X	X	X	X	X	X	X

SRST_EN Software reset enable :

0 : Disable Software Reset.

1 : Enable Software Reset.

25. Motor Driving Engine (MDE)

25.1 MDE Architecture

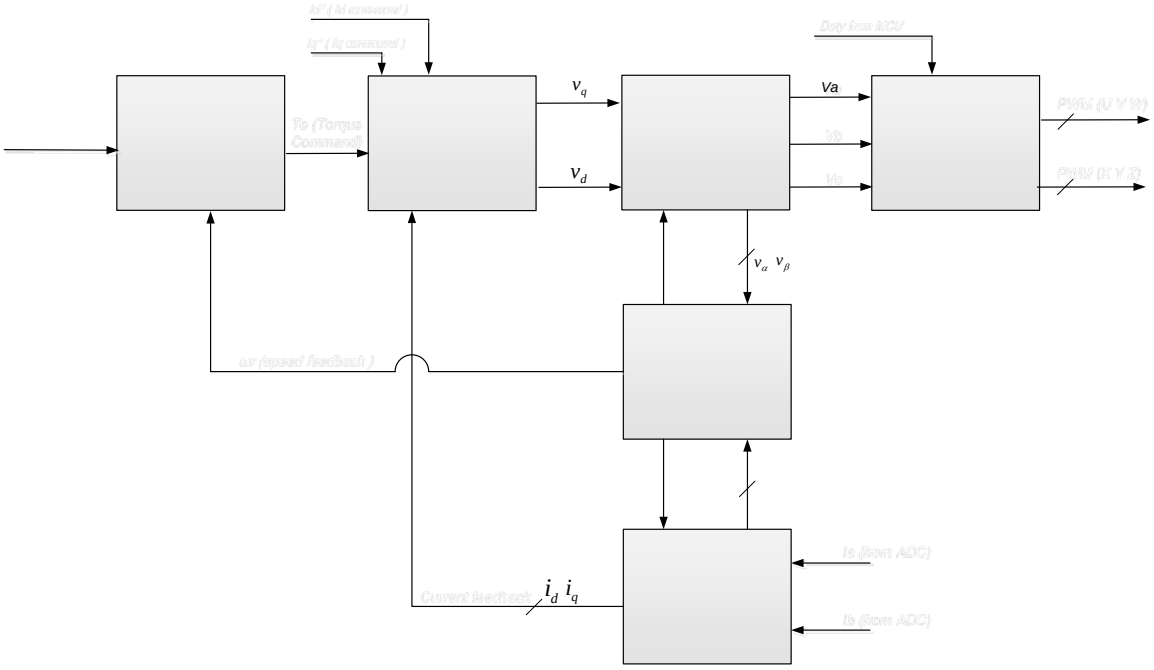


Figure 25.1 : MDE Architecture

CGF0X2AXX support PMSM Field Oriented Control, The description of FOC control architecture is shown in Figure 25.1, which includes 6 control blocks including Speed Control Loop, Current Control Loop, SMO Rotor Position Estimation, SVPWM Engine, Coordinate Transformation and Inverse Coordinate Transformation.

First, the ia and ib phase current signals of the 3 phase DC motor are converted into iα and iβ signals in the form of stationary referenece frame through the Coordinate Transformation blcok. At the same time, the control voltages Vd and Vq are also converted into the stationary referenece frame form through the Inverse Coordinate Transformation Block. The Vα, Vβ signals, these stationary referenece frame signals iα, iβ, Vα and Vβ are used in the SMO Rotor Position Estimation Block to estimate the motor rotor position signal θr and rotational speed signal ωr.

In the Speed Control Loop Block, the speed command *ωs* and the estimated speed signal *ωr* are used to calculate the toque control quantity *Te* through the speed control law. *Te* is also the current command of the Current Control Loop Block, using the estimated motor rotor position signal θr, In Coordinate Transformation Block, the stationary reference frame current signal is converted into a synchronized reference frame current signal, Id and Iq, as the feedback signal of the Current Control Loop Block, the command and feedback current calculate the voltage control quantities Vd and Vq through the current control law, The voltage control quantities Vd and Vq are then converted into the three-phase motor control phase voltages Va, Vb and Vc through the Inverse Coordinate Transformation Block.

The SVPWM Engine generates the pluse width modulation signals PWM_{UVW} and PWM_{XYZ} of the frequency conversion circuit. The detailed control function description of each Block will be detailed in the following chapters.

25.2 Coordinate Transformation Block

25.2.1 Coordinate Transformation Block Architecture

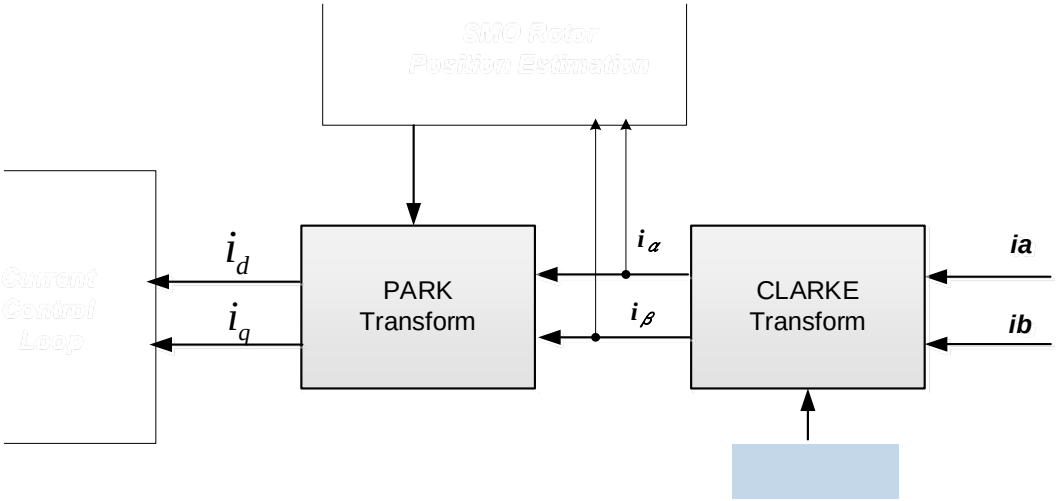
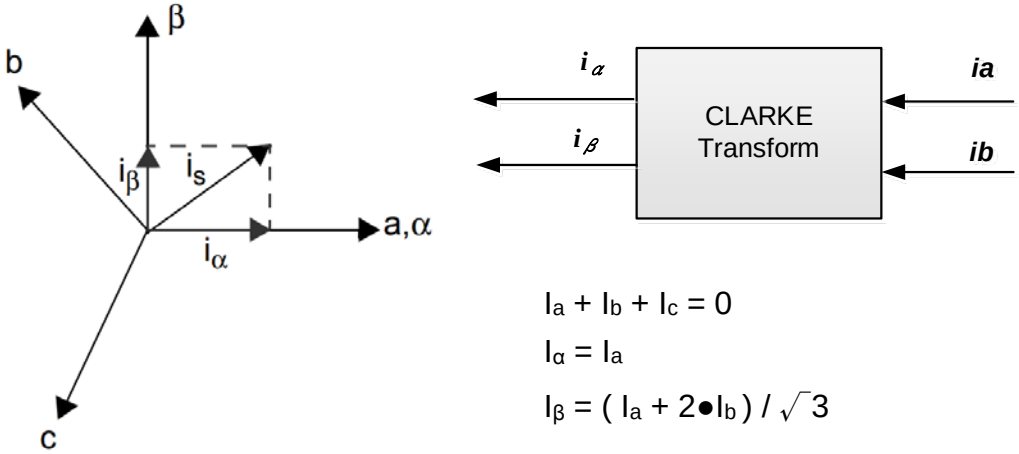


Figure 25.2.1 : Coordinate Transformation Block Architecture

25.2.2 CLAEKE Transform

The first coordinate transform, called the CLAEKE Transform, moves a three-axis, two-dimensional coordinate system, referenced to the stator, onto a two-axis system, keeping the same reference (see Figure25.2.2, where ia, ib and ic are the individual phase currents).



$I_a + I_b + I_c = 0$

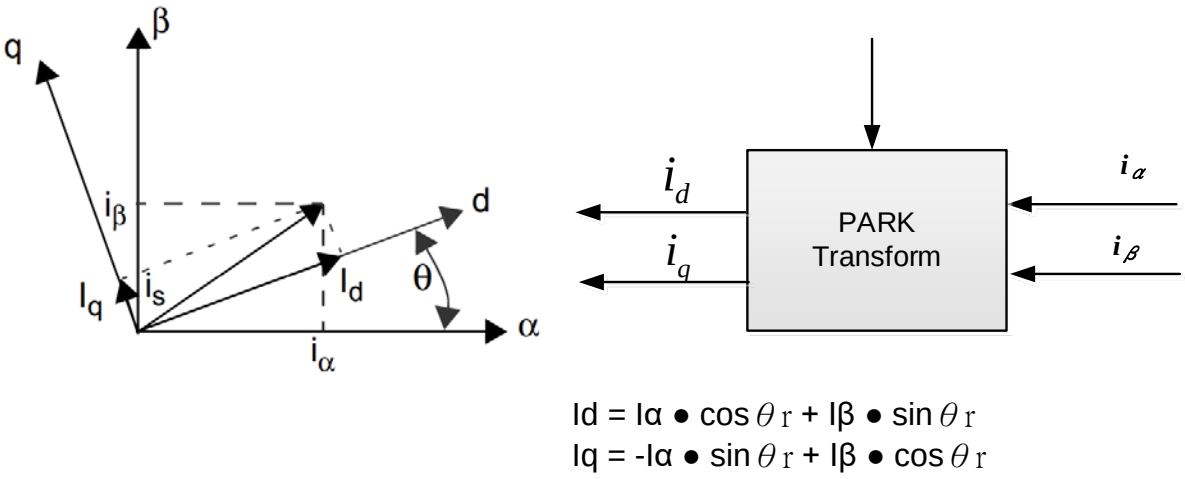
$I_\alpha = I_a$

$I_\beta = (I_a + 2 \bullet I_b) / \sqrt{3}$

Figure 25.2.2 : Cleake Transform

25.2.3 PARK Transform

At this point, you have the stator current represented on a two-axis orthogonal system with the axis called α-β. The next step is to transform into another two-axis system that is rotating with the rotor flux. This transformation uses the PARK Transform, as illustrated in Figure25.2.3. This two-axis rotating coordinate system is called the d-q axis. θ represents the rotor angle.



Id = Iα ● cos θr + Iβ ● sin θr
Iq = -Iα ● sin θr + Iβ ● cos θr

Figure 25.2.3 : Park Transform

25.2.4 Coordinate Transformation Block description and setting

ESTCR control register is in the FOCCONT[6:4] (Address : D6H) , ESTCR can set the scaling ratio of iα and iβ when entering SMO Rotor Position Estimation Block. Please refer to the ESTCR description in the table below.

FOCCONT Field Oriented Control Register		Address = D6H				Reset Value = 0x00H		
Bit	PI CLEAR	ESTCR			INV ADCD	ADC TRIG	PLL EN	SPEED EN
		6	5	4				
Type		R/W	R/W	R/W				

ESTCR	Iα and Iβ current ratio (to SMO Rotor Position Estimation block)	
[6:4]	000 : Current ValueX1	100 : Current Value/2
	001 : Current ValueX2	101 : Current Value/4
	010 : Current ValueX4	110 : Current Value/8
	011 : Current ValueX8	111 : Current Value/16

Table25.2.1 : IESTCR SFR

In the Coordinate Transformation Block, the ia, ib, iα and iβ current value can be observed through XSFR IA, IB, IALPHA, and IBETA; iq and id current value can be observed through the page0 and page1 of PI_FB SFR.

For the related reading process in page of SFR, first set the SFR_PAGE (SFR Address = ECH) where the signal is located, and then read the corresponding SFR position data to read the signal value.

The reading process is as follows:

(1) Read Ia signal vale : Read XSFR IA content directly (address = 1060H / 1061H)

(2) Read Ib signal vale : Read XSFR IB content directly (address = 1058H / 1059H)

(3) Read iα signal vale : Read XSFR IALPHA content directly (address = 1062H / 1063H)

(4) Read Iβ signal vale : Read XSFR IBETA content directly (address = 105AH / 105BH)

(5) Read iq signal vale :
Step1 : Setting SFR_PAGE = 0(Address=ECH)
Step2 : Read the IQ_FB SFR value of PI_FB (address = AEH / AFH)

(6) Read Id signal vale :
Step1 : Setting SFR_PAGE = 1(Address=ECH)
Step2 : Read the ID_FB SFR value of PI_FB (address = AEH / AFH)

Please refer to the XSFR and SFR description in the Table25.2.2 and Table25.2.3.

XSFR	Description	Address	Characteristic
IA_L	ADC Output of phase A current (Ia) Low Byte	1060H	Read Only
IA_H	ADC Output of phase A current (Ia) High Byte	1061H	Read Only
IALPHA_L	α-axis current (iα) Low Byte of CLARKE Transform	1062H	Read Only
IALPHA_H	α-axis current (iα) High Byte of CLARKE Transform	1063H	Read Only
IB_L	ADC Output of phase B current (Ib) Low Byte	1058H	Read Only
IB_H	ADC Output of phase B current (Ib) High Byte	1059H	Read Only
IBETA_L	β-axis current (iβ) Low Byte of CLARKE Transform	105AH	Read Only
IBETA_H	β-axis current (iβ) High Byte of CLARKE Transform	105BH	Read Only

Table25.2.2 : Coordinate Transformation Block Signals observe

SFR	Description		Address
PI_FB_L	PI Control Feedback Data Register Low byte		AEH
PI_FB_H	PI Control Feedback Data Register High byte		AFH
	Parameters	Description	
SFR_PAGE = 0	IQ_FB	q-axis current (Iq) of PARK Transform	Read-Only
SFR_PAGE = 1	ID_FB	d-axis current (Id) of PARK Transform	Read-Only

Table25.2.3 : Id and Iq Signals observe

25.3 Current Control Loop Block

25.3.1 Current Control Loop Block Architecture

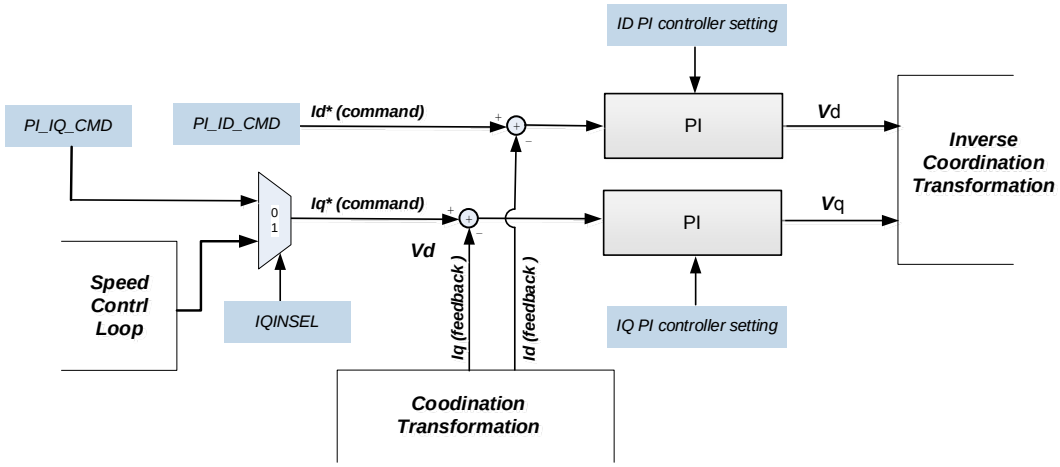


Figure 25.3.1 : Current Control Loop Block Architecture

25.3.2 Current Control Loop Block description and setting

Current Control Loop Block consists of a Multiplexer and two PI controllers. The Multiplexer can set the Iq command (Iq*) source from SFR or Speed Control Loop Block which input, The two PI controllers are d-axis current loop PI controller and q-axis current loop PI controller, whose function is to set and adjust Id and Iq current response. PI controller related functions and architecture will be explained in detail in Chapter 25.8 PI controller.

Id command (Id*) is set by ID_PI_CMD SFR. For the related reading process, first set the SFR_PAGE = 1 (SFR Address = ECH) where the signal is located, and then set the PI_CMD (SFR Address = A6/A7 H) position. access and set the Id command (Id*).

Iq command (Iq*) can be selected by IQINSEL SFR register (see Table25.3.1) for Toque or Speed control. When IQINSEL = 0, it is Motor toque control, which is set by IQ_PI_CMD SFR. For the related reading process, first set the SFR_PAGE = 0 (SFR Address = ECH) where the signal is located, and then set the PI_CMD (see Table25.3.2) location to access and set the Iq command of Toque control of Iq command (Iq*). When IQINSEL = 1 is Motor Speed control, Iq command (Iq*) is connected from Speed Control Loop Block output to iq PI control current loop input to perform speed control.

Please refer to the SFR description in the table below.

MOTOR_CONT1 Motor Control Register 1			Address = BFH			Reset Value = 0x00H		
Bit	SD MODE	MPWM DUSEL	MPWM EN	IQINSE 4	FOC ANGSEL	USER_PI _ACT	GEN_LPF_ ACT	SPFB FILTER
Type				R/W				

IQINSEL [4]	IQ current loop PI Control input select 0 : From IQ_PI_CMD SFR (SFR_PAGE = 0 & SFR Address = A6/A7 H) 1 : From Speed control loop output (speed PI- control output)

Table25.3.1 : IQINSEL SFR

SFR	Description		Address
PI_CMD_L	PI-Control Command Data Low byte		A6H
PI_CMD_H	PI-Control Command Data High byte		A7H
	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_PI_CMD	Command value of IQ PI Control	0x0000H
SFR_PAGE = 1	ID_PI_CMD	Command value of ID PI Control	0x0000H

Table25.3.2 : Current Loop IQ / ID PI control Command

25.4 Inverse Coordinate Transformation Block

25.4.1 Inverse Coordinate Transformation Block Architecture

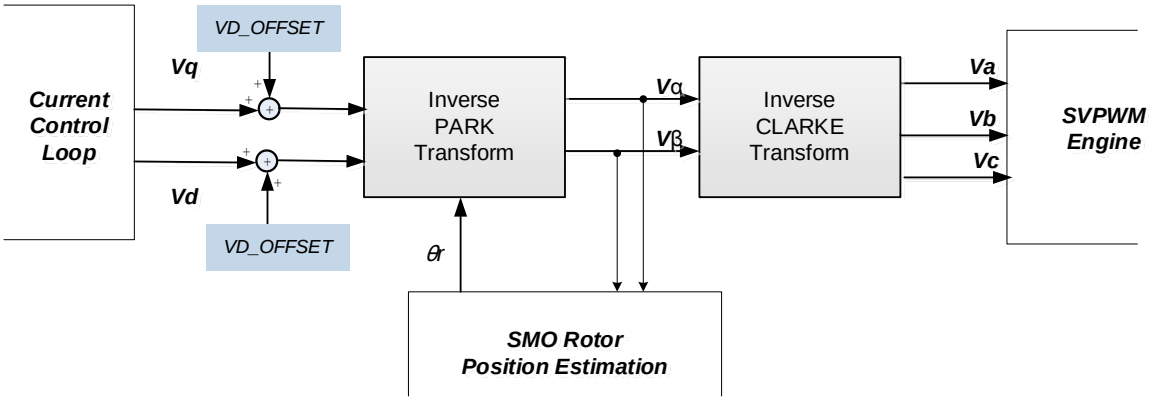


Figure 17.4.1 : Inverse Coordinate Transformation Block Architecture

25.4.2 Inverse PARK Transform

After the PI iteration, you have two voltage component vectors in the rotating d-q axis. You will need to go through complementary inverse transforms to get back to the 3-phase motor voltage. First, you transform from the two-axis rotating d-q frame to the two-axis stationary frame α-β. This transformation uses the Inverse Park Transform, as illustrated in Figure 25.4.2.

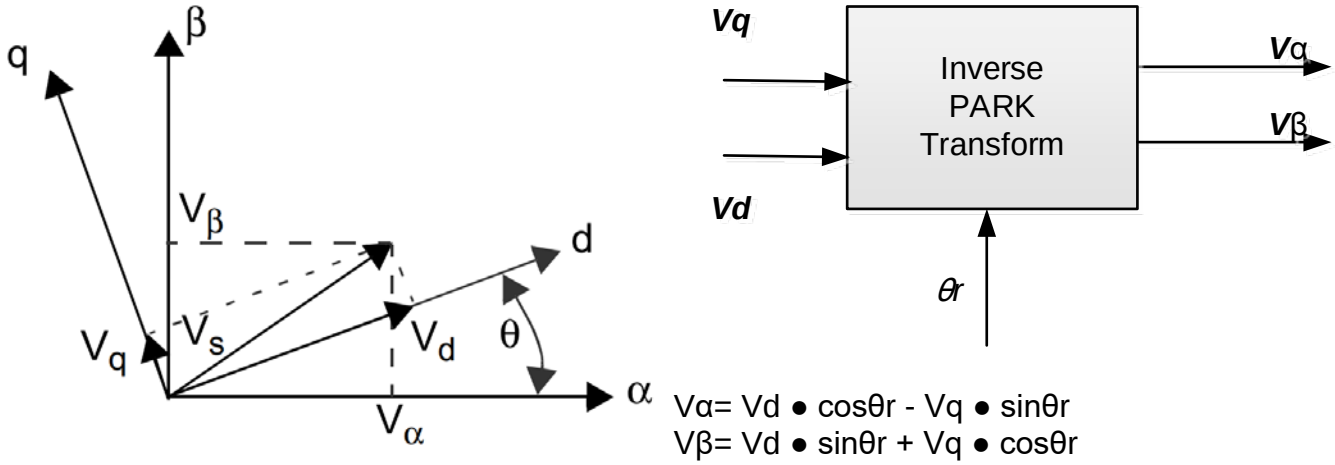


Figure 25.4.2 : Inverse Park Transform

25.4.3 Inverse CLARKE Transform

The next step is to transform from the stationary two-axis α-β frame to the stationary three-axis, 3-phase reference frame of the stator. Mathematically, this transformation is accomplished with the Inverse CLARKE Transform, as illustrated in Figure 25.4.3.

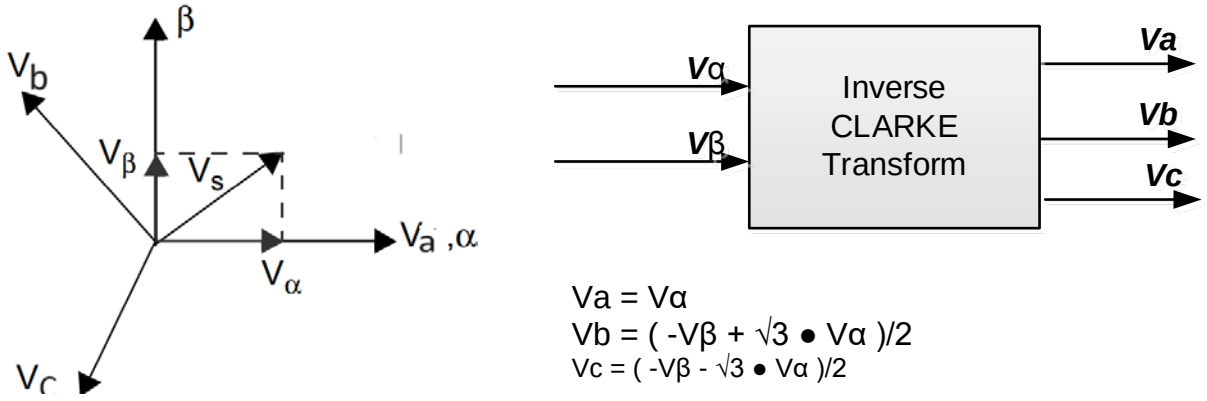


Figure 25.4.3 : Inverse Cleake Transform

25.4.4 Inverse Coordinate Transformation Block description and setting

In Inverse Coordinate Transformation Block, the Va, Vb Vα, Vβ value can be observed through XSFR VA, VB, VALPHA, and VBETA; Vq and Vd voltage value can be observed through the page0 and page1 of PI_OUT SFR.

For the related reading process in page of SFR, first set the SFR_PAGE (SFR Address = ECH) where the signal is located, and then read the corresponding SFR position data to read the signal value.

The reading process is as follows:

(1) Read Vq signal value :
Step1 : Setting SFR_PAGE = 0(Address=ECH)
Step2 : Read the IQ_PI_OUT SFR value of PI_OUT (address = ACH / ADH)

(2) Read Vd signal value :
Step1 : Setting SFR_PAGE = 1(Address=ECH)
Step2 : Read the ID_PI_OUT SFR value of PI_OUT (address = ACH / ADH)

(3) Read Vα signal value : Read XSFR VALPHA content directly (address = 1064H / 1065H)

(4) Read Vβ signal value : Read XSFR VBETA content directly (address = 105CH / 105DH)

(5) Read Va signal value : Read XSFR VA content directly (address = 1066H / 1067H)

(6) Read Vb signal value : Read XSFR VB content directly (address = 105EH / 105FH)

Please refer to the XSFR and SFR description in the Table25.4.1 and Table25.4.2.

XSFR	Description	Address	Characteristic
VALPHA_L(Read)	α-axis Stator Voltage Data Register Low Byte	1064H	Read Only
VALPHA_H(Read)	α-axis Stator Voltage Data Register High Byte	1065H	Read Only
VA_L(Read)	Phase A Drive Voltage Data Register Low Byte	1066H	Read Only
VA_H(Read)	Phase A Drive Voltage Data Register High Byte	1067H	Read Only
VBETA_L(Read)	β-axis Stator Voltage Data Register Low Byte	105CH	Read Only
VBETA_H(Read)	β-axis Stator Voltage Data Register High Byte	105DH	Read Only
VB_L(Read)	Phase B Drive Voltage Data Register Low Byte	105EH	Read Only
VB_H(Read)	Phase B Drive Voltage Data Register High Byte	105FH	Read Only

Table25.4.1 : Inverse Coordinate Transformation Block Signals observe

SFR	Description		Address
PI_OUT_L	PI Control Output Data Register Low byte		ACH
PI_OUT_H	PI Control Output Data Register High byte		ADH
	Parameters	Description	
SFR_PAGE = 0	IQ_PI_OUT	PI Control output voltage of IQ current Loop (Vq)	Read-Only
SFR_PAGE = 1	ID_PI_OUT	PI Control output voltage of ID current Loop (Vd)	Read-Only

Table25.4.2 : Vd and Vq Signals observe

In the Current Loop Block output signal Vd and Vq, the compensation amount (VD_OFFSET and VQ_OFFSET) can be added respectively according to the system requirements. For the related reading process, first set the SFR_PAGE (SFR Address = ECH) where the signal is located, and then set the FOC_D (SFR Address = D4/D5 H) position to access and perform compensation and control. Please refer to the SFR description in the following table, see Table25.4.3.

SFR	Description		Address
FOC_D_L	Field Oriented Control Data Low byte		D4H
FOC_D_H	Field Oriented Control Data High byte		D5H
	Parameters	Description	Reset Value
SFR_PAGE = 0	VD_OFFSET	d-axis voltage offset	0x0000H
SFR_PAGE = 1	VQ_OFFSET	q-axis voltage offset	0x0000H
SFR_PAGE = 3	AS	Angle Supplement Data	0x0000H
SFR_PAGE = 4	CPU_ANG	CPU Angle Data Register	0x0000H
SFR_PAGE = 5	FOC_ANG	Park and Inverse Park transformation angle input ※ **Read: Theta for FOC angle.** **Write: Theta offset.**	0x0000H
SFR_PAGE = 6	SVPWM_Amp	Amplitude value of SVPWM transformation	0x4000H

Table25.4.3 : FOC_D SFR

25.5 SMO Rotor Position Estimation Block

25.5.1 SMO Rotor Position Estimation Block Architecture

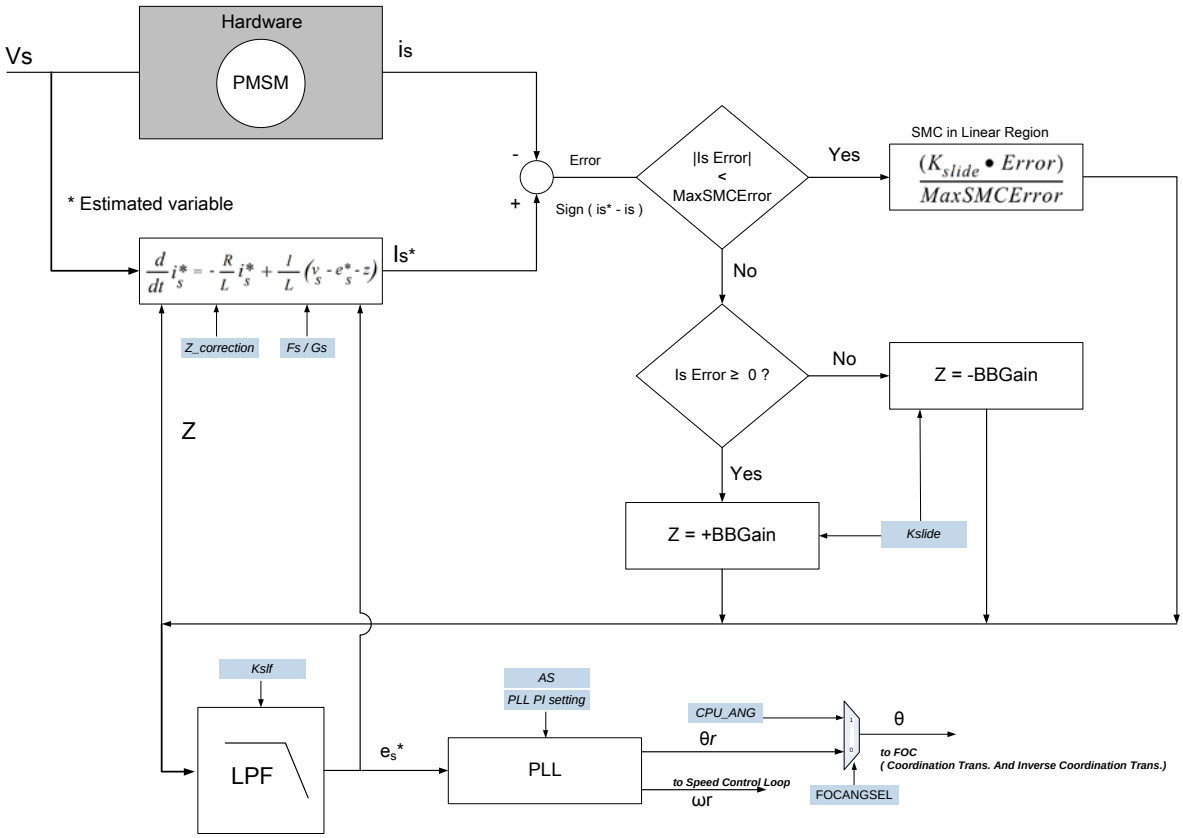


Figure 25.5.1 : SMO Rotor Position Estimation Block Architecture

25.5.2 SENSORLESS FOC For PMSM

An important part of the algorithm is how to calculate the commutation angle needed for FOC. This section of the application note explains the process of estimating commutation angle (θr) and motor speed (ωr).

The sensor-less control technique implements the FOC algorithm by estimating the position of the motor without using position sensors.

Motor position and speed are estimated based on measured currents and calculated voltages.

25.5.3 Motor Model

You can estimate the PMSM position by using a model of a DC Motor, which can be represented by winding resistance, winding inductance and BEMF, as shown in Figure 25.5.2.

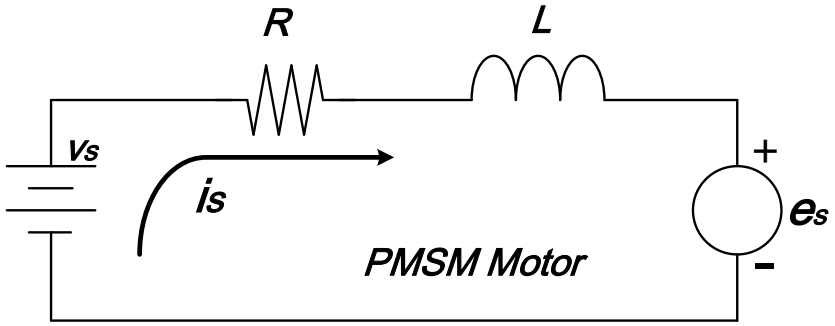


Figure 25.5.2 : Motor Model

From the motor model, the input voltage can be obtained:

$$v_s = Ri_s + L\frac{d}{dt}i_s + e_s$$

Where:

i_s = Motor Current Vector
v_s = Input Voltage Vector
e_s = Back-EMF Vector
R = Winding Resistance
L = Winding Inductance
T_s = Control Period

Motor current is obtained by solving for i_s:

$$\frac{d}{dt}i_s = \left(-\frac{R}{L}\right)i_s + \frac{1}{L}(v_s - e_s)$$

In the digital domain, this equation becomes:

$$\frac{i_s(n+1) - i_s(n)}{T_s} = \left(-\frac{R}{L}\right)i_s(n) + \frac{1}{L}(v_s(n) - e_s(n))$$

Solving for i_s:

$$i_s(n+1) = \left(1 - Ts \bullet \frac{R}{L}\right)i_s(n) + \frac{T_s}{L}(v_s(n) - e_s(n))$$

$$F = 1 - Ts \bullet \frac{R}{L}$$

$$G = \frac{T_S}{L}$$

F and G parameter can be setting via Fs and Gs SRF (see Table25.5.1 and see Table25.5.2).

SFR	Description		Address
SMO_D1_L	Field Oriented Control Data1 Low byte		BBH
SMO_D1_H	Field Oriented Control Data1 High byte		BCH
	Parameters	Description	
SFR_PAGE = 0	G_S	Parameters GS for motor model	0x7FFFH
SFR_PAGE = 1	Kslide	Parameters Slide mode Gain for angle estimation	0x3E80H
SFR_PAGE = 2	SMO_ANGBAS	Parameters Angle Base for angle estimation	0x0B2FH
SFR_PAGE = 3	Z_correction	SMO Zgain correction: $Z_{gain(new)} = \frac{Z_{gain} \times Z\_correction}{32767}$	0x7FFFH
SFR_PAGE = 4	SMO_ANG	Estimated value of SMO angle	Read-Only
SFR_PAGE = 5	BBGain	Parameters Slide mode Bang Bang Control Gain for angle estimation	0x7FFFH

Table25.5.1 : SMO_D1 SRF

SFR	Description		Address
SMO_D2_L	Field Oriented Control Data1 Low byte		BDH
SMO_D2_H	Field Oriented Control Data1 High byte		BEH
	Parameters	Description	
SFR_PAGE = 0	F_S	Parameters FS for motor model.	0x7FFFH

SFR_PAGE = 1	Kslf	Parameters LPF for BEMF e_s estimation	0x0064H
SFR_PAGE = 3	MaxSMCError	MAXSMC Error for SMC in Linear Region	0x7FFFH
SFR_PAGE = 4	EST-SP	ω* (Output value of PLL-PI-Control)	Read-Only

Table25.5.2 : SMO_D2 SRF

25.5.3 Current Observer

The position and speed estimator is based on a current observer. This observer is a digitized model of the motor. Variables and constants include:

(1) Motor Phase Current (i_s)
(2) Input voltage (v_s)
(3) BEMF (e_s)
(4) Winding resistance (R)
(5) Winding inductance (L)
(6) Control period (T_s)
(7) Output Correction Factor Voltage (z)

The digitized model provides a software representation of the hardware. However, in order to match the measured current and estimated current, the digitized motor model needs to be corrected using the closed loop, as shown in Figure 25.5.1.

Considering two motor representations, one in hardware (shaded area) and one in software, with the same input (v_s) fed into both systems, and matching the measured current (i_s) with estimated current (i_s*)from the model, we can presume that BEMF (e_s*) from our digitized model is the same as the BEMF (e_s) from the motor.

The sliding-mode controller has a limit value MaxSMCError. When the error value is lesser than the MaxSMCError, the output of sliding-mode controller works in the linear range as given by the equation beneath the PMSM block in Figure 25.5.1. K_{slide} parameter can be setting via K_{slide} SRF (see Table25.5.1).

For an error value outside of the linear range, the output of the sliding mode controller is (+BBGain)/(-BBGain) depending on the sign of the error.

BBGain parameter can be setting via BBGain SRF (see Table25.5.1).

A slide-mode controller, or SMC, is used to compensate the digitized motor model. A SMC consists of a summation point that calculates the sign of the error between measured current from the motor and estimated current from the digitized motor model. The computed sign of the error (+1 or -1) is multiplied by a SMC gain (K). The output of the SMC controller is the correction factor (Z). This gain is added to the voltage term from the digitized model, and the process repeats every control cycle until the error between measured current (i_s) and estimated current (i_s*) is zero (i.e., until the measured current and estimated current match).

25.5.4 BEMF Estimation

After compensating the digitized model, you have a motor model with the same variable values for the input voltage (V_s) and for current (i_s*). Once the digitized model is compensated, the next step is to estimate BEMF (e_s*) by filtering the correction factor (Z), as shown in Figure 25.5.1. The BEMF estimation (e_s*) is fed back to the model to update the variable e_s* after every control cycle. Values e_α and e_β (vector components of e_s) are used for the estimated Theta calculation.

The Cut-off freq. of LPF (BEMF low pass filter) can be setting via Kslf SRF (see Table25.5.2).

25.5.5 Rotor Position Calculation

PLL takes e_α* and e_β* as input signals and uses PI Control to perform θr calculations. PI controller operating output is ω*. After converting the angle unit through SMO_ANGBAS (see Table25.5.1), ωr is obtained. Then an integral unit is designed to calculate θr and perform negative feedback operation. See Figure 25.5.3 for details. PI controller related functions and architecture will be explained in detail in Chapter 25.8 PI controller.

The SMO_ANGBAS unit conversion formula is as follows:

SMO_ANGBAS = 384 ● 64 / (PWM_freq.) ●(motor pole/2/60)

The calculation time of SMO Rotor Position Estimation block will cause time delay of θr. AS (page3 of FOC_D, Table25.4.3) can be used for angle compensation. Its compensation range is -128 ~ 127 degrade. CGF0X2AXX angle calculation is 384 degrade / 360°, see Figure 25.5.4.

The θ of FOC (see Figure 25.5.1), is designed with a Multiplexer that select signal is FOCANGSEL (MOTOR_CONT1[3], SFR Address = BFH). When FOCANGSEL= 1, FOC_ANG = CPU_ANG (page4 of FOC_D, Table25.4.3), if FOCANGSEL = 0, FOC_ANG = θr to FOC (see Figure 25.5.1).

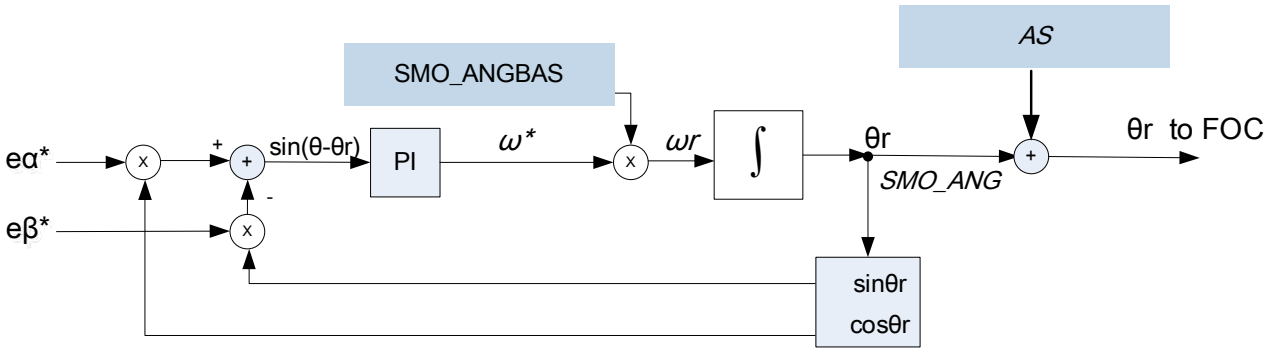


Figure 25.5.3 : PLL Architecture

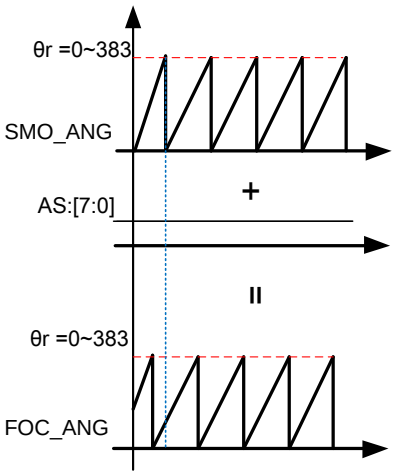


Figure 25.5.4 : Angle Supplement

25.5.6 SMO Rotor Position Estimation Block description and setting

In Inverse Coordinate Transformation Block, the Va, Vb Vα, Vβ value can be observed through XSFR VA, VB, VALPHA, and VBETA; Vq and Vd voltage value can be observed through the page0 and page1 of PI_OUT SFR.

In the SMO Rotor Position Estimation Block, the Zα, Zβ, iα*, iβ*, eα* and eβ* signal status can be observed through XSFR ZALPHA, ZBETA, ES_IALPHA, ES_IBETA, ES_EALPHA, and ES_EBETA. Please refer to the XSFR description in the Table25.5.3. The θr and ω* can be observed through the two SFR addresses SMO_D1 and SMO_D2.

For the related reading process, first set the SFR_PAGE (SFR Address = ECH) where the signal is located, and then read the corresponding SMO_D1 or SMO_D2 position data to read the signal value.

XSFR	Description	Address	Characteristic
ZALPHA_L	The Z gain of EEMF Eα Data Register Low Byte	1070H	Read Only
ZALPHA_H	The Z gain of EEMF Eα Data Register High Byte	1071H	Read Only
ZBETA_L	The Z gain of EEMF Eβ Data Register Low Byte	1068H	Read Only
ZBETA_H	The Z gain of EEMF Eβ Data Register High Byte	1069H	Read Only
ES_IALPHA_L	Estimate Current Iα Data Register Low Byte	1072H	Read Only
ES_IALPHA_H	Estimate Current Iα Data Register High Byte	1073H	Read Only
ES_EALPHA_L	Estimate EEMF Eα Data Register Low Byte	1074H	Read Only
ES_EALPHA_H	Estimate EEMF Eα Data Register High Byte	1075H	Read Only
ES_IBETA_L	Estimate Current Iβ Data Register Low Byte	106AH	Read Only
ES_IBETA_H	Estimate Current Iβ Data Register High Byte	106BH	Read Only
ES_EBETA_L	Estimate EEMF Eβ Data Register Low Byte	106CH	Read Only
ES_EBETA_H	Estimate EEMF Eβ Data Register High Byte	106DH	Read Only

Table25.5.3 : BEMF Estimation Signals observe

The reading process is as follows :

(1) Read Z_α signal value : Read XSFR ZALPHA content directly (address = 1070H / 1071H)

(2) Read Z_β signal value : Read XSFR ZBETA content directly (address = 1068H / 1069H)

(3) Read i_α* signal value : Read XSFR ES_IALPHA content directly (address = 1072H / 1073H)

(4) Read i_β* signal value : Read XSFR ES_IBETA content directly (address = 106AH / 106BH)

(5) Read e_α* signal value : Read XSFR ES_EALPHA content directly (address = 1074H / 1075H)

(6) Read e_β* signal value : Read XSFR ES_EBETA content directly (address = 106CH / 106DH)

(7) Read θr signal value :
Step1 : Setting SFR_PAGE = 4(Address=ECH)
Step2 : Read the SMO_ANG SFR value of SMO_D1 (address = BCH / BBH)

(8) Read ω* signal value :
Step1 : Setting SFR_PAGE = 4(Address=ECH)
Step2 : Read the EST-SP SFR value of SMO_D2 (address = BEH / BDH)

25.6 SVPWM Engine Block

25.6.1 SVPWM Engine Block Architecture

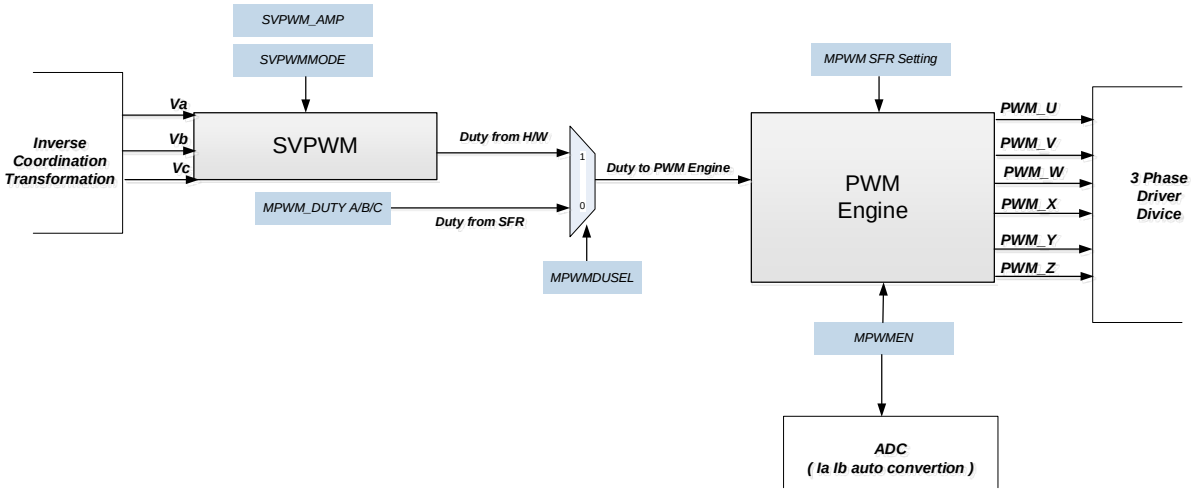


Figure 25.6.1 : SVPWM Engine Block Architecture

25.6.2 SVPWM Engine Block Architecture

The final step in the vector control process is to generate pulse-width modulation signals for the 3-phase motor voltage signals. If you use Space Vector Pulse Width Modulation (SVPWM) techniques, the process of generating the pulse width for each of the three phases is reduced to a few simple equations. In this implementation, the Inverse Clarke Transform has been folded into the SVPWM routine, which further simplifies the calculations.

Each of the three inverter outputs can be in one of two states. The inverter output can be connected to either the plus (+) bus rail or the minus (-) bus rail, which allows for $2^3 = 8$ possible states of the output as shown in Table 25.6.1.

The two states in which all three outputs are connected to either the plus (+) bus or the minus (-) bus 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 SVM star. The remaining six states are represented as vectors with 60 degree rotation between each state, as shown in Figure 25.6.2.

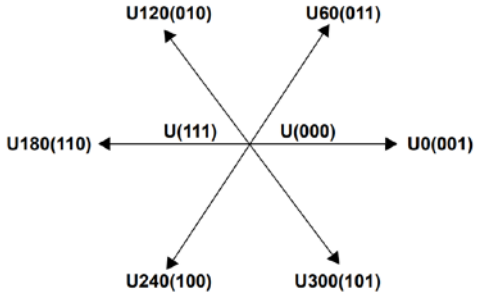


Figure 25.6.2 : Space Vector Pulse Width Modulation (SVPWM)

The process of SVM allows the representation of any resultant vector by the sum of the components of the two adjacent vectors. In Figure 25.6.3, U_{OUT} is the desired resultant. It lies in the sector between U60 and U0. If during a given PWM period T, U0 is output for T1/T and U60 is output for T2/T, the average for the period will be U_{OUT}.

T0 = Null Vector

T = T1 + T2 + T0 = PWM Period

UOUT = (T1/T • U0) + (T2/T • U60)

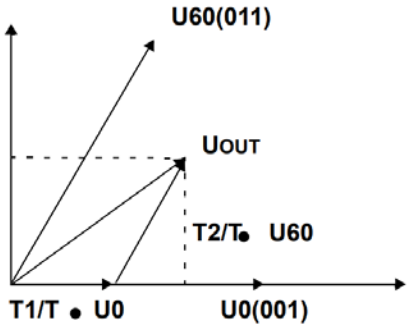


Figure 25.6.3 : AVERAGE SVPWM

T0 represents a time where no effective voltage is applied into the windings; that is, where a null vector is applied. The values for T1 and T2 can be extracted with no extra calculations by using a modified Inverse Clark transformation. If you reverse Vα and Vβ, a reference axis is generated that is shifted by 30 degrees from the SVM star. As a result, for each of the six segments, one axis is exactly opposite that segment and the other two axes symmetrically bound the segment. The values of the vector components along those two bounding axis are equal to T1 and T2.

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	U_0
0	1	1	0	V_{DC}	$-V_{DC}$	$V_{DC}/3$	$V_{DC}/3$	U_{60}
0	1	0	$-V_{DC}$	V_{DC}	0	$-V_{DC}/3$	$V_{DC}/3$	U_{120}
1	1	0	$-V_{DC}$	0	V_{DC}	$-2V_{DC}/3$	0	U_{180}
1	0	0	0	$-V_{DC}$	V_{DC}	$-V_{DC}/3$	$-V_{DC}/3$	U_{240}
1	0	1	V_{DC}	$-V_{DC}$	0	$V_{DC}/3$	$-V_{DC}/3$	U_{300}
1	1	1	0	0	0	0	0	U(111)

Table25.6.1 : SPACE VECTOR PULSE WIDTH MODULATION INVERTER STATES

25.6.3 7-Segment SVPWM

MDE is set to 7-Segment SVPWM by default. The control signal to select SVPWM mode is SVPWMMODE (see Table17.6.2). When SVPWMMODE = 0, MDE is set to 7-Segment SVPWM mode. 7-Segment SVPWM output voltage, as shown in Figure 25.6.4.

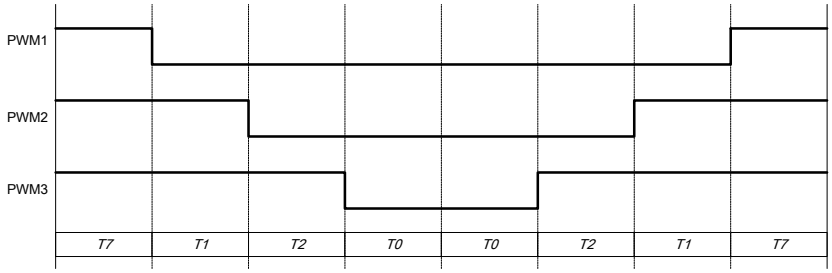


Figure25.6.4 : 7-segment SVPWM output voltage

MOTOR_CONT2		Address = 9FH			Reset Value = 0x00H			
Motor Control Register 2								
	----	----	----	SVPWM MODE	SVPWM PS	----	IAE SOFEN	----
Bit				4				
Type				R/W				

SVPWMMODE [4]	SVPWM mode select: 0 : 7-Segment SVPWM mode. 1 : 5-Segment SVPWM mode.

Table25.6.2 : MOTOR_CONT2. SVPWMMODE SFR Description

25.6.4 5-Segment SVPWM

When SVPWMMODE = 1, MDE is set to 5-Segment SVPWM mode. 5-Segment SVPWM output voltage, as shown in Figure 25.6.5.

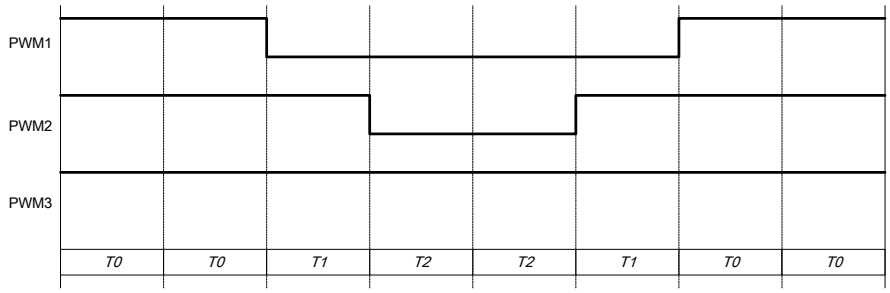


Figure25.6.5 : 5-segment SVPWM output voltage

25.6.5 SVPWM Over Modulation

MDE SVPWM Engine can use the Over Modulation function. You can set the SFR SVPWM_Amp to amplify the Va / Vb / Vc amplitude to achieve Over Modulation. The SVPWM_Amp setting process is as follows:

Step1 : Setting SFR_PAGE = 6(Address=ECH)
Step2 : Access SVPWM_Amp of FOC_D value

25.6.6 Motor PWM Output Setting

Motor PWM Engine is based MPWMEN (see Table25.6.3, MOTOR_CONT1[5]) to active PWM counter, when MPWMEN = 1, Motor PWM output (U, V, W, X, Y, Z) and motor current Ia and Ib of ADC auto conversion is active.

MPWMDUSEL (see Table25.6.3, MOTOR_CONT1[6]) can select the Duty input source of Motor PWM. When MPWMDUSEL = 0, Motor PWM determines PWM Duty by PWM_DUTY_A / PWM_DUTY_B / PWM_DUTY_C (see Table17.6.4), and outputs PWM signal.

MPWMDUSEL = 1, the calculated PWM Duty value of Va, Vb and Vc is directly input to the SVPWM Engine for PWM modulation.

MOTOR_CONT1 Address = BFH Reset Value = 0x00H
Motor Control Register 1

Bit	----	MPWMDUSEL	MPWMEN	IQINSEL	FOCANGSEL	USER_PI_ACT	GEN_LPF_ACT	SPFB FILTER
		6	5					
Type		R/W	R/W					

MPWMDUSEL [6]	MPWM duty select : 0 : From PWM_DUTY_A /B /C (MPWMDAT, SFR Address = E1H/E2H) 1 : From SVPWM H/W
MPWMEN [5]	MPWM timer run control enable: 0 : Disable 1 : Enable MPWMEN =1, ADC CH0 & CH1 Auto Converter:

Table25.6.3 : MOTOR_CONT1 SFR Description

Period of Motor PWM is based on MPWM_CYC (see Table25.6.4) setting value. The base frequency of Motor PWM counter is 48MHz. The counter counts up to MPWM_CYC and then counts down to 0 (see Figure25.6.6). Therefore, the value set by MPWM_CYC determines the frequency of Motor PWM. For example : MPWM_CYC = 1250, Period of Motor PWM = 1250 x 2 x 1/(48MHz) = 52.08us, Freq. of Motor PWM is 19.2KHz.

SFR	Description		Address
MPWMDATL	MPWM Data Low Byte(SYNC)		E1H
MPWMDATH	MPWM Data High Byte(SYNC)		E2H
	Parameters	Description	Reset Value
SFR_PAGE = 0	MPWM_CYC	Motor PWM cycle value	0x04B0H
SFR_PAGE = 1	PWM_DUTY_A	Motor PWM Duty value for Phase A	0x0258H
SFR_PAGE = 2	PWM_DUTY_B	Motor PWM Duty value for Phase B	0x0258H
SFR_PAGE = 3	PWM_DUTY_C	Motor PWM Duty value for Phase C	0x0258H

Table25.6.4 : MPWMDAT SFR Description

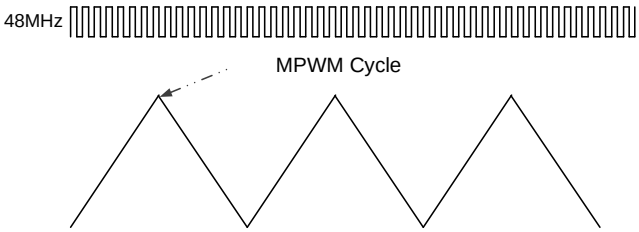


Figure25.6.6 : Motor PWM is count up and down timer

Motor PWM output (U, V, W, X, Y, Z) has 4 type mode to be setting. High side PWM (U、V、W) can be setting mode by HIGHSIDE (see Table25.6.4) , Low side PWM (X、Y、Z) can be setting mode by LOWSIDE (see Table25.6.5), the 4 type mode is “Force Low”, “Force High”, “Force Low”, “Active High” and “Active Low”.

In order to connect with the 3 phase Gate-driver and optimize PCB Layout considerations, the Motor PWM output (U, V, W, X, Y, Z) has a Swap function, which can be performed via HALFSWAP, ALLSWAP, UWXZSWAP (see Table25.6.5) A combination of Swap, as shown in Figure25.6.7.

MPWMCONT(SYNC) Address = E3H Reset Value = 0x00H
MPWM Control Register

Bit	-----	HALF SWAP	ALL SWAP	UWXZ SWAP	LOWSIDE		HIGHSIDE	
	7	6	5	4	3	2	1	0
Type	X	R/W	R/W	R/W	R/W	R/W	R/W	R/W

HALFSWAP [6]	Half pin swap (X,V swap),(Y,W swap) and (X,W swap) 0: Normal 1: Half change
ALLSWAP [5]	High side and Low side pin swap(U->X , X->U、V->Y,Y->V、W->X,X->W) 0: Normal 1: Inverse
UWXZSWAP [4]	U、W pin swap and X、Z pin swap 0: Normal 1: Swap
LOWSIDE [2:3]	Low-side (X、Y、Z) 00: Force Low 10: Active High 01: Force High 11: Active Low
HIGHSIDE [1:0]	High-side (U、V、W) 00: Force Low 10: Active High 01: Force High 11: Active Low

Table25.6.5 : MPWMCONT SFR Description

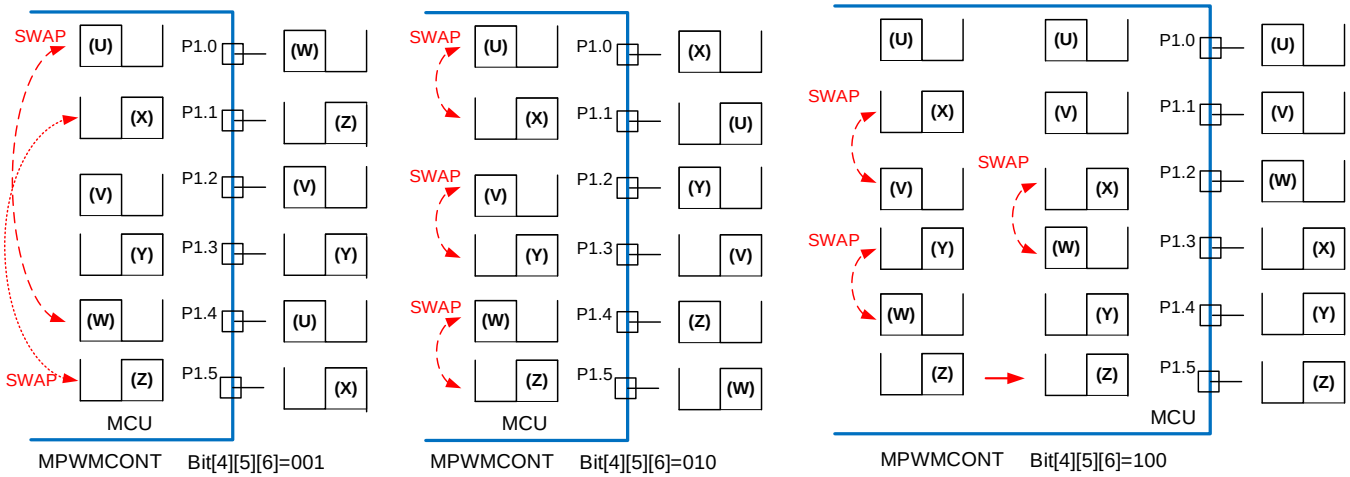


Figure25.6.7 : PWM Pin swap Unit Architecture

MPWMDT(see Table25.6.6) Compensation PWM output with Dead-Time is use to prevent short-though between high-side and low-side power device. Dead-time setting example : if MPWMDT = 100, Dead Time = 50 x (1 / 48MHz) = 2.08us.

MPWMDT (SYNC) Address = E5H Reset Value = 0x78H
Motor PWM Dead-Time Register

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

Table25.6.6 : MPWMDT SFR Description

Via MPWMINV (see Table25.6.7) setting , Motor U,V,W X,Y,Z PWM output support Inverse function that is to inverse the PWM output signal at the last stage output.

MPWMINV (SYNC) Address = E4H Reset Value = 0x00H
MPWM Inverse Selection Register

Bit	-----	-----	ZINV	WINV	YINV	VINV	XINV	UINV
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

ZINV [5]	Low-side PWM Z output inverse select : 0 : Non-inverse 1 : Inverse
WINV [4]	High-side PWM W output inverse select : 0 : Non-inverse 1 : Inverse
YINV [3]	Low-side PWM Y output inverse select : 0 : Non-inverse 1 : Inverse
VINV [2]	High-side PWM V output inverse select : 0 : Non-inverse 1 : Inverse
XINV [1]	Low-side PWM X output inverse select : 0 : Non-inverse 1 : Inverse
UINV [0]	High-side PWM U output inverse select : 0 : Non-inverse 1 : Inverse

Table25.6.7 : MPWMINV SFR Description

Figure25.6.8 illustrates the waveform output by Motor PWM when HIGHSIDE, LOWSIDE, MPWMINV and MPWMDT are set at the same time.

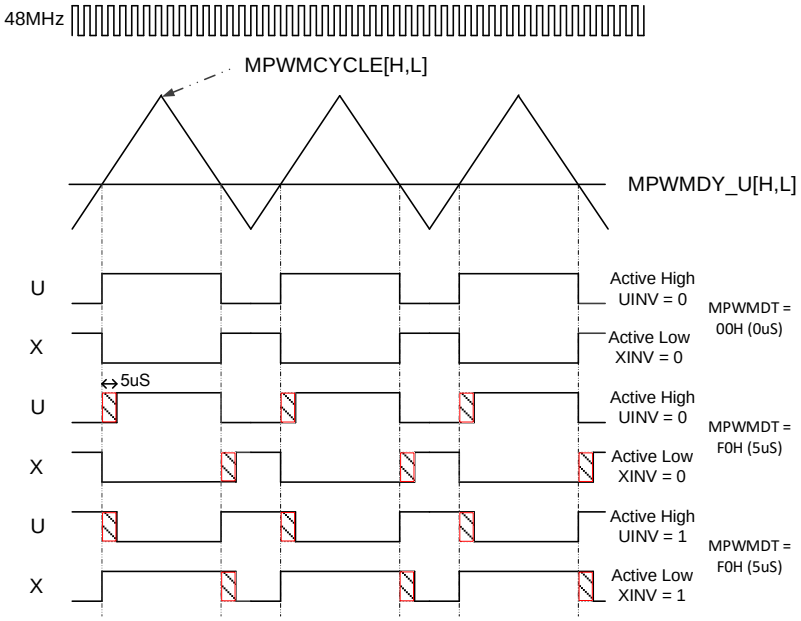


Figure25.6.8 PWM Output with Dead Time and PWM mode

MDE provides motor reverse setting and simply set SVPWMPS (see inverse Table25.6.7) to reverse the motor.

MOTOR_CONT2 Motor Control Register 2	Address = 9FH		Reset Value = 0x00H		
----	SVPWMMODE	SVPWMPS	----	IAE SOFEN	----
Bit		3			
Type		R/W			

SVPWMPS [3]	SVPWM output phase sequence: 0 : Positive sequence (A、B、C) 1 : Negative sequence (B、A、C)

Table25.6.7 : MOTOR_CONT2. SVPWMPS SFR Description

25.7 Speed Control Loop Block

25.7.1 Speed Control Loop Block Architecture

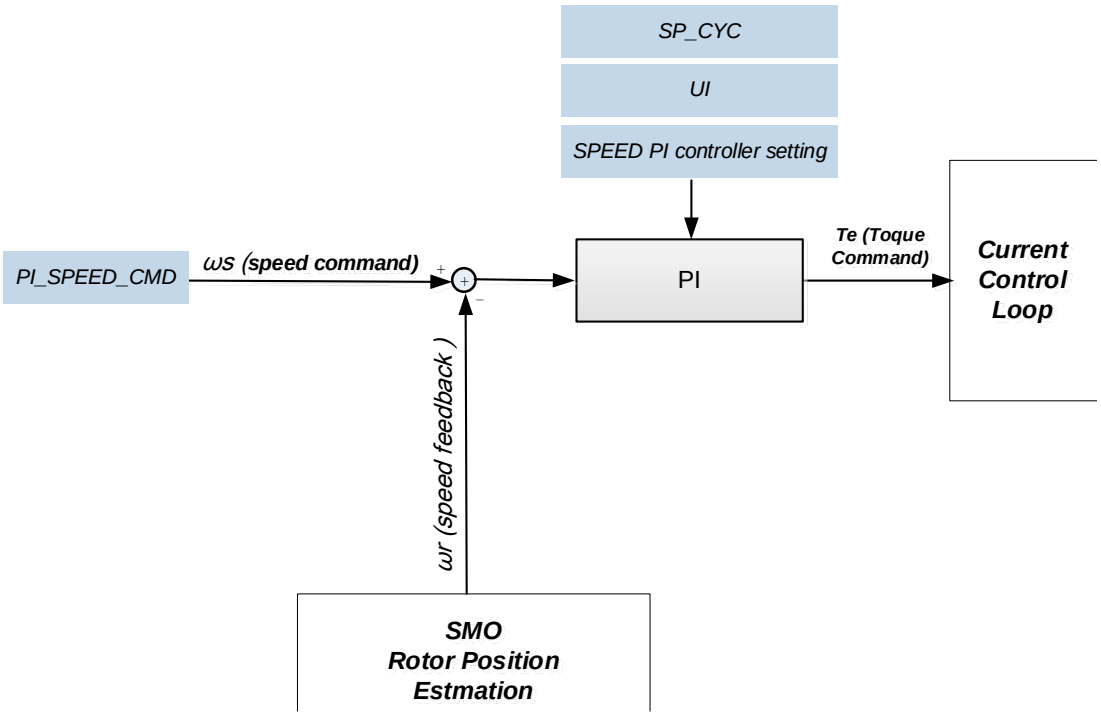


Figure 25.7.1 : Speed Control Loop Block Architecture

25.7.2 Speed Control Loop Block description and setting

In the Speed Control Loop Block, the speed command *ωs* and the estimated speed signal *ωr* are used to calculate the Toque control quantity *Te* through the speed control law. *Te* is also the current command of the Current Control Loop Block. The speed control law is controlled using PI Control. PI controller related functions and architecture will be explained in detail in Chapter 25.8 PI controller.

PI controller of Speed Control Loop Block, contains an enable signal SPEEDEN (see Table25.7.1), when SPEEDEN = 1, the PI controller will be active the motor speed control loop.

FOCCONT Address = D6H Reset Value = 0x00H
Field Oriented Control Register

Bit	PI CLEAR	ESTCR	INV ADCD	ADCTRIG	PLLEN	SPEEDEN
						0
Type						R/W

SPEEDEN	SPEED-Control enable:
[0]	0: Disable
	1: Enable

Table25.7.1 : FOCCONT. SPEEDEN SFR Description

The Speed Control Loop Block contains a speed loop sampling frequency adjustment SRF SP_CYC (see Table25.7.2), and the filled value of SP_CYC[7:0] is the following formula:

SP_CYC = PWM Freq. / speed loop sampling Freq.

For example : PWM Freq. is 20KHz, and the designed speed loop sampling frequency is 1KHz, then SP_CYC = 20KHz / 1KHz = 20

SP_CYC SPEED Loop control cycle			Address = EDH			Reset Value = 0x26H		
Bit	SP-CYC[7:0]							
	7	6	5	4	3	2	1	0
Type	W	W	W	W	W	W	W	W

Table25.7.2 : SP_CYC SFR Description

25.8 PI Controller

25.8.1 PI Controller Block Architecture

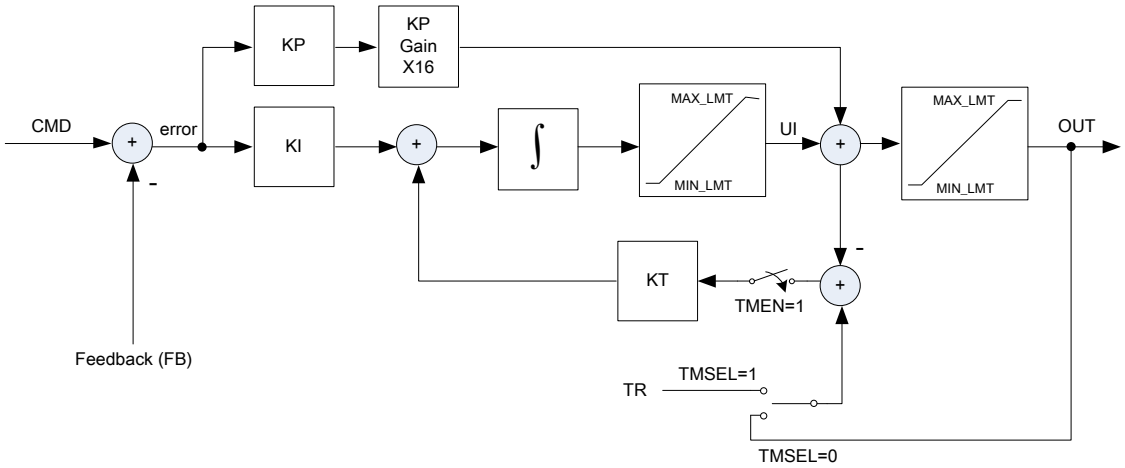


Figure 25.8.1 : PI Block Architecture

25.8.2 PI controller Block description and setting

MDE has four PI controllers: IQ, ID, PLL and SPEED Loop. Their architectures are the same (see Figure 25.8.1: PI Block Architecture). If you write 1 to PICLEAR (see Table25.8.1), you can reset IQ , ID, PLL and SPEED Loop 4 PI controllers to initial status. In addition, PLL and SPEED Loop can be enabled by PLLEN and SPEEDEN (see Table25.8.1).

FOCCONT Address = D6H Reset Value = 0x00H

Field Oriented Control Register

Bit	PICLEAR	ESTCR	INVADCD	ADCTRIG	PLLEN	SPEEDEN
	7				1	0
Type	R/W				R/W	R/W

PICLEAR	Clear register value of PI-Control:
[7]	0: Disable
	1: Enable
PLLEN	PLL-Control enable:
[1]	0: Disable
	1: Enable
SPEEDEN	SPEED-Control enable:
[0]	0: Disable
	1: Enable

Table25.8.1 : PICLEAR/ PLLEN/ SPEEDEN SFR Description

The PI controller has a total of 12 control parameters: CMD, KP, KI, MIN_LMT, MAX_LMT, KT, TR, FB, UI, OUT, GAIN, and TMSR (see Table 25.8.2). Except for GAIN and TMSR, which do not have pagination, all other parameters have four pages each for four PIs to set parameters. Among them, KP, KI, MIN_LMT, MAX_LMT, and KT can only be written, and the others can be read and written. The setting method is selected by SFR_PAGE. SFR_PAGE=0 selects IQ PI; SFR_PAGE=1 selects ID PI; SFR_PAGE=2 selects PLL PI; SFR_PAGE=3 selects SPEED PI, the setting process is:

CMD (PI command) : Access PI_CMD (SFR Address 0xA7H and 0xA6H)
UI (integral value) : Access PI_UI (SFR Address 0x9EH and 0x9DH)
OUT (output value) : Access PI_OUT (SFR Address 0xADH and 0xACH)
FB (feedback value) : Access PI_FB (SFR Address 0xAFH and 0xAEH)
KP (PI KP Gain) : Write PI_KP (SFR Address 0x94H and 0x93H)
KI (PI KI Gain) : Write PI_KI (SFR Address 0x92H and 0x91H)
MAX_LMT (PI output Maximum Limit) : Write PI_MAX_LMT(SFR Address 0x96H and 0x95H)
MIN_LMT (PI output Minimum Limit) : Write PI_MIN_LMT(SFR Address 0xB3H and 0xB2H)
KT (PI KT Gain) : Write PI_KT (SFR Address 0xDFH and 0xDEH)

SFR	Description	Address	Reset value	R/W
PI_GAIN	PI-control KPx16 Gain Select Register	E6H	F7H	R/W
PI_TMSR	PI-control Tracking Mode Select Register	D7H	00H	R/W
PI_KI_L	PI-control KI Parameter Low byte	91H	00H	Write Only
PI_KI_H	PI-control KI Parameter High byte	92H	00H	Write Only
PI_KP_L	PI-control KP Parameter Low byte	93H	00H	Write Only
PI_KP_H	PI-control KP Parameter High byte	94H	00H	Write Only
PI_MAX_LMT_L	PI-control Maximum Limit Data Low byte	95H	FFH	Write Only
PI_MAX_LMT_H	PI-control Maximum Limit Data High byte	96H	7FH	Write Only
PI_MIN_LMT_L	PI-control Minimal Limit Data Low byte	B2H	01H	Write Only
PI_MIN_LMT_H	PI-control Minimal Limit Data High byte	B3H	80H	Write Only
PI_CMD_L	PI-control Command Data Low byte	A6H	00H	R/W
PI_CMD_H	PI-control Command Data High byte	A7H	00H	R/W
PI_UI_L	PI-control Integral Data Low byte	9DH	00H	R/W
PI_UI_H	PI-control Integral Data High byte	9EH	00H	R/W
PI_OUT_L	PI-control Output Data Low byte	ACH	00H	R/W
PI_OUT_H	PI-control Output Data High byte	ADH	00H	R/W
PI_FB_L	PI-control Feedback Data Low byte	AEH	00H	R/W
PI_FB_H	PI-control Feedback Data High byte	AFH	00H	R/W
PI_TR_L	PI-control TR Parameter Low byte	AAH	00H	R/W
PI_TR_H	PI-control TR Parameter High byte	ABH	00H	R/W
PI_KT_L	PI-control KT Parameter Low byte	DEH	00H	Write Only
PI_KT_H	PI-control KT Parameter High byte	DFH	00H	Write Only

Table 25.8.2 : PI SFR List

All four PI controllers can amplify the error in proportional term by 16 times, and then connect them in series to KP. The amplification enable SFR setting is placed in PI_GAIN (see Table25.8.3), and the PLL and SPEED UI (see Table 25.8.1) Before enable, the initial value of the integral can be set through PI_UI (SFR Address = 0x9EH / 0x9DH).

PI_GAIN Address = E6H Reset Value = 0xF7H
PI KPx16 Gain Select Register

	IQKPGEN	---	IDKPGEN	---	SPKPGEN	---	PLLKPEN	---
Bit	7	6	5	4	3	2	1	0
Type	R/W	---	R/W	---	R/W	---	R/W	---

IQKPGEN [7]	IQ KP Gain enable: 0: Disable 1: Enable
IDKPGEN [5]	ID KP Gain enable: 0: Disable 1: Enable
SPKPGEN [3]	SPEED KP Gain enable: 0: Disable

	1: Enable
PLLKPGEN [1]	PLL KP Gain enable: 0: Disable 1: Enable

Table25.8.3 : PI_GAIN SFR Description

In addition, bumpless control is added to the PI architecture, and the control switch can be set by PI_TMSR SFR. The content of PI_TMSR SFR is shown in Table 25.8.4.

PI_TMSR Address = D7H Reset Value = 0x00H

PI Tracking Mode Select Register

	IQ_TM SEL	IQ_TM EN	ID_TM SEL	ID_TM EN	SP_TM SEL	SP_TM EN	PLL_TM SEL	PLL_TM EN
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IQ_TMSEL [7]	IQ-Tracking Mode Select: 0: Auto Mode 1: Manual Mode
IQ_TMEN [6]	IQ-Tracking Mode Enable: 0: Disable 1: Enable
ID_TMSEL [5]	ID-Tracking Mode Select: 0: Auto Mode 1: Manual Mode
ID_TMEN [4]	ID-Tracking Mode Enable: 0: Disable 1: Enable
SP_TMSEL [3]	SPEED-Tracking Mode Select: 0: Auto Mode 1: Manual Mode
SP_TMEN [2]	SPEED -Tracking Mode Enable: 0: Disable 1: Enable
PLL_TMSEL [1]	PLL-Tracking Mode Select: 0: Auto Mode 1: Manual Mode
PLL_TMEN [0]	PLL-Tracking Mode Enable: 0: Disable 1: Enable

Table25.8.4 : PI_TMSR SFR Description

25.8.3 User PI controller Block description and setting

MDE provides an independent PI for users to use, and its architecture is shown in Figure 25.8.2. Its related parameters are CMD, KP, KI, MAX_LMT, MIN_LMT, UI, OUT, and FB, which are placed in corresponding Page 4 of PI SFR. Therefore, when using it, you must set SFR_PAGE=4 before selecting USER PI.

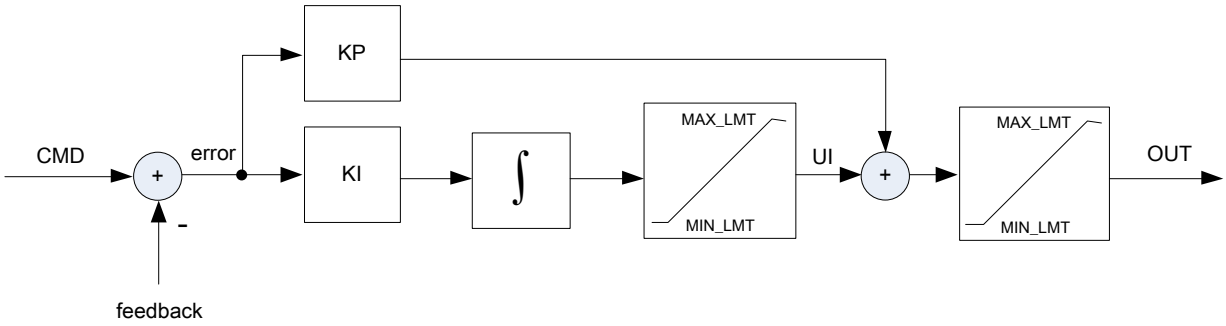


Figure 25.8.2 : User PI Block Architecture

25.9 IPD (Initial Position Detection)

Initial Position Detection (IPD) is the method to detect the motor position before it spins up. It uses the characteristic of magnetic field varying when driving with current in motor winding inductance to determine the rotor initial position. We setting the AOCP short level and apply PWM pattern across two of the three phases: VW, WV, UV, VU, WU, and UW sequentially and the PWM pattern will be stopped when the AOCP short level is reached. The minimum time that reaches the AOCP short level among the 4 or 6 PWM patterns is the rotor position before it spins up. See Figure 25.9.1.

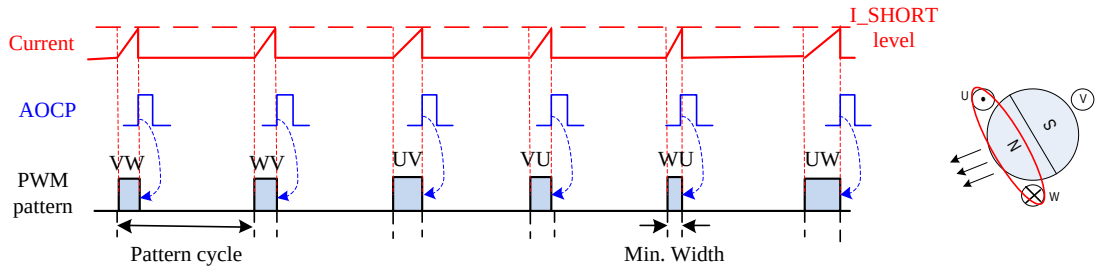


Figure 25.9.1 : IPD principle

25.9.1 IPD Control Register

IPD (initial position detection) can choose to use 6 patterns or 4 patterns (set by IPDPNS in Table 25.9.1 and see Figure 25.9.2) to search for the initial position of the rotor. There are 4 different settings for the sending interval between each pattern (set by IPDPCYC in Table25.9.1), after setting IPDPNS and IPDPCYC, write "1" in IPD_CTRL bit [0] (IPDSTR) and start IPD detection. When the IPD_CTRL bit [7] (IPDS) is "1", it means that the IPD is detecting. When IPDS=0, the initial position result of the detection will be stored in the IPD_CTRL bits [6:4] (IPDMIN).

Since IPD detection uses the speed of magnetic saturation (compared with the time speed of overcurrent) to determine the initial position of the rotor, if you find that the width of the printed pattern is the same during detection, you can try to adjust Table 25.9.1 The I_SHORT value enables the initial position of the rotor to be determined smoothly.

IPD_CTRL Address = A5H Reset Value = 0x18H

Initial Position Detection Control Register

Bit	7	6	5	4	3	2	1	0
	IPDS	IPDMIN			IPDPNS	IPDCYC		IPDSTR
Type	R	R	R	R	R/W	R/W	R/W	R/W

IPDS [7]	IPD State: 0:Normal 1:Busy	
IPDMIN [6:4]	IPD Minimum Pattern: 001: Pattern1 011: Pattern3 101: Pattern5	 010: Pattern2 100: Pattern4 110: Pattern6
IPDPNS [3]	IPD Pattern Number Select: 0: 4 pattern 1: 6 pattern	
IPDPCYC [2:1]	IPD Pattern Cycle: 00: 13ms (48MHz Count) 10: 54ms (12MHz Count)	 01: 27ms (24MHz Count) 11: 110ms (6MHz Count)
IPDSTR [0]	IPD Start: 1: Start IPD	

Table25.9.1 : IPD_CTRL SFR Description

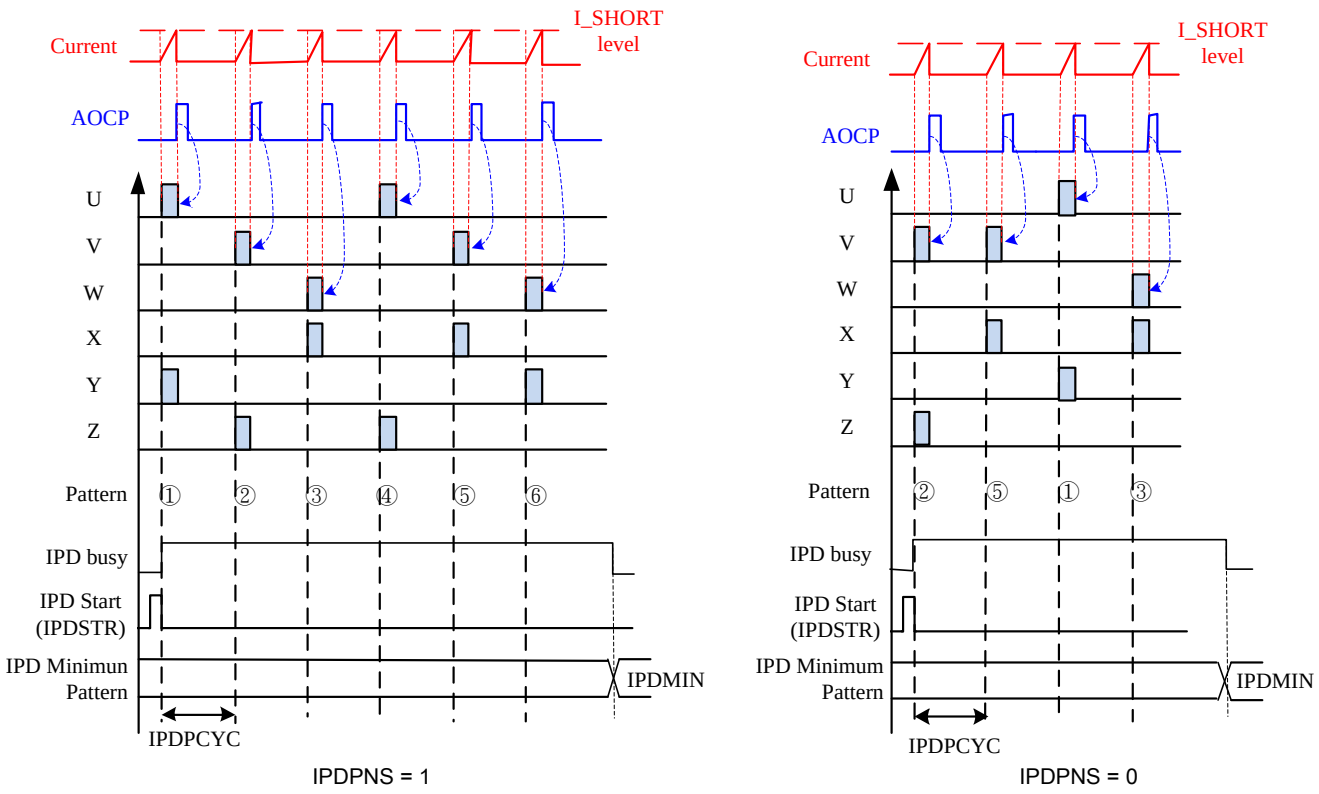


Figure 25.9.2 : IPD for rotor position detect before spin up in 6-pattern or 4-pattern

25.9.2 IPD Pattern Define Register

The PWM of which upper side PWM or lower side PWM should be turned on at the same time according to the IPD pattern can also be set by user.

For example, the initial value of IPDPAT21_DEF shown in the following table is 0xEB, which means pattern2 high/low side on define = 0xE：high side PWM_W on，low side PWM_Y on；pattern1 high/low side on define = 0xB：high side PWM_V on，low side PWM_Z on.

By analogy, the initial value of IPDPAT43_DEF is 0x96, which means that the high side PWM_V of pattern 4 is on and the low side PWM_X is on; the high side PWM_U of pattern 3 is on and the low side PWM_Y is on. The initial value of IPDPAT65_DEF is 0x7D, which means that the high side PWM_U on and low side PWM_Z of pattern 6 are on; the high side PWM_W on and low side PWM_X on of pattern 5 are on. (see Figure 25.9.3)

In order to avoid setting it to an invalid switch state, it is recommended that the user does not need to reset this register and just use the initial setting value.

SFR	Description			
IPD_PD	Initial Position Detection Pattern Define Register			
	Parameters		Description	Reset Value
SFR_PAGE = 0	IPDPAT21_DEF		IPD Pattern 2、1 Define	0xEBH
SFR_PAGE = 1	IPDPAT43_DEF		IPD Pattern 4、3 Define	0x96H
SFR_PAGE = 2	IPDPAT65_DEF		IPD Pattern 6、5 Define	0x7DH

IPD_PD	Address = A4H		Reset Value = 0x00H					
	IPD_PD [7:0]							
	7	6	5	4	3	2	1	0
Type	W	W	W	W	W	W	W	W

SFR_PAGE = 0	Pattern2 High/Low Side on Define		Pattern1 High/Low Side on Define	
SFR_PAGE = 1	Pattern4 High/Low Side on Define		Pattern3 High/Low Side on Define	
SFR_PAGE = 2	Pattern6 High/Low Side on Define		Pattern5 High/Low Side on Define	
	U、V、W Bit[7:6]	X、Y、Z Bit[5:4]	U、V、W Bit[3:2]	X、Y、Z Bit[1:0]
	00 : -- 01 : U 10 : V 11 : W	00 : -- 01 : X 10 : Y 11 : Z	00 : -- 01 : U 10 : V 11 : W	00 : -- 01 : X 10 : Y 11 : Z

Table25.9.2 : IPD_PD SFR Description

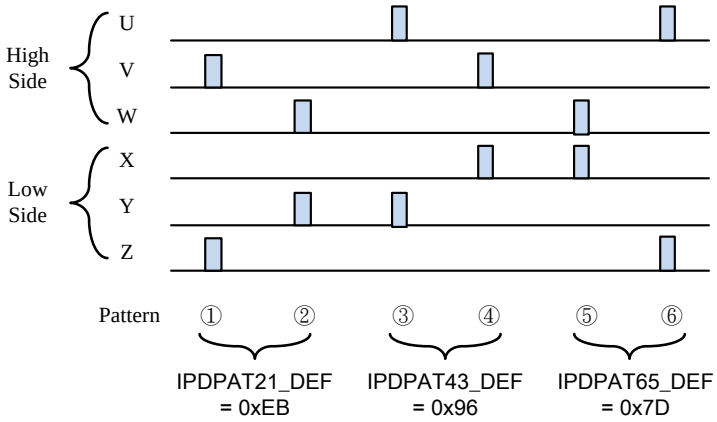


Figure 25.9.3 : IPD pattern high/low side on state initial setting

25.10 OCP (Over Current Protection)

25.10.1 OCP Block Architecture

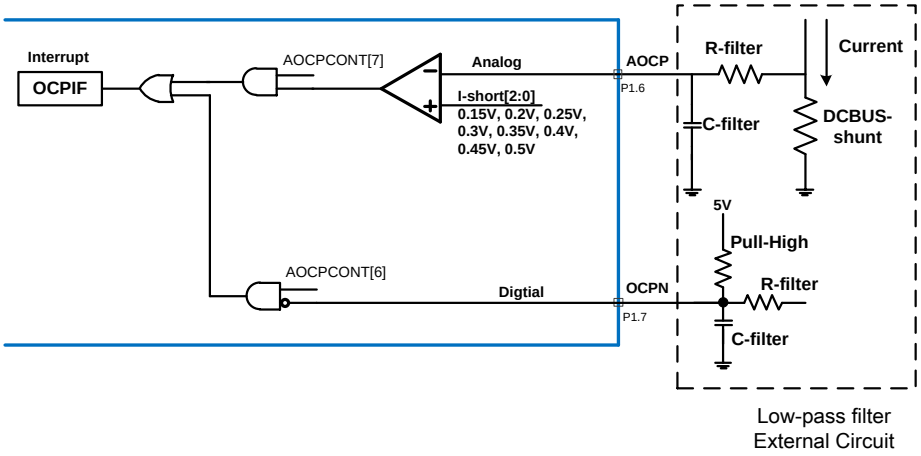


Figure 25.10.1 : Analog OCP and Digital OCPN Unit Architecture

25.10.2 OCP Block description and setting

CGF0X2AXX is embedded H/W Over Current Protection Circuit, supports Analog and Digital OCP two modes, when AOCPEN = 1 (see Table25.10.1), that is, set to Analog OCP, the built-in OCP comparator will be used to detect over current status, the voltage setting value of its comparator is I_ SHORT (see Table25.10.1), When the AOCP voltage value is greater than the I_ SHORT set voltage value, H/W will immediately force the PWM to 0 to protect the Power Device (MOS, IGBT..). DOCPEN = 1 (see Table25.10.1), turn on Digital OCP mode, Digital OCP is generally used when connected to the IPM module. The Fault PIN of the IPM is directly connected to the DOCP for OCP protection.

AOCPCONT Address = EEH Reset Value = 0xE7H

Analog OCP Control Register

Bit	DOCPNEN	AOCPEN	OPAPD	-----	-----	I_SHORT[3:0]		
Bit	7	6	5	4	3	2	1	0
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W

DOCPNEN [7]	Digital OCPN enable: 0 : Disable 1 : Enable
AOCPEN [6]	Analog OCP enable: 0 : Disable 1 : Enable
OPAPD [5]	OPA Power Down 0 : Normal 1 : OPA Power Down
I_ SHORT [2:0]	Analog OCP SHORT level select : (OCP interrupt :OCPIF) 000 : 0.15V 100 : 0.35V 001 : 0.2V 101 : 0.4V 010 : 0.25V 110 : 0.45V

011 : 0.3V	111 : 0.5V(default)

Table25.10.1 : AOCPCONT SFR Description

The MCU may read the OCPST (see Table25.10.2) to observe the over current status of the system. When OCPST =1, the system is in the over current state.

Analog OCP can set digital de-bounce time to avoid OCP malfunction caused by noise. By setting AOCPDBT (see Table25.10.2), Analog OCP 0~1.291uS de-bounce time can be set.

There are 2 modes to release OCP status, Auto Mode and User Mode. For its setting and action, please refer to Figure 25.10.2: OCP Short occur and PWM out.

DOCPCONT Address = EFH Reset Value = 0x05H

Digital OCP Control Register

Bit	OCPST	AOCPDBT[4:0]					OCPC	OCPMS
Bit	7	6	5	4	3	2	1	0
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

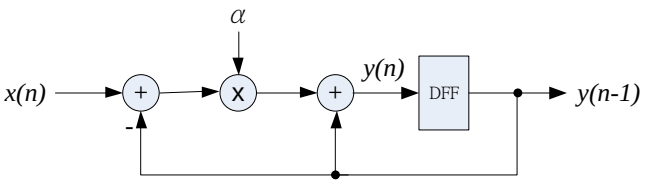
OCPST [7]	OCP Short status : 0 : No Over current Short. 1 : Over current Short occur. (hardware set OCPC = '0') Six PWM output is high-impedance.
AOCPDBT [6:2]	Analog OCP input de-bounce time (default 41.67nS) 0~31 = 0~1.291uS (48MHz/2 fixed)
OCPC [1]	OCP status clear bit : When OCP is occur, hard ware will set OCPC = '0'. In User mode, user can write '1' to clear OCP status, the PWM will output at next PWM cycle.
OCPMS [0]	OCP mode select : 0 : Auto mode. 1: User mode.

Table25.10.2 : DOCPCONT SFR Description

25.11 GEN_LPF (General Low-Pass Filter)

25.11.1 LPF Block Architecture

A simple first order digital low-pass filter is a filter that passes signals with a frequency lower than a selected cutoff frequency and attenuates signals with frequencies higher than the cutoff frequency (Figure 25.11.1).



$$\alpha = \frac{\omega_c T_s}{1 + \omega_c T_s}$$

ω_c : cutoff angular frequency of filter
T_s : sampling preiod

Figure 25.11.1 : first order digital low-pass filter Architecture

25.10.2 LPF Block description and setting

CGF0X2AXX built-in a simple first order digital low-pass filter, its architecture is shown in Figure 25.11.2, and the corresponding SFR is shown in Table 25.11.1. When using LPF, you must first fill in the parameters into the corresponding SFR and then issue an operation request (GEN_LPF_ACT, see Table 25.11.2) to the LPF. Then an LPF operation can be performed.

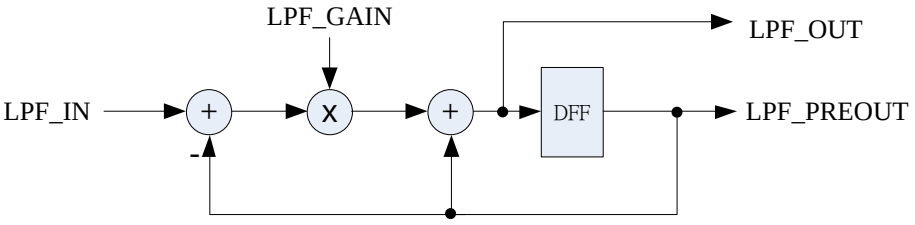


Figure 25.11.2 : LPF architecture and corresponding parameters

SFR	Description							
GEN_LPF	General Low Pass Filter Parameter							
	Parameters	Description						Reset Value
SFR_PAGE = 0	GEN_GAIN	General LPF Gain parameter						0x0000H
SFR_PAGE = 1	LPF_IN	General LPF input data						0x0000H
SFR_PAGE = 2	LPF_OUT	General LPF output data						0x0000H
SFR_PAGE = 3	LPF_PREOUT	General LPF last output data						0x0000H
GEN_LPF_L	Address = A1H	Reset Value = 0x00H						
General Low Pass Filter Parameter Low Byte								
Bit	GEN_LPF[7:0]							
Type	7	6	5	4	3	2	1	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
GEN_LPF_H	Address = A2H	Reset Value = 0x00H						
General Low Pass Filter Parameter High Byte								
Bit	GEN_LPF[15:8]							
Type	7	6	5	4	3	2	1	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table25.11.1 : LPF SFR Description

MOTOR_CONT1 Motor Control Register 1			Address = BFH			Reset Value = 0x00H		
Bit	-----	MPWM DUSEL	MPWM EN	IQIN SEL	FOC ANGSEL	USER_ PI_ACT	GEN_LPF_ACT	SPFB FILTER
Type							2	
							R/W	

GEN_LPF_ACT	General Low Pass Filter active 1 : LPF Active, Clear by Hardware

Table25.11.2 : GEN_LPF_ACT Control Bit

25.12 FG (Frequency Generator Control)

Frequency Generator (FG) is an output signal that indicates the rotation speed of rotor. The system can observe the FG signal to know the current motor speed.

25.12.1 FG Control

CGF0X2AXX has a built-in Frequency Generator H/W. To use FG, you must first enable FG. The number of pulses output during the electrical rotation period of a rotor can be set by FGCTRL (Figure 25.12.1). The output waveform of FG is as follows As shown in Figure 25.12.1.

FGCTRL Address = 1025H Reset Value = 0x00H

FG Control Register

Bit	7	6	5	4	3	2	1	0
	FG_EN	---	---	---	---	FGPULSE_SEL		
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W

FG_EN	FG output enable:
[7]	0 : Disable
	1 : Enable
FGPULSE_SEL	FG pulse number selection:
[2:0]	000 : 1 pulses/cycle
	001 : 2 pulses/cycle
	010 : 4 pulses/cycle
	011 : 8 pulses/cycle
	100 : 12 pulses/cycle

Table 25.12.1 :FG Control SFR

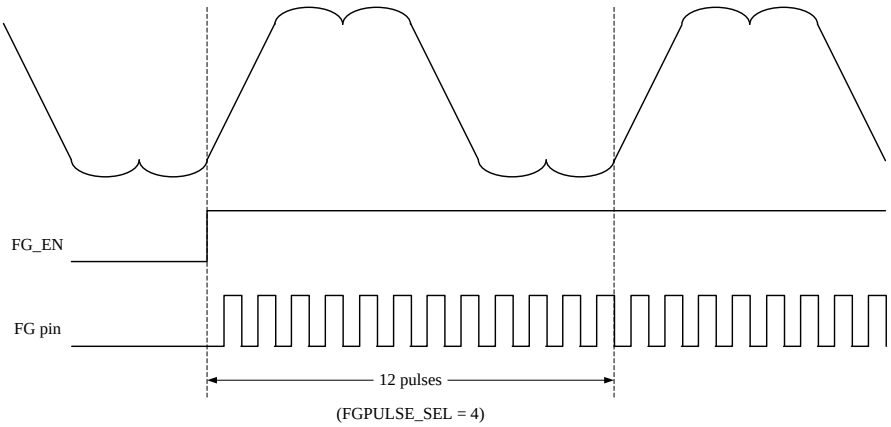


Figure 25.12.1 :FG Output Waveform

25.13 MDE SFR List

25.13.1 MOTOR_CONT1 SFR

MOTOR_CONT1 Address = BFH Reset Value = 0x00H

Motor Control Register 1

Bit	---	MPWMDUSEL	MPWMEN	IQINSEL	FOCANGSEL	USER_PI_ACT	GEN_LPF_ACT	SPFB
	7	6	5	4	3	2	1	0
Type	x	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MPWMDUSEL [6]	IQ current loop PI Control input select 0 : From IQ_PI_CMD SFR (SFR_PAGE = 0 & SFR Address = A6/A7 H) 1 : From Speed control loop output (speed PI- control output)
MPWMEN [5]	MPWM timer run control enable: 0 : Disable 1 : Enable MPWMEN =1, ADC CH1 & CH2 Auto Converter:
IQINSEL [4]	IQ current loop PI Control input select 0 : From IQ_PI_CMD SFR (SFR_PAGE = 0 & SFR Address = A6/A7 H) 1 : From Speed control loop output (speed PI- control output)
FOCANGSEL [3]	FOC input angle select: 0 : From CPU_ANG (Write value of CPU_ANG_H & CPU_ANG_L) 1 : From SMO_ANG (Estimated angle) Inverse park trans formation input select of angle
USER_PI_ACT [2]	User PI active 0 : disable user PI 1 : active user PI
GEN_LPF_ACT [1]	General Low Pass Filter active 1 : LPF Active, Clear by Hardware
SPFB_FILETER [0]	Speed PI Feedback Filter enable: 0 : Disable Filter 1 : Enable Filter

25.13.2 MOTOR_CONT2 SRF

MOTOR_CONT2 Address = 9FH Reset Value = 0xA4H

Motor Control Register 2

Bit	DUTY_SUPRS_EN	DUTY_SUPRS		SVPWMMODE	SVPWMPS	----	IAESOFEN	----
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	X	R/W	X

DUTY_SUPRS_EN [7]	Duty suppression enable: 0 : Duty suppression disable 1 : Duty suppression enable.
DUTY_SUPRS [6:5]	Duty suppression select: (when DUTY_SUPRS_EN = 1) 00 : No Minimum Duty Limit. 01 : Full amplitude suppression

	11 : Lowest duty limit in 2xDeadband.
SVPWMMODE [4]	SVPWM mode select: 0 : 7-Segment SVPWM mode. 1 : 5-Segment SVPWM mode.
SVPWMPS [3]	SVPWM output phase sequence: 0 : Positive sequence (A、B、C) 1 : Negative sequence (B、A、C)
IAESOFEN [1]	Initial Angle estimation soft start enable: 0 : Disable 1 : Enable

25.13.3 Field Oriented Control SFR

FOCCONT Address = D6H Reset Value = 0x00H

Field Oriented Control Register

Bit	PI CLEAR	ESTCR			INV ADCD	ADC TRIG	PLL EN	SPEED EN
	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PICLEAR [7]	Clear register value of PI-Control: 0: Disable 1: Enable
ESTCR [6:4]	Iα and Iβ current ratio (to SMO Rotor Position Estimation block) 000 : Current ValueX1 001 : Current ValueX2 010 : Current ValueX4 011 : Current ValueX8 100 : Current Value/2 101 : Current Value/4 110 : Current Value/8 111 : Current Value/16
INVADCD [3]	Inverse ADC register data: 0: Disable 1: Enable
ADCTRIG [2]	ADC Trigger (Phase A and Phase B): 0 : PWM counter max 1 : PWM counter min
PLLEN [1]	PLL PI Control enable: 0: Disable 1: Enable
SPEEDEN [0]	SPEED PI Control enable: 0: Disable 1: Enable

25.13.4 PI_GAIN SFR (PI- Controller GAIN)

PI_GAIN Address = E6H Reset Value = 0xF7H

PI KPx16 Gain Select Register

	IQKPGEN	---	IDKPGEN	---	SPKPGEN	---	PLLKPEN	---
Bit	7	6	5	4	3	2	1	0
Type	R/W	---	R/W	---	R/W	---	R/W	---

IQKPGEN [7]	IQ KP Gain enable: 0: Disable 1: Enable
IDKPGEN [5]	ID KP Gain enable: 0: Disable 1: Enable
SPKPGEN [3]	SPEED KP Gain enable: 0: Disable 1: Enable
PLLKPGEN [1]	PLL KP Gain enable: 0: Disable 1: Enable

25.13.5 PI_TMSR SFR (PI Tracking Mode Select Register)

PI_TMSR Address = D7H Reset Value = 0x00H

PI Tracking Mode Select Register

	IQ_TMSEL	IQ_TMEN	ID_TMSEL	ID_TMEN	SP_TMSEL	SP_TMEN	PLL_TMSEL	PLL_TMEN
Bit	7	6	5	4	3	2	1	0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IQ_TMSEL [7]	IQ-Tracking Mode Select: 0: Auto Mode 1: Manual Mode
IQ_TMEN [6]	IQ-Tracking Mode Enable: 0: Disable 1: Enable
ID_TMSEL [5]	ID-Tracking Mode Select: 0: Auto Mode 1: Manual Mode
ID_TMEN [4]	ID-Tracking Mode Enable: 0: Disable 1: Enable
SP_TMSEL [3]	SPEED-Tracking Mode Select: 0: Auto Mode 1: Manual Mode
SP_TMEN [2]	SPEED -Tracking Mode Enable: 0: Disable 1: Enable
PLL_TMSEL [1]	PLL-Tracking Mode Select: 0: Auto Mode 1: Manual Mode
PLL_TMEN [0]	PLL-Tracking Mode Enable: 0: Disable 1: Enable

25.13.6 SP_CYC SFR

SP_CYC Address = EDH Reset Value = 0x26H

SPEED Loop control cycle

	SP-CYC[7:0]							
Bit	7	6	5	4	3	2	1	0
Type	W	W	W	W	W	W	W	W

25.13.7 PI-Control Data SFR

SFR	Description		Address
PI_KI_L	PI-Control KI Parameter Low byte		91H
PI_KI_H	PI-Control KI Parameter High byte		92H
SFR_PAGE	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_KI	Parameters KI for IQ - Control	0x0000H
SFR_PAGE = 1	ID_KI	Parameters KI for ID - Control	0x0000H
SFR_PAGE = 2	SPEED_KI	Parameters KI for SPEED - Control	0x0000H
SFR_PAGE = 3	PLL_KI	Parameters KI for PLL - Control	0x09C4H
SFR_PAGE = 4	USER_KI	Parameters KI for USER - Control	0x0000H

SFR	Description		Address
PI_KP_L	PI-Control KP Parameter Low byte		93H
PI_KP_H	PI-Control KP Parameter High byte		94H
SFR_PAGE	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_KP	Parameters KP for IQ - Control	0x0000H
SFR_PAGE = 1	ID_KP	Parameters KP for ID - Control	0x0000H
SFR_PAGE = 2	SPEED_KP	Parameters KP for SPEED - Control	0x0000H
SFR_PAGE = 3	PLL_KP	Parameters KP for PLL - Control	0x36B0H
SFR_PAGE = 4	USER_KP	Parameters KP for USER - Control	0x0000H

SFR	Description		Address
PI_MAX_LMT_L	PI-Control Maximum Limit Data Low byte		95H
PI_MAX_LMT_H	PI-Control Maximum Limit Data High byte		96H
	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_MAX	Maximum Limit value of IQ - Control	0x7FFFH
SFR_PAGE = 1	ID_MAX	Maximum Limit value of ID - Control	0x7FFFH
SFR_PAGE = 2	SPEED_MAX	Maximum Limit value of SPEED - Control	0x7FFFH
SFR_PAGE = 3	PLL_MAX	Maximum Limit value of PLL - Control	0x7FFFH
SFR_PAGE = 4	USER_MAX	Maximum Limit value of USER - Control	0x0000H

SFR	Description		Address
PI_MIN_LMT_L	PI-Control Minimum Limit Data Low byte		B2H
PI_MIN_LMT_H	PI-Control Minimum Limit Data High byte		B3H
	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_MIN	Minimum Limit value of IQ - Control	0x8001H
SFR_PAGE = 1	ID_MIN	Minimum Limit value of ID - Control	0x8001H
SFR_PAGE = 2	SPEED_MIN	Minimum Limit value of SPEED - Control	0x8001H
SFR_PAGE = 3	PLL_MIN	Minimum Limit value of PLL - Control	0x8001H
SFR_PAGE = 4	USER_MIN	Minimum Limit value of USER - Control	0x0000H

SFR	Description		Address
PI_CMD_L	PI-Control Command Data Low byte		A6H
PI_CMD_H	PI-Control Command Data High byte		A7H
	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_CMD	Command value of IQ - Control	0x0000H
SFR_PAGE = 1	ID_CMD	Command value of ID - Control	0x0000H
SFR_PAGE = 2	SPEED_CMD	Command value of SPEED - Control	0x0000H
SFR_PAGE = 3	PLL_CMD	Command value of PLL - Control	0x0000H
SFR_PAGE = 4	USER_CMD	Command value of USER - Control	0x0000H

SFR	Description		Address
PI_UI_L	PI-Control Integral Data Low byte		9DH
PI_UI_H	PI-Control Integral Data High byte		9EH
	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_UI	Integral value of IQ - Control	0x0000H
SFR_PAGE = 1	ID_UI	Integral value of ID - Control	0x0000H
SFR_PAGE = 2	SPEED_UI	Integral value of SPEED - Control	0x0000H
SFR_PAGE = 3	PLL_UI	Integral value of PLL - Control	0x0000H
SFR_PAGE = 4	USER_UI	Integral value of USER - Control	0x0000H

SFR	Description		Address
PI_OUT_L	PI-Control Output Data Low byte		ACH
PI_OUT_H	PI-Control Output Data High byte		ADH
	Parameters	Description	Reset Value

SFR_PAGE = 0	IQ_OUT	Output Data of IQ - Control	0x0000H
SFR_PAGE = 1	ID_OUT	Output Data of ID - Control	0x0000H
SFR_PAGE = 2	SPEED_OUT	Output Data of SPEED - Control	0x0000H
SFR_PAGE = 3	PLL_OUT	Output Data of PLL - Control	0x0000H
SFR_PAGE = 4	USER_OUT	Output Data of USER - Control	0x0000H

SFR	Description		Address
PI_FB_L	PI-Control Feedback Data Low byte		AEH
PI_FB_H	PI-Control Feedback Data High byte		AFH
	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_FB	Feedback Data of IQ - Control	0x0000H
SFR_PAGE = 1	ID_FB	Feedback Data of ID - Control	0x0000H
SFR_PAGE = 2	SPEED_FB	Feedback Data of SPEED - Control	0x0000H
SFR_PAGE = 3	PLL_FB	Feedback Data of PLL - Control	0x0000H
SFR_PAGE = 4	USER_FB	Feedback Data of USER - Control	0x0000H

SFR	Description		Address
PI_KT_L	PI-Control KT Parameter Low byte		DEH
PI_KT_H	PI-Control KT Parameter High byte		DFH
SFR_PAGE	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_KT	Parameters KT for IQ - Control	0x0000H
SFR_PAGE = 1	ID_KT	Parameters KT for ID - Control	0x0000H
SFR_PAGE = 2	SPEED_KT	Parameters KT for SPEED - Control	0x0000H
SFR_PAGE = 3	PLL_KT	Parameters KT for PLL - Control	0x0000H

SFR	Description		Address
PI_TR_L	PI-Control TR Parameter Low byte		AAH
PI_TR_H	PI-Control TR Parameter High byte		ABH
SFR_PAGE	Parameters	Description	Reset Value
SFR_PAGE = 0	IQ_TR	Parameters TR for IQ - Control	0x0000H
SFR_PAGE = 1	ID_TR	Parameters TR for ID - Control	0x0000H
SFR_PAGE = 2	SPEED_TR	Parameters TR for SPEED - Control	0x0000H
SFR_PAGE = 3	PLL_TR	Parameters TR for PLL - Control	0x0000H

25.13.8 Sliding Mode Observer Data SFR

SFR	Description		Address
SMO_D1_L	Field Oriented Control Data1 Low byte		BBH
SMO_D1_H	Field Oriented Control Data1 High byte		BCH
	Parameters	Description	
SFR_PAGE = 0	G_S	Parameters GS for motor model	0x7FFFH
SFR_PAGE = 1	Kslide	Parameters Slide mode Gain for angle estimation	0x3E80H
SFR_PAGE = 2	SMO_ANGBAS	Parameters Angle Base for angle estimation	0x0B2FH
SFR_PAGE = 3	Z_correction	SMO Zgain correction: $Z_{gain(new)} = \frac{Z_{gain} \times Z\_correction}{32767}$	0x7FFFH
SFR_PAGE = 4	SMO_ANG	Estimated value of SMO angle	Read-Only
SFR_PAGE = 5	BBGain	Parameters Slide mode Bang Bang Control Gain for angle estimation	0x7FFFH

SFR	Description		Address
SMO_D2_L	Field Oriented Control Data1 Low byte		BDH
SMO_D2_H	Field Oriented Control Data1 High byte		BEH
	Parameters	Description	
SFR_PAGE = 0	F_S	Parameters FS for motor model.	0x7FFFH
SFR_PAGE = 1	Kslf	Parameters LPF for BEMF e_s estimation	0x0064H
SFR_PAGE = 3	MaxSMCError	MAXSMC Error for SMC in Linear Region	0x7FFFH
SFR_PAGE = 4	EST-SP	ω* (Output value of PLL-PI-Control)	Read-Only

25.13.9 Field Oriented Controller Data SFR

SFR	Description		Address
FOC_D_L	Field Oriented Control Data Low byte		D4H
FOC_D_H	Field Oriented Control Data High byte		D5H
	Parameters	Description	Reset Value
SFR_PAGE = 0	VD_OFFSET	d-axis voltage offset	0x0000H
SFR_PAGE = 1	VQ_OFFSET	q-axis voltage offset	0x0000H
SFR_PAGE = 3	AS	Angle Supplement Data	0x0000H
SFR_PAGE = 4	CPU_ANG	CPU Angle Data Register	0x0000H
SFR_PAGE = 5	FOC_ANG	Park and Inverse Park transformation angle input ※ **Read: Theta for FOC angle.** **Write: Theta offset.**	0x0000H
SFR_PAGE = 6	SVPWM_Amp	Amplitude value of SVPWM transformation	0x4000H

25.13.10 MPWMCONT SFR

MPWMCONT(SYNC) Address = E3H Reset Value = 0x00H

MPWM Control Register

Bit	-----	HALF SWAP	ALL SWAP	UWXZ SWAP	LOWSIDE		HIGHSIDE	
	7	6	5	4	3	2	1	0
Type	X	R/W	R/W	R/W	R/W	R/W	R/W	R/W

HALFSWAP [6]	Half pin swap (X,V swap),(Y,W swap) and (X,W swap) 0: Normal 1: Half change
ALLSWAP [5]	High side and Low side pin swap(U->X , X->U、V->Y,Y->V、W->X,X->W) 0: Normal 1: Inverse
UWXZSWAP [4]	U、W pin swap and X、Z pin swap 0: Normal 1: Swap
LOWSIDE [2:3]	Low-side (X、Y、Z) 00: Force Low 10: Active High 01: Force High 11: Active Low
HIGHSIDE [1:0]	High-side (U、V、W) 00: Force Low 10: Active High 01: Force High 11: Active Low

25.13.11 MPWMDT SFR

MPWMDT (SYNC) Address = E5H Reset Value = 0x78H
Motor PWM Dead-Time Register

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

25.13.12 MPWMINV SFR

MPWMINV (SYNC) Address = E4H Reset Value = 0x00H
MPWM Inverse Selection Register

Bit	-----	-----	ZINV	WINV	YINV	VINV	XINV	UINV
	7	6	5	4	3	2	1	0
Type	X	X	R/W	R/W	R/W	R/W	R/W	R/W

ZINV [5]	Low-side PWM Z output inverse select : 0 : Non-inverse 1 : Inverse
WINV [4]	High-side PWM W output inverse select : 0 : Non-inverse 1 : Inverse
YINV [3]	Low-side PWM Y output inverse select : 0 : Non-inverse 1 : Inverse
VINV [2]	High-side PWM V output inverse select : 0 : Non-inverse 1 : Inverse
XINV [1]	Low-side PWM X output inverse select : 0 : Non-inverse 1 : Inverse
UINV [0]	High-side PWM U output inverse select : 0 : Non-inverse 1 : Inverse

25.13.13 MPWM DATA SFR

SFR	Description		Address
MPWMDATL	MPWM Data Low Byte(SYNC)		E2H
MPWMDATH	MPWM Data High Byte(SYNC)		E1H
	Parameters	Description	Reset Value
SFR_PAGE = 0	MPWM Cycle	Motor PWM cycle value	0x04B0H
SFR_PAGE = 1	Phase A	Motor PWM Duty value for Phase A	0x0258H
SFR_PAGE = 2	Phase B	Motor PWM Duty value for Phase B	0x0258H
SFR_PAGE = 3	Phase C	Motor PWM Duty value for Phase C	0x0258H

25.13.14 Analog OCP Control SFR

AOCPCONT Address = EEH Reset Value = 0xE7H

Analog OCP Control Register

Bit	DOCPNEN	AOCPEN	OPAPD	-----	-----	I_SHORT[3:0]		
Bit	7	6	5	4	3	2	1	0
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W

DOCPNEN [7]	Digital OCPN enable: 0 : Disable 1 : Enable
AOCPEN [6]	Analog OCP enable: 0 : Disable 1 : Enable
OPAPD [5]	OPA Power Down 0 : Normal 1 : OPA Power Down
I_ SHORT [2:0]	Analog OCP SHORT level select : (OCP interrupt :OCPIF) 000 : 0.15V 100 : 0.35V 001 : 0.2V 101 : 0.4V 010 : 0.25V 110 : 0.45V 011 : 0.3V 111 : 0.5V(default)

25.13.15 DOCPN Control SFR

DOCPCONT Address = EFH Reset Value = 0x05H

Digital OCP Control Register

Bit	OCPST	AOCPDBT[4:0]					OCPC	OCPMS
Bit	7	6	5	4	3	2	1	0
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

OCPST [7]	OCP Short status : 0 : No Over current Short. 1 : Over current Short occur. (hardware set OCPC = '0') Six PWM output is high-impedance.
AOCPDBT [6:2]	Analog OCP input de-bounce time (default 41.67nS) 0~31 = 0~1.291uS (48MHz/2 fixed)
OCPC [1]	OCP status clear bit : When OCP is occur, hard ware will set OCPC = '0'. In User mode, user can write '1' to clear OCP status, the PWM will output at next PWM cycle.
OCPMS [0]	OCP mode select : 0 : Auto mode. 1: User mode.

25.13.16 IPD Control SFR

IPD_CTRL Address = A5H Reset Value = 0x18H

Initial Position Detection Control Register

Bit	IPDS	IPDMIN			IPDPNS	IPDCYC		IPDSTR
Bit	7	6	5	4	3	2	1	0
Type	R	R	R	R	R/W	R/W	R/W	R/W

IPDS [7]	IPD State: 0:Normal 1:Busy	
IPDMIN [6:4]	IPD Minimum Pattern: 001: Pattern1 011: Pattern3 101: Pattern5	 010: Pattern2 100: Pattern4 110: Pattern6
IPDPNS [3]	IPD Pattern Number Select: 0: 4 pattern 1: 6 pattern	
IPDPCYC [2:1]	IPD Pattern Cycle: 00: 13ms (48MHz Count) 10: 54ms (12MHz Count)	 01: 27ms (24MHz Count) 11: 110ms (6MHz Count)
IPDSTR [0]	IPD Start: 1: Start IPD	

25.13.17 IPD Data SFR

SFR	Description							
IPD_PD	Initial Position Detection Pattern Define Register							
	Parameters		Description					Reset Value
SFR_PAGE = 0	IPDPAT21_DEF		IPD Pattern 2、1 Define					0xEBH
SFR_PAGE = 1	IPDPAT43_DEF		IPD Pattern 4、3 Define					0x96H
SFR_PAGE = 2	IPDPAT65_DEF		IPD Pattern 6、5 Define					0x7DH
IPD_PD	Address = A4H		Reset Value = 0x00H					
	IPD_PD [7:0]							
	7	6	5	4	3	2	1	0
Type	W	W	W	W	W	W	W	W
SFR_PAGE = 0	Pattern2 High/Low Side on Define				Pattern1 High/Low Side on Define			
SFR_PAGE = 1	Pattern4 High/Low Side on Define				Pattern3 High/Low Side on Define			
SFR_PAGE = 2	Pattern6 High/Low Side on Define				Pattern5 High/Low Side on Define			
	U、V、W Bit[7:6]		X、Y、Z Bit[5:4]		U、V、W Bit[3:2]		X、Y、Z Bit[1:0]	
	00 : -- 01 : U 10 : V 11 : W		00 : -- 01 : X 10 : Y 11 : Z		00 : -- 01 : U 10 : V 11 : W		00 : -- 01 : X 10 : Y 11 : Z	

25.13.18 General Low Pass Filter SFR

SFR	Description		
GEN_LPF	General Low Pass Filter Parameter		
	Parameters	Description	Reset Value
SFR_PAGE = 0	GEN_GAIN	General LPF Gain parameter	0x0000H
SFR_PAGE = 1	LPF_IN	General LPF input data	0x0000H
SFR_PAGE = 2	LPF_OUT	General LPF output data	0x0000H
SFR_PAGE = 3	LPF_PREOUT	General LPF last output data	0x0000H

25.13.19 FG Control Register SFR

FGCTRL Address = 1025H Reset Value = 0x00H

FG Control Register

	FG_EN	---	---	---	---	FGPULSE_SEL		
Bit	7	6	5	4	3	2	1	0
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W

FG_EN [7] FG output enable:

0 : Disable

1 : Enable

FGPULSE_SEL [2:0] FG pulse number selection:

000 : 1 pulses/cycle

001 : 2 pulses/cycle

010 : 4 pulses/cycle

011 : 8 pulses/cycle

100 : 12 pulses/cycle

25.13.20 Coordinate Transformation Block Signals observe XSFR

XSFR	Description	Address	Characteristic
IA_L	ADC Output of phase A current (Ia) Low Byte	1060H	Read Only
IA_H	ADC Output of phase A current (Ia) High Byte	1061H	Read Only
IALPHA_L	α-axis current (iα) Low Byte of CLARKE Transform	1062H	Read Only
IALPHA_H	α-axis current (iα) High Byte of CLARKE Transform	1063H	Read Only
IB_L	ADC Output of phase B current (Ib) Low Byte	1058H	Read Only
IB_H	ADC Output of phase B current (Ib) High Byte	1059H	Read Only
IBETA_L	β-axis current (iβ) Low Byte of CLARKE Transform	105AH	Read Only
IBETA_H	β-axis current (iβ) High Byte of CLARKE Transform	105BH	Read Only

25.13.21 Inverse Coordinate Transformation Block Signals observe XSFR

XSFR	Description	Address	Characteristic
VALPHA_L(Read)	α-axis Stator Voltage Data Register Low Byte	1064H	Read Only
VALPHA_H(Read)	α-axis Stator Voltage Data Register High Byte	1065H	Read Only
VA_L(Read)	Phase A Drive Voltage Data Register Low Byte	1066H	Read Only
VA_H(Read)	Phase A Drive Voltage Data Register High Byte	1067H	Read Only
VBETA_L(Read)	β-axis Stator Voltage Data Register Low Byte	105CH	Read Only
VBETA_H(Read)	β-axis Stator Voltage Data Register High Byte	105DH	Read Only
VB_L(Read)	Phase B Drive Voltage Data Register Low Byte	105EH	Read Only
VB_H(Read)	Phase B Drive Voltage Data Register High Byte	105FH	Read Only

25.13.22 BEMF Estimation Signals observe XSFR

XSFR	Description	Address	Characteristic
ZALPHA_L	The Z gain of EEMF Eα Data Register Low Byte	1070H	Read Only
ZALPHA_H	The Z gain of EEMF Eα Data Register High Byte	1071H	Read Only
ZBETA_L	The Z gain of EEMF Eβ Data Register Low Byte	1068H	Read Only
ZBETA_H	The Z gain of EEMF Eβ Data Register High Byte	1069H	Read Only
ES_IALPHA_L	Estimate Current Iα Data Register Low Byte	1072H	Read Only
ES_IALPHA_H	Estimate Current Iα Data Register High Byte	1073H	Read Only
ES_EALPHA_L	Estimate EEMF Eα Data Register Low Byte	1074H	Read Only
ES_EALPHA_H	Estimate EEMF Eα Data Register High Byte	1075H	Read Only
ES_IBETA_L	Estimate Current Iβ Data Register Low Byte	106AH	Read Only
ES_IBETA_H	Estimate Current Iβ Data Register High Byte	106BH	Read Only
ES_EBETA_L	Estimate EEMF Eβ Data Register Low Byte	106CH	Read Only
ES_EBETA_H	Estimate EEMF Eβ Data Register High Byte	106DH	Read Only

26. SYNC

MDE behavior is synchronized with MPWM, many MDE SFRs have shadow register that is used to update these SFRs at the same time with SYNC register. Write SYNC any value will synchronization update these SFRs at the same time.

SYNC Address = 8FH Reset Value = 00000000B

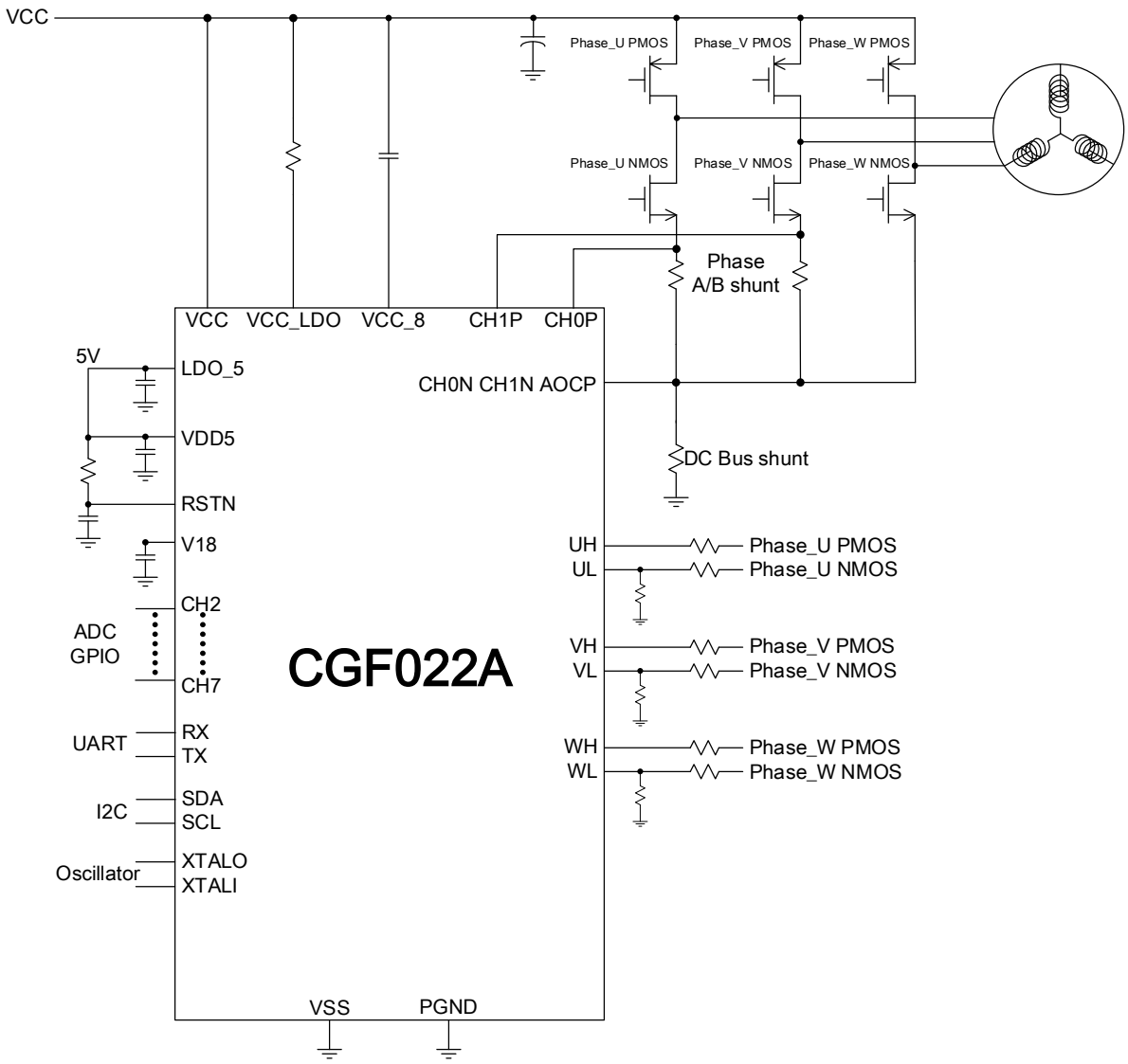
MDE Sync Register

Bit	SYNC[7:0]							
	7	6	5	4	3	2	1	0
Type	W	W	W	W	W	W	W	W

Write only.

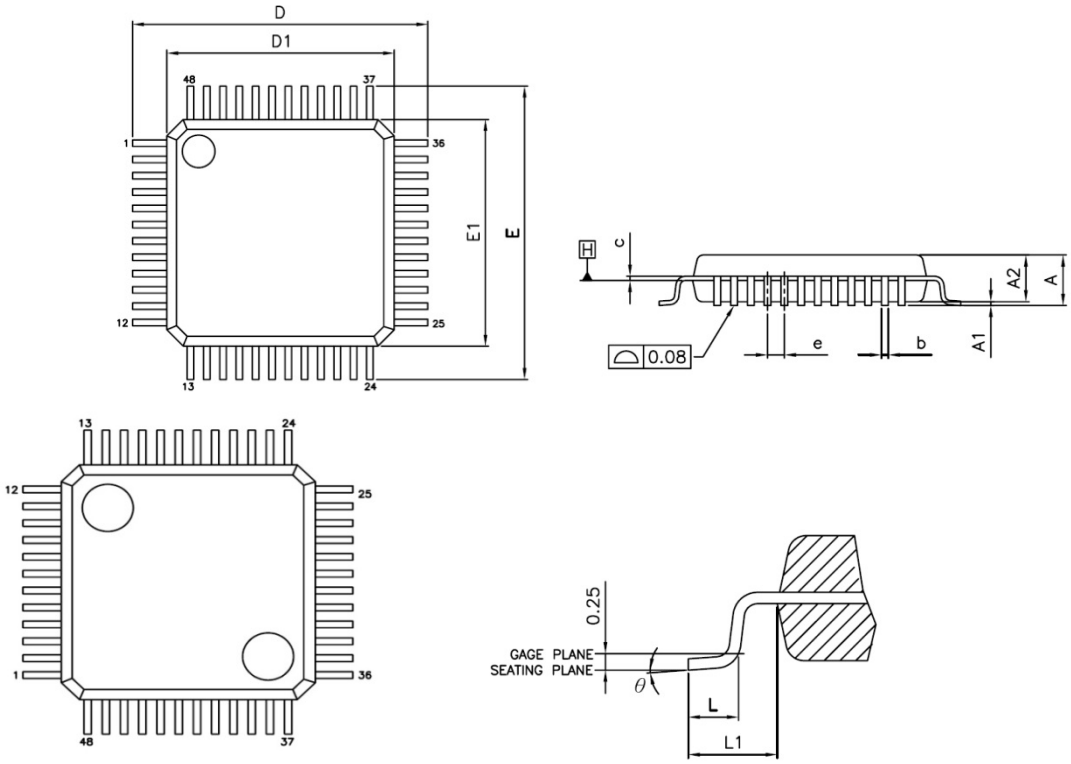
Shadow register: (need SYNC)
MPWMCONT
MPWMDT
MPWMINV
MPWMDATAL
MPWMDATAH
CPU_ANG_L
CPU_ANG_H
VDQ_OFFSET_L
VDQ_OFFSET_H
SVPWM_AMP

27. Typical Application Circuit



28. Package Information

28.1 LQFP-48 7x7mm (AD48) Outline Dimensions



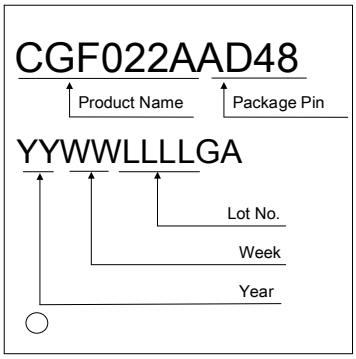
Unit	mm			inch		
Symbols	Min.	Nom.	Max.	Min.	Nom.	Max.
A	---	---	1.60	---	---	0.063
A1	0.05	---	0.15	0.002	---	0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
b	0.17	0.22	0.27	0.007	0.009	0.011
c	0.09	---	0.20	0.004	---	0.008
D	9.00 BSC			0.354 BSC		
D1	7.00 BSC			0.276 BSC		
E	9.00 BSC			0.354 BSC		
E1	7.00 BSC			0.276 BSC		
e	0.50 BSC			0.020 BSC		
L	0.45	0.60	0.75	0.018	0.024	0.030
L1	1.00 REF			0.039 REF		
ɵ	0˚	3.5˚	7˚	0˚	3.5˚	7˚

28.2 Marking Distinguish

28.2.1 Standard Ink (w/o code)

Figure 28.1 Standard Ink (w/o code)

LQFP48



28.2.2 Customization Ink (with customization code)

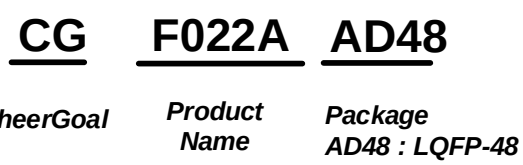
Figure 28.2 Customization Ink (with customization code)

LQFP48

29. Ordering Information

29.1 Standard Product Name

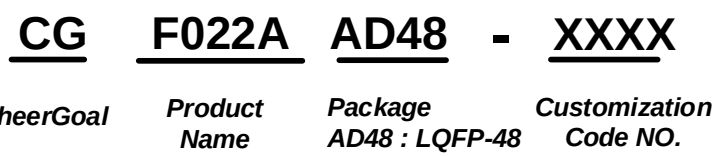
Figure 29.1 Standard Product Name



29.2 Customization Product Name

- With customization code, produce by order.

Figure 29.2 Customization Product Name



30. Revision History

Update Date	Version	Modify content
2024/05/09	V0.0	Preliminary
2024/08/16	V0.1	1. The links of contents updated 2. Features updated

31. Disclaimers

Herein, CheerGoal stands for "***CheerGoal Technology Co., Ltd.***"

Life Support — This product is not designed for use in medical, life-saving or life-sustaining applications, or systems where malfunction of this product can reasonably be expected to result in personal injury. Customers using or selling this product for use in such applications do so at their own risk and agree to fully indemnify CheerGoal for any damages resulting from such improper use or sale.

Right to Make Changes — CheerGoal reserves the right to make changes in the products - including circuits, standard cells, and/or software - described or contained herein in order to improve design and/or performance. When the product is in mass production, relevant changes will be communicated via an Engineering Change Notification (ECN).