Freescale Semiconductor
Application Note

Document Number: AN4949
Rev 0, 08/2014

Kinetis-M RTC Crystal Clock Frequency Compensation

by: Ashish Sharma

Contents

1 Introduction....1
2 RTC key features 1
3 How to compensate for crystal clock frequency error 2
3.1 Coarse compensation....3
3.2 Fine compensation 4
3.3 PPM error due to temperature and its correction.... 4
3.4 Temperature measurement through SAR ADC.... 5
3.5 PPM error computation from temperature.... 5
4 Example....6
5 Revision history.... 9

1 Introduction

Freescale's Kinetis M series microcontrollers, based on the ARM® Cortex®-M0+ core, have an RTC peripheral that runs on an external 32 kHz crystal. This clock source has some errors. Firstly, every crystal has some offset error, so at room temperature, its clock is not exactly 32.768 kHz. Also, crystals deviate with temperature. Hence, a correction mechanism that encompasses offset error and ambient temperature is required.

The Kinetis M controller has a mechanism for this compensation that is supported by the RTC peripheral. This document details the problem and how to compensate for the crystal errors.

2 RTC key features

- Completely powered by battery
- Does not have a power switch
  - RTC reset occurs only when battery is removed and inserted again
  - RTC is not affected by MCU reset
- Separate counters for days, hours, minutes, and seconds
- Calendaring support – Separate counters for month, year, and day of the week

    - Automatic adjustment for daylight saving time with user-defined parameters
    - Automatic month and leap year adjustment
- Clock compensation to correct frequency drift in 32.768 kHz crystal
    - Coarse Compensation – corrects error up to 0.119 PPM and maintains accurate time (less than 5 PPM accuracy)
    - Fine Compensation – generates accurate (5 PPM accuracy) 1 Hz clock output with 0.88 PPM resolution
    - Hardware compensation runs in all modes — no need for CPU to be active all the time
- Programmable alarm with interrupt
    - Alarm is output from MCU via PXBAR
- Register write protection mechanism to protect against runaway code
- 32 bytes of general purpose RAM
- About 1.3 µA consumption from battery
- Tamper
    - Enhanced Tamper Detection
    - Detects illegal access into the system
    - Active and passive tamper detection
    - Tamper event queuing with time stamp
    - Stores up to four tamper event information items in a queue
    - Tamper status not updated for same tamper event until acknowledged by CPU
    - Tamper sources
    - Up to 3 external tamper detect pins
    - One internal tamper detect event for battery removal
    - Selectable polarity (active high or low)
    - Tamper interrupts can be individually enabled or disabled
    - Status is stored irrespective of interrupt enable setting
    - External tamper sources are filtered for noise and glitches by the RTC
    - Filter width configurable by selecting clock source and filter duration
    - Internal tamper event not filtered
- Alarm
    - Periodic interrupts
    - 1 day, 1 hour, 1 minute, 512 Hz to 2 Hz and 1 Hz interrupts
- Async interrupt can wake MCU from low-power modes

3 How to compensate for crystal clock frequency error

Compensating for the crystal clock frequency error requires the execution of a few basic steps. First, it is important to keep in mind how variations in temperature affect the frequency characteristics of a crystal as shown in Figure 1.

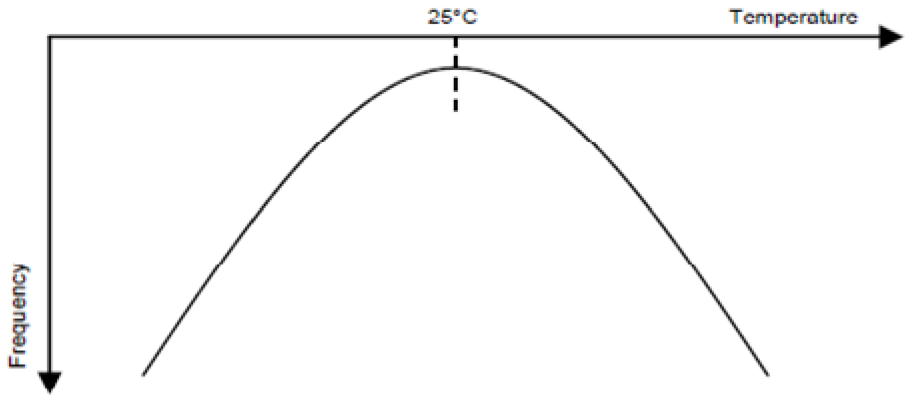

Figure 1. Typical temperature vs. frequency characteristics of a crystal

The basic steps in crystal error correction are:

1. Read ADC channel to get ambient temperature.
2. From formula (explained later in this document), get the PPM error for this temperature via interpolation.
3. Add offset PPM error.
4. Get the correction factors.
5. Update RTC registers.

Kinetis-M RTC has two compensation modes:

1. Coarse
2. Fine

In coarse compensation, one can compensate only in steps of one RTC clock step. Remember, RTC clock is driven at 32.768 kHz. One can, of course, apply the correction spread across multiple second counts – for example adding 18 clocks across five seconds, resulting in 3.6 clock corrections per second. This allows fractions of clocks to be compensated. However, each second is not spread out evenly; the correction is spaced out so some seconds will be "longer" than others.

In fine compensation, the compensation occurs in terms of 4 MHz IRC counts. Since 4 MHz is approximately 32 kHz times 128, this gives an extra 1/128th of resolution to be applied.

Example:

If actual 32k clock = 32769.146 Hz (35 PPM error)

Ideal 32k clock = 32768 Hz

Compensation parameters:

Error to be compensated = (35/1000000) × 32768 = 1.14688

3.1 Coarse compensation

The error approximates to 7 counts to be adjusted in 6 seconds. This is computed by taking the fractional part of the error (0.14688) and finding a multiple such that it is close to a whole number. The easiest way is to try 1/0.14688, which is 6, plus the whole error each time.

In RTC_COMPEN register:

Set:

RTC_COMPEN[7:0] to –7 or 0xF9 (signed 2's complement)

RTC_COMPEN[15:8] to 6

3.2 Fine compensation

Multiply the fractional part by 128, will give 18.8 or 19 when rounded up.

In RTC_COMPEN register:

Set:

RTC_COMPEN[15:12] to 1 or 0x0F (signed 2's complement)

RTC_COMPEN[11:7] to 0 (reserved)

RTC_COMPEN[6:0] to 19 (always positive when fine compensation is enabled)

A second by second correction is shown in Table 1. The fine one-second output will be generated when the 4 MHz clock counts to Accumulator counts past each second. The accumulator rolls over at 128.

Table 1. Accumulator values

Accumulator Value(n)	19	38	57	76	95	114	5	24	43	62
Coarse Adder	0	0	0	0	0	0	1	0	0	0
Coarse Compensation	1	1	1	1	1	1	1	1	1	1

For fine compensation, the source of correction is the IRC clock running at 4194304 Hz. This itself is prone to errors such as:

- The IRC is not a guaranteed source of clock and is expected to have a tolerance of about 10%.
- Any change in ambient temperature will affect the clock.

But there is a self-correction algorithm in the RTC implemented to take care of IRC deviation. Say, for example, the clock becomes 5 MHz. The compare value is adjusted by the clock ratio.

Please note that when the clock is different from 4.194 MHz, the accumulator will accumulate faster as well (because it accumulates once every 4.194 M clock cycles). At 5 MHz, the accumulator will roll over at:

$128 \times 5000000/4194304 = 152$. In this case, the fine 1 Hz clock will be generated at:

Table 2. Coarse compensation values

Coarse Compensation	23	45	67	90	112	135	7	29	52	74

The compensation for deviation of the IRC clock is done by the RTC peripheral itself and the user is not required to do anything for this purpose.

3.3 PPM error due to temperature and its correction

Here is a look at how the 1 Hz clock and fine compensation clock are generated, and how to compensate for crystal frequency deviation due to temperature.

Measure the temperature once every few minutes (say 15) and recalculate the error due to temperature. This is quite reasonable, since the temperature is not expected to change very frequently.

3.4 Temperature measurement through SAR ADC

The SAR ADC module includes a temperature sensor whose output is connected to one of the ADC analog channel inputs. The following equation provides an approximate transfer function for the temperature sensor.

$$Temp = 25 - ((V_{TEMP} - V_{TEMP25}) \div m)$$

Equation 1.

where:

- V_{TEMP} is the voltage of the temperature sensor channel at the ambient temperature.
- V_{TEMP25} is the voltage of the temperature sensor channel at 25 °C.
- *m* is the temperature sensor slope in the device data sheet. It is the hot or cold voltage versus temperature slope in V/°C.

Now, from the data sheet of the crystal used and from actual measurements done on a Kinetis-M device:

- m (Temperature Sensor Slope) = 1.070 mV / C
- V_{TEMP25} = 719 mV (measured)

Using this equation and the value substituted, we measure the temperature.

3.5 PPM error computation from temperature

A typical data sheet of a crystal looks like this:

Table 3. Typical operating values

Item		Symbol	Specifications
Nominal frequency		f	32.768 kHz
Temperature range	Storage temperature	T_{STG}	–55 °C to 125 °C
	Operating temperature	T_{OPR}	–40 °C to 85 °C
Maximum drive level		GL	0.5 μW Max.
Recommended drive level		DL	0.1 μW
Frequency tolerance (standard)		Δf/f	$\pm 20 \times 10^{-6}$
Peak temperature (frequency)		θT	25 °C ±5 °C
Temperature coefficient (frequency)		A	$-0.04 \times 10^{-6}/°C^2$ Max.
Load capacitance		C_L	9.0 pF, 12.5 pF
Series resistance		R_1	70 kΩ Max.
Motional capacitance		C_1	1.7 fF Typ.
Shunt capacitance		C_0	1.3 pF Typ.
Insulation resistance		IR	500 MΩ Min.
Aging		fa	$\pm 3 \times 10^{-6}$/year Max.
Shock resistance		S.R.	$\pm 8 \times 10^{-6}$ Max.

The value of interest here is temperature coefficient (frequency):

Temperature coefficient (frequency)	A	–0.04 × 10^{-6}/°C^2 Max.

This is substituted in the equation:

$$f = f_0[1 - 0.04ppm(T - T_0)^2]$$

Equation 2.

Where:
- f_0 is frequency of the crystal, 32768 Hz
- T is measured ambient temperature
- T_0 is 25 °C

The final frequency f compared with f0 gives the PPM error of the crystal at temperature (T). This has to be added to the offset PPM error to get the total PPM error, which is used to compensate the crystal error.

NOTE

The crystal error correction algorithm is much dependent on the type of crystal selected. To be able to compensate for crystal error, the user must:

- Compensate for the offset error for each crystal, since each crystal has a different offset error.
- Compensate for one type of crystal for temperature effects on accuracy of its clock. Each make and type has one characteristic error.

RTC Clockout signal must be brought out at a suitable place so that it can be easily used by a frequency counter type of device.

4 Example

An experiment was conducted on a Kinetis-M reference board to measure frequency. Figure 2 shows that the crystal has 147 PPM error. This indicates that the frequency is 1.000147636 Hz instead of 1.000000000 Hz. It has an error of 0.000147 or 147 PPM. Use this PPM error in code to set the PPM offset error.

If the PPM error is negative, then the period shown on the frequency counter will be less than 1.000000. For example, if the value shown is 0.999853212, then the PPM error is –147 and has to be compensated accordingly.

Figure 2. Output with uncorrected crystal error

After computing and programming integral and fractional values for RTC correction and enabling fine compensation, the frequency output measured is shown in Figure 3 . The error is less than 1 PPM. This error stays within 5 PPM for the temperature range –20 to +70 °C.

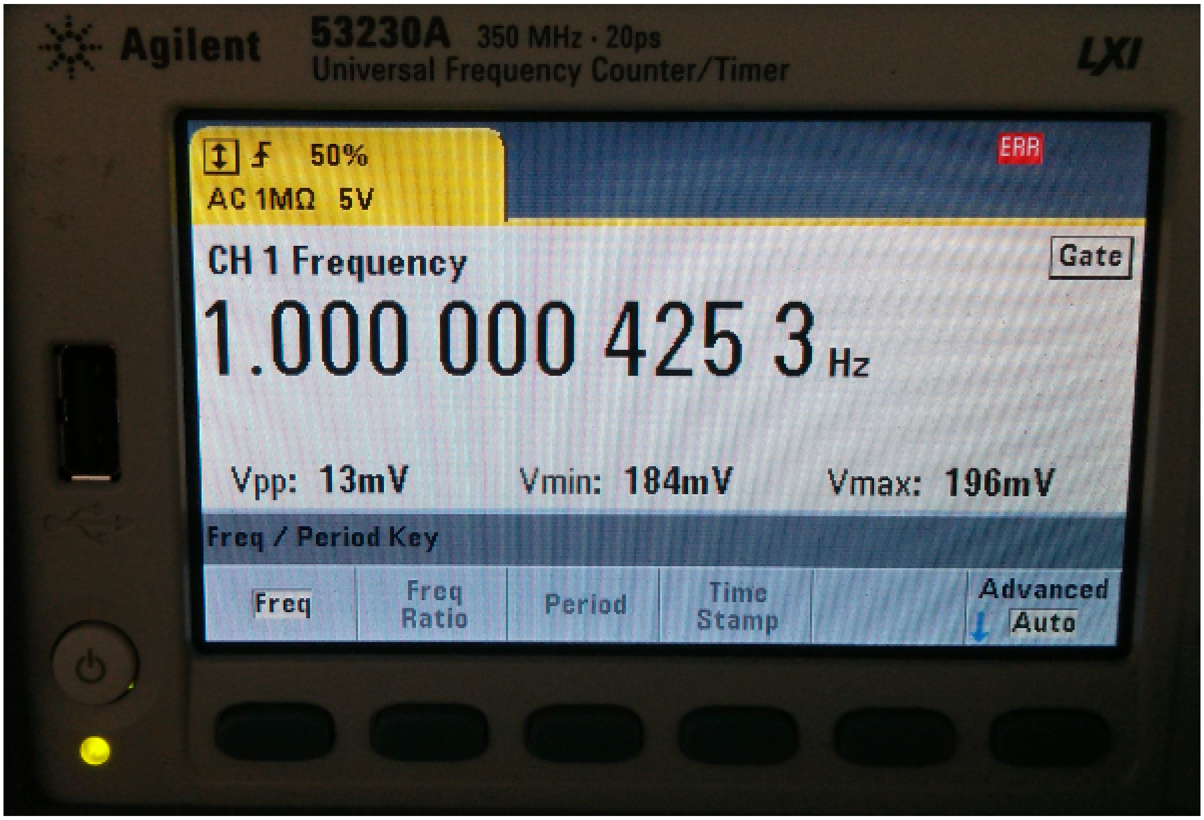

Figure 3. Output after fine compensation correction

However, if instead coarse compensation is enabled after computing and programming integral values for RTC correction, the frequency output measured is shown in Figure 4. Please note that the coarse compensated output is not as precise as fine compensated output. The error is about 5 PPM.

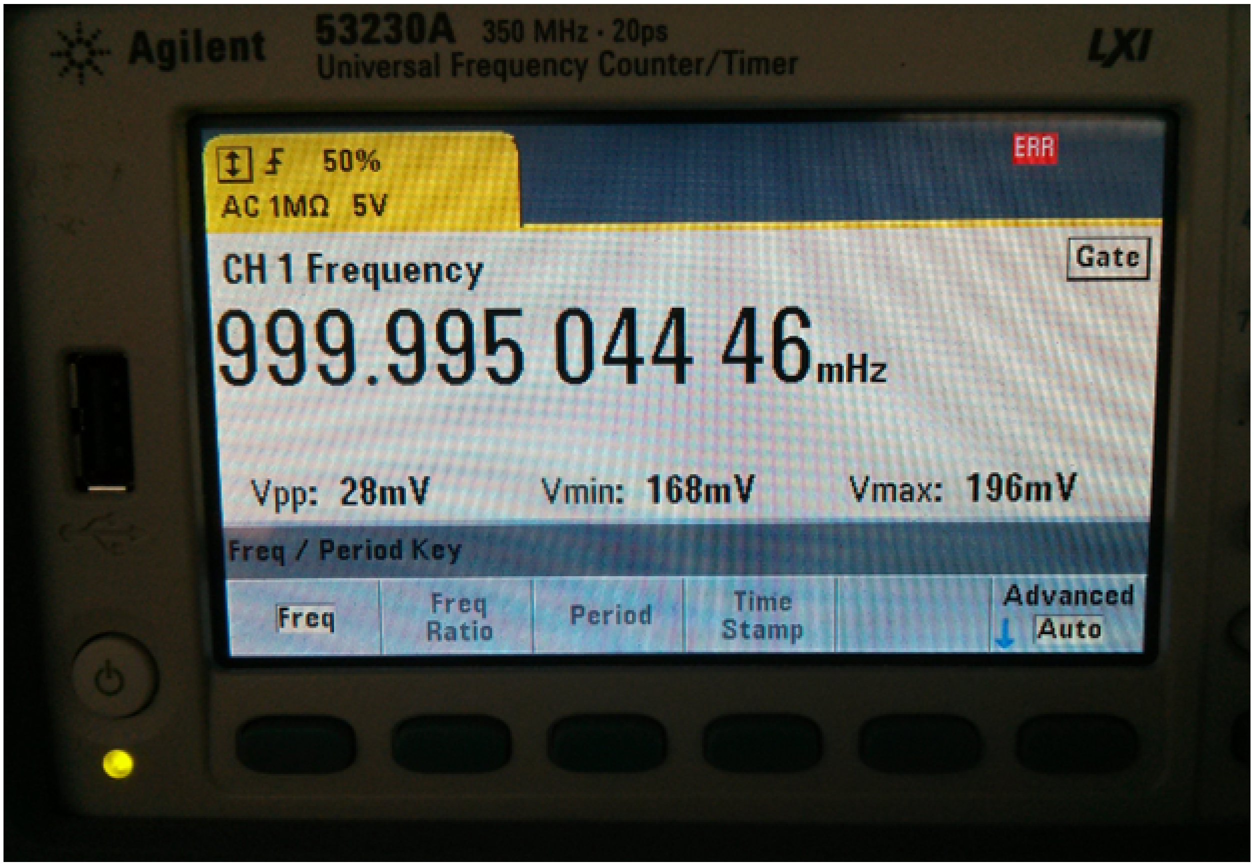

Figure 4. Output after coarse compensation correction

5 Revision history

Table 4. Revision history

Revision number	Date	Substantial changes
0	07/2014	Initial release

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number AN4949
Revision 0, 08/2014