



# RTC 校准方案 应用指导

文档版本 00B01

发布日期 2012-12-26

**版权所有 © 深圳市海思半导体有限公司 2012。保留一切权利。**

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

### **商标声明**



**HISILICON**、海思和其他海思商标均为深圳市海思半导体有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

### **注意**

您购买的产品、服务或特性等应受海思公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，海思公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

## **深圳市海思半导体有限公司**

地址：深圳市龙岗区坂田华为基地华为电气生产中心 邮编：518129

网址：<http://www.hisilicon.com>

客户服务电话：+86-755-28788858

客户服务传真：+86-755-28357515

客户服务邮箱：[support@hisilicon.com](mailto:support@hisilicon.com)



# 前 言

## 概述

本文档主要介绍 RTC 的校准方案，确保 RTC 计时准确。

## 产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本
Hi3518 芯片	V100

## 读者对象

本文档（本指南）主要适用于以下工程师：

- 技术支持工程师

## 修订记录

修订记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

修订日期	版本	修订说明
2012-12-26	00B01	第 1 次版本发布。



# 目 录

前 言 .....	i
<b>1 概述.....</b>	<b>1-1</b>
<b>2 RTC 的硬件参考电路.....</b>	<b>2-1</b>
2.1 硬件参考电路.....	2-1
2.1 选择晶体.....	2-1
2.2 选择电容.....	2-2
<b>3 RTC 校正功能的实现.....</b>	<b>3-1</b>
3.1 测量晶体温度频率曲线与配置 RTC 温度频率曲线.....	3-1
3.2 动态更新 RTC 校正电路中的温度值.....	3-2
<b>4 RTC 驱动使用说明 .....</b>	<b>4-1</b>
4.1 准备.....	4-1
4.2 编译.....	4-1
4.3 使用.....	4-1
<b>5 Q&amp;A.....</b>	<b>5-1</b>
5.1 振荡器不振.....	5-1
5.2 振荡器的输出频率是 200K .....	5-1
5.3 振荡频率虽然是 32.768K 附近，但是频率却不准？ .....	5-2



# 1 概述

为了保证 RTC 计时准确，Hi3518 中的 RTC 的计时时钟要恒为 100Hz。RTC 计时时钟计算公式为：RTC 的计时时钟 = 内部振荡电路产生的时钟 / 分频比 (327.xx)。

当温度变化时，内部振荡电路产生的时钟受到影响也会变化，此时可通过调节分频比来确保 RTC 计时时钟恒定。RTC 的 32K 晶体的输出振荡频率与温度的关系如图 1-1 所示。

图1-1 温度与晶体频率输出关系

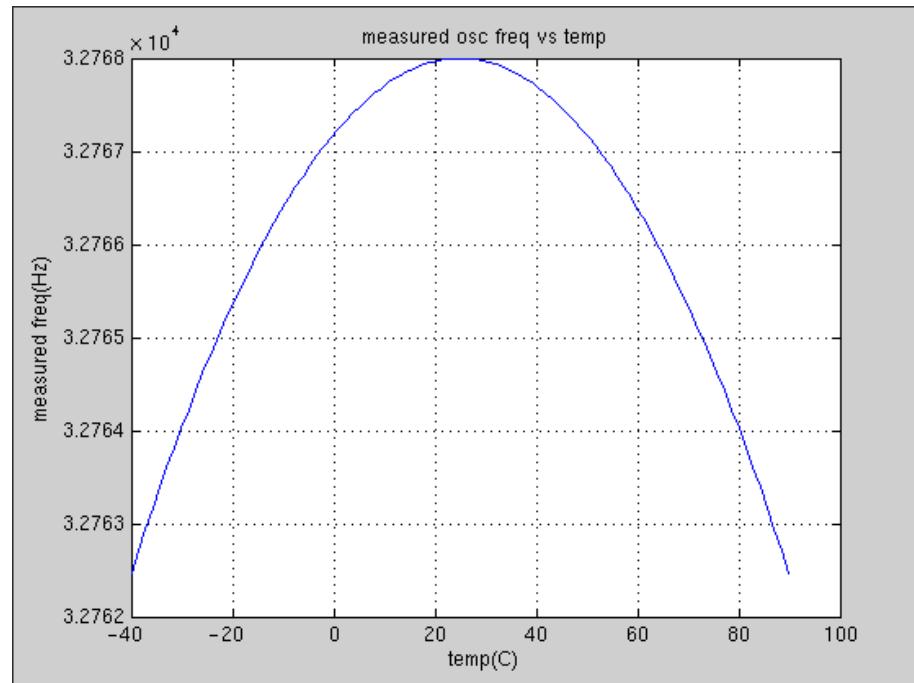


图 1-1 中的曲线可以公式表示： $F = (K_s * (T - T_0)^2 + 1 + C) * 32768$

其中：

- F 为晶体在温度 T 下的振荡频率，单体为 Hz。



- $K_s$  为抛物线二次项系数，和选取的晶体相关，[图 1-1](#) 中的  $K_s$  为  $-4 \times 10^{-8} / ^\circ C^2$ 。
- $T_0$  是抛物线的转折温度点，一般为  $25 \pm 5^\circ C$ 。[图 1-1](#) 中的值为  $24.94^\circ C$ 。（RTC 中一个温度码字表示  $(140^\circ C - (-40^\circ C)) / 255 = 0.705882^\circ C$ ，温度码字和温度的对应关系请参见《RTC 晶体校正参数生成表》）
- $C$  为晶体在转折温度点的频率偏差，[图 1-1](#) 中为 0。 $C$  受以下两部分因素影响：
  - 负载电容
  - 晶体差异

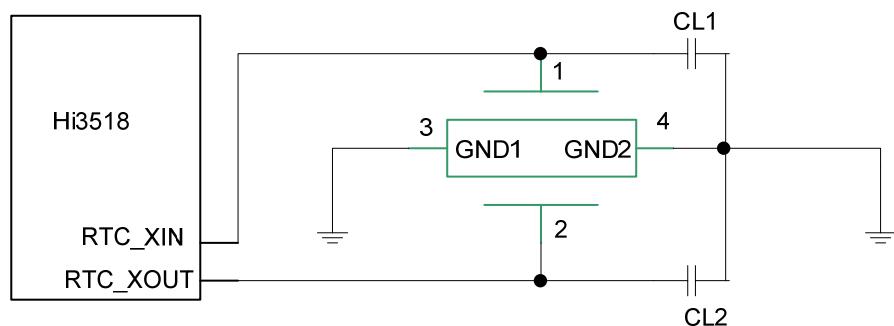


# 2 RTC 的硬件参考电路

## 2.1 硬件参考电路

RTC 的硬件参考电路如图 2-1 所示，主要涉及晶体和电容的选择。

图2-1 RTC 晶体的硬件参考电路



## 2.1 选择晶体

选择晶体需要注意以下几个指标：

- 标准负载电容 (Load capacitance/CL): 晶体的标准负载电容，晶体对负载电容有着严格的规定，只有实际负载电容和晶体的 SPEC 中的负载电容一致的时候，晶体频率才能达到标称频率。  
Hi3518 芯片中的晶体振荡电路针对  $CL=12.5\text{pF}$  的晶体设计，且在 32.768K 晶体市场中， $CL=12.5\text{pF}$  的晶体为市场主流，请选用此规格晶体。
- 串联电阻 (Series resistance/Rs/ESR): 晶体的谐振腔等效串联电阻，当 ESR 越大，表示晶体约难以驱动。晶体规格中会指出 Rs 的典型值与最大值。  
Hi3518 芯片晶体振荡电路适用于  $Rs$  最大值 $<50\text{K}\Omega$  的晶体，请选用满足此规格的晶体。
- 最大驱动级别 (Max Drive Level/DL): 表示晶体最大的振荡幅度，当振荡幅度超过一定幅度时，晶体容易发生损坏。



Hi3518 芯片晶体振荡电路内部限制了 RTC\_XIN 与 RTC\_XOUT 管脚振荡的振荡幅度，可以通过以下公式估计电路工作时的实际 Drive Level，并确定此值小于晶体规格中规定的最大 Drive Level。

$$DL_{actual}=0.5*Rs_{max}*(\pi*f*Vpp_{XIN}*CL*2)^2$$

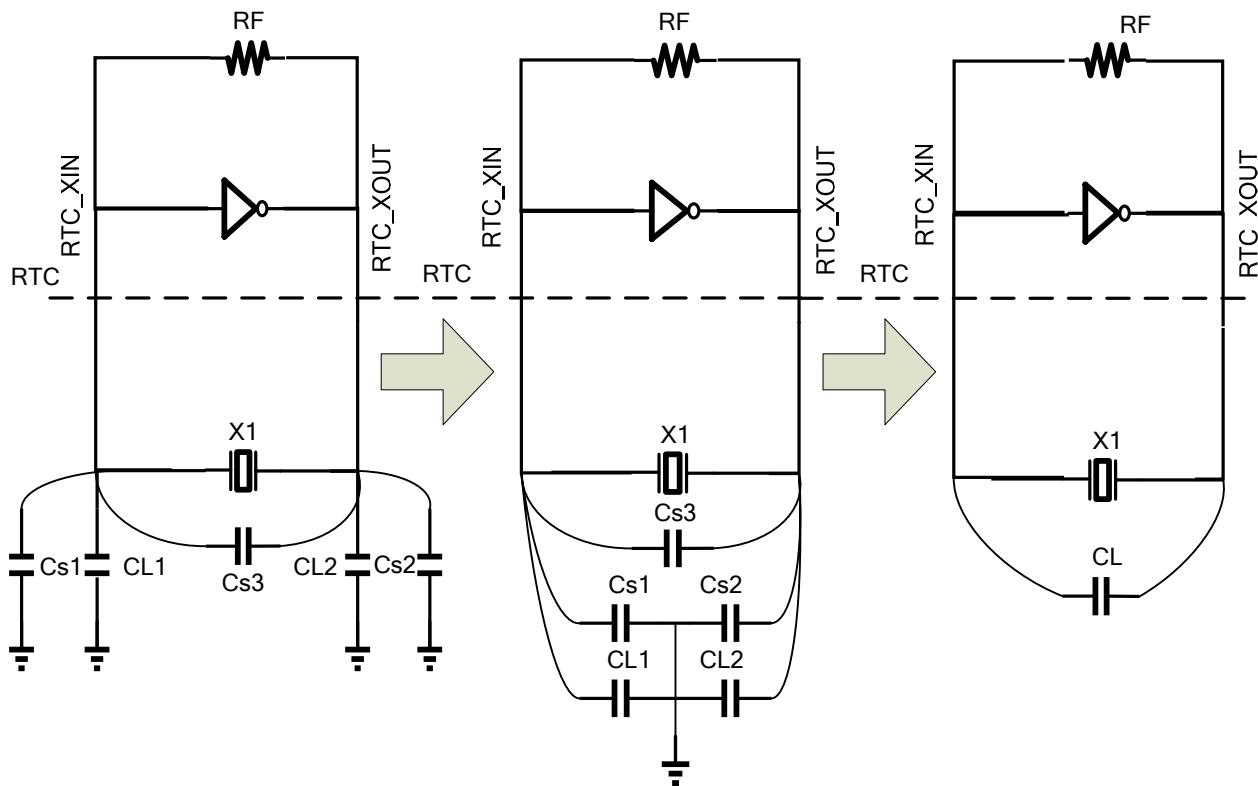
其中：

- $Rs_{max}$  为晶体规格书中的串联电阻的最大值
- $f$  为晶体的谐振频率
- $Vpp_{XIN}$  为示波器测量的 RTC\_XIN 管脚的 peak to peak 电压
- $CL$  为晶体规格书的标准负载电容

## 2.2 选择电容

实际 CL 的示意图如图 2-2 所示。

图2-2 实际 CL 的示意图



Pierce 振荡器中，一般将 CL1 与 CL2 取相同的值，可以通过以下公式来确定 CL1 与 CL2 的取值。

$$CL1=CL2=CL\_SPEC*2-3\sim5pF$$

其中  $CL\_SPEC$  为晶体规格书中规定的标准负载电容， $3\sim5pF$  代表的 PCB 板可能引入的杂散电容。以  $12.5pF$  的晶体为例， $CL1$  与  $CL2$  一般取值为  $12.5pF*2-3pF=22pF$ 。因



为杂散电容随 PCB 板设计不同而变化，故亦可确定 PCB 板以后，通过选取不同容值的 CL1 来获得最接近 32.768K 的输出频率。



# 3 RTC 校正功能的实现

RTC 校正的频率范围为 32760~32776Hz (全温度范围内)。RTC 校正功能的实现，需要两个步骤：

1. 测量晶体温度频率曲线和配置 RTC 温度频率曲线。
2. 动态更新 RTC 校正电路中的温度值。

----结束

## 3.1 测量晶体温度频率曲线与配置 RTC 温度频率曲线

### 测量晶体的温度频率曲线

通过将测量的温度与晶体振荡输出频率写入《RTC 晶体校正参数生成表》，即可生成温度频率曲线。在测量温度频率曲线时，根据测量的精度和成本因素，选择以下几种测量方法：

- 多点测量法。

在全温度范围内测量多个温度与晶体振荡输出频率的关系，该方案适用于晶体参数需要重新评估的情况。

- 三点测量法。

只取-20°C、25°C 和 80°C 下测量到的频率值，该方案适用于快速确定晶体参数。

- 两点测量法。

取-20°C 和 80°C 下测量到的频率值，该方法适合于已知晶体型号的情况下使用，因为对同一型号的晶体温度频率曲线抛物线二次系数 Ks 已知，且变化不明显（晶体一致性较好的情况下）。

- 单点测量法。

只测量室温下的晶体频率输出值，该方法适合于使用同一型号同一批次的晶体，因为此时晶体温度频率曲线抛物线系数二次 Ks 和抛物线的转折温度点 T<sub>0</sub> 已知，且变化不明显（晶体一致性较好的情况下）。



## 配置 RTC 的温度频率曲线

晶体频率输出和温度的曲线对应 RTC 内部寄存器的寄存器，其中寄存器  $LUT_n$  与  $K_s * (T_n - 24.94)^2$  ( $n \in [1, 47]$ ,  $T_n = -40 + n * (180 / 255)$ ) 相关，寄存器 TEMP\_OFFSET 与  $T_0$  相关，寄存器 TOT\_OFFSET 与 C 相关。

选择合适的方法后，将测量到的频率和温度测量数据填入相应的表格中，点击生成即可。将生成的 `rtc_temp_lut_tbl.h` 文件覆盖 RTC 驱动中相应的文件，重新编译 RTC 的驱动，这样芯片在每次启动时就会把曲线配置到 RTC 中。

由于晶体之间的差异，生成的温度和频率曲线不同。综合考虑精度和成本，配置 RTC 驱动文件的方案有以下三种：

- 每个晶体都进行测量，描述出自己特有的曲线进行配置。这种方法精度可以达到 5ppm，但是每个产品都要进行校正曲线，根据生成的 `rtc_temp_lut_tbl.h` 曲线配置文件。
- 对同一批次中一定数目的晶体进行测量，对拟合出的曲线进行再拟合，然后所有的晶体都按这个再拟合的曲线进行配置。这种方法的精度为 5ppm+晶体个体与一致性的偏差，精度受晶体生产的一致性的影响。
- 不做任何测量，采用芯片初始的曲线或者晶体厂商提供的数据生成的曲线配置文件。这种方案的精度完全由晶体的一致性以及晶体固有的温度频率曲线与 RTC 配置的晶体温度频率曲线的吻合度来决定。

## 3.2 动态更新 RTC 校正电路中的温度值

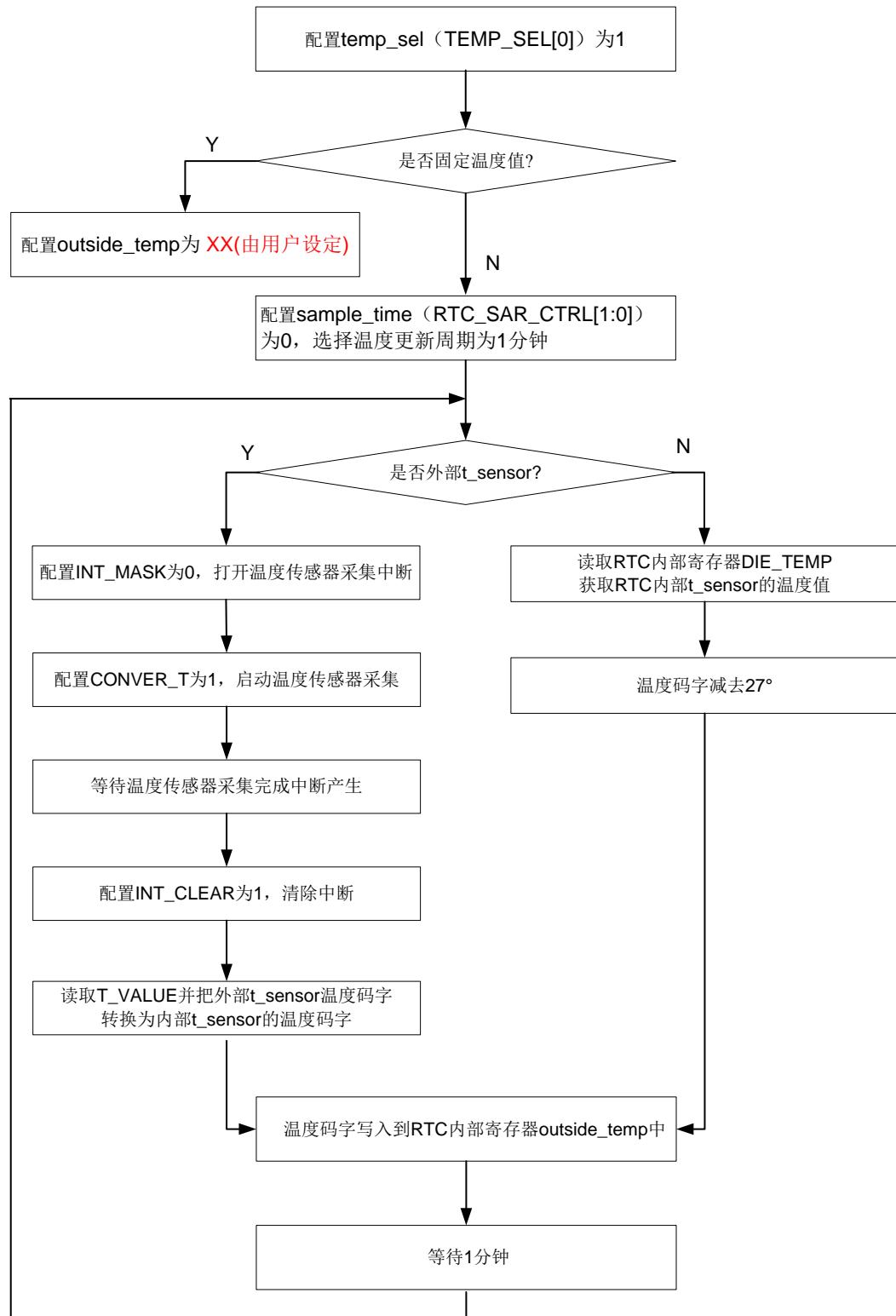
校正温度的来源以下三种：

- 固定温度值。  
适用于芯片工作后，晶体温度变化不明显的情况。
- 采用 RTC 内部  $T_{sensor}$  的测量值。  
温度值需要减去一个经验偏差值（由于芯片工作造成的芯片内部温度与环境温度的偏差值）。因为内部  $T_{sensor}$  获取到的温度值与板上晶体温度的关系不确定，所以只能减去一个经验值(27°C)，这样做就会给校正的精度带入误差（在高低温的情况下，每偏差 0.7 度会造成 5ppm 左右的误差）。
- 采用外部  $T_{sensor}$  的测量值。  
外部  $T_{sensor}$  贴近晶体表面，能够精确反应晶体的温度，RTC 校正的精度达 5ppm。

在配置好晶体振荡曲线相关的寄存器后，只需定时给 RTC 的温度寄存器刷新温度就可以实现校正。温度更新配置的流程如所图 3-1 示。



图3-1 温度更新配置流程图





# 4 RTC 驱动使用说明

## 4.1 准备

通过 [3.1 测量晶体温度频率曲线与配置 RTC 温度频率曲线](#) 的描述，在《RTC 晶体校正参数生成表》中得到晶体温度频率曲线，点击 [Create rtc\\_temp\\_lut\\_tbl.h](#) 生成 rtc\_temp\_lut\_tbl.h 头文件，用此文件更新 RTC 目录中的同名文件。

## 4.2 编译

在 RTC 目录下执行下述命令即可生成对应的驱动 hirtc.ko 及示例程序 test。

```
cd rtc
make
make test
```

## 4.3 使用

将 hirtc.ko 拷贝到单板，并执行如下命令插入驱动模块：

```
insmod hirtc.ko t_second=T
```

其中 t\_second 模块参数表示温度采集的时间间隔，以秒为单位，驱动默认为 5s。如不必修改，则无需传入模块参数。

RTC 驱动提供的功能通过单板上运行的 test 示例程序说明，如[图 4-1](#) 所示。



图4-1 示例程序用法

```
# ./test

Usage: ./test [options] [parameter1] ...
Options:
  -s(set)      Set time/alarm,      e.g '-s time 2012/7/15/13/37/59'
  -g(get)       Get time/alarm,     e.g '-g alarm'
  -w(write)    Write RTC register, e.g '-w <reg> <val>'
  -r(read)     Read RTC register,  e.g '-r <reg>'
  -a(alarm)    Alarm ON/OFF'
  -reset       RTC reset
  -m(mode)     Mode of temperature gather, e.g '-m <mode> <temp>, mode[0-2]'
```

## 设置获取时间

通过如下命令可设置 RTC 时间：

```
./test -s time <year/month/day/hour/minute/second>
```

通过如下命令可获取 RTC 时间：

```
./test -g time
```

## 设置获取闹钟时间

通过如下命令可设置 RTC 闹钟时间：

```
./test -s alarm <year/month/day/hour/minute/second>
```

通过如下命令可获取 RTC 闹钟时间：

```
./test -g alarm
```

通过如下命令设置闹钟到期是否产生中断，驱动中断例程由用户根据需求自由补充。

```
./test -a ON/OFF
```

## 读取、设置 RTC 内部寄存器

通过如下命令可读取 RTC 内部寄存器，此功能多用于辅助调试，比如读取内部温度传感器采集的温度值，读取设置的 RTC 更新温度值等。

```
./test -r <reg>
```

通过如下命令可设置 RTC 内部寄存器，此功能多用于辅助调试。

```
./test -w <reg> <value>
```

reg 取值，请参见《Hi3518 HD IP Camera SoC 用户指南.pdf》3.9 节实时时钟部分。

## 复位 RTC 模块

通过如下命令可复位 RTC 模块。



```
./test -reset
```

## 设置温度采集方式

通过如下命令可设置温度采集方式：

```
./test -m <mode> <value>
```

Mode 和 value 取值如表 4-1 所示。

表4-1 Mode 和 value 取值表

温度采集方式	Mode	Value
固定模式，手动更新	0	晶体环境温度值。
外部 sensor 采集，自动更新	1	NA
内部 sensor 采集，自动更新	2	NA

## 用户接口

请参看 hi\_RTC.h 文件。



# 5 Q&A

## 5.1 振荡器不振

### 【现象】

32.768K 时钟无输出， RTC 计时电路中的秒寄存器值恒定不变。

### 【分析】

使用示波器探头观察 RTC\_XIN 管脚振荡波形，引起不同的振荡波形的情况有以下几种：

- 如果无振荡波形，可能是晶体损坏。
- 如果有 32K 左右频率正弦波，且 peak to peak 幅度小于 600mV，则可能是因为 CL1 与 CL2 过大，导致振荡电路驱动不足，从而幅度偏小，振荡波形无法通过后续的施密特触发器。
- 如果有 200K 左右频率的正弦波，且 peak to peak 幅度小于 600mV，则可能是因为 CL1 与 CL2 偏小，导致振荡电路振荡到 200K 频点，且由于 200K 频点振幅较小，所以无法通过后续的施密特触发器。

### 【解决】

- 如果确定晶体损坏，请更换晶体。
- 如果为 32K 左右频率正弦波，幅度不够，请检查 CL1 与 CL2 是否偏大，并更换正确的电容。
- 如果为 200K 左右频率正弦波，幅度不够，请检查 CL1 与 CL2 是否偏小，并更换正确的电容。

## 5.2 振荡器的输出频率是 200K

### 【现象】

32.768K 时钟输出频率接近 200K， RTC 计时电路中的秒寄存器值每秒钟增加 6。

### 【分析】

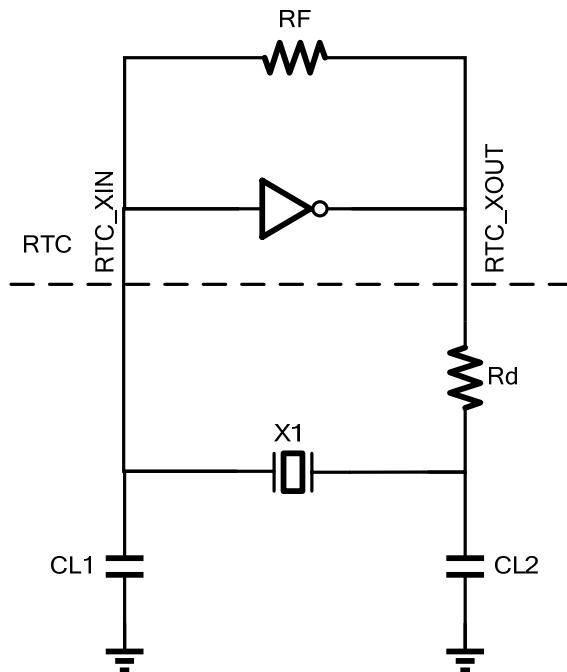


因为 32.768K 晶体存在 6.1 倍频的谐振点，如果晶体有异常，则可能振荡到 6 倍频附近。

### 【解决】

建议首先检查 CL1 与 CL2 是否偏小；如果 CL1 于 CL2 为正确值，但振荡频率仍然为 200K，则可以在电路中添加如图 5-1 所示的 Rd，Rd 取值为  $1/(2\pi \times 32768 \times CL2)$ ，Rd 与 CL2 可以形成一个 RC 滤波器，降低 6.1 倍频处的环路增益。

图5-1 200K 振荡解决方案电路



注意：一般情况不建议增加 Rd，如果采用增加 Rd 的方法，需要确定 RTC\_XOUT 管脚信号幅度不会过小。

## 5.3 振荡频率虽然是 32.768K 附近，但是频率却不准？

### 【现象】

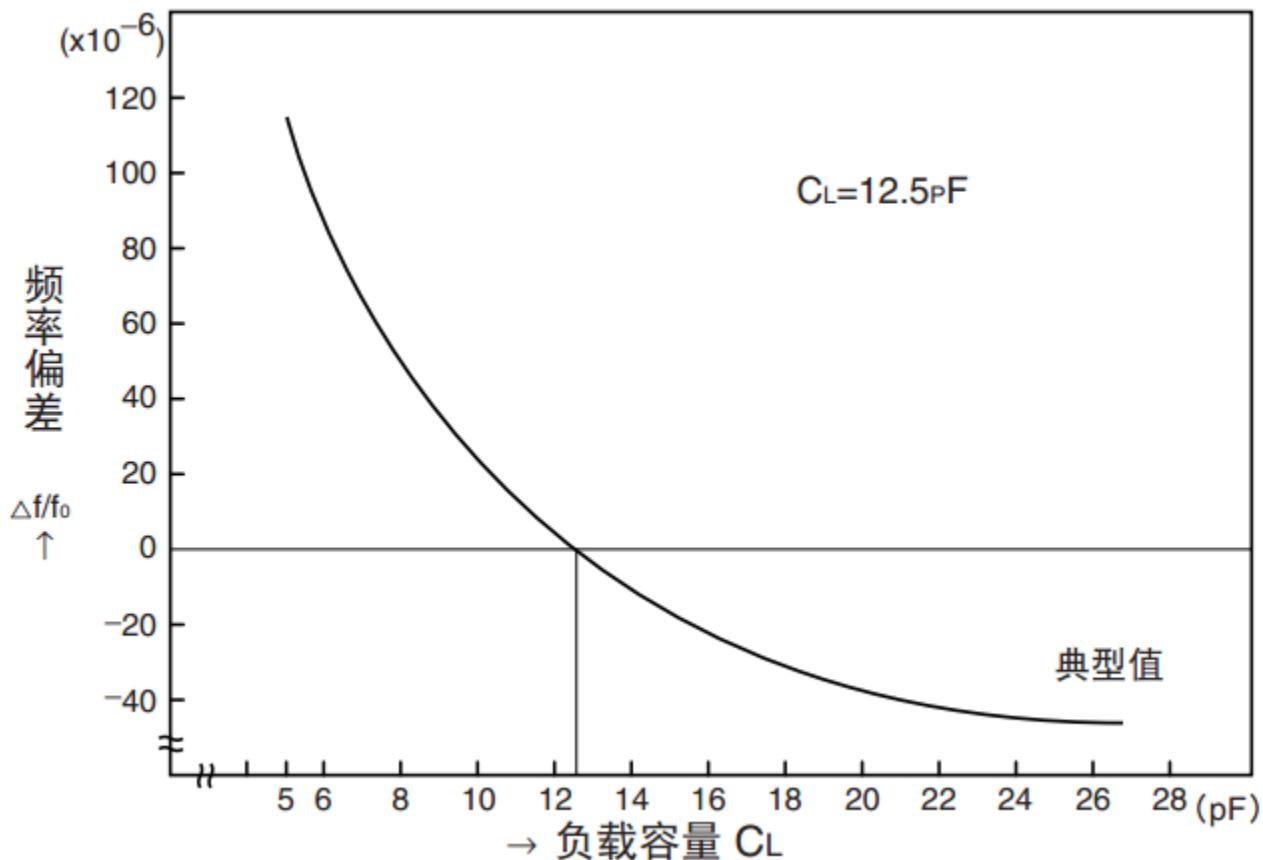
32.768K 时钟输出频率偏离 32.768K。

### 【分析】

振荡电路振荡频率主要由晶体和负载电容共同保证，晶体本身确定了频率的大致范围（即图 5-2 所示中的 0 偏差对应的频率），而实际负载电容的大小则确定了频率的偏移量（即图 5-2 所示中的实际频率偏离 0 的值）。



图5-2 频率偏差和负载容量  $C_L$  的关系



### 【解决】

如果频率偏离了 32.768K，首先需要确认晶体管脚弯折不会对内部晶体部分施加应力，并确定焊接过程中的温度符合 datasheet 规范；其次，检查  $CL1$  与  $CL2$  取值是否正确，并更换正确的电容。