

# S3C44B0X 中文数据手册

## 目 录

S3C44B0X 中文数据手册 .....	1
14 RTC (实时时钟) .....	2
14. 1 闰年产生器 .....	2
14. 2 读/写寄存器 .....	2
14. 3 备用电池操作 .....	2
14. 4 报警功能 .....	2
14. 5 节拍中断 .....	3
14. 6 循环复位功能 .....	3
14. 7 实时时钟控制寄存器 (RTCCON) .....	3

## 14 RTC (实时时钟)

实时时钟 (RTC) 单元可以通过备用电池供电, 因此, 即使系统电源关闭, 它也可以继续工作。RTC 可以通过 STRB/LDRB 指令将 8 位 BCD 码数据送至 CPU。这些 BCD 数据包括秒, 分, 时, 日期, 星期, 月和年。RTC 单元通过一个外部的 32.768KHz 晶振提供时钟, 并具有定时报警的功能。

### 14. 1 闰年产生器

这个模块可以决定每个月的最后日期是 28, 29, 30 还是 31, 并可以根据是否是闰年来决定日期。由于 8 位的计数器只能表示 2 个 BCD 码, 因此它不能判断 00 年究竟是不是闰年。例如它不能够判断 1900 年和 2000 年的差别。为了解决这个问题, S3C44B0X 内的 RTC 模块中有一个固定的逻辑, 用来支持 2000 年为闰年。请注意虽然 2000 年是闰年, 但 1900 年不是闰年。因此, S3C44B0X 中 00 代表 2000 年, 而不是 1900 年。

### 14. 2 读/写寄存器

要求设置 RTCON 寄存器的 0 位来表示读和写 RTC 模块中的寄存器。为了显示秒, 分, 小时, 日期, 月和年, CPU 会从 BCDSEC, BCDMIN, BCDHOUR, BCDDAY, BCDDATE, BCDMON, 和 BCDYEAR 寄存器中读取数据。但是, 由于多个寄存器的读取, 可能产生 1 秒钟的偏离。例如, 如果用户读取寄存器 BCDYEAR 到 BCDMIN, 假设结果为 1959 年, 12 月, 31 日, 23 点, 59 分。在用户读取 BCDSEC 寄存器时, 如果结果是 1~59, 肯定没有问题。但如果结果是 0, 那么很有可能年、月、日、时、分已经变成了 1960 年 1 月 1 日 0 时 0 分, 这就是上面所说的 1 秒偏离。解决的方法是, 当读取到的 BCDSEC 等于 0 时, 用户应该再读取一次 BCDYEAR 到 BCDSEC 的值。

### 14. 3 备用电池操作

RTC 逻辑模块可以通过一个备用电池供电。备用电池的阳极通过 RTCVDD 脚接至内部的 RTC 模块, 即使系统电源关闭, 也能够提供电能保证 RTC 模块正常工作。在系统电源关闭时, CPU 和 RTC 逻辑模块之间的接口被锁住, 使备用电池仅需驱动 RTC 的晶振电路和 BCD 计数器, 从而使功耗降到最低。

### 14. 4 报警功能

在掉电模式或正常工作模式下, RTC 能够在指定的时间产生一个报警信号。在正常工作模式下, 报警中断 (ALMINT) 被激活。在掉电模式下, 电源管理苏醒信号也与 ALMINT

一样处于激活状态。RTC 的报警寄存器：RTCALM 可以决定报警的使能或禁止和报警时间的设置条件。

## 14. 5 节拍中断

RTC 节拍时间用于中断请求。TICNT 寄存器具有一个中断使能位，同时其中的计数值用于中断。当计数值到达 0 时，节拍时间中断就会触发。中断的间隔时间计算如下：

$$\text{Period}=(n+1)/128 \text{ 秒}$$

n: 节拍时间计数值 (1~127)

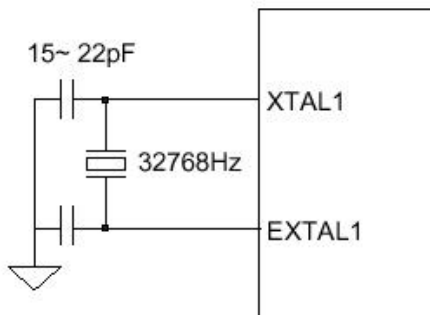
这个 RTC 时间节拍中断功能可以作为 RTOS (实时操作系统) 内核的时间节拍。如果节拍从 RTC 时间节拍产生，则 RTOS 内部与时间相关的功能将一直与实时时钟同步。

## 14. 6 循环复位功能

循环复位功能可以通过 RTC 循环复位寄存器 RTCRST 来操作。你可以选还循环边界时间 (30, 40, 或 50 秒), 在循环复位操作中, 这个时间将加载到秒进位产生器, 当秒计时到达该时间时就进位。例如, 如果当前时间是 23: 37: 47, 循环边界选择为 40 秒, 则循环复位操作将当前时间修改为 23: 38: 00。

**注意:** 所有的 RTC 寄存器必须使用 STRB, LDRB 指令或 char 类型指针, 以字节方式操作。

下图为 32.768KHZ 晶振的电路连接示例:



## 14. 7 实时时钟控制寄存器 (RTCCON)

RTCCON 寄存器包括 4 个位例如 RTCEN, 用来控制对 BCD 寄存器的读/写使能。CLKSEL, CNTSEL 和 CLKRST 用来测试。

RTCEN 位可以控制所有 CPU 和 RTC 之间的所有接口, 因此在 RTC 控制程序中, 应当将它置 1 从而使得在系统复位之后能够读写这些寄存器。而在电源关闭之前, RTCEN 位依你更改清 0 从而阻止无意中对 RTC 寄存器的写入。

RTCCON	Bit	Description	Initial State
RTCEN	[0]	RTC read/write enable 0 = Disable, 1 = Enable	0
CLKSEL	[1]	BCD clock select 0 = XTAL 1/2 <sup>15</sup> divided clock 1 = Reserved (XTAL clock)	0
CNTSEL	[2]	BCD count select 0 = Merge BCD counters 1 = Reserved (Separate BCD counters)	0
CLKRST	[3]	RTC clock count reset 0 = No reset, 1 = Reset	0

注意：所有的 RTC 寄存器都应该以字节为单位进行操作。

RTCALM 寄存器决定了是否使能报警功能，以及报警时间。注意在掉电模式下，RTCALM 寄存器通过 ALMINT 和 PMWKUP 产生报警信号，但在正常工作模式下只有 ALMINT 模式。

RTCALM	Bit	Description	Initial State
Reserved	[7]		0
ALMEN	[6]	Alarm global enable 0 = Disable, 1 = Enable	0
YEAREN	[5]	Year alarm enable 0 = Disable, 1 = Enable	0
MONREN	[4]	Month alarm enable 0 = Disable, 1 = Enable	0
DAYEN	[3]	Day alarm enable 0 = Disable, 1 = Enable	0
HOUREN	[2]	Hour alarm enable 0 = Disable, 1 = Enable	0
MINEN	[1]	Minute alarm enable 0 = Disable, 1 = Enable	0
SECEN	[0]	Second alarm enable 0 = Disable, 1 = Enable	0

ALMSEC~ALMYEAR (略)

RTCRST	Bit	Description	Initial State
SRSTEN	[3]	Round second reset enable 0 = Disable, 1 = Enable	-
SECCR	[2:0]	Round boundary for second carry generation. (note) 011 = over than 30 sec 100 = over than 40 sec 101 = over than 50 sec	-

BCDSEC~BCDYEAR (略)

TICNT	Bit	Description	Initial State
TICK INT ENABLE	[7]	Tick time interrupt enable 0 = disable 1 = enable	0
TICK TIME COUNT	[6:0]	Tick time count value. (1~127) This counter value decrease internally, users can not read this real counter value on processing.	000000