

# **FT62F08X**

## **Application note**

## 目录

1. 慢时钟测量.....	3
2 应用范例 .....	6

## FT62F08x 慢时钟测量的应用

### 1. 慢时钟测量

芯片集成了两个内部 RC 振荡器，一个是经过出厂校准的高速高精度的 16M 快时钟 HIRC，一个是低速低功耗的 32K 时钟 LIRC，利用慢时钟测量功能可以把 LIRC 的周期用系统时钟计算出来。此功能可以比较精准的测量内部慢时钟周期。

#### 1.1. 测量原理

慢时钟测量类似于定时器的捕捉模式，处于这种模式下，被测量时钟 LIRC 的边沿（任意沿）将会触发定时器，在另一高速时钟（如 HIRC）的作用下开始计数，在此后的第 2 个（或第 8 个，平均模式时）LIRC 边沿到来时，定时器停止计数，同时把定时器的值锁存到 SOSCPRH/L 寄存器。

注意：慢时钟测量使用的定时器是 Timer2。

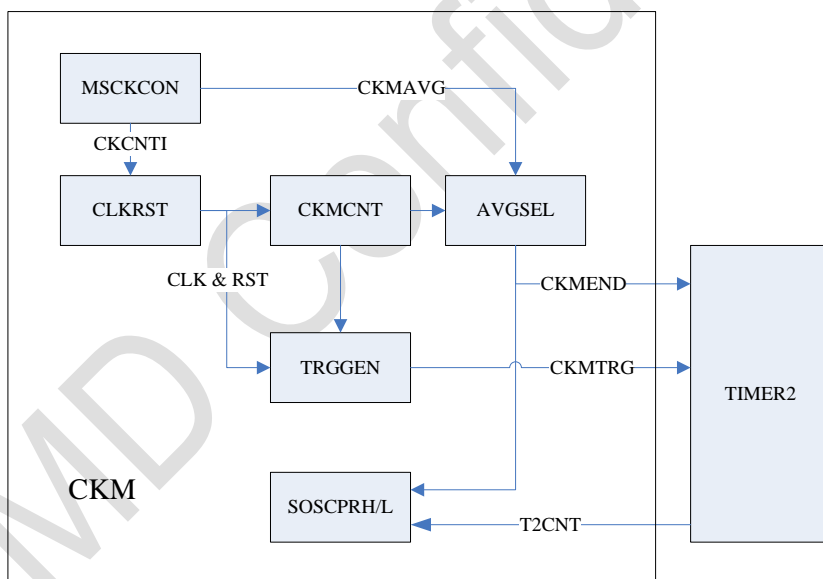


图 19.1 慢时钟测量模式原理框图

注意：

1. 在慢时钟测量过程中软件不要写 SOSCPRH/L；
2. 不要在单步调试下做慢时钟测量，因为暂停模式下 Timer2 被停止，这样会导致测量结果不正确；
3. 若 SYSON bit 为 0 时，慢时钟测量无法在 SLEEP 模式下进行，不要在测量运行时进入 SLEEP 模式。

## 1.2. 上电自动测量

在上电后慢时钟测量将会自动启动，此时 CKCNTI 置 1，CKMAVG 为 0，打开 LIRC 和 HIRC。Timer2 的时钟被自动配置为 16M 的内部高速时钟，即类似设置 T2CKSRC 为 001、TIM2EN=1 的功能，但未配置这些位。Timer2 使用默认配置，无需置位 CEN 使能 Timer2 计数，此时不能配置 Timer2。

在测量过程中，实际应用程序已在运行，若要使用 Timer2 则需要查询 CKCNTI。若 CKCNTI 为 0 即可使用 Timer2，此时 SOSCPR 寄存器的值为有效值，其单位为  $F_{HSI}$  时钟的个数。

注意：

1. 上电自动测量不会置位 CKM 中断标志；

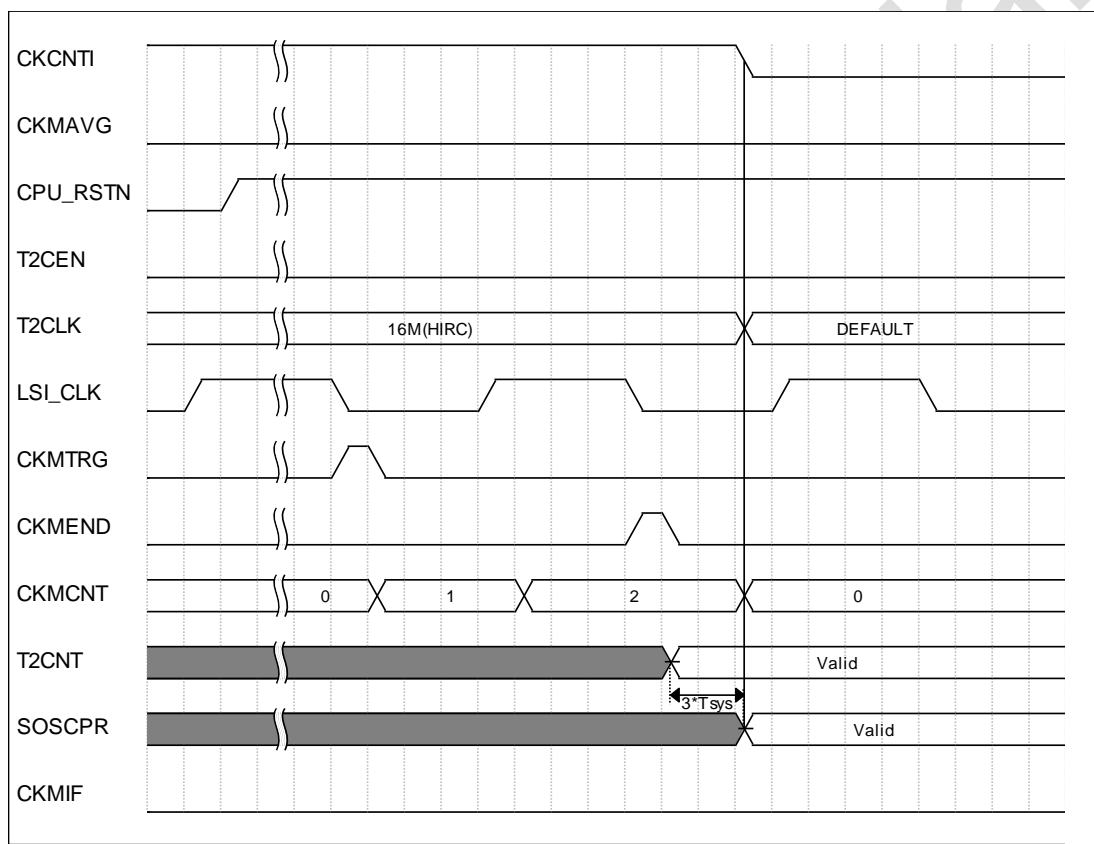


图 19.2 上电慢时钟自动测量时序图

## 1.3. 操作步骤

1. 为提高计量精度，建议设置 T2CKSRC 为 001，TIM2EN=1，选择 16M 的内部高速时钟；
2. 关闭 Timer2 的相关中断使能，设置 TIM2ARRH/L 为最大值，设置 TIM2PSC 为 0000；
3. 设置 TIM2CR1 为复位值，再将 CEN 置 1，使能 Timer2；
4. 如果选择 4 次平均，则把 MSCKCON.1 置 1，否则把它清 0；
5. 置位 MSCKCON.0，开始测量；
6. 测量结束后 MSCKCON.0 自动清 0，中断标志置 1；
7. 可以用查询或中断的方式等待结束；

8. 当查询到中断标志为 1 时读取得到的 SOSCPR 即为最终结果。

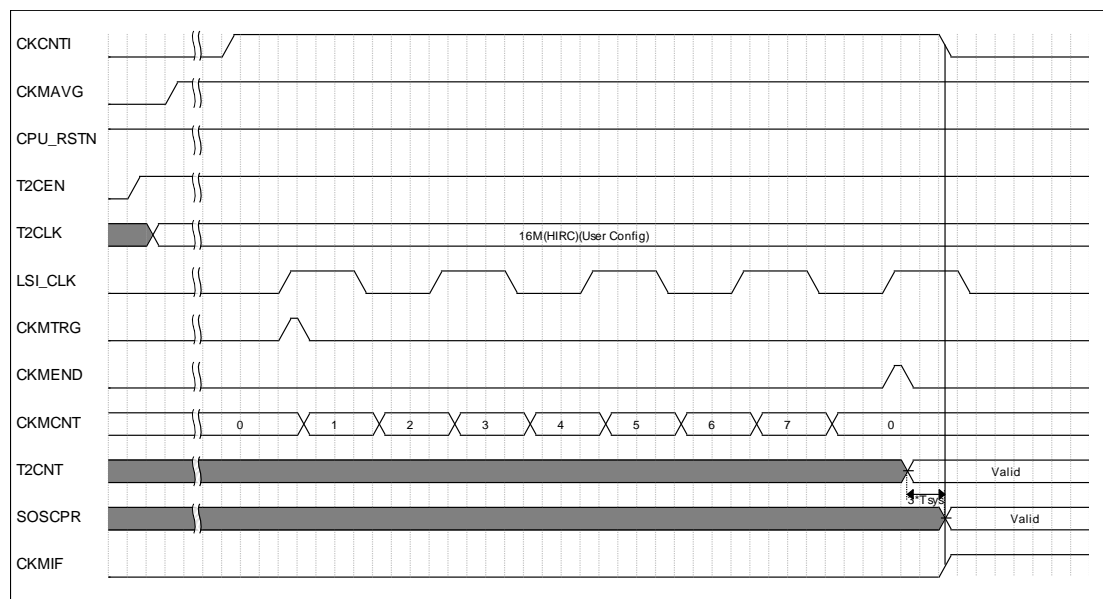


图 19.3 慢时钟测量模式时序图

## 1.4. 与慢时钟测量相关寄存器汇总

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
MSCKCON	0x41D	—	—	—	—	—	—	CKMAVG	CKCNTI	---- --01
SOSCPR	0x41E	SOSCPR[7:0]								1111 1111
SOSCPRH	0x41F	—				SOSCPR[11:8]				---- 1111

### 1.4.1. MSCKCON 寄存器，地址 0x41D

Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	CKMAVG	CKCNTI
Reset	—	—	—	—	—	—	0	1
Type	RW	RW	RW	RW	RO-0	RW	RW	RW

Bit	Name	Function
7:2	N/A	保留位，读 0
1	CKMAVG	快时钟测量慢时钟周期的测量平均模式 1 = 打开平均模式（自动测量并累加 4 次） 0 = 关闭平均模式
0	CKCNTI	Clock Count Init –使能快时钟测量慢时钟周期 1 = 使能快时钟测量慢时钟周期 0 = 关闭快时钟测量慢时钟周期

		注：这一位在测量完毕后会自动归零
--	--	------------------

## 1.4.2. SOSCPR 寄存器，地址 0x41E, 41F

SOSCPRL, 地址 0x41E

Bit	7	6	5	4	3	2	1	0
Name	SOSCPR[7:0]							
Reset	8'hFF							
Type	RW							

SOSCPRH, 地址 0x41F

Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	SOSCPR[11:8]			
Reset	—	—	—	—	4'hF			
Type	RO-0	RO-0	RO-0	RO-0	RW			

Bit	Name	Function
0x41E: 7:0	SOSCPR[11:0]	低速振荡器周期（单位：Timer2 时钟周期数）
0x41F: 3:0		用于慢时钟测量功能， $T_{LSI} = SOSCPR * T_{TIM2}$

## 2 应用范例

```

//*****
/* 文件名: TEST_62F08x_MSCK.c
* 功能: FT62F08x-MSCK 功能演示
* IC: FT62F088 LQFP32
* 内部: 16M
* empno: 500
* 说明: 测量慢时钟
*
*
*
* 参考原理图 TEST_62F08x_sch.pdf
*/
//*****

```

```

#include "SYSCFG.h"
/*****
/*****宏定义*****/

#define uchar      unsigned char
#define uint       unsigned int
#define ulong      unsigned long

#define DemoPortOut RB3
#define DemoPortIn  RC3

//volatile uchar mydata; //全局查看变量定义
volatile uint TestBuff;

/*-----
* 函数名: interrupt ISR
* 功能:   中断处理, 包括定时器 0 中断和外部中断
* 输入:   无
* 输出:   无
-----*/
void interrupt ISR(void)           //PIC_HI-TECH 使用
{

}

/*-----
* 函数名: POWER_INITIAL
* 功能:   上电系统初始化
* 输入:   无
* 输出:   无
-----*/
void POWER_INITIAL (void)
{

    OSCCON = 0B01110001; //WDT 32KHZ IRCF=111=16MHZ/2
                        //Bit0=1,系统时钟为内部振荡器
                        //Bit0=0,时钟源由 FOSC<2: 0>决定即编译选项时选择

    INTCON = 0;          //暂禁止所有中断

    PORTA = 0B00000000;
    TRISA = 0B11111111; //PA 输入输出 0-输出 1-输入
    PORTB = 0B00000000;
    TRISB = 0B11110111; //PB 输入输出 0-输出 1-输入
    PORTC = 0B00000000;
    TRISC = 0B11111111; //PC 输入输出 0-输出 1-输入

```

```

PORTD = 0B00000000;
TRISD = 0B11111111;    //PD 输入输出 0-输出 1-输入

WPUA = 0B00000000;    //PA 端口上拉控制 1-开上拉 0-关上拉
WPUB = 0B00000000;    //PB 端口上拉控制 1-开上拉 0-关上拉
WPUC = 0B00001000;    //PC 端口上拉控制 1-开上拉 0-关上拉
WPUD = 0B00000000;    //PD 端口上拉控制 1-开上拉 0-关上拉

WPDA = 0B00000000;    //PA 端口上拉控制 1-开下拉 0-关下拉
WPDB = 0B00000000;    //PB 端口上拉控制 1-开下拉 0-关下拉
WPDC = 0B00000000;    //PC 端口上拉控制 1-开下拉 0-关下拉
WPDD = 0B00000000;    //PD 端口上拉控制 1-开下拉 0-关下拉

PSRC0 = 0B11111111;    //PORTA,PORTB 源电流设置最大
//BIT7~BIT6:PORTB[7:4]源电流能力控制,BIT5~BIT4:PORTB[3:0]源电流能力控制
//BIT3~BIT2:PORTA[7:4]源电流能力控制,BIT1~BIT0:PORTA[3:0]源电流能力控制

PSRC1 = 0B11111111;    //PORTC,PORTD 源电流设置最大
//BIT7~BIT6:PORTD[7:4]源电流能力控制,BIT5~BIT4:PORTD[3:0]源电流能力控制
//BIT3~BIT2:PORTC[7:4]源电流能力控制,BIT1~BIT0:PORTC[3:0]源电流能力控制

PSINK0 = 0B11111111;    //PORTA 灌电流设置最大 0:最小, 1:最大
PSINK1 = 0B11111111;    //PORTB 灌电流设置最大 0:最小, 1:最大
PSINK2 = 0B11111111;    //PORTC 灌电流设置最大 0:最小, 1:最大
PSINK3 = 0B11111111;    //PORTD 灌电流设置最大 0:最小, 1:最大

ANSELA = 0B00000000;    //全为数字管脚

}
/*-----
* 函数名称: DelayUs
* 功能: 短延时函数 --16M-2T--大概快 1%左右.
* 输入参数: Time 延时时间长度 延时时长 Time Us
* 返回参数: 无
-----*/
void DelayUs(unsigned char Time)
{
    unsigned char a;
    for(a=0;a<Time;a++)
    {
        NOP();
    }
}
/*-----

```



- \* 函数名称: DelayMs
- \* 功能: 短延时函数
- \* 输入参数: Time 延时时间长度 延时时长 Time ms
- \* 返回参数: 无

-----\*/

void DelayMs(unsigned char Time)

```
{
    unsigned char a,b;
    for(a=0;a<Time;a++)
    {
        for(b=0;b<5;b++)
        {
            DelayUs(197); //快 1%
        }
    }
}
```

/\*-----

- \* 函数名称: DelayS
- \* 功能: 短延时函数
- \* 输入参数: Time 延时时间长度 延时时长 Time S
- \* 返回参数: 无

-----\*/

void DelayS(unsigned char Time)

```
{
    unsigned char a,b;
    for(a=0;a<Time;a++)
    {
        for(b=0;b<10;b++)
        {
            DelayMs(100);
        }
    }
}
```

/\*-----

- \* 函数名: SlowTimeTest
- \* 功能: 快时钟测量慢时钟
- \* 输入: 无
- \* 输出: 慢时钟时钟测量值 TestTime  
不开平均模式慢时钟频率=16M/TestTime(2T)  
开平均模式慢时钟频率=16M/TestTime/4(2T)

-----\*/

uint SlowTimeTest(void)

```
{
    PCKEN |= 0B00000100; //TIM2 时钟模块使能
```

```
TCKSRC= 0B00010000; //TIM2 时钟 HIRC
T2CEN = 1;           //开定时器 2
CKMAVG = 1;          //关闭平均模式
                        //注:打开平均模式输出数据为四个周期的时钟数(单周期*4)
CKCNT1 = 1;          //使能快时钟测量位,开始测量
NOP();
while(!CKMIF);
CKMIF=0;
return (uint)(SOSCPRH<<8|SOSCPRL);
}
/*-----
* 函数名: main
* 功能:   主函数
* 输入:   无
* 输出:   无
-----*/
void main(void)
{
    POWER_INITIAL();           //系统初始化
    while(1)
    {
        TestBuff = SlowTimeTest(); //时钟测量值
                                    //TestBuff≈488(理论值=16000000/32000=500)
        NOP();
        NOP();
        NOP();
        DelayMs(200);           //延时 200ms
    }
}
```

Fremont Micro Devices (SZ) Limited

#5-8, 10/F, Changhong Building, Ke-Ji Nan 12 Road, Nanshan District, Shenzhen, Guangdong 518057

Tel: (86 755) 86117811

Fax: (86 755) 86117810

Fremont Micro Devices (Hong Kong) Limited

#16, 16/F, Blk B, Veristrong Industrial Centre, 34-36 Au Pui Wan Street, Fotan, Shatin, Hong Kong

Tel: (852) 27811186

Fax: (852) 27811144

Fremont Micro Devices (USA), Inc.

42982 Osgood Road Fremont, CA 94539

Tel: (1-510) 668-1321

Fax: (1-510) 226-9918

Web Site: <http://www.fremontmicro.com/>

\* Information furnished is believed to be accurate and reliable. However, Fremont Micro Devices, Incorporated (BVI) assumes no responsibility for the consequences of use of such information or for any infringement of patents of other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent rights of Fremont Micro Devices, Incorporated (BVI). Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. Fremont Micro Devices, Incorporated (BVI) products are not authorized for use as critical components in life support devices or systems without express written approval of Fremont Micro Devices, Incorporated (BVI). The FMD logo is a registered trademark of Fremont Micro Devices, Incorporated (BVI). All other names are the property of their respective own.