

## SH66P58

### 6K 一次性编程，液晶驱动和 16 位 $\Sigma$ - $\Delta$ ADC 型 4 位单片机

#### 特性

- 基于SH6610D，液晶驱动和16位ADC型4位单片机
- OTP ROM: 6K X 16位
- RAM: 553 X 4位
  - 57个系统控制寄存器
  - 464个数据存储器
  - 32个LCD数据存储器
- 工作电压:  $V_{DD} = 2.4V - 5.5V$
- 20双向I/O端口
- PORTA - PORTE内建上拉电阻
- 1个8位自动重载定时/计数器
- 时基定时器
- ROM数据读出列表 (RDT)
- 8层堆栈 (包括中断)
- 中断源:
  - 定时器0中断
  - 时基定时器中断
  - 模/数中断
  - 外部中断: PORTB (下降沿)
- 振荡器
  - 晶体振荡器: 32.768kHz
  - PLL振荡器: 4.2MHz
- 指令周期时间 ( $4/f_{osc}$ )
- LCD驱动器:
  - 32 SEG X 4 COM (1/4占空比, 1/3偏置)
- 16位  $\Sigma$ - $\Delta$ 模/数转换器 (ADC)
- 内建可编程增益放大器 (PGA)
- 内建蜂鸣音发生器
- 内建升压及稳压电路
- 预热计数器
- 复位
  - 内建看门狗定时器 (WDT) (代码选项)
  - 内建上电复位 (POR)
  - 内建低电压复位 (LVR) (代码选项)
- 内建低电压检测功能
- 两种低功耗工作模式: HALT和STOP
- 提供裸片形式

#### 概述

SH66P58 是一种先进的 CMOS 4 位单片机。该器件集成了 SH6610D CPU 内核，RAM，ROM，定时器，LCD 驱动器，I/O 端口，16 位  $\Sigma$ - $\Delta$ ADC，升压及稳压电路，蜂鸣音发生器，PWM，内建 PLL，看门狗定时器，低电压复位和低电压检测电路。SH66P58 适应于高精度称重应用。





## 焊垫描述

焊垫编号	焊垫命名	焊垫性质	说明
1	CP	I	电容正端，提供给升压电路
2	PVDD	P	升压电路输入电压
3	CN	I	电容负端，提供给升压电路
4	PGND	P	升压和稳压电路地
5	VDD	P	数字电源引脚
6	VLCD	I	LCD 电源输入引脚
10 - 7	COM4 - 1	O	LCD COM 输出信号
38 - 11	SEG28 - 1	O	LCD SEG 输出信号
39	PE.0 /SEG29	I/O O	可编程 I/O LCD SEG输出信号
40	PE.1 /SEG30	I/O O	可编程 I/O LCD SEG输出信号
41	PE.2 /SEG31	I/O O	可编程 I/O LCD SEG输出信号
42	PE.3 /SEG32	I/O O	可编程 I/O LCD SEG输出信号
43	PD.0 /PWM0	I/O O	可编程 I/O PWM0输出
44	PD.1 /PWM1	I/O O	可编程 I/O PWM1输出
46 - 45	PD.3 - 2	I/O	可编程 I/O
50 - 47	PC.3 - 0	I/O	可编程 I/O
54 - 51	PB.3 - 0	I/O	可编程 I/O
55	VDD	P	数字电源引脚
56	OSCI	I	时钟输入引脚
57	OSCO	O	时钟输出引脚
58	GND	P	数字电源地
59	PLL	I	内建 PLL 电容引脚（对地接 1000pF 电容）
60	RESET	I	复位引脚
61	PA.0 /ALM	I/O O	可编程 I/O 蜂鸣音输出引脚
62	PA.1 /T0	I/O I	可编程 I/O 定时器/计数器 0 外部信号输入引脚
63	PA.2 /AIN3- /AN5	I/O I I	可编程 I/O PGA 差分 3 负端输入 PGA 单端输入 5
64	PA.3 /AIN3+ /AN4	I/O I I	可编程 I/O PGA 差分 3 正端输入 PGA 单端输入 4
65	NC		



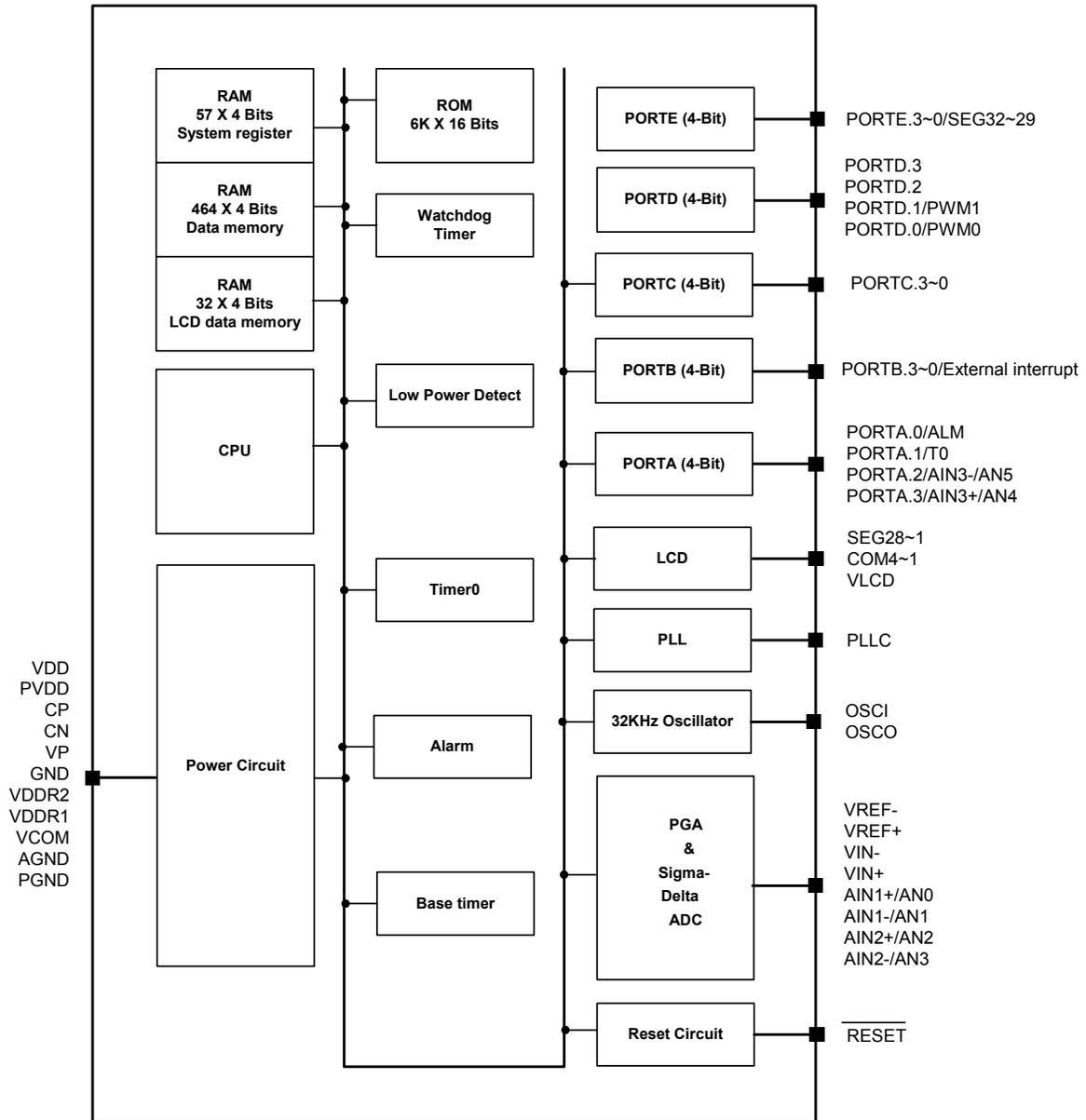
## 焊垫说明 (续)

焊垫编号	焊垫命名	焊垫性质	说明
66	VREF+OUT	O	ADC 参考电压正端输出脚, 必须和焊垫 67 打在一起
67	VREF+IN	I	ADC 参考电压正端输入脚, 必须和焊垫 66 打在一起
68	VREF-OUT	O	ADC 参考电压负端输出脚, 必须和焊垫 69 打在一起
69	VREF-IN	I	ADC 参考电压负端输入脚, 必须和焊垫 68 打在一起
70	VCOMIN	P	单端增益放大器负端输入电压, 必须和焊垫 71 打在一起
71	VCOMOUT	P	单端增益放大器负端输入电压, 必须和焊垫 70 打在一起
72	VIN-	I	ADC 差分负端输入
73	VIN+	I	ADC 差分正端输入 (VIN+和 VIN-之间需要接一只 104 电容)
74	AIN2- /AN3	I I	PGA 差分 2 负端输入 PGA 单端输入 3
75	AIN2+ /AN2	I I	PGA 差分 2 正端输入 PGA 单端输入 2
76	AIN1- /AN1	I I	PGA 差分 1 负端输入 PGA 单端输入 3
77	AIN1+ /AN0	I I	PGA 差分 1 正端输入 PGA 单端输入 0
78	AGND2	P	模拟电源地, 必须和焊垫 10 和焊垫 11 打在一起
79	AGND1	P	模拟电源地, 必须和焊垫 9 和焊垫 11 打在一起
80	AGND0	P	模拟电源地, 必须和焊垫 9 和焊垫 10 打在一起
81	VDDR1IN	P	稳压 1 输入, 必须和焊垫 13 打在一起
82	VDDR1OUT	P	稳压 1 输出, 必须和焊垫 12 打在一起
83	VDDR2IN	P	稳压 2 输入, 必须和焊垫 15 打在一起
84	VDDR2OUT	P	稳压 2 输出, 必须和焊垫 14 打在一起
85	VP	P	升压输出引脚

其中, I: 输入; O: 输出; P: 电源; Z: 高阻



方框图





## 功能说明

### 1. CPU

CPU 包含以下功能模块：程序计数器（PC），算术逻辑单元（ALU），进位标志（CY），累加器，查表寄存器，数据指针（INX, DPH, DPM 和 DPL）和堆栈。

#### 1.1. PC

程序计数器用于寻址程序 ROM。该计数器有 12 位：页寄存器（PC11），和循环递增计数器（PC10, PC9, PC8, PC7, PC6, PC5, PC4, PC3, PC2, PC1, PC0）。

程序计数器装入与该条指令相关的数据。对于目标地址大于 2K 的 ROM 空间，可通过无条件跳转指令（JMP）中设置页寄存器位的值实现跳转。

程序计数器只能寻址 4K 程序 ROM 空间(参考 ROM 说明)。

#### 1.2. ALU 和 CY

ALU 执行算术运算和逻辑操作。ALU 具有下述功能：

二进制加法/减法（ADC, ADCM, ADD, ADDM, SBC, SBCM, SUB, SUBM, ADI, ADIM, SBI, SBIM）

加法/减法的十进制调整（DAA, DAS）

逻辑操作（AND, ANDM, EOR, EORM, OR, ORM, ANDIM, EORIM, ORIM）

条件跳转（BA0, BA1, BA2, BA3, BAZ, BNZ, BC, BNC）

逻辑移位（SHR）

进位标志（CY）记录 ALU 算术运算操作中的进位/借位状态。在中断或子程序调用过程中，进位标志被压入堆栈中并于执行 RTNI 指令时由堆栈中弹出。它不受 RTNW 指令的影响。

#### 1.3. 累加器（AC）

累加器是一个4位寄存器，用于保存算术逻辑单元的运算结果。它和 ALU 一起，完成与系统寄存器数据存储器之间的数据传送。

### 2. RAM

内建 RAM 由通用数据存储器 and 系统寄存器组成。由于 RAM 的静态特性，数据存储器能在 CPU 进入 STOP 或者 HALT 方式后保持其中的数据不变。

#### 2.1. RAM 寻址

用一条指令能直接访问数据存储器 and 系统寄存器。下列为存储器空间分配：

系统寄存器： \$000 - \$02F (exclude \$01D), \$320 - \$329

数据存储器： \$030 - \$1FF

LCD 数据存储器： \$300 - \$31F

RAM 页切换如下所示：

Bank 0 B = 0	Bank 1 B = 1	Bank 2 B = 2	Bank 3 B = 3	Bank 4 B = 4	Bank 5 B = 5	Bank 6 B = 6	Bank 7 B = 7
\$020 - \$07F	\$080 - \$0FF	\$100 - \$17F	\$180 - \$1FF	\$200 - \$27F	\$280 - \$2FF	\$300 - \$37F	\$380 - \$3FF

其中，B在指令集中代表RAM页

#### 1.4. 查表寄存器（TBR）

通过查表指令（TJMP）和常数返回指令（RTNW）可以实现读取保存在程序存储器中的表格数据。查表指令执行时，查表寄存器 TBR 和 AC 中存放的是待读取 ROM 的低 8 位地址。TJMP 指令指向的 ROM 地址为 ((PC11 - PC8) X (2<sup>8</sup>) + (TBR, AC))。由 RTNW 指令将查表所得值返回至 (TBR, AC) 中。表格数据的第 7 位至第 4 位存放在 TBR 中，第 3 位至第 0 位存放在 AC 中。

#### 1.5. 数据指针

数据指针能直接寻址数据存储器。指针地址储存在寄存器 DPH (3 位)，DPM (3 位) 和 DPL (4 位)。最大寻址范围为 3FFH。通过索引寄存器 (INX)，可以读写由 DPH, DPM 和 DPL 指定的数据存储器。

#### 1.6. 堆栈

堆栈是一组寄存器，在每次子程序调用或中断时能顺序保存 CY 和 PC (11-0) 中的值，最高位保存 CY 值。其结构为 13 位 X 8 层。当遇到返回指令（RTNI/RTNW）时，堆栈中的内容将按照顺序返回到 PC 中。堆栈中的数据按照先进后出的方式处理。

#### 注意：

堆栈嵌套包括子程序调用和中断请求子程序调用，其最大值为 8 层。如果程序调用和中断请求的数量超过 8 层，堆栈底部将溢出，程序将无法正常运行。



2.2. 系统寄存器配置

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$00	IEADC	IET0	IEBT	IEP	读/写	中断允许标志寄存器
\$01	IRQADC	IRQT0	IRQBT	IRQP	读/写	中断请求标志寄存器
\$02	-	T0S	T0M.1	T0M.0	读/写	第 2-0 位: 定时器 0 模式寄存器 第 3 位: T0 信号源选择寄存器
\$03	T0L.3	T0L.2	T0L.1	T0L.0	读/写	定时器 0 载入/计数器低位寄存器
\$04	T0H.3	T0H.2	T0H.1	T0H.0	读/写	定时器 0 载入/计数器高位寄存器
\$05	BTM1	BTM0	BTEN	-	读/写	第 1 位: 时基控制位 第 3-2 位: 时基溢出频率设置寄存器
\$06	FCT	FSTEN	RLCD0	-	读/写	LCD 控制寄存器
\$07	LCDON	O/S2	O/S1	O/S0	读/写	第 2-0 位: LCD 端口共用控制寄存器 第 3 位: LCD 打开设置寄存器
\$08	PA.3	PA.2	PA.1	PA.0	读/写	PORTA 数据寄存器
\$09	PB.3	PB.2	PB.1	PB.0	读/写	PORTB 数据寄存器
\$0A	PC.3	PC.2	PC.1	PC.0	读/写	PORTC 数据寄存器
\$0B	PD.3	PD.2	PD.1	PD.0	读/写	PORTD 数据寄存器
\$0C	PE.3	PE.2	PE.1	PE.0	读/写	PORTE 数据寄存器
\$0D	PULLEN	ALMF1	ALMF0	ALMEN	读/写	第 0 位: ALM 控制寄存器 第 2-1 位: ALM 输出频率选择寄存器 第 3 位: 端口上拉控制寄存器
\$0E	TBR.3	TBR.2	TBR.1	TBR.0	读/写	查表寄存器
\$0F	INX.3	INX.2	INX.1	INX.0	读/写	索引寄存器
\$10	DPL.3	DPL.2	DPL.1	DPL.0	读/写	索引地址低位寄存器
\$11	-	DPM.2	DPM.1	DPM.0	读/写	索引地址中位寄存器
\$12	-	DPH.2	DPH.1	DPH.0	读/写	索引地址高位寄存器
\$13	FS1	FS0	OXS	PLLON	读/写	振荡器控制寄存器
\$14	RDT.3	RDT.2	RDT.1	RDT.0	读/写	ROM 数据列表地址/数据寄存器
\$15	RDT.7	RDT.6	RDT.5	RDT.4	读/写	ROM 数据列表地址/数据寄存器
\$16	RDT.11	RDT.10	RDT.9	RDT.8	读/写	ROM 数据列表地址/数据寄存器
\$17	RDT.15	RDT.14	RDT.13	RDT.12	读/写	ROM 数据列表地址/数据寄存器
\$18	PACR.3	PACR.2	PACR.1	PACR.0	读/写	PORTA 输入/输出控制寄存器
\$19	PBCR.3	PBCR.2	PBCR.1	PBCR.0	读/写	PORTB 输入/输出控制寄存器
\$1A	PCCR.3	PCCR.2	PCCR.1	PCCR.0	读/写	PORTC 输入/输出控制寄存器
\$1B	PDCR.3	PDCR.2	PDCR.1	PDCR.0	读/写	PORTD 输入/输出控制寄存器
\$1C	PECR.3	PECR.2	PECR.1	PECR.0	读/写	PORTE 输入/输出控制寄存器
\$1D	-	-	-	-	-	保留
\$1E	WDTF	WDT.2	WDT.1	WDT.0	读/写 只读	第 2-0 位: 看门狗定时器控制寄存器 第 3 位: 看门狗定时器溢出标志寄存器
\$1F	-	-	-	BNK0	读/写	ROM 页寄存器



系统寄存器配置(续)

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$20	ADCEN	-	REFS1	REFS0	读/写	第 1-0 位: ADC 参考电压选择位 第 3 位: ADC 开/关控制位
\$21	PGAEN	GAIN2	GAIN1	GAIN0	读/写	第 2-0 位: PGA 增益选择位 第 3 位: PGA 开/关控制位
\$22	-	CHOPEN	CHOPCLK1	CHOPCLK0	读/写	第 1-0 位: 斩波时钟选择位 第 2 位: 斩波开/关控制位
\$23	CHS3	CHS2	CHS1	CHS0	读/写	PGA 和 ADC 通道选择寄存器
\$24	TADC3	TADC2	TADC1	TADC0	读/写	ADC 时钟低位寄存器
\$25	TADC7	TADC6	TADC5	TADC4	读/写	ADC 时钟高位寄存器
\$26	-	-	AN5S	AN4S	读/写	第 1-0 位: ADC 和端口复用寄存器
\$27	D3	D2	D1	D0	只读	ADC 数据寄存器 0
\$28	D7	D6	D5	D4	只读	ADC 数据寄存器 1
\$29	D11	D10	D9	D8	只读	ADC 数据寄存器 2
\$2A	D15	D14	D13	D12	只读	ADC 数据寄存器 3
\$2B	LPDEN	LPDF	LPDS1	LPDS0	读/写 只读	第 1-0 位: 检测电压选择位 第 2 位: LPD 标志位 第 3 位: LPD 打开控制位
\$2C	PMOD	PCLK	PAUTOEN	PEN	读/写	第 0 位: 升压控制位 第 2-1 位: 升压时钟选择位 第 3 位: 升压模式选择位
\$2D	-	R2S	REN2	REN1	读/写	第 0 位: 稳压 1 控制位 第 2-1 位: 稳压 2 控制位
\$2E	PIEN.3	PIEN.2	PIEN.1	PIEN.0	读/写	PORTB 中断允许标志寄存器
\$2F	PIF.3	PIF.2	PIF.1	PIF.0	读/写	PORTB 中断请求标志寄存器
\$320	PWM0S	T0CK1	T0CK0	PWM0_EN	读/写	第 0 位: PWM0 输出允许设置寄存器 第 2-1 位: PWM0 时钟选择寄存器 第 3 位: PWM0 占空比的输出模式设置寄存器
\$321	PWM1S	T1CK1	T1CK0	PWM1_EN	读/写	第 0 位: PWM1 输出允许设置寄存器 第 2-1 位: PWM1 时钟选择寄存器 第 3 位: PWM1 占空比的输出模式设置寄存器
\$322	PP0.3	PP0.2	PP0.1	PP0.0	读/写	PWM0 周期低 4 位寄存器
\$323	PP0.7	PP0.6	PP0.5	PP0.4	读/写	PWM0 周期高 4 位寄存器
\$324	PD0.3	PD0.2	PD0.1	PD0.0	读/写	PWM0 占空比低 4 位寄存器
\$325	PD0.7	PD0.6	PD0.5	PD0.4	读/写	PWM0 占空比高 4 位寄存器
\$326	PP1.3	PP1.2	PP1.1	PP1.0	读/写	PWM1 周期低 4 位寄存器
\$327	PP1.7	PP1.6	PP1.5	PP1.4	读/写	PWM1 周期高 4 位寄存器
\$328	PD1.3	PD1.2	PD1.1	PD1.0	读/写	PWM1 占空比低 4 位寄存器
\$329	PD1.7	PD1.6	PD1.5	PD1.4	读/写	PWM1 占空比高 4 位寄存器



### 3. ROM

ROM 能寻址 6144 X 16 位程序空间，地址由\$000 到\$17FF。

#### 3.1. 矢量地址区 (\$000 到\$004)

程序顺序执行。从地址\$000到\$004的区域是为特殊中断服务程序保留的，作为中断服务的入口地址。

地址	指令	说明
000H	JMP*	跳转至 RESET 服务程序
001H	JMP*	跳转至 ADC 中断服务程序
002H	JMP*	跳转至定时器 0 中断服务程序
003H	JMP*	跳转至时基中断服务程序
004H	JMP*	跳转至端口中断服务程序

\*JMP指令能由任意指令代替。

#### 3.2. ROM 页切换

程序计数器 (PC11 - PC0) 只能寻址 4K 的 ROM 空间。页切换技术用于扩展 CPU 寻址范围。CPU 地址空间的前 2K 映像为 ROM 空间的低 2K (BANK0)。CPU 地址空间的高 2K 映像为 2 个页。(BNK.0 = \$00 - \$01 这 2 个页位于高位 4K ROM 中。)

页切换如下所示：

CPU 地址	ROM 空间	
	BNK = \$00	BNK = \$01
低 2K 地址	\$0000 - \$07FF (BANK 0)	\$0000 - \$07FF (BANK 0)
高 2K 地址	\$0800 - \$0FFF (BANK 1)	\$1000 - \$17FF (BANK 2)



4. 初始状态

4.1. 系统寄存器初始状态

地址	第 3 位	第 2 位	第 1 位	第 0 位	上电复位/Reset 引脚复位 /低电压复位	WDT 复位
\$00	IEADC	IET0	IEBT	IEP	0000	0000
\$01	IRQADC	IRQT0	IRQBT	IRQP	0000	0000
\$02	-	T0S	T0M.1	T0M.0	0000	-uuu
\$03	T0L.3	T0L.2	T0L.1	T0L.0	xxxx	uuuu
\$04	T0H.3	T0H.2	T0H.1	T0H.0	xxxx	uuuu
\$05	BTM1	BTM0	BTEN	-	000-	uuu-
\$06	FCT	FSTEN	RLCD0	-	000-	000-
\$07	LCDON	O/S2	O/S1	O/S0	0000	0000
\$08	PA.3	PA.2	PA.1	PA.0	0000	0000
\$09	PB.3	PB.2	PB.1	PB.0	0000	0000
\$0A	PC.3	PC.2	PC.1	PC.0	0000	0000
\$0B	PD.3	PD.2	PD.1	PD.0	0000	0000
\$0C	PE3	PE.2	PE.1	PE.0	0000	0000
\$0D	PULLEN	ALRMF1	ALRMF0	ALRMEN	0000	0000
\$0E	TBR.3	TBR.2	TBR.1	TBR.0	xxxx	uuuu
\$0F	INX.3	INX.2	INX.1	INX.0	xxxx	uuuu
\$10	DPL.3	DPL.2	DPL.1	DPL.0	xxxx	uuuu
\$11	-	DPM.2	DPM.1	DPM.0	-xxx	-uuu
\$12	-	DPH.2	DPH.1	DPH.0	-xxx	-uuu
\$13	FS1	FS0	OXS	PLLON	0000	0000
\$14	RDT.3	RDT.2	RDT.1	RDT.0	xxxx	uuuu
\$15	RDT.7	RDT.6	RDT.5	RDT.4	xxxx	uuuu
\$16	RDT.11	RDT.10	RDT.9	RDT.8	xxxx	uuuu
\$17	RDT.15	RDT.14	RDT.13	RDT.12	xxxx	uuuu
\$18	PACR.3	PACR.2	PACR.1	PACR.0	0000	0000
\$19	PBCR.3	PBCR.2	PBCR.1	PBCR.0	0000	0000
\$1A	PCCR.3	PCCR.2	PCCR.1	PCCR.0	0000	0000
\$1B	PDCR.3	PDCR.2	PDCR.1	PDCR.0	0000	0000
\$1C	PECR.3	PECR.2	PECR.1	PECR.0	0000	0000
\$1D	-	-	-	-	xxxx	uuuu
\$1E	WDTF	WDT.2	WDT.1	WDT.0	0000	1000
\$1F	-	-	-	BNK0	---0	---0

说明：x = 不定，u = 未更改，- = 未使用，读出值为“0”。



系统寄存器初始状态 (续)

地址	第 3 位	第 2 位	第 1 位	第 0 位	上电复位/Reset 引脚复位 /低电压复位	WDT 复位
\$20	ADCEN	-	REFS1	REFS0	0-00	0-00
\$21	PGAEN	GAIN2	GAIN1	GAIN0	0000	0000
\$22	-	CHOPEN	CHOPCLK1	CHOPCLK0	-000	-000
\$23	CHS3	CHS2	CHS1	CHS0	0000	0000
\$24	TADC3	TADC2	TADC1	TADC0	0000	0000
\$25	TADC7	TADC6	TADC5	TADC4	0000	0000
\$26	-	-	AN5S	AN4S	--00	--00
\$27	D3	D2	D1	D0	0000	0000
\$28	D7	D6	D5	D4	0000	0000
\$29	D11	D10	D9	D8	0000	0000
\$2A	D15	D14	D13	D12	0000	0000
\$2B	LPDEN	LPDF	LPDS1	LPDS0	0000	0000
\$2C	PMOD	PCLK	PAUTOEN	PEN	0000	0000
\$2D	-	R2S	REN2	REN1	-000	-000
\$2E	PIEN.3	PIEN.2	PIEN.1	PIEN.0	0000	0000
\$2F	PIF.3	PIF.2	PIF.1	PIF.0	0000	0000
\$320	PWM0S	T0CK1	T0CK0	PWM0_EN	0000	uuu0
\$321	PWM1S	T1CK1	T1CK0	PWM1_EN	0000	uuu0
\$322	PP0.3	PP0.2	PP0.1	PP0.0	xxxx	uuuu
\$323	PP0.7	PP0.6	PP0.5	PP0.4	xxxx	uuuu
\$324	PD0.3	PD0.2	PD0.1	PD0.0	xxxx	uuuu
\$325	PD0.7	PD0.6	PD0.5	PD0.4	xxxx	uuuu
\$326	PP1.3	PP1.2	PP1.1	PP1.0	xxxx	uuuu
\$327	PP1.7	PP1.6	PP1.5	PP1.4	xxxx	uuuu
\$328	PD1.3	PD1.2	PD1.1	PD1.0	xxxx	uuuu
\$329	PD1.7	PD1.6	PD1.5	PD1.4	xxxx	uuuu

说明: x = 不定, u = 未更改, - = 未使用, 读出值为“0”。

4.2. 其它初始状态

其它	复位后
程序计数器 (PC)	\$000
CY	不定
累加器 (AC)	不定
数据存储器	不定



### 5. 系统时钟和振荡器

振荡器振荡产生的脉冲为 CPU 和片上电路提供系统时钟。

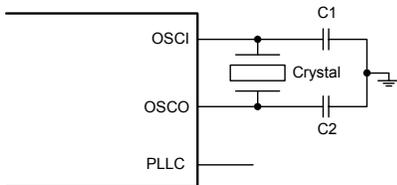
系统时钟  $f_{sys} = f_{osc}/4$

#### 5.1. 指令周期

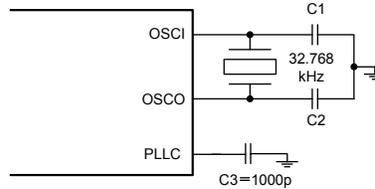
- (1) 对于 32.768kHz 的振荡器，为  $4/32.768\text{kHz} (\approx 122\mu\text{s})$ 。
- (2) 对于 PLL 4.2MHz 的振荡器，为  $4/4.2\text{MHz} (\approx 1\mu\text{s})$ 。

#### 5.2. 振荡器类型

(1) 晶体谐振器：32.768kHz



(2) PLL 谐振器：4.2MHz



#### 5.3. 振荡器的控制

谐振控制寄存器：\$13

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$13	FS1	FS0	OXS	PLLON	读/写	第 0 位：打开/关闭 PLL 第 1 位：时钟源选择 (1: PLL, 0: 32.768kHz) 第 3-2 位：PLL 频率选择
	X	X	0	0	读/写	关闭 PLL，时钟源选择为 32kHz
	X	X	0	1	读/写	打开 PLL，但是时钟源选择为 32kHz
	X	X	1	0	读/写	无效设置
	0	0	1	1	读/写	PLL 为时钟源提供 4.2MHz 时钟
	0	1	1	1	读/写	PLL 为时钟源提供 2.1MHz 时钟
	1	0	1	1	读/写	PLL 为时钟源提供 1.05MHz 时钟
	1	1	1	1	读/写	PLL 为时钟源提供 524kHz 时钟

#### 注意事项 1:

设置 PLL 为系统时钟源：

1. 配置谐振控制寄存器中的 FS1 和 FS0。
2. 设置 PLLON = 1，打开 PLL。
3. 等待至少 2 毫秒。
4. 设置 OXS = 1，选择 PLL 为系统时钟源。

#### 注意事项 2:

可以选择 PLL 时钟使 CPU 高速工作，也可以选择 32.768kHz 时钟使 CPU 低速工作。在复位初始化的开始，32.768kHz 时钟启动振荡器。复位初始化结束后，32.768kHz 时钟立即自动被选择为系统时钟源。

#### 5.4. 谐振器负载电容选择

晶体谐振器			推荐型号	生产厂
频率	C1	C2		
32.768kHz	5 - 12.5pF	5 - 12.5pF	DT 38 (φ 3x8)	KDS

#### 注意事项:

1. 表中负载电容为设计参考数据!
2. 以上电容值可通过谐振器基本的起振和运行测试，并非最优值。
3. 请注意印制板上的杂散电容，用户应在超过应用电压和温度的条件下测试谐振器的性能。

在应用晶体谐振器之前，用户需向谐振器生产厂要求相关应用参数以获得最佳性能。请登陆 <http://www.sinowalth.com> 以取得更多的推荐谐振器生产厂。



6. I/O 端口

SH66P58 提供 20 个可编程双向 I/O 端口，端口数据为寄存器\$08 - \$0C 中。端口控制寄存器（\$18 - \$1C）控制端口为输入或者输出。每个 I/O 端口带有内部上拉电阻。当端口作为输入时，通过\$0D 的 PULLEN 和端口的数据来控制上拉功能。

I/O 寄存器：\$08 - \$0C, \$18 - \$1C

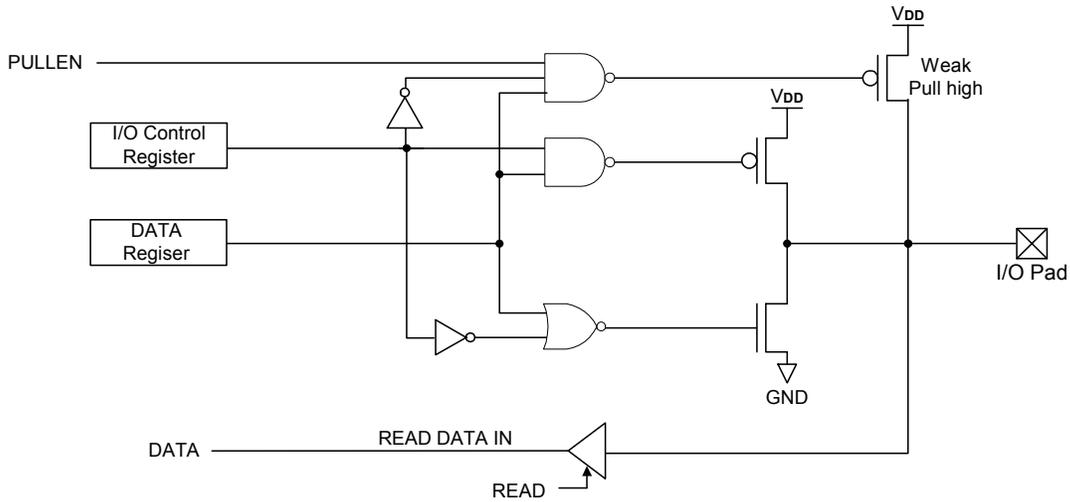
地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$08	PA.3	PA.2	PA.1	PA.0	读/写	PORTA 数据寄存器
\$09	PB.3	PB.2	PB.1	PB.0	读/写	PORTB 数据寄存器
\$0A	PC.3	PC.2	PC.1	PC.0	读/写	PORTC 数据寄存器
\$0B	PD.3	PD.2	PD.1	PD.0	读/写	PORTD 数据寄存器
\$0C	PE.3	PE.2	PE.1	PE.0	读/写	PORTE 数据寄存器
\$18	PACR.3	PACR.2	PACR.1	PACR.0	读/写	PORTA 输入/输出控制寄存器
\$19	PBCR.3	PBCR.2	PBCR.1	PBCR.0	读/写	PORTB 输入/输出控制寄存器
\$1A	PCCR.3	PCCR.2	PCCR.1	PCCR.0	读/写	PORTC 输入/输出控制寄存器
\$1B	PDCR.3	PDCR.2	PDCR.1	PDCR.0	读/写	PORTD 输入/输出控制寄存器
\$1C	PECR.3	PECR.2	PECR.1	PECR.0	读/写	PORTE 输入/输出控制寄存器

PA (/B/C/D/E) CR.n, (n = 0, 1, 2, 3)

0: 设置为输入口。(初始值)

1: 设置为输出口。

I/O 引脚的等效电路





上拉控制寄存器: \$0D

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$0D	PULLEN	ALMF1	ALMF0	ALMEN	读/写	第 0 位: ALM 控制寄存器 第 2-1 位: ALM 输出频率选择寄存器 第 3 位: 端口上拉控制寄存器
	0	x	x	x	读/写	端口上拉禁止
	1	x	x	x	读/写	端口上拉允许

注意事项:

1. 为了打开上拉, 用户必须设置 PULLEN 为 “1”, 且对端口数据寄存器写 “1”。
2. 用户最好不要对这些端口使用 “ANDIM, ORIM, EORIM” 指令, 因为这可能导致端口数据寄存器变为 “0”, 上拉关闭。

端口中断

PORTB 用作端口中断源。由于 PORTB 是位可编程的 I/O, 因此只有 PORTB 用作数字输入端口时, 数字信号引脚上 V<sub>DD</sub> 到 GND 的跳变才能产生端口中断。

中断控制标志映像为系统寄存器的 \$2E - \$2F。通过软件可以读写这两个寄存器。芯片复位后所有标志被清 0。

端口中断可以将 CPU 从 HALT 或者 STOP 模式唤醒。

端口中断允许寄存器: \$2E

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$2E	PIEN.3	PIEN.2	PIEN.1	PIEN.0	读/写	PORTB 中断允许标志寄存器

PIEN.n, (n = 0, 1, 2, 3)

0: 禁止中断。(初始值)

1: 允许中断。

端口中断标志寄存器: \$2F

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$2F	PIF.3	PIF.2	PIF.1	PIF.0	读/写	PORTB 中断请求标志寄存器

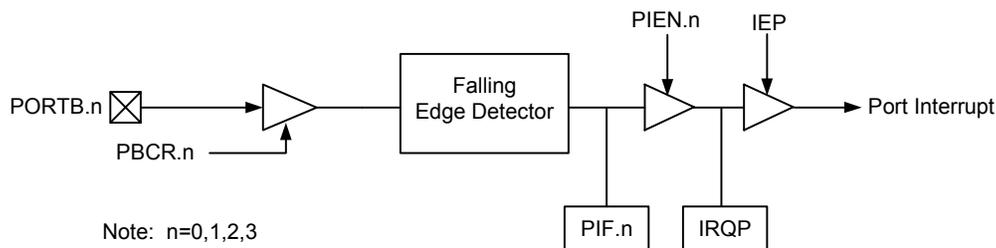
PIF.n, (n = 0, 1, 2, 3)

0: 没有发生中断。(初始值)

1: 已经发生中断。

该寄存器只能清 0。

以下为端口中断功能模块图。仅供参考。



端口中断编程注意事项:

1. PORTB端口的任何一个引脚上电平出现由V<sub>DD</sub>到GND的跳变将置PBIF.x为1, 而跟PORTB其它引脚上的电平无关。
2. 如果PBIEN.x = 1和IEP = 1, PORTB的x输入引脚上电平出现由V<sub>DD</sub>到GND的跳变将产生一个中断请求 (PBIF.x = 1) 并产生中断, 端口中断可以用来将CPU从HALT或者STOP模式唤醒。

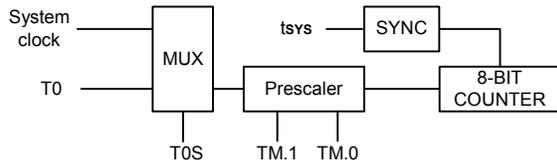


## 7. 定时器

8 位 Timer 有下述特性:

- 8 位递增计数
- 自动重载
- 4 级预分频
- 计数值由\$FF 到\$00 时, 产生溢出中断请求。

Timer 框图:



Timer 功能:

- 可编程定时功能
- 计数值可读

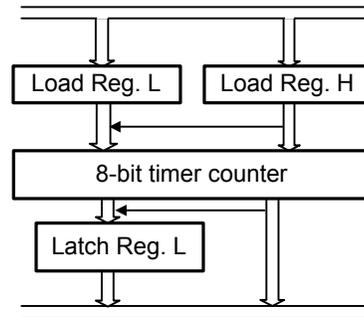
### 7.1. Timer0 结构和操作

Timer0 由一个 8 位只写载入寄存器 (TLOL, TLOH) 和一个 8 位只读计数器 (TCOL, TCOH) 构成。每个计数器由低四位和高四位组成。将数据写入选入寄存器 (TLOL, TLOH) 就可以初始化 Timer。

当高 4 位载入寄存器被写入或 Timer 计数值由\$FF 到\$00 溢出时, Timer 将自动载入预设值。

由于寄存器的高 4 位控制 Timer 的读写操作, 使用中请根据以下步骤操作:

- 写操作
  - 先写低四位,
  - 再写高四位以更新计数器。
- 读操作
  - 先读高四位,
  - 再读低四位。



### 7.2. Timer0 模式寄存器

通过设置 Timer 模式寄存器 (TOM) 可以使 Timer 工作在不同的模式。系统时钟经过预分频器分频后, 进入计数器。Timer 模式寄存器中 TxM.1-0 用于设定分频比。

Timer0 模式寄存器: \$02

TOM.1	TOM.0	预分频器分频比	时钟源
0	0	$/2^6$	系统时钟/T0
0	1	$/2^4$	系统时钟/T0
1	0	$/2^2$	系统时钟/T0
1	1	$/2^0$	系统时钟/T0

Timer 的时钟源则通过下述控制寄存器选择:

系统寄存器: \$02

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$02		TOS	TOM.1	TOM.0	读/写	第 2 位: T0 信号源选择寄存器
		0	X	X	读/写	Timer0 时钟源为系统时钟
		1	X	X	读/写	Timer0 时钟源为 T0 引脚输入脉冲



### 7.3. 外部时钟/事件 T0 作为 Timer0 的时钟源

当外部时钟/事件 T0 输入作为 Timer0 的时钟源时，它与 CPU 的系统时钟进行同步。这个外部信号源必须符合以下条件：

Timer 在一个指令周期中通过系统时钟进行采样，因此对外部时钟高电平（至少  $2 t_{osc}$ ）和低电平（至少  $2 t_{osc}$ ）的要求如下：

$$T0H \text{ (T0 高电平时间)} \geq 2 * t_{osc} + \Delta T$$

$$T0L \text{ (T0 低电平时间)} \geq 2 * t_{osc} + \Delta T \quad ; \Delta T = 20ns$$

当选择其它的分频比时，T0M 通过异步脉冲计数器来分频，且预分频器的输出信号是对称的。

那么：

$$T0 \text{ high time} = T0 \text{ low time} = \frac{N * T0}{2}$$

其中：

T0 = Timer0 输入周期

N = 预分频值

因此，需要满足的条件是：

$$\frac{N * T0}{2} \geq 2 * t_{osc} + \Delta T \quad \text{或} \quad T0 \geq \frac{4 * t_{osc} + 2 * \Delta T}{N}$$

上述条件仅限于 T0 用作 Timer 输入时钟源，对 T0 脉宽没有限制。概括如下：

$$T0 = \text{Timer0period} \geq \frac{4 * t_{osc} + 2 * \Delta T}{N}$$

**8. 时基定时器 (BT)**

SH66P58 有一个时基定时器。它可以通过设置 BT 模式寄存器 (BTM) 的值来产生不同频率的时钟中断。

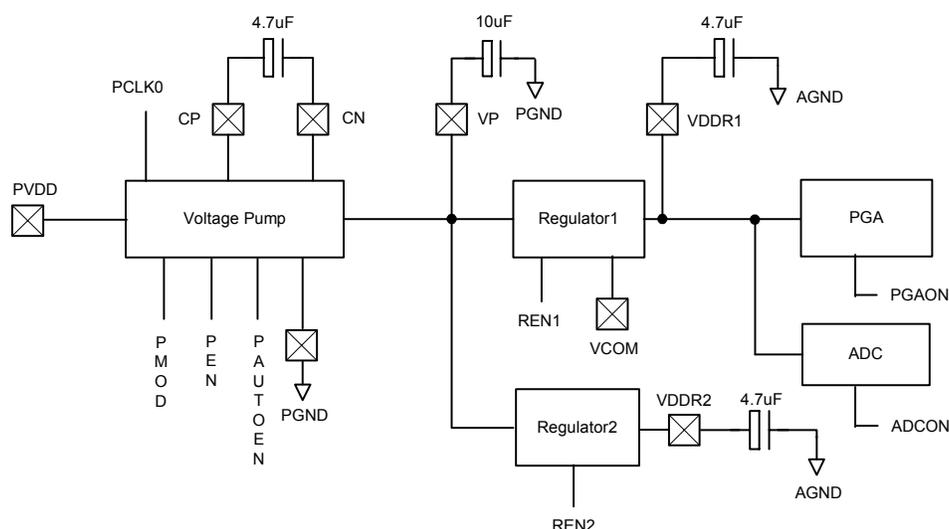
时基定时器模式寄存器: \$05

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$05	BTM1	BTM0	BTEN	-	读/写	第 1 位: BT 控制位 第 3-2 位: BT 溢出频率设置寄存器
	X	X	0	-	读/写	禁止 BT
	0	0	1	-	读/写	允许 BT, 溢出率为 2Hz
	0	1	1	-	读/写	允许 BT, 溢出率为 8Hz
	1	0	1	-	读/写	允许 BT, 溢出率为 32Hz
	1	1	1	-	读/写	允许 BT, 溢出率为 128Hz



## 9. 电源系统

### 9.1. 电源系统框图



### 9.2. 寄存器

#### 升压 (Pump) 控制寄存器: \$2C

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$2C	PMOD	PCLK0	PAUTOEN	PEN	读/写	第 0 位: Pump 控制位 第 2-1 位: Pump 时钟选择位 第 3 位: Pump 模式选择位
	X	X	X	0	读/写	Pump 总是禁止, VP = PVDD
	X	X	0	1	读/写	Pump 总是允许, VP = 2PVDD
	X	X	1	1	读/写	Pump 处于自动模式 如果 PVDD ≥ 3.6V, Pump 总是禁止, VP = PVDD 如果 3.6 > PVDD ≥ 2.75, Pump 间歇地开/关, VP = 5.5V 如果 PVDD < 2.75, Pump 总是允许, VP = 2PVDD
	X	0	X	X	读/写	Pump 时钟 = 2048Hz
	X	1	X	X	读/写	Pump 时钟 = 16384Hz
	0	X	X	X	读/写	Pump 处于正常模式
	1	X	X	X	读/写	Pump 处于低功耗模式

稳压 (Regulator) 1 输出 3.0V, Regulator2 输出 3.0V 或者 1.5V。Regulator1 提供电源给可编程增益放大器 (PGA) 和模数转换器 (ADC)。如果 LCD 需要稳定的显示, 不受 V<sub>DD</sub> 波动的影响, 应该把 VLCD 引脚连接至 VDDR1 引脚。VDDR2 为外部传感器提供稳定的电压, 而且可以被单独关闭以节省功耗。

如果 V<sub>DD</sub> > 3.6V, PEN 不应该被设置为“1”, 因为 3.6V 已经比较高足够为 Regulator 提供电压源。如果升压允许, 所有 VDDR1 和 VDDR2 的电流将增加一倍。

如果 Pump 不带大负载, 可以设置为低功耗模式。例如, Regulator2 关闭, 传感器不需要供电; 同时 Regulator1 允许, PGA 和 ADC 禁止, VLCD 引脚连接 VDDR1, Regulator1 仅仅为 LCD 提供电源。



稳压 (Regulator) 控制寄存器: \$2D

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$2D	-	R2S	REN2	REN1	读/写	第 0 位: Regulator1 控制位 第 2-1 位: Regulator2 控制位
	-	X	X	0	读/写	Regulator1 禁止, VDDR1 = 0, PGA, ADC 无电源供给
	-	X	X	1	读/写	Regulator1 允许, VDDR1 = 3.0V
	-	X	0	X	读/写	Regulator2 禁止, VDDR2 = 0, 传感器无电源供给
	-	0	1	X	读/写	Regulator2 允许, VDDR2 = 3.0V, 给传感器供电
	-	1	1	X	读/写	Regulator2 允许, VDDR2 = 1.5V, 给传感器供电

注意:

1. 如果 Regulator 和 Pump 都允许, REN1 或者 REN2 必须在 PEN 置“1”后延时 10 毫秒才能置“1”。因为 Regulator 电压源来自 Pump, 如果 Pump 还没有稳定 REN 就被置“1”, Pump 可能建立失败。
2. 如果 Pump 允许, 所有 VDDR1 和 VDDR2 的电流将增加一倍。
3. CPU 进入 HALT 或 STOP 模式对 Pump 或 Regulator 没有影响。所以如果需要 CPU 进入 HALT 或 STOP 模式以省电, Pump 和 Regulator 应该通过软件禁止。唤醒后, Pump 和 Regulator 再按照第一条注意允许。

如果 VLCD 连接到 VDDR1 (建议)

PMOD	PAUTOEN	PEN	REN1	REN2	VP	VDDR1	VDDR2	使用条件说明
0	0	0	0		VDD	0		All disable
0	0	0	1		VDD	3.0V		VDD >= 3.6V (PGA、ADC 和 LCD 允许)
0	0	1	1		2VDD	3.0V		VDD < 3.6V (PGA、ADC 和 LCD 允许)
1	0	1	1	0	2VDD	3.0V	0	LCD 允许, PGA 和 ADC 禁止
0	1	1	1		参照注意*	3.0V		所有可能用到的条件

注意\*:

在自动模式:

- 如果 PVDD ≥ 3.6V, Pump 总是禁止, VP = PVDD。  
 如果 3.6 > PVDD ≥ 2.75, Pump 间歇开/关, VP = 5.5V。  
 如果 PVDD < 2.75, Pump 总是允许, VP = 2 PVDD。



### 10. 模数转换器 (ADC)

16 位  $\Sigma$ - $\Delta$  ADC 内建于微控制器中。模拟输入经 PGA (可编程增益放大器) 放大后进入 ADC 模块。有 3 种模拟输入, 即差分输入、单端输入和短路输入。6 个外部的 ADC 通道可被配置为差分输入或单端输入, 也可以配置为短路输入, 用于零点偏移检测。

ADC 转换步骤:

- 设置 ADCON 寄存器中的 VREFS[1:0], 选择参考电压。
- 设置 CHSCON 寄存器中的 CHS[3:0], 选择 ADC 通道。
- 设置 PGACON 寄存器中的 GAIN[2:0], 选择 PGA 增益。
- 设置 TADC 寄存器, 选择 ADC 时钟。
- 设置 CHOPCON 寄存器, 允许斩波 (CHOP) 和选择 CHOP 时钟。
- 设置 ADCEN, 开始 ADC 转换。
- 当一次 ADC 转换完成时, IRQADC 置 “1”, 新的 ADC 数据被存在 ADC 数据寄存器中。如果 IEADC (\$00.3) 被置 “1”, 一个中断请求将发送给 CPU。
- 清零 IRQADC 位, 以产生下一次 ADC 中断。

**注意:**

当 ADCEN 置 “1” 时, 转换总是进行, 当一次转换完成, IRQADC 置 “1”。但是 ADCEN 置 “1” 后的开始 3 次转换结果无效, 因为  $\Sigma$ - $\Delta$  ADC 需要建立时间。

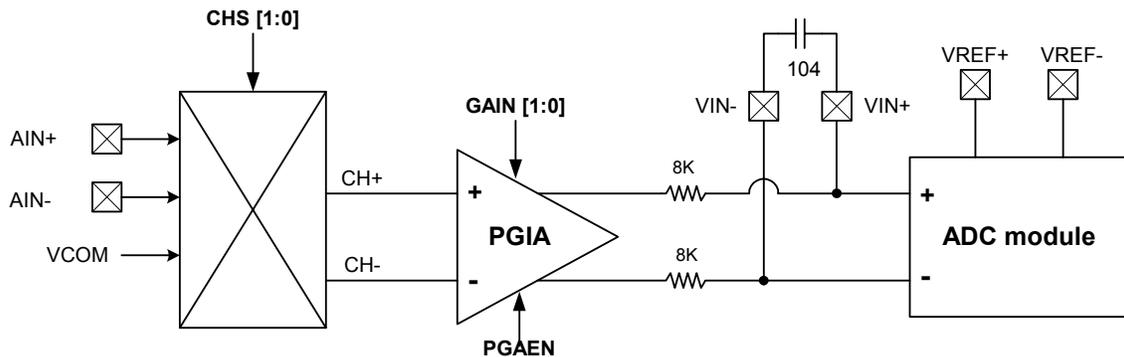
#### 10.1. ADC 控制寄存器: \$20

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$20	ADCEN	-	REFS1	REFS0	读/写	第 1-0 位: ADC 参考电压选择位 第 3 位: ADC 开/关控制位
	0	-	X	X	读/写	ADC 禁止
	1	-	X	X	读/写	ADC 允许
	X	-	1	X	读/写	选择外部 ADC 参考电压 VREF+, VREF-
	X	-	0	0	读/写	内部参考电压为 1.2V, 0.4V (VDDR2 = 3.0V)
	X	-	0	1	读/写	内部参考电压为 0.8V, 0.4V (VDDR2 = 3.0V)

**注意:**

由于内部参考电压由 VDDR2 分压产生, 所以内部参考电压将随 VDDR2 的变化而变化。如果 R2S 置 “1”, VDDR2 输出 1.5V。VREF+ 和 VREF- 将分别变为 0.6, 0.2V (如果 VREFS0 = 0) 或者变为 0.4, 0.2V (如果 VREFS0 = 1)。

#### 10.2. ADC 模块框图





10.3. PGA 增益寄存器: \$21

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$21	PGAEN	GAIN2	GAIN1	GAIN0	读/写	第 2-0 位: PGA 增益选择位 第 3 位: PGA 开/关控制位
	0	X	X	X	读/写	禁止 PGA, 信号不被放大, Gain = 1
	1	0	0	0	读/写	允许 PGA, 信号被放大, Gain = 12.5
	1	0	0	1	读/写	允许 PGA, 信号被放大, Gain = 25
	1	0	1	0	读/写	允许 PGA, 信号被放大, Gain = 50
	1	0	1	1	读/写	允许 PGA, 信号被放大, Gain = 75
	1	1	0	0	读/写	允许 PGA, 信号被放大, Gain = 100
	1	1	0	1	读/写	允许 PGA, 信号被放大, Gain = 125
	1	1	1	0	读/写	允许 PGA, 信号被放大, Gain = 150
	1	1	1	1	读/写	允许 PGA, 信号被放大, Gain = 200

10.4. PGA 斩波器控制寄存器: \$22

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$22	-	CHOPEN	CHOPCLK1	CHOPCLK0	读/写	第 1-0 位: 斩波时钟选择位 第 2 位: 斩波开/关控制位
	-	0	X	X	读/写	禁止斩波器
	-	1	0	0	读/写	允许斩波器, 工作时钟 = fosc/128
	-	1	0	1	读/写	允许斩波器, 工作时钟 = fosc/64
	-	1	1	0	读/写	允许斩波器, 工作时钟 = fosc/32
	-	1	1	1	读/写	允许斩波器, 工作时钟 = fosc/16

注意: 当 chopper 需要被打开始时, 工作时钟应被选择为 2KHz, 即寄存器\$22 被写为 0111B。

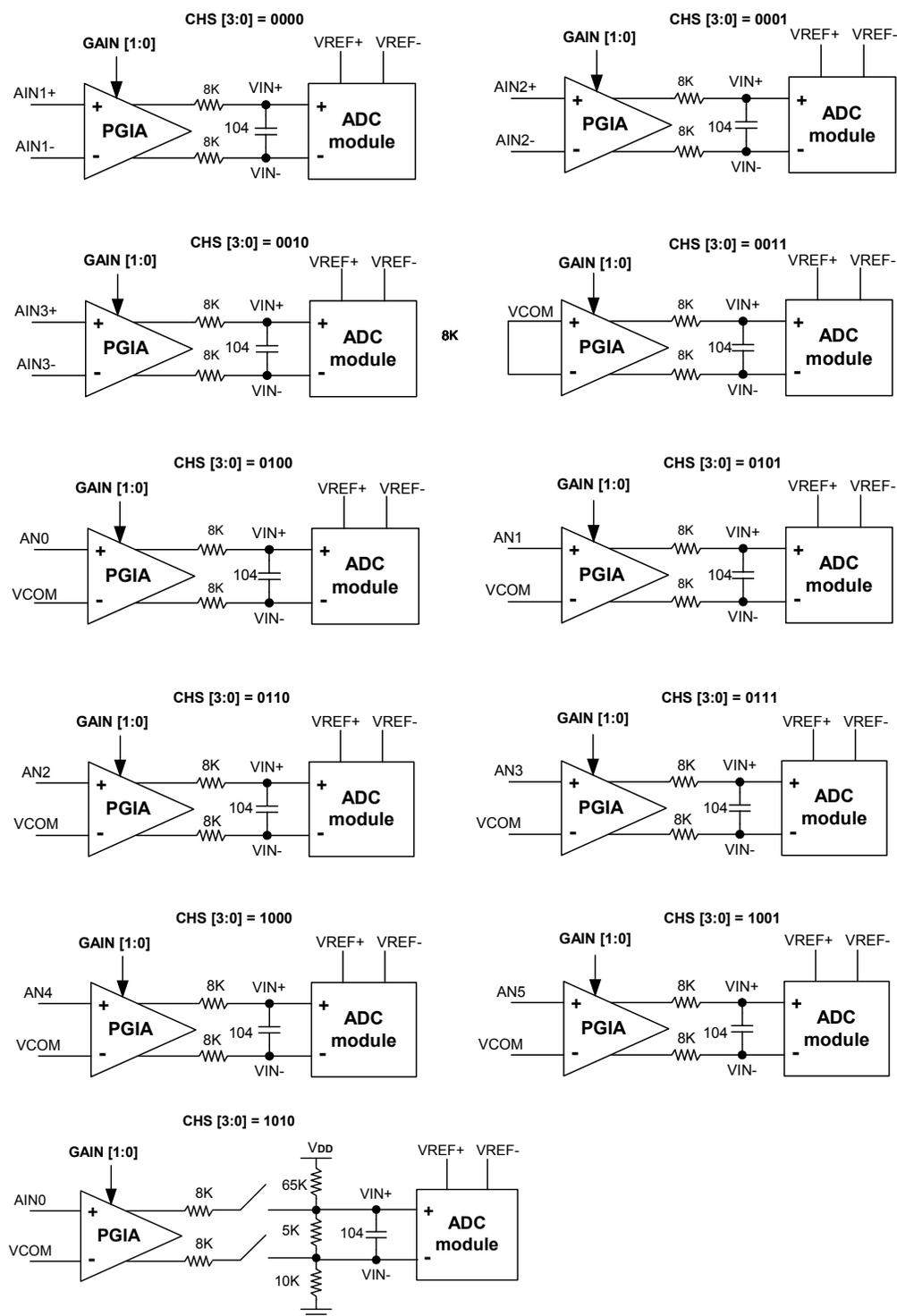
10.5. PGA 和 ADC 通道选择寄存器: \$23

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$23	CHS3	CHS2	CHS1	CHS0	读/写	第 3-0 位: PGA 和 ADC 通道选择
	0	0	0	0	读/写	差分输入, AIN1+, AIN1-
	0	0	0	1	读/写	差分输入, AIN2+, AIN2-
	0	0	1	0	读/写	差分输入, AIN3+, AIN3-
	0	0	1	1	读/写	AIN+, AIN-内部短接 (均连接 VCOM)
	0	1	0	0	读/写	单端输入 0, AN0
	0	1	0	1	读/写	单端输入 1, AN1
	0	1	1	0	读/写	单端输入 2, AN2
	0	1	1	1	读/写	单端输入 3, AN3
	1	0	0	0	读/写	单端输入 4, AN4
	1	0	0	1	读/写	单端输入 5, AN5
	1	0	1	0	读/写	电源检测 Vin+ = 3/16 VDD, Vin- = 2/16VDD
	1	0	1	1	读/写	保留
	1	1	x	x	读/写	保留

注意: 当单端输入被选择时, VCOM 自动连接到 PGA 的负端输入。



ADC 通道选择示意图:



注意: 如果 CHS[3:0] = 1010, ADC 用来测量电源电压。一个 80k 的内部电阻连接 VDD 和 GND, 用以产生采样点。



**10.6. ADC 时钟寄存器: \$24 - \$25**

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$24	TADC3	TADC2	TADC1	TADC0	读/写	ADC 时钟选择寄存器低位
\$25	TADC7	TADC6	TADC5	TADC4	读/写	ADC 时钟选择寄存器高位

TADC 寄存器用来设置 ADC 工作时钟，100kHz 被推荐作为 ADC 工作时钟。当写这两个寄存器时，低位先写，高位后写，写高位的同时计数器初始值被更新。

ADC 时钟和 TADC 的关系为：

ADC 时钟 = 4.2MHz/2 / (256-TADC [7:0])。

ADC 转换速率 = ADC 时钟/4000。

ADC 时钟由 PLL 时钟分频而来，所以如果 ADC 转换被需要，PLL 必须要允许，但是系统时钟可以被选择为 32kHz 或者 PLL。

**注意：**

1. TADC7-0 最大值是 FEH，不能被设置为 FFH。
2. 如果 ADC 转换速率为 25Hz，50Hz 交流电噪声能被滤除。
3. 如果 ADC 转换速率为 20Hz，60Hz 交流电噪声能被滤除。
4. 如果 ADC 转换速率为 10Hz，50Hz 和 60Hz 交流电噪声都能被滤除。

**10.7. ADC 端口共用寄存器: \$26**

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$26	-	-	AN5S	AN4S	读/写	ADC 端口共用寄存器
	-	-	x	0	读/写	PORTA.3用作I/O端口
	-	-	x	1	读/写	PORTA.3用作AN4
	-	-	0	x	读/写	PORTA.2用作I/O端口
	-	-	1	x	读/写	PORTA.2用作AN5

**10.8. ADC 数据寄存器: \$27 - \$2A**

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$27	D3	D2	D1	D0	只读	ADC 数据寄存器 0
\$28	D7	D6	D5	D4	只读	ADC 数据寄存器 1
\$29	D11	D10	D9	D8	只读	ADC 数据寄存器 2
\$2A	D15	D14	D13	D12	只读	ADC 数据寄存器 3

ADC 转换结果被储存在 D15-0中，包括符号位。D15是 ADC 数据的符号位。ADC 转换数据计算公式如下：

ADC 数据 = ((VIN+) - (VIN-)) / ((VREF+)-(VREF-)) X 31250

当(VIN+) - (VIN-)为正时，D15 为 0，表明 ADC 数据为正，ADC 数据为 D15-0。

当(VIN+) - (VIN-)为负时，D15 为 1，表明 ADC 数据为负，ADC 数据为 10000 减去 D15-0。

<b>ADC 数据</b>	7A12H	4000H	0001H	0000H	FFFFH	C000H	85EE
<b>十进制数值</b>	31250	16384	1	0	-1	-16384	-31250

**注意：**

1. 为了获得 ADC 最大范围输出，需要 ADC 输入电压的最大值，即 VIN+与 VIN-的差值接近于但不能超过参考电压 VREF+与 VREF-的差值。
2. 在转换期间清零 ADCEN 位，ADC 转换将停止。
3. 当转换未完全完成时，ADC 数据寄存器不会被更新。
4. 在 HALT 模式 ADC 继续工作，当执行了“STOP”指令后 ADC 将自动停止。
5. 如果 ADC 中断允许，ADC 可以把 CPU 从 HALT 模式中唤醒。



11. LCD Driver

LCD 驱动器包含一个控制器，一个电压发生器，4 COM 驱动器焊垫/引脚和 32 SEG 驱动器[焊垫/引脚]。有一种驱动模式：1/4 占空比和 1/3 偏置电压。控制器由显示数据 RAM 和占空比发生器组成。

LCD SEG29 - 32 可以作为 I/O 端口 (PORTE)，通过系统寄存器\$07 的第 2-0 位选择。如果需要，LCD RAM 可以用作数据存储。

当执行了"STOP"指令后，LCD 驱动器将关闭，但是 LCD RAM 中的数据将保持不变。

当 LCD 驱动器被关闭时，COM 和 SEG 都输出地电平。

11.1. LCD 控制寄存器: \$06

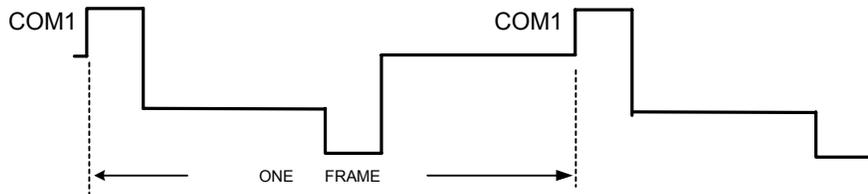
地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$06	FCT	FSTEN	RLCD0	-	读/写	LCD控制寄存器
	X	0	0	X	读/写	R1 = R2 = R3 = 150k (默认)
	X	0	1	X	读/写	R1 = R2 = R3 = 20k
	0	1	X	X	读/写	快速充电模式，快速充电时间 = 1/8 LCD周期
	1	1	X	X	读/写	快速充电模式，快速充电时间 = 1/32 LCD周期

11.2. LCD 端口控制寄存器: \$07

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$07	LCDON	O/S2	O/S1	O/S0	读/写	第2-0位: LCD端口控制寄存器 第3位: LCD打开设置寄存器
	X	X	X	0	读/写	PORTE.0作为I/O端口
	X	X	X	1	读/写	PORTE.0作为LCD SEG 29
	X	X	0	X	读/写	PORTE.1作为I/O端口
	X	X	1	X	读/写	PORTE.1作为LCD SEG 30
	X	0	X	X	读/写	PORTE.2-3作为I/O端口
	X	1	X	X	读/写	PORTE.2-3作为LCD SEG 31-32
	0	X	X	X	读/写	关闭LCD (默认)
	1	X	X	X	读/写	打开LCD

注意:

1. 当使用大负载 LCD 面板时，用户可以通过设置系统寄存器\$06 的值来增加偏置电流以得到更好的 LCD 显示效果。但是当使用阻值较小的偏置电阻时将会消耗更多的电流。
2. 当 CPU 在 STOP 模式时，COMx 和 SEGx 端口输出低电平。可以被按键（端口中断）唤醒。



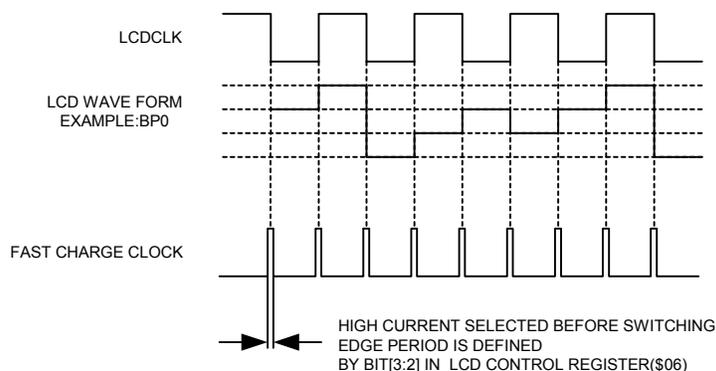
LCD输出帧频



### 11.3. 快速充电模式

每一个偏置电阻（RLCD）的默认值都是 150k。选择 20k 偏置电阻可以得到较好的显示效果，但电流相对会大一些，不适合低功耗的应用。可以设置 LCD 控制寄存器的 RLCD0 位选择 150k 偏置电阻，虽然可以达到较低的功耗，但 LCD 显示效果会变得差一些。

因此，MCU 提供了兼顾低功耗和显示效果的模式：快速充电模式。这种模式在显示数据刷新时刻选择 20k 偏置电阻，提供较大的驱动电流，在数据保持期间选择 150k 偏置电阻，提供较小的驱动电流。设置 FSTEN 位选择此种模式。在每一个 LCD 波形的电平切换边沿，RLCD0 的值选择 20k 偏置电阻，在稳定期间偏置电阻又回到 150k。由 LCD 控制寄存器中的 FCT 位选择充电时间为 LCD 周期的 1/8 或 1/32。

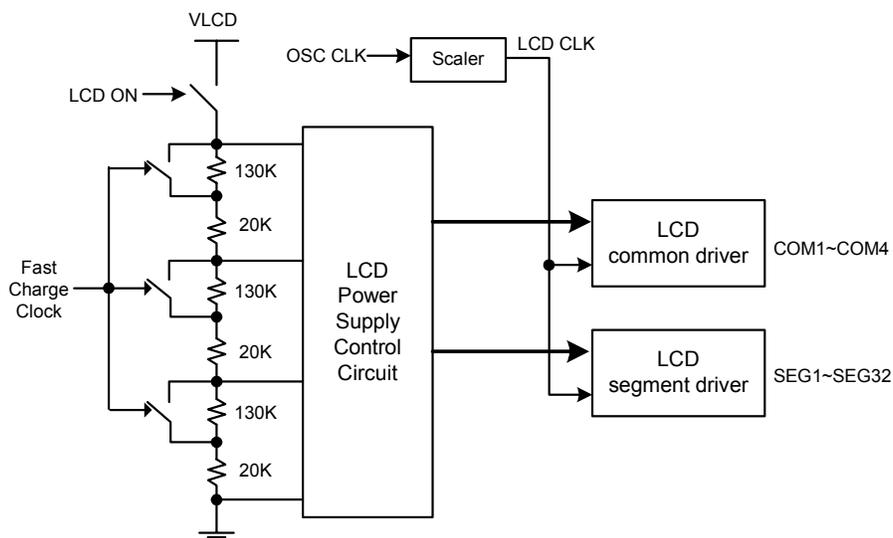


### 11.4. LCD RAM 配置：（SEG 1-32, 1/4 占空比）

地址	第 3 位	第 2 位	第 1 位	第 0 位	地址	第 3 位	第 2 位	第 1 位	第 0 位
	COM4	COM3	COM2	COM1		COM4	COM3	COM2	COM1
300H	SEG1	SEG1	SEG1	SEG1	310H	SEG17	SEG17	SEG17	SEG17
301H	SEG2	SEG2	SEG2	SEG2	311H	SEG18	SEG18	SEG18	SEG18
302H	SEG3	SEG3	SEG3	SEG3	312H	SEG19	SEG19	SEG19	SEG19
303H	SEG4	SEG4	SEG4	SEG4	313H	SEG20	SEG20	SEG20	SEG20
304H	SEG5	SEG5	SEG5	SEG5	314H	SEG21	SEG21	SEG21	SEG21
305H	SEG6	SEG6	SEG6	SEG6	315H	SEG22	SEG22	SEG22	SEG22
306H	SEG7	SEG7	SEG7	SEG7	316H	SEG23	SEG23	SEG23	SEG23
307H	SEG8	SEG8	SEG8	SEG8	317H	SEG24	SEG24	SEG24	SEG24
308H	SEG9	SEG9	SEG9	SEG9	318H	SEG25	SEG25	SEG25	SEG25
309H	SEG10	SEG10	SEG10	SEG10	319H	SEG26	SEG26	SEG26	SEG26
30AH	SEG11	SEG11	SEG11	SEG11	31AH	SEG27	SEG27	SEG27	SEG27
30BH	SEG12	SEG12	SEG12	SEG12	31BH	SEG28	SEG28	SEG28	SEG28
30CH	SEG13	SEG13	SEG13	SEG13	31CH	SEG29	SEG29	SEG29	SEG29
30DH	SEG14	SEG14	SEG14	SEG14	31DH	SEG30	SEG30	SEG3	SEG30
30EH	SEG15	SEG15	SEG15	SEG15	31EH	SEG31	SEG31	SEG31	SEG31
30FH	SEG16	SEG16	SEG16	SEG16	31FH	SEG32	SEG32	SEG32	SEG32



### 11.5. LCD 电源



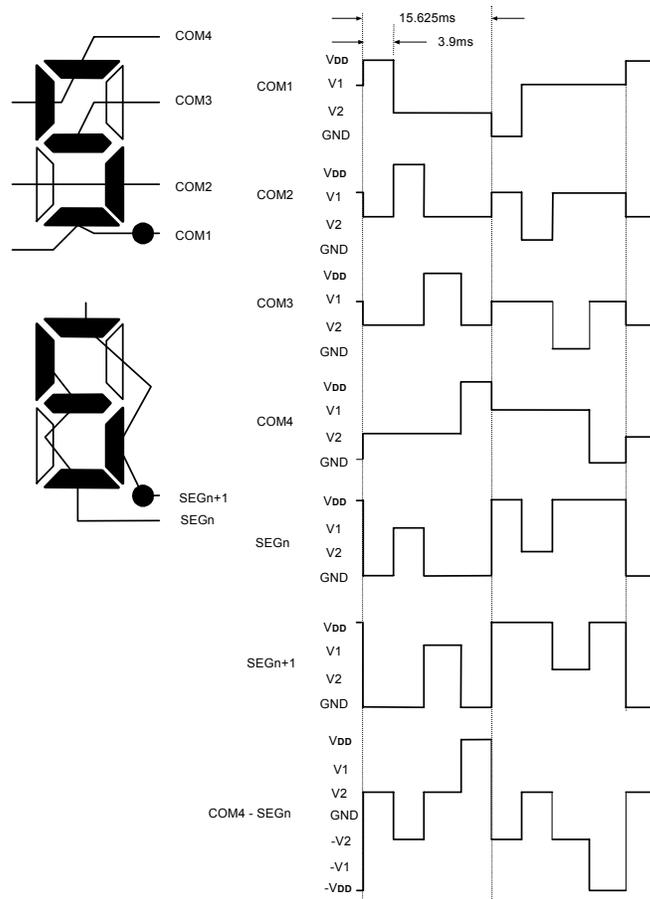
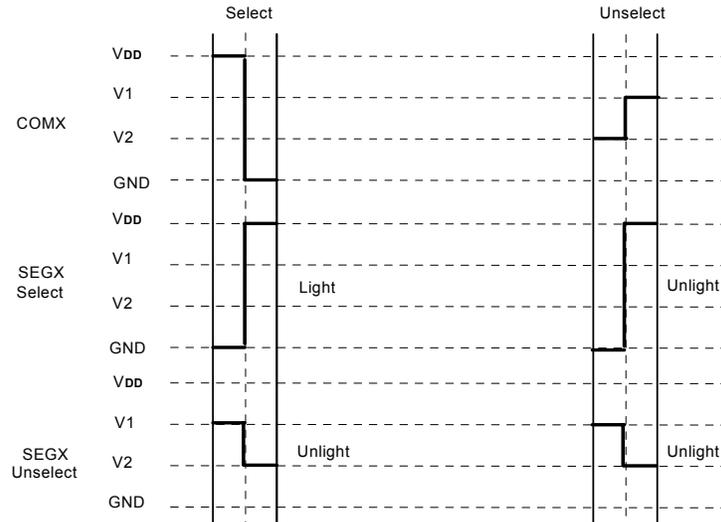
LCD 偏置电阻通过 LCDON 开关连接到 VLCD 引脚。

用户可以把 VLCD 连接到 VDDR1 使 LCD 得到稳定的电源（REN1 位必须置“1”，打开 VDDR1）。如果 Pump 打开，电流将加倍。如果  $V_{DD} > 3.6V$ ，可以关闭 Pump。

用户也可以直接把 VLCD 连接至 VDD（如果 VDD 是稳定的电源），这样耗电比较低。



11.6. 1/4占空比, 1/3偏置电压时的LCD驱动波形



1/4 占空比, 1/3 偏置电压时的 LCD 驱动波形



## 12. 中断

SH66P58 有四个中断源:

- ADC 中断
- Timer0 中断
- Base Timer 中断
- PORTB 中断 (下降沿)

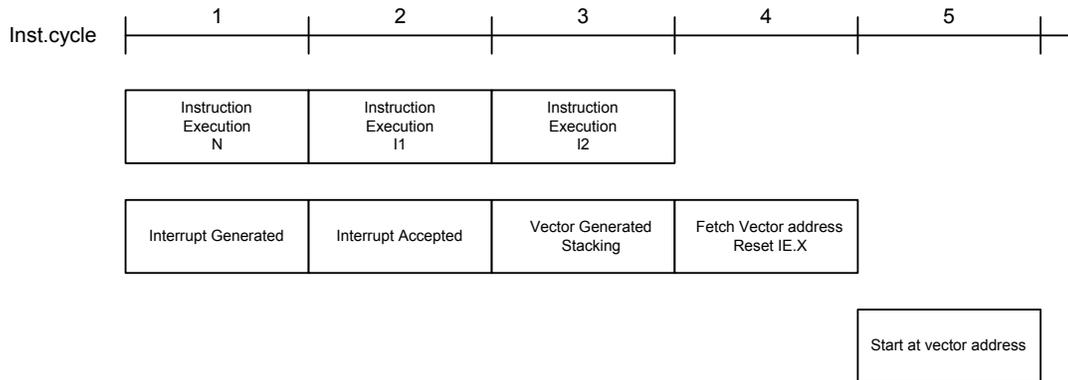
### 中断控制标志位和中断服务

中断控制标志位为系统寄存器的\$00和\$01。这两个寄存器能够由软件访问和设置。芯片上电复位后，这些标志位被清0。

### 系统寄存器: \$00 - \$01

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$00	IEADC	IET0	IEBT	IEP	读/写	中断允许标志寄存器
\$01	IRQADC	IRQT0	IRQBT	IRQP	读/写	中断请求标志寄存器

当 IEx 设置为1且有中断请求时 (IRQx 为1)，中断被激活并且根据中断优先级产生相应的中断矢量地址。当发生中断时，PC 和 CY 标志将被保存在堆栈存储器中，同时程序跳转至中断服务矢量地址处执行。在中断发生后，所有中断允许标志 (IEx) 自动复位为0，因此在 IRQx = 1时 IEx 标志再次设置为1时，将可能再次产生中断。



中断服务流程图

### 中断嵌套

在 CPU 中断服务期间，用户可以在中断返回前设置任何中断允许标志。中断服务流程图中标示下个中断和将要发生的下一个中断嵌套。如果中断请求已经产生且执行允许 IE 允许的指令 N，那么在两个指令周期后将执行中断程序。但是，如果指令 I1 或指令 I2清除中断请求或允许标志，那么中断服务将被取消。

### ADC 中断

当 ADCEN 被置“1”，每一次 ADC 转换完成产生一个中断请求 (IRQADC = 1)，如果中断允许位被允许 (IEADC = 1) 则进入 ADC 中断服务程序。ADC 中断能用来将 CPU 从 HALT 方式中唤醒。

### 定时器中断

Timer0 的计数时钟是以系统时钟或外部时钟/事件 T0 为基准的。Timer 计数值由\$FF 到\$00 溢出时将产生一个内部中断请求 (IRQT0 = 1)，如果中断允许位被允许 (IET0 = 1) 则进入定时器中断服务程序。定时器中断同样也能用于从 HALT 模式唤醒 CPU。

### BT 中断

BT 的计数时钟是以谐振器时钟 (32.768kHz) 为基准的。BT 每 2<sup>14</sup> (2<sup>12</sup>, 2<sup>10</sup>, 2<sup>8</sup>) 个谐振器时钟溢出，产生一个中断请求 (IRQBT = 1)，如果中断允许位被允许 (IEBT = 1) 则进入 BT 中断服务程序。BT 中断同样也能用于从 HALT 模式唤醒 CPU。

**端口下降沿中断**

PORTB 作为外部端口中断源，且仅仅在输入状态时才有可能产生端口中断。I/O 端口的任何输入引脚上的下降沿将产生中断请求（IRQP = 1，PIF.X = 1）。如果中断允许位被允许（IEP = 1，PIEN.X = 1）则进入端口中断服务程序。端口中断可以用来将 CPU 从 HALT 或者 STOP 模式唤醒。

**端口中断允许寄存器：\$2E**

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$2E	PIEN.3	PIEN.2	PIEN.1	PIEN.0	读/写	PORTB 中断允许标志寄存器

PIEN.n, (n = 0, 1, 2, 3)

0: 禁止中断。(初始值)

1: 允许中断。

**端口中断标志寄存器：\$2F**

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$2F	PIF.3	PIF.2	PIF.1	PIF.0	读/写	PORTB 中断请求标志寄存器

PIF.n, (n = 0, 1, 2, 3)

0: 没有发生中断。(初始值)

1: 已经发生中断。

该寄存器只能清0。

**注意:**

在输入状态，PORTB 端口的任何输入引脚上的下降沿都将置 PIF.X 为“1”，不管其他引脚是何电平。

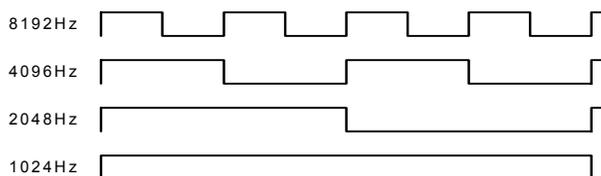


13. 蜂鸣音驱动输出

载波频率设置寄存器: \$0D

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$0D	PULLEN	ALMF1	ALMF0	ALMEN	读/写	第 0 位: 蜂鸣音控制寄存器 第 2-1 位: 蜂鸣音载波频率寄存器 第 3 位: 端口上拉寄存器
	x	X	X	0	读/写	禁止蜂鸣音载波输出, PA.0 作为 I/O 端口
	x	0	0	1	读/写	蜂鸣音载波频率为 1024Hz
	x	0	1	1	读/写	蜂鸣音载波频率为 2048Hz
	x	1	0	1	读/写	蜂鸣音载波频率为 4096Hz
	x	1	1	1	读/写	蜂鸣音载波频率为 8192Hz
	0	x	x	x	读/写	端口上拉禁止
	1	x	x	x	读/写	端口上拉允许

以下是可编程蜂鸣音驱动波形:



蜂鸣音驱动波形



#### 14. 脉冲宽度调制 (PWM)

SH66P58 包含两个 8 位 PWM 模块。PWM 模块可以产生周期和占空比分别可以调整的脉宽调制波形。PWM 模块的工作模式由 PWMC 寄存器来设置。PWM 输出的周期由 PWMP 寄存器来设置。而 PWM 输出波形的占空比由 PWMD 寄存器来设置。

##### 系统寄存器\$320, \$321: PWM 控制寄存器 (PWMC)

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$320, \$321	PWMnS	TnCK1	TnCK0	PWMn_EN	读/写	第 0 位: PWMn 输出允许设置寄存器 第 2-1 位: PWMn 时钟选择寄存器 第 3 位: PWMn 占空比的输出模式设置寄存器
	X	X	X	0	读/写	PWM 功能关闭 (初始值)
	X	X	X	1	读/写	PWM 功能打开
	X	0	0	X	读/写	PWMn 时钟 = tosc (初始值)
	X	0	1	X	读/写	PWMn 时钟 = 2tosc
	X	1	0	X	读/写	PWMn 时钟 = 4tosc
	X	1	1	X	读/写	PWMn 时钟 = 8tosc
	0	X	X	X	读/写	PWMn 占空比输出普通模式 (高电平有效) (初始值)
	1	X	X	X	读/写	PWMn 占空比输出反转模式 (低电平有效)

n = 0 或者 1

PWM0 输出引脚与 PORTD.0 共用。

PWM1 输出引脚与 PORTD.1 共用。

##### 系统寄存器\$322 - \$323, \$326 - \$327: PWM 周期控制寄存器 (PWMP)

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$322, \$326	PPn.3	PPn.2	PPn.1	PPn.0	读/写	PWMn 周期低 4 位寄存器
\$323, \$327	PPn.7	PPn.6	PPn.5	PPn.4	读/写	PWMn 周期高 4 位寄存器

n = 0 或者 1

PWM 输出周期 = [PPn.7, PPn.0] X PWMn 时钟。

当[PPn.7, PPn.0] = 00H, 如果 PWMnS 位设置为 0, PWMn 输出低电平。

当[PPn.7, PPn.0] = 00H, 如果 PWMnS 位设置为 1, PWMn 输出高电平。

##### 系统寄存器\$324 - \$325, \$328 - \$329: PWM 占空比控制寄存器 (PWMD)

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$324, \$328	PDn.3	PDn.2	PDn.1	PDn.0	读/写	PWMn 占空比低 4 位寄存器
\$325, \$329	PDn.7	PDn.6	PDn.5	PDn.4	读/写	PWMn 占空比高 4 位寄存器

n = 0 或者 1

PWM 输出占空比 = [PDn.7, PDn.0] X PWM1 时钟。

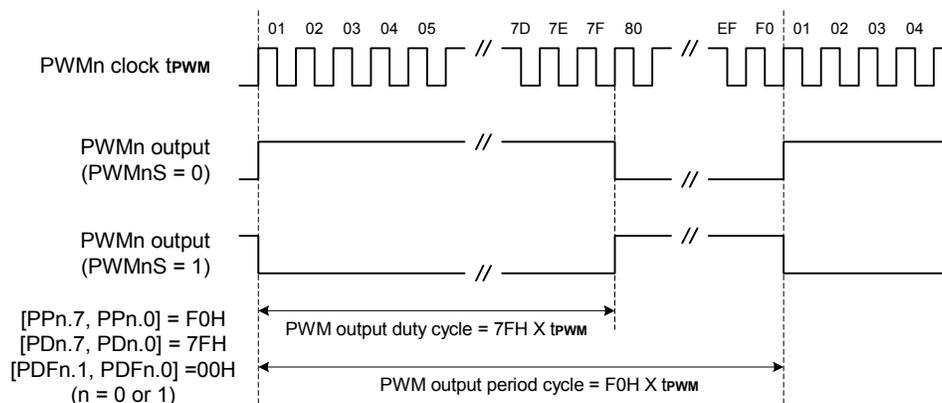
如果[PPn.7, PPn.0] ≤ [PDn.7, PDn.0], 当 PWMnS 位设置为 0, PWMn 输出高电平。

如果[PPn.7, PPn.0] ≤ [PDn.7, PDn.0], 如果 PWMnS 位设置为 1, PWMn 输出低电平。

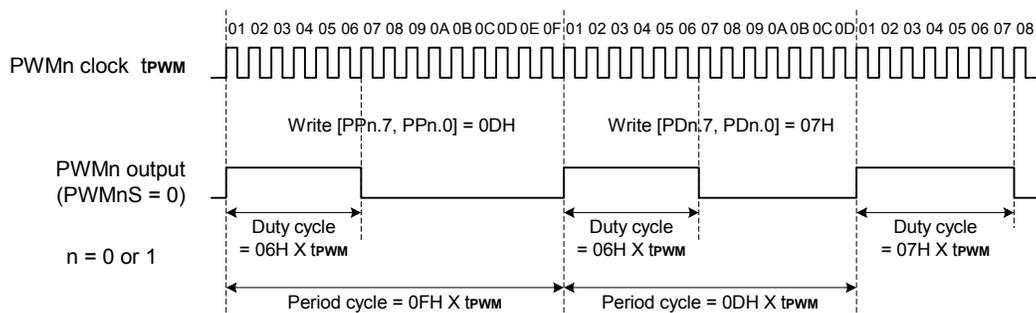


**编程注意事项:**

- 选择 PWM 模块时钟源。
- 通过写适当的值到 PWM 占空比控制寄存器 (PWMD) 设置 PWM 占空比: 首先设置低四位, 然后设置高四位。
- 通过写适当的值到 PWM 周期控制寄存器 (PWMP) 设置 PWM 周期: 先设置低四位, 然后设置高四位。
- 通过写 PWM 控制寄存器 (PWMC) 的 PWMS 位选择 PWM 占空比的输出模式。
- 为了输出适当的 PWM 波形, 通过写 PWM 控制寄存器 (PWMC) 中的 PWM\_EN 位为“1”来允许 PWM 模块工作。
- 如果 PWM 周期或者占空比需要改变, 操作流程如同步骤 b 或者步骤 c 说明。修改后的重载入计数器的值在下一个周期开始有效。



**PWM 输出举例**



**PWM 输出周期或者占空比周期变化举例**



**15. 低电压复位 (LVR)**

LVR 用于监控电源电压并产生芯片内部复位。它一般用于交流供电电路或有较大负载的电路，这些电路工作时负载的启动会引起器件工作电压暂时低于电路的最低允许工作电压。

LVR 功能可以通过代码选项永久开启或禁止。

当 LVR 功能开启时其功能如下：

- 当  $V_{DD} \leq V_{LVR}$  时产生系统复位。
- 当  $V_{DD} > V_{LVR}$  时释放系统复位。

这里， $V_{DD}$ ：电源电压， $V_{LVR}$ ：LVR 检测电压。

**16. 低电压检测 (LPD)**

LPD 功能用来监测电池电压，产生 LPD 标志。

**LPD 控制寄存器：\$2B**

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$2B	LPDEN	LPDF	LPDS1	LPDS0	读/写	第 1-0 位：检测电压选择位 第 2 位：LPD 标志位 第 3 位：LPD 允许控制位
	0	x	x	x	读/写	禁止低电压检测功能
	1	x	0	0	读/写	允许低电压检测功能，当 $V_{DD} < 2.4V$ 时，LPDF = 1
	1	x	0	1	读/写	允许低电压检测功能，当 $V_{DD} < 3.0V$ 时，LPDF = 1
	1	x	1	0	读/写	允许低电压检测功能，当 $V_{DD} < 3.3V$ 时，LPDF = 1
	1	x	1	1	读/写	允许低电压检测功能，当 $V_{DD} < 3.6V$ 时，LPDF = 1
	x	0	x	x	只读	无标志
	x	1	x	x	只读	标志产生

**17. ROM 数据读出列表 (RDT)**

**系统寄存器：\$14 - \$17**

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$14	RDT.3	RDT.2	RDT.1	RDT.0	读/写	ROM 数据列表地址/数据寄存器
\$15	RDT.7	RDT.6	RDT.5	RDT.4	读/写	ROM 数据列表地址/数据寄存器
\$16	RDT.11	RDT.10	RDT.9	RDT.8	读/写	ROM 数据列表地址/数据寄存器
\$17	RDT.15	RDT.14	RDT.13	RDT.12	读/写	ROM 数据列表地址/数据寄存器

RDT 寄存器由一个 13 位只写地址寄存器 (RDT.12 - RDT.0) 和一个 16 位只读 ROM 表数据读出寄存器组成 (RDT.15 - RDT.0)。为了读出 ROM 表数据，用户应该先把寄存器高 3 位 (bit13 - 15) 置 0，再写入 ROM 表地址到 RDT 寄存器 (先是高半字节后低半字节)，在一条指令后，指定地址的数据将会自动存入 RDT 寄存器 (写地址的最低位寄存器将会启动数据读出动作)。



## 18. 看门狗定时器 (WDT)

看门狗定时器是一个递减计数器，拥有独立内建 RC 振荡器作为时钟源，因此在 STOP 模式下仍会持续运行。当定时器溢出时，WDT 将复位 CPU。通过代码选项可以允许或禁止该功能。WDT 控制位 (\$1E 第 2-0 位) 用来选择不同的溢出时间。定时器溢出后，WDT 溢出标志 (\$1E 第 3 位) 将由硬件自动设置为“1”。通过读或者写系统寄存器 \$1E，WDT 会在溢出前重新开始计数。

### 看门狗定时器 (WDT): \$1E

地址	第 3 位	第 2 位	第 1 位	第 0 位	读/写	说明
\$1E	WDT	WDT.2	WDT.1	WDT.0	读/写 只读	第 2-0 位: 看门狗定时器控制寄存器 第 3 位: 看门狗定时器溢出标志寄存器
	X	0	0	0	读/写	WDT 溢出周期为 4096ms
	X	0	0	1	读/写	WDT 溢出周期为 1024ms
	X	0	1	0	读/写	WDT 溢出周期为 256ms
	X	0	1	1	读/写	WDT 溢出周期为 128ms
	X	1	0	0	读/写	WDT 溢出周期为 64ms
	X	1	0	1	读/写	WDT 溢出周期为 16ms
	X	1	1	0	读/写	WDT 溢出周期为 4ms
	X	1	1	1	读/写	WDT 溢出周期为 1ms
	0	X	X	X	只读	未发生 WDT 溢出复位
	1	X	X	X	只读	WDT 溢出, 发生 WDT 复位

**注意:** 看门狗定时器溢出周期是当 VDD = 5V 时的参考值。

## 19. HALT 和 STOP 模式

在执行 HALT 指令后，CPU 将进入待机模式 1 (HALT)。在 HALT 模式下，CPU 将停止工作。但是其周边电路 (Timer0, 时基定时器, 升压, 稳压, LPD, ADC...) 将继续工作。

在执行 STOP 指令后，CPU 将进入待机模式 2 (STOP)。在 STOP 模式下，除了看门狗定时器，升压，稳压 LPD 和 ADC 电路外，整个芯片 (包括振荡器) 将停止工作。

在 HALT 模式下，发生任何中断 CPU 将被唤醒。

在 STOP 模式下，发生端口中断 CPU 将被唤醒。

当通过任何中断，CPU 从 HALT/STOP 被唤醒，将会首先执行相关中断服务子程序。然后才会执行 HALT/STOP 的下一条指令。

**注意:** 为了节省功耗，用户在 HALT/STOP 指令前必须禁止升压，稳压，LPD 和 ADC。



## 20. 预热计数器

本芯片内建振荡器预热计数器，它能消除振荡器在下列情况下起振时的不稳定状态：

- (1) RESET 引脚复位
- (2) 上电复位
- (3) WDT 复位
- (4) LVR 复位
- (5) 由 STOP 模式唤醒

在晶体谐振器模式下，预热计数器预分频比为  $1/2^{14}$  (16384)。

## 21. 代码选项

### 21.1. 看门狗定时器

OP\_WDT:

0 = 关闭 (初始值)

1 = 打开

### 21.2. 低电压复位

OP\_LVR:

0 = 关闭 (初始值)

1 = 打开



OPT 编程引脚/焊垫说明 (OTP 编程模式)

焊垫编号	引脚编号	焊垫/引脚命名	引脚性质	共用引脚
	VDD	P	VDD	编程电源 (+5.5V)
	VPP	P	RESET	编程高压电源 (+11V)
	GND	P	GND	数字电源地
	SCK	I	OSCI	编程时钟输入引脚
	SDA	I/O	PORTA.0	编程数据引脚

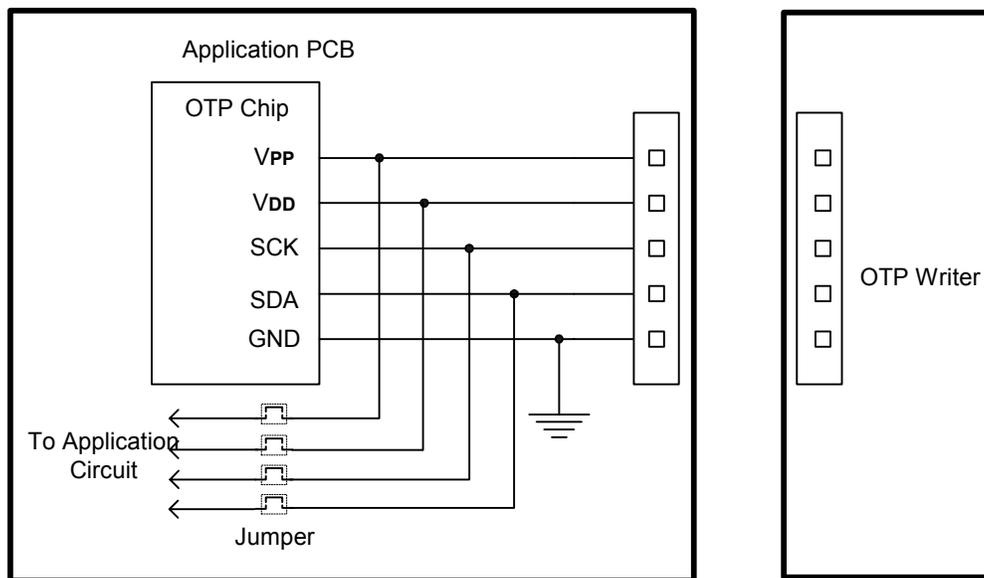
其中, I: 输入; O: 输出; P: 电源; Z: 高阻

OTP 在系统烧写时注意事项

OTP在系统编程时注意事项只对OTP芯片有效。

对于用户采用COB (Chip on Board) 组装方式时, OTP芯片可以使用在系统编程 (In System Programming) 方式编程。使用在系统编程方式编程时, 用户必须在印制板 (PCB) 上预留出OTP芯片的编程接口, 以便连接OTP编程器进行编程。在此模式下, 用户可在OTP芯片编程前将包括OTP芯片在内的所有器件组装在PCB上后, 再对OTP芯片进行编程。当然也可以可先将OTP芯片组装到PCB上, 对OTP芯片编程完成后再组装其它器件。

为了提高OTP编程的可靠性, 在编程操作时OTP编程信号线必须直接连接到OTP编程器上, 不允许有其它器件或外加电路与之并联。所以在PCB上必须预留4组跳线或分割焊盘, 将OTP编程接口 (VDD, VPP, SDA, SCK) 与应用电路分隔开, 如下图所示:



具体操作步骤如下:

- (1) 在 OTP 芯片编程前将 4 组跳线断开。
- (2) 将 OTP 芯片的编程接口连接到 OTP 编程器, 完成代码编程。
- (3) 将用户板与 OTP 烧写器编程器断开, 将 4 组跳线短接。

有关 OTP 编程的更多详细资料, 请参见 OTP 编程器的用户手册。



## 指令集

所有的指令都是单周期和单字的指令。具有面向存储器的操作特性。

### 1. 以下为算术和逻辑指令

#### 1.1. 累加器类型

助记符	指令代码	功能	标志位改变
ADC X (, B)	00000 0bbb xxx xxxx	$AC \leftarrow Mx + AC + CY$	CY
ADCM X (, B)	00000 1bbb xxx xxxx	$AC, Mx \leftarrow Mx + AC + CY$	CY
ADD X (, B)	00001 0bbb xxx xxxx	$AC \leftarrow Mx + AC$	CY
ADDM X (, B)	00001 1bbb xxx xxxx	$AC, Mx \leftarrow Mx + AC$	CY
SBC X (, B)	00010 0bbb xxx xxxx	$AC \leftarrow Mx + -AC + CY$	CY
SBCM X (, B)	00010 1bbb xxx xxxx	$AC, Mx \leftarrow Mx + -AC + CY$	CY
SUB X (, B)	00011 0bbb xxx xxxx	$AC \leftarrow Mx + -AC + 1$	CY
SUBM X (, B)	00011 1bbb xxx xxxx	$AC, Mx \leftarrow Mx + -AC + 1$	CY
EOR X (, B)	00100 0bbb xxx xxxx	$AC \leftarrow Mx \oplus AC$	
EORM X (, B)	00100 1bbb xxx xxxx	$AC, Mx \leftarrow Mx \oplus AC$	
OR X (, B)	00101 0bbb xxx xxxx	$AC \leftarrow Mx   AC$	
ORM X (, B)	00101 1bbb xxx xxxx	$AC, Mx \leftarrow Mx   AC$	
AND X (, B)	00110 0bbb xxx xxxx	$AC \leftarrow Mx \& AC$	
ANDM X (, B)	00110 1bbb xxx xxxx	$AC, Mx \leftarrow Mx \& AC$	
SHR	11110 0000 000 0000	$0 \rightarrow AC[3], AC[0] \rightarrow CY; AC$ 右移 1 位	CY

#### 1.2. 立即数类型

助记符	指令代码	功能	标志位改变
ADI X,I	01000 iiiii xxx xxxx	$AC \leftarrow Mx + I$	CY
ADIM X,I	01001 iiiii xxx xxxx	$AC, Mx \leftarrow Mx + I$	CY
SBI X,I	01010 iiiii xxx xxxx	$AC \leftarrow Mx + -I + 1$	CY
SBIM X,I	01011 iiiii xxx xxxx	$AC, Mx \leftarrow Mx + -I + 1$	CY
EORIM X,I	01100 iiiii xxx xxxx	$AC, Mx \leftarrow Mx \oplus I$	
ORIM X,I	01101 iiiii xxx xxxx	$AC, Mx \leftarrow Mx   I$	
ANDIM X,I	01110 iiiii xxx xxxx	$AC, Mx \leftarrow Mx \& I$	

#### 1.3. 十进制调整

助记符	指令代码	功能	标志位改变
DAA X	11001 0110 xxx xxxx	$AC, Mx \leftarrow$ 加法的十进制调整	CY
DAS X	11001 1010 xxx xxxx	$AC, Mx \leftarrow$ 减法的十进制调整	CY



2. 传输指令

助记符	指令代码	功能	标志位改变
LDA X (, B)	00111 0bbb xxx xxxx	AC ← Mx	
STA X (, B)	00111 1bbb xxx xxxx	Mx ← AC	
LDI X, I	01111 iii xxx xxxx	AC, Mx ← I	

3. 控制指令

助记符	指令代码	功能	标志位改变
BAZ X	10010 xxxx xxx xxxx	PC ← X, 如果 AC = 0	
BNZ X	10000 xxxx xxx xxxx	PC ← X, 如果 AC ≠ 0	
BC X	10011 xxxx xxx xxxx	PC ← X, 如果 CY = 1	
BNC X	10001 xxxx xxx xxxx	PC ← X, 如果 CY ≠ 1	
BA0 X	10100 xxxx xxx xxxx	PC ← X, 如果 AC (0) = 1	
BA1 X	10101 xxxx xxx xxxx	PC ← X, 如果 AC (1) = 1	
BA2 X	10110 xxxx xxx xxxx	PC ← X, 如果 AC (2) = 1	
BA3 X	10111 xxxx xxx xxxx	PC ← X, 如果 AC (3) = 1	
CALL X	11000 xxxx xxx xxxx	ST ← CY, PC +1 PC ← X (不包括 p)	
RTNW H, L	11010 000h hhh llll	PC ← ST; TBR ← hhhh, AC ← llll	
RTNI	11010 1000 000 0000	CY, PC ← ST	CY
HALT	11011 0000 000 0000		
STOP	11011 1000 000 0000		
JMP X	1110p xxxx xxx xxxx	PC ← X (包括 p)	
TJMP	11110 1111 111 1111	PC ← (PC11-PC8) (TBR) (AC)	
NOP	11111 1111 111 1111	空操作	

其中,

PC	程序计数器	I	立即数
AC	累加器	⊕	逻辑异或
-AC	累加器的反码		逻辑或
CY	进位标志位	&	逻辑与
Mx	数据存储器	bbb	RAM 页
p	ROM 页	B	RAM 页
ST	堆栈	TBR	查表寄存器



电气特性

极限参数\*

直流供电电压.....-0.3V to +7.0V  
 输入信号电压..... GND - 0.3V to VDD + 0.3V  
 工作环境温度.....-40°C to +85°C  
 存储温度.....-55°C to +125°C

\*注释

如果器件的工作条件超过左列“极限参数”的范围，将造成器件永久性破坏。只有当器件工作在说明书所规定的范围内时功能才能得到保障。器件在极限参数列举的条件下工作将会影响到器件工作的可靠性。

直流电气特性 (VDD = 2.4 - 5.5V, GND = 0V, TA = 25°C, 除非另有说明)

参数	符号	最小值	典型值*	最大值	单位	条件
工作电压	VDD	2.4	3.0	5.5	V	32.768kHz ≤ fosc ≤ 4.192MHz
LVR电压	VLVR	2.3	2.4	2.5	V	LVR有效
LPD电压	VLPD	3.5	3.6	3.7	V	LPD有效, 且LPDS[1:0] = 11
工作电流	IOP	-	10	20	μA	fosc = 32.768kHz, VDD = 3.0V 所有输出引脚无负载, 执行 NOP 指令, WDT, LVR, LCD, 升压, 稳压, ADC关闭
		-	0.6	1.0	mA	fosc = 4.2MHz, VDD = 3.0V 所有输出引脚无负载, 执行 NOP 指令, WDT, LVR, LCD, 升压, 稳压, ADC关闭
待机电流	ISB	-	4	8	μA	fosc = 32.768kHz, VDD = 3.0V 所有输出引脚无负载 (包括所有数字输入引脚不浮动), 执行 HALT 指令, WDT, LVR, LCD, 升压, 稳压, ADC关闭
		-	-	1.0	μA	VDD = 3.0V 所有输出引脚无负载(包括所有数字输入引脚不浮动), 执行 STOP 指令, WDT, LVR, LCD, 升压, 稳压, ADC关闭
WDT电流	IWDT	-	-	10	μA	
LVR电流	ILVR	-	2	3	μA	
LPD电流	ILPD	-	-	20	μA	
升压电流	IPUMP	-	5	10	μA	升压输出无负载
稳压1电流	IR1	-	20	30	μA	稳压1输出无负载, VDDR1 = 3.0V
稳压2电流	IR2	-	100	150	μA	稳压2输出无负载, VDDR2 = 3.0V
输入低电压	VIL	GND	-	VDD X 0.3	V	I/O端口
		GND	-	VDD X 0.2	V	RESET, T0 (施密特触发输入)
输入高电压	VIH	VDD X 0.7	-	VDD	V	I/O端口
		VDD X 0.8	-	VDD	V	RESET, T0 (施密特触发输入)
输入漏电流	IIL	-1	-	1	μA	I/O端口, GND < VIN < VDD
上拉电阻	RP	-	50	-	kΩ	上拉电阻 (VDD = 3.0V, VIN = GND)
		-	30	-	kΩ	上拉电阻 (VDD = 5.0V, VIN = GND)
输出低电压	VOL1			GND + 0.6	V	I/O端口, 不包括PC.0, IOL = 5mA (VDD = 3.0V)
						PC.0, IOL = 10mA (VDD = 3.0V)



直流电气特性 (VDD = 2.4 - 5.5V, GND = 0V, TA = 25°C, 除非另有说明) (续)

参数	符号	最小值	典型值*	最大值	单位	条件
输出高电压	VOH1	VDD - 0.7	-	-	V	I/O端口, IOH = -3mA (VDD = 3.0V)
LCD输出内阻	RON	-	5	-	kΩ	LCD COMx, LCD SEGx, V1, V2, V3电压的漂移小于0.2V
LCD偏置电阻	RLCD	-	150	-	kΩ	FSTEN = 0, RLCD0 = 0
		-	20	-	kΩ	FSTEN = 0, RLCD0 = 1
稳压1输出	VDDR1	2.85	3.0	3.15	V	VDD = 3.0V, 稳压1输出无负载
稳压1输出电流能力	IDRV1	-	2	-	mA	VDD = 3.0V, 升压允许, ΔVDDR1 < 0.2V
稳压2输出	VDDR2	2.85	3.0	3.15	V	VDD = 3.0V, 稳压2输出无负载
稳压2输出电流能力	IDRV2	-	10	-	mA	VDD = 3.0V, 升压允许, ΔVDDR2 < 0.2V
稳压2输出温度系数	tVDDR2	-	100	-	ppm/°C	T = 0 - 50°C
VCOM输出	VCOM	1.425	1.5	1.575	V	VDD = 3.0V, ADC on
VCOM 输出电流能力	ISDR	-	10	-	μA	VDD = 3.0V, 升压允许, ADC允许, ΔVCOM < 0.1V
VCOM 输入电流能力	ISNK	-	1	-	mA	VDD = 3.0V, 升压允许, ADC允许, ΔVCOM < 0.1V

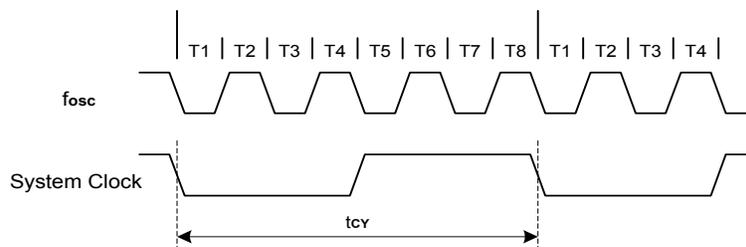
交流电气特性 (VDD = 2.4 - 5.5V, GND = 0V, TA = 25°C, 除非另有说明)

参数	符号	最小值	典型值*	最大值	单位	条件
振荡器起振时间	tOSC		1	2	s	32.768kHz晶体谐振器
复位脉冲宽度 (低电平)	tRESET	10	-	-	μs	VDD = 3.0V
WDT时间	tWDT	1	-	-	ms	VDD = 3.0V
T0输入宽度	tIW	(tcY + 40)/N	-	-	ns	N = 预分频比
输入脉冲宽度	tIPW	tIW/2	-	-	ns	
LVR 低脉冲宽度	tLVR	-	30	-	μs	
PLL频率漂移范围	ΔF /F	-	-	0.5	%	持续256个时钟的平均值

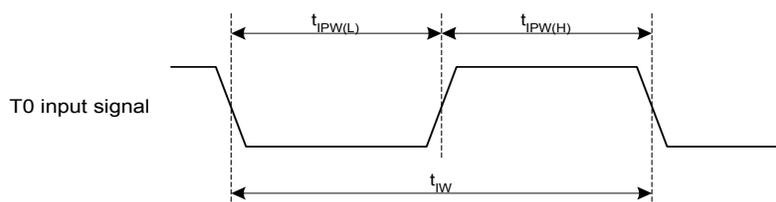


时序波形

(a) 系统时钟时序波形:



(b) T0 输入波形:



ADC 电器特性

( $V_{DD} = 2.4 - 5.5V$ ,  $GND = 0V$ , 升压允许, PGA 增益 = 100,200,  $T_A = 25^\circ C$ , 系统时钟 = 4.2/4MHz, 除非另有说明。)

参数	符号	最小值	典型值*	最大值	单位	条件
工作电流	I <sub>ADC</sub>	-	1	1.5	mA	
精度	NR	-	16	-	bit	$0.4 \leq AIN \leq VREF$
参考电压	V <sub>REF+</sub> , V <sub>REF-</sub>	0.4	-	1.2	V	内部参考电压
		0.4	-	2.0		外部参考电压
ADC 输入电压	AIN+, AIN-	0.4	-	2.0	V	
噪声	NFC	-	14	16	bit	ADC 时钟 = 100kHz
微分非线性误差	DNL	-	±0.5	±0.5	LSB	ADC 时钟 = 100kHz
积分非线性误差	INL	-	±0.004	±0.008	% of FSR	$VREF \pm 0.60V$ , ADC 数据 = -25000 - 31250
有效位数	ENOB	14	-	16	bit	
ADC 时钟	FAD	50	-	200	kHz	

PGA 电气特性

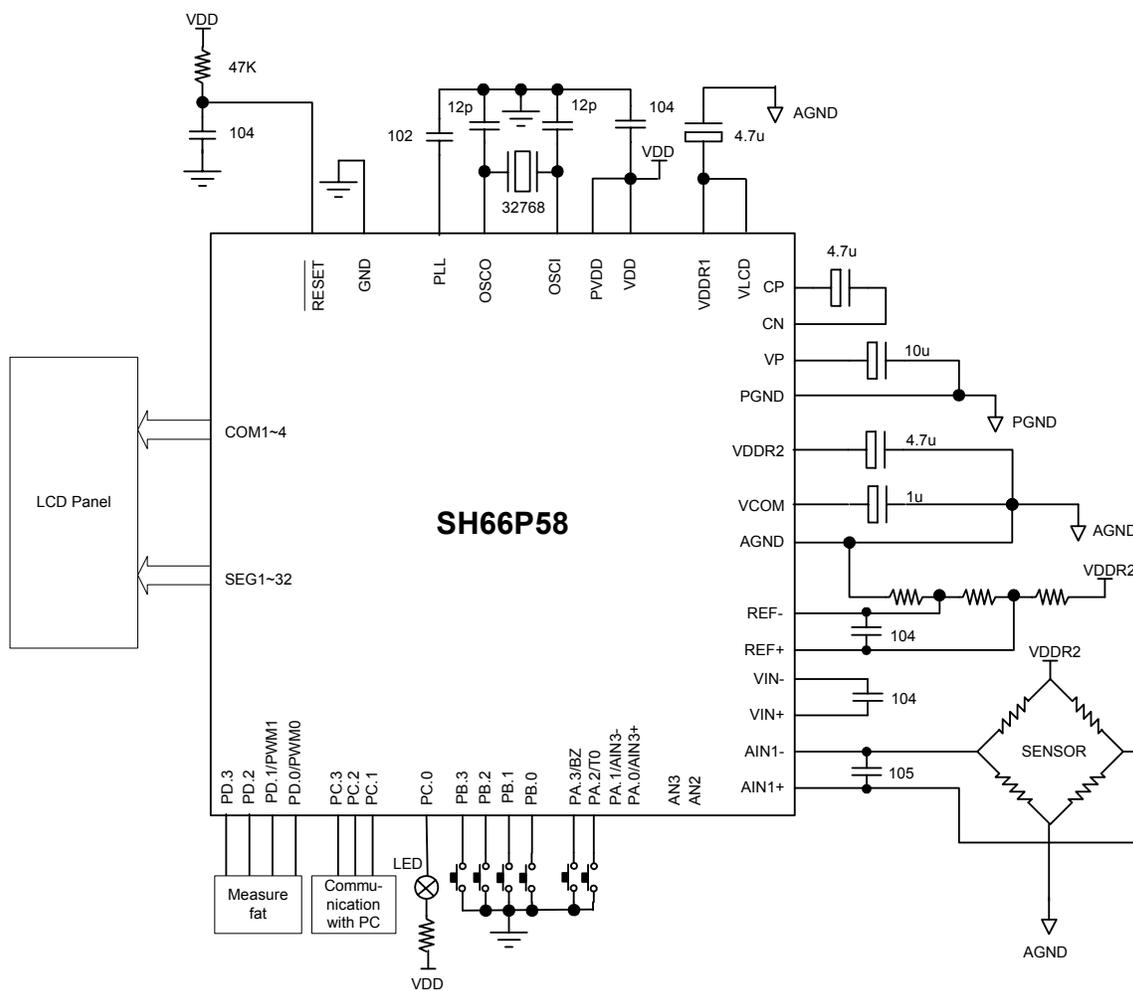
( $V_{DD} = 2.4 - 5.5V$ ,  $GND = 0V$ , 升压允许, PGA 增益 = 100,200,  $T_A = 25^\circ C$ , 系统时钟 = 4.2/4MHz, 除非另有说明。)

参数	符号	最小值	典型值*	最大值	单位	条件
输入失调 1	V <sub>ZER1</sub>		20	50	μV	斩波允许, 升压禁止
输入失调 2	V <sub>ZER2</sub>		1	2	mV	斩波禁止, 升压禁止
PGA 输入范围	V <sub>PGIN</sub>	0.4	-	2.0	V	
PGA 输出范围	V <sub>PGOUT</sub>	0.4	-	2.0	V	
PGA 增益误差	GAIN	0	-	4	%	增益 = 200, 斩波禁止



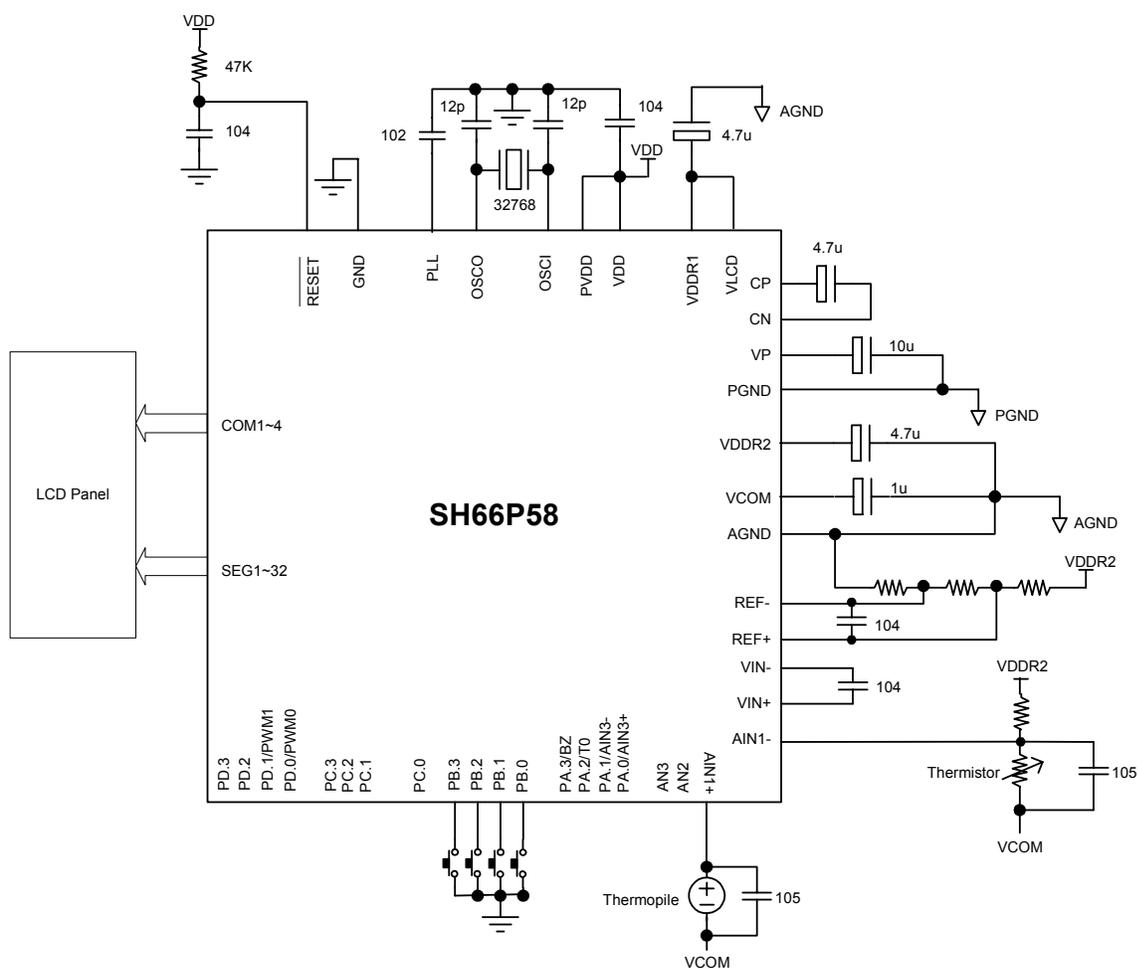
应用电路 (仅供参考)

1. 称重应用





2. 温度计应用







焊垫坐标

单位:  $\mu\text{m}$ 

焊垫编号	名称	X	Y	焊垫编号	名称	X	Y
1	CP	-837.9	-1275.36	44	PD.1/PWM1	897.75	1159.34
2	PVDD	-726.75	-1274.76	45	PD.2	897.75	1274.76
3	CN	-624.15	-1274.76	46	PD.3	778.05	1274.76
4	PGND	-521.55	-1274.76	47	PC.0	675.45	1274.76
5	VDD	-427.5	-1274.76	48	PC.1	571.57	1274.76
6	VLCD	-333.45	-1274.76	49	PC.2	481.19	1274.76
7	COM1	-239.4	-1274.76	50	PC.3	390.82	1274.76
8	COM2	-149.03	-1274.76	51	PB.0	300.45	1274.76
9	COM3	-58.65	-1274.76	52	PB.1	210.07	1274.76
10	COM4	31.72	-1274.76	53	PB.2	119.7	1274.76
11	SEG1	122.09	-1274.76	54	PB.3	22.57	1274.76
12	SEG2	212.47	-1274.76	55	VDD	-67.8	1274.76
13	SEG3	302.84	-1274.76	56	OSCI	-158.17	1274.76
14	SEG4	393.21	-1274.76	57	OSCO	-389.54	1274.76
15	SEG5	483.59	-1274.76	58	GND	-484.53	1274.76
16	SEG6	573.96	-1274.76	59	PLL	-579.52	1274.76
17	SEG7	675.45	-1274.76	60	RESET	-684	1275.36
18	SEG8	778.05	-1274.76	61	PA.0/ALM	-897.75	1260.31
19	SEG9	897.75	-1274.76	62	PA.1/T0	-897.75	1146.6
20	SEG10	897.75	-1159.34	63	PA.2/AIN3-/AN5	-897.75	1044
21	SEG11	897.75	-1045.62	64	PA.3/AIN3+/AN4	-897.75	915.75
22	SEG12	897.75	-932.76	65	NC	-897.75	825.37
23	SEG13	897.75	-830.16	66	VREF+ OUT	-897.75	731.84
24	SEG14	897.75	-739.79	67	VREF+ IN	-897.75	641.46
25	SEG15	897.75	-649.41	68	VREF- OUT	-897.75	551.09
26	SEG16	897.75	-559.04	69	VREF- IN	-897.75	460.72
27	SEG17	897.75	-468.67	70	VCOM IN	-897.75	370.34
28	SEG18	897.75	-378.29	71	VCOM OUT	-897.75	279.97
29	SEG19	897.75	-287.92	72	VIN-	-897.75	189.6
30	SEG20	897.75	-197.55	73	VIN+	-897.75	99.22
31	SEG21	897.75	-107.17	74	AIN2-/AN3	-897.75	8.85
32	SEG22	897.75	-16.8	75	AIN2+/AN2	-897.75	-81.52
33	SEG23	897.75	73.57	76	AIN1-/AN1	-897.75	-171.9
34	SEG24	897.75	163.95	77	AIN1+/AN0	-897.75	-262.27
35	SEG25	897.75	254.32	78	AGND2	-897.75	-352.64
36	SEG26	897.75	344.69	79	AGND1	-897.75	-443.02
37	SEG27	897.75	435.07	80	AGND0	-897.75	-533.39
38	SEG28	897.75	525.44	81	VDDR1 IN	-897.75	-623.77
39	PE.0/SEG29	897.75	615.81	82	VDDR1 OUT	-897.75	-714.14
40	PE.1/SEG30	897.75	714.74	83	VDDR2 IN	-897.75	-804.51
41	PE.2/SEG31	897.75	821.61	84	VDDR2 OUT	-897.75	-907.11
42	PE.3/SEG32	897.75	928.49	85	VP	-898.35	-1018.26
43	PD.0/PWM0	897.75	1043.91				



订购信息

产品编号	封装
SH66P58H	Chip form



产品规格更改记录

版本	记录	日期
1.0	初始版本	2009年8月