



MG65PG5A08A

数据手册

8 位带 40 点 LCD 驱动的微控制器

版本 **2.01**

目录表

1	特性	5
1.1	应用领域	5
2	一般描述	6
3	引脚结构	7
3.1	SOP20	7
3.2	SSOP28	8
3.3	QFN32	10
4	方框图	12
5	功能描述	13
5.1	寄存器	13
5.2	累加器	13
5.3	索引寄存器(X,Y)	13
5.4	处理器状态寄存器	13
5.5	程序计数器(PC)	13
5.6	堆栈指针(S)	13
6	存储空间结构	14
6.1	特殊功能寄存器(SFR)编址	15
6.2	条件写保护寄存器	16
7	中断	17
7.1	中断寄存器	17
7.1.1	中断请求(IRQ)使能寄存器	17
7.1.2	中断请求(IRQ)状态标志	17
7.1.3	中断请求(IRQ)清零标志	17
7.1.4	系统状态标志	18
7.2	中断系统	18
8	复位	19
8.1	低电压复位(LVR)	19
8.2	看门狗定时器(WDT)	20
8.3	复位好	21
8.4	编程注意	21
9	电源控制寄存器	22
9.1	省电控制	22
10	系统时钟寄存器	23
10.1	时钟寄存器	23
11	实时时钟(RTC)分频器	24
11.1	实时时钟(RTC)分频器寄存器	24
12	实时时钟	25
12.1	实时时钟(RTC)寄存器	25
13	定时器	26
13.1	定时器 0(TM0)	26
13.2	定时器 1(TM1)	27
13.3	定时器 2(TM2)	28

14	脉宽调制(PWM)	29
14.1	脉宽调制(PWM)控制寄存器	29
14.2	脉宽调制(PWM)缓冲寄存器	30
14.3	脉宽调制(PWM)死区时间选择寄存器	30
14.4	脉宽调制(PWM)输出控制寄存器	31
15	输入输出(I/O)端口配置	32
15.1	端口 0(P0)	32
15.1.1	端口 0(P0)	32
15.1.2	端口 0(P0)输出缓冲	32
15.1.3	端口 0(P0)方向寄存器	32
15.1.4	端口 0(P0)上拉控制寄存器	32
15.1.5	端口 0(P0)模拟功能控制寄存器	32
15.1.6	端口 0(P0)中断沿选择寄存器	32
15.2	端口 1(P1)	33
15.2.1	端口 1(P1)	33
15.2.2	端口 1(P1)输出缓冲	33
15.2.3	端口 1(P1)方向寄存器	33
15.2.4	端口 1(P1)上拉控制寄存器	34
15.3	端口 0(P0)及端口 1(P1)功能控制寄存器	34
15.3.1	端口 0(P0)及端口 1(P1)输出控制寄存器	34
15.4	端口 2(P2)	34
15.4.1	端口 2(P2)输出缓冲	34
15.5	端口 3(P3)	35
15.5.1	端口 3(P3)输出缓冲	35
15.6	LCD控制器/驱动器	36
15.6.1	LCD控制寄存器	36
15.6.2	Segment /输入输出(I/O)选择	36
16	模数转换器(ADC)	38
16.1	模数转换器(ADC)控制寄存器	38
16.2	模数转换器(ADC)数据总线	39
17	数模转换器(DAC)	40
17.1	数模转换器 0(DAC0)	40
17.2	模数转换器 1(DAC1)	40
17.3	模数转换器(DAC)输出控制寄存器	41
18	运算放大器(OPA)和比较器	42
18.1	比较器 0 控制寄存器	42
18.2	比较器 1 控制寄存器	42
18.3	运算放大器(OPA)控制寄存器	42
18.4	比较器和运算放大器(OPA)输出状态	43
18.5	比较器和运算放大器(OPA)去抖控制寄存器	43
19	在应用可编程(IAP)	44
20	选项寄存器	45
21	应用电路	46
21.1	参考原理图	46
22	电气特性	47
22.1	绝对最大值	47

22.2	直流特性.....	47
22.3	交流特性.....	48
22.4	模数转换器(ADC),运算放大器(OPA)和比较器特性.....	48
23	版本历史	49

1 特性

- 单芯片 8 位中央处理单元(CPU)
- 存储器
 - 程序空间 OTP (ROM): 8K 字节
 - 数据空间(RAM) : 128 字节
 - LCD 数据空间(RAM): 8 字节
- 工作电压: 2.0V 到 5.5V
- 16 个可编程通用输入输出(GPIO)
 - 输入/输出脚 P0[7:0]
 - 输入/输出脚 P1[7:0]
 - 两个 PWM 输出脚 (PWM1, PWM0 共享 P1.7, P1.6)
 - 输出脚 P2[7:0] / Segment2 ~ Segment9
 - 输出脚 P3[1:0] / Segment0 ~ Segment1
 - 输出脚 P3[5:2] / Common3 ~ Common0
- LCD 驱动输出
 - 最大 10 segment x 4 common
 - 1/4 duty 及 1/3 bias 驱动模式
- 内置一个看门狗(watchdog)定时器
- 内置一个 8 通道 12 位的模数转换器(ADC)
- 内置两个 8 位的数模转换器(DAC)
- 内置两个电压比较器
- 内置一个运算放大器
- 内置实时时钟(RTC)
- 一个可重装载 8 位定时器
- 一个可重装载 10 位定时器
- 一个可重装载 16 位定时器
- 一个 10 位的脉宽调制 (PWM)
- 一个 16 位的脉宽调制 (PWM)
- 省电功能的暂停(HALT)和停止(STOP)模式
- 内置三个振荡电路:
 - 32MHz ($\pm 2\%$)内振
 - 32KHz 内振
 - 32KHz 晶振
 - 双时钟工作
 - 由选项选择振荡器脚或 P1.2~ P1.3
- 内置低压侦测器(典型电压:低于 2.4V 或 2.7V 由特殊功能寄存器 SFR 选择)和低电压复位(典型电压:低于 2.0V)

1.1 应用领域

通用 LCD 控制器, 手持游戏, 玩具, 电源管理

2 一般描述

MG65PG5A08A 是笙泉科技出品的一颗高性价比的 8 位微控制器。单芯片上集成了 8 位中央处理单元(CPU)，OTP ROM，RAM，定时器，LCD 驱动器，输入输出端口(I/O)和系统控制电路。MG65PG5A08A 提供一个内置振荡器作为时钟源。非常适合一般的 LCD 控制器, 手持游戏, 玩具控制器和其他产品。

3 引脚结构

3.1 SOP20

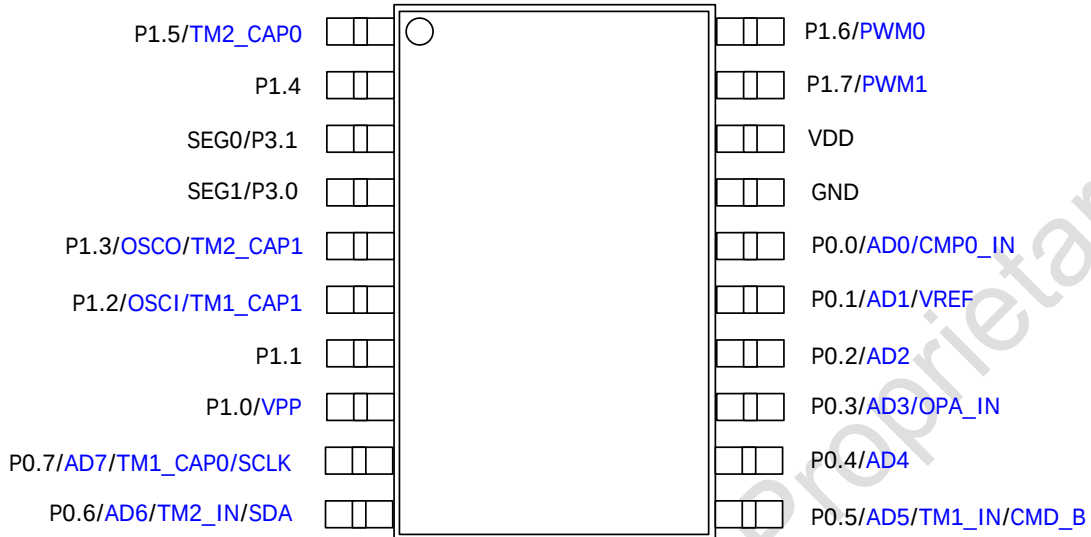


图 3-1 封装 SOP20

脚位	名称	类型	描述
1	P1.5	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。
2	P1.4	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。
3	SEG0 / P3.1	O	LCD segment 信号输出脚及 NMOS 上拉输出脚。
4	SEG1 / P3.0	O	LCD segment 信号输出脚及 NMOS 上拉输出脚。
5	P1.3 / OSCO	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。共用 32K 振荡器(OSC)脚。
6	P1.2 / OSCI	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。共用 32K 振荡器(OSC)脚。
7	P1.1	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。
8	P1.0 / VPP	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。OTP 接口的 VPP 引脚共用 P1.0。
9	P0.7 / AD7 / SCLK	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 7 输入和 OTP 接口的 SCLK 引脚共用 P0.7。
10	P0.6 / AD6 / SDA	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 6 输入和 OTP 接口的 SDA 引脚共用 P0.6。
11	P0.5 / AD5 / CMD_B	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 5 输入和 OTP 接口的 CMD_B 引脚共用 P0.5。
12	P0.4 / AD4	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 4 输入共用 P0.4。
13	P0.3 / AD3 / OPA_IN	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 3 输入和 OPA 输入共用 P0.3。
14	P0.2 / AD2	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 2 输入共用 P0.2。
15	P0.1 / AD1 / VREF	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 1 输入和模数转换(ADC)外部参考电压输入引脚共用 P0.1。

16	P0.0 / AD0 / CMP0_IN	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 0 输入和电压比较器输入引脚共用 P0.0。
17	GND	G	地
18	VDD	P	电源正极
19	P1.7 / PWM1	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。PWM1 输出共用 P1.7。
20	P1.6 / PWM0	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。PWM0 输出共用 P1.6。

注意:类型列表

“I”意思是仅输入。

“O”意思是仅输出。

“B”意思是双向口。

“P”意思是电源。

“G”意思是地。

3.2 SSOP28

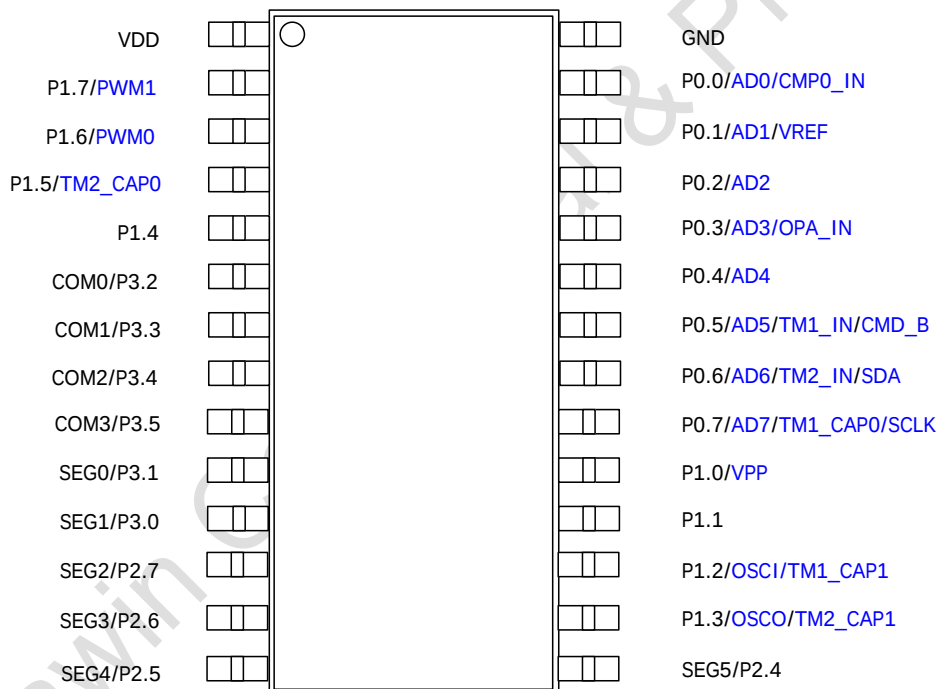


图 3-2 封装 SSOP28

脚位	名称	类型	描述
1	VDD	P	电源正极
2	P1.7 / PWM1	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。PWM1 输出共用 P1.7。
3	P1.6 / PWM0	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。PWM0 输出共用 P1.6。
4	P1.5	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。
5	P1.4	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。
6	COM0 / P3.2	O	LCD segment 信号输出脚 COM0 及 NMOS 上拉输出脚 P3.2。

7	COM1 / P3.3	O	LCD segment 信号输出脚 COM1 及 NMOS 上拉输出脚 P3.3。
8	COM2 / P3.4	O	LCD segment 信号输出脚 COM2 及 NMOS 上拉输出脚 P3.4。
9	COM3 / P3.5	O	LCD segment 信号输出脚 COM3 及 NMOS 上拉输出脚 P3.5。
10	SEG0 / P3.1	O	LCD segment 信号输出脚 SEG0 及 NMOS 上拉输出脚 P3.1。
11	SEG1 / P3.0	O	LCD segment 信号输出脚 SEG1 及 NMOS 上拉输出脚 P3.0。
12	SEG2 / P2.7	O	LCD segment 信号输出脚 SEG2 及 NMOS 上拉输出脚 P2.7。
13	SEG3 / P2.6	O	LCD segment 信号输出脚 SEG3 及 NMOS 上拉输出脚 P2.6。
14	SEG4 / P2.5	O	LCD segment 信号输出脚 SEG4 及 NMOS 上拉输出脚 P2.5。
15	SEG5 / P2.4	O	LCD segment 信号输出脚 SEG5 及 NMOS 上拉输出脚 P2.4。
16	P1.3 / OSC0	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。共用 32K 振荡器(OSC)脚。
17	P1.2 / OSC1	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。共用 32K 振荡器(OSC)脚。
18	P1.1	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。
19	P1.0 / VPP	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。OTP 接口的 VPP 引脚共用 P1.0。
20	P0.7 / AD7 / SCLK	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 7 输入和 OTP 接口的 SCLK 引脚共用 P0.7。
21	P0.6 / AD6 / SDA	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 6 输入和 OTP 接口的 SDA 引脚共用 P0.6。
22	P0.5 / AD5 / CMD_B	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 5 输入和 OTP 接口的 CMD_B 引脚共用 P0.5。
23	P0.4 / AD4	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 4 输入共用 P0.4。
24	P0.3 / AD3	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 3 输入和 OPA 输入共用 P0.3。
25	P0.2 / AD2	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 2 输入共用 P0.2。
26	P0.1 / AD1 / VREF	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 1 输入和模数转换(ADC)外部参考电压输入引脚共用 P0.1。
27	P0.0 / AD0	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 0 输入和电压比较器输入引脚共用 P0.0。
28	GND	G	地

注意:类型列表

“I”意思是仅输入。

“O”意思是仅输出。

“B”意思是双向口。

“P”意思是电源。

“G”意思是地。

3.3 QFN32

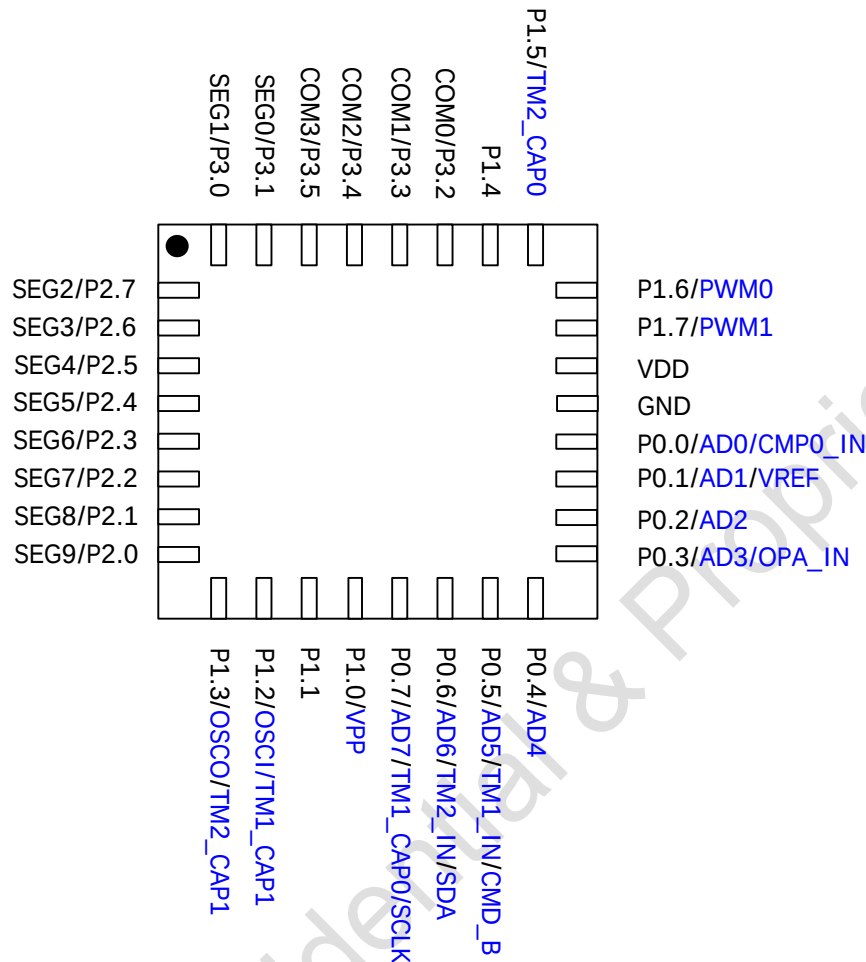


图 3-3 封装 QFN32

脚位	名称	类型	描述
1	SEG2 / P2.7	O	LCD segment 信号输出脚 SEG2 及 NMOS 上拉输出脚 P2.7。
2	SEG3 / P2.6	O	LCD segment 信号输出脚 SEG3 及 NMOS 上拉输出脚 P2.6。
3	SEG4 / P2.5	O	LCD segment 信号输出脚 SEG4 及 NMOS 上拉输出脚 P2.5。
4	SEG5 / P2.4	O	LCD segment 信号输出脚 SEG5 及 NMOS 上拉输出脚 P2.4。
5	SEG6 / P2.3	O	LCD segment 信号输出脚 SEG6 及 NMOS 上拉输出脚 P2.3。
6	SEG7 / P2.2	O	LCD segment 信号输出脚 SEG7 及 NMOS 上拉输出脚 P2.2。
7	SEG8 / P2.1	O	LCD segment 信号输出脚 SEG8 及 NMOS 上拉输出脚 P2.1。
8	SEG9 / P2.0	O	LCD segment 信号输出脚 SEG9 及 NMOS 上拉输出脚 P2.0。
9	P1.3 / OSCO	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。共用 32K 振荡器(OSC)脚。
10	P1.2 / OSCI	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。共用 32K 振荡器(OSC)脚。
11	P1.1	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。

12	P1.0 / VPP	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。OTP 接口的 VPP 引脚共用 P1.0。
13	P0.7 / AD7 / SCLK	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 7 输入和 OTP 接口的 SCLK 引脚共用 P0.7。
14	P0.6 / AD6 / SDA	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 6 输入和 OTP 接口的 SDA 引脚共用 P0.6。
15	P0.5 / AD5 / CMD_B	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 5 输入和 OTP 接口的 CMD_B 引脚共用 P0.5。
16	P0.4 / AD4	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 4 输入共用 P0.4。
17	P0.3 / AD3	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 3 输入和 OPA 输入共用 P0.3。
18	P0.2 / AD2	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 2 输入共用 P0.2。
19	P0.1 / AD1 / VREF	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 1 输入和模数转换(ADC)外部参考电压输入引脚共用 P0.1。
20	P0.0 / AD0	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。模数转换(A/D)通道 0 输入和电压比较器输入引脚共用 P0.0。
21	GND	G	地
22	VDD	P	电源正极
23	P1.7 / PWM1	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。PWM1 输出共用 P1.7。
24	P1.6 / PWM0	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。PWM0 输出共用 P1.6。
25	P1.5	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。
26	P1.4	B	可编程输入输出(I/O), CMOS 输出, 上拉输入和输入中断功能。
27	COM0 / P3.2	O	LCD segment 信号输出脚 COM0 及 NMOS 上拉输出脚 P3.2。
28	COM1 / P3.3	O	LCD segment 信号输出脚 COM1 及 NMOS 上拉输出脚 P3.3。
29	COM2 / P3.4	O	LCD segment 信号输出脚 COM2 及 NMOS 上拉输出脚 P3.4。
30	COM3 / P3.5	O	LCD segment 信号输出脚 COM3 及 NMOS 上拉输出脚 P3.5。
31	SEG0 / P3.1	O	LCD segment 信号输出脚 SEG0 及 NMOS 上拉输出脚 P3.1。
32	SEG1 / P3.0	O	LCD segment 信号输出脚 SEG1 及 NMOS 上拉输出脚 P3.0。

注意:类型列表

“I”意思是仅输入。

“O”意思是仅输出。

“B”意思是双向口。

“P”意思是电源。

“G”意思是地。

4 方框图

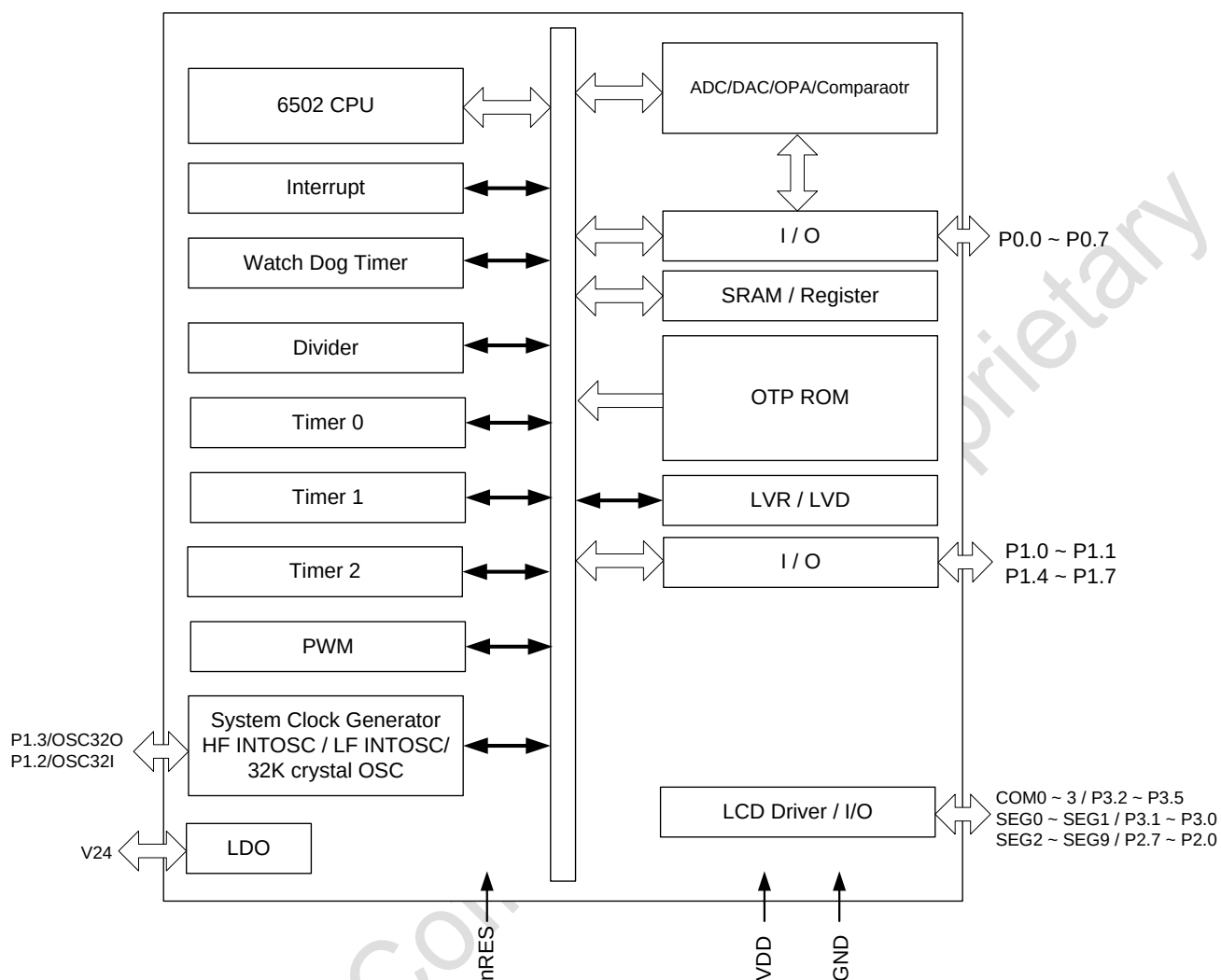


图 4-1 方框图

5 功能描述

5.1 寄存器

	A
	Y
	X
	P
PCH	PCL
1	S

5.2 累加器

累加器是一个通用 8 位寄存器，存储着大多数的算术和逻辑运算的结果。另外，累加器通常包含在这些工作中两个数据字的一个。

5.3 索引寄存器(X,Y)

有两个 8 位索引寄存器(X 和 Y)，可以用于计数程序步骤或在产生一个有效地址时提供一个索引值。当执行指定被索引编址的指令时，中央处理器(CPU)获取操作代码(OP)和基本地址并且在执行期望的操作之前通过添加索引寄存器的值来修改操作地址。前索引后索引都是可能的。

5.4 处理器状态寄存器

8 位处理器状态寄存器包含 7 个状态标志。部分标志由程序控制，其他则由程序和中央处理器(CPU)共同控制。

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
N	V	1	B	D	I	Z	C

N: 符号标志, 1 = 负数, 0 = 正数

V: 溢出标志, 1 = 溢出, 0 = 无溢出

B: BRK 中断命令, 1 = BRK, 0 = IRQB

D: 十进制模式, 1 = 十进制, 0 = 二进制

I: IRQB 禁止标志, 1 = 中断禁止, 0 = 中断使能

Z: 零标志, 1 = true, 0 = false

C: 进位标志, 1 = true, 0 = false

5.5 程序计数器(PC)

16 位程序计数器通过系列程序指令压入微控制器提供地址。微控制器每一次从程序空间获取一个指令，程序计数器低字节(PCL)放入地址总线的低 8 位及程序计数器高字节(PCH)放入地址总线的高 8 位。程序计数器每一次增加则从程序空间获取一个指令或数据

5.6 堆栈指针(S)

堆栈指针是个用于控制不同长度堆栈编址的 8 位寄存器。在程序或中断(/NMI 或/IRQ)指引下微控制器执行堆栈操纵时堆栈指针将自动增加或减少。堆栈允许嵌套子程序的简单执行和多层中断。堆栈指针由用户软件初始化。

6 存储空间结构

MG65PG5A08A 内有地址为 0000H 到 007FH 的 128 字节的静态随机存储器(SRAM)。根据应用程序的分配它们用作工作寄存器(RAM) 或堆栈。从上来看, 0000H 到 007FH 及 0100H 到 017FH 是重叠的。换句话说, 0000H 到 007FH 的任何位置访问等同于 0100H 到 017FH 相应位置访问。所有的特殊功能寄存器(SFR)都位于 00B0H 到 00FFH。这样的编排利于更快速访问零页。

MG65PG5A08A 有 8K 字节的程序/数据只读存储器(ROM)。只读存储器(ROM)编址从 E000H 到 FFFFH 可以存储程序和数据。MG65PG5A08A 地址分布表如下所示。

MG65PG5A08A Memory Map

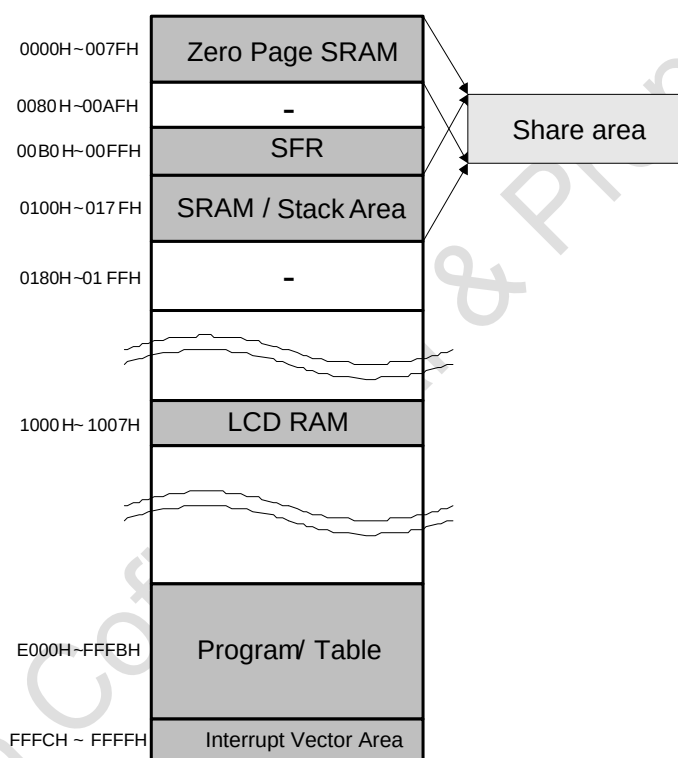


图 6-1 存储空间图

6.1 特殊功能寄存器(SFR)编址

00B0H 到 00FFH 保留给特殊功能寄存器(SFR)。特殊功能寄存器(SFR)用于控制或存储输入输出(I/O)的状态，定时器，系统时钟和其它外设。

※所有特殊功能寄存器(SFR)不支持位操作指令

表 6-1 SFR 表

SFR (特殊功能寄存器): 00B0H~00FFH

地址	名称	缺省值
00B0	RESOK	XXXX----
00B1		-----
00B2	IRQ_EN	00000000
00B3		-----
00B4	IRQ_ST/IRQ_CLR	00000000
00B5		-----
00B6		-----
00B7	SYS_ST	0-----X
00B8		-----
00B9	DIV_SEL/DIV_ST	-----00
00BA	SCK_SEL	0-000-0
00BB		-----
00BC	PWR_CR	-0-0-000
00BD		-----
00BE	RTC	--000000
00BF		-----

地址	名称	缺省值	地址	名称	缺省值
00C0	LCD_CR	0-----0	00D0		-----
00C1	SEG_SEL	-----111	00D1	TM0	11111111
00C2		-----	00D2	TM0_CTL	00---000
00C3	PWMO_CTL	00000000	00D3		-----
00C4	DACO_CTL	0000-000	00D4	TM1L	11111111
00C5		-----	00D5	TM1H	-----11
00C6	DAC0	00000000	00D6	TM1_CAPL	-----11
00C7	DAC1	00000000	00D7	TM1_CAPH	11111111
00C8		-----	00D8	TM1_CTL	00000000
00C9	DT_SEL	-0000000	00D9		-----
00CA	PB_SEL	----0000	00DA	TM2L	11111111
00CB		-----	00DB	TM2H	11111111
00CC	CMP0_CTL	00000000	00DC	TM2_CAPL / PWMR_L	11111111
00CD	CMP1_CTL	00000000	00DD	TM2_CAPH / PWMR_H	11111111
00CE	OPA_CTL	00000000	00DE	TM2_CTL	00000000
00CF	COCP	----000	00DF		-----

地址	名称	缺省值	地址	名称	缺省值
00E0	PWMR0	00000000	00F0		-----
00E1	PWMR1	-----10	00F1	P2obuf	00000000
00E2	PWM_CTL / PWM_ST	00001111	00F2		-----
00E3		-----	00F3	P3obuf	--000000
00E4	P0port / P0obuf	00000000 / XXXXXXXX	00F4		-----
00E5	P0dir	00000000	00F5	ADC_CTL0/ADC_STS	00000000
00E6	P0plh	11111111	00F6	ADC_CTL1	----0000
00E7	P0an	00000000	00F7	ADB_L	0000----
00E8	P0es	00000000	00F8	ADB_H	00000000
00E9		-----	00F9		-----

00EA	P1port / p1obuf	00000000/ XXXXXXXX	00FA	CWPR	XXXXXXXX
00EB	P1dir	00000000	00FB	IAP_PR	XXXXXXXX
00EC	P1plh	00111111	00FC		-----
00ED		-----	00FD		-----
00EE	POsel	--000000	00FE		-----
00EF		-----	00FF		-----

6.2 条件写保护寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00FAH	CWPR	PT7	PT6	PT5	PT4	PT3	PT2	PT1	PT0	-	√

条件写保护标志寄存器(CWPR)用于保护 [SYS_ST.7 \(WDT\)](#), [PWR_CR.1 \(CKC0\)](#), [PWR_CR.2 \(CKC1\)](#), [PWR_CR.0 \(HALT\)](#) 和 [SCK_SEL](#)。如果想更改这些特殊功能寄存器(SFR)，则必须先写“78H”到寄存器(CWPR)。

[PT7~PT0](#):写保护模式。MG65PG5A08A 的写保护模式是“78H”

注意:

1. When CWPR is written by firmware, 当软件写 CWPR 时，在软件的“下一次写动作”之后它将由硬件自动清零。

※此寄存器(SFR)不支持位操作指令

7 中断

MG65PG5A08A 有 8 个中断源。IRQ_EN 和 IRQ_ST 标志用于中断控制。当 IRQ_ST 标志由硬件置位并且 IRQ_EN 标志的相应位由软件置位则一个中断将产生。当一个中断出现，所有中断被抑制直到 CLI 或 STA IRQ_EN, #I 指令的调用。执行 SEI 指令也能禁止所有中断。

表 7-1 中断向量表

向量地址	项目	标志	特性	备注
FFFEH, FFFFH	P0 IRQ	P0evt	Ext.	P0.0 ~ P0.7 中断
	P1 IRQ	P1evt	Ext.	P1.0 ~ P1.7 中断
	TM0 IRQ	TM0evt	Int.	TM0 溢出中断
	TM1 IRQ	TM1evt	Int.	TM1 溢出中断
	TM2 IRQ	TM2evt	Int.	TM2 溢出中断
	DIV	DIVevt	Int.	分频器(Divider)完成中断
	RTC	RTCevt	Int.	实时时钟(RTC)中断
FFFCH, FFFDH	ADC	ADCevt	Int.	模数转换(ADC)完成中断
	RESB	None	Ext.	外部复位信号
	WDT	SYS_ST.7	Int.	看门狗定时器(WDT)复位
	LVR	None	Int.	低电压复位

7.1 中断寄存器

7.1.1 中断请求(IRQ)使能寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00B2H	IRQ_EN	RTC	ADC	DIV	TM2	TM1	TM0	P1	P0	-	√

程序通过此寄存器可以使能或禁止触发中断请求(IRQ)的能力。

0: 禁止 (初始化的缺省值为 "0")

1: 使能

P0: 在 P0 中断模式 P0 的上升或下降沿出现 (由 P0ES 设置)

P1: P1 输入模式的下降沿出现

TM0: 定时器 0 溢出

TM1: 定时器 1 溢出

TM2: 定时器 2 溢出

DIV: 分频器(Divider)选择的中断频率出现

ADC: 模数转换(ADC)数据完成

RTC: 实时时钟(RTC) 0.5S 中断

7.1.2 中断请求(IRQ)状态标志

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00B4H	IRQ_ST	RTCevt	ADCevt	DIVevt	TM2evt	TM1evt	TM0evt	P1evt	P0evt	√	-

当中断请求(IRQ)出现，程序可以读取此寄存器来识别哪个中断触发中断请求(IRQ)。如果中断触发使能并且中断事件可以接受，在中断向量载入到程序计数器(PC)之后相一致的中断请求(IRQ)状态标志需要程序清零。

7.1.3 中断请求(IRQ)清零标志

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00B4H	IRQ_CLR	RTC	ADC	DIV	TM2	TM1	TM0	P1	P0	-	√

程序在相一致的位写'1'则清除中断事件。

7.1.4 系统状态标志

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00B7H	SYS_ST	WDT	-	-	-	-	-	-	LVD	√	√

WDT: 看门狗(WDT)复位标志。(由 POR, LVR 和外部复位清零)

1: 看门狗(WDT)复位出现。

0: 此位由硬件置位并且写‘1’清零。(此寄存器由 CWPR 写保护)

LVD: 低电压侦测。(由 WDT, POR, LVR 和外部复位清零)

1: VDD 在 2.4V 或 2.7V 之下置位。

0: 此位由硬件置位并且写‘1’清零。

7.2 中断系统

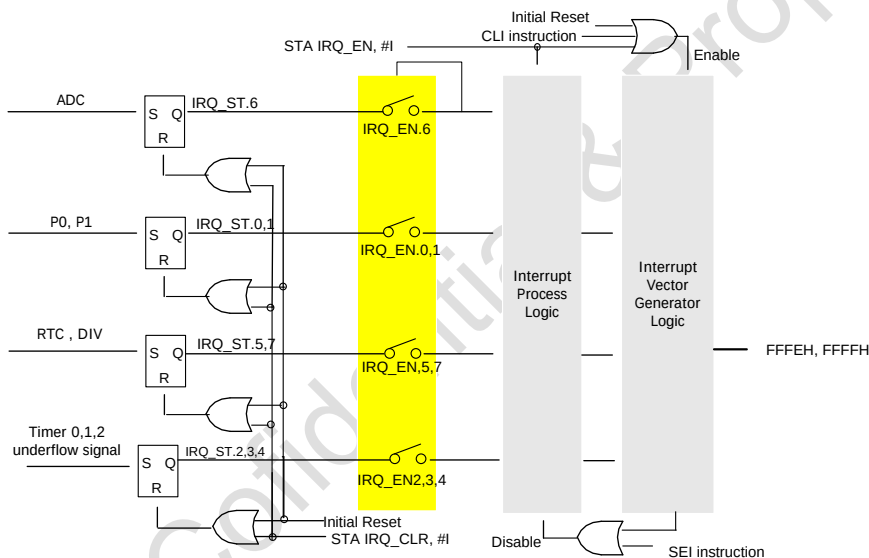


图 7-1 中断系统图

8 复位

MG65PG5A08A 提供 5 种复位源(外部复位, LVR, POR, WDT 和复位好)。芯片复位电路如下所示:

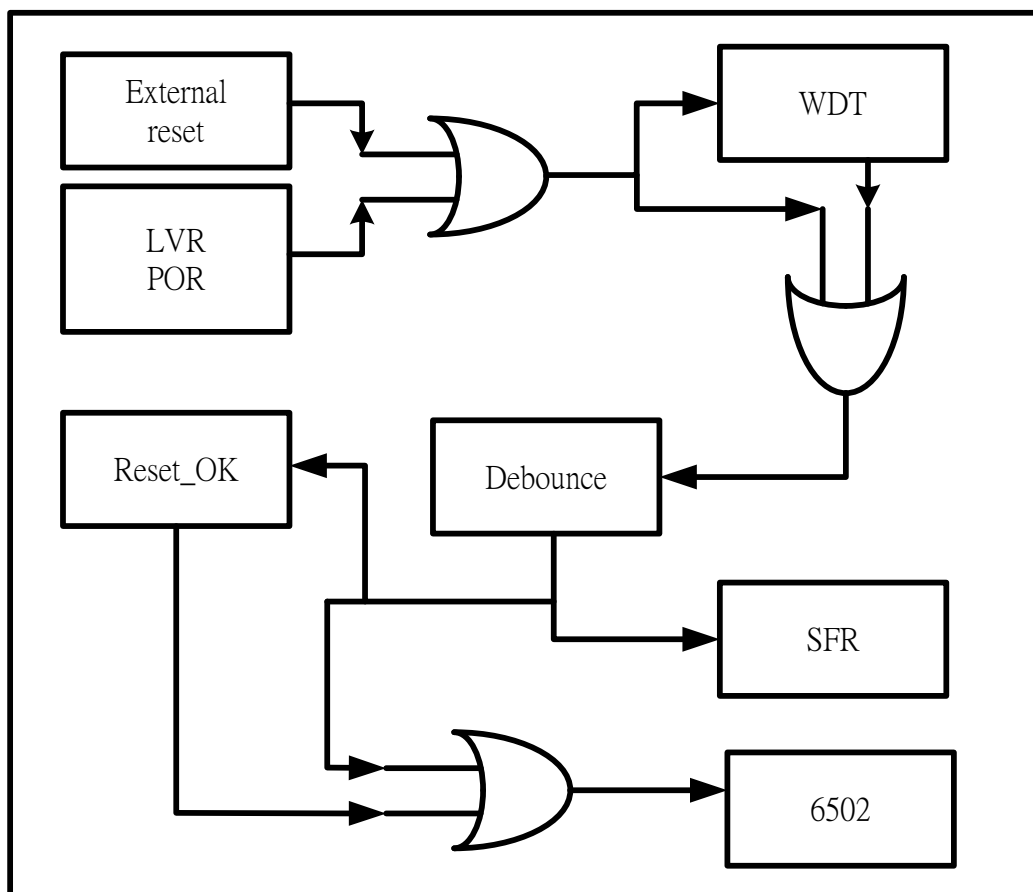


图 8-1 系统复位图

8.1 低电压复位(LVR)

MG65PG5A08A 为了监控器件的供给电压提供低电压复位电路。如果器件的供给电压在 0.9V ~ VLVR 之间, 比如更换电池, 低电压复位(LVR)将内部自动复位器件。

低电压复位(LVR)有以下规格

1. 低电压(0.9V~VLVR)状态需要保持超过 1 毫秒(ms)。如果低电压状态小于 1 毫秒(ms), 则低电压复位(LVR)被忽略并且不执行复位。
2. 在低电压复位(LVR)模式, 静态随机存储器(SRAM)将保持。端口 1(P1)和端口 2(P2)将是三态。LCD 驱动是缺省值。

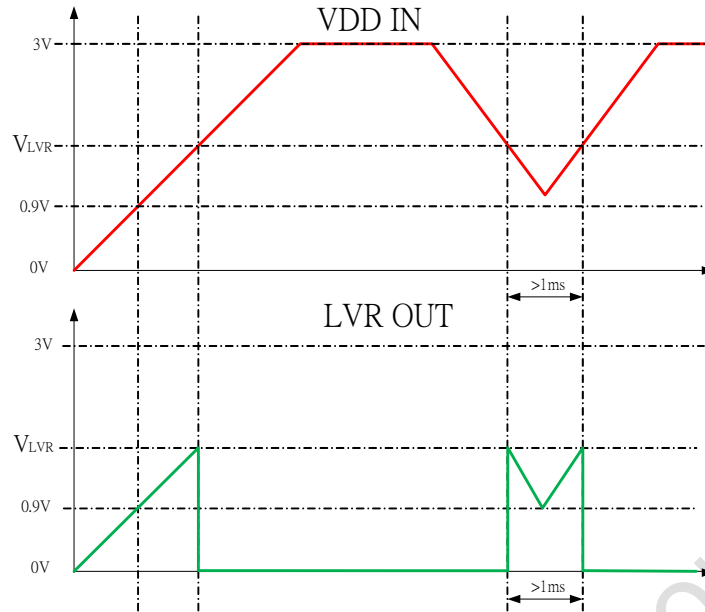


图 8-2 低电压复位保持时间

8.2 看门狗定时器(WDT)

(例子是基于 32.768 KHz 及 SCK_SEL.Bit4 = 0)

名称		Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
WDT	-	1Hz	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz	-	-

看门狗定时器溢出周期由 SCK_SEL.Bit4 (CKS4)选择的等式($FSOSC / 256$)/128 或($FMOSC / 256$)/128 获取。
在看门狗定时器溢出之前，程序必须写 ‘1’ 到 STS_ST.7 清除看门狗定时器(WDT)的位 7。看门狗定时器(WDT)溢出将导致系统复位并且 SYS_ST.7 置位。

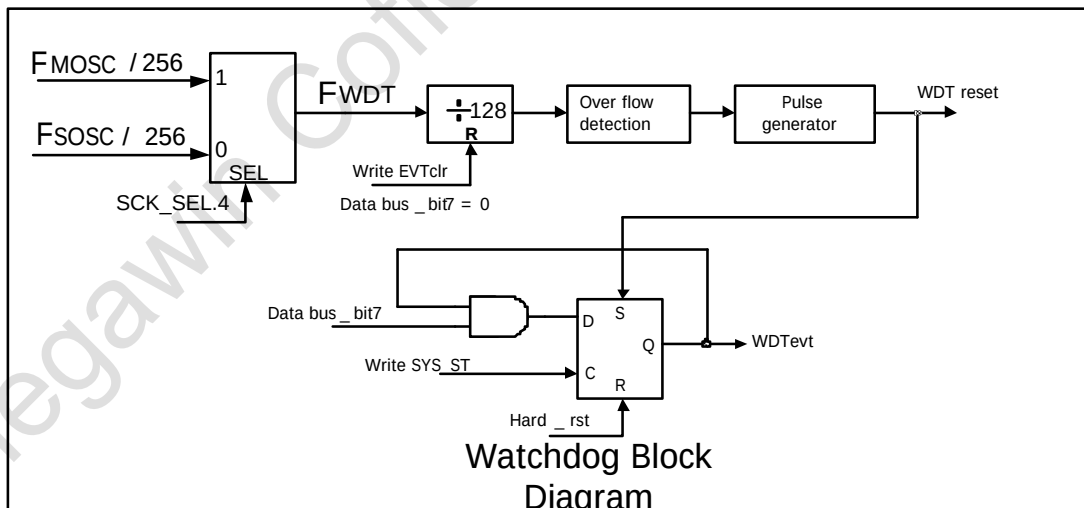


图 8-2 看门狗框图

8.3 复位好

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00B0H	RESOK	RK7	RK6	RK5	RK4	-	-	-	-	-	√

RESOK (复位好): 如果器件复位好并且工作正常, 必须写#\$90 到此寄存器。

例如:

```
Program_start:      LDA    #10010000b
                   STA    $B0
```

8.4 编程注意

不同的复位状态见下表:

	上电复位	CPU /外部(RES)复位
SRAM 数据	未知	不变
CPU 寄存器	未知	未知
特殊功能寄存器(SFR)	默认值	默认值

9 电源控制寄存器

9.1 省电控制

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00BCH	PWR_CR	-	IO_RES	-	LVD_SEL	-	CKC1	CKC0	HALT	-	√

IO_RES: 输入输出(IO)复位选择。

0: 输入输出(IO)状态由 WDT, LVR 和 POR, 外部复位(默认)复位。

1: 输入输出(IO)状态由 LVR, POR 和外部复位复位。

LVD_SEL: 低电压检测(LVD)电压选择

0: 低电压检测(LVD)电压是 2.4V。

1: 低电压检测(LVD)电压是 2.7V。

※ CKC1, CKC0 和 HALT 由 CWPR 保护

CKC1	CKC0	系统时钟控制
0	0	F _{MOSC} 使能, F _{SOSC} 使能(双频模式)
0	1	F _{MOSC} 使能, F _{SOSC} 禁止(单频模式)
1	0	F _{MOSC} 禁止, F _{SOSC} 使能(低频模式)
1	1	F _{MOSC} 禁止, F _{SOSC} 禁止(停止模式)

HALT: F_{CPU} 离线控制位。1: F_{CPU} 离线, 0: F_{CPU} 在线

为了省电通过此寄存器程序可以切换正常操作模式为省电模式。此系统有三种省电模式。

低速模式: (PWR_CR.CKC1 = 1, PWR_CR.CKC0 = 0)

主频(F_{MOSC})停止振荡。微控制器仅需要非常低的功耗保持运行。

停止模式: (PWR_CR.CKC1 = 1, PWR_CR.CKC0 = 1)

双系统时钟停止振荡。微控制器从停止模式唤醒有 4 种途径: 端口 0(P0)下降沿, 端口 0(P0)上升沿, 端口 1(P1)下降沿, 硬件复位或上电复位。当停止模式释放之后, **振荡器会再次使能。**

如果微控制器时钟源是 F_{SOSC} 并且系统进入停止模式 (设置 PWR_CR[2:1] = 11), 当微控制器通过端口 0(P0)或端口 1 (P1)唤醒则 F_{SOSC} 使能并且 F_{MOSC} 保持之前的状态。

暂停模式: (PWR_CR.HALT = 1)

FCPU 时钟进入离线状态。振荡器任然保持之前状态。微控制器从暂停模式唤醒有 3 种途径: 中断事件, 硬件复位或上电复位。

10 系统时钟寄存器

10.1 时钟寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00BAH	SCK_SEL	CKS7	-	-	CKS4	CKS3	CKS2	-	CKS0	-	√

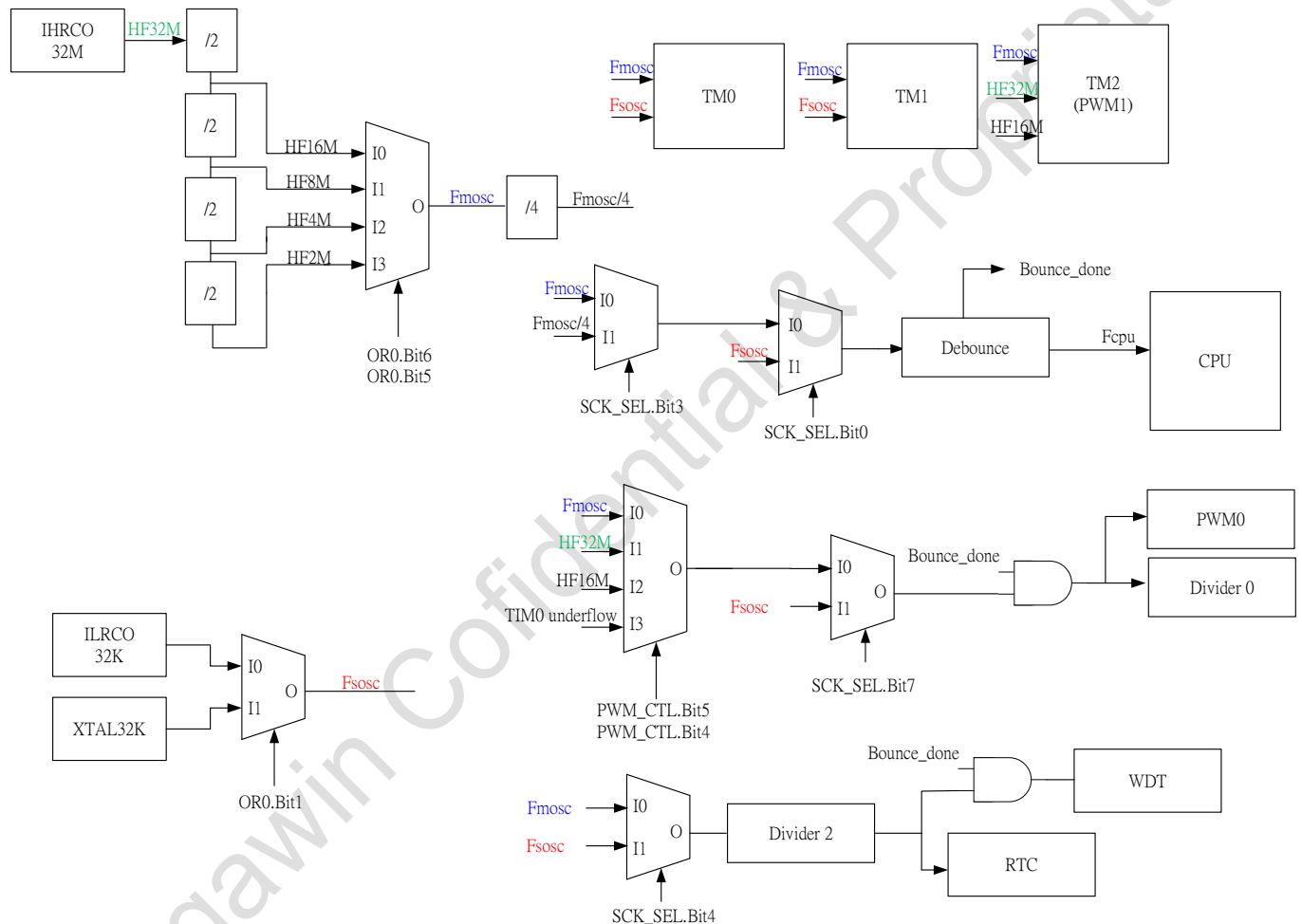
CKS0: F_{CPU} 时钟源选择。0: F_{MOSC} , 1: F_{SOSC}

CKS2: 去抖(从停止模式唤醒) 时间选择。0: $F_{CPU}/16384$, 1: $F_{CPU}/256$

CKS3: F_{CPU} 时钟源选择。0: $F_{MOSC}/4$, 1: F_{MOSC}

CKS4: 选择分频器(divider2)输入时钟源 (F_{RTC} 和 F_{WDT} 时钟)。0: F_{SOSC} 1: F_{MOSC}

CKS7: 选择分频器(divider0)输入时钟源(PWM 时钟)。0: F_{MOSC} , HF32M, HF16M 或 Tim0 溢出 1: F_{SOSC}



11 实时时钟(RTC)分频器

MG65PG5A08A 有一个 8 位分频器(divider)和 4 种中断选择使用。分频器(divider)时钟源是 F_{MOSC} 或 F_{SOSC} 。分频器由 POR 或 LVR 复位为 00h

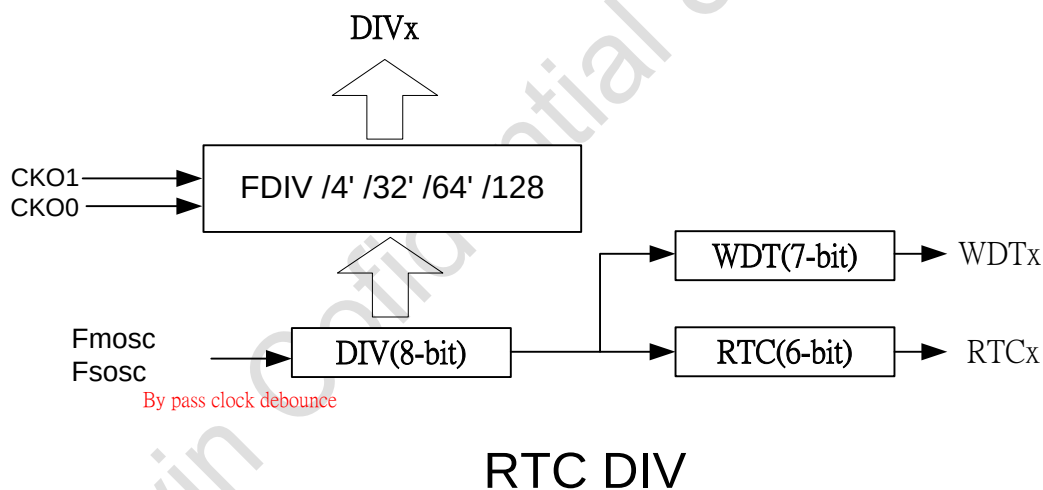
11.1 实时时钟(RTC)分频器寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00B9H	DIV_ST	FDIV/ 256	FDIV/ 128	FDIV/ 64	FDIV/ 32	FDIV/ 16	FDIV/ 8	FDIV/ 4	FDIV/ 2	√	-
00B9H	DIV_SEL			-	-	-		CKO1	CKO0	-	√

CKO1, CKO0:选择分频器(DIV)中断频率

例如: (Fdiv 是 32768Hz)

CKO1	CKO0	选择分频器(DIV)频率 (FDIV= F_{SOSC})
0	0	FDIV / 4 (8192 Hz)
0	1	FDIV / 32 (1024 Hz)
1	0	FDIV / 64 (512 Hz)
1	1	FDIV / 128 (256 Hz)



12 实时时钟

12.1 实时时钟(RTC)寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00BEH	RTC	-	-	S5	S4	S3	S2	S1	S0	√	√

实时时钟(RTC)包括一个自动加一的 6 位寄存器，一个有集成电容的片内 32.768 kHz 振荡器，一个频率分配器作为实时时钟(RTC)的时钟源。

程序能通过 IRQ_EN.7 寄存器禁止或使能触发实时时钟(RTC)中断，读取 IRQ_ST.7 便知道是否实时时钟(RTC)触发中断。实时时钟(RTC)寄存器和 IRQ_ST.7 可以被 POR, LVR 复位。程序可以通过写'1'到 IRQ_CLR.7 来清除实时时钟(RTC)中断事件。

(例子频率是 $F_s = 32.768 \text{ KHz}$)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00BEH	RTC	-	-	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz	√	√

13 定时器

13.1 定时器 0(TM0)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00D1H	TM0	T7	T6	T5	T4	T3	T2	T1	T0	√	√
00D2H	TM0_CTL	STC	RL/S	-	-	-	TCS0	TKI1	TKI0		√

定时器 0 是一个 8 位的倒计数计数器。

STC:启动/停止计数。1: 启动并预装载值到计数器, 0: 停止定时器时钟

RL/S:自动装载禁止/使能。1: 禁止自动装载, 0: 使能自动装载

TCS0:定时器 0 的输入时钟选择。0: F_{MOSC} , 1: F_{SOSC}

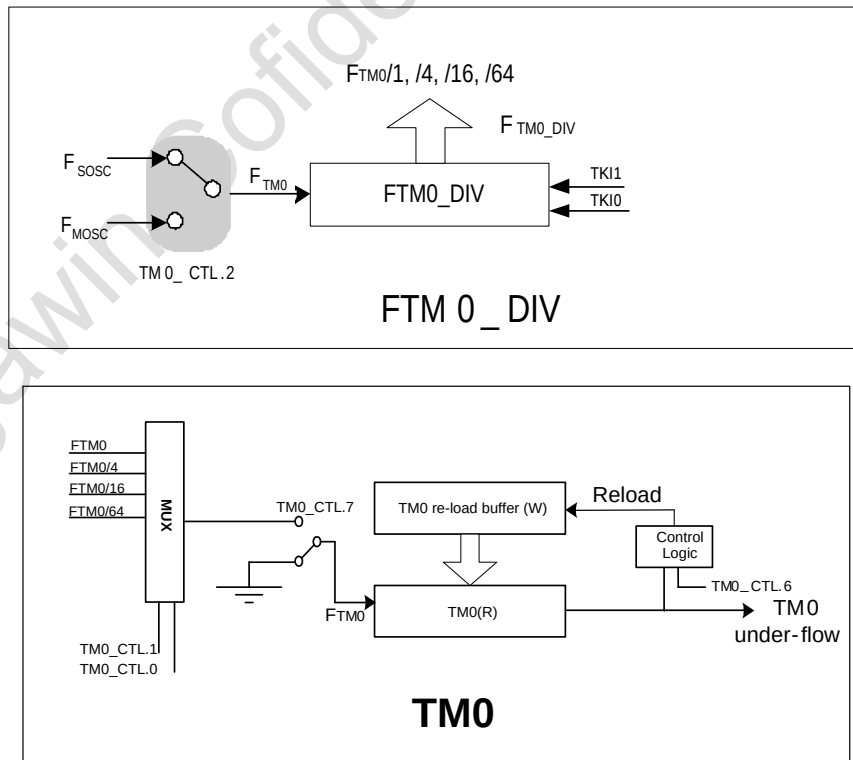
TKI1	TKI0	TM0 输入时钟选择 (F_{TM0_DIV})
0	0	$F_{TM0} / 1$
0	1	$F_{TM0} / 4$
1	0	$F_{TM0} / 16$
1	1	$F_{TM0} / 64$

F_{TM0_UV} 可以由等式计算:

$F_{TM0_UV} = F_{TM0} / (TM0+1)$,这里通过 TKI1 和 TKI0 设置定时器输入频率是 F_{TM0} 。

例如: (假如 $F_{TM0} = 2.000MHz$, TKI1=TKI0=0)

TM0	频率
00H	保留
01H	1.000MHz
02H	667kHz
...	...
FFH	7.84kHz



13.2 定时器 1(TM1)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00D5H	TM1H	-	-	-	-	-	-	T9	T8	√	√
00D4H	TM1L	T7	T6	T5	T4	T3	T2	T1	T0	√	√
00D7H	TM1_CAPH							C9	C8	√	-
00D6H	TM1_CAPL	C7	C6	C5	C4	C3	C2	C1	T0	√	-
00D8H	TM1_CTL	STC	RL/S	TKES	CPS	CTKS	TCS1	TKI1	TKI0	-	√

定时器 1 是一个 10 位的倒计时计数器。

STC: 启动/停止计数。1: 启动并预装载值到计数器, 0: 停止定时器时钟

RL/S: 自动装载禁止/使能。1: 禁止自动装载, 0: 使能自动装载

TKES: 事件或连续时钟输入触发沿选择。0: 上升沿, 1: 下降沿

CPS: 捕捉 TM1 计数值触发源选择。0: P0.7, 1: P1.2.

CTKS: 捕捉源触发沿选择。0: 上升沿, 1: 下降沿

TCS1: 定时器 1 的输入时钟选择。0: F_{MOSC} , 1: F_{SOSC}

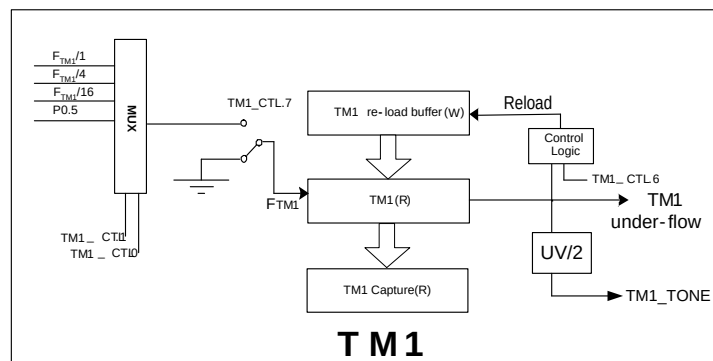
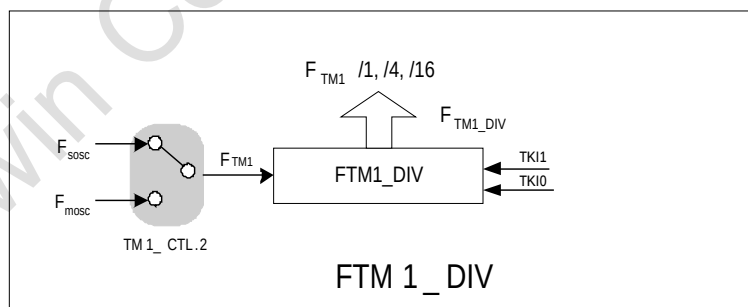
TKI1	TKI0	TM1 输入时钟选择(F_{TM1_DIV})
0	0	$F_{TM1} / 1$
0	1	$F_{TM1} / 4$
1	0	$F_{TM1} / 16$
1	1	P0.5

F_{TM1_UV} 可以由等式计算:

$F_{TM1_UV} = F_{TM1} / (TM1+1)$, 这里通过 TKI1 和 TKI0 设置定时器输入频率是 F_{TM1} 。

例如: (假如 $F_{TM1} = 2.000\text{MHz}$, $TKI1=TKI0=0$)

TM1	频率
000H	保留
001H	1.000MHz
002H	667kHz
...	...
0FFH	7.84kHz



13.3 定时器 2(TM2)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00DBH	TM2H	T15	T14	T13	T12	T11	T10	T9	T8	√	√
00DAH	TM2L	T7	T6	T5	T4	T3	T2	T1	T0	√	√
00DDH	TM2_CAP_H PWMR_H	C15	C14	C13	C12	C11	C10	C9	T8	√	√
00DCH	TM2_CAPL PWMR_L	C7	C6	C5	C4	C3	C2	C1	T0	√	√
00DEH	TM2_CTL	STC	RL/S	TKES	CPS	CTKS	ENCP	TKI1	TKI0		√

※ 此寄存器不能使用位操作指令。

定时器 2 是一个 16 位的倒计时计数器。

STC: 启动/停止计数。1: 启动并预装载值到计数器, 0: 停止定时器时钟

RL/S: 自动装载禁止/使能。1: 禁止自动装载, 0: 使能自动装载

TKES: 事件或连续时钟输入触发沿选择。0: 上升沿, 1: 下降沿

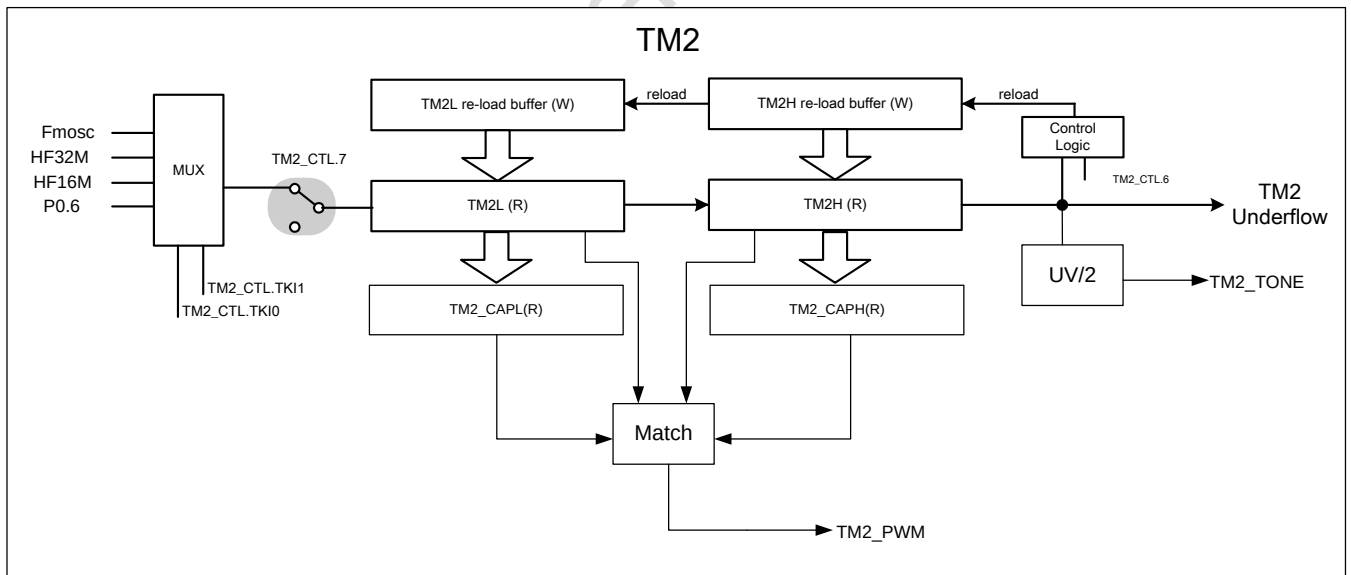
CPS: 捕捉 TM2 计数值触发源选择。0: P1.5, 1: P1.3.

CTKS: 捕捉源触发沿选择。0: 上升沿, 1: 下降沿

ENCP: TM2 捕捉功能控制。0: 禁止(PWM 缓冲使能), 1: 使能 (捕捉缓冲)

※ 当 TM2_CAP/PWMR 被配置成捕捉缓冲时, TM2 PWM 输出将被禁止。

TKI1	TKI0	TM2 输入时钟选择
0	0	Fmosc
0	1	HF32M
1	0	HF16M
1	1	P0.6



14 脉宽调制(PWM)

14.1 脉宽调制(PWM)控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E2H	PWM_CTL	DT_SCR	PWM_MOD	PWKS1	PWKS0	PWC03	PWC02	PWC01	PWC00	-	√
00E2H	PWM_ST	-	-	-	-	-	-	PWM1	PWM0	√	-

DTSCR: 脉宽调制(PWM)死区时间源控制电路选择:

0: 10 位脉宽调制(PWM)输出

1: 16 位脉宽调制(PWM)输出(与 TM2 功能共享)

PWM_MOD: 脉宽调制(PWM)输出模式选择:

PWM_MOD	PWM1 (P1.7)	PWM0 (P1.6)
0	TM2 PWM 输出	10 位 PWM 输出
1	PWMH	PWML

PWKS1, PWKS0: 脉宽调制(PWM)时钟源选择:

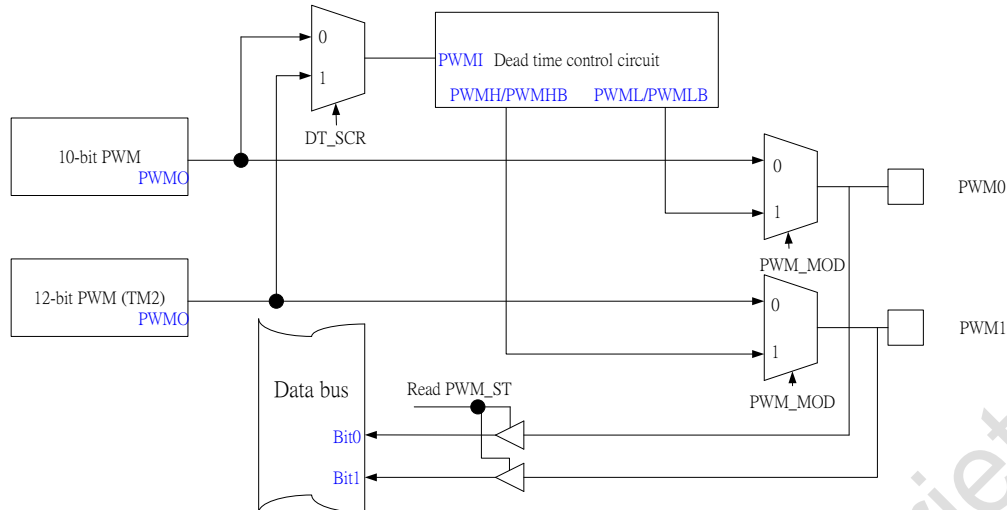
PWKS1	PWKS0	PWM 时钟源
0	0	Fmosc
0	1	HF32M
1	0	HF16M
1	1	TM0 溢出

PWC03, PWC02, PWC01, PWC00: 脉宽调制(PWM)比较位选择。

PWC03	PWC02	PWC01	PWC00	脉宽调制(PWM)计数器比较位									
				Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	0			-	-	-	-	-	-	-	PW00
0	0	0	1			-	-	-	-	-	-	PW01	PW00
0	0	1	0			-	-	-	-	-	PW02	PW01	PW00
0	0	1	1			-	-	-	-	PW03	PW02	PW01	PW00
0	1	0	0			-	-	-	PW04	PW03	PW02	PW01	PW00
0	1	0	1			-	-	PW05	PW04	PW03	PW02	PW01	PW00
0	1	1	0			-	PW06	PW05	PW04	PW03	PW02	PW01	PW00
0	1	1	1			PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00
1	0	0	0		PW08	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00
1	0	0	1	PW09	PW08	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00
1	0	1	0	PW09	PW08	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00
1	0	1	1	PW09	PW08	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00
1	1	0	0	PW09	PW08	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00
1	1	0	1	PW09	PW08	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00
1	1	1	0	PW09	PW08	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00
1	1	1	1	PW09	PW08	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00

PWM1: PWM1 输出状态。

PWM0: PWM0 输出状态。



14.2 脉宽调制(PWM)缓冲寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E1H	PWMR1	-	-	-	-	-	-	PW9	PW8	√	√
地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E0H	PWMR0	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00	√	√

脉宽调制(PWM)时钟源由 PWKS1 和 PWKS0 控制。

P1.6 引脚输出脉宽调制(PWM)波形和输出 PWM 缓冲寄存器值成比例的占空比。

14.3 脉宽调制(PWM)死区时间选择寄存器

MG65PG5A08A 由 DT_SEL 选择 PWMH 和 PWML 死区时间。

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00C9H	DT_SEL	-	PWMH_SL	PWML_SL	DTSC1	DTSC0	SEL2	SEL1	SEL0	-	√

PWMH_SL: PWMH 输出反转选择: 0 PWMH 1: 反转

PWML_SL: PWML 输出反转选择: 0 PWML 1: 反转

DTSC1, DTSC0: 死区时间时钟源选择

00: $T_d = 1/HF32M$

01: $T_d = 1/HF16M$

10: $T_d = 1/HF8M$

11: $T_d = 1/HF4M$

SEL2, SEL1, SEL0: 死区时间选择

000: 死区时间是 $[T_d - (1/HF32M)] \sim (T_d)$

001: 死区时间是 $[T_d - (1/HF32M)] \sim (2T_d)$

010: 死区时间是 $[T_d - (1/HF32M)] \sim (3T_d)$

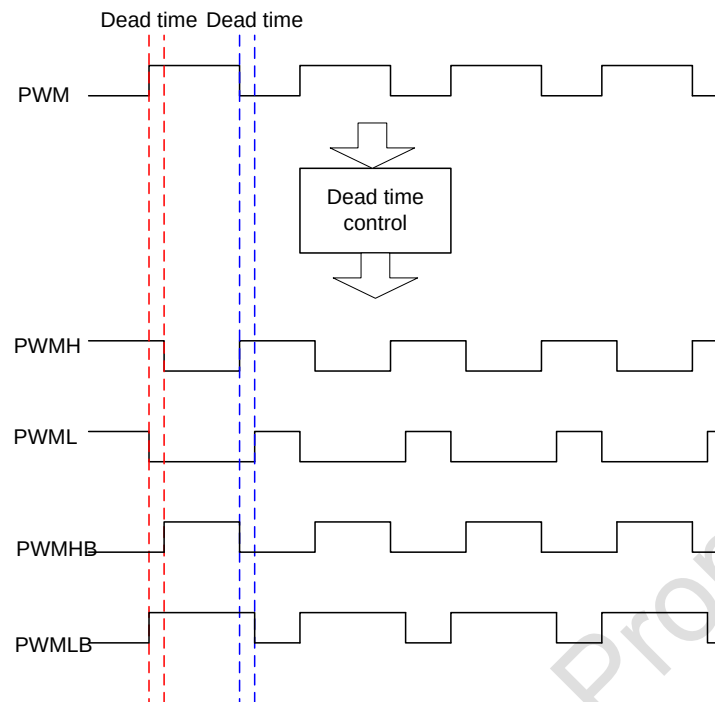
011: 死区时间是 $[T_d - (1/HF32M)] \sim (4T_d)$

100: 死区时间是 $[T_d - (1/HF32M)] \sim (5T_d)$

101: 死区时间是 $[T_d - (1/HF32M)] \sim (6T_d)$

110: 死区时间是 $[T_d - (1/HF32M)] \sim (7T_d)$

111: 死区时间是 $[T_d - (1/HF32M)] \sim (8T_d)$



14.4 脉宽调制(PWM)输出控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00C3H	PWMO_CTL	IP_PDB1	IP_PDB0	CMP1HEN	CMP1LEN	CMP0HEN	CMP0LEN	CMP1HY	CMP0HY	-	√

IP_PDB1: IP 掉电信号: 0: 掉电 OPA, CMP1 和 DAC1 1: 使能

IP_PDB0: IP 掉电信号: 0: 掉电 CMP0 和 DAC0 1: 使能

CMP1HEN: PWMH 输出控制: 0: 禁止

1: 使能→当 CMP1 输出高, P1.7 的 PWM 将被禁止。

CMP1LEN: PWML 输出控制: 0: 禁止

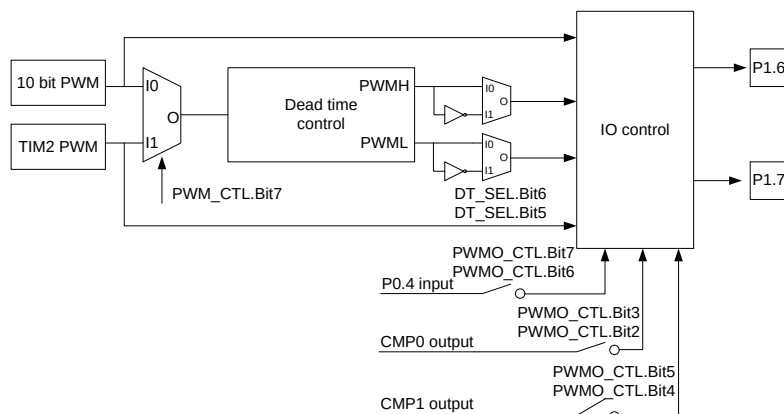
1: 使能→当 CMP1 输出高, P1.6 的 PWM 将被禁止。

CMP0HEN: PWMH 输出保护: 0: 禁止

1: 使能→当 CMP0 输出高, P1.7 的 PWM 将被禁止。

CMP0LEN: PWML 输出保护: 0: 禁止

1: 使能→当 CMP0 输出高, P1.6 的 PWM 将被禁止。



15 输入输出(I/O)端口配置

15.1 端口 0(P0)

15.1.1 端口 0(P0)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E4H	P0port	P07	P06	P05	P04	P03	P02	P01	P00	√	-

端口 0(P0)是一个 8 位的输入输出(I/O)端口。P0.7~P0.0 可以各自编程为输入或输出。当 P0.n 配置为输出口, P0.n 引脚将输出内部 P0obuf.n (P0 输出缓冲)的逻辑值。P0obuf 的默认值是 00000000b。

当 P0.n 配置为输出模式时, 读取 P0.n 总是为'0'。

当 P0.n 配置为输入模式时, 读取 P0.n 总是为相应引脚的逻辑值。

15.1.2 端口 0(P0)输出缓冲

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E4H	P0obuf	P07	P06	P05	P04	P03	P02	P01	P00	-	√

在输出模式此寄存器用于缓存 P0.7 ~ P0.0 的输出值并且只能写。

※ 此寄存器不能使用位操作。

15.1.3 端口 0(P0)方向寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E5H	P0dir	DR7	DR6	DR5	DR4	DR3	DR2	DR1	DR0	-	√

P0_DR (端口 0 方向)

P0_DR.n = 0: P0.n 配置为输入引脚。(默认值)

1: P0.n 配置为输出引脚。

※ 此寄存器不能使用位操作。

15.1.4 端口 0(P0)上拉控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E6H	P0plh	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0	-	√

1:使能内部上拉(默认值); 0:禁止内部上拉。

PHn: 控制位用于使能 P0.n 引脚的上拉。

※ 此寄存器不能使用位操作。

15.1.5 端口 0(P0)模拟功能控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E7H	P0an	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	-	√

0:正常输入输出(I/O)功能; 1:模拟功能

MFn:控制位用于 P0.n 的功能选择。

※ 此寄存器不能使用位操作。

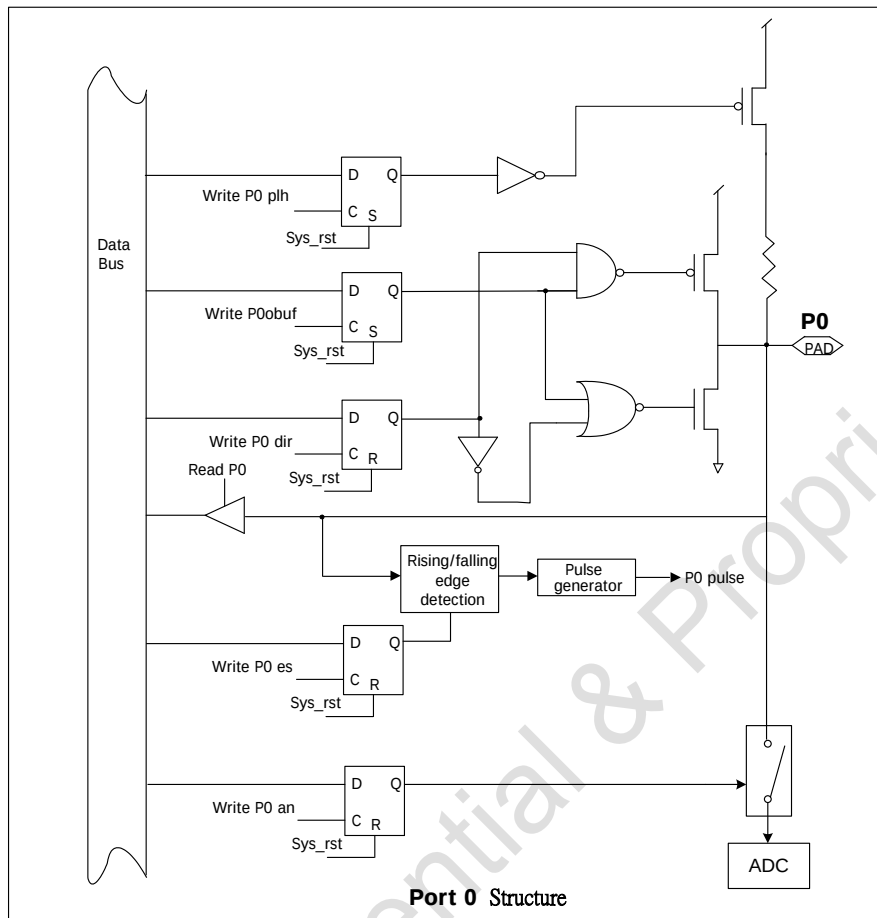
15.1.6 端口 0(P0)中断沿选择寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00E8H	P0es	ES7	ES6	ES5	ES4	ES3	ES2	ES1	ES0	-	√

0:下降沿; 1:上升沿

MFn: 控制位用于 P0.n 的中断沿选择。

※ 此寄存器不能使用位操作。



15.2 端口 1(P1)

15.2.1 端口 1(P1)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00EAH	P1port	P17	P16	P15	P14	P13	P12	P11	P10	√	-

端口 1(P1)是一个 8 位的输入输出(I/O)端口。P1.7~P1.0 可以各自编程为输入或输出。当 P1.n 配置为输出时，P1.n 引脚将输出内部 P1obuf.n (P1 输出缓冲)的逻辑值。P1obuf 的默认值是 0000000b。

当 P1.n 配置为输出模式时，读取 P1.n 总是为'0'。

当 P1.n 配置为输入模式时，读取 P1.n 总是为相应引脚的逻辑值。

15.2.2 端口 1(P1)输出缓冲

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00EAH	P1obuf	P17	P16	P15	P14	P13	P12	P11	P10	-	√

在输出模式此寄存器用于缓存 P1.7 ~ P1.0 的输出值并且只能写。

※ 此寄存器不能使用位操作。

15.2.3 端口 1(P1)方向寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00EBH	P1dir	DR7	DR6	DR5	DR4	DR3	DR2	DR1	DR0	-	√

P1_DR (端口 1 方向)

P1_DR.n = 0: P1.n 配置为输入引脚。(默认值)

1: P1.n 配置为输出引脚。

※ 此寄存器不能使用位操作。

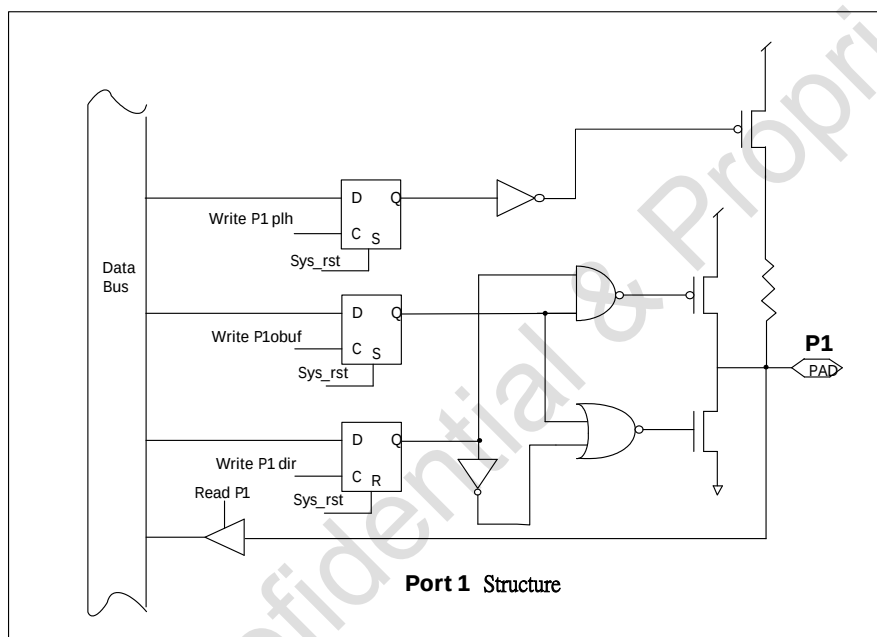
15.2.4 端口 1(P1)上拉控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00ECH	P1plh	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0	-	√

1:使能内部上拉(默认值); 0:禁止内部上拉。

PHn: 控制位用于使能 P1.n 引脚的上拉。

※ 此寄存器不能使用位操作。



15.3 端口 0(P0)及端口 1(P1)功能控制寄存器

15.3.1 端口 0(P0)及端口 1(P1)输出控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00EEH	POsel	-	-	PO5	PO4	PO3	PO2	PO1	PO0	-	√

POn:控制位用于 P0 或 P1 输出选择。

PO0: P0.7 输出选择。0: P0.7obuf 输出, 1: TM1_tone 输出。

PO1: P1.2 输出选择。0: P1.2obuf 输出, 1: TM1_tone 输出。

PO2: P1.5 输出选择。0: P1.5obuf 输出, 1: TM2_tone 输出。

PO3: P1.3 输出选择。0: P1.3obuf 输出, 1: TM2_tone 输出。

PO4: P1.6 输出选择。0: P1.6obuf 输出, 1: PWM0 输出。

PO5: P1.7 输出选择。0: P1.7obuf 输出, 1: PWM1 输出。

※ 此寄存器不能使用位操作。

15.4 端口 2(P2)

15.4.1 端口 2(P2)输出缓冲

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00F1H	P2obuf	P27	P26	P25	P24	P23	P22	P21	P20	-	√

在输出模式此寄存器用于缓存 P2.7 ~ P2.0 的输出值并且只能写。

※ 此寄存器不能使用位操作。

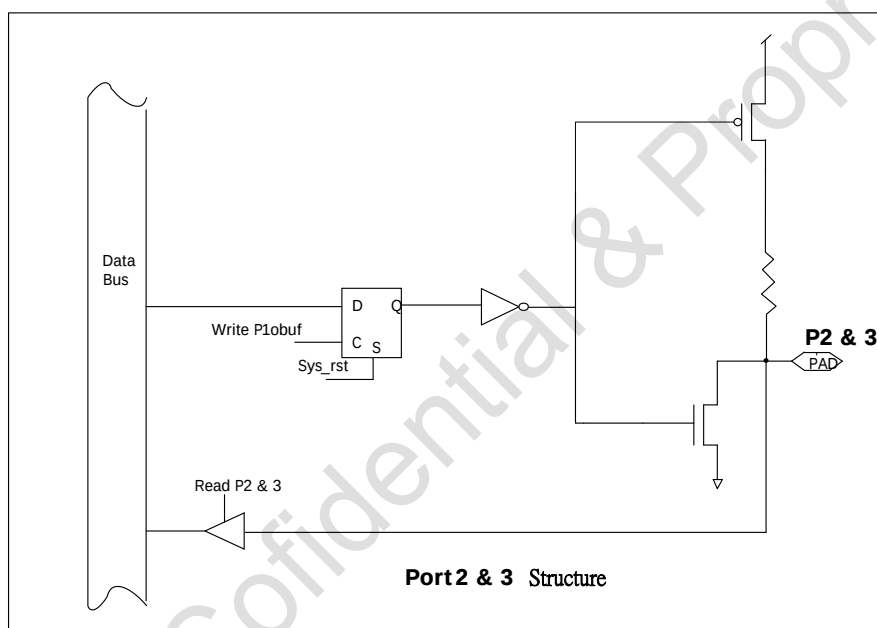
15.5 端口 3(P3)

15.5.1 端口 3(P3)输出缓冲

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00F3H	P3obuf	-	-	P35	P34	P33	P32	P31	P30	-	√

在输出模式此寄存器用于缓存 P3.5 ~ P3.0 的输出值并且只能写。

※ 此寄存器不能使用位操作。



15.6 LCD控制器/驱动器

MG65P5A08A 可以直接驱动 10 个 segment 和 4 个 common 共 10 X 4 点的 LCD。LCD 控制寄存器可以用于选择 LCD 显示配置。LCD 驱动模式是 1/3 bias 和 1/4 duty 及帧频率大约是 81.38Hz。当 CPU 访问 LCD RAM 区，LCD RAM 的访问路径将自动从 LCD 启动器到 CPU。

15.6.1 LCD控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00C0H	LCD_CR	LCDON	-	-	-	-			CKS	√	√

LCDON: LCD 开/关控制位。0: 关 (默认值), 1: 开

LCD 扫描率, F_{COM} 来源于分频器(divider)的时钟源。CKS, F_{COM} 和 F_{LCD} 之间的关系如下表所示。

CKS0	F_{COM} 频率选择
0	HF4M / 12288
1	$F_{SOSC} / 96$

LCD 帧频率的计算等式:

$$FFRAME = F_{COM} / COM\ No.$$

1/4 duty 的典型选择如下表所示:

COM 数	F_{DIV_IN}	F_{COM} 频率选择	$FFRAME$
4	4M	$F_{LCD} / 12288$	81.38
4	32K	$F_{LCD} / 96$	85.33

15.6.2 Segment /输入输出(I/O)选择

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00C1H	SEG_SEL	-	-	-	-	-	SSEL2	SSEL1	SSEL0	-	√

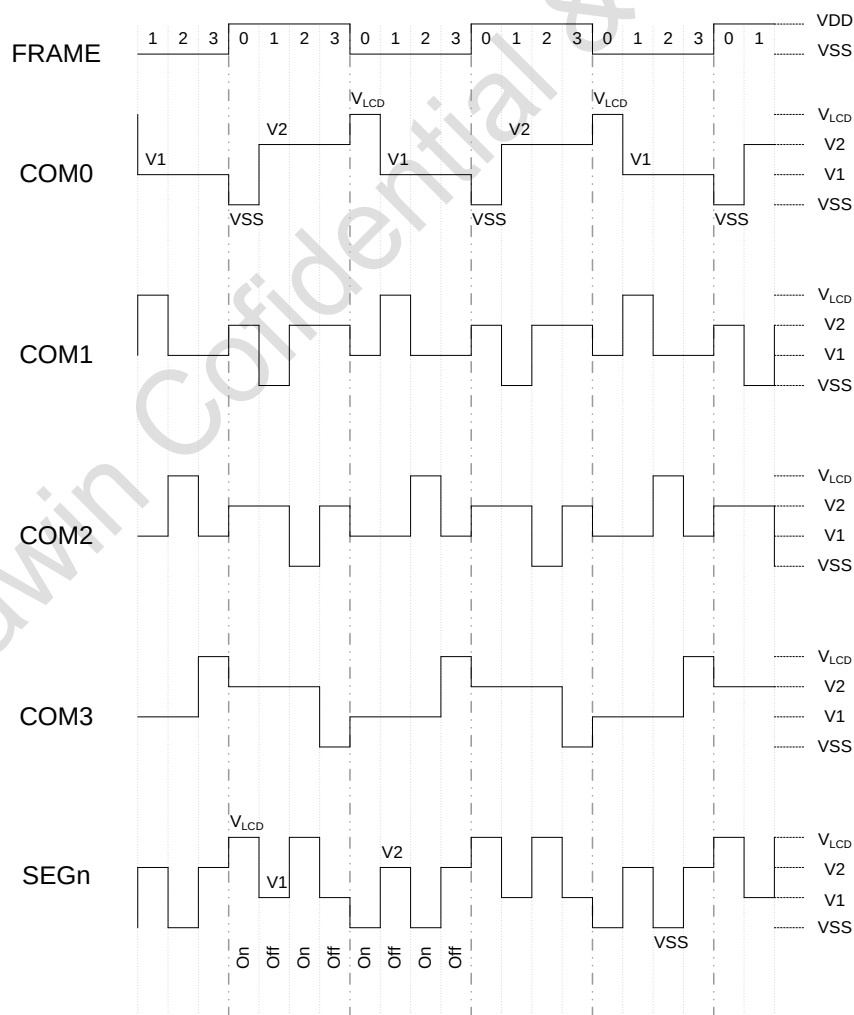
LCD Segment 输出或输入输出(I/O)选择:

功能									
SSEL2	SSEL1	SSEL0	COM3~0 SEG1~0	SEG3~2	SEG5~4	SEG			
						6	7	8	9
0	0	0	P3[5:2] P3[0:1]	P2[6:7]	P2[4:5]	P2[3]	P2[2]	P2[1]	P2[0]
0	0	1	COM3~0 SEG1~0	P2[6:7]	P2[4:5]	P2[3]	P2[2]	P2[1]	P2[0]
0	1	0	COM3~0 SEG1~0	SEG3~2	P2[5:4]	P2[3]	P2[2]	P2[1]	P2[0]
0	1	1	COM3~0 SEG1~0	SEG3~2	SEG5~4	P2[3]	P2[2]	P2[1]	P2[0]
1	0	0	COM3~0 SEG1~0	SEG3~2	SEG5~4	SEG6	P2[2]	P2[1]	P2[0]
1	0	1	COM3~0 SEG1~0	SEG3~2	SEG5~4	SEG6	SEG7	P2[1]	P2[0]
1	1	0	COM3~0 SEG1~0	SEG3~2	SEG5~4	SEG6	SEG7	SEG8	P2[0]
1	1	1	COM3~0 SEG1~0	SEG3~2	SEG5~4	SEG6	SEG7	SEG8	SEG9

MG65PG5A08A 的 8 个 LCD 数据 RAM。当 LCD 数据 RAM 的位为“1”，则 LCD 显示。当 LCD 数据 RAM 的位为“0”，则 LCD 不显示。LCD 数据 RAM 的内容通过直接存储空间访问发送至 SEG0 到 SEG9 引脚上。LCD 数据 RAM 和 SEG/COM 引脚的关系如下表所示。

LCD Data RAM	COM _x	Bit 7 SEG	Bit 6 SEG	Bit 5 SEG	Bit 4 SEG	Bit 3 SEG	Bit 2 SEG	Bit 1 SEG	Bit 0 SEG
1000H	COM 0	0/1 (07)	0/1 (06)	0/1 (05)	0/1 (04)	0/1 (03)	0/1 (02)	0/1 (01)	0/1 (00)
1001H		RAM	RAM	RAM	RAM	RAM	RAM	0/1 (09)	0/1 (08)
1002H	COM 1	0/1 (07)	0/1 (06)	0/1 (05)	0/1 (04)	0/1 (03)	0/1 (02)	0/1 (01)	0/1 (00)
1003H		RAM	RAM	RAM	RAM	RAM	RAM	0/1 (09)	0/1 (08)
1004H	COM 2	0/1 (07)	0/1 (06)	0/1 (05)	0/1 (04)	0/1 (03)	0/1 (02)	0/1 (01)	0/1 (00)
1005H		RAM	RAM	RAM	RAM	RAM	RAM	0/1 (09)	0/1 (08)
1006H	COM 3	0/1 (07)	0/1 (06)	0/1 (05)	0/1 (04)	0/1 (03)	0/1 (02)	0/1 (01)	0/1 (00)
1007H		RAM	RAM	RAM	RAM	RAM	RAM	0/1 (09)	0/1 (08)

1/4 duty 1/3 bias



16 模数转换器(ADC)

MG65PG5A08A 有一个 8 通道 12 位的模数转换器(ADC)。模数转换器(ADC)的输入与 P0.0 ~P0.7 共享。

16.1 模数转换器(ADC)控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00F5H	ADC_CTL0	ADEN	MOD_SEL	VREF_SEL	CS3	CS2	CS1	CS0	SOC	-	√
00F6H	ADC_CTL1	-	-	-	CH4_SEL	EN_BUF	EN_AZ	CK1	CK0	-	√
00F5H	ADC_STS	-	-	-	-	-	-	-	RDY	√	-

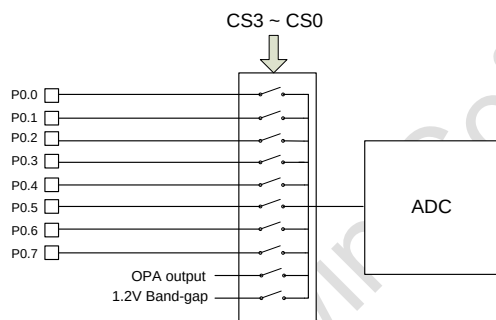
ADEN: “0”: 禁止模数转换(ADC)功能(默认值), “1”:使能模数转换(ADC)功能

MOD_SEL:模数转换(ADC)模式选择 0:单端模式 1:差分模式

VREF_SEL: ADC 参考电压选择。0: VDD 1: 外部参考电压

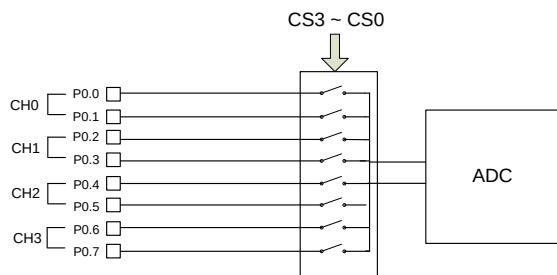
CS3 ~ CS0 模拟通道选择。(单端模式)

CS3	CS2	CS1	CS0	ADC 输入通道
0	0	0	0	VIPA=通道 0 (P0.0), VINA=GND
0	0	0	1	VIPA=通道 1 (P0.1), VINA=GND
0	0	1	0	VIPA=通道 2 (P0.2), VINA=GND
0	0	1	1	VIPA=通道 3 (P0.3), VINA=GND
0	1	0	0	VIPA=通道 4 (P0.4), VINA=GND
0	1	0	1	VIPA=通道 5 (P0.5), VINA=GND
0	1	1	0	VIPA=通道 6 (P0.6), VINA=GND
0	1	1	1	VIPA=通道 7 (P0.7), VINA=GND
1	0	0	0	VIPA=OCP 电路输出, VINA=GND
1	0	0	1	VIPA=1.2V 带隙输出, VINA=GND



CS3 ~ CS0 模拟通道选择。(差分模式)

CS3	CS2	CS1	CS0	ADC 输入通道
0	0	0	X	VIPA=P0.0, VINA=P0.1
0	0	1	X	VIPA=P0.2, VINA=P0.3
0	1	0	X	VIPA=P0.4, VINA=P0.5
0	1	1	X	VIPA=P0.6, VINA=P0.7



SOC:启动模数(A/D)转换。(0→1 = 启动)

CK1 ~ CK0: ADC 时钟选择

CK1	CK0	ADC 时钟输入
0	0	2MHz
0	1	1MHz
1	0	0.5MHz
1	1	32KHz

EN_AZ: ADC 的偏移量控制信号。0: 禁止偏移量输出 1: 使能偏移量输出

EN_BUF: ADC 的输入缓冲控制信号。0: 禁止 1: 使能

CH4_SEL: ADC 通道 4 内部电阻控制。0: 1:1 输入 1: 1/2 偏置输入

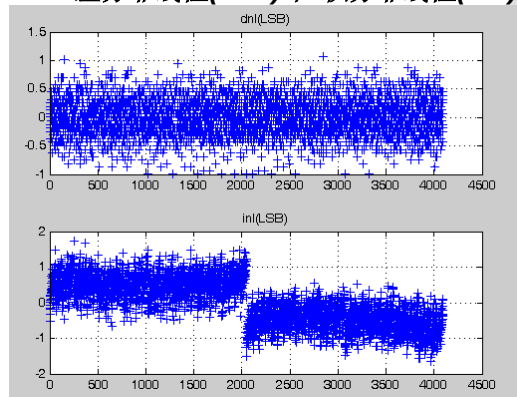
RDY: 由硬件置位并且 ADC_CTL.0 的启动信号清零。ADC_STS.0 设置为“1”意味着模数(A/D)转换完成。

16.2 模数转换器(ADC)数据总线

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00F7H	ADB_L	ADB3	ADB2	ADB1	ADB0					√	-
00F8H	ADB_H	ADB11	ADB10	ADB9	ADB8	ADB7	ADB6	ADB5	ADB4	√	-

在转换完成之后，从 ADB_H 和 ADB_L 可以获取到转换结果数据。

ADC 差分非线性(DNL) 和 积分非线性(INL)

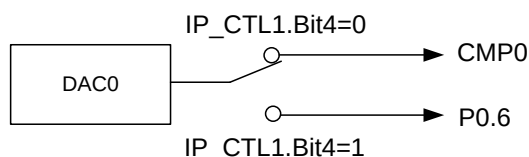


17 数模转换器(DAC)

17.1 数模转换器 0(DAC0)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00C6H	DAC0	DAC07	DAC06	DAC05	DAC04	DAC03	DAC02	DAC01	DAC00	-	√

数模转换器 0(DAC0)提供模拟电压输出。



DAC0 输出电压

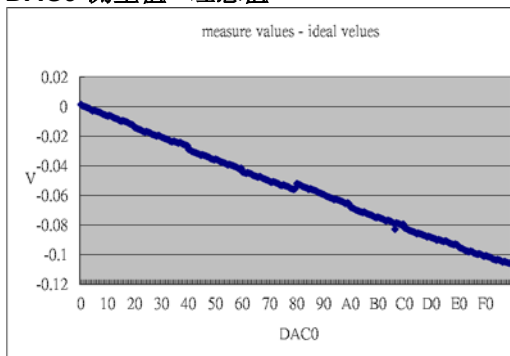
DAC0 值	DAC0 输出
00H	$0.00392 \times \text{ADC}_{\text{VREF}} \times 0$
01H	$0.00392 \times \text{ADC}_{\text{VREF}} \times 1$
02H	$0.00392 \times \text{ADC}_{\text{VREF}} \times 2$
03H	$0.00392 \times \text{ADC}_{\text{VREF}} \times 3$
.....	
FDH	$0.00392 \times \text{ADC}_{\text{VREF}} \times 253$
FEH	$0.00392 \times \text{ADC}_{\text{VREF}} \times 254$
FFH	$0.00392 \times \text{ADC}_{\text{VREF}} \times 255$

例如:

$\text{ADC}_{\text{VREF}} = 5\text{V}$, DAC0 值 = 80H

DAC0 out = $0.00392 \times 5 \times 128 = 2.5088\text{v}$

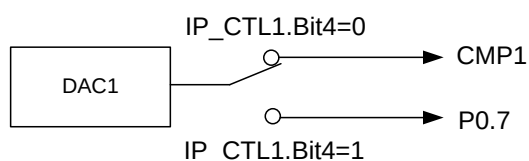
DAC0 测量值-理想值



17.2 模数转换器 1(DAC1)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00C7H	DAC1	DAC17	DAC16	DAC15	DAC14	DAC13	DAC12	DAC11	DAC10	-	√

数模转换器 1(DAC1)提供模拟电压输出。



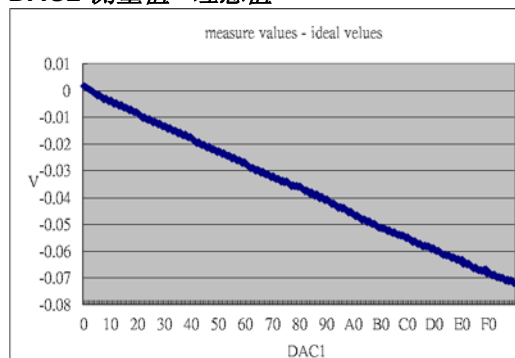
DAC1 输出电压

DAC1 值	DAC1 输出
00H	$0.00392 \times \text{ADC}_{\text{VREF}} \times 0$
01H	$0.00392 \times \text{ADC}_{\text{VREF}} \times 1$
02H	$0.00392 \times \text{ADC}_{\text{VREF}} \times 2$
03H	$0.00392 \times \text{ADC}_{\text{VREF}} \times 3$
.....	
FDH	$0.00392 \times \text{ADC}_{\text{VREF}} \times 253$
FEH	$0.00392 \times \text{ADC}_{\text{VREF}} \times 254$
FFH	$0.00392 \times \text{ADC}_{\text{VREF}} \times 255$

例如:

$\text{ADC}_{\text{VREF}} = 2.5\text{V}$, DAC 值 = 80H

$\text{DAC1 out} = 0.00392 \times 2.5 \times 128 = 1.2224\text{V}$

DAC1 测量值-理想值


17.3 模数转换器(DAC)输出控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00C4H	DACO_CTL	P04HEN	P04LEN	DACMOD1	DACMOD0	-	GAIN1	GAIN0	OPA_MOD	-	√

P04HEN: PWMH 输出控制 (P0.4). 0: 禁止

1: 使能 → 当 P0.4 输入高, P1.7 的 PWM 将被禁止。

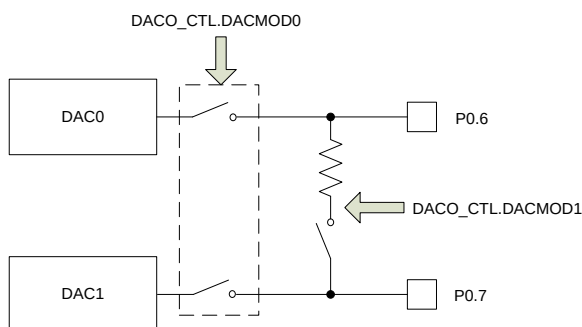
P04LEN: PWML 输出控制 (P0.4). 0: 禁止

1: 使能 → 当 P0.4 输入高, P1.6 的 PWM 将被禁止。

DACMOD1: DAC 模式选择 1。0: 禁止短路 P0.6 和 P0.7 功能, 1: 使能短路 P0.6 和 P0.7 功能

DACMOD0: DAC 模式选择 0。0: 禁止 DAC0 和 DAC1 电压输出功能

1: 使能 DAC0 和 DAC1 电压输出功能



18 运算放大器(OPA)和比较器

18.1 比较器 0 控制寄存器

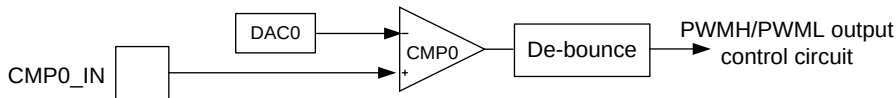
地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00CCH	CMP0_CTL	CMP0_SL	CMP0_RS	OF5	OF4	OF3	OF2	OF1	OF0	-	√
00C3H	PWMO_CTL	IP_PDB1	IP_PDB0	CMP1HEN	CMP1LEN	CMP0HEN	CMP0LEN	CMP1HY	CMP0HY	-	√

CMP0_SL: 比较器 0 功能选择。0: 比较器, 1: 偏移量取消

CMP0_RS: 比较器 0 偏移量参考输入选择。0: 负极输入, 1: 正极输入

OF5 ~ OF0: 比较器 0 偏移量取消输入

CMP0HY: 比较器 0 滞后窗口控制信号。0: 禁止 1: 使能



18.2 比较器 1 控制寄存器

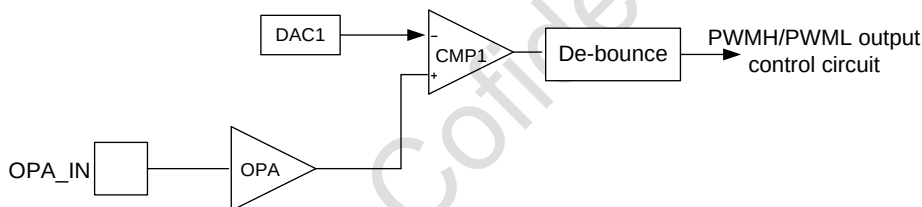
地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00CDH	CMP1_CTL	CMP1_SL	CMP1_RS	OF5	OF4	OF3	OF2	OF1	OF0	-	√
00C3H	PWMO_CTL	IP_PDB1	IP_PDB0	CMP1HEN	CMP1LEN	CMP0HEN	CMP0LEN	CMP1HY	CMP0HY	-	√

CMP1_SL: 比较器 1 功能选择。0: 比较器, 1: 偏移量取消

CMP1_RS: 比较器 1 偏移量参考输入选择。0: 负极输入, 1: 正极输入

OF5 ~ OF0: 比较器 1 偏移量取消输入

CMP1HY: 比较器 1 滞后窗口控制信号。0: 禁止 1: 使能



18.3 运算放大器(OPA)控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00CEH	OPA_CTL	OPA_SL	OPA_RS	OF5	OF4	OF3	OF2	OF1	OF0	-	√
00C4H	DACO_CTL	P04HEN	P04LEN	DACMOD1	DACMOD0	-	GAIN1	GAIN0	OPA_MOD	-	√

OPA_SL: 运算放大器功能选择。0: 运算放大器, 1: 偏移量取消。

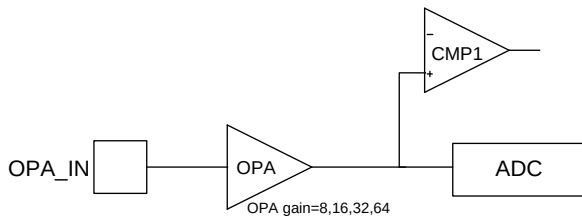
OPA_RS: 运算放大器偏移量参考输入选择。0: 负极输入, 1: 正极输入。

OF5 ~ OF0: 运算放大器偏移量取消输入。

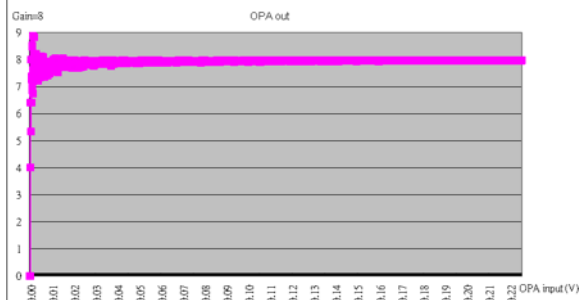
GAIN1~ GAIN0: 运算放大器(OPA)增益选择

GAIN1	GAIN0	增益
0	0	8
0	1	16
1	0	32
1	1	64

OPA_MOD: 运算放大器(OPA)模式选择: 0: 非反转模式 1: 反转模式



运算放大器(OPA)输入偏移量 (设置 OPA 增益 = 8)



18.4 比较器和运算放大器(OPA)输出状态

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00CFH	COPO	-	-	-	-	-	OPAO	CMP1O	CMP0O	√	-

OPAO: 运算放大器数字输出

CMP1O: 比较器 1 输出

CMP0O: 比较器 0 输出

18.5 比较器和运算放大器(OPA)去抖控制寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00CAH	PB_SEL	-	-	-	-	DECMP11	DECMP10	DECMP01	DECMP00	-	√

DECMP11~10: 比较器 1 输出去抖时序选择

00: 无去抖

01: 去抖时间 500ns

10: 去抖时间 1000ns

11: 去抖时间 2000ns

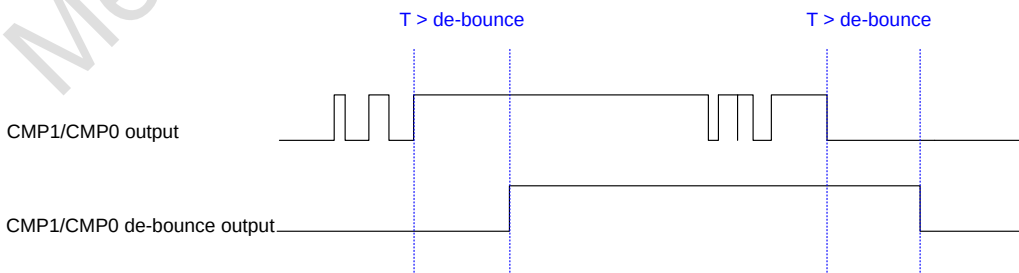
DECMP01~00: 比较器 0 输出去抖时序选择

00: 无去抖

01: 去抖时间 500ns

10: 去抖时间 1000ns

11: 去抖时间 2000ns



19 在应用可编程(IAP)

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R	W
00FBH	IAP_PR	PR7	PR6	PR5	PR4	PR3	PR2	PR1	PR0	-	√

PR7 ~ PR0: 写保护模式。

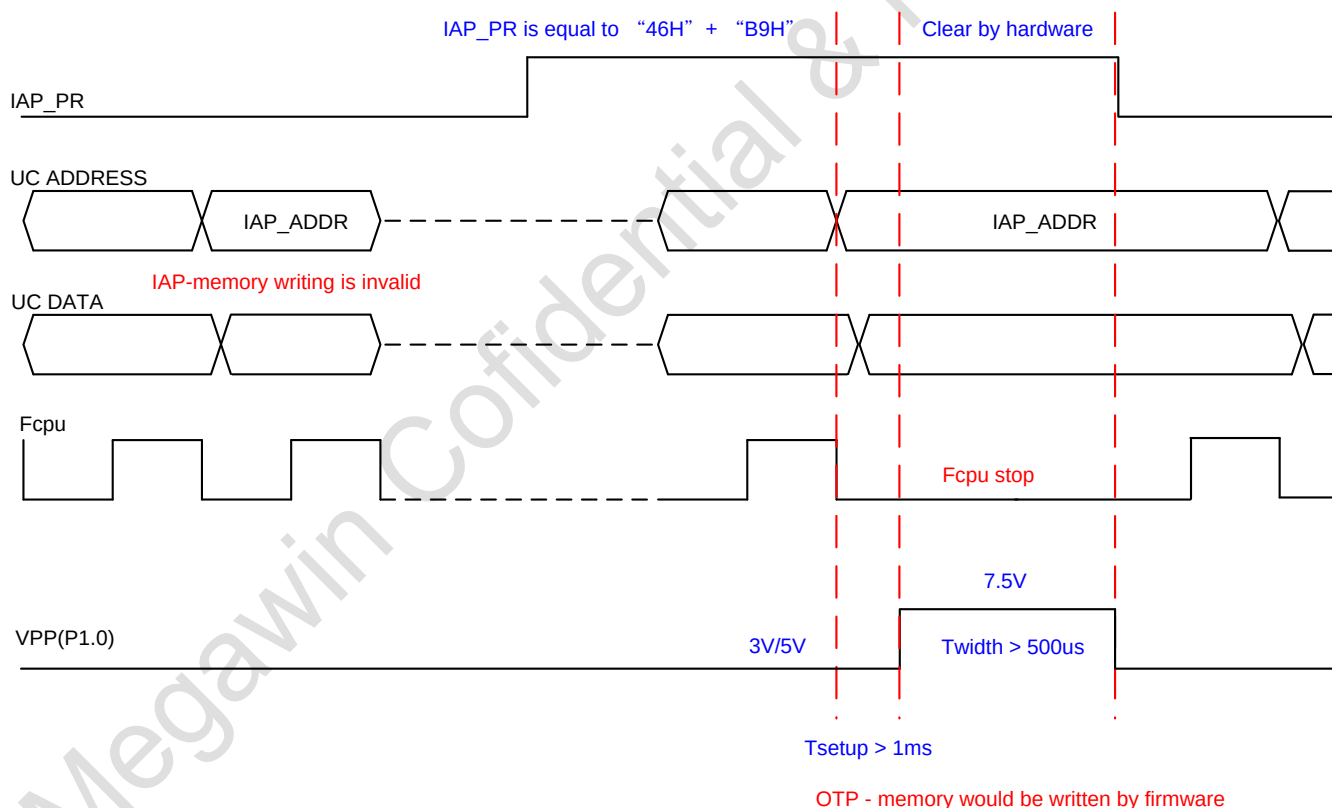
当 IAP_WP 写入“46H”然后写入“B9H”， IAP 存储模块将被硬件写入。

在下一个微控制器写行为或 OTP VPP 下降沿出现，IAP_WP 会被自动清零。

例如：

```

Sei
lda    #78H
sda    cwpr
lda    #80H
sta    SYS_ST    ;;(B7H)
lda    #46H
sta    IAP_PR    ;;(F1h)
lda    #B9H
sta    IAP_PR    ;;(F1h)
lda    #$40      ;;数据被写入 OTP。
sta    $E000     ;;IAP_AREA (E000h ~ FFFFh)
cli
  
```



20 选项寄存器

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	FMOSC1	FMOSC0	LOCK	-	WDT	FSOSC	ENSCK

Bit6 ~ Bit5: 主频 Fmosc 时钟源选择

FMOSC1, FMOSC0

00 (HF16M/1): Fmosc 使用 16MHz 时钟

01 (HF16M /8): Fmosc 使用 2MHz 时钟

10 (HF16M /2): Fmosc 使用 8MHz 时钟

11 (HF16M /4): Fmosc 使用 4MHz 时钟

Bit 4: LOCK: 在线编程(ICP)接口锁位

0: 代码下载被锁。(默认值)

1: 代码下载打开。

Bit2: WDT: 看门狗(WDT)控制位

0 (禁止): 禁止看门狗定时器功能。

1 (使能): 使能看门狗功能。

Bit1: FSOSC: P1.2 和 P1.3 功能选择

0(外部): 外部 32K 晶振。

1(内部): 内部 32K 振荡器。

Bit0: ENSCK: 实时时钟(RTC)功能选择

0(使能): 使能实时时钟(RTC)功能 (32K 晶振振荡器总是打开的)。

1(禁止): 禁止实时时钟(RTC)功能。

21 应用电路

21.1 参考原理图

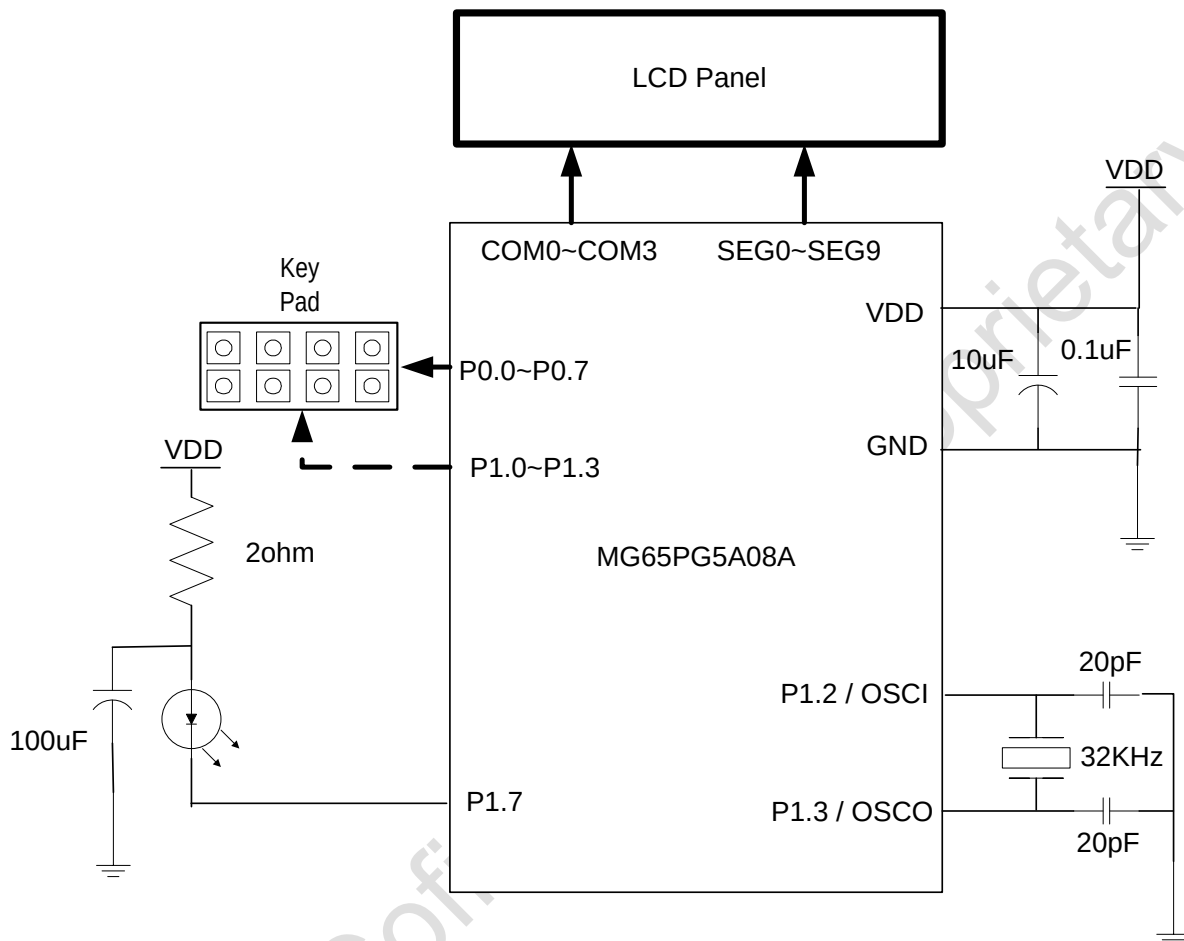


图 20-1 应用电路– LCD 显示控制器

22 电气特性

22.1 绝对最大值

参数	范围	单位
电源到地的电位	VSS-0.3 to VSS+4.0	V
应用输入/输出电压	VSS-0.3 to VDD+0.3	V
工作环境温度	0 to +70	°C
储藏温度	-50 to +125	°C

注意：超出上表的绝对最大值可能损害器件的寿命和可靠性。

22.2 直流特性

(VDD-VSS = 3.0 V, FOSC = 4MHz, Ta = 25° C; 除非特别说明)

参数	符号	条件	最小值	典型值	最大值	单位
输入高电压	VIH	-	0.7 VDD	-	VDD	V
输入低电压	VIL	-	0	-	0.3 VDD	V
工作电流	IOP	双频模式， 无负载，LCD 开 Fcpu=4Mhz		2.0	5.6	mA
暂停模式电流	ISTB1	低速模式，暂停(HALT)， Fcpu=32768Hz ,DIVx INT 关		10		uA
停止模式电流	ISTB2	停止(STOP)模式	-	1		uA
P0 ~ P3 内部上拉电阻	RPH1	VIL = 0V	30K	50K	70K	Ω
/RES 上拉电阻	RRES	VIL = 0V	-	30K	-	Ω
端口 0(P 0)驱动电流	IOH1	VOH = 2.4V, VDD = 3.0V	5	-	-	mA
端口 P0, P2 和 P3 灌电流	IOL1	VOL = 0.4V, VDD = 3.0V	10	-	-	mA
端口 P1.0 ~P1.5 驱动电流	IOH2	VOH = 2.4V, VDD = 3.0V	5	-	-	mA
端口 P1.0 ~P1.5 灌电流	IOL2	VOL = 0.4V, VDD = 3.0V	10	-	-	mA
端口 P1.6 ~P1.7 驱动电流	IOH3	VOH = 2.6V, VDD = 3.0V	50	-	-	mA
		VOH = 4.6V, VDD = 5.0V	100			
端口 P1.6 ~P1.7 灌电流	IOL3	VOL = 0.4V, VDD = 3.0V	50	-	-	mA
		VOL = 0.4V, VDD = 5.0V	100			
COM,SEG 驱动电流	IOH4	VOH = 2.7V, VLCD = 3.0V	0.1	2	-	mA
COM,SEG 灌电流	IOL4	VOL = 0.3V, VLCD = 3.0V	0.3	3	-	mA
低电压检测 0	VLVD0	VDD > 2.4V	-	2.4	-	V
低电压检测 1	VLVD1	VDD > 2.7V	-	2.7	-	V
低电压复位	VLVR	-	-	2.0	-	V

22.3 交流特性

参数	符号	条件	最小值	典型值	最大值	单位
内置 CPU 工作频率	FCPU	内部 OSC, VDD = 3.0V	0.008	4	8	MHz
POR 持续时间	TPOR	FOSC = 4 MHz	-	4	1	mS
系统启动时间	TSst	上电复位		16384		1/FCPU
系统唤醒时间	TSwt	停止(STOP)模式唤醒	256		16384	1/FCPU

22.4 模数转换器(ADC),运算放大器(OPA)和比较器特性

参数	符号	条件	最小值	典型值	最大值	单位
ADC 差分非线性(DNL)	ADC _{DNL}	VDD=5V	-1.5	-	+1.5	LSB
ADC 积分非线性(INL)	ADC _{INL}	VDD=5V	-2	-	+2	LSB
ADC 时钟周期	ADC _{DUTY}	VDD=5V	0.5	-	30	uS
ADC 转换时间	ADC _{TCON}	VDD=5V	-	30	-	Clock
OPA 输入偏移量	OPA _{OFFSET}	VDD=5V	-1	-	+1	mV
CMP0 输入偏移量	CMP0 _{OFFSET}	VDD=5V	-1	-	+1	mV
CMP1 输入偏移量	CMP1 _{OFFSET}	VDD=5V	-1	-	+1	mV

23 版本历史

版本	页数	描述	日期
V1.10		1. 初版	2014/04/21
V2.00		1.更新数据手册到 V2.00	2014/05/20
V2.01		1.修改特性描述	2014/06/16
		2.增加运算放大器(V2.00)，比较器和数模转换(DAC)描述章节	