RTC Correction Scheme

Application Guide

Issue	**03**
Date	**2015-05-29**

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between HiSilicon and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

HiSilicon Technologies Co., Ltd.

Address:	Huawei Industrial Base Bantian, Longgang Shenzhen 518129 People's Republic of China
Website:	http://www.hisilicon.com
Email:	support@hisilicon.com

About This Document

Purpose

This document describes the RTC correction scheme, ensuring that the RTC counts correctly.

NOTE

This document uses the Hi3516A as an example. Unless otherwise stated, Hi3516D, Hi3518E V200/V201, Hi3516C V200, and Hi3516A contents are consistent.

Unless otherwise stated, the contents of Hi3516C V200 are consistent with those of Hi3518E V200/Hi3518E V201.

Related Versions

The following table lists the product version related to this document.

Product Name	Version
Hi3516A	V100
Hi3516D	V100
Hi3518E	V200
Hi3518E	V201
Hi3516C	V200

Intended Audience

This document is intended for technical support personnel.

Change History

Changes between document issues are cumulative. Therefore, the latest document issue contains all changes made in previous issues.

Issue 03 (2015-05-29)

The contents related to the Hi3518EV200, Hi3518EV201 and Hi3516CV200 are added.

Issue 02 (2015-02-09)

The descriptions in sections 1.2 and 2.2 are updated.

Issue 01 (2014-12-20)

The contents related to the Hi3516D are added.

Issue 00B01 (2014-09-14)

This issue is the first draft release.

Contents

Figures

Tables

1 Overview

1.1 RTC Classification

Typically, real-time clocks (RTCs) are classified into three types:

- Non-integrated RTC: This RTC has only the RTC timing circuit but no integrated crystal oscillator and temperature compensation circuit. The timing accuracy of this RTC depends on the accuracy of the external crystal oscillator and is susceptible to temperature change. Typically, the timing accuracy is high at ambient temperature, and the timing deviation gradually increases when the temperature increases or decreases.
- RTC with integrated crystal oscillator: This RTC has integrated RTC timing circuit and crystal oscillator but no temperature compensation circuit. The timing accuracy of this RTC reaches the highest at ambient temperature but is also affected by temperature change.
- Integrated RTC: This RTC has integrated RTC timing circuit, crystal oscillator, and temperature compensation circuit (including the temperature sensor). It needs to be calibrated before delivery. As this RTC has the temperature compensation circuit, it features high timing accuracy and is slightly affected by temperature change.

1.2 RTC Working Modes

The embedded RTC of the Hi3516A/HI3518E V200 supports two working modes: fixed frequency-division mode and temperature compensation mode.

- Fixed Frequency-Division Mode

  Similar to the non-integrated RTC, the embedded RTC of the Hi3516A/HI3518E V200 uses the frequency-division clock generated by the external crystal oscillator and oscillation circuit. The frequency divider is fixed during working. In this mode, the RTC timing accuracy depends on the frequency accuracy of the external crystal oscillator and is affected by ambient temperature change. You can replace the non-integrated RTC with the embedded RTC of the Hi3516A/HI3518E V200 to reduce costs.
- Temperature Compensation Mode

  Because the Hi3516A/HI3518E V200 has embedded RTC timing circuit and temperature compensation circuit (including the temperature sensor), the timing deviation caused by temperature change can be corrected. However, the temperature sensor is embedded in the Hi3516A/HI3518E V200. The sensor cannot reflect the actual temperature of the external crystal oscillator. As a result, the corrected timing accuracy is not satisfied.

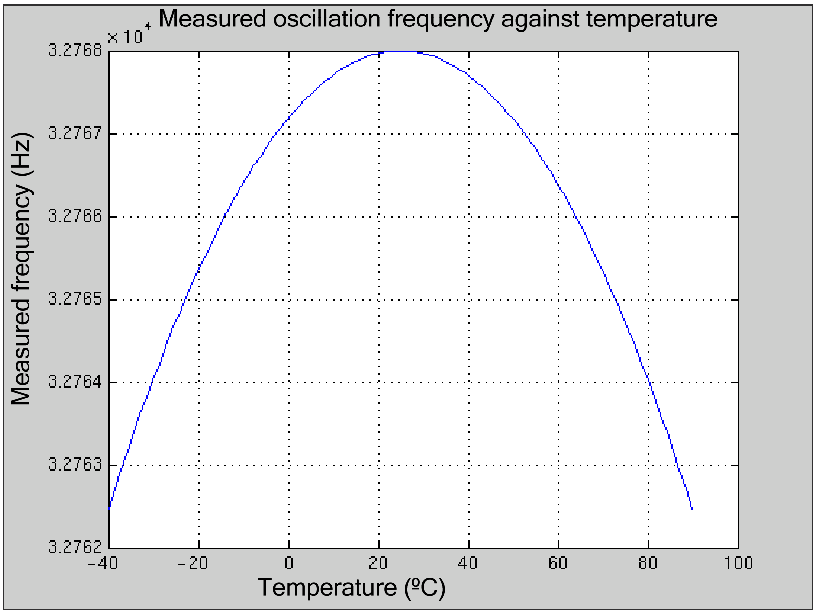
If high timing accuracy is required, you are advised to select the external RTC with an embedded crystal oscillator or the RTC with the temperature compensation function.

1.3 Relationship Between the Crystal Oscillator Frequency and the Temperature

The RTC timing frequency is calculated as follows: RTC timing frequency = Frequency of the clock generated by the internal oscillation circuits/Frequency divider (327.xx).

The frequency of the clock generated by the internal oscillation circuit varies according to the temperature. In this case, the frequency divider can be adjusted to ensure a constant RTC timing frequency. Figure 1-1 shows the relationship between the oscillation frequency output by the 32 kHz crystal oscillator and the temperature.

Figure 1-1 Relationship between the oscillation frequency output by the 32 kHz crystal oscillator and the temperature



The curve in Figure 1-1 is expressed by the following formula: $F = [K_s \times (T - T_0)^2 + 1 + C] \times 32768$

where

- F indicates the crystal oscillation frequency (in Hz) at the temperature T.
- K_s indicates the quadratic coefficient of the curve and it is related to the selected crystal oscillator. K_s in Figure 1-1 is calculated as follows: -4×10^{-8}ºC^{-2}.
- T_0 indicates the transition temperature and the value range is 25±5ºC. Figure 1-1 The T_0 value in Figure 1-1 is 24.94ºC (76.89ºF). A temperature code in the RTC indicates 0.705882ºC (33.27ºF). For details about the relationship between the temperature code and the temperature, see the *RTC Crystal Correction Parameter Creation Table*.
- C indicates the frequency offset at the transition temperature and its value is **0** in the preceding figure. C is affected by the following two factors:

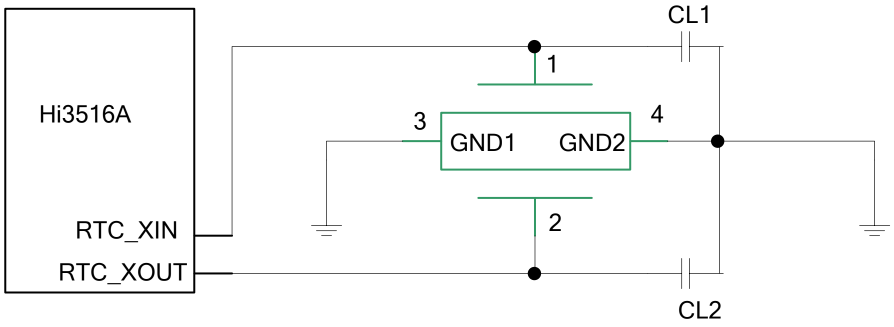
- Load capacitance (CL)
- Crystal difference

2 Hardware Reference Circuit of the Crystal

2.1 Hardware Reference Circuit

Crystals and capacitors need to be selected for the hardware reference circuit, as shown in Figure 2-1.

Figure 2-1 Hardware reference circuit of the crystal



2.2 Selecting Crystal Oscillators

The following specifications must be taken into account when you select crystal oscillators:

- Standard CL: Crystal oscillators impose strict requirements on the load capacitance. The crystal oscillator frequency reaches the nominal value only when the actual load capacitance is the same as the load capacitance defined in crystal oscillator specifications.

  The Hi3516A/HI3518E V200 crystal oscillation circuit is designed based on the 12.5 pF crystal oscillator. You are advised to use the 12.5 pF crystal oscillator because this crystal oscillator is the mainstream in 32.768 kHz crystal oscillators. If you want to use other crystal oscillators, you need to select matched capacitors based on the factors that affect the RTC accuracy.

- Series resistance: crystal resonance equivalent series resistance (ESR). Greater ESR indicates that the crystal is more difficult to drive. The typical and maximum values of R_s are specified in crystal specifications.

The crystal oscillator whose maximum series resistance is less than 70 kilohms is recommended for the Hi3516A/HI3518E V200 crystal oscillation circuit. This ensures that the voltage amplitude of RTC_XOUT is greater than or equal to 850 mV.

- Maximum drive level: maximum crystal oscillation amplitude. If the oscillation amplitude exceeds a specified amplitude, the crystal oscillator is easy to be damaged.

  The oscillation amplitude of the pins RTC_XIN and RTC_XOUT are specified on the Hi3516A/HI3518E V200 crystal oscillation circuit. The actual drive level (less than the maximum drive level in crystal oscillator specifications) when the circuit works is calculated as follows:

  $DL\_actual = 0.5 \times Rs\_max \times (\pi \times f \times V_{ppXIN} \times CL \times 2)^2$

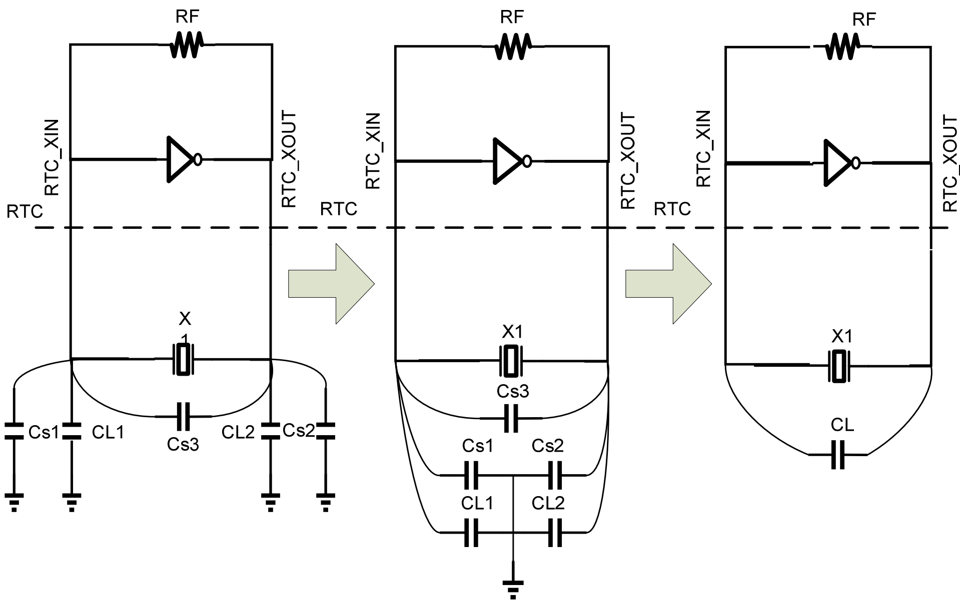
  where

  – R_s_max indicates the maximum series resistance in crystal oscillator specifications.
  – f indicates the crystal oscillator resonance frequency.
  – V_{ppXIN} indicates the peak-to-peak voltage of the oscillation waveform for the RTC_XIN pin measured by using the oscilloscope.
  – CL is the load capacitance in crystal oscillator specifications.

2.3 Selecting Capacitors

Figure 2-2 shows the actual CL diagram.

Figure 2-2 Actual CL diagram



Typically, the values of CL1 and CL2 are the same on the Pierce oscillator. The values are calculated as follows:

CL1 = CL2 = CL_SPEC x 2 – Stray capacitance

CL_SPEC is the load capacitance in crystal oscillator specifications. The stray capacitance might be caused by the PCB and its value range is 3–5 pF. For example, if the load capacitance of a crystal oscillator is 12.5 pF, the values of CL1 and CL2 are calculated as follows: 12.5 pF x 2 – 3 pF = 22 pF. The stray capacitance varies according to the PCB. Therefore, you can obtain the output frequency that is approximate to 32.768 kHz by selecting an appropriate CL1 after the PCB is determined.

3 Implementation of the Fixed Frequency-Division Mode

There is no temperature compensation for the RTC in fixed frequency-division mode. The RTC clock is the one that is generated by the external crystal oscillator and oscillation circuit and then is divided by 327.xx. The RTC timing accuracy depends on the accuracy of the clock provided by the external crystal oscillator. The decimal frequency divider can be adjusted. The timing accuracy in fixed frequency-division mode is close to that of the non-integrated RTC.

In fixed frequency-division mode, the internal RTC registers whose offset addresses are 0x21, 0x51, and 0x52 need to be configured:

- The value of the register whose offset register is 0x21 needs to be set to **0x6** to set the RTC clock to the external square wave frequency-division clock and disable temperature compensation.
- The registers whose offset addresses are 0x51 and 0x52 are combined to form a 16-bit register. Their value determines the decimal frequency divider which is calculated as follows:

  Frequency divider = 327 + (Value read from the registers whose offset addresses are 0x51 and 0x52/3052)

  For example, the value read from the register whose offset address is 051 is **0x8**, the value read from the register whose offset address is 0x52 is **0x1b**, and the value of the combined 16-bit register is **0x81b** (**2075** in decimal). Therefore, the frequency divider is calculated as follows: Frequency divider = 327 + (2075/3052) = 327 + 0.68 = 327.68

The decimal frequency divider can be fine-tuned to ensure that the frequency-division clock is close to 100 Hz. In this way, the RTC timing accuracy is increased. You are advised to tune the frequency divider when all clocks are faster or slower than the expected values. For example, if the output frequency of the crystal oscillator is 32767.00 Hz and the default frequency divider 327.68 is used, the frequency-division clock is 99.97 Hz, which is slower than the expected value. If the frequency divider is 327.67, the frequency-division clock is 100 Hz, and timing deviation is decreased.

4 RTC Correction

The RTC correction frequency ranges from 32760 Hz to 32776 Hz (at chip operating temperatures). To implement RTC correction, perform the following steps:

Step 1 Obtain the temperature-frequency curve of the crystal oscillator.

Typically, the nominal quadratic coefficient $\mathbf{K_s}$ and turning point temperature $\mathbf{T_0}$ of the parabolic are provided in the crystal parameter manuals from crystal oscillator vendors. If you only need to measure the output frequency of the crystal oscillator at the ambient temperature and record the ambient temperature of the crystal oscillator, use the single-point test method. If you are not satisfied with nominal parameters, do as follows to obtain a more accurate curve:

1. Obtain the test tools including the temperature box, frequency meter, and external temperature sensor.
2. Obtain a board, power it on to start the Linux system, enable the VI_ADC_CLK pin to be multiplexed as RTC test clock output (0x200F_0000 = 03b'110), and configure the RTC_CLK register to output the oscillation clock of the crystal oscillator (02b'00).
3. Use the multimetering method to take measurement and record data in the *RTC Crystal Correction Creation Table.*

   If you use the parameters provided by crystal oscillator vendors, calculate multiple groups of temperature-frequency data, and record data in the *Crystal Correction Creation Table.*
4. Click **Create rtc_temp_lut_tbl.h** in the *Crystal Correction Creation Table*. The **rtc_temp_lut_tbl.h** file is automatically generated.

Step 2 Configure the RTC based on the temperature-frequency curve.

1. Replace the **rtc_temp_lut_tbl.h** file in the RTC source code directory with the generated file, and recompile the RTC driver.
2. Load the new driver **hirtc.ko**.

   After the driver is successfully loaded, the RTC successfully starts.
3. Use the provided test program to perform operations such as configuring the time.

Step 3 Configure the RTC correction function.

1. During power-on, if the internal T-Sensor is used, the constant temperature mode is recommended. In this case, select an appropriate RTC temperature based on the ambient temperature. For example, when the ambient temperature is 25ºC (77ºF), the crystal oscillator temperature may be 35ºC (95ºF). You can set the RTC temperature to 35ºC (95ºF). If the external T-Sensor is used, refer to the configuration process described in

section 4.1 "Measuring the Temperature-Frequency Curve of the Crystal Oscillator and Configuring the RTC."

2. During power-off, no operation is required. The RTC is automatically switched to the temperature measurement mode of the internal T-Sensor.

----End

4.1 Measuring the Temperature-Frequency Curve of the Crystal Oscillator and Configuring the RTC

Measuring the Temperature-Frequency Curve of the Crystal Oscillator

You can obtain curves of the oscillation frequency offset and temperature by entering the measured temperature and the oscillation frequency output by the crystal oscillator in the *RTC Crystal Correction Creation Table*. You can select the following methods to measure curves of the oscillation frequency offset and temperature by taking the measurement accuracy and cost into account:

- Multimetering

  You can obtain the relationship between temperatures and oscillation frequencies at chip operating temperatures. This scheme applies to the scenario in which crystal parameters need to be reevaluated.

- Three-point test

  Frequencies measured only at –20°C (–4ºF), +25°C (+77ºF), and +80°C (+176ºF) are used. This scheme applies to the scenario in which crystal parameters need to be fast determined.

- Two-point test

  Frequencies measured only at –20°C (–4ºF) and +80°C (+176ºF) are used. This method applies to the scenario in which the crystal oscillator model is known because the quadratic coefficient K^s of the oscillation frequency offset and temperature function is known and there is no obvious change (the function coefficients of various crystal oscillators are almost the same).

- Single-point test

  The frequency measured at the room temperature is used. This method applies to the scenario in which crystal oscillators of the same model and batch are used because the quadratic coefficient K_s of the oscillation frequency offset and temperature function and the transition temperature T_0 are known and there is no obvious change (the function coefficients of various crystal oscillators are almost the same).

Notice

Note the following:

- Test tools: temperature box, frequency meter, and external temperature sensor
- Services during tests: The kernel mode is optimal.
- Test temperature points: The multimetering test, three-point test, two-point test, and single-point test are supported. More test temperature points indicate higher curve accuracy in theory.

Typically, there is 20 ppm individual variation among the crystal oscillators from the same lot. You can measure the temperature-frequency curves of multiple crystal oscillators and select a medium curve. If fitting is appropriate, the error can decrease by 10 ppm theoretically. The multimetering test is used if conditions are allowed. The recommended test temperature points include −20ºC (−4ºF), −10ºC (+14ºF), +15ºC (+59ºF), +25ºC (+77ºF), +35ºC (+95ºF), +60ºC (+140ºF), and +70ºC (+158ºF).

- Test environment: Because RTC curve measurement is susceptible to the temperature and interference, the test environment must meet the following conditions:
  - Attach the external temperature sensor to the crystal oscillator with silicone to ensure accurate test temperature.
  - Never keep the probe and crystal oscillator pin in contact when the frequency is being tested.
  - Use the RTC frequency test multiplexing pin (VI_ADC pin) to test the frequency.
  - Fan out test signals by using copper traces with good anti-interference performance. Typically, the RTC output frequency is fixed at two decimal places. For example, if the RTC output frequency is 32767.97*xx* Hz, *xx* indicates variable decimal places and can be ignored. An unstable frequency indicates that interference occurs. In this case, check the grounding connection and test environment or change the test environment.
- Test and data record: Adjust the temperature of the temperature box, read the temperature of the external temperature sensor (that is, the crystal oscillator temperature) after the temperature of the temperature box is stable for a period of time, and record the RTC output frequency. The obtained temperatures and frequencies are formed a quadratic frequency-temperature curve.
- If conditions are allowed, you are advised to measure the curves of more than four crystal oscillators from the same lot, fit the frequency-temperature curves, and then select the most appropriate one.

Measuring the Temperature-Frequency Curve of the RTC

Coefficients of the oscillation frequency offset and temperature function correspond to registers in the RTC. LUT_n is related to K_s x $(T_n – 24.94)^2$ ([n ∈ [1, 47], Tn = –40 + n x (180/255)], TEMP_OFFSET is related to T_0, and TOT_OFFSET is related to C.

After selecting an appropriate measurement method, enter measured frequencies and temperatures into the desired sheet and click **Create rtc_temp_lut_tbl.h** to create the configuration file. Replace the file in the RTC driver with the generated **rtc_temp_lut_tbl.h** file and recompile the RTC driver. In this way, the coefficients are configured to the RTC each time the chip is booted.

The generated curves of the oscillation frequency offset and the temperature vary according to the crystal oscillator. You can use the following three methods to configure the RTC driver file based on the measurement accuracy and cost:

- Measure the function coefficients of various crystal oscillators and configure the coefficients to the RTC. The maximum RTC counting error is 5 ppm. The function coefficients of each crystal oscillator must be configured by using the configuration file **rtc_temp_lut_tbl.h**.
- Measure a specified number of crystal oscillators of the same batch, recalculate the calculated coefficients, and configure the crystal oscillators based on the recalculated coefficients. The maximum RTC counting error is calculated as follows: 5 ppm + error caused by crystal oscillator differences. The counting error is affected by crystal oscillator production consistency.

- Measure no crystal oscillator, and use the initial chip coefficients or the coefficient configuration file generated by using the data provided by the crystal oscillator manufacturer. This RTC counting error is determined by the crystal oscillator consistency and the match degree of the inherent coefficients of the oscillation frequency and temperature function and the configured coefficients of the oscillation frequency and temperature function in the RTC.

Factors Affecting the RTC Accuracy

The RTC errors include the matched error of capacitors and the crystal oscillator, individual variation of crystal oscillators, temperature offset error, and temperature deviation (if any).

- Matched error of capacitors and the crystal oscillator. According to the crystal oscillator manuals, the CL meets the following condition: C` + (C0 + C1)/2 = CL. C` indicates the board trace capacitance, C0 and C1 are the capacitance of the crystal oscillators in series, and CL indicates the crystal oscillator capacitance. The RTC accuracy is ensured only when the crystal oscillator matches the capacitance. If the capacitance is too large, the output frequency of the crystal oscillator is too small. If the capacitance is too small, the output frequency of the crystal oscillator is too large, and the frequency may be doubled. Therefore, the selected capacitors must match the crystal oscillator. If a correction or temperature compensation solution is used, the error caused by the mismatch of capacitors and the crystal oscillator can be eliminated in theory. After elimination, the error can be ignored.
- Individual variation of crystal oscillators. There is 20 ppm individual variation among the crystal oscillators from the same lot. You are advised to select standard-compliant crystal oscillators to ensure high accuracy. The best method of reducing the individual variation is to measure the temperature-frequency curve for each crystal oscillator and then correct the curve. After correction, the individual variation can be reduced to less than 5 ppm. This method is adopted for high-end RTC chips. An alternative is to select a curve of the crystal oscillators from the same lot or select the fitted curve among multiple curves. This method is convenient but causes errors. If the curves of multiple crystal oscillators are measured and then fitted, the accuracy is higher, and the error is about 10 ppm. If the curve of only one crystal oscillator is measured, the random error is 10 ppm to 20 ppm.
- Temperature offset error. The crystal oscillator temperature varies according to the inside temperature of the product with the cover. The crystal oscillator frequency offset occurs when the crystal oscillator temperature changes. The output frequency of the crystal oscillator is closely related to the crystal oscillator temperature, especially the high temperature and low temperature. The frequency error that occurs when the crystal oscillator temperature changes by 1ºC (33.8ºF) varies according to the temperature range. The frequency offset changes obviously at a high or low temperature but not at transition temperature (T0). If the curves of crystal oscillators are measured and temperature compensation is performed, the offset error can be reduced or even eliminated.
- Temperature deviation. The temperature deviation occurs only when the internal T-Sensor is used for correction. During internal T-Sensor correction, the chip internal temperature is read to deduce the crystal oscillator temperature, and temperature compensation is performed based on the crystal oscillator temperature. If the same temperature deviation is used for all chips, the error is caused for the following reasons:
  - Error caused by the internal T-Sensor. The read value of the internal T-Sensor varies according to chips. The error is about 5ºC (41ºC).
  - Crystal oscillator temperature obtained by subtracting an estimated deviation from the master chip temperature measured by the T-Sensor. In normal scenarios, the master chip temperature and temperature deviation of the crystal oscillator may

significantly fluctuate, which results in a large error. During power-off, the mater chip temperature is the same as the crystal oscillator temperature, and the error is small.

- Crystal oscillator positions and heat dissipation conditions inside the product with the cover.

4.2 RTC Correction When Services Are Running

The correction temperature may be one of the following:

- Fixed temperature

  This method applies to the scenario in which temperature variance is not obvious when the chip runs.

- Temperature obtained by the temperature sensor in the RTC

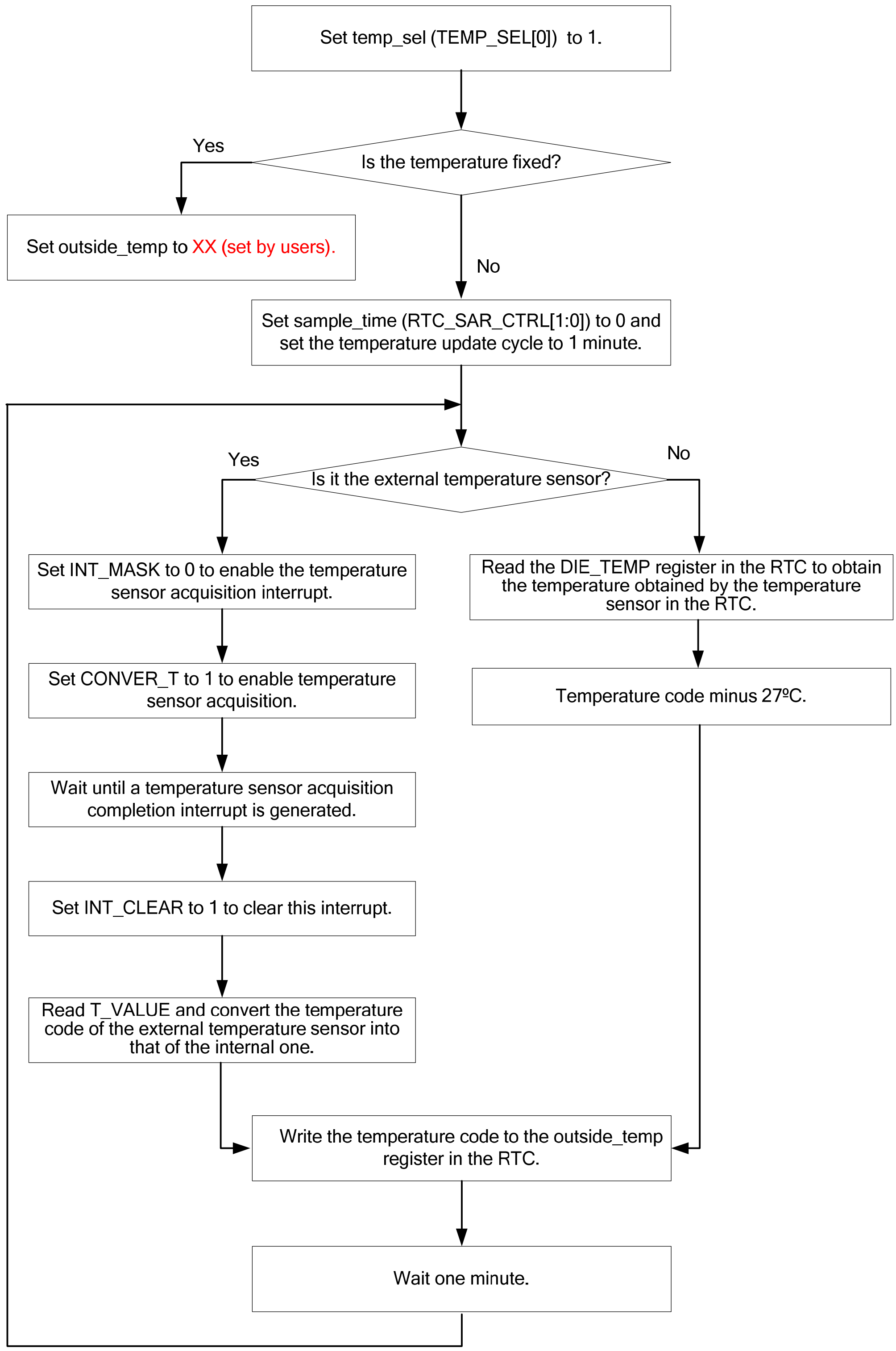
  To be specific, the temperature is obtained by subtracting an empirical deviation (for example, 27ºC or 80.6ºF) from the read value of the T-Sensor. The RTC accuracy is not high in actual scenarios.

- Temperature obtained by the external temperature sensor

  The external T_Sensor is placed close to the crystal oscillator surface to obtain accurate crystal oscillator temperature. If the T-Sensor accuracy is high, the RTC correction accuracy can reach 5 ppm in theory.

You can implement RTC correction by updating temperatures of RTC temperature registers regularly after registers for the crystal oscillation function are configured, as shown in Figure 4-1

Figure 4-1 Configuration process of updating temperatures



4.3 Automatic RTC Correction When the Board Is Powered Off

When the board is powered off, the battery power supply mechanism is automatically switched if the RTC circuit detects a board power-off signal. If there is an external T-Sensor when the battery supplies power, the external T-Sensor does not work, and the temperature is read from the internal register DIE_TEMP of the RTC. In this case, the ambient temperature is considered the same as the chip temperature. In addition, the value of the RTC internal register is retained, and the used correction curve is the same as the one during power-on. When the board is powered on again, the RTC is switched to the working mode that is previously configured during power-on.

5 RTC Driver Usage

5.1 Preparation

Obtain the function coefficients of the oscillation frequency offset and temperature in the *RTC Crystal Correction Creation Table* based on descriptions in section 4.2, click **Create rtc_temp_lut_tbl.h** to generate the **rtc_temp_lut_tbl.h** header file, and replace the file of the same name in the RTC directory with the generated file. In fixed frequency-division mode, use the **rtc_temp_lut_tbl.h** file directly.

5.2 Compilation

Run the following commands in the RTC directory to generate the hirtc.ko driver and sample program test:

```
cd rtc
make
make test
```

5.3 Usage

Copy hirtc.ko on the board and run the following command to insert the driver module:

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

The **t_second** parameter indicates the temperature collection interval in second. The default interval is 5s. Do not input this parameter if the default value remains. In fixed frequency-division mode, run **insmod hi_rtc.ko** to load the **hirtc.ko** driver. No parameter values need to be transferred.

RTC driver functions are described in the test sample program running on the board, as shown in Figure 5-1.

Figure 5-1 Test sample program usage

```
# ./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(ead)          Read RTC register,   e.g '-r <reg>'
        -a(alarm)        Alarm ON/OFF',       e.g '-a ON'
        -reset           RTC reset
        -c(ompensation)  temperature compensation ON/OFF, eg '-c ON'
        -f(requency)     frequency precise adjustment, eg '-f <val>'
        -m(mode)         Mode of temperature gather, e.g '-m <mode> <temp>, mode[0-2]'
```

Configuring and Obtaining the RTC Time

Run the following command to configure the RTC time:

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

Run the following command to obtain the RTC time:

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

Configuring and Obtaining the RTC Alarm Time

Run the following command to configure the RTC alarm time:

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

Run the following command to obtain the RTC alarm time:

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

Run the following command to set whether an interrupt is generated when the alarm time reaches. The driver interrupt routine is added by users.

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

Reading and Configuring Registers in the RTC

Run the following command to read a register in the RTC. This function is used for auxiliary tests, such as reading temperatures collected by the internal T_Sensor and reading configured updated RTC temperatures.

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

Run the following command to configure the register in the RTC. This function is used for auxiliary tests.

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

For details about the reg value, see section 3.10 in the *Hi3518E V20X/Hi3516C V200 Economical HD IP Camera Soc Data Sheet*.pdf.

Resetting the RTC

Run the following command to reset the RTC:

```
./test -reset
```

Switching the Temperature Compensation Mode

Run **./test –c ON/OFF** to enable or disable the temperature compensation mode. **ON** indicates that RTC temperature compensation is enabled, and **OFF** indicates that temperature compensation is disabled and the fixed frequency-division mode is used. If the board is reset or the board is restarted with the battery installed, the system runs in fixed frequency-division mode after the .ko driver is loaded. If the temperature compensation mode is used, you need to enable this mode again after the .ko driver is loaded.

Fine-Tuning the Frequency Divider in Fixed Frequency-Division Mode

Run the following command to set the frequency divider for adjusting the clock forward or backward:

```
./test -f <val>
```

<val> is 10000 times the frequency divider to be set. For example, if the frequency divider is 327.60, **val** is **3276000**. Run **./test –f** to view the current frequency divider. The frequency divider ranges from 327.60 to 327.70.

NOTE

Set the frequency divider only in fixed frequency-division mode. Disable temperature compensation before setting the frequency divider. You can also use the default frequency divider.

Configuring the Temperature Measurement Mode

Run the following command to configure the temperature measurement mode:

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

Table 5-1 describes the temperature measurement modes and their values.

Table 5-1 Temperatures measurement modes and their values

Temperature Measurement Mode	Mode No.	Value
Fixed temperature, manual update	0	Crystal oscillator ambient temperature
Temperature obtained by the external temperature sensor, automatic update	1	N/A
Temperature obtained by the internal temperature sensor, automatic update	2	Empirical deviation

NOTE

Configure the temperature measurement mode only in temperature compensation mode. Enable temperature compensation before configuring the temperature measurement mode.

User Interface

See the **hi_rtc.h** file.

6 Q&A

6.1 No Oscillation on the Oscillator

[Symptom]

The 32.768 kHz clock has no output, and the value of the second register on the RTC timing circuit is constant.

[Analysis]

Use the oscilloscope probe to check oscillation waveforms on the RTC_XIN pin, and various oscillation waveforms are caused in the following situations:

- If there is no oscillation waveform on the pin, the crystal oscillator may be damaged.
- If about 32 kHz sine waves are detected and the peak-to-peak amplitude is less than 600 mV, capacitance of CL1 and CL2 may be large, causing the oscillation circuit drive capability to be insufficient and the peak-to-peak amplitude is smaller than that at 32.768 kHz frequency. Therefore, oscillation waveforms cannot pass the subsequent Schmitt trigger.
- If about 200 kHz sine waves are detected and the peak-to-peak amplitude is less than 600 mV, capacitance of CL1 and CL2 may be small, causing the oscillation circuit to oscillate to 200 kHz frequency at which the amplitude is smaller than that at 32.768 kHz frequency. Therefore, oscillation waveforms cannot pass the subsequent Schmitt trigger.

[Solution]

- Replace the crystal oscillator if it is damaged.
- If about 32 kHz sine waves are detected and the peak-to-peak amplitude is small, check whether capacitance of CL1 and CL2 is large and replace the capacitors with appropriate capacitors.
- If about 200 kHz sine waves are detected and the peak-to-peak amplitude is small, check whether capacitance of CL1 and CL2 is small and replace the capacitors with appropriate capacitors.

6.2 200 kHz Frequency Output by the Oscillator

[Symptom]

The frequency output by the 32.768 kHz clock is approximate to 200 kHz, and the value of the second register on the RTC timing circuit increases by 6 per second.

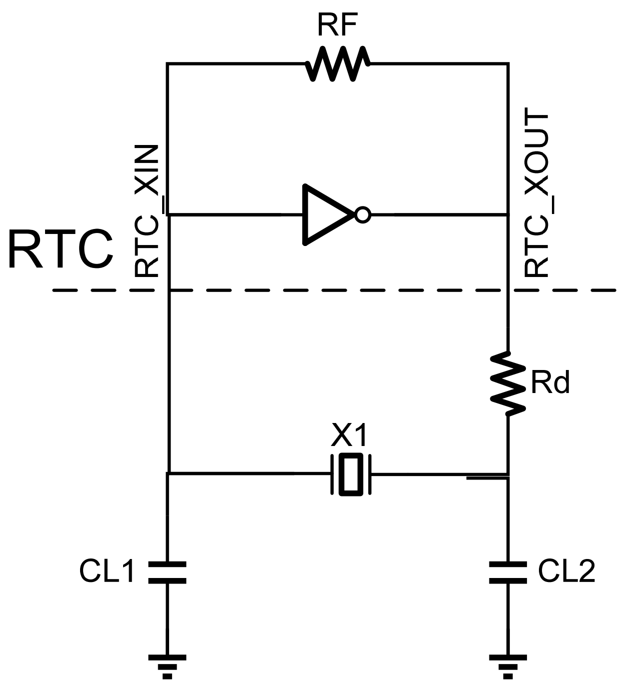
[Analysis]

If exceptions occur on the 32.768 kHz crystal oscillator, the crystal oscillator may oscillate near to six times of the fundamental frequency because the resonance point of 6.1 times of the fundamental frequency is available on this crystal oscillator.

[Solution]

Check whether capacitance of CL1 and CL2 is small. If capacitance of the two capacitors are appropriate and the oscillation frequency is 200 kHz, add an Rd whose value is 1/(2 π x 32768 x CL2) to the circuit, as shown in Figure 6-1. The Rd and CL2 form an RC filter, reducing the loop gain at 6.1 times of the fundamental frequency.

Figure 6-1 Circuit for resolving the 200 kHz output frequency issue



NOTE

You are not advised to add an Rd to the circuit. Ensure that the signal amplitude on the RTC_XOUT pin is not small before adding an Rd to the circuit.

6.3 Incorrect Oscillation Frequency

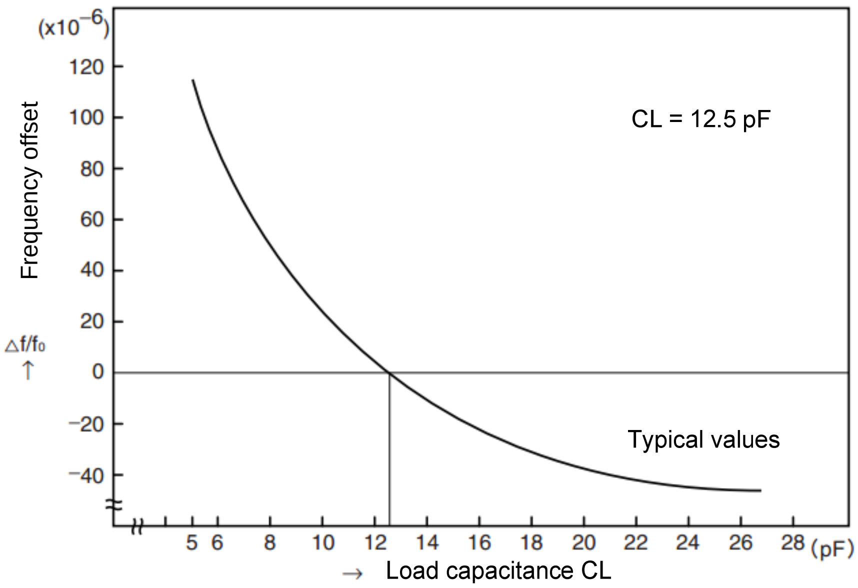
[Symptom]

The frequency output by the 32.768 kHz clock is not 32.768 kHz.

[Analysis]

The crystal oscillator and load capacitance work together to determine the oscillation frequency on the oscillation circuit. The crystal oscillator determines the frequency range (frequencies corresponding to the frequency offset 0 in Figure 6-2, and the actual load capacitance determines the frequency offset (the value of the frequency offset 0 in Figure 6-2).

Figure 6-2 Relationship between the frequency offset and the actual load capacitance CL



[Solution]

Firstly, ensure that crystal oscillator pin bending does not apply stress to the internal crystal oscillator and the soldering temperature complies with specifications in the data sheet. Secondly, check whether capacitance of CL1 and CL2 are appropriate and replace the two capacitors with appropriate ones.