



版本号	修改时间	修改记录	修改人
V1.0	2022.7.19	初稿	FANMH
V1.1	2023.02.14	加入寄存器表定义	FANMH
V1.2	2024.11.1	修改部分字体	ZHANGLH
V1.3	2025.1.7	修改频率输出描述	ZHANGLH



## SD8801 全温度补偿和超小封装的实时时钟芯片

### 1、概述

SD8801 是一种具有标准 IIC 接口的实时时钟芯片, CPU 可使用该接口通过 7 位地址寻址来读写片内寄存器的数据。

SD8801 具有一个后备电池输入脚 VBAT, 内部的充电电路可对外接的充电电池进行智能充电, 也可对电池电量进行检测和欠压报警指示。

SD8801 内置时间更新/倒计时/报警中断/事件捕获功能。

SD8801 每一颗芯片具有唯一的 8 字节的身份识别码 (ID)。

SD8901 管脚兼容 8901。

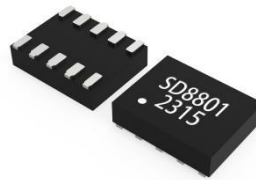
### 2、特性

- 低功耗: 0.6  $\mu$ A 典型值 ( $T_a=25^{\circ}\text{C}$ )。
- 工作电压: 2.5V~5.5V; 计时电压: 1.5V~5.5V。
- 工作温度:  $-40^{\circ}\text{C}\sim 125^{\circ}\text{C}$ 。
- 标准 IIC 总线接口方式, 最高速度 400kHz。
- 年、月、日、时、分、秒的 BCD 码输入/输出, 并可通过独立的地址访问各时间寄存器。
- 闰年自动调整功能 (2000 年~2099 年)。
- 内置星期/日期、时、分共 3 字节的报警数据寄存器。
- 周期性频率中断输出: 32768Hz、1024Hz 和 1Hz 共三种方波。
- 自动重载的 12 位的倒计时定时器, 可选的 4 种时钟源 (4096Hz、1024Hz、1 秒、1 分钟), 最小定时为 244 $\mu$ s, 最长定时为 68 个小时, 通过计算可获得精确的毫秒级定时值。
- 时间更新中断具有分钟中断与秒中断两种功能。
- 时间报警中断、倒计时中断、时间更新中断、事件捕获中断可通过寄存器配置选择是否从 INT 脚输出报警信号, 四种中断各具有一个中断标志位。
- 具有一次性或充电的后备电池输入脚 VBAT, 其内部的稳压充电电路可选择性地对外接的充电电池进行自动充电 (特别注意: SD8801 的充电功能上电默认打开, 如果不使用充电功能, 建议关闭)。
- 内置电池电压检测功能, 可读取当前电池电压值; 可读取寄存器了解是否发生过欠压或过压的情况。
- 内置通信校验功能, 进一步提高通信的可靠性。
- 内置 1/1024 秒寄存器, 读取时间能够精确到 1/1024 秒。
- 内置 IIC 总线 0.5 秒自动复位功能 (从 start 命令开始计时), 该功能可以避免 IIC 总线挂死问题。
- 内置写保护功能, 避免对数据的误写操作, 可更好地保护数据。
- 内置上电指示位 RTCF, 当包括电池在内的所有电源第一次上电时该位置 1。

- 内置停振检测位OSF，当内部振荡器停止振荡时该位置1。
- 内置晶振和谐振电容，芯片内部通过高精度补偿方法，实现在宽温范围内高精度的计时功能： $-40^{\circ}\text{C} \sim +85^{\circ}\text{C} < \pm 5\text{ppm}$ ； $+86^{\circ}\text{C} \sim +105^{\circ}\text{C} < \pm 10\text{ppm}$ ； $+106^{\circ}\text{C} \sim +125^{\circ}\text{C} < \pm 20\text{ppm}$ （出厂精度）。
- 内置70字节通用SRAM寄存器可用于存储用户的一般数据
- 内置8字节的ID码，芯片出厂之前设定的、全球唯一的身份识别码。
- 芯片在兴威帆的评估板上可通过4KV的群脉冲(EFT)干扰
- 芯片通过ESD HBM 4KV/CDM 1KV测试
- CMOS 工艺
- 封装形式：3225(3.2mmx2.5mmx0.75mm)。

### 3、管脚定义和功能框图

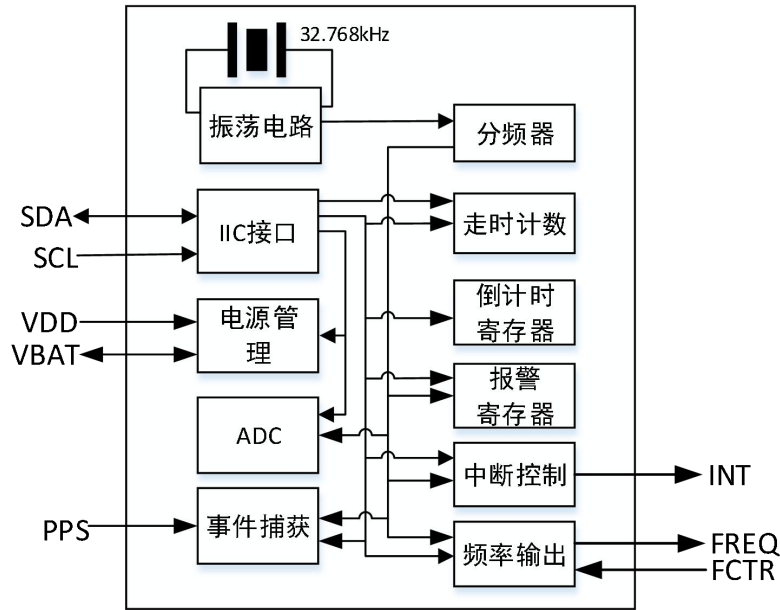
VDD	1	10	NC
NC	2	9	GND
VBAT	3	8	INT
FREQ	4	7	SDA
SCL	5	6	PPS



SD8801 俯视图

管脚定义说明：

引脚	名称	功能	特征
1	VDD	正电源脚	
2、10	NC	没有与芯片内部连接	
3	VBAT	备用电池输入脚，内置稳压及充电电流可选的充电电路。	2.3V~3.6V, 不用时应将其接GND。
4	FREQ	频率输出端口	CMOS 输出
5	SCL	串行时钟输入脚，由于在SCL上升/下降沿处理信号，要特别注意SCL信号的上升/下降沿时间，应严格遵守说明书。为了减少SCL上升沿时间，MCU与SCL连接的端口可设为CMOS输出，不要设置为开漏输出。	CMOS 输入
6	PPS	时钟同步与事件捕获的输入脚	CMOS 输入
7	SDA	串行数据输入/输出脚，此管脚通常用一电阻上拉至VDD，并与其它漏极开路或集电极开路输出的器件通过线与方式连接。	开漏输出，CMOS 输入
8	INT	报警中断输出脚	CMOS 输入
9	GND	电源地（GND）	



功能框图

## 4、基本功能定义

### 4.1 寄存器列表

地址	寄存器段	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0	复位值
00H/10H	实时时钟寄存器	秒	0	S40	S20	S10	S8	S4	S2	S1	0XXX-XXXX
01H/11H		分钟	0	MN40	MN20	MN10	MN8	MN4	MN2	MN1	0XXX-XXXX
02H/12H		小时	0	0	H20	H10	H8	H4	H2	H1	00XX-XXXX
03H/13H		星期	0	W6	W5	W4	W3	W2	W1	W0	0XXX-XXXX
04H/14H		日	0	0	D20	D10	D8	D4	D2	D1	00XX-XXXX
05H/15H		月	0	0	0	MO10	MO8	MO4	MO2	MO1	000X-XXXX
06H/16H		年	Y80	Y40	Y20	Y10	Y8	Y4	Y2	Y1	XXXX-XXXX
07H	用户 RAM	(1Byte)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	XXXX-XXXX
08H	时间报警寄存器	分钟报警	EAMN	AMN40	AMN20	AMN10	AMN8	AMN4	AMN2	AMN1	1000-0000
09H		小时报警	EAH	-	AH20	AH10	AH8	AH4	AH2	AH1	1X00-0000
0AH		星期报警	EAW	AW6	AW5	AW4	AW3	AW2	AW1	AW0	1000-0000
	日报警	EAD	0	AD20	AD10	AD8	AD4	AD2	AD1		
0BH/1BH	倒计时寄存器	倒计时计数器	TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0	0000-0000
0CH/1CH			-	-	-	-	TD11	TD10	TD9	TD8	XXXX-0000
0DH/1DH	控制寄存器 1	CTR1	0	EDEW	US	DE	FS1	FS0	TDS1	TDS0	0000-0010
0EH/1EH	状态寄存器 1	FLAG1	0	0	INTUF	INTDF	INTAF	0	RTCF	VDET	0000-0011
0FH/1FH	控制寄存器 2	CTR2	-	-	INTUE	INTDE	INTAE	0	0	0	1000-0000



17H	温度寄存器	TEMP	TM8	TM7	TM6	TM5	TM4	TM3	TM2	TM1	0000-0000
18H	充电寄存器 1	CHARGE	-	-	-	-	-	SWOFF	-	-	0000-0000
4FH	状态寄存器 2	FLAG2	0	OSF	-	-	-	0	-	-	0100-0001
57H	控制寄存器 3	CTR3	BATIIC	TM0	BSY	CONT	0	0	0	0	0000_0000
58H	充电寄存器 2	CHARGE	ENCH	0	0	0	0	CH2	CH1	CH0	0000-0000
59H	控制寄存器 4	CTR4	-	-	-	CONT_B AT	INTTHE	INTTLE	INTBHE	INTBLE	0000-0000
5AH	状态寄存器 3	FLAG3	BAT8_VA L	SYS	OSC_RD Y	0	THF	TLF	BHF	BLF	0000-0000
5BH	电池电量		BAT7_VA L	BAT6_V AL	BAT5_V AL	BAT4_V AL	BAT3_V AL	BAT2_V AL	BAT1_VA L	BAT0_VA L	0100-0000
5CH	温度报警寄存器	低温报警	TEMP7_A L	TEMP6_ AL	TEMP5_ AL	TEMP4_ AL	TEMP3_ AL	TEMP2_ AL	TEMP1_A L	TEMPO_A L	0000-0000
5DH		高温报警	TEMP7_A H	TEMP6_ AH	TEMP5_ AH	TEMP4_ AH	TEMP3_ AH	TEMP2_ AH	TEMP1_A H	TEMPO_A H	0000-0000
6DH~ B1H	用户 RAM	(69Bytes)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	XXXX-XXXX
B2H~ B9H	ID(只读)	(8Bytes)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	XXXX-XXXX
F2H	时间戳寄存器	秒	0	TS40	TS20	TS10	TS8	TS4	TS2	TS1	XXXX-XXXX
F3H		分钟	0	TMN40	TMN20	TMN10	TMN8	TMN4	TMN2	TMN1	XXXX-XXXX
F4H		小时	0	0	TH20	TH10	TH8	TH4	TH2	TH1	XXXX-XXXX
F5H		星期	0	TW6	TW5	TW4	TW3	TW2	TW1	TW0	XXXX-XXXX
F6H		日	0	TDA40	TDA20	TDA10	TDA8	TDA4	TDA2	TDA1	XXXX-XXXX
F7H		月	0	0	0	TMO10	TMO8	TMO4	TMO2	TMO1	XXXX-XXXX
F8H		年	TY80	TY40	TY20	TY10	TY8	TY4	TY2	TY1	XXXX-XXXX
F9H	事件捕获寄存器	事件捕获控制位	ECP	EHL	EPU	RCE	EIE	ERST	ET1	ET0	0000-0000
FAH		事件捕获标志位	EF	0	0	0	EVMON	0	0	0	0000-0000
FBH	通信校验寄存器	BCC	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	XXXX-XXXX
FCH	数据保护寄存器	WP	WPF	BIT6	BIT5	BIT4	BIT3	BIT2	0	0	0000-0000
FEH	1/1024 秒寄存器	1/1024S	128	64	32	16	8	4	2	1	0000_0000
FFH			0	0	0	0	0	0	0	512	256

## 4.2 实时时钟寄存器 (00H~06H 或 10H~16H)

实时时钟数据寄存器是 7 字节的存储器, 包括年、月、日、星期、时、分、秒的数据。

### 4.2.1 秒寄存器 [00H 或 10H 地址]

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
00H/10H	秒	0	S40	S20	S10	S8	S4	S2	S1

BCD 码格式, 如 08H-09H-10H-11H, 数据范围: 00H~59H。

### 4.2.2 分钟寄存器 [01H 或 11H 地址]

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
01H/11H	分钟	0	MN40	MN20	MN10	MN8	MN4	MN2	MN1

BCD 码格式, 如 08H-09H-10H-11H, 数据范围: 00H~59H。

### 4.2.3 小时寄存器 [02H 或 12H 地址]

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
02H/12H	小时	0	0	H20	H10	H8	H4	H2	H1

BCD 码格式, 如 08H-09H-10H-11H, 数据范围: 00H~23H。

### 4.2.4 星期寄存器 [03H 或 13H 地址]

星期寄存器的 D0 位至 D6 位分别用来表示星期日, 星期一, 星期二……一直到星期六, 具体如下表所示:

星期	D7	D6	D5	D4	D3	D2	D1	D0	数值
星期日	0	0	0	0	0	0	0	1	01H
星期一	0	0	0	0	0	0	1	0	02H
星期二	0	0	0	0	0	1	0	0	04H
星期三	0	0	0	0	1	0	0	0	08H
星期四	0	0	0	1	0	0	0	0	10H
星期五	0	0	1	0	0	0	0	0	20H
星期六	0	1	0	0	0	0	0	0	40H

注: 不要同时设定多位为“1”的情况。

### 4.2.5 日期寄存器 [04H 或 14H 地址]

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
04H/14H	日期	0	0	D20	D10	D8	D4	D2	D1

BCD 码格式, 如 08H-09H-10H-11H。

每月包含的天数通过自动日历功能来更改, 范围如下:

1, 3, 5, 7, 8, 10, 12: 1H~31H

4, 6, 9, 11: 1H~30H

2 (闰年): 1H~29H

2 (平年) : 1H~28H

#### 4.2.6 月寄存器[05H 或 15H 地址]

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
05H/15H	月	0	0	0	M010	M08	M04	M02	M01

BCD 格式, 如 08H~09H~10H~11H, 数据范围: 01H~12H。

#### 4.2.7 年寄存器[06H 或 16H 地址]

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
06H/16H	年	Y80	Y40	Y20	Y10	Y8	Y4	Y2	Y1

BCD 格式, 如 08H~09H~10H~11H, 数据范围: 00H~99H。

#### 4.2.8 时间设置示例

设置时间为 2006 年 12 月 20 日星期三 18 点 19 分 20 秒, 寄存器 00H~06H 分别设置: 20H、19H、18H、08H、20H、12H、06H。

注:

(1) 上电复位时, 芯片内部不对实时时钟数据寄存器作清零或置位处理。

(2) 当写实时数据时 (00H~06H), 不可以单独对七个时间数据中的某一位进行写操作, 否则可能引起时间数据的错误进位, 所以要修改其中某一个数据, 应一次性写入全部七个实时时钟数据。

(3) 当芯片收到读实时时钟数据命令, 则所有实时时钟数据被锁存 (时钟走时并不受影响), 此功能可以避免时间数据的错读现象。

### 4.3 寄存器功能

#### 4.3.1 用户 RAM[07H、6DH~B1H 地址]

地址为 07H 和 6DH~B1H, 共计 70 字节的用户数据 RAM。

#### 4.3.2 时间报警中断

地址 08H, 09H, 0AH 除了最高位的时间报警允许使能位之外, 还存放报警时间的分钟、小时、星期/日期的数据, 具体如下表所示:

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
08H	分钟报警	EAMN	AMN40	AMN20	AMN10	AMN8	AMN4	AMN2	AMN1
09H	小时报警	EAH	-	AH20	AH10	AH8	AH4	AH2	AH1
0AH	星期报警	EAW	AW6	AW5	AW4	AW3	AW2	AW1	AW0
	日报警	EAD	-	AD20	AD10	AD8	AD4	AD2	AD1
0DH/1DH	CTR1	0	EDEW	US	DE	FS1	FS0	TDS1	TDS0
0EH/1EH	FLAG1	0	0	INTUF	INTDF	INTAF	0	RTCF	VDET
0FH/1FH	CTR2	-	-	INTUE	INTDE	INTAE	0	0	0

报警中断的报警时间由时间报警寄存器（08H~0AH）来确定。

EDEW: 星期/日期报警的功能选择位。EDEW=1, 选择日期报警; EDEW=0, 选择星期报警。

INTAE: 时间报警中断输出使能位。INTAE=1, 允许 INT 端口输出时间报警信号; INTAE=0, 禁止 INT 端口输出时间报警信号。

EAX 在地址 08H, 09H, 0AH 的最高位。EAX 为时间报警寄存器的使能位, 用于确定哪些时间报警寄存器（分钟、小时、星期/日期）需要与实时时钟寄存器作比较。EAX=0, 则该寄存器的报警时间需要与对应的实时时钟寄存器的时间对比; EAX=1, 则直接忽略该寄存器的报警时间, 不进行对比。

INTAF: 时间报警事件发生标志位。INTAF=1, 表示时间报警已发生; INTAF=0, 表示时间报警未发生。

注: 往 INTAF 写入 0 清除时间报警标志位, 写入 1 无效。

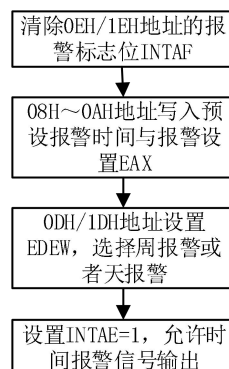
当设定的报警时间和实时时间相匹配时, 就会触发一次报警中断, 同时报警中断标志位 INTAF 置 1, 如果报警中断输出允许位 INTAE=1, 则 INT 输出时间报警信号。

报警设置如下:

地址	寄存器	D7	D6	D5	D4	D3	D2	D1	D0
08H	分钟报警	EAMN 为分钟报警使能位	AMN40	AMN20	AMN10	AMN8	AMN4	AMN2	AMN1
		EAMN=0; 开启分钟报警 EAMN=1; 关闭分钟报警	BCD 码格式, 一个小时只能报警某一分钟。范围 00~59						
09H	小时报警	EAH 为小时报警使能位	BIT6	AH20	AH10	AH8	AH4	AH2	AH1
		EAH=0; 开启小时报警 EAH=1; 关闭小时报警	BCD 码格式, 一天内只能报警某一小时。范围 00~23						
0AH	EDEW=0, 选择星期 报警	EAW 为周报警使能位	AW6	AW5	AW4	AW3	AW2	AW1	AW0
		周报警映射	周六	周五	周四	周三	周二	周一	周日
	EAW=0; 开启周报警 EAW=1; 关闭周报警	星期报警, 可设置多天报警							
	EDEW=1, 选择日期 报警	EAD 为日期报警使能位	BIT6	AD20	AD10	AD8	AD4	AD2	AD1
EAD=0; 开启日期报警 EAD=1; 关闭日期报警		BCD 码格式, 只能报警某天。范围见 4.2.5							

注: 08H, 09H, 0AH 的最高位都被写入 1 时, 则忽略 EDEW, 每分钟发生一次报警中断。

启用时间报警功能的流程图如下:

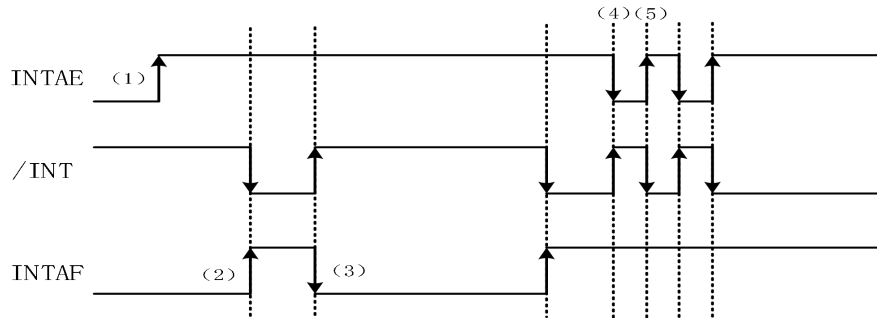




时间报警举例：

- (1) 0EH/1EH 地址的 INTAF=0；清除时间报警中断标志位。
- (2) 设置报警时间：每周周三和周六的 8 点 10 分报警。则 08H, 09H, 0AH 的依次写入 10H, 08H, 48H。
- (3) 0DH/1DH 地址的 EDEW=0，设置为星期报警。
- (4) 0FH/1FH 地址的 INTAE=1，允许 INT 端口输出时间报警中断信号。

时间报警功能时序图如下所示：



- (1) INTAE=1，允许 INT 端口输出时间报警中断的报警信号。
- (2) 设定的报警时间和实时时间相匹配，发生报警事件，报警中断标志位 INTAF=1，INT 端口输出低电平的报警信号。

注：如果设置当前具体日期/时间进行报警，将会是下次满足条件时发生报警，而不是立即报警。

- (3) INTAF=0，INT 立即变成高阻态。
- (4) INTAE=0，INT 立即变成高阻态。
- (5) INTAE=1，INT 立即变成低电平。

#### 4.3.3 倒计时中断

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
0BH/1BH	倒计时计数器	TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0
0CH/1CH		-	-	-	-	TD11	TD10	TD9	TD8
0DH/1DH	CTR1	0	EDEW	US	DE	FS1	FS0	TDS1	TDS0
0EH/1EH	FLAG1	0	0	INTUF	INTDF	INTAF	0	RTCF	VDET
0FH/1FH	CTR2	-	-	INTUE	INTDE	INTAE	0	0	0

与倒计时中断相关的寄存器是 0BH, 0CH, 0DH, 0EH, 0FH。

倒计时计数器：预设的倒计时计数数值。

INTDE：倒计时中断输出使能位。INTDE=1，允许 INT 端口输出倒计时中断报警信号；INTDE=0，禁止 INT 端口输出倒计时中断报警信号。

DE：倒计时中断启停位。DE=1，启动倒计时中断功能，倒计时中断的内部计数器启动并开始计数；DE=0，停止倒计时中断的功能，倒计时中断的内部计数器停止。

注：每次 DE=1 启动时，倒计时中断都会重新载入预设的倒计时中断计数数值，并进行

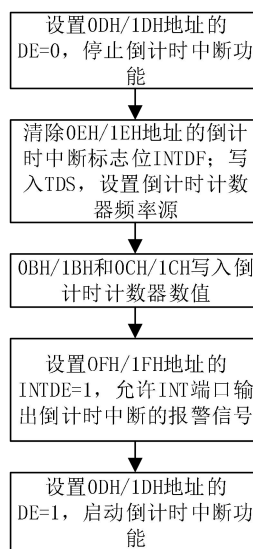
计数。

INTDF: 倒计时中断事件发生的标志位。INTDF=1, 表示倒计时中断事件已发生; INTDF=0, 表示倒计时中断事件未发生。

TDS: 倒计时中断的频率源选择位, 具体如下表:

TDS1	TDS0	倒计时中断时钟源频率	低脉宽 t
0	0	4096Hz	122us
0	1	64Hz	7.8ms
1	0	1Hz	7.8ms
1	1	1/60Hz	7.8ms

启用倒计时中断功能的流程图如下:

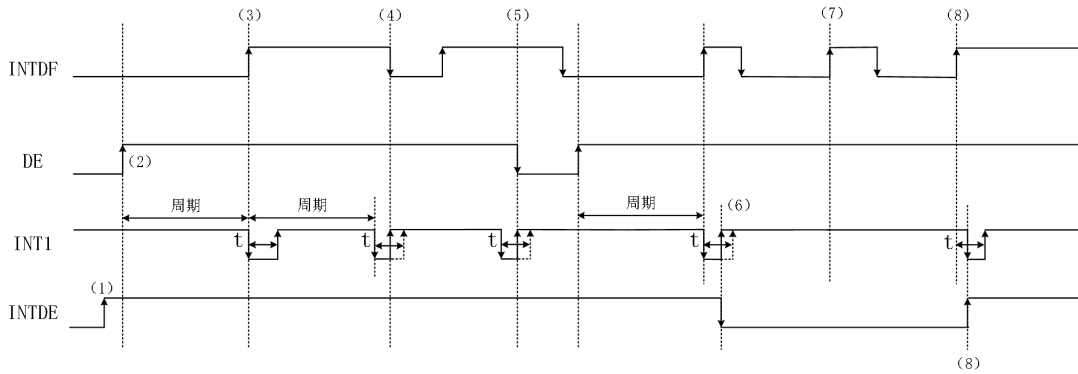


注: 当重新配置倒计时中断时, 需要关闭倒计时, 即置 DE=0, 然后再置 DE=1, 才可以启用新的倒计时中断。

倒计时中断应用举例: 频率源选择 64Hz, 计数数值设置 124H, 报警时间为  $292/64=4.5625s$ 。即每 4.5625s 倒计时中断报警一次。

- (1) 0DH/1DH 地址的 DE=0, 停止倒计时中断; TDS=01, 频率源选择 64Hz。
- (2) 0EH/1EH 地址 INTDF=0, 清除倒计时中断标志位 INTDF。
- (3) 0BH/1BH 地址写入 24H, 0CH/1CH 地址写入 01H, 设置倒计时中断计数数值。
- (4) 0FH/1FH 地址的 INTDE=1, 允许 INT 端口输出倒计时中断的报警信号。
- (5) 0DH/1DH 地址的 DE=1, 启动倒计时中断。

倒计时中断功能时序图如下所示:



(1) INTDE=1, 允许 INT 端口输出倒计时报警信号。

(2) DE=1, 倒计时计数器加载预设计数数值并开始计时。

(3) 倒计时计数器基于选择的时钟源开始向下计数, 当计数数值从 001H 变为 000H 时, 倒计时标志位 INTDF 置=1, INT 输出中断报警信号。

注 1: 当计数数值从 001H 变为 000H 时发生中断事件, 计数器会自动重新加载预设值, 并再次开始计数。

注 2: INTDF 置 1 后, 只能通过软件进行写 0 清除。

(4) 在倒计时报警中断输出的低电平期间 INTDF 被软件清零, INT 立刻变成高阻态。

(5) 在倒计时报警中断输出的低电平期间 DE=0, 倒计时功能停止, INT 立刻变成高阻态。

(6) 在倒计时报警中断输出的低电平时间里 INTDE=0, INT 立刻变成高阻态。

(7) INTDE=0 时发生倒计时报警事件, INT 无倒计时报警中断信号输出。

(8) INTDF=1, INTDE=0 时发生倒计时报警事件。在不到  $t$  的时间里 INTDE=1, 则 INT 端口立刻输出低电平报警信号。

#### 4.3.4 频率输出

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
0DH/1DH	CTR1	0	EDEW	US	DE	<b>FS1</b>	<b>FS0</b>	TDS1	TDS0
地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
0DH/1DH	CTR1	0	EDEW	US	DE	<b>FS1</b>	<b>FS0</b>	TDS1	TDS0

FS: 输出信号的频率选择, 频率输出选择如下表:

FS1	FS0	频率
0	0	无
0	1	1Hz
1	0	1024Hz
1	1	32768Hz

### 4.3.5 时间更新中断

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
0DH/1DH	CTR1	0	EDEW	<b>US</b>	DE	FS1	FS0	TDS1	TDS0
0EH/1EH	FLAG1	0	0	<b>INTUF</b>	INTDF	INTAF	0	RTCF	VDET
0FH/1FH	CTR2	-	-	<b>INTUE</b>	INTDE	INTAE	0	0	0

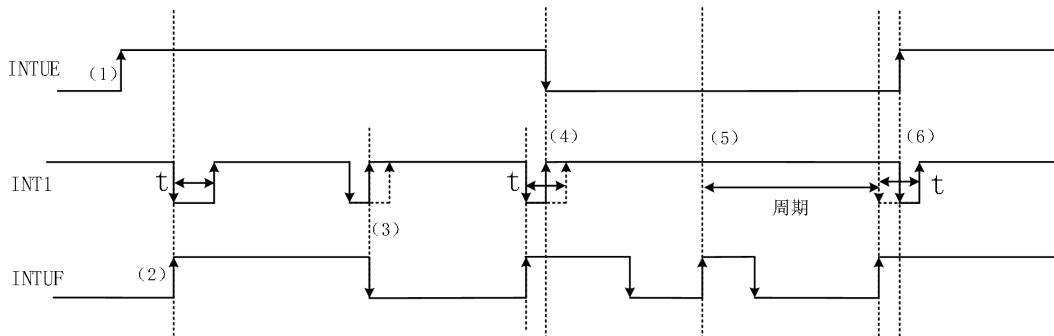
**INTUE**: 时间更新中断输出使能位。INTUE=1, 允许 INT 端口输出时间更新中断报警信号; INTUE=0, 禁止 INT 端口输出时间更新中断报警信号。

**INTUF**: 时间更新中断事件发生的标志位, INTUF=1, 表示时间更新中断事件已发生; INTUF=0, 表示时间更新中断事件未发生。

**US**: 时间更新中断类型选择如下表:

US	时间更新中断类型选择	低脉宽 t
0	秒中断	500ms
1	分钟中断	15.6ms

时间更新中断时序图如下:



(1) INTUE=1, 时间更新中断报警功能开启, 允许 INT 端口输出时间更新中断报警信号。

(2) 当分钟/秒发生更新时 (由 US 位决定), 就会发生时间更新中断事件, INTUF 被置为 1。如果 INTUE=1, INT 端口输出时间更新中断报警信号。

(3) 时间更新中断报警信号的低电平期间 INTUF 被软件清零, INT 立刻变成高阻态。

(4) 时间更新中断报警信号的低电平期间 INTUE=0, INT 立刻变成高阻态。

(5) INTUE=0 时, 发生时间更新中断事件, INTUF 置 1, INT 端口无时间更新中断报警信号输出。

(6) INTUE=0 时, 发生时间更新中断事件, INTUF=1, 在不到 t 的时间里 INTUE=1, INT 端口立刻输出低电平报警信号。

### 4.3.6 状态寄存器

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
0EH/ 1EH	标志位寄存器	0	0	INTUF	INTDF	INTAF	0	RTCF	VDET
4FH	状态寄存器 2	0	<b>OSF</b>	-	-	-	0	-	-

57H	控制寄存器 3	BATIIC	TM0	BSY	CONT	-	-	-	-
5AH	扩展控制寄存器 2	BAT8_VAL	SYS	OSC_RDY	0	THF	TLF	BHF	BLF

RTCF: 上电标志位。电源失效后再上电则该位置 1, 为只读位。上电后的第一次有效写(只要写一个字节即可)就可以将 RTCF 位清为 0。

VDET: 温度补偿欠压标志位, 当电源电压低于  $V_{temp}$  时, VDET 置 1。标志位可以通过写 0 进行清除。

OSF: 停振标志位, OSF=1, 表示之前有过停振事件发生, 默认值为 0。

BATIIC: VBAT 模式下的 IIC 通信允许位, 上电默认值为 0。BATIIC=0, VBAT 模式下禁止 IIC 通信; BATIIC=1, VBAT 模式下允许 IIC 通信。

SYS: 系统配置完成标志位。系统上电 80ms 后, SYS 置 1, 在 SYS 置 1 之前, 芯片输出禁止, IIC 不可操作。

OSC\_RDY: 外部 OSC 振荡器起振标志位, 该位在 OSC 起振 1 秒后置 1。

#### 4.3.7 温度测量与数据保存

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
17H	温度寄存器	TM8	TM7	TM6	TM5	TM4	TM3	TM2	TM1
57H	控制寄存器 3	BATIIC	TM0	BSY	CONT	-	-	-	-

TM: 保存测量的温度值。温度计算公式:  $T(^{\circ}\text{C}) = (\text{TEMP}[8:1] * 2 - 187.19) / 3.218$ 。

BSY: 状态标志位, BSY=1, 表示芯片正在进行电池电压或者温度转换; BSY=0, 表示处于空闲状态。

CONT: 强制温度测量位, 当 CONT=1 且 BSY=0 时, 进行强制温度测量。测量结果存放在寄存器 TM 中。(强制转换完成后, BSY=0, CONT=0)

#### 4.3.8 电池电压测量与数据保存

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
59H	扩展控制寄存器 1	-	-	-	CONT_BAT	INTTHE	INTTLE	INTBHE	INTBLE
5AH	扩展控制寄存器 2	BAT8_VAL	SYS	OSC_RDY	0	THF	TLF	BHF	BLF
5BH	电池电量	BAT7_VAL	BAT6_VAL	BAT5_VAL	BAT4_VAL	BAT3_VAL	BAT2_VAL	BAT1_VAL	BAT0_VAL

BAT\_VAL: 保存测量的电压。直接从对应地址读取当前电压。BAT8\_VAL=1, BAT0\_VAL~BAT7\_VAL=35H, 则当前所测的电池电量=135H=309 (十进制)=3.09V。

BSY: 状态标志位, BSY=1, 表示芯片正在进行电池电压或者温度转换; BSY=0, 表示处于空闲状态。

CONT\_BAT: 强制电压测量位, 当 CONT\_BAT=1 且 BSY=0 (空闲状态) 时, 进行强制电压

测量。测量结果存放在寄存器 VBAT\_VAL 中。（强制转换完成后，BSY=0，CONT\_BAT=0）

#### 4.3.9 充电功能

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
18H	充电寄存器 1	-	-	-	-	-	<b>SWOFF</b>	-	-
58H	充电寄存器 2	<b>ENCH</b>	-	-	-	-	<b>CH2</b>	<b>CH1</b>	<b>CH0</b>

SWOFF：充电寄存器 1 的充电选择位。SWOFF=0，通过 VDD 充电；SWOFF=1，通过 4.1V 充电。

Bit[7:4]	SWOFF	功能
其他	0	VDD 充电
	1	4.1V 充电
4' b1010	X	关闭充电

18H 地址的 Bit[7:4] 控制充电寄存器 1 的充电功能的开关：Bit[7:4]=4' b1010，则关闭充电寄存器 1 的充电功能；否则开启充电寄存器 1 的充电功能。

18H 地址的充电寄存器 1 关闭时，58H 的充电寄存器 2 设置生效。

使能	选择位			功能
	ENCH	CH2	CH1	
0	x	x	x	不充电
1	0	0	0	不充电
1	0	0	1	通过 2K 电阻充电到 3.1V 电压
1	0	1	0	通过 2K 电阻充电到 3.1V 电压
1	0	1	1	通过 2K 电阻充电到 3.1V 电压
1	1	0	0	通过 2K 电阻充电到 4.1V 电压
1	1	0	1	通过 2K 电阻充电到 VDD 电压
1	1	1	0	不充电
1	1	1	1	不充电

ENCH：充电寄存器 2 的充电功能使能位。ENCH=0，关闭充电功能；ENCH=1，由 CH 决定充电模式。

CH：充电模式选择位。具体功能如上表所示。

备注：1、当需要充电电压为 3.1V 或 4.1V 时，VDD 需满足  $VDD - 0.2V \geq VBAT$ 。

2、当充电功能打开时，芯片会增加 80uA 左右的工作电流。

3、充电功能上电默认打开，如果不使用充电功能，建议关闭。

#### 4.3.10 电池电压报警

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
59H	扩展控制寄存器 1	-	-	-	CONT_BAT	INTTHE	INTTLE	INTBHE	INTBLE
5AH	扩展控制寄存器 2	BAT8_VAL	SYS	OSC_RDY	0	THF	TLF	BHF	BLF

INTBHE: 电池高压报警使能位。INTBHE=1, 使能电池高压报警。

INTBLE: 电池低压报警使能位。INTBLE=1, 使能电池低压报警。

BHF: 电池高压标志位。

BLF: 电池欠压标志位。

注 1: BHF, BLF 不能通过软件进行设置或清除。

注 2: 低压报警的电压典型值为 2.5V; 3.3V 充电的高压报警的电压典型值为 2.9V; 4.2V 充电的高压报警的电压典型值为 3.95V。

#### 4.3.11 温度报警

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
59H	扩展控制寄存器 1	-	-	-	CONT_BAT	INTTHE	INTTLE	INTBHE	INTBLE
5AH	扩展控制寄存器 2	BAT8_VAL	SYS	OSC_RDY	0	THF	TLF	BHF	BLF
5CH	低温报警值	TEMP7_AL	TEMP6_AL	TEMP5_AL	TEMP4_AL	TEMP3_AL	TEMP2_AL	TEMP1_AL	TEMP0_AL
5DH	高温报警值	TEMP7_AH	TEMP6_AH	TEMP5_AH	TEMP4_AH	TEMP3_AH	TEMP2_AH	TEMP1_AH	TEMP0_AH

INTTHE: 高温报警使能位。INTTHE=1, 允许高温报警; INTTHE=0, 禁止高温报警。

INTTLE: 低温报警使能位。INTTLE=1, 允许低温报警; INTTLE=0, 禁止低温报警。

THF: 高温报警标志位。当温度高于或者等于设定的高温报警值时, THF=1; 当温度低于设定的高温报警值时, THF=0。

TLF: 低温报警标志位。当温度低于或者等于设定的低温报警值时, TLF=1; 当温度高于设定的高温报警值时, TLF=0。

TEMP\_AL: 可设置的低温报警阈值。

TEMP\_HL: 可设置的高温报警阈值。

注: THF, TLF 不能通过软件进行设置或清除。

#### 4.3.12 事件捕获功能

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
F2H	秒	0	TS40	TS20	TS10	TS8	TS4	TS2	TS1
F3H	分钟	0	TMN40	TMN20	TMN10	TMN8	TMN4	TMN2	TMN1
F4H	小时	0	0	TH20	TH10	TH8	TH4	TH2	TH1
F5H	星期	0	0	TW6	TW5	TW4	TW3	TW2	TW1
F6H	日	0	TDA40	TDA20	TDA10	TDA8	TDA4	TDA2	TDA1
F7H	月	0	0	0	TM010	TM08	TM04	TM02	TM01
F8H	年	TY80	TY40	TY20	TY10	TY8	TY4	TY2	TY1
F9H	事件捕获控制位	ECP	EHL	EPU	RCE	EIE	ERST	ET1	ET0
FAH	事件捕获标志位	EF	0	0	0	EVMON	0	0	0

ECP: 事件捕获使能位。ECP=0, 关闭事件捕获功能; ECP=1, 启用事件捕获功能。

EHL: 外部输入电平极性选择位。EHL=0, 则输入低电平有效; EHL=1, 则输入高电平有效。

EPU: 上拉电阻使能位。EPU=0, 禁止使用上拉电阻; EPU=1, 启用上拉电阻。

RCE: 重复捕获使能位。RCE=0, 事件捕获后, ECP 位自动清 0, 不再继续进行事件捕获; RCE=1, 完成一次事件捕获后, ECP 不清 0。

EIE: 事件捕获中断输出允许位。EIE=0, 中断事件发生时, INT 端口禁止中断信号输出; EIE=1, 中断事件发生时, INT 端口允许中断信号输出。

ERST: 时间同步位。当 ERST=1 时, 捕获到有效的事件, 清零秒以下的计数器, 同时 ERST 自动清零。

EF: 事件捕获触发标志位。当事件发生时, EF=1, 可以通过软件清零。

EVMON: PPS 状态指示位, PPS=1, 则 EVMON=1; PPS=0, 则 EVMON=0。

ET: 滤波周期选择位。当且仅当 PPS 信号持续时间大于滤波周期, 才会被视为有效信号。

注 1: 时间戳寄存器为只读寄存器。

ET1	ET0	滤波周期
0	0	无
0	1	3.9ms
1	0	15.6ms
1	1	125ms

事件捕获实例:

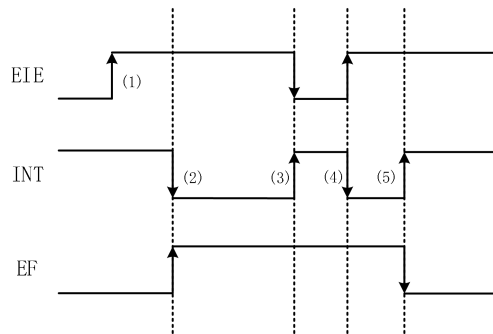
- (1) F9H 地址的 ECP=0, 停止事件捕获功能。
- (2) FAH 地址写入 00; 清除事件捕获触发标志位 EF。
- (3) 一次性往 F9H 地址写入对应的参数: EPU=1, 电阻上拉; RCE=1, 允许重复捕获事件; EIE=1, 允许 INT 端口输出事件捕获的中断信号; ET=01, 选择 3.9ms 滤波; EHL=1, 选



择外部输入高电平有效。

(4) F9H 地址的 ECP=1, 启用事件捕获功能。

事件捕获的功能时序图如下图所示：



(1) EIE=1, 允许事件捕获的中断输出

(2) 事件发生, EF=1, 此时 EIE 为 1, 则 INT 端口输出中断信号。

(3) EIE=0, INT 立刻停止输出报警信号。

(4) EIE=1, 此时 EF=1, INT 端口输出中断信号。

(5) EF 清零, INT 立刻停止输出报警信号。

#### 4.3.13 ID 码

ID 码的地址与内容对应的关系如下表：

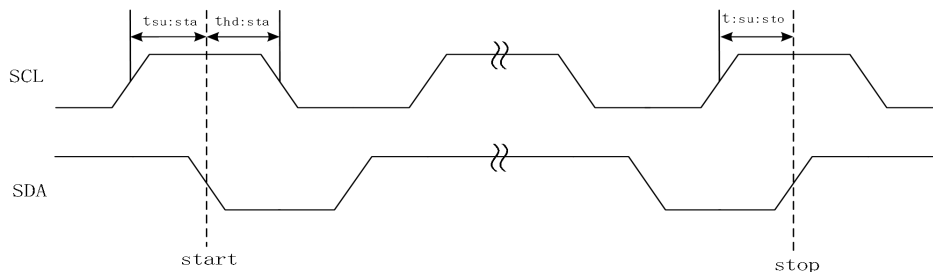
ID 码地址	B2H	B3H	B4H	B5H	B6H	B7H	B8H	B9H
说明	生产年份： 0~99	生产月 份：1~12	日期： 1~31	生产机 台编号	四位生产工单 号：如 A394		工单内序号： 0000~9999	

#### 4.3.14 通讯校验

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
FBH	BCC	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

IIC 通信校验寄存器用于校验 IIC 总线上面的通信数据, 提高传输的可靠性。

采用异或校验的方式, 即: 每次接收/发送的数据和上一次数据做异或运算, 数据校验从 START(包括 RESTART) 开始到 STOP 信号结束, 包括总线上所有的传输数据。



通讯校验范围

每次 RTC 收到 STOP 命令之后, 校验数据保存在校验寄存器 BCC 中, MCU 可以通过 IIC 进行读取。例:

1、IIC 写校验：往 00H、01H、02H、03H、04H、05H、06H 地址依次写入 28H, 41H, 14H, 04H, 15H, 11H, 22H；则从 FBH 地址读出的通信校验结果应为  $64H \oplus 00H \oplus 28H \oplus 41H \oplus 14H \oplus 04H \oplus 15H \oplus 11H \oplus 22H = 3BH$ 。

2、IIC 读校验：从 04H、05H、06H 地址读出结果依次为 0x08, 0x04, 0x10；则从 FBH 地址读出的通信校验结果应为  $65H \oplus 08H \oplus 04H \oplus 10H = 79H$ 。

注：在每一次的 S 或者 Sr 信号之后，通信校验功能就会重新开始。

具体 IIC 通信方式见 5.2 数据传输格式。

#### 4.3.15 写保护序列

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
FCH	写保护序列	WPF	BIT6	BIT5	BIT4	BIT3	BIT2	0	0

为了提高数据的可靠性，在写保护使能时，除写保护控制的寄存器可写入外，其他寄存器都不能写入。

WPF：写保护标志位，默认为 0。WPF=1，寄存器写禁止状态，此时不能对寄存器进行写入；WPF=0，寄存器写使能状态，可以对寄存器进行写入。

寄存器写禁止操作步骤：

- (1) 向寄存器的 Bit6~Bit2 写入 5' b00000，复位检测序列，进入第 2 步；
- (2) 向寄存器的 Bit6~Bit2 写入 5' b10101，进入第 3 步；向寄存器写入其他数值或向其他地址寄存写入任何值停留在第 2 步。
- (3) 向寄存器的 Bit6~Bit2 写入 5' b01010，进入第 4 步；向寄存器写入其他数值或向其他地址寄存写入任何值返回第 2 步；
- (4) 向寄存器的 Bit6~Bit2 写入 5' b10111，WPF=1，返回第 2 步；向寄存器写入其他数值或向其他地址寄存写入任何值返回第 2 步；

寄存器写使能操作步骤：

- (1) 向寄存器的 Bit6~Bit2 写入 5' b00000，复位检测序列，进入第 6 步。
- (2) 向寄存器的 Bit6~Bit2 写入 5' b11100，进入第 7 步；向寄存器写入其他数值或向其他地址寄存写入任何值停留第 6 步；
- (3) 向寄存器的 Bit6~Bit2 写入 5' b00011，进入第 8 步；向寄存器写入其他数值或向其他地址寄存写入任何值返回第 6 步；
- (4) 向寄存器的 Bit6~Bit2 写入 5' b01110，WPF=0，返回第 6 步；向寄存器写入其他数值或向其他地址寄存写入任何值返回第 6 步；

#### 4.3.16 1/1024 秒

地址	寄存器名称	D7	D6	D5	D4	D3	D2	D1	D0
FEH	1/1024 秒	128	64	32	16	8	4	2	1
FFH		0	0	0	0	0	0	512	256

1/1024 秒为 10 位只读寄存器，计数范围为 0~1023，分辨率为 1/1024 秒。

## 5、串行 IIC 接口

### 5.1、SD8801 通过两线式 IIC 串行接口方式接收各种命令并读写数据。

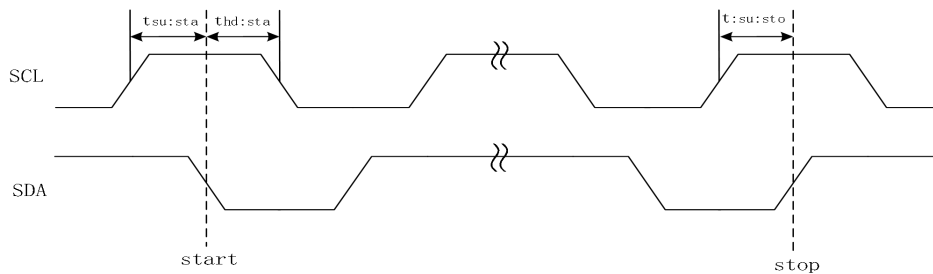
两线式串行 IIC 接口方式描述如下：

#### (1) 开始条件

当 SCL 处于高电平时，SDA 由高电平变成低电平构成一个开始条件，对 SD8801 的所有操作均必须由开始条件开始。

#### (2) 停止条件

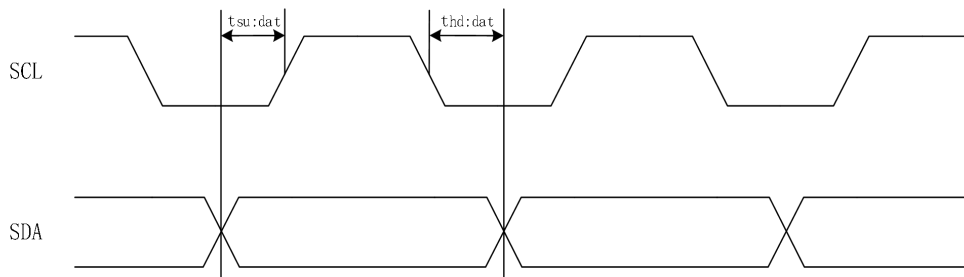
当 SCL 处于高电平时，SDA 由低电平变成高电平构成一个停止条件，对 SD8801 的所有操作均停止，系统进入待机状态。



实时时钟串行接口

#### (3) 数据传输

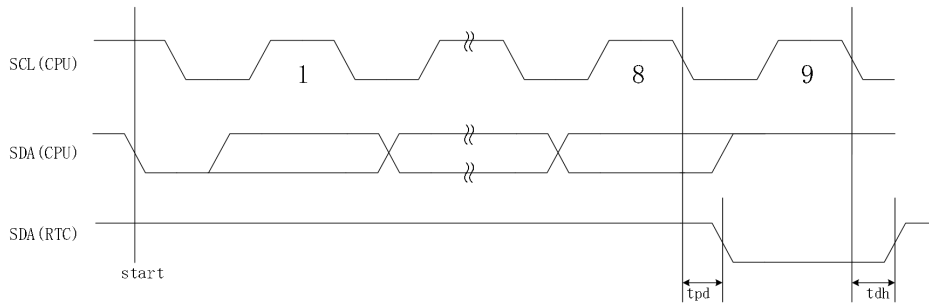
当 SCL 为低电平，且 SDA 线电平变化时，则数据由 CPU 传输给 SD8801（高位在前、低位在后，下同）；当 SCL 为高电平，且 SDA 电平保持不变时，则 CPU 读取 SD8801 发送来的数据；当 SCL 为高电平，且 SDA 电平变化时，SD8801 收到一个开始或停止条件。



实时时钟数据传输时序

#### (4) 确认

数据传输以 8 为序列进行。SD8801 在第九个时钟周期时将 SDA 置位为低电平，即送出一个确认信号（Acknowledge bit，以下简称“ACK”），表明数据已经被其收到。



实时时钟确认信号

## 5.2、数据、指令传输格式

当 CPU 发出开始条件与实时时钟建立连接后，CPU 首先通过 SDA 总线连续输出 7 位器件地址+读写位来唤醒 SD8801。

(1) 器件地址：

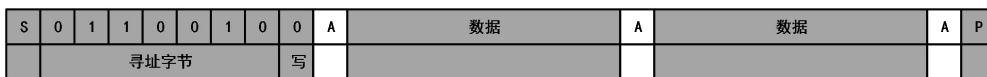
从机地址								读/写选择位
ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
65H	0	1	1	0	0	1	0	1 (读)
64H	0	1	1	0	0	1	0	0 (写)

其中高 7 位 bit7~bit1 为器件地址，固定为“0110010”；bit0 为读/写位。“1”为读操作，“0”为写操作。

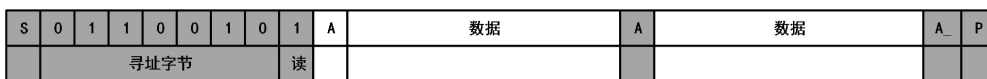
(2) 数据传输格式：

在数据发送/接收停止信号到来时，将结束其数据传输，如果只有开始信号，而没有结束信号，接着重新产生起始信号，还要重新设置器件代码（在传输方向需要改变时，就用这种传输方式，如下面的读数据方式）。

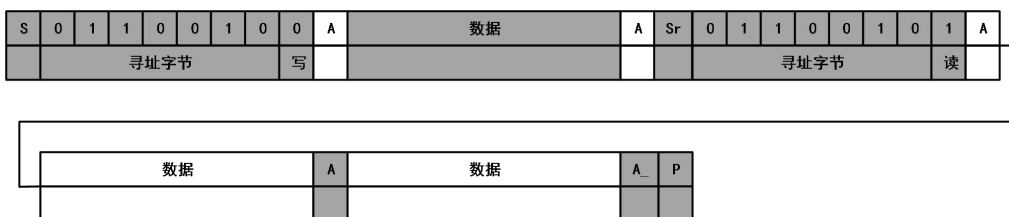
主设备向从设备写入数据过程如下图：



主设备向从设备直接读取数据过程如下图：



数据传输时改变其传输方向过程图：



■ CPU到时钟    □ 时钟到CPU    □ A    □ A    □ A\_ 响应信号

■ S 起始信号    □ P 停止信号    □ Sr 重新起始信号

### (3) SD8801 数据传输的写模式

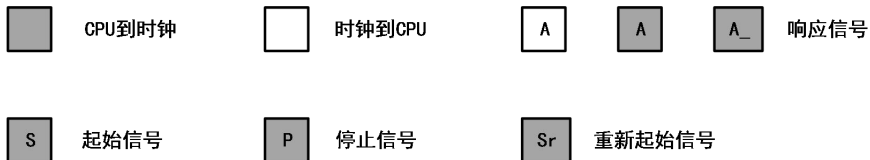
先送 7 位器件地址 (0110010)，第 8 位送入写命令 (“0”)，第 9 位是 SD8801 的响应位 (ACK)，SD8801 进入写状态。

接下来的 8 位数据 (一个字节) 是确定 SD8801 的内部地址，第 9 位才是 SD8801 的响应位。

开始写数据: 每写完 1 个字节的数据之后, 都经过 1 位的响应信号才写下 1 字节的数据, 如果要结束写数据的过程, 则在 ACK 后送出停止命令即可。

SD8801 写数据示例 (向 08H, 09H 地址写数据):

S	0	1	1	0	0	1	0	0	A	0	0	0	0	1	0	0	0	A	数据	A	数据	A	P		
寻址字节									写	设置08H内部地址								08H	09H						



特别注意:

- 1、对寄存器的写操作必须确认芯片处于写允许状态，否则写无效。具体操作细则见 4.3.15 的写保护功能。
- 2、写时间同步: 每次对实时时间秒寄存器的写操作时, 当秒数据的 8 个 bit 完全写入并收到 ACK 信号后, 就会对秒以下的内部计数器清零, 使时间同步。
- 3、从当前地址开始, 每次读写完一个字节地址自动加 1。
- 4、为了提高数据的可靠性, 当写完成后, **应将芯片置于写禁止状态。**
- 5、有关写实时时间数据的位数的特别要求请参见 (4.2.8 的注 2)。

### (4) 数据传输的读模式

1、与写模式的前两步一样。

2、重新发出开始命令以改变两线接口数据传输方向。

3、再送 7 位器件地址 (0110010)，第 8 位送入读命令 (“1”)，第 9 位是 SD8801 的响应位 (ACK)，SD8801 进入读状态。

4、开始读数据: 每读完 1 个字节的数据之后, CPU 都要送出 1 位的响应信号 (ACK 低电平) 才能读下 1 字节的数据。如果想要结束读数据过程, 则 CPU 要送出 1 位的响应信号 (ACK 高电平), ACK\_后送出停止命令即可。

SD8801 读数据方法一示例 (从 7H~9H 地址读取数据):



### (5) SD8801 在特殊条件下的数据传输

为了保证读写数据的有效性，SD8801 的两线通信开始到结束仅在此 0.5 秒之内，如此可避免总线挂死的现象。

因此在 SD8801 中，IIC 会在第一个开始信号（START）到来的 0.5 秒之后自动终止本次通信。所以，要注意：从开始信号进行读/写数据，直到停止信号，读写/操作过程必须在 0.5 秒之内完成。

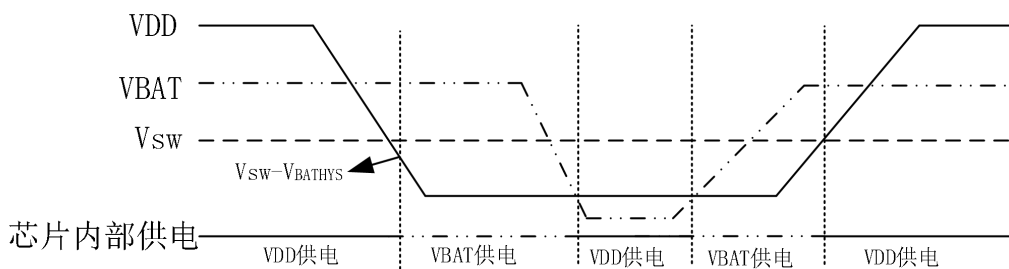
## 6、上电复位

芯片内部具有上电复位电路：当芯片第一次加上电源时，芯片内部复位。复位操作对内部分寄存器进行置初值但不包括实时时钟数据寄存器、通用 RAM。

## 7、备用电池

备用电池通过 VBAT 脚输入，芯片内部根据备用电池的电压 VBAT 和主电源的电压 VDD 进行自动切换，切换的条件如下图所示。

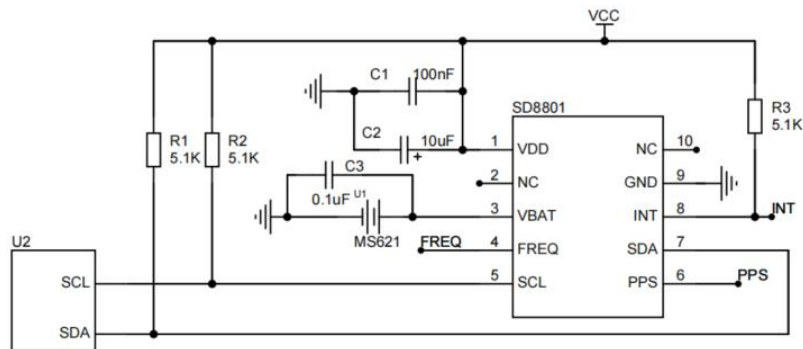
备用电池通过 VBAT 脚输入，芯片内部根据备用电池的电压 VBAT 和主电源的电压 VDD 进行自动切换，切换的条件如下图所示。



说明：(1) 当  $VDD > V_{sw} - V_{BATHYS}$  (VDD 上升时  $V_{sw}$ ) 或者  $VDD > VBAT$  时，芯片内部由 VDD 供电。

(2) 当  $VDD < V_{sw} - V_{BATHYS}$  (VDD 上升时  $V_{sw}$ ) 并且  $VDD < VBAT$  时，芯片内部由 VBAT 供电。

## 8、应用参考电路



## 9、PCB 排版

SD8801 在排 PCB 时要注意:在 SD8801 的背面不要排布大电流、强干扰线路;SCL、SDA 线分别与 MCU 用于 IIC 通讯的 I/O 口之间不要串联超过 100 欧的电阻。

由于 SD8801 内置晶振,在运输、生产环节请注意不要撞击 SD8801,也不要使用超声波清洗 SD8801,以免造成 SD8801 永久损坏。

## 10、极限参数

$V_{DD}$ 、 $V_{BAT}$ 、SCL、SDA、FREQ、PPS 和 INT 引脚上的电压（相对于地）	.....	-0.5V 至 7.0V
贮存温度	.....	-55℃ 至 +125℃
引线温度（焊接，10 秒）	.....	260℃

**注：强度超出所列的极限参数可能导致器件的永久性损坏。这些仅仅是极限参数，并不意味着在极限条件下或在任何其它超出推荐工作条件所示参数的情况下器件能有效地工作。延长在极限参数条件上的工作时间会影响器件的可靠性。**

因内置晶振的固有特性，用户使用过程中 RTC 存在晶振老化、频率偏移的现象，高温焊接会加速内置晶振的负向老化过程。

## 11、直流特性

符号	参数	条件	最小值	典型值	最大值	单位
$V_{DD}$	主电源		2.5		5.5	V
$V_{keep}$	计时电压		1.5		5.5	V
$V_{BAT}$	备用电池供电电压		1.5		5.5	V
$I_{DD1}$	主电源电流	$V_{DD}=5V$		0.6	1.2	$\mu A$
		$V_{DD}=3V$		0.5	1.0	$\mu A$
$I_{DD2}$	IIC 通信时的电源电流	$V_{DD}=5V$		40	120	$\mu A$
$I_{DD3}$	充电启用时的电源电流	$V_{DD}=5V$		80		$\mu A$
$I_{BAT}$	备用电池供电电流	$V_{BAT}=3.3V$		0.5		$\mu A$
$I_{L1}$	SCL 的输入漏电流			100		nA
$I_{L0}$	SDA 的输入/输出漏电流			100		nA

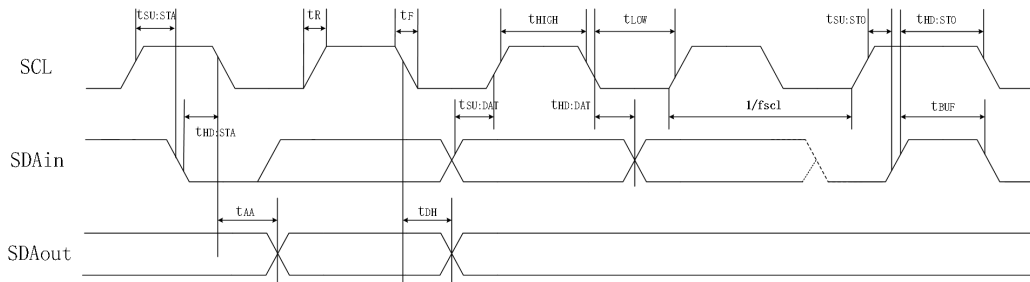
$V_{BATHYS}$	$V_{BAT}$ 和 $V_{DD}$ 之间切换的迟滞电压			85		mV
$V_{sw}$	$V_{BAT}$ 和 $V_{DD}$ 之间切换的电压	$T_a=25^\circ\text{C}$		2.4		V
$V_{OL}$	INT /SDA 低电平输出电压	$V_{DD}=5\text{V}$ $I_{OL}=0.5\text{mA}$	0.1	0.2	0.3	V
$V_{DOR}$	电源复位时 VDD 上升速率		0.1		1	V/ms
$V_{temp}$	温度补偿阈值电压			2.4		V

## 12、交流特性

符号	参数	条件	标准模式 (fSCL=100kHz)			快速模式 (fSCL=400kHz)			单位
			最小值	典型值	最大值	最小值	典型值	最大值	
$f_{SCL}$	SCL 频率				100			400	kHz
$V_{IL}$	SDA 和 SCL 低电平输入电压		-0.3		$0.3 \times V_{DD}$	-0.3		$0.3 \times V_{DD}$	V
$V_{IH}$	SDA 和 SCL 高电平输入电压		$0.7 \times V_{DD}$		$V_{DD} + 0.3$	$0.7 \times V_{DD}$		$V_{DD} + 0.3$	V
$V_{hys}$	SDA 和 SCL 施密特触发输入滞后		$0.05 \times V_{DD}$			$0.05 \times V_{DD}$			V
$V_{OL}$	低电平输出电压	SDA 输出低电平状态下, 吸收 2mA 电流时的电压		0.4			0.4		V
$C_{pin}$	SDA 和 SCL 引脚电容	$T_A=25^\circ\text{C}$ $f=1\text{MHz}$ $V_{DD}=5\text{V}$ $V_{IN}=0\text{V}$ $V_{OUT}=0\text{V}$			10			10	pF
$t_{IN}$	SDA 和 SCL 输入端的脉冲宽度抑制时间				100			50	ns
$t_{AA}$	SCL 下降沿到 SDA 输出数据有效	SCL 下降到 $0.3 \times V_{DD}$ , 直到 SDA 不在 $0.3 \times V_{DD}$ 至 $0.7 \times V_{DD}$ 区间。			900			900	ns
$t_{BUF}$	总线在 STOP 和 START 之间的空闲时间	SDA 在 STOP 条件下上升到 $0.7 \times V_{DD}$ , 在开始条件下下降到 $0.7 \times V_{DD}$ 。	4700			1300			ns
$t_{LOW}$	时钟低电平时间	在 $0.3 \times V_{DD}$ 处测量	4700			1300			ns
$t_{HIGH}$	时钟高电平时间	在 $0.7 \times V_{DD}$ 处测量	4000			600			ns
$t_{SU:STA}$	IIC 启动信号的建立时间	SCL 上升到 $0.7 \times V_{DD}$ 至 SDA 下降沿到 $0.7 \times V_{DD}$	4700			600			ns
$t_{HD:STA}$	IIC 停止信号的保持时间	在 SDA 下降沿到 $0.3 \times V_{DD}$ , SCL 下降沿到 $0.7 \times V_{DD}$	4000			600			ns
$t_{SU:DAT}$	输入数据的建立时	从 SDA 不在 $0.3 \times V_{DD}$ 至	250			100			ns



	间	$0.7 \times V_{DD}$ 范围, 到 SCL 上升沿的 $0.3 \times V_{DD}$							
$t_{HD:DAT}$	输入数据保持时间	SCL 下降沿的 $0.3 \times V_{DD}$ , 到 SDA 在 $0.3 \times V_{DD}$ 至 $0.7 \times V_{DD}$ 区间	200		100				ns
$t_{SU:STO}$	停止条件的建立时间	从 SCL 上升沿经过 $0.7 \times V_{DD}$ , 到 SDA 上升沿 $0.3 \times V_{DD}$	4000		600				ns
$t_{HD:STO}$	输出条件保持时间	从 SDA 上升沿到 SCL 下降沿。两者均超过 $0.7 \times V_{DD}$	600		600				ns
$t_{DH}$	数据输出的保持时间	从 SCL 下降沿 $0.7 \times V_{DD}$ , 到 SDA $0.3 \times V_{DD}$ 至 $0.7 \times V_{DD}$ 区间	0		0				ns
$t_R$	SDA 和 SCL 的上升时间	$0.3 \times V_{DD}$ 至 $0.7 \times V_{DD}$ 区间			1000			300	ns
$t_F$	SDA 和 SCL 的下降时间	$0.3 \times V_{DD}$ 至 $0.7 \times V_{DD}$ 区间			300			300	ns

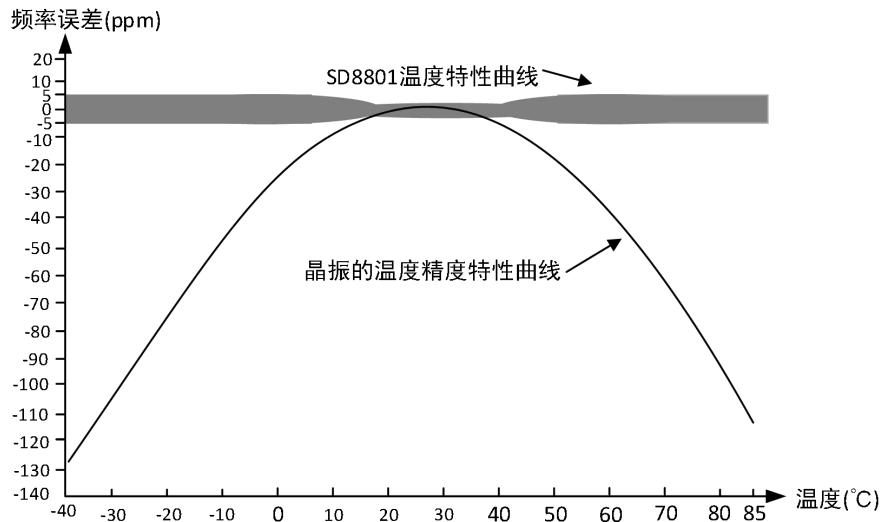


IIC 时序图

### 13、频率特性

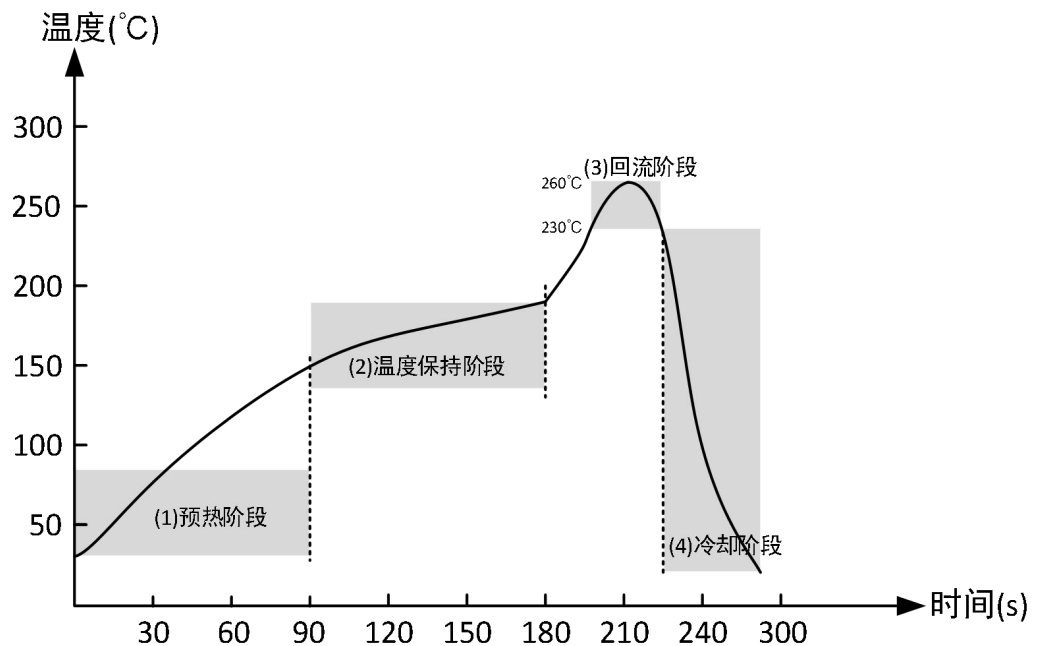
符号	参数	条件	最小值	典型值	最大值	单位	备注
$\Delta f/f$	频率精度	$T_a = -40^\circ\text{C} \sim +85^\circ\text{C}$	-5.0		5.0	ppm	出货精度
		$T_a = +86^\circ\text{C} \sim +105^\circ\text{C}$	-10.0		10.0	ppm	
		$T_a = +106^\circ\text{C} \sim +125^\circ\text{C}$	-20.0		20.0	ppm	
$f/V$	频率电压特性	$T_a = 25^\circ\text{C}, V_{DD} = 2.5 \text{ to } 5.5 \text{ V}$	-1.0		1.0	ppm/V	
tSTA	晶振启动时间	$T_a = 25^\circ\text{C}, V_{DD} = 3.3 \text{ V}$			1.0	s	
fa	老化率	$T_a = 25^\circ\text{C}, V_{DD} = 3.3 \text{ V}$	-3.0		3.0	ppm/year	

## 14、频率误差&温度关系曲线(与没有温补功能的时钟芯片进行对比)



## 15、焊接要求

回流焊温度曲线，包括四个阶段：预热阶段、温度保持阶段、回流阶段和冷却阶段。



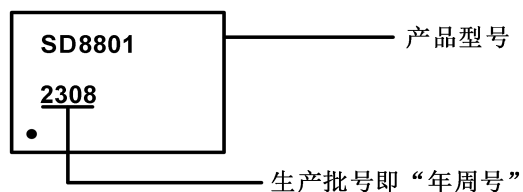
**预热阶段：**慢慢加热芯片和电路板，防止热冲击。温度从室温上升到大约 150°C，时间通常为 60-120 秒。

**温度保持阶段：**保持温度在 150°C-180°C，这个阶段有助于活化焊膏中的助焊剂，通常持续 60-120 秒。

**回流阶段：**温度快速升高至峰值 255-265°C，这是焊接的关键阶段，焊膏熔化形成焊点。在 230 摄氏度以上持续约 20-40 秒。

**冷却阶段：**温度迅速下降到室温，冷却速度一般为 3-4°C/秒，以确保焊点稳定。

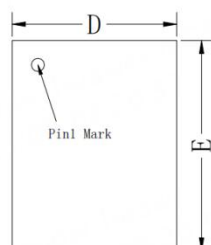
## 16、芯片顶部字符说明



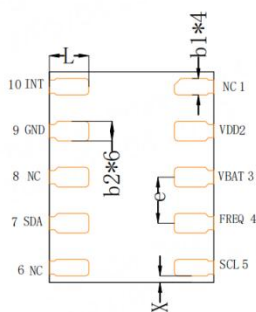
## 17、封装尺寸（单位：毫米）

SD8801 3225 封装尺寸图

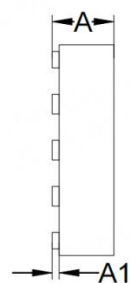
Package Top View



Package Bottom View



Package Side View



符号	尺寸 (mm)		
	最小值	平均值	最大值
A	0.720	0.750	0.780
A1	0.007	0.012	0.017
D	2.400	2.500	2.600
E	3.100	3.200	3.300
e	0.650	0.700	0.750
b1	0.200	0.250	0.300
b2	0.250	0.300	0.350
L	0.550	0.600	0.650
X1	0.050	0.100	0.150

备注：SD8801 为湿敏一级 (MSL1), 真空盘带包装 4000PCS/盘。



## ■ 编后语

感谢您阅读本资料。由于经验和水平的欠缺，本文难免有错误和遗漏。如果您在使用过程中发现错误或不恰当的地方，请拨打电话：0755-83246178 或请 E-mail: support@whwave.com.cn, 我们将尽快予以答复。

感谢您的支持与合作！

注：

本资料中的内容如有变化，恕不另行通知。

本资料提供的应用线路及程序仅供参考，本公司不承担由此而引起的任何损失。

由于本公司的产品不断更新和提高,希望您经常与本公司联系，以索取最新资料。

本公司不承担任何使用过程中引起的侵犯第三方专利和其它权利的责任。

注：本文档受中国版权法保护,非授权禁止拷贝、复制、引用或传播

(SD 及 WAVE 均为我公司注册商标)

深圳市兴威帆电子技术有限公司