XHSC 小华半导体
XIAOHUA SEMICONDUCTOR

32 位微控制器

HC32 系列 RTC 补偿功能和实时时钟补偿方法

应用笔记

Rev1.02　2025 年 04 月

适用对象

产品系列	产品型号	产品系列	产品型号	产品系列	产品型号
HC32L110	ALL	**HC32L17x**	ALL	**HC32L19x**	ALL
HC32L07x	ALL	**HC32L16x**	ALL	**HC32L18x**	ALL
HC32L13x	ALL	**HC32F052**	ALL	**HC32F472**	ALL
HC32F460	ALL	**HC32F4A0**	ALL	**HC32F448**	ALL
HC32F45x	ALL	**HC32A460**	ALL	**HC32A4A0**	ALL
HC32A472	ALL	**HC32A448**	ALL	**HC32M441**	ALL

声　明

★ 小华半导体有限公司（以下简称：“XHSC”）保留随时更改、更正、增强、修改小华半导体产品和/或本文档的权利，恕不另行通知。用户可在下单前获取最新相关信息。XHSC 产品依据购销基本合同中载明的销售条款和条件进行销售。

★ 客户应针对您的应用选择合适的 XHSC 产品，并设计、验证和测试您的应用，以确保您的应用满足相应标准以及任何安全、安保或其它要求。客户应对此独自承担全部责任。

★ XHSC 在此确认未以明示或暗示方式授予任何知识产权许可。

★ XHSC 产品的转售，若其条款与此处规定不同，XHSC 对此类产品的任何保修承诺无效。

★ 任何带有“®”或“™”标识的图形或字样是 XHSC 的商标。所有其他在 XHSC 产品上显示的产品或服务名称均为其各自所有者的财产。

★ 本通知中的信息取代并替换先前版本中的信息。

目　录

1 概述

带有实时时钟的应用通常在硬件板上外置 32768Hz 高精度 ppm 的石英晶体，其在一定温度范围内的温度特性曲线近似抛物线。应用上需要根据当前温度计算出晶体的 ppm 偏差，并通过微处理器内部的 RTC 模块补偿，实现实际输出 ppm 的校正。

本篇应用笔记主要包括：

- 小华微处理器 RTC 模块的补偿功能
- 晶体温度特性曲线
- 实时时钟补偿方法

注意：

– 本应用笔记为小华半导体 MCU 的应用补充材料，不能代替参考手册，具体功能及寄存器的操作等相关事项请以参考手册为准。

2 RTC 补偿功能介绍

本章节介绍小华微处理器 RTC 模块的补偿功能。

小华微处理器 RTC 模块的补偿功能有两种方法：

- 基于自身时钟的误差补偿，即一定时间（如 32 秒）内进行平均补偿，在某些微处理器用户手册中也被称为分布式补偿
- 基于高速时钟的误差补偿，即每秒补偿，在某些微处理器用户手册中也被称为均匀式补偿

本应用手册介绍及使用的是第一种，即基于自身时钟，在一定时间内平均补偿的方法。

2.1 RTC 补偿原理

在计数器选择 32768Hz 的时钟计数下，如果需要对每秒精度进行补偿，只能按照 32768Hz 的整数周期补偿，则每秒补偿的最小单位为（1 / 32768）* 10^6 = 30.5ppm，无法满足高精度的要求。

为了在 32768Hz 的计数时钟下实现高精度的要求，需要在算法上做调整。

本应用笔记中涉及的 RTC 模块采用将补偿周期扩大到 2^n 秒的方法，则每秒补偿精度计算公式如下：

$$补偿精度 = （1 / 32768）/ 2^n * 10^6 \ ppm$$

公式 1 补偿精度

- 当 n=5 时，补偿周期扩大到 32 秒，每秒补偿精度为（1 / 32768）/ 2^5 * 10^6 = 0.954ppm

这样在补偿周期内实现了更高的补偿精度。

2.2 RTC 补偿功能实现

RTC 补偿功能在实现上通过在寄存器中引入小数设定的方法。

例如：当 n=5，补偿寄存器为 9 位时，高 4 位为整数部分，低 5 位为小数部分。

8	7	6	5	4	3	2	1	0
整数部分				小数部分				

整数部分指 32 秒内每秒都要补偿的值，小数部分指 32 秒内有多少秒需要补偿 1。

例如，当前时钟补偿目标值为 185.11ppm 时：

185.11ppm / （1 / 32768 * 10^6 ppm）= 6.0657

- 整数部分 = 6，即 32 秒内每秒都要补偿 6
- 小数部分 = 0.0657 *（1 / 32768 * 10^6）= 2，即 32 秒内有 2 秒需要补偿 1

按照以上说明，在 32 秒周期内，具有 5 位小数部分的 9 位补偿寄存器最大的补偿值计算公式如下：

最大补偿值 =（2^9 – 1）*（1 / 32768 / 2^5) * 10^6 = 487.33ppm

公式 2 最大补偿值

但实际时钟的 ppm 偏差值可能是正值也可能是负值，因为：

ppm = [（实际频率 - 设定频率）/ 设定频率] x 10^6

公式 3 ppm 定义

- 当实际频率 > 设定频率，即 ppm 值为正值时，则需要增加时钟周期，即补偿值为正值
- 当实际频率 < 设定频率，即 ppm 值为负值时，则需要减少时钟周期，即补偿值为负值

所以实际寄存器实现的补偿值也需设定为满足正值和负值 ppm 补偿的要求。

以下表格是计数时钟为 32768Hz 时，HC32L196 的补偿寄存器的描述：

时钟误差补偿寄存器（RTC_COMPEN）

复位值：0x0000 0020

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Res															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
EN	Res						CR								

位	标记	位名	功能	读写
b31~b16	Res	保留位	保留，必须保持复位值	
b15	EN	补偿使能	0：禁止时钟误差补偿 1：使能时钟误差补偿	RW
b14~b9	Res	保留位	保留，必须保持复位值	
b8~b0	CR	补偿值	通过补偿值设定，可针对每秒进行 +/-0.954ppm 的精度补偿。 可补偿范围 -274.7ppm ~ 212.7ppm。 最小微分误差 +/-0.477ppm。最小分辨率 0.954ppm。 具体补偿精度请参考下表：	RW

补偿值设定										补偿数
EN	CR[8:0]									
1	1	0	0	0	0	0	0	0	0	-274.7ppm
	1	0	0	0	0	0	0	0	1	-273.7ppm
	:	:	:	:	:	:	:	:	:	:
	0	0	0	0	1	1	1	1	1	-0.954ppm
	0	0	0	1	0	0	0	0	0	0ppm
	:	:	:	:	:	:	:	:	:	:
	0	1	1	1	1	1	1	1	0	+211.7ppm
	0	1	1	1	1	1	1	1	1	+212.7ppm
0	X	X	X	X	X	X	X	X	X	无补偿

- 当需补偿目标值 = （32767.71 – 32768）/ 32768 * 10^6 = -8.85ppm 时，补偿的步长 = -8.85 / (1 / 32768 / 2^5 * 10^6) = -9.28 ≈ -9
  因为 0ppm 的寄存器值为 0x20，所以写入寄存器的值 = 0x20 – 9 = 23 = 0b000010111
- 当需补偿目标值 = （32766.93 – 32768）/ 32768 * 10^6 = -32.65ppm 时，补偿的步长 = -32.65 /（1 / 32768 / 2^5 * 10^6）= -34.24 ≈ -34
  因为 0ppm 的寄存器值为 0x20，所以写入寄存器的值 = 0x100 +（0x20 – 34）= 510 = 0b111111110
- 当需补偿目标值 = （32768.91 – 32768）/32768 * 106 = 27.775ppm 时，补偿的步长 = 27.775 /（1 / 32768 / 2^5 * 10^6）= 29.12 ≈ 29
  因为 0ppm 的寄存器值为 0x20，所以写入寄存器的值 = 0x20 + 29 = 61 = 0b000111101

注：在此计算过程中添加四舍五入的运算，可进一步增加精度。

3 晶体温度特性曲线

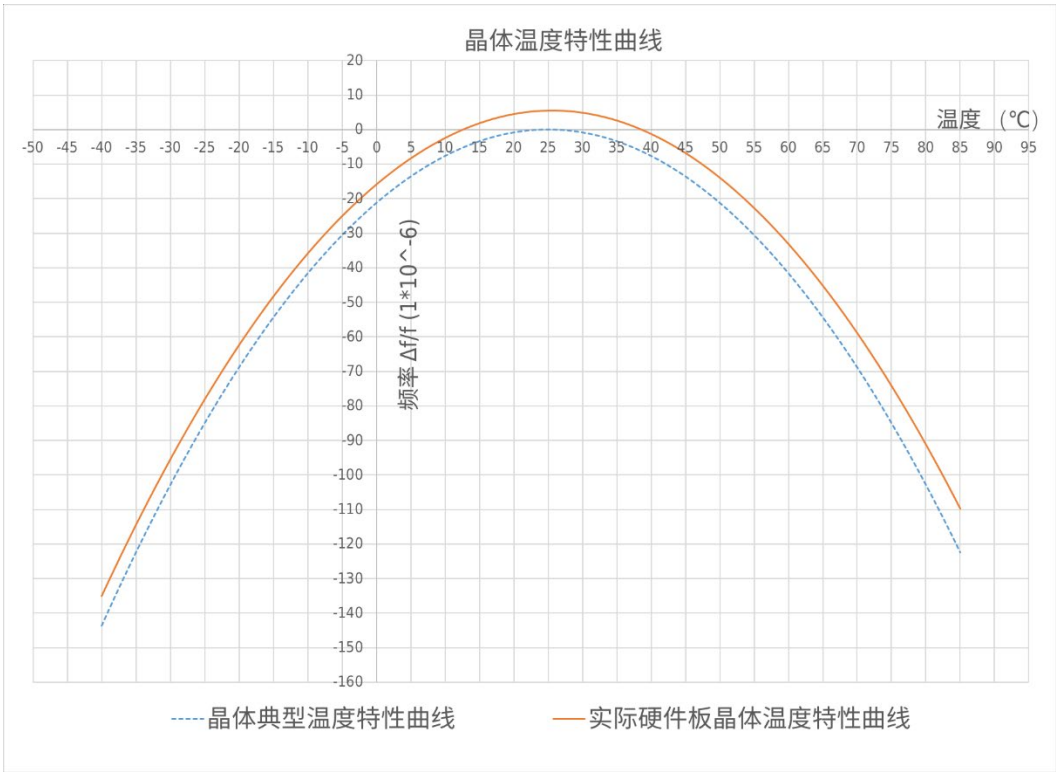

图 3-1 晶体温度特性曲线

3.1 晶体典型温度特性曲线

带有实时时钟的应用通常在硬件板上外置 32768Hz 的高精度 ppm 的石英晶体，其在一定温度范围内的温度特性曲线近似为一条开口向下的抛物线，温度越偏离顶点温度，晶体的精度偏差越大，如图中晶体典型温度曲线所示。

晶体典型温度曲线可由公式表示为：

$$\Delta f / f = \alpha * (T - Ti) \hat{} 2$$

公式 4 晶体典型温度特性曲线

- α：抛物线系数，常数
- T：实际温度（单位：℃）
- Ti：拐点温度（单位：℃），常数

根据晶体规格书，可得α和 Ti 的值，例如某款晶体：

- α = -（0.034 ± 0.0015），单位 10^-6 / ℃²
- Ti = +25 ± 5，单位 ℃

生产一致性较好的石英晶体在应用上可使用同一个参数来计算晶体精度偏差。

3.2 实际硬件板晶体温度特性曲线

但实际应用时，由于叠加板上寄生电容，MCU 管脚电容等因素，晶体的实际温度特性曲线会发生偏移，如图中的实际硬件板晶体温度特性曲线所示。

实际硬件板晶体的温度特性曲线可由公式表示为：

$$\Delta f / f = \alpha * (T - Ti)\ ^\wedge\ 2 + ppm\_t$$

公式 5 实际硬件板晶体温度特性曲线

- α：抛物线系数，常数
- T：实际温度（单位：℃）
- Ti：拐点温度（单位：℃），常数
- ppm_t：顶点 PPM 偏差，常数

为了实现全工作温度区域的实时时钟的高精度要求，α、Ti 和 ppm_t 三个参数需要根据实际系统实测得出。

4 实时时钟补偿方法

本章节介绍在应用上如何实测晶体温度特性曲线，根据当前温度计算出晶体的 ppm 偏差，并通过微处理器内部的 RTC 模块补偿，实现实际输出 ppm 的校正。

以下为实时时钟补偿的流程，首次使用时需要测量实际硬件板晶体在不同温度下的精度，对该精度数据结合温度做数据拟合，生成公式。实时补偿的功能由程序执行，软件根据当前温度和晶体温度特性公式计算出 ppm 偏差，再通过微处理器内部的 RTC 模块去补偿输出的精度。

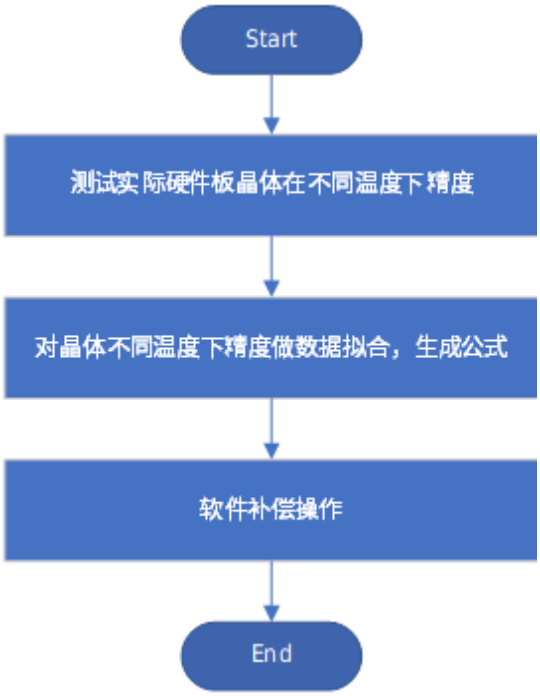

图 4-1 实时时钟补偿流程

4.1 获取晶体温度特性曲线

为了实现全工作温度区域的实时时钟的高精度要求，应用上需要实测晶体的温度特性曲线，拟合出公式并确认α、Ti 和 ppm_t 三个参数。

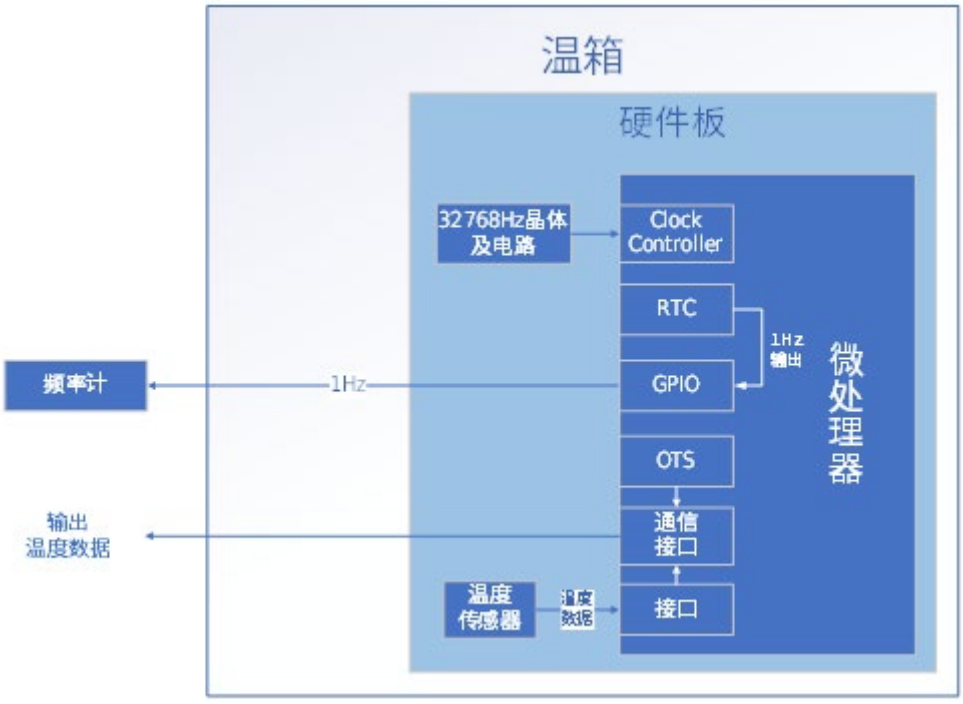

图 4-2　晶体温度特性曲线测试系统框图

注：温度数据可从片内 *OTS*（*On-chip Temperature Sensor*）或外置温度传感器获取，其可以通过通信接口打印出来或直接在调试器中读取温度变量值。

1)　测试实际硬件板晶体在不同温度下精度

将硬件板放入温箱，频率计接 1Hz 输出，调节温箱在不同温度下并记录当前的温度值和频率计的 ppm 值。

注:

- 因为晶体温度特性决定，越在抛物线的两端，晶体频率的偏差随温度变化越大。所以在晶体高精度的要求下，此处特别需要注意记录的温度的精准度
- 温度传感器尽量靠近晶体安放

2)　对晶体不同温度下精度做数据拟合，生成公式

根据上一步骤记录的数据拟合出晶体温度特性公式。

注：在不能用同一个公式时，可采用分温度区域，使用不同的公式。

作为示例，以下将使用统一的公式，如

$$\Delta f / f = -0.0327 * (T - 25.6)\ \hat{}\ 2 + 5.6$$

公式 6　实测经数据拟合后的晶体温度特性公式

- α = -0.0327
- Ti = 25.6
- ppm_t = 5.6

4.2 软件补偿操作

如图 4-2 晶体温度特性曲线测试系统框图所示，微处理器需获取当前的温度，其可以通过片内的 OTS 或者片外的温度传感器来获取。

软件根据获取的当前温度值，按照实际的晶体温度特性公式计算出当前温度下的晶体精度偏差，然后将计算的晶体精度偏差值补偿到 RTC 补偿寄存器中，流程如下：

图 4-3 软件补偿流程

1) 将当前的温度值带入实际的晶体温度特性公式Δf / f = -0.0327 *（T – 25.6）^ 2 + 5.6 中，计算出当前温度下该晶体的 ppm 偏差
2) 将 ppm 偏差值转化为以 RTC 寄存器补偿精度为单位的数据值
   RCT 寄存器 1 所指代的补偿是 1 / 32768 / 2^5 = 0.954ppm，通过计算公式将 1）的结果转换为基于 RTC 寄存器的 ppm 偏差
3) 将基于 RTC 寄存器的 ppm 偏差数据写入 RTC 补偿寄存器的补偿值中
   例程：

```
/* RTC register macro definition */
// 32768/32 --> 2^20
#define RTC_PRECISION_BITS      20
// 0 ppm value for RTC compensate register
#define RTC_ZERO_VALUE            0x20
// RTC compensate register bits = 9
#define RTC_COMPEN_REG_BITS     9
// Minimum RTC compensate value
```

```
#define RTC_COMPEN_MIN_VALUE   (uint16_t)((1 << (RTC_COMPEN_REG_BITS)) - 1)
// RTC shift bits = 7
#define RTC_COMPEN_SHIFT_BITS (16 - RTC_COMPEN_REG_BITS)
// Round up value
#define ROUND_UP_RTC_COMPEN_SHIFT (1 << (RTC_COMPEN_SHIFT_BITS - 1))

int32_t   i32RtcCompenRValue;
uint16_t u16RtcRegRValue;

1)
// 计算晶体 ppm 偏差值Δf / f =   -0.0327 *（T  −  25.6）^ 2 + 5.6
2)
// 将 ppm 偏差值转化为以 RTC 寄存器补偿精度为单位的数据值
// i32RtcCompenRValue =  晶体 ppm 偏差值 * (2 ^ RTC_PRECISION_BITS) / 1000000
3)
// RTC 补偿寄存器补偿值计算
u16RtcRegRValue        = (uint16_t)((int16_t)(RTC_ZERO_VALUE << RTC_COMPEN_SHIFT_BITS)
                          + (int16_t)(i32RtcCompenRValue << RTC_COMPEN_SHIFT_BITS));
u16RtcRegRValue        = (u16RtcRegRValue + ROUND_UP_RTC_COMPEN_SHIFT) >> RTC_COMPEN_SHIFT_BITS;
RTC->COMPEN_f.CR       = u16RtcRegRValue & RTC_COMPEN_MIN_VALUE;
```

注：

- 以上例程基于 HC32L196 的寄存器配置，包含示意代码
- 以上样例基于 RTC->COMPEN_f.EN 已设为 1，即时钟误差补偿已使能

至此，根据预先拟合的晶体温度特性公式，微处理器将获取的当前温度带入公式计算出需补偿的 ppm 偏差，再将偏差值写入到 RTC 寄存器的补偿值中，从而实现了全工作温度区域内的 ppm 的补偿。

5 总结

本应用笔记介绍了小华微处理器的 RTC 补偿功能和实时时钟补偿方法，实现应用上对于高精度实时时钟的要求。

版本修订记录

版本号	修订日期	修订内容
Rev1.00	2024/12/11	初版发布。
Rev1.01	2025/04/02	适用对象章节，新增 HC32A448、HC32A472、HC32A460、HC32A4A0 系列型号，产品型号描述修改。
Rev1.02	2025/04/11	适用对象添加 HC32M441 系列。