

HT47C20 实时时钟（Real Time Clock）使用介绍

作者： 盛扬半导体（上海）有限公司软件部
日期： 2001/8/6
适用单片机： HT47C20、HT47C20L

本文主要介绍 HT47C20 单片机实时时钟（Real Time Clock）的使用及注意事项。

介绍

HT47C20 单片机具有一个实时时钟（Real Time Clock），它可以产生周期性的内部中断。实时时钟的时钟来源（Clock source）可由掩膜选择设定为 WDT 时钟、RTC 时钟或指令时钟（系统时钟/4）；其超时周期可由程序指令设定为“时钟来源”/2⁸~“时钟来源”/2¹⁵。如果实时时钟产生超时的现象，则其对应的中断请求标志（RTF）会被置位，并且产生一个中断服务到 0CH 的地址。实时时钟的超时信号，也可提供给定时/计数器做为时钟来源，以便得到更长的超时时间周期。

如果实时时钟的时钟来源选择为 WDT 时钟或 RTC 时钟，则当系统进入 HALT 模式后，实时时钟会继续计数；如果选择时钟来源为指令时钟，则在 HALT 模式中实时时钟会停止计数。使用时要注意以上区别。

当选定时钟来源后，实时时钟的超时周期由软件设定。寄存器 RTCC 是一个实时时钟控制寄存器，将数据写入 RT2、RT1 和 RT0（RTCC 第 2、第 1、第 0 位）之中，会产生不同的超时周期，如下表所示：

RTCC 寄存器

寄存器	位	名称	读/写	复位	功能
RTCC (09H)	0	RT0	读/写	1	八选一输入控制，用来决定实时时钟 预分频器的级数
	1	RT1		1	
	2	RT2		1	
	3	—	—	—	未定义，此位必须设置为“0”
	4	SAVE	读/写	0	RTC OSC 振荡器的快速振荡控制位 “0” 允许 “1” 禁止 上电后，将此位设置为“1”可以降低 电源损耗
	5	—	—	—	未定义，读取值未知
	6, 7	—	—	—	未定义，读取时为“0”

实时时钟中断周期表

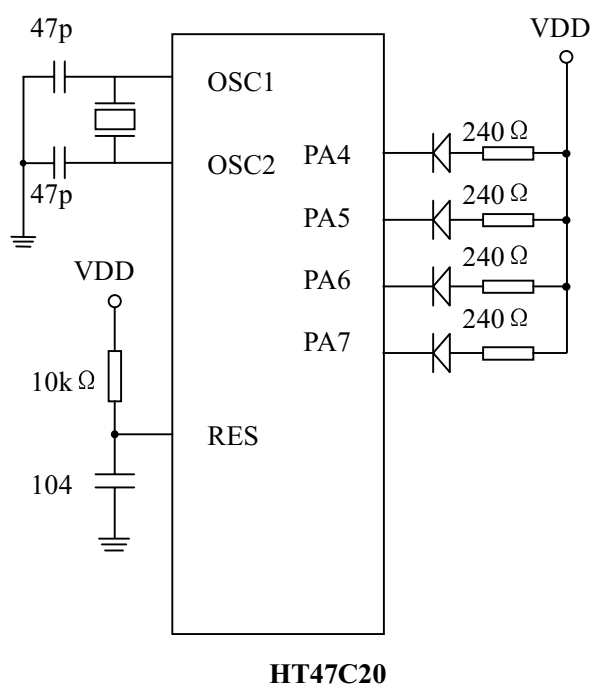
RTC 分频级数				Clock Source		
RT2	RT1	RT0		12kHz (WDTCLK)	32.768kHz (RTC)	System frequency/4
0	0	0	2^8	21.33 ms	7.8125 ms	2.133 ms
0	0	1	2^9	42.66 ms	15.625 ms	4.266 ms
0	1	0	2^{10}	85.33 ms	31.250 ms	8.533 ms
0	1	1	2^{11}	170.66 ms	62.500 ms	17.066 ms
1	0	0	2^{12}	341.33 ms	125.000 ms	34.133 ms
1	0	1	2^{13}	682.66 ms	250.000 ms	68.266 ms
1	1	0	2^{14}	1365.33 ms	500.000 ms	136.533 ms
1	1	1	2^{15}	2730.66 ms	1000.000 ms	273.066 ms

注：System clock 为 480kHz

实时时钟（Real Time Clock）的使用

硬件部分：

PA3~PA7 引脚分别接发光二极管 LED，如图：



掩膜设定：

clock source 设定为 32KHz（RTC）

软件部分：

下面的程序介绍了 RTC 的具体使用。此程序选择 RTC 的晶振频率为 32.768kHz，超时周期为“时钟来源/ 2^{15} ”，即 RTC 每一秒钟产生一次中断。程序用 RTC 中断控制 LED 循环发光，形成一个流水灯。

程序清单:

```
include ht47c20.inc
data .section 'data'
count db ?
code .section at 0 'code'
    org    00h
    jmp     start
    org    04h
    reti
    org    08h
    reti
    org    0ch                ; 实时时钟中断入口地址
    jmp     rtc_int
    org    10h
    reti

; -----
start:
    clr     intc0
    clr     intc1
    mov     a,77h
    mov     count,a
    mov     a,07h            ; 设定实时时钟超时周期
    mov     rtcc,a
    set     intc0.3          ; 实时时钟中断允许
    set     intc0.0          ; 总中断允许
    jmp     $                ; 原地踏步, 等待实时时钟中断

; -----
rtc_int:                    ; 实时时钟中断服务子程序
    rl     count
    mov     a,count
    mov     pa,a            ; 输出到 PA 口, 点亮 LED
    reti
```