

### 如何校准 STM32L1xx 内部 RC 振荡器

#### 前言

STM32L1xx 微控制器有两个内部 RC 振荡器可做为系统时钟源。它们是 HSI（高速内部）和 MSI（多速内部）振荡器。HSI 振荡器的典型频率为 16 MHz。MSI 振荡器为低速、低功耗时钟源。

工作温度对 RC 振荡器的精度有影响。在 25 °C 时，HSI 和 MSI 振荡器的典型精度为  $\pm 1\%$ ，但在 -40 到 105 °C 温度范围，精度会降低。

为补偿温度对内部 RC 振荡器精度的影响，STM32L1xx 微控制器的内置一些功能可帮您校准 HSI 和 MSI 振荡器及测量 MSI 和 LSI（低速内部）振荡器频率。

本应用笔记侧重如何校准内部 RC 振荡器：HSI 和 MSI。它展示了两个方法：第一个方法基于寻找具有最低误差的频率，第二个方法是寻找具有最大允许误差的频率。两个方法都通过提供精确参考信号的方式实现。

MSI 和 LSI 振荡器的测量都是通过将振荡器连至定时器输入捕获执行。

- 注：
- 1 在本文中，STM32L1xx 指超低功耗大容量、中容量增强型、中容量器件。
  - 2 超低功耗中容量器件为 STM32L151xx 和 STM32L152xx 微控制器，其闪存密度范围为 64 至 128K 字节。
  - 3 超低功耗中容量增强型器件为 STM32L151xx、STM32L152xx 和 STM32L162xx 微控制器，其闪存密度为 256 K 字节。
  - 4 超低功耗大容量器件为 STM32L151xx、STM32L152xx 和 STM32L162xx 微控制器，其闪存密度为 384 K 字节。

# 目录

<b>1</b>	<b>STM32L1xx 系统时钟</b>	<b>4</b>
<b>2</b>	<b>内部 RC 振荡器校准</b>	<b>6</b>
2.1	校准原理	8
2.2	硬件实现	9
2.2.1	LSE 用作参考频率的情况	9
2.2.2	其它源用作参考频率的情况	10
2.3	内部振荡器校准固件说明	11
2.3.1	具有最小误差的内部振荡器校准	11
2.3.2	具有固定误差的 HSI 校准	13
2.3.3	具有固定误差的 MSI 校准	15
2.3.4	内部振荡器频率测量	17
2.4	校准库使用建议	19
2.5	校准过程性能	19
2.5.1	校准过程的时间	19
<b>3</b>	<b>内部振荡器测量</b>	<b>21</b>
3.1	测量原理	21
3.2	内部振荡器测量固件说明	23
3.3	内部振荡器校准 / 测量演示说明	23
<b>4</b>	<b>结论</b>	<b>26</b>
<b>5</b>	<b>修订历史</b>	<b>27</b>

图片索引

图 1. 简化的时钟树 ..... 4

图 2. HSI 振荡器微调特性 ..... 6

图 3. MSI 微调行为 ..... 8

图 4. 内部振荡器校准时序图 ..... 9

图 5. 使用 LSE 作为参考频率时的硬件连接 ..... 10

图 6. 使用外部参考频率时的硬件连接 ..... 10

图 7. 内部振荡器校准：找到最小频率误差 ..... 12

图 8. “弹簧环” ..... 13

图 9. HSI 校准流程图：最大允许频率误差 ..... 14

图 10. MSI 校准流程图：最大允许频率误差 ..... 16

图 11. 内部振荡器频率测量流程图 ..... 18

图 12. MSI 测量配置 ..... 21

图 13. LSI 测量配置 ..... 22

图 14. 内部 RC 振荡器测量的时序图 ..... 22

图 15. HSI 校准 ..... 24

图 16. MSI 校准 ..... 25

# 1 STM32L1xx 系统时钟

STM32L1xx 微控制器系列有多个时钟源可用于驱动系统时钟：

- 16 MHz 高速内部（HSI）RC 振荡器时钟
- 1 至 24 MHz 高速外部（HSE）振荡器时钟
- 65.5 kHz 至 4.2 MHz 多速内部（MSI）RC 振荡器时钟
- 2 至 24 MHz 锁相环（PLL），由 HSI 或 HSE 振荡器提供时钟

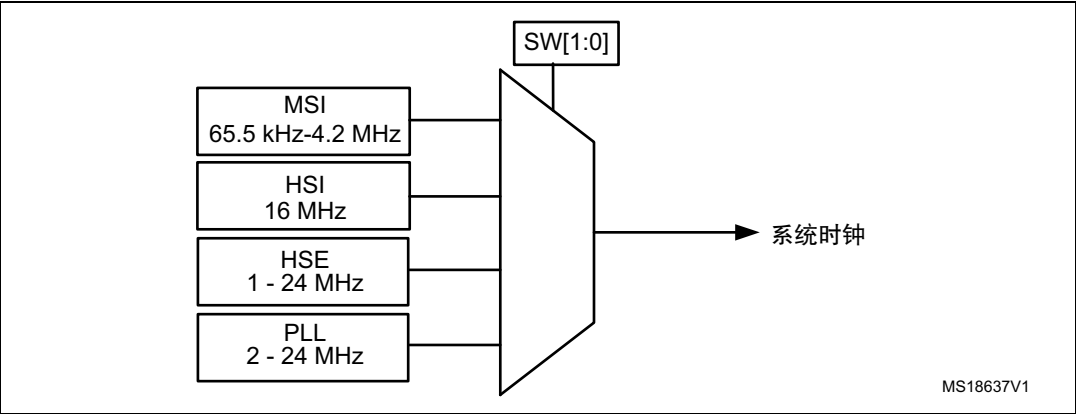
高速内部（HSI）RC 振荡器一般频率为 16 MHz，耗电 100 μA。

多速内部（MSI）RC 振荡器提供了七个频率范围：65.5 kHz、131 kHz、262 kHz、524 kHz、1.05 MHz、2.1 MHz（默认值）和 4.2 MHz。它的设计为工作电流与频率成正比（请参考产品手册以获得 MSI 耗电与所选频率范围之间关系的更详细信息），因此当 CPU 在低频运行时，内部振荡器耗电可最小化。在从复位重启、从停止唤醒、待机低功耗模式后，MSI 时钟被用作系统时钟。

内部 RC 振荡器（HSI 和 MSI）提供了低成本时钟源（不需外部元件），这是其优点。它还具有比外部振荡器更快的启动时间和更低的功耗。可校准 HSI 和 MSI 振荡器以提高其精度。但即使校准后，内部RC振荡器频率也不如外部晶振或陶瓷谐振器的频率精度（几十ppm）高。

注：若外部振荡器发生故障，内部 RC 振荡器（HSI 和 MSI）还可作为备份时钟源（辅助时钟）使用。

图 1. 简化的时钟树



STM32L1x 器件还有两个次级时钟源（不能用作系统时钟源）：

- 37 kHz 低速内部（LSI）RC，设计为驱动独立看门狗，也可驱动实时时钟（RTC）。该 LSI 振荡器无法校准，但可对其测量以评估（由温度和电压变化引起的）频率偏差
- 32.768 kHz 低速外部晶振（LSE 晶振），用于驱动实时时钟 (RTC)

## 2 内部 RC 振荡器校准

由于生产过程的不同，每个芯片的内部 RC 振荡器的频率都可能不同。因此，MSI 和 HSI RC 振荡器由 ST 工厂校准，以获得  $T_A = 25\text{ }^{\circ}\text{C}$  时 1% 的精度。复位后，工厂校准值自动加载到内部校准位中。

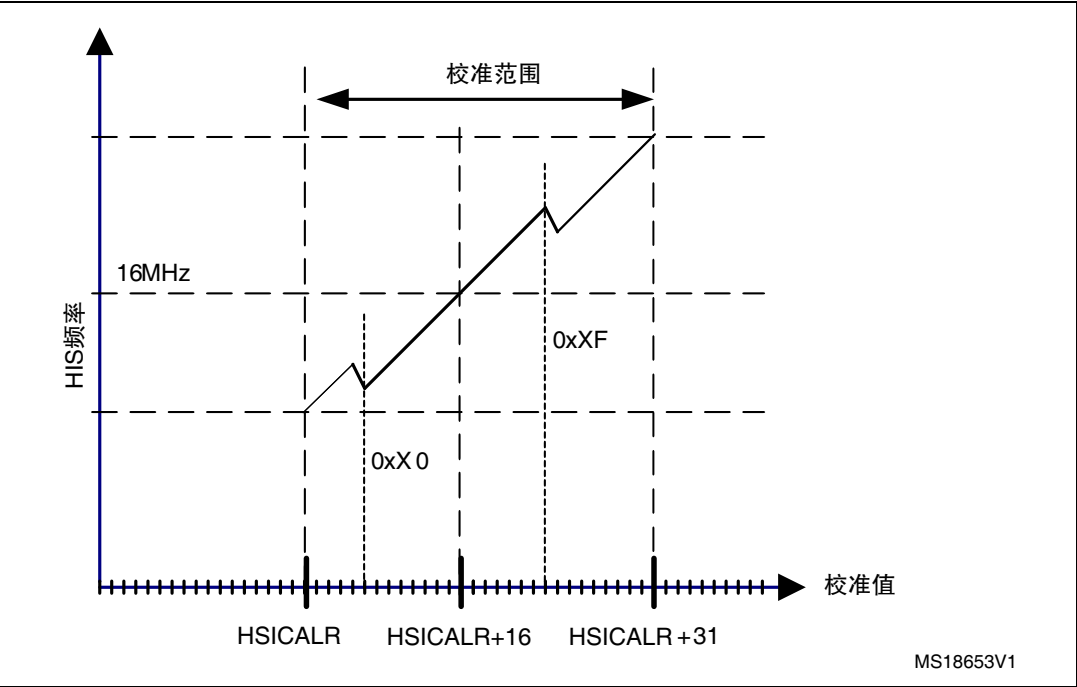
可微调内部 RC 振荡器的频率，以在更宽的温度和供电范围内达到更好的精度。这就是微调位的作用。

对于 HSI 振荡器，复位后校准值加载于 HSICAL[7:0] 位中。有五个微调位 HSITRIM[4:0] 用于微调。默认微调值为 16。增加 / 降低此微调值会增加 / 降低 HSI 频率。HSI 振荡器微调步长为 0.5%（在 80 kHz 周围）。

- 将微调值写为 17 至 31 会增加 HSI 频率。
- 将微调值写为 0 至 15 会降低 HSI 频率。
- 将微调值写为 16，HSI 频率会保持为默认值。

下图显示了随校准值变化的 HSI 振荡器行为。HSI 振荡器频率随校准值增加（校准值 = 默认的 HSICAL[7:0] + HSITRIM[4:0]），除了在模 16 处。在这些校准值处，负步进会达到正步进的三倍。

图 2. HSI 振荡器微调特性



- 对于 MSI 振荡器，复位后校准值加载于 MSICAL[7:0] 位中。使用了八个微调位 MSITRIM[7:0]，因此有更宽的调整范围。校准基于将默认的 MSICAL[7:0] 位（复位值）加上 MSITRIM[7:0] 位。

结果储存于 MSICAL[7:0] 中：

$$\text{MSICAL}[7:0] = \text{默认的 MSICAL}[7:0] + \text{MSITRIM}[7:0]$$

示例：

假设默认的 MSI 校准值 MSICAL[7:0] 为 0x80。

1. 将 0x01 到 0x7F 之间的值写入 MSITRIM[7:0] 会导致校准值 MSICAL[7:0] 的范围为：

$$\text{MSICAL}[7:0] = 0x80 + 0x01 = 0x81$$

$$\text{及 MSICAL}[7:0] = 0x80 + 0x7F = 0xFF$$

这些结果大于 0x80（默认的 MSI[7:0] 值），因此 MSI 频率增加 1 步（0x81 - 0x80）至 127 步（0xFF - 0x80）。

2. 将 0x81 到 0xFF 之间的值写入 MSITRIM[7:0] 会导致校准值 MSICAL[7:0] 的范围为：

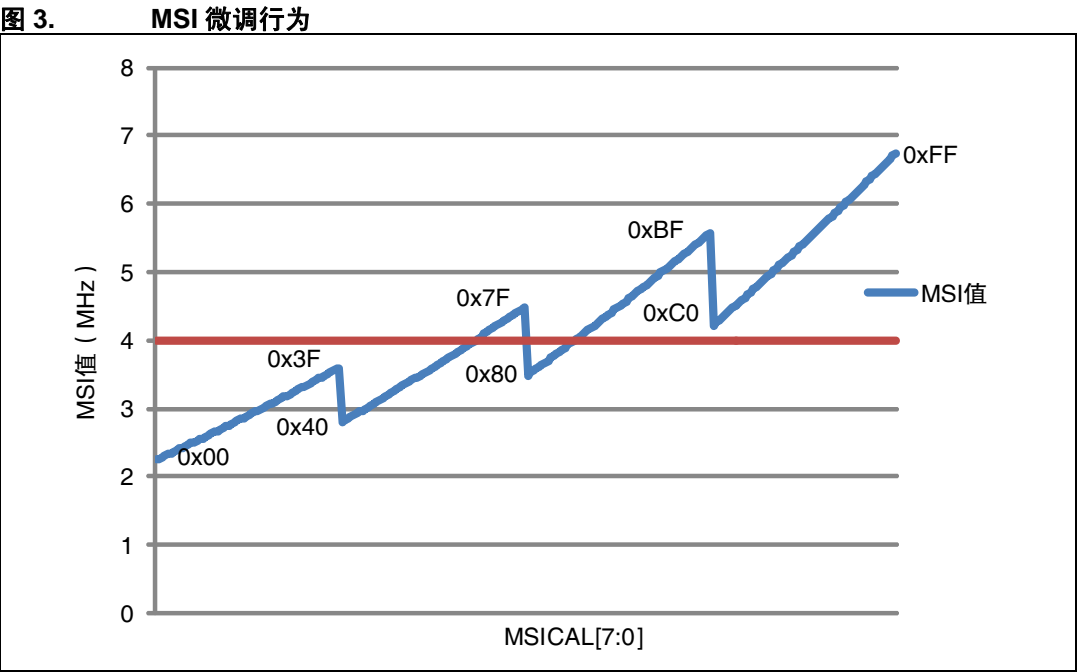
$$\text{MSICAL}[7:0] = 0x80 + 0x81 = 0x01$$

$$\text{及 MSICAL}[7:0] = 0x80 + 0xFF = 0x7F$$

这些结果小于 0x80（默认的 MSI[7:0] 值），因此 MSI 频率降低 1 步（0x01）至 127 步（0x7F）。

3. 在 MSITRIM[7:0] 中写入默认校准值（0x80）会导致校准值 MSICAL[7:0] 等于  $\text{MSICAL}[7:0] = 0x80 + 0x80 = 0x00$ ，因此 MSI 频率会降低 128 步（最低频率）。

下面的图 3 显示了在范围 6（4194304 Hz）处 MSI 行为随 MSICAL[7:0] 的变化。



请注意当两个 MSB 位从“00”到“01”、从“01”到“10”、从“10”到“11”时的负步进。对六个 LSB 的更改会导致每步大约 0.5% 的微调。

## 2.1 校准原理

校准原理包括：

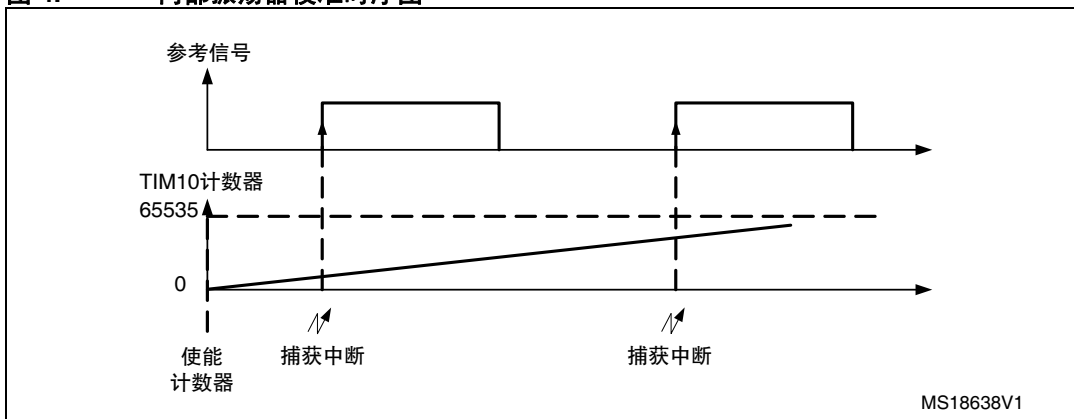
1. 将（需要校准的）内部 RC 振荡器设置为系统时钟，
2. 测量每个微调值的内部 RC 振荡器（HSI 或 MSI）频率，
3. 计算每个微调值的频率误差，
4. 最后，将微调位设置为最优值（对应于最低频率误差）。

内部振荡器频率并非直接测量，而是使用定时器计算时钟脉冲个数，与典型值比较。为此，必须有一个非常精确的参考频率，比如由外部 32.768 kHz 晶振或 50 Hz/60 Hz 电源频率（请参考第 2.2.2 章节：其它源用作参考频率的情况）。

下图显示了怎样使用定时器计数个数测量参考信号周期。



图 4. 内部振荡器校准时序图



启用定时器计数后，当第一个参考信号上升沿发生时，捕捉定时器计数值，储存于 IC1ReadValue1 中。在第二个上升沿，又捕捉到定时器计数，储存于 IC1ReadValue2 中。在两个连续上升沿之间的时间（IC1ReadValue2 - IC1ReadValue1）表示了参考信号的整个周期。

因为定时器计数器的时钟由系统时钟提供（内部 RC 振荡器 HSI 或 MSI），因此与参考信号有关的内部 RC 振荡器生成的真正频率为：

$$\text{Measuredfrequency} = (\text{IC1ReadValue2} - \text{IC1ReadValue1}) \times \text{referencefrequency}$$

误差（单位 Hz）为测量频率与典型值之差的绝对值。

因此，内部振荡器频率误差表示为：

$$\text{Error(Hz)} = |\text{Measuredfrequency} - \text{typicalvalue}|$$

对每个微调值计算误差之后，算法会决定最优微调值（对应于最接近典型值的频率），编程于微调位中（请参考第 2.3 章节：内部振荡器校准固件说明以获取更详细信息）。

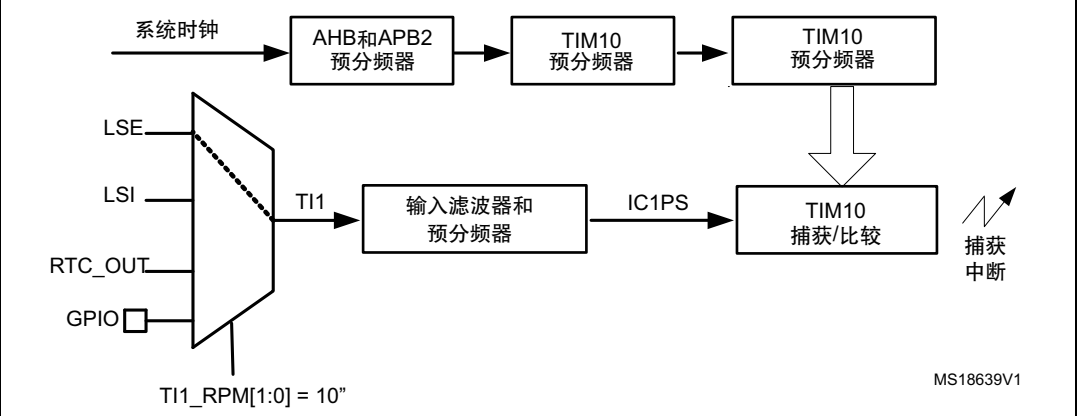
## 2.2 硬件实现

### 2.2.1 LSE 用作参考频率的情况

STM32L1xx 提供了一个有用的特性，即能够将低速外部（LSE）振荡器内部连至定时器 10 通道 1。这样，LSE 时钟可用作内部振荡器校准的参考信号，而不需要额外硬件连接。仅 LSE 振荡器应连至 OSC32\_IN 和 OSC32\_OUT。

图 5 显示了使用 LSE 作为校准的精确频率源时，内部振荡器校准需要的硬件连接。

图 5. 使用 LSE 作为参考频率时的硬件连接

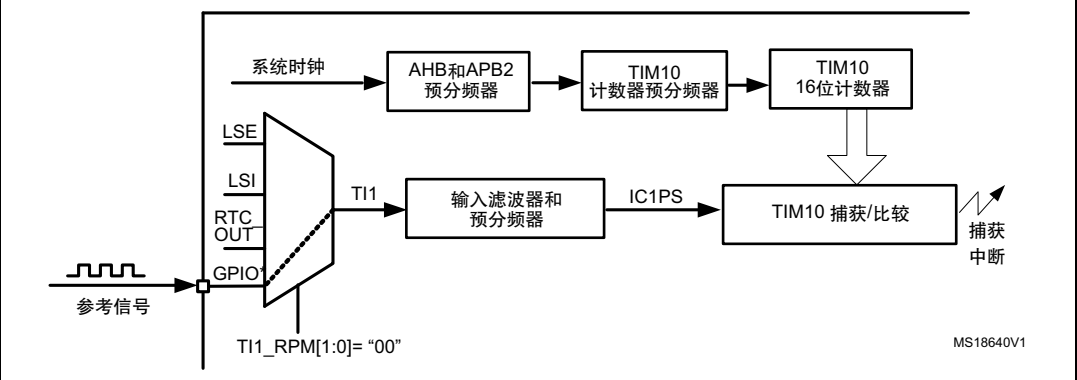


在上图中：  
TI1\_RPM: TIM10 输入 1 重映射  
TI1\_RPM: 定时器输入 1  
IC1PS: 输入捕获 1 预分频

2.2.2 其它源用作参考频率的情况

任何有精确频率的信号都可用于内部振荡器校准，例如主频。  
如下面的图 6 所示，参考信号应连至定时器 10 通道 1。

图 6. 使用外部参考频率时的硬件连接



在上图中：  
TI1\_RPM: TIM10 输入 1 重映射  
TI1: 定时器输入 1  
IC1PS: 输入捕获 1 预分频  
\*PA6、PB8、PB12 或 PE0 可用作 TIM10 通道 1

注：当使用外部信号作为参考时，可不使用 TIM10，而使用定时器 TIM2、TIM3、TIM4、TIM9 或 TIM11。本应用笔记提供的固件使用的是 TIM10。  
请参考应用笔记 AN2868“内部 RC 振荡器（HSI）校准”以获取使用主频校准的更详细信息。

## 2.3 内部振荡器校准固件说明

本应用笔记提供的内部 RC 振荡器校准固件包括三个主要函数：

- `uint32_t InternOsc_CalibrateMinError(void)`
- `uint32_t HSI_CalibrateFixedError(void)`
- `uint32_t MSI_CalibrateFixedError(void)`

### 2.3.1 具有最小误差的内部振荡器校准

`InternOsc_CalibrateMinError()` 将内部振荡器校准至最接近典型值的频率。它测量不同微调值的所有频率，并提供对应于最小误差频率的微调值。以此获得的微调值编程于微调位中。

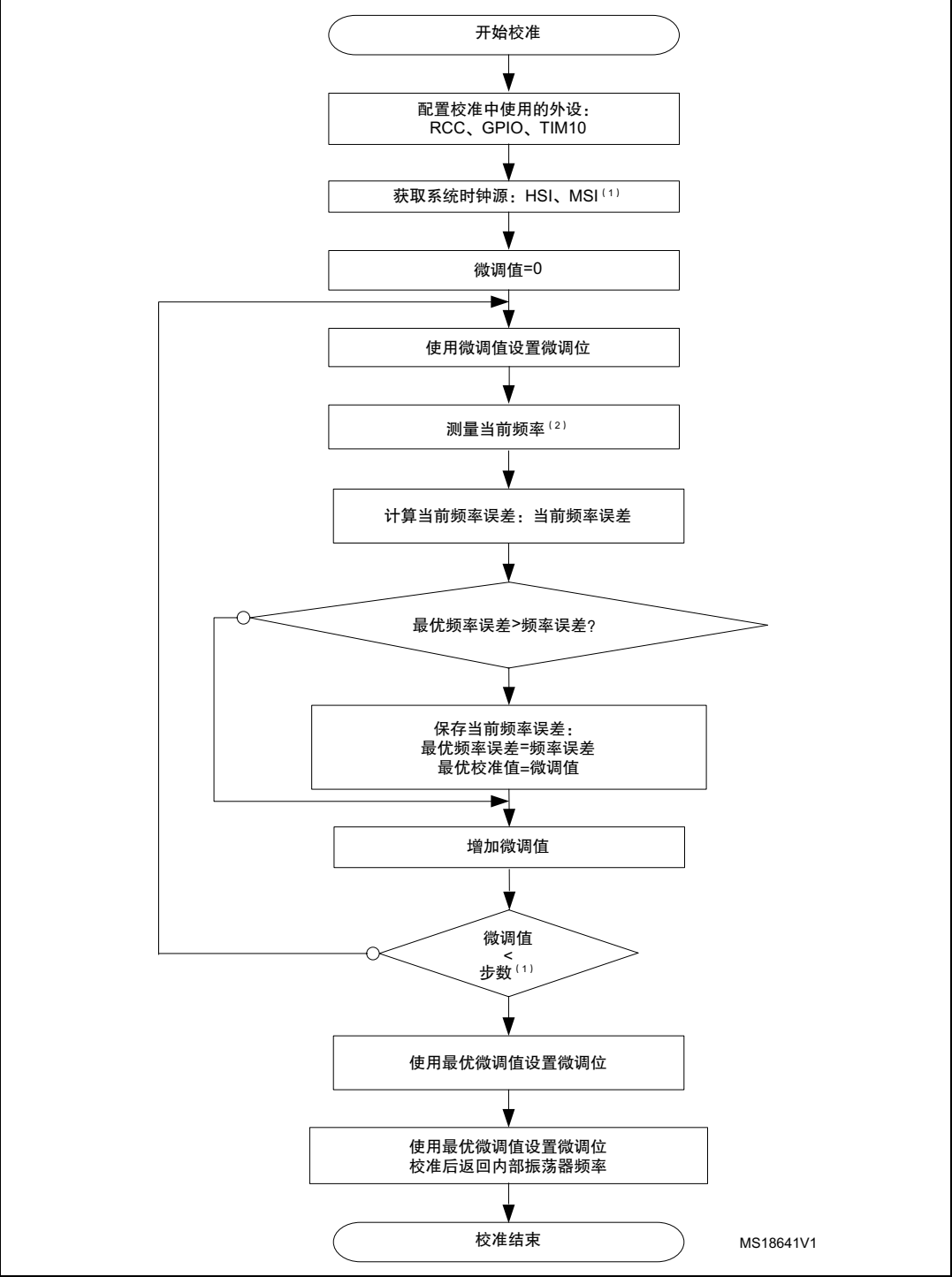
校准后，`InternOsc_CalibrateMinError()` 函数会返回内部振荡器频率值，类型为无符号 32 位整数（`uint32_t`）。

[图 7](#) 中的流程图提供了此函数的算法。

#### 示例

```
uint32_t InternOscAfterCalib = 0;
{
    .....
    /* 得到校准后的内部振荡器（HSI/MSI）值 */
    InternOscAfterCalib = InternOsc_CalibrateMinError();
}
```

图 7. 内部振荡器校准：找到最小频率误差



1. 若系统时钟源为 HSI，则微调位长度为 5 位，步数为 32。若系统时钟源为 MSI，则微调位长度为 8 位，步数为 256。

2. [第 2.3.4 章节](#)中详细说明了频率测量。

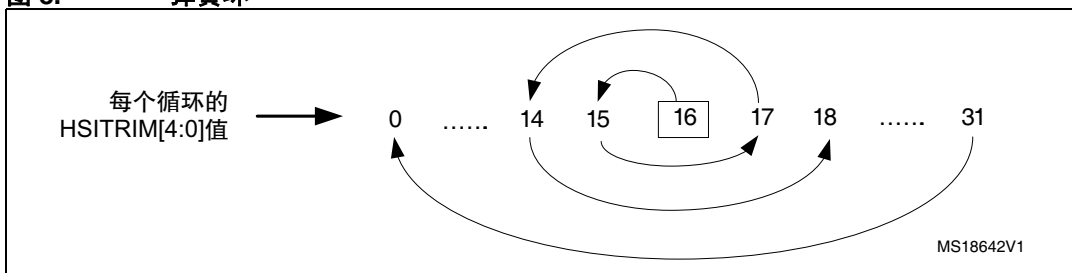


### 2.3.2 具有固定误差的 HSI 校准

HSI\_CalibrateFixedError() 函数可将 HSI 振荡器校准为最大允许的频率误差之内。该误差绝对值由用户配置，单位赫兹（第一个参数: *MaxAllowedError*）。此函数与 InternOsc\_CalibrateMinError() 相同（请参考第2.3.1章节: [具有最小误差的内部振荡器校准](#)），但它会搜索误差小于等于 *MaxAllowedError* 的频率。

- 若它找到了此频率，则停止搜索，根据此频率配置微调位 HSITRIM[4:0] 并返回 SUCCESS，意为校准操作成功。
  - 否则，它会继续搜索，直到 HSITRIM 位 = 31（第 32 个频率）。然后它会将微调位 HSITRIM[4:0] 设为默认微调值并返回 ERROR，意为校准失败，没有发现误差小于等于 *MaxAllowedError* 的频率。
- 频率测量从 HSTRIM = 16 开始。在循环中计算 HSITRIM 的值，以找到下一个值。即，HSITRIM 的值从 16 开始，然后变为左边的下一个值，然后变为右边的下一个值，然后变为左边的第二个值，以此类推，直到到达 31，形成一个“弹簧环”（如[图 8](#)所示）。此算法的原理为，当 HSITRIM[4:0] 的值越接近 16 时，找到最小误差频率的概率越高。因此，此算法的实现可最小化校准过程消耗的时间。

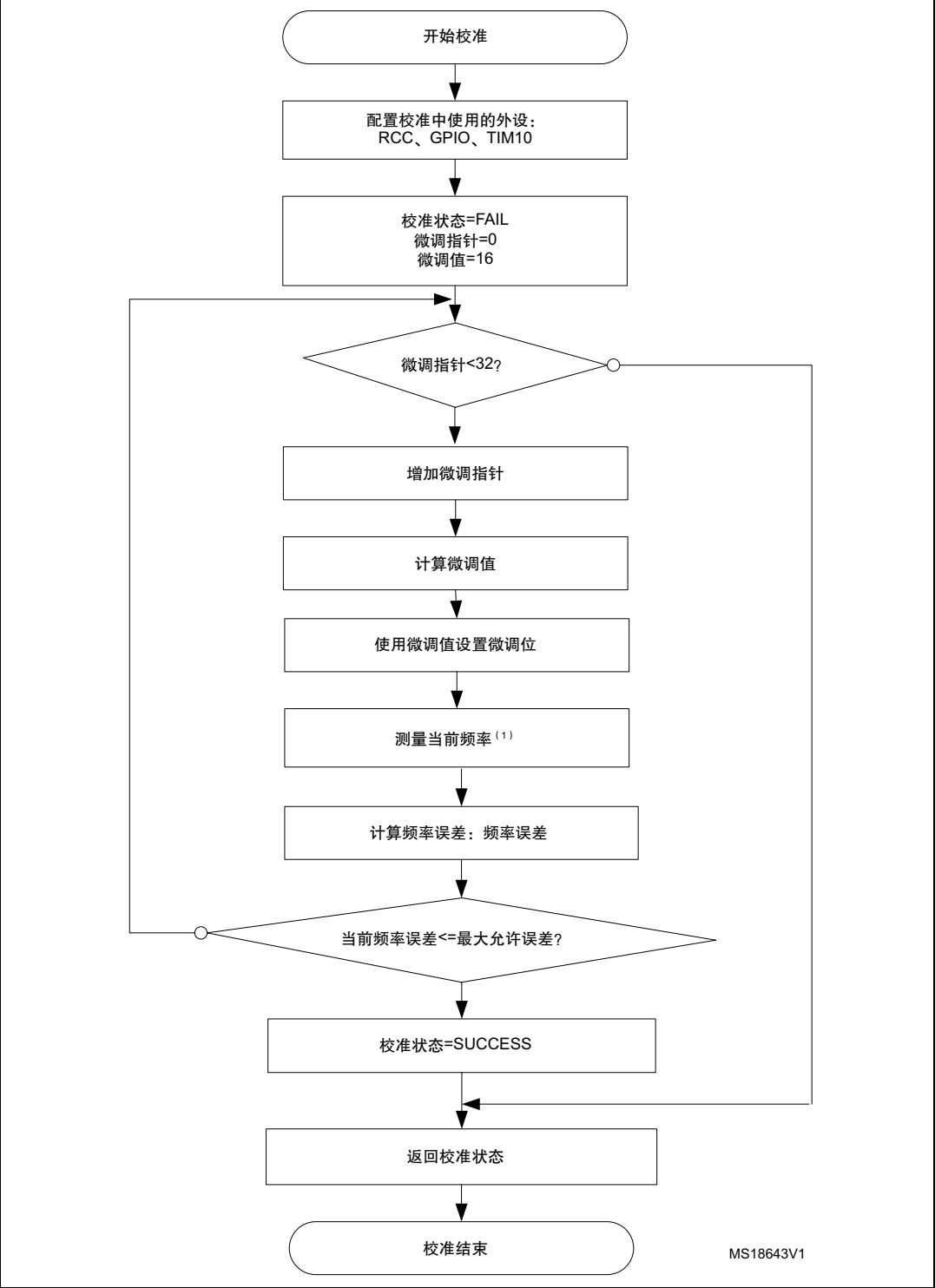
图 8. “弹簧环”



第二个参数是用于在校准后得到频率（单位赫兹），其类型为无符号 32 位整数（unit32\_t）。

图 9 中的流程图提供了此函数的算法。

图 9. HSI 校准流程图：最大允许频率误差



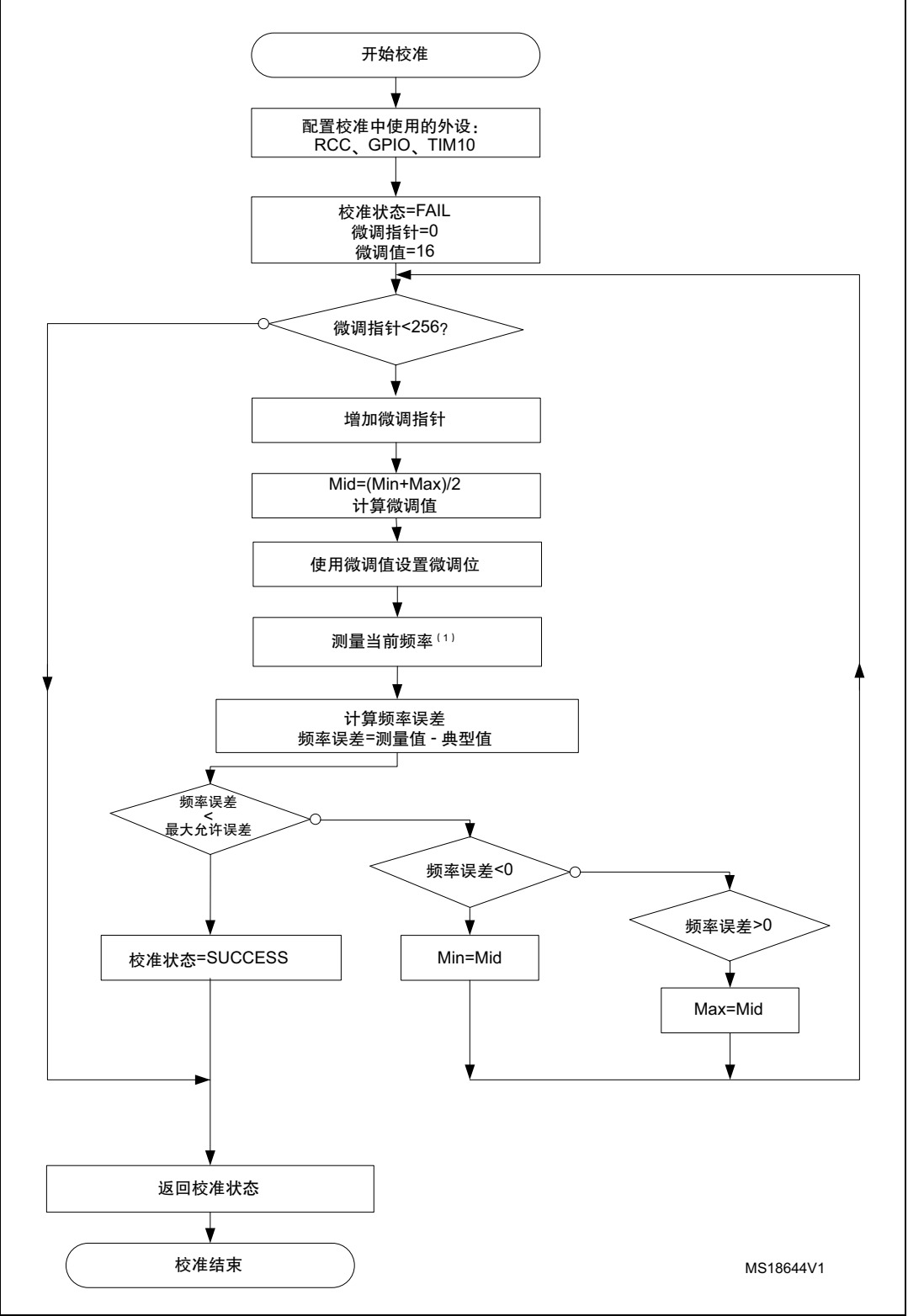
1. 第 2.3.4 章节: 内部振荡器频率测量中详细说明了频率测量。

### 2.3.3 具有固定误差的 MSI 校准

MSI\_CalibrateFixedError() 函数可在最短时间内（而不是扫描所有 256 个微调值），将 MSI 振荡器校准于最大允许频率误差之内。允许误差绝对值由用户配置，单位赫兹（第一个参数：MaxAllowedError）。此函数与 HSI\_CalibrateFixedError() 相同（请参考[第 2.3.1 章节：具有最小误差的内部振荡器校准](#)），区别仅在于它使用二进制搜索算法校准 MSI 振荡器。

[图 10](#) 中的流程图提供了此函数的算法。

图 10. MSI 校准流程图：最大允许频率误差



1. 第 3 章节：内部振荡器测量中详细说明了频率测量。



### 2.3.4 内部振荡器频率测量

通过定时器 10 捕获中断执行内部振荡器频率测量。在定时器 TIM10 ISR 中，计算了内部振荡器频率的完整周期。为每个微调值测量的周期数由用户在 *InternOscCalibration.h* 文件中如下配置：

```
#define NUMBER_OF_LOOPS 10 /* 测量的周期数 = 10 */
```

使用平均方法使频率测量误差最小。因此，若循环计数器达到 NUMBER\_OF\_LOOPS，则计算所有测量频率的平均值。

您可很方便地配置参考源的频率。它在头文件 *InternOscCalibration.h* 中如下定义：

- 若 LSE 时钟用作参考频率，则请去掉下面一行的注释，以确保配置了 LSE，并内部连接至定时器 10 通道 1：

```
#define USE_REFERENCE_LSE
```

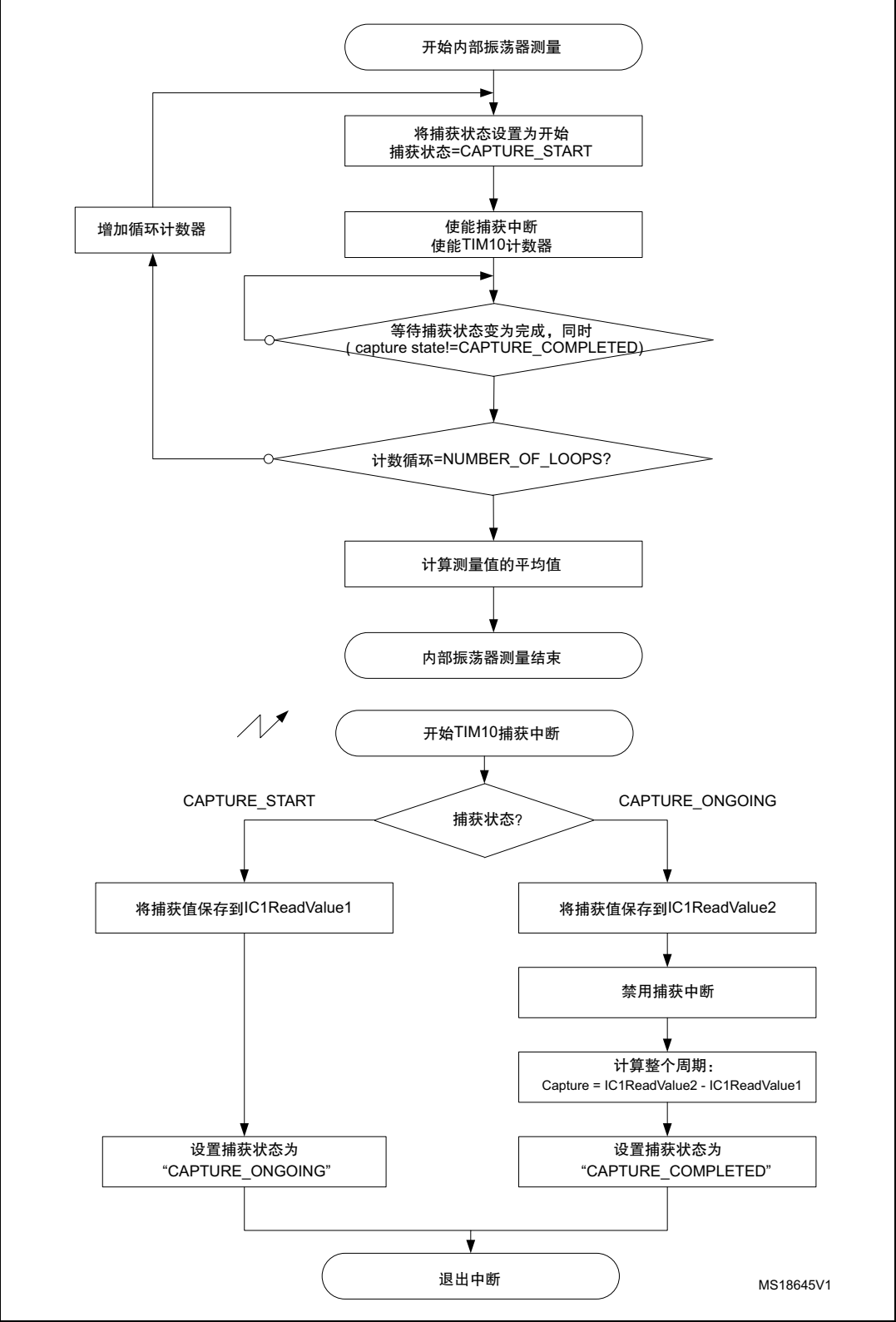
- 若参考频率为 50 Hz 的主源频率，则请注释上面一行，并如下定义参考频率：

```
#define REFERENCE_FREQUENCY (uint32_t)50 /* 参考频率值，单位 Hz */
```

频率测量的计算不依赖于源参考信号的占空比。它取决于其频率，因为捕获 1 中断配置为在每个参考信号的上升沿发生（请参考图 4）。

注：图 11 提供了频率测量算法。

图 11. 内部振荡器频率测量流程图



## 2.4 校准库使用建议

1. 若外部信号频率低于系统时钟 /65535，则应使用 TIM10 计数器预分频支持低频。
2. 若外部信号频率高于系统时钟 /100，则应使用 TIM10 输入捕获预分频器（分频器）支持高频。
3. 建议在校准过程之前停止所有应用活动，并在调用校准函数后重新启动它们。  
因此，应用必须停止通信、ADC 测量和其它过程（除了将 ADC 用于校准时的情况。请参考下面的 5. 步骤）。  
这些过程一般使用与校准过程中不同的时钟配置。否则，应用中可能引入错误：读 / 发帧时的错误、采样时间更改后的 ADC 读错误等等。
4. 内部 RC 振荡器校准固件使用下列外设：复位和时钟控制（用于微调内部 RC 振荡器）、定时器 10（用于测量内部 RC 振荡器）。因此，建议在运行校准程序后重新配置这些外设（若用于应用）。
5. 若应用运行时，环境温度显著改变，则可使用实时校准 vs. 温度。内部温度传感器可与具有两个门限的 ADC 看门狗共同使用。每次发生 ADC 看门狗中断时，都必须执行新的校准过程，根据当前温度更新两个门限（本应用笔记提供的固件中未实现此特性）：  

$$\text{Threshold\_High} = \text{CurrentTemperatureValue} + \text{TemperatureOffset}$$

$$\text{Threshold\_Low} = \text{CurrentTemperatureValue} - \text{TemperatureOffset}$$

## 2.5 校准过程性能

### 2.5.1 校准过程的时间

校准过程的时间取决于：

1. 参考信号的频率（预分频值）“REFERENCE\_FREQUENCY”、
2. 每个微调值的测量周期个数“NUMBER\_OF\_LOOPS”、
3. 校准过程期间测量频率的个数“步数”。

外设配置好并就绪之后（主要是 LSE 振荡器），校准过程的时间约为：

$$\text{时间} = (2 \times (\text{NUMBER\_OF\_LOOPS} + 1) \times \text{步数}) / \text{REFERENCE\_FREQUENCY}$$

若校准过程运行的是 HSI 振荡器最小频率误差（*InternOsc\_CalibrateMinError()*），则步数等于 32。若使用 LSE 振荡器作为参考频率（REFERENCE\_FREQUENCY = LSE 值 / 输入捕获预分频 = 32768/8 = 4096 Hz），所选的测量周期数为 10，则校准大约耗时：

$$\text{时间} = (2 \times 11 \times 32) / 4096 = 172 \text{ 毫秒}$$

当运行 MSI 振荡器的校准过程时，步数等于 256。若使用 LSE 振荡器作为参考频率（REFERENCE\_FREQUENCY = LSE 值 / 输入捕获预分频 =  $32768/8 = 4096$  Hz），所选的测量周期数为 10，则校准大约耗时：

$$\text{时间} = (2 \times 11 \times 256) / 4096 = 1.4 \text{ 秒}$$

使用最大允许误差时的校准过程时间小于等于使用最小频率误差过程时的校准时间。

注： 在上面时间公式中乘以 2 是因为在参考信号和由定时器启动计数之间没有同步。

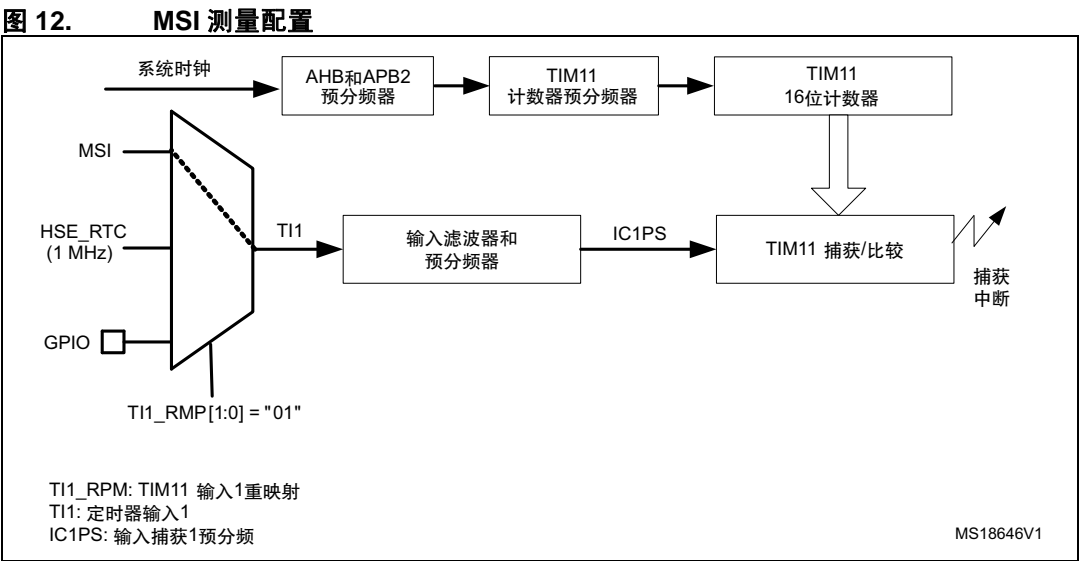
### 3 内部振荡器测量

内部的 MSI 和 LSI RC 振荡器为低功耗和低成本时钟源。在 STM32L1x 微控制器系列中，内部 RC 振荡器（MSI 和 LSI）和嵌入式定时器（TIM10 和 TIM11）之间有内部连接，以方便测量过程。

#### 3.1 测量原理

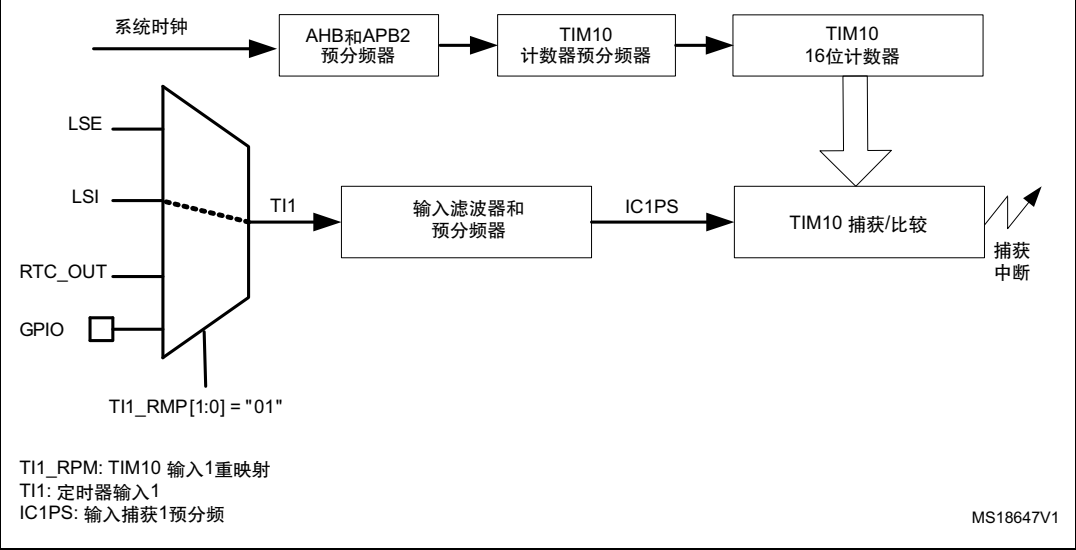
内部 RC 振荡器测量过程包括使用 HSI 时钟运行定时器计数器、将定时器配置于输入捕获模式、将（需要测量的）内部 RC 振荡器连至定时器。

下图显示了执行 MSI 测量所使用的配置。如图所示，TIM11 可内部连至 TIM11 输入 1（TI1）。



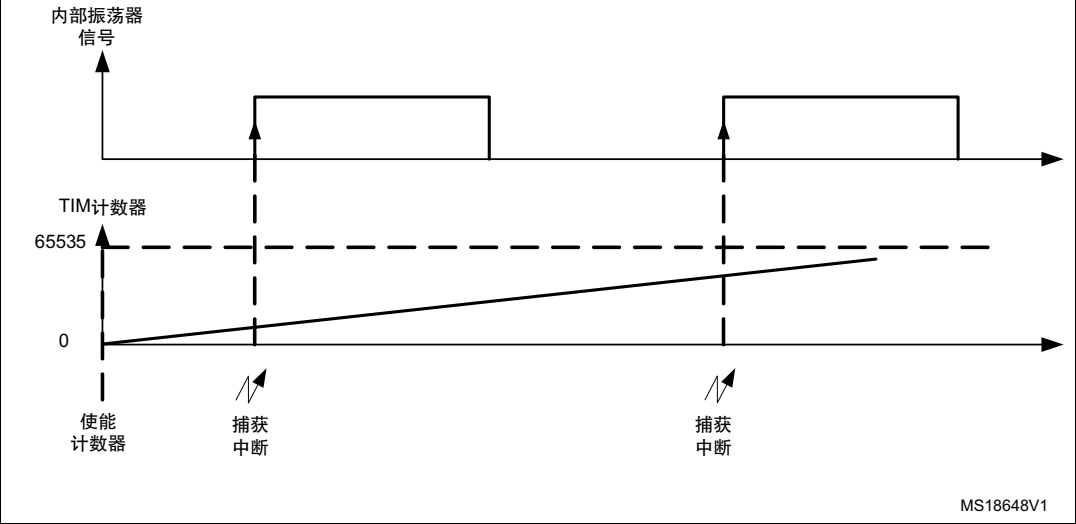
下图显示了执行 LSI 测量所使用的配置。如图所示，TIM10 可内部连至 TIM10 输入 1（TI1）。

图 13. LSI 测量配置



启用定时器计数后，当第一个要测量的内部振荡器信号上升沿发生时，捕捉定时器计数值，储存于 IC1ReadValue1 中。在第二个上升沿，又捕捉到定时器计数，储存于 IC1ReadValue2 中。在时钟的两个连续上升沿之间的时间代表整个周期。下图显示了内部 RC 振荡器测量的时序图。

图 14. 内部 RC 振荡器测量的时序图



内部振荡器频率值如下式所示：

内部振荡器频率 = HSI\_Value / 捕获数

其中：

- HSI\_Value 为 HSI 频率值：典型值为 16 MHz，
- 捕获数表示内部 RC 振荡器（MSI 或 LSI）的整个周期：IC1ReadValue2 - IC1ReadValue1。



从上式您可得出，频率测量精度取决于 HSI 频率精度。因此，若参考信号可用，则您可在执行内部 RC 振荡器测量过程之前，运行 [第 2 章节：内部 RC 振荡器校准](#) 中所述的内部 RC 振荡器校准程序。

可使用输入捕获预分频以得到更好的测量精度，这样上式变为：

$$\text{LSI\_Frequency} = \text{InputCapturePrescaler} * \text{HSI\_Value} / \text{Capture\_Value}.$$

[图 11](#) 中所示的相同算法也用于测量 LSI 和 MSI 振荡器频率。请注意，TIM10 ISR 用于 MSI 测量，TIM11 ISR 用于 LSI 测量。

## 3.2 内部振荡器测量固件说明

本应用笔记提供的内部振荡器测量固件包括两个 C 源文件：

- LSIMeasurement.c 使用 LSI\_FreqMeasure() 函数，执行 LSI 频率测量
- MSIMeasurement.c 使用 MSI\_FreqMeasure() 函数，执行 MSI 频率测量

按照预定义的周期数测量内部 RC 振荡器（MSI 或 LSI）。然后，它返回平均值，以最小化所测频率的误差。

您可在 lsi\_measurement.h 文件中更改此参数（LSI 周期数）：

```
#define LSI_PERIOD_NUMBERS 10
```

用同样方式，可在 msi\_measurement.h 中更改周期数：

```
#define MSI_PERIOD_NUMBERS 10
```

## 3.3 内部振荡器校准 / 测量演示说明

本应用笔记提供的演示显示了固件校准内部 RC 振荡器（HSI 和 MSI）的能力，给出了怎样用它测量 STM32L1x 微控制器内部 RC 振荡器（MSI 和 LSI）的例子。

在运行校准程序之前，请在 system\_STM32L1xx.c 中选择系统时钟源。若 HSI（表示 MSI）振荡器被选为系统时钟源，则校准 HSI（或 MSI）。在此演示中，使用 LSE 振荡器作为参考校准内部 RC 振荡器（MSI 或 HSI）。

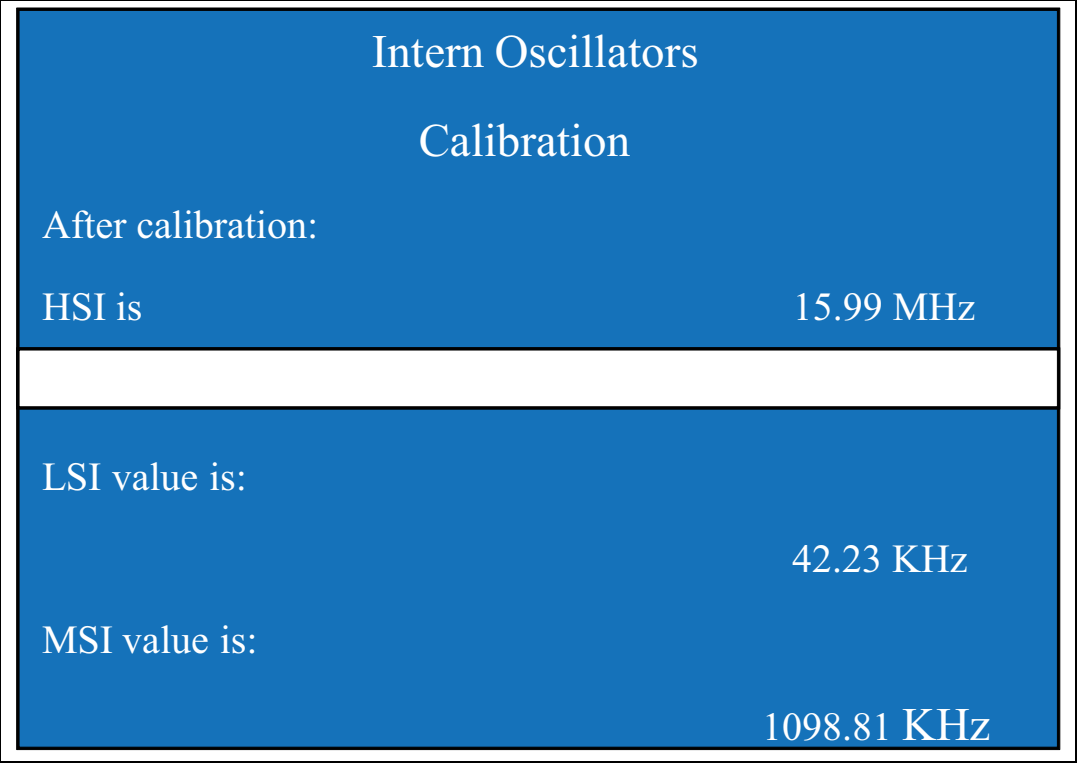
默认情况下，该演示使用最小误差方法校准 HSI 振荡器。

若要使用固定误差频率方法运行校准过程，您必须在 main.c 文件中注释下面的宏定义。

```
#define CALIBRATION_MIN_ERROR
```

下面的图 15 显示了在 STM32L1xx-EVAL 板上实现的情况下，当选择 HSI 为系统时钟源运行演示时，LCD 上显示的消息。校准 HSI 振荡器，然后测量 MSI 和 LSI 振荡器。

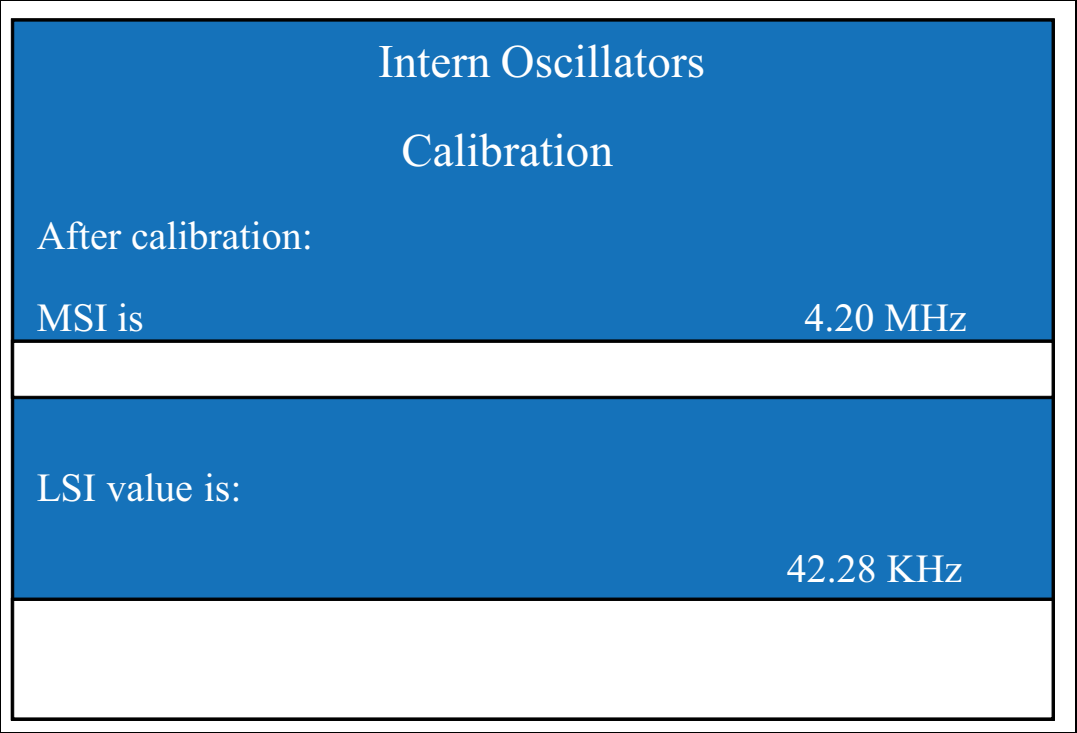
图 15. HSI 校准





下图显示了当选择MSI为系统时钟源运行演示时，LCD上显示的消息。在4 MHz下校准MSI，然后测量LSI 频率。

图 16. MSI 校准



## 4 结论

即使内部 RC 振荡器已经有工厂校准，若应用中需要高精度时钟，则用户还应在工作环境校准它们。

本应用笔记提供了两个程序：

- **多速和高速内部振荡器校准：**怎样将振荡器精调至典型值
- **多速和低速内部振荡器测量：**怎样得到“确切”的 LSI/MSI 频率值

可使用多种频率源校准内部 RC 振荡器（HSI 和 MSI）：LSE 晶振、AC 线等等。不管参考频率源如何，内部振荡器校准原理是相同的：必须提供参考信号，并由定时器测量。参考信号频率精度越高，内部振荡器频率测量的精度就越好。对每个微调值，误差的计算为典型频率值与测量值之差的绝对值。用这个方法计算校准值，然后编程于微调位中。

本应用笔记的第二部分讨论了 LSI 和 MSI 振荡器的测量。在 STM32L1xx 微控制器系列中，内部振荡器和嵌入式定时器之间的内部连接即用于此目的。该定时器使用系统时钟源作为时钟，配置为输入捕获模式。在内部振荡器两个连续上升沿之间捕获的时间表示了整个周期。

## 5 修订历史

表 1. 文档修订历史

日期	修订	变更
2011 年 4 月 07 日	1	初始版本。
2012 年 1 月 31 日	2	在封面页增加了注释，包括了超低功耗中容量增强型及超低功耗大容量器件。

**请仔细阅读：**

中文翻译仅为方便阅读之目的。该翻译也许不是对本文档最新版本的翻译，如有任何不同，以最新版本的英文原版文档为准。

本文档中信息的提供仅与 ST 产品有关。意法半导体公司及其子公司（“ST”）保留随时对本文档及本文所述产品与服务进行变更、更正、修改或改进的权利，恕不另行通知。

所有 ST 产品均根据 ST 的销售条款出售。

买方自行负责对本文所述 ST 产品和服务的选择和使用，ST 概不承担与选择或使用本文所述 ST 产品和服务相关的任何责任。

无论之前是否有任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为 ST 授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在 ST 的销售条款中另有说明，否则，ST 对 ST 产品的使用和 / 或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途（及其依据任何司法管辖区的法律的对应情况），或侵犯任何专利、版权或其他知识产权的默示保证。

意法半导体的产品不得应用于武器。此外，意法半导体产品也不是为下列用途而设计并不得应用于下列用途：（A）对安全性有特别要求的应用，例如，生命支持、主动植入设备或对产品功能安全有要求的系统；（B）航空应用；（C）汽车应用或汽车环境，且 / 或（D）航天应用或航天环境。如果意法半导体产品不是为前述应用设计的，而采购商擅自将其用于前述应用，即使采购商向意法半导体发出了书面通知，采购商仍将独自承担因此而导致的任何风险，意法半导体的产品规格明确指定的汽车、汽车安全或医疗工业领域专用产品除外。根据相关政府主管部门的规定，ESCC、QML 或 JAN 正式认证产品适用于航天应用。

经销的 ST 产品如有不同于本文档中提出的声明和 / 或技术特点的规定，将立即导致 ST 针对本文所述 ST 产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大 ST 的任何责任。

ST 和 ST 徽标是 ST 在各个国家或地区的商标或注册商标。

本文档中的信息取代之前提供的所有信息。

ST 徽标是意法半导体公司的注册商标。其他所有名称是其各自所有者的财产。

© 2014 STMicroelectronics 保留所有权利

意法半导体集团公司

澳大利亚 - 比利时 - 巴西 - 加拿大 - 中国 - 捷克共和国 - 芬兰 - 法国 - 德国 - 中国香港 - 印度 - 以色列 - 意大利 - 日本 - 马来西亚 - 马耳他 - 摩洛哥 - 菲律宾 - 新加坡 - 西班牙 - 瑞典 - 瑞士 - 英国 - 美国

[www.st.com](http://www.st.com)