

---

---

## AVR®单片机的实时时钟校准和补偿

---

---

### 简介

作者：Eivind Berntsen 和 Per Andreas Gulbrandsen, Microchip Technology Inc.

嵌入式计时应用通常使用 32.768 kHz 外部晶振作为时钟源，因为此类晶振的精度高于内部振荡器的精度。然而，精度会受到多种因素的不利影响。本文档主要介绍 PCB 设计、老化和温度漂移。此外，还将对晶振制造容差进行简要说明。

为了确保高精度的计时，可以提高振荡器的精度。通过测量振荡器的误差并对其进行补偿，即可实现此目的。对于频率测量，可以使用 tinyAVR® 1 系列和 megaAVR® 0 系列的外部参考及内部外设来完成，也可以使用外部元件完成。这两种方法在本文档中均有介绍。

当误差已知时，可以采用基于软件的补偿方式来减小误差。本文档给出了几个示例，并提供了一些实用技巧以便有效地实现基于软件的补偿。

### 特性

简介和测量：

- 晶振频率误差的影响因素简介
- 有关如何在 tinyAVR 1 系列和 megaAVR 0 系列上测量 32.768 kHz 晶振时钟频率的说明

示例：

- 显示如何使用 RTC 补偿静态晶振误差的示例
- 显示如何使用 RTC、内部温度传感器和 ADC 补偿静态晶振误差和由温度漂移引起的误差的示例
- 可从 Atmel | START 下载四个示例，其中两个示例显示了频率测量相关信息，另外两个示例显示了补偿算法

## 目录

简介.....	1
特性.....	1
1. 相关器件.....	3
1.1. tinyAVR 1 系列.....	3
1.2. megaAVR 0 系列.....	3
2. 误差和漂移的来源.....	5
2.1. 容性负载不匹配.....	5
2.2. 晶振制造容差.....	6
2.3. 32.768 kHz 晶振温度漂移.....	6
2.4. 32.768 kHz 晶振老化漂移.....	6
3. 频率测量.....	8
3.1. 起振时间.....	8
3.2. 外部频率测量.....	8
3.3. 内部时钟测量.....	10
4. RTC 补偿.....	11
4.1. 晶振和负载电容偏差补偿.....	11
4.2. 温度漂移补偿.....	11
4.3. 晶振补偿查找表示例.....	11
5. 从 Atmel   START 获取源代码.....	13
6. 版本历史.....	14
Microchip 网站.....	15
产品变更通知服务.....	15
客户支持.....	15
Microchip 器件代码保护功能.....	15
法律声明.....	15
商标.....	16
质量管理体系.....	16
全球销售及服务网点.....	17

## 1. 相关器件

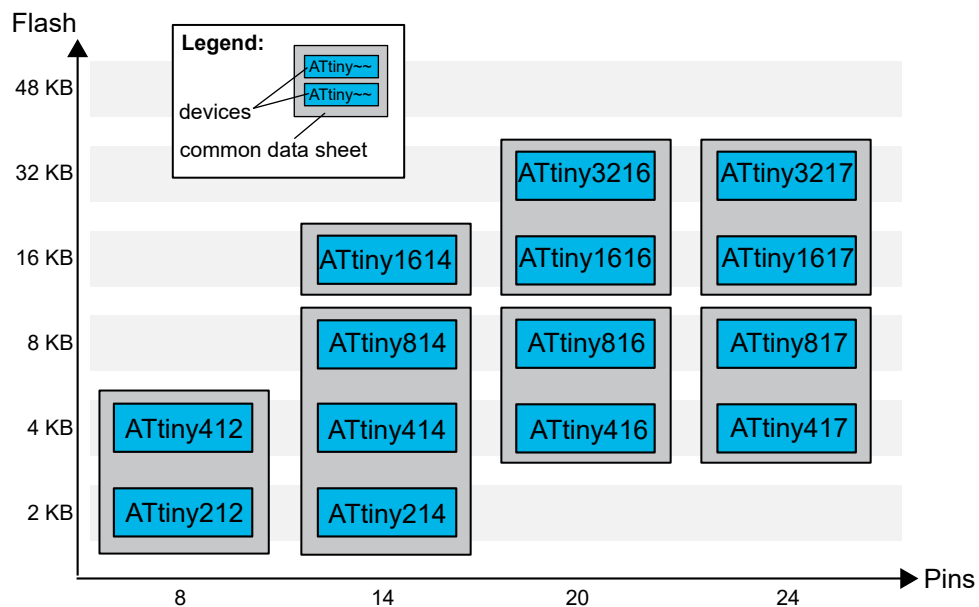
本章列出了本文档的相关器件。

### 1.1 tinyAVR 1 系列

下图所示为 tinyAVR 1 系列器件，注明了不同的引脚数与存储器大小：

- 垂直迁移无需修改代码，因为这些器件的引脚互相兼容，后者可提供相同甚至更多的功能。而从下到上迁移可能需要修改代码，因为部分外设的可用引脚数较少。
- 从右到左迁移会减少引脚数，进而减少可用的功能。

图 1-1. tinyAVR 1 系列概览



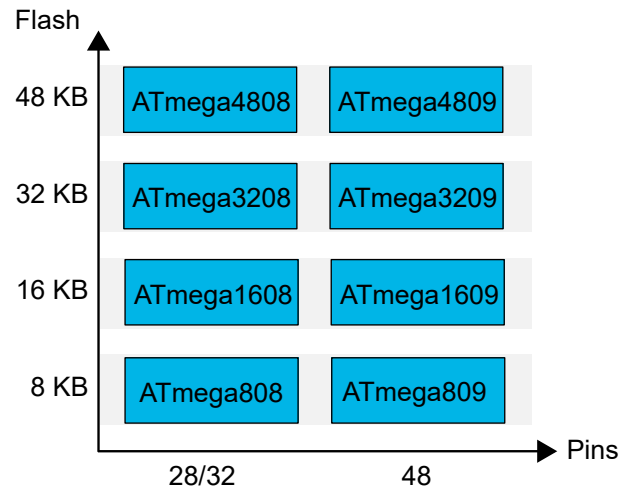
具有不同闪存大小的器件通常也具有不同的 SRAM 和 EEPROM。

### 1.2 megaAVR 0 系列

下图所示为 megaAVR 0 系列器件，注明了不同的引脚数与存储器大小：

- 垂直迁移无需修改代码，因为这些器件的引脚和功能完全兼容。
- 从右到左迁移会减少引脚数，进而减少可用的功能。

图 1-2. megaAVR 0 系列概览



具有不同闪存大小的器件通常也具有不同的 SRAM 和 EEPROM。

## 2. 误差和漂移的来源

32.768 kHz 晶振的精度高于 RC 振荡器的精度。不过，晶振仍会受到频率误差的影响。其中包括静态误差，这种误差主要是由 PCB 设计和制造容差引起的。此外，频率误差还会受到温度漂移的影响，这意味着输出频率将随着工作温度的变化而变化。

### 2.1 容性负载不匹配

用于晶振的外部容性负载由晶振制造商指定。容性负载不匹配将导致晶振精度降低，甚至无法振荡。晶振数据手册中列出了频率与负载电容的典型特性曲线，如图 2-1 所示。

图 2-1. 频率与负载电容特性曲线

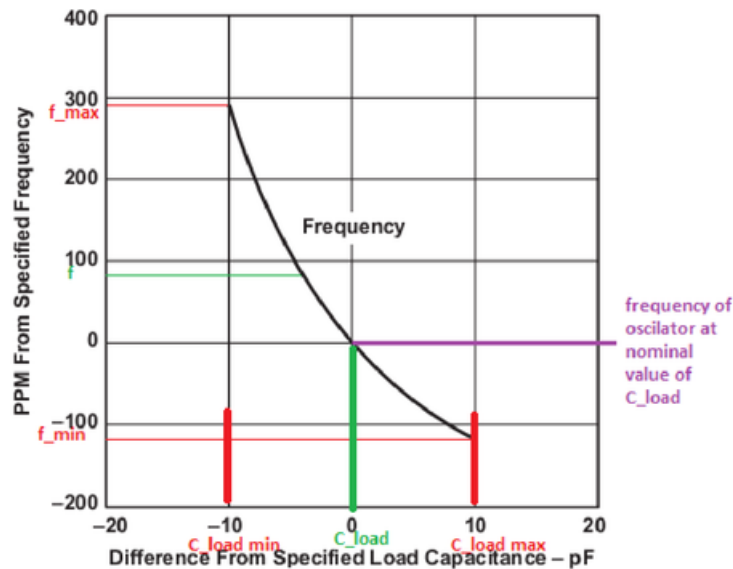
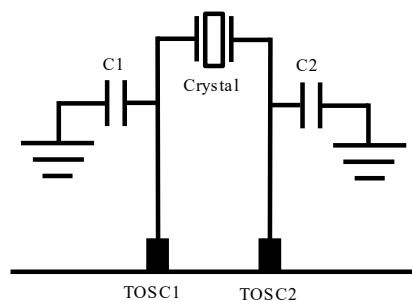


图 2-2 显示了典型的 32.768 kHz 晶振电路示意图。晶振连接到 MCU 的 TOSC1 和 TOSC2 引脚。C1 和 C2 电容用于平衡负载电容。

图 2-2. 32.768 kHz 晶振电路



有多个源会导致晶振电路中出现负载电容：

- PCB 寄生电容，即 PCB 走线之间的电容。
- MCU 寄生电容。有关更多详细信息，请参见器件数据手册的电气特性部分。
- 贴装电容 C1 和 C2，分别放置在 MCU 的 TOSC1 引脚与 GND 以及 TOSC2 引脚与 GND 之间。

在计算 C1 和 C2 的值时，必须考虑 MCU 寄生电容和 PCB 寄生电容。PCB 寄生电容取决于 PCB 设计。例如，走线长度和宽度的选择会影响此电容。有关这方面最佳实践的更多信息，请参见 [AVR4100:Selecting and testing 32.768 kHz crystal oscillators for Atmel AVR microcontrollers](#)。

所有元件都存在制造容差。这意味着正确选择 C1 和 C2 不能确保生产的所有器件都具有相同的精度。因此，必须执行基于软件的补偿，以在计时中实现最佳精度。

## 2.2 晶振制造容差

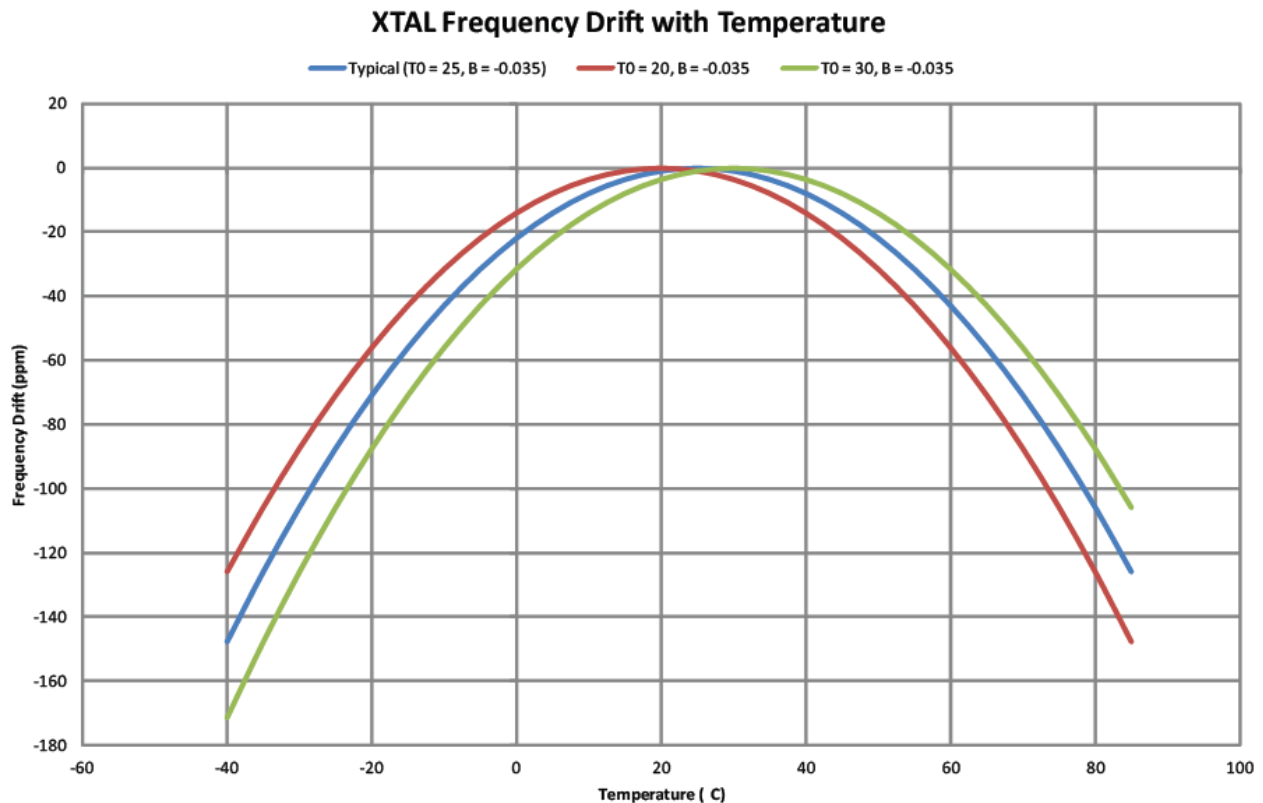
正确应用负载时，32.768 kHz 晶振的精度通常在 $\pm 20$  ppm 范围内。1 ppm 的误差相当于每年 30.5s 的误差。20 ppm 的误差相当于每年约 10 分钟的误差。通常，制造容差是由晶体切割精度和晶体纯度导致的。

## 2.3 32.768 kHz 晶振温度漂移

如图 2-3 所示，低成本 32.768 kHz 晶振的频率与温度之间通常存在抛物线关系，晶振制造商为此指定：

- 温度系数 (B)，单位为 ppm/°C<sup>2</sup>
- 拐点温度 (T0)，单位为 °C

图 2-3. 晶振频率漂移与温度的关系



温度系数 B 为负，这意味着晶振的速度会在高温和低温时降低。温度系数具有容差，通常为 $\pm 15\%$ 。通常，拐点温度 T0 为 25°C，容差的范围为 $\pm 5^\circ\text{C}$ 。温度为 T 时的频率漂移由以下公式确定：

$$\Delta f / F0(\text{ppm}) = B \cdot (T - T0)^2$$

举例来说，假设 B = -0.035 ppm/°C<sup>2</sup>  $\pm 15\%$ ，T0 = 25°C  $\pm 5^\circ\text{C}$  且 T = 85°C。此时，典型的漂移为：

$$\Delta f / F0 = 126\text{ppm} \pm 40\text{ppm}$$

## 2.4 32.768 kHz 晶振老化漂移

大多数晶振制造商会指定晶体在运行一年后的老化程度。通常，第一年的老化范围为 $\pm 3\text{ppm}$ 。此规范表明晶振漂移的方向和大小是不可预测的。尽管大部分频率漂移会在第一年出现，但也有部分漂移在一年后出现。导致频率漂移的

一些因素包括氧化，以及附着在晶体上的包装内残留物。这会改变晶体的总质量，从而影响谐振频率。有关此内容的更详细说明，请参见 [en.wikipedia.org/wiki/Crystal\\_oscillator#Stability\\_and\\_aging](https://en.wikipedia.org/wiki/Crystal_oscillator#Stability_and_aging)。

负载电容会随着时间的推移而变化。如果负载电容发生变化，则将导致频率漂移，但由于这种变化的幅度不大且晶振引脚之间的变化是对称的，此影响可以忽略不计。PCB 寄生电容和 MCU 寄生电容几乎不会老化。NP0/C0G 型电容在整个生命周期内的老化程度小于  $\pm 0.1\%$ ，即对于 10 pF 电容，老化程度小于 10 fF。考虑到 ppm 误差，由于负载电容变化产生的漂移小于 0.1 ppm。

尽管 PCB 寄生电容一般不会老化太多，但在 PCB 上添加硅或氨基甲酸酯来提高抗振性或防潮性，仍可以稍微改变 PCB 走线的电容，因为这些材料与空气相比具有不同的介电常数。

### 3. 频率测量

只有在频率误差大小已知的情况下才能进行补偿，因此需要测量晶振的频率。必须在 PCB 上已贴装晶振和器件时执行测量。

与依赖制造商的数据相比，在一定温度范围内进行多次测量通常会得到更好的结果。

对于 32.768 kHz 晶振的频率，可以使用外部测量工具或使用内部资源进行测量。使用外部工具测量时，会将晶振频率作为输入。而使用内部资源（如定时器/计数器）进行测量时，必须提供高精度参考。测量精度绝不会高于参考精度，因为参考精度是 32.768 kHz 晶振频率与之相比的精度。

在这两种情况下，重要的是要确保晶体有足够的时间来启动和稳定，然后再测量频率。

#### 3.1 起振时间

所有晶振都有起振时间，这是晶振输出达到稳定状态所需的时间。有关给定晶振的起振时间，请参见晶振数据手册。tinyAVR 1 系列和 megaAVR 0 系列提供多种选项来设置起振时间，最大值为 65535。这意味着器件以 32.768 kHz 的频率计数 65536 个周期，相当于 2 秒的起振时间。当超过起振时间后，主时钟状态寄存器中的 XOSC32K 状态位（MCLKSTATUS.XOSC32KS）将置 1。

**注：**当 32.768 kHz 晶振控制 A 寄存器中的使能位（XOSC32KCTRLA.ENABLE）写入 1 时，32.768 kHz 晶振将不会自动启动。仅当有外设（如主时钟或 RTC）请求 32.768 kHz 时钟或 XOSC32KCTRLA 中的运行待机位（XOSC32KCTRLA.RUNSTDBY）置 1 时，32.768 kHz 晶振才会启动。

**注：**除非有外设请求 32.768 kHz 时钟，否则 MCLKSTATUS.XOSC32KS 将不会置 1。XOSC32KCTRLA.RUNSTDBY 置 1 的情况也是如此。

#### 3.2 外部频率测量

使用外部测量工具测量频率时，必须在引脚上提供 32.768 kHz 晶振输出。为此，可以使用来自 RTC 或 PIT 的事件，并将引脚配置为用户，或者使用 32.768 kHz 晶振作为主时钟。

RTC 使能时，将生成以下输出事件：

- 溢出（OVF）：计数器达到其最高值并计满返回到 0 时生成。生成的选通信号与 CLK\_RTC 同步并持续一个 CLK\_RTC 周期。
- 比较（CMP）：指示计数器值和比较寄存器之间的匹配情况。生成的选通信号与 CLK\_RTC 同步并持续一个 CLK\_RTC 周期。

PIT 使能时，将在其事件输出上生成以下 50% 占空比的时钟信号：

- 事件 0：时钟周期 = 8192 个 RTC 时钟周期
- 事件 1：时钟周期 = 4096 个 RTC 时钟周期
- 事件 2：时钟周期 = 2048 个 RTC 时钟周期
- 事件 3：时钟周期 = 1024 个 RTC 时钟周期
- 事件 4：时钟周期 = 512 个 RTC 时钟周期
- 事件 5：时钟周期 = 256 个 RTC 时钟周期
- 事件 6：时钟周期 = 128 个 RTC 时钟周期
- 事件 7：时钟周期 = 64 个 RTC 时钟周期

事件用户由事件系统（EVSYS）配置。

当使用 32.768 kHz 晶振作为主时钟源时，有几种布线方法可以使时钟从引脚移出。

- 直接将时钟连接到 CLKOUT 引脚。
- 使用 TCA、TCB、TCD 或 SPI 输出主时钟频率。

测量频率时，不建议将探针直接连接到晶振电路。否则将改变电路的特性，并改变晶振频率。



以下两个示例显示了如何将 32.768 kHz 晶振输出到引脚。这两个示例分别说明如何使用系统时钟以及如何使用 RTC 和事件系统。

### 3.2.1 使用晶振时钟作为系统时钟

本示例说明了如何将 32.768 kHz 晶振配置为系统时钟，以及如何将系统时钟设置为 CLKOUT 引脚的输出。有关项目链接，请参见 5. 从 Atmel | START 获取源代码。

由于 32.768 kHz 时钟将为 CPU 提供时钟，因此建议：

1. 使能晶振，并将运行待机作为 main() 中执行的第一个操作。
2. 运行其他生产测试和电路校准（如果有）。
3. 将主时钟切换为晶振。
4. 将系统时钟配置为 CLKOUT 引脚上的输出。

该过程可以确保在执行其他生产测试时，晶振有足够的时间稳定下来。

在下面的示例代码中，晶振由第一个受保护的写操作启动，同时还会对运行待机位进行写操作。这样可以确保即使没有外设或 CPU 请求时钟，晶振也可以启动。在检查晶振是否稳定之前，应切换主时钟，因为除非有某个元件请求时钟，否则 XOSC32KS 位将不会置 1。

```
// 禁止寄存器的更改保护，在此处启动 XOSC32K
// 起振时间设置为最大 64K 个周期（约 2 秒）
_PROTECTED_WRITE(CLKCTRL_XOSC32KCTRLA, CLKCTRL_ENABLE_bm | \
    CLKCTRL_CSUT_64K_gc | CLKCTRL_RUNSTDBY_bm);

// 禁止寄存器的更改保护，将主时钟
// 切换为 32K 振荡器并将时钟输出到引脚
_PROTECTED_WRITE(CLKCTRL_MCLKCTRLA, CLKCTRL_MCLKCTRLA = \
    (CLKCTRL_CKSEL1_bm) | CLKCTRL_CLKOUT_bm);

// 等待 XOSC32K 报告为稳定状态
while(!(CLKCTRL_MCLKSTATUS & CLKCTRL_XOSC32KS_bm));

// 禁止寄存器的更改保护，确保预分频器已禁止。
_PROTECTED_WRITE(CLKCTRL_MCLKCTRLB, 0);
```

在输出 32.768 kHz 时钟的情况下，重要的是考虑输出引脚上的容性负载。负载越高，越难以在测试引脚上获得良好的信号输出。此外在测试时，必须确保时钟信号不会对连接到引脚的其他元件造成不利影响。因此，应将生产测试安排在设计过程的早期阶段，以确保引脚可用于处理此信号。

### 3.2.2 使用 RTC 和事件系统

在本示例中，RTC 的周期性中断定时器（Periodic Interrupt Timer, PIT）功能与事件系统一起使用。此时，无需将系统时钟切换到慢速振荡器。有关项目链接，请参见 5. 从 Atmel | START 获取源代码。

可以使用事件系统输出在 PIT 中产生的周期性事件。如上面所述，可选择多种不同的周期。在本示例中，使用了最小周期 PIT 事件输出，如 EVSYS\_ASYNCCH3 = EVSYS\_ASYNCCH3\_PIT\_DIV64\_gc; 一行所示。当 RTC 时钟的分频系数设置为 64 时，将在所选输出引脚上生成一个周期为 1.95 ms 的脉冲。

执行写保护操作，以尽早使能 32.768 kHz 晶振。为了更有效地利用时间，建议在晶振稳定的过程中执行其他生产测试。以下代码显示了如何将 PIT 设置为事件生成器，以及如何在引脚上输出信号：

```
// 禁止寄存器的更改保护，在此处启动 XOSC32K
// 起振时间设置为最大 64K 个周期（约 2 秒）
_PROTECTED_WRITE(CLKCTRL_XOSC32KCTRLA, CLKCTRL_ENABLE_bm \
    | CLKCTRL_CSUT_64K_gc | CLKCTRL_RUNSTDBY_bm);

// 使用事件系统通道 3 作为 PIT 事件的路径
EVSYS_ASYNCCH3 = EVSYS_ASYNCCH3_PIT_DIV64_gc;
// 将通道 3 连接到异步用户 8 (evout0)
EVSYS_ASYNCUSER8 = EVSYS_ASYNCUSER8_ASYNCCH3_gc;
// 将 evout0 通过多路开关输出到 PA2
PORTMUX_CTRLA = PORTMUX_EVOUT0_bm;

// 在 RTC 中使能 PIT
```

```
RTC_PITCTRLA = RTC_PITEN_bm;  
// 将晶振设置为 RTC 的时钟源  
RTC_CLKSEL = RTC_CLKSEL_TOSC32K_gc;  
// 使能 RTC  
RTC_CTRLA = RTC_RTCEN_bm;
```

### 3.3 内部时钟测量

内部频率测量可以随时进行，而不仅仅是在生产过程中。进行内部测量时，需要用到参考时钟。可以使用高频时钟或低频时钟，也可以使用时标，例如 GPS 模块中的一秒时标。

在这两种方法中，使用高频参考时钟的方法将在最短时间内获得最佳结果。

#### 3.3.1 使用高频输入进行校准

具体理念是，在由晶振驱动运行的 RTC 给出的设定周期内，使用 TCD 来计数快速参考频率。可将 TCD 溢出次数和最终捕捉值与预期值进行比较。有关项目的链接，请参见 [5. 从 Atmel | START 获取源代码](#)。

该方法包含以下步骤：

1. 将外部参考频率应用到器件的 EXTCLK 引脚。
2. 配置 TCD 以将该引脚用作其时钟源。
3. 配置 RTC 以使用 XOSC32K 时钟。
4. 配置事件系统以从 RTC 发送溢出信号，以便从 TCD 获取捕捉值。

在第一次捕捉后，TCD 将从零开始计数，因此应丢弃捕捉到的第一个值。计算 RTC 周期内的 TCD 溢出次数。当 RTC 周期结束后，存储捕捉到的第二个值，并使用该值和 TCD 溢出次数作为测量值。理想预期值与测量值之差即表示晶振的频率误差。

在本示例中，假设晶振精确，所测量的 RTC 周期设置为一秒。本示例中使用的外部参考频率为 20 MHz。如果晶振恰好以 32.768 kHz 的频率运行，且周期设为一秒，则 TCD 计数值应达到 20,000,000。由于 TCD 宽度仅为 12 位，因此无法达到此计数值。这种情况下，它将溢出 4882 次，最终捕捉值应为 3328。

现在，我们假设最终捕捉值未达到此理想值，而是达到了 2296，同时溢出次数仍为 4882。此时，TCD 计数的周期数比其预期计数少  $3328 - 2296 = 1032$  个周期。晶振的运行速度比预期速度快，这会导致 RTC 周期短于预期周期。如果以秒为单位，则时间差为  $1032 / 20,000,000 = 51.6 \mu\text{s}$ 。

## 4. RTC 补偿

如 2.3 32.768 kHz 晶振温度漂移所示，典型的 32.768 kHz 晶振将在 25°C 左右对称漂移。通过对 RTC 应用补偿算法，可以针对漂移进行调整。

### 4.1 晶振和负载电容偏差补偿

在上面的示例中，晶振的速度过快，达到了 51.6  $\mu\text{s/s}$ 。如果在 RTC 中每秒计数两个额外的周期，则 RTC 的速度将变慢  $2 \cdot (1/32768) = 60 \mu\text{s/s}$ 。此时，误差减小到  $60 - 51.6 = 8.6 \mu\text{s/s}$ ，此速度过慢。可以使用软件变量来跟踪每秒的累积误差。当累积误差溢出 32.768 kHz 晶振时钟的一个周期时，RTC 周期每秒减少一个周期。

在 Atmel | START 中提供的晶振补偿示例中，补偿变量的精度为 0.1  $\mu\text{s/s}$ 。这会增加计为误差的小数位数。在温度补偿示例中，补偿精度提高到 0.01  $\mu\text{s/s}$ 。有关项目的链接，请参见 5. 从 Atmel | START 获取源代码。

### 4.2 温度漂移补偿

为了能够控制晶振的容性负载，并降低引入晶振电路的噪声，应将 32.768 kHz 晶振放置在靠近器件的位置。这也意味着器件和晶振将处于相同的温度条件下。

tinyAVR 1 系列和 megaAVR 0 系列器件具有内部温度传感器。从相应器件数据手册中的特性部分中可以看出，温度传感器的精度相对较低。即使应用器件签名行中提供的校准值时也是如此，因为这些校准值是在硅片仍位于晶圆上时测得的。切割和封装都可能对硅片产生影响，进而改变温度读数。将器件焊接到 PCB 上时，校准值可能会进一步变化。

如果在生产过程中测量新的增益和偏移值，则可以使用内部温度传感器来获得更好的结果。为此，需要执行校准步骤，最好是在将器件焊接到 PCB 上之后再执行此步骤。如果需要在宽温度范围内保持良好的性能，则应执行两点校准或三点校准。其中两个测试点应略高于或略低于需要确保精度的温度范围，而另一个测试点应接近于中间范围。如果在这三个点上测量晶振频率，则可以获得比晶振数据手册中特性部分给出的抛物线温度曲线更精确的曲线。

**注：** 尽管理想的情况是使用高精度温度传感器，但使用低精度温度传感器进行温度补偿仍然有用。尤其是在工作温度远低于 25°C 的条件下，此时根据不精确温度读数选择的补偿值比 25°C 时给出的值更精确。

### 4.3 晶振补偿查找表示例

为了应对温度的变化，可以对温度进行持续监测，并在 RTC ISR 中对此进行补偿。这将有助于提高时钟的精度。为了避免复杂的浮点或定点数学运算，可以将晶振温度补偿值存储在查找表中。

晶振制造商通常会给出一张图或一个方程来描述漂移大小与温度之间的函数关系。如果所需精度高于制造商规范，则可以在不同的温度条件下针对每个 PCB 执行三点测量。这样便可找到每个晶振和电路对应的正确曲线。如果同时对内部温度传感器进行测量，则各数据点应在晶振漂移和测量温度之间对齐。

在 Atmel | START 提供的温度补偿示例中，抛物线根据以下数学函数得出：

$$\Delta f/F0 = -0.04 \cdot (T-25)^2$$

RTC ISR 中使用的补偿值直接取自基于上述函数生成的表。RTC 每秒都会溢出，表中的误差值会加到用于跟踪每秒累积的总误差的变量中。当此累积误差变量大于 32.768 kHz 晶振的一个周期时，将从 RTC 周期中减去一个或多个周期。当累积误差小于一个周期时，RTC 周期值将设置为常规的 1 秒周期值。

由于该方程是一个抛物线，并且查找表中的值围绕顶点对称，因此可以仅存储 25°C 到 105°C 温度范围内的值，并在温度介于 25°C 到 -40°C 之间时使用 25°C 到 90°C 温度范围内值的子集。这样将减少所需的闪存大小。

在本示例中，大多数计算都是在 RTC 和 ADC ISR 中完成的，并且所执行的代码耗时较少。

在 RTC ISR 中，将计算累积误差。其中包含尚未得到补偿的累积误差，以及每秒的静态晶振误差和每秒的温度补偿误差。可通过在 ADC ISR 中使用查找表来实现温度补偿。ADC 转换从 RTC ISR 开始。

**注：**在 RTC ISR 中写入 RTC 周期寄存器。写入的值必须与 32.768 kHz 域中的寄存器同步。只要之前写入到相同寄存器中的值已同步，CPU 就无需等待此操作完成。在 ISR 代码中执行此操作不会产生问题。如果不是通过 ISR 写入寄存器，或者如果 PER 值极低（1-3 个周期），那么在快速连续执行 ISR 时，可能会出现问題。请注意，RTC 对每个寄存器都有单独的同步机制。因此，在 RTC 中快速连续地写入两个不同的寄存器不会引起问題。

在 ADC ISR 中，将读出 ADC 结果，执行增益和偏移补偿，并将结果转换为开氏度。然后，根据该结果从温度补偿查找表中找到要使用的补偿值。

示例中的代码使用器件的内部温度传感器。此外，也可以使用更精确的外部温度传感器。在这种情况下，时钟精度将会提高。

## 5. 从 Atmel | START 获取源代码

示例代码可通过 Atmel | START 获得，Atmel | START 是一种基于 Web 的工具，可通过图形用户界面（Graphical User Interface, GUI）配置应用程序代码。可以通过下面提供的直接示例代码链接或 Atmel | START 起始页上的 *Browse examples*（浏览示例）按钮，下载 Atmel Studio 和 IAR Embedded Workbench® 对应的代码。

Atmel | START 网页：[ATMEL START](#)

### 示例代码

- 频率输出：
  - [http://start.atmel.com/#example/Atmel:rtc\\_calibration\\_and\\_compensation:1.0.0::Application:RTC\\_Crystal\\_Frequency\\_Output](http://start.atmel.com/#example/Atmel:rtc_calibration_and_compensation:1.0.0::Application:RTC_Crystal_Frequency_Output):
- 内部频率校准：
  - [http://start.atmel.com/#example/Atmel:rtc\\_calibration\\_and\\_compensation:1.0.0::Application:RTC\\_Internal\\_Frequency\\_Calibration](http://start.atmel.com/#example/Atmel:rtc_calibration_and_compensation:1.0.0::Application:RTC_Internal_Frequency_Calibration):
- 晶振误差补偿：
  - [http://start.atmel.com/#example/Atmel:rtc\\_calibration\\_and\\_compensation:1.0.0::Application:RTC\\_Crystal\\_Error\\_Compensation](http://start.atmel.com/#example/Atmel:rtc_calibration_and_compensation:1.0.0::Application:RTC_Crystal_Error_Compensation):
- 晶振补偿查找表：
  - [http://start.atmel.com/#example/Atmel:rtc\\_calibration\\_and\\_compensation:1.0.0::Application:RTC\\_Crystal\\_Compensation\\_Look-Up\\_Table](http://start.atmel.com/#example/Atmel:rtc_calibration_and_compensation:1.0.0::Application:RTC_Crystal_Compensation_Look-Up_Table):

有关详细信息和示例项目的相关信息，请单击 Atmel | START 中的 *User guide*（用户指南）。*User guide* 按钮可以在该网页中找到，方法是在 Atmel | START 项目配置器中的仪表板视图中单击项目名称。

### Atmel Studio

在 Atmel | START 的示例浏览器中单击 *Download selected example*（下载所选示例），下载 Atmel Studio 对应的代码并保存为 .atzip 文件。要从 Atmel | START 下载文件，单击 *Export project*（导出项目），然后单击 *Download pack*（下载数据包）。

双击下载的 .atzip 文件，将项目导入到 Atmel Studio 7.0。

### IAR Embedded Workbench

有关如何在 IAR Embedded Workbench 中导入项目的信息，请打开 [Atmel | START User Guide](#)（Atmel | START 用户指南），选择 *Using Atmel Start Output in External Tools*（使用外部工具中的 Atmel Start 输出），然后选择 *IAR Embedded Workbench*。单击 Atmel | START 起始页右上角的 *Help*（帮助）或项目配置器中右上角的 *Help And Support*（帮助和支持），均可找到 Atmel | START 用户指南的链接。

## 6. 版本历史

文档版本	日期	备注
B	2018 年 10 月	在“相关器件”章节中增加 8/16 KB megaAVR 0 系列器件。
A	2018 年 07 月	文档初始版本。

---

## Microchip 网站

---

Microchip 网站 ([www.microchip.com/](http://www.microchip.com/)) 为客户提供在线支持。客户可通过该网站方便地获取文件和信息。我们的网站提供以下内容：

- **产品支持**——数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及归档软件
- **一般技术支持**——常见问题解答 (FAQ)、技术支持请求、在线讨论组以及 Microchip 设计伙伴计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

---

## 产品变更通知服务

---

Microchip 的产品变更通知服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

欲注册，请访问 [www.microchip.com/pcn](http://www.microchip.com/pcn)，然后按照注册说明进行操作。

---

## 客户支持

---

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师 (ESE)
- 技术支持

客户应联系其代理商、代表或 ESE 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过 [www.microchip.com/support](http://www.microchip.com/support) 获得网上技术支持。

---

## Microchip 器件代码保护功能

---

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术规范。
- Microchip 确信：在正常使用的情况下，Microchip 系列产品非常安全。
- 目前，仍存在着用恶意、甚至是非法的方法来试图破坏代码保护功能的行为。我们确信，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这种试图破坏代码保护功能的行为极可能侵犯 Microchip 的知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。代码保护功能处于持续发展中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案 (Digital Millennium Copyright Act)》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

---

## 法律声明

---

提供本文档的中文版本仅为为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中提供的信息仅仅是为方便您使用 Microchip 产品或使用这些产品来进行设计。本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。

Microchip “按原样”提供这些信息。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对非侵权性、适销性和特定用途的适用性的暗示担保，或针对其使用情况、质量或性能的担保。

在任何情况下，对于因这些信息或使用这些信息而产生的任何间接的、特殊的、惩罚性的、偶然的或间接的损失、损害或任何类型的开销，Microchip 概不承担任何责任，即使 Microchip 已被告知可能发生损害或损害可以预见。在法律允许的最大范围内，对于因这些信息或使用这些信息而产生的所有索赔，Microchip 在任何情况下所承担的全部责任均不超出您为获得这些信息向 Microchip 直接支付的金额（如有）。如果将 Microchip 器件用于生命维持和/或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切损害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任。除非另外声明，在 Microchip 知识产权保护下，不得暗或以其他方式转让任何许可证。

## 商标

Microchip 的名称和徽标组合、Microchip 徽标、Adaptec、AnyRate、AVR、AVR 徽标、AVR Freaks、BesTime、BitCloud、chipKIT、chipKIT 徽标、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、HELDO、IGLOO、JukeBlox、KeeLoq、Kleer、LANCheck、LinkMD、maXStylus、maXTouch、MediaLB、megaAVR、Microsemi、Microsemi 徽标、MOST、MOST 徽标、MPLAB、OptoLyzer、PackeTime、PIC、picoPower、PICSTART、PIC32 徽标、PolarFire、Prochip Designer、QTouch、SAM-BA、SenGenuity、SpyNIC、SST、SST 徽标、SuperFlash、Symmetricom、SyncServer、Tachyon、TimeSource、tinyAVR、UNI/O、Vectron 及 XMEGA 均为 Microchip Technology Incorporated 在美国和其他国家或地区的注册商标。

AgileSwitch、APT、ClockWorks、The Embedded Control Solutions Company、EtherSynch、FlashTec、Hyper Speed Control、HyperLight Load、IntelliMOS、Libero、motorBench、mTouch、Powermite 3、Precision Edge、ProASIC、ProASIC Plus、ProASIC Plus 徽标、Quiet-Wire、SmartFusion、SyncWorld、Temux、TimeCesium、TimeHub、TimePictra、TimeProvider、WinPath 和 ZL 均为 Microchip Technology Incorporated 在美国的注册商标。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、Augmented Switching、BlueSky、BodyCom、CodeGuard、CryptoAuthentication、CryptoAutomotive、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、Espresso T1S、EtherGREEN、IdealBridge、In-Circuit Serial Programming、ICSP、INICnet、Intelligent Paralleling、Inter-Chip Connectivity、JitterBlocker、maxCrypto、maxView、memBrain、Mindi、MiWi、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICKit、PICtail、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、RTAX、RTG4、SAM-ICE、Serial Quad I/O、simpleMAP、SimpliPHY、SmartBuffer、SMART-I.S.、storClad、SQI、SuperSwitcher、SuperSwitcher II、Switchtec、SynchroPHY、Total Endurance、TSHARC、USBCheck、VariSense、VectorBlox、VeriPHY、ViewSpan、WiperLock、XpressConnect 和 ZENA 均为 Microchip Technology Incorporated 在美国和其他国家或地区的商标。

SQTP 为 Microchip Technology Incorporated 在美国的服务标记。

Adaptec 徽标、Frequency on Demand、Silicon Storage Technology 和 Symmcom 均为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

GestIC 为 Microchip Technology Inc. 的子公司 Microchip Technology Germany II GmbH & Co. KG 在除美国外的国家或地区的注册商标。

在此提及的所有其他商标均为各持有公司所有。

© 2021, Microchip Technology Incorporated 版权所有。

ISBN: 978-1-5224-6967-4

## 质量管理体系

有关 Microchip 的质量管理体系的信息，请访问 [www.microchip.com/quality](http://www.microchip.com/quality)。



## 全球销售及服务中心

美洲	亚太地区	亚太地区	欧洲
<b>公司总部</b> 2355 West Chandler Blvd. Chandler, AZ 85224-6199 电话: 480-792-7200 传真: 480-792-7277 技术支持: <a href="http://www.microchip.com/support">www.microchip.com/support</a> 网址: <a href="http://www.microchip.com">www.microchip.com</a>	<b>澳大利亚 - 悉尼</b> 电话: 61-2-9868-6733 <b>中国 - 北京</b> 电话: 86-10-8569-7000 <b>中国 - 成都</b> 电话: 86-28-8665-5511 <b>中国 - 重庆</b> 电话: 86-23-8980-9588 <b>中国 - 东莞</b> 电话: 86-769-8702-9880 <b>中国 - 广州</b> 电话: 86-20-8755-8029 <b>中国 - 杭州</b> 电话: 86-571-8792-8115 <b>中国 - 香港特别行政区</b> 电话: 852-2943-5100 <b>中国 - 南京</b> 电话: 86-25-8473-2460 <b>中国 - 青岛</b> 电话: 86-532-8502-7355 <b>中国 - 上海</b> 电话: 86-21-3326-8000 <b>中国 - 沈阳</b> 电话: 86-24-2334-2829 <b>中国 - 深圳</b> 电话: 86-755-8864-2200 <b>中国 - 苏州</b> 电话: 86-186-6233-1526 <b>中国 - 武汉</b> 电话: 86-27-5980-5300 <b>中国 - 西安</b> 电话: 86-29-8833-7252 <b>中国 - 厦门</b> 电话: 86-592-2388138 <b>中国 - 珠海</b> 电话: 86-756-3210040	<b>印度 - 班加罗尔</b> 电话: 91-80-3090-4444 <b>印度 - 新德里</b> 电话: 91-11-4160-8631 <b>印度 - 浦那</b> 电话: 91-20-4121-0141 <b>日本 - 大阪</b> 电话: 81-6-6152-7160 <b>日本 - 东京</b> 电话: 81-3-6880-3770 <b>韩国 - 大邱</b> 电话: 82-53-744-4301 <b>韩国 - 首尔</b> 电话: 82-2-554-7200 <b>马来西亚 - 吉隆坡</b> 电话: 60-3-7651-7906 <b>马来西亚 - 槟榔屿</b> 电话: 60-4-227-8870 <b>菲律宾 - 马尼拉</b> 电话: 63-2-634-9065 <b>新加坡</b> 电话: 65-6334-8870 <b>台湾地区 - 新竹</b> 电话: 886-3-577-8366 <b>台湾地区 - 高雄</b> 电话: 886-7-213-7830 <b>台湾地区 - 台北</b> 电话: 886-2-2508-8600 <b>泰国 - 曼谷</b> 电话: 66-2-694-1351 <b>越南 - 胡志明市</b> 电话: 84-28-5448-2100	<b>奥地利 - 韦尔斯</b> 电话: 43-7242-2244-39 传真: 43-7242-2244-393 <b>丹麦 - 哥本哈根</b> 电话: 45-4485-5910 传真: 45-4485-2829 <b>芬兰 - 埃斯波</b> 电话: 358-9-4520-820 <b>法国 - 巴黎</b> 电话: 33-1-69-53-63-20 传真: 33-1-69-30-90-79 <b>德国 - 加兴</b> 电话: 49-8931-9700 <b>德国 - 哈恩</b> 电话: 49-2129-3766400 <b>德国 - 海尔布隆</b> 电话: 49-7131-72400 <b>德国 - 卡尔斯鲁厄</b> 电话: 49-721-625370 <b>德国 - 慕尼黑</b> 电话: 49-89-627-144-0 传真: 49-89-627-144-44 <b>德国 - 罗森海姆</b> 电话: 49-8031-354-560 <b>以色列 - 若那那市</b> 电话: 972-9-744-7705 <b>意大利 - 米兰</b> 电话: 39-0331-742611 传真: 39-0331-466781 <b>意大利 - 帕多瓦</b> 电话: 39-049-7625286 <b>荷兰 - 德卢内市</b> 电话: 31-416-690399 传真: 31-416-690340 <b>挪威 - 特隆赫姆</b> 电话: 47-72884388 <b>波兰 - 华沙</b> 电话: 48-22-3325737 <b>罗马尼亚 - 布加勒斯特</b> 电话: 40-21-407-87-50 <b>西班牙 - 马德里</b> 电话: 34-91-708-08-90 传真: 34-91-708-08-91 <b>瑞典 - 哥德堡</b> 电话: 46-31-704-60-40 <b>瑞典 - 斯德哥尔摩</b> 电话: 46-8-5090-4654 <b>英国 - 沃金厄姆</b> 电话: 44-118-921-5800 传真: 44-118-921-5820
<b>亚特兰大</b> 德卢斯, 佐治亚州 电话: 678-957-9614 传真: 678-957-1455 <b>奥斯汀, 德克萨斯州</b> 电话: 512-257-3370 <b>波士顿</b> 韦斯特伯鲁, 马萨诸塞州 电话: 774-760-0087 传真: 774-760-0088 <b>芝加哥</b> 艾塔斯卡, 伊利诺伊州 电话: 630-285-0071 传真: 630-285-0075 <b>达拉斯</b> 阿迪森, 德克萨斯州 电话: 972-818-7423 传真: 972-818-2924 <b>底特律</b> 诺维, 密歇根州 电话: 248-848-4000 <b>休斯顿, 德克萨斯州</b> 电话: 281-894-5983 <b>印第安纳波利斯</b> 诺布尔斯特维尔, 印第安纳州 电话: 317-773-8323 传真: 317-773-5453 电话: 317-536-2380 <b>洛杉矶</b> 米慎维荷, 加利福尼亚州 电话: 949-462-9523 传真: 949-462-9608 电话: 951-273-7800 <b>罗利, 北卡罗来纳州</b> 电话: 919-844-7510 <b>纽约, 纽约州</b> 电话: 631-435-6000 <b>圣何塞, 加利福尼亚州</b> 电话: 408-735-9110 电话: 408-436-4270 <b>加拿大 - 多伦多</b> 电话: 905-695-1980 传真: 905-695-2078			