AN593

Temperature Sensor Usage and Frequency Calibration for the Si4x3x

1. Introduction

This document describes how to use and calibrate the built-in temperature sensor of EZRadioPRO radios. It also describes how to calibrate the frequency offset caused by crystal temperature dependency.

2. Temperature Sensor

An integrated on-chip analog temperature sensor is available in the radio. The temperature sensor is automatically enabled when it is selected as the input of the internal General Purpose ADC or when the analog voltage output of the temperature sensor is selected to be on the analog test bus. If the General Purpose ADC is used, the temperature value can be read over the SPI bus through register, “11h—ADC Value”.

2.1. Registers for Temperature Sensor

Table 1 lists the registers used to operate and calibrate the temperature sensor. Tsrange[1:0] defines the temperature sensor range and operating mode, and tstrim[3:0] provides calibration options for the temperature sensor. Tvoffs[7:0] can be used to automatically compensate for the offset of the measured temperature value.

The range of the temperature sensor is configurable. Table 2 lists the settings for different temperature ranges and performance.

Table 1. Temperature Sensor Registers

Add	R/W	Function	D7	D6	D5	D4	D3	D2	D1	D0
12	R/W	Temperature Sensor Calibration	tsrange[1:0]		entsoff	entstrim	tstrim[3:0]			
13	R/W	Temperature Value Offset	tvoffs[7:0]							

Table 2. Temperature Sensor Range

entoff	tsrange[1]	tsrange[0]	Temperature Range	Unit	Slope	ADC8 LSB
1	0	0	–64 to 64	°C	8 mV/°C	0.5 °C
1	0	1	–64 to 192	°C	4 mV/°C	1 °C
1	1	0	0 to 128	°C	8 mV/°C	0.5 °C
1	1	1	–40 to 216	°F	4 mV/°F	1 °F
0*	1	0	0 to 341	°K	3 mV/°K	1.333 °K

***Note:** Absolute temperature mode; no temperature shift. This mode is only for test purposes. POR value of EN_TOFF is 1.

2.2. Operation of the Temperature Sensor

To use the temperature sensor, perform the following steps:

1. Select the temperature sensor as the input of the general purpose ADC, “Register 0Fh. ADCconfiguration”, adcsel[2:0] = 000
2. Set the reference for ADC, “Register 0Fh. ADC Configuration”, adcref[1:0] = 00
3. Set the temperature range for ADC, "Register 12h. Temperature Sensor Calibration", tsrange[1:0]
4. Measure and set the offset of measured temperature, “Register 12h. Temperature Sensor Calibration”, entsoffs = 1
5. Start the measurement by triggering the ADC, “Register 0Fh. ADC Configuration”, adcstart = 1
6. Read temperature value, read contents of “Register 11h. ADC Value” after 350 µs.

The result measured by the General Purpose ADC can be translated to temperature by the following equation:

T = ADC x ADC LSB + Lowest Temperature in the temp range.

When the chip is in receiving or transmitting mode, the temperature of the die may rise. To avoid measurement error, it is highly recommended to measure the temperature just before transmitting or receiving a packet.

2.3. Calibration of Temperature Sensor

The slope of the temperature sensor is linear and monotonic. However, for better than 10 °C absolute accuracy, each chip has to be calibrated. This document describes two different calibration methods. After using one of the two or both calibrations, better than ±3 °C absolute accuracy can be achieved.

2.3.1. Calibration by the Offset Register

The temperature offset error has to be measured for all parts individually. The offset can be written into the Temperature Sensor Offset register. After enabling the automatic temperature calibration (by setting the entsoffs bit in the “Register 12h.Temperature Sensor Control” register), the radio automatically adjusts the offset error for each temperature value read.

In Register 12h, the MSB (tvoffs[7]) is the sign bit, and the remaining bits hold the offset value. The measured temperature value will increase if the offset register value changes from –64 to 63 (0xFF to 0x7F).

2.3.2. Calibration by Trimming

The temperature measurement is realized as an analog circuit. The reference voltage of the circuit can be calibrated by the tstrim bits if the entstrim bit is set.

If tstrim[3] = 1, the measured temperature value will increase when the tstrim[2:0] value increases.

If tstrim[3] = 0, the measured temperature value will decrease when the tstrim[2:0] value increases.

3. Frequency Calibration

Usually there are frequency differences between the transmitter and receiver reference frequencies due to the inaccuracy of the crystal. Worldwide regulatory standards control the transmit frequency accuracy; however, there are no such rules for the receiver. The frequency offset between transmit and receive frequencies degrades the quality of the wireless RF link.

3.1. Automatic Frequency Calibration Mechanism

There are internal autonomous algorithms that can help cancel the offset (Internal modem compensation and Automatic Frequency Calibration).

The internal modem compensation circuit is part of the digital modem. That circuit works when the radio is set to receive mode. It can tolerate a frequency offset up to 0.25 times the IF bandwidth.

There is another circuit called Auto Frequency Compensation (AFC), which also works in receive mode. The AFC slightly modifies the Synthesizer Local Oscillator frequency and tries to determine the transmit frequency. When AFC is enabled, the received signal will be centered in the passband of the IF filter, providing optimal sensitivity and selectivity over a wider range of frequency offsets up to 0.35 times the IF bandwidth.

AFC works during preamble reception. One of the drawbacks of AFC is that it requires one additional byte of preamble.

The circuit can be enabled by setting enafc=1 in register 0x1D.

Note: If AFC is enabled, the measured offset can be read from the Frequency Offset registers. This information can be used later for that particular transmitter board by disabling AFC and entering this value into the Frequency Offset register.

However, these mechanisms are sometimes not suitable for the application, and it can happen that the offset has to be canceled for each board manually. For this purpose, the Frequency Offset register and the Crystal Oscillator Load Capacitance register can be used.

If the AFC is disabled and the frequency error is known, the synthesizer frequency may be adjusted by programming the Frequency Offset register fo[9:0] in “Register 73h. Frequency Offset 1” and “Register 74h. Frequency Offset 2”.

The frequency offset adjustment is implemented by shifting the Synthesizer Local Oscillator frequency. This register is a signed register. In order to get a negative offset, it is necessary to take the 2s complement of the positive offset number. The offset can be calculated by the following formula:

Desired Offset = 156.25 Hz x (hbsel +1) x fo[9:0], where hbsel is high band select in Register 75h.

Note: AFC and Frequency Offset register cannot be used simultaneously, since they share the same registers.

The following sections are a summary about the registers involved in different methods.

3.1.1. Frequency Offset Registers

Register 73h. Frequency Offset 1

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	fo[7:0]							
Type	R/W							

Register 74h. Frequency Offset 2

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name							fo[9:8]	
Type	R						R/W	

3.1.2. Crystal Load Capacitance Register

Register 09h. 30 MHz Crystal Oscillator Load Capacitance

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	xtalshft	xlc[6:0]						
Type	R/W	R/W						

3.1.3. AFC Control

Register 1Dh. AFC Loop Gearshift Override

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	afcbd	enafc	afcgearh[2:0]			afcgearl[2:0]		
Type	R/W	R/W	R/W			R/W		

3.2. Manual Calibration of Frequency

The accuracy of the crystal oscillator determines the accuracy of the carrier frequency. The crystal accuracy is the sum of the following factors:

1. Initial (production) tolerance
2. Crystal load capacitance mismatch
3. Temperature dependency
4. Aging

All these numbers are on the order of 1–100 ppm for a typical 30 MHz crystal.

The offset coming from the initial tolerance and the crystal load capacitance mismatch can be compensated for during manufacturing. However, the temperature dependency and aging can be compensated for only during runtime.

3.2.1. Calibration of Frequency Offset Caused by Initial Tolerance and Load Capacitance

The easiest way to compensate the frequency error from the initial tolerance and the crystal load capacitance mismatch is to tune the Crystal Oscillator Load capacitance value as long as the center frequency is accurate and there is not more than ±1 kHz offset.

The method described in the next section can be used for initial calibration purposes.

3.2.1.1. Using a Spectrum Analyzer (Works only in TX Mode, not for RX)

Turn the radio on in continuous wave transmit mode and measure the center frequency of the transmitted signal. Make sure that a small resolution bandwidth is used so that the frequency can be measured accurately and so the Crystal Oscillator Load Capacitance register can be tuned. This tuning can happen as long as the frequency accuracy is better than 1 kHz.

3.2.1.2. Using Frequency Counter

The microcontroller clock output can be multiplexed onto GPIO2 of the radio. Since that clock is divided from the XTAL frequency, measuring it accurately with a frequency counter provides the accuracy of the crystal. This allows for tuning of the Crystal Oscillator Load Capacitance register as long as the exact microcontroller clock output is observed.

Note: An oscilloscope does not provide enough accuracy for this test, and it is necessary to use a frequency counter that can measure the clock with ppm accuracy.

The crystal load capacitance can be digitally programmed to accommodate crystals with various load capacitance requirements and to adjust the frequency of the crystal oscillator. The tuning of the crystal load capacitance is programmed through the xlc[6:0] field of “Register 09h. 30 MHz Crystal Oscillator Load Capacitance”. The total internal capacitance is 16.3 pF and is adjustable in approximately 127 steps (97 fF/step). The xtalshift bit provides a coarse shift in frequency but is not binary with xlc[6:0].

The typical value of the total on-chip capacitance Cint can be calculated as follows:

Cint = 1.8 pF + 0.085 pF x xlc[6:0] + 3.7 pF x xtalshift

It is important to provide the required amount of load capacitance specified by the crystal manufacturer. The total load capacitance, Cload, seen by the crystal can be calculated by adding the sum of all external parasitic PCB capacitances to Cint.

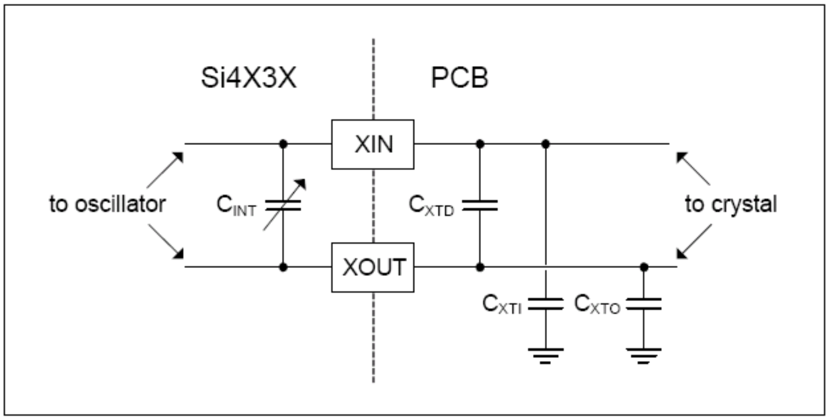

Figure 1. Connected Capacitance

The resulting load capacitance will be:

Cload = Cint + CXTD + CXTI x CXTO / (CXTI + CXTO)

CXTD, CXTI, and CXTO are present as PCB parasitics.

The internal load capacitance register value for the customer boards should be decided by measurement under the operating frequency.

The diagram in Figure 1 can be used in deciding the load capacitance value.

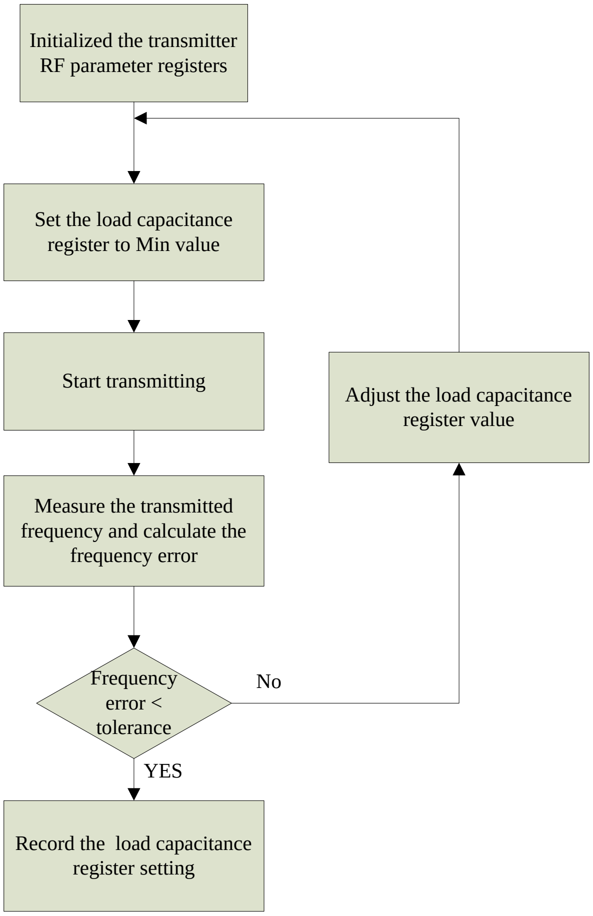
Initialized the transmitter RF parameter registers
Set the load capacitance register to Min value
Start transmitting
Measure the transmitted frequency and calculate the frequency error
Frequency error < tolerance
No
YES
Adjust the load capacitance register value
Record the load capacitance register setting

Figure 2. Determining the Load Capacitance Value

3.2.2. Calibration of Frequency Offset Caused by Temperature Change

In several narrow-band systems, the accuracy requirements cannot be achieved without compensating for the temperature dependency of the crystal during runtime.

A cost-effective method is to utilize the on-chip temperature sensor and digitally adjust the frequency offset register with a suitable software control loop.

Figure 3 shows the temperature characteristics of the reference frequency, measured for recommended crystals from various vendors.

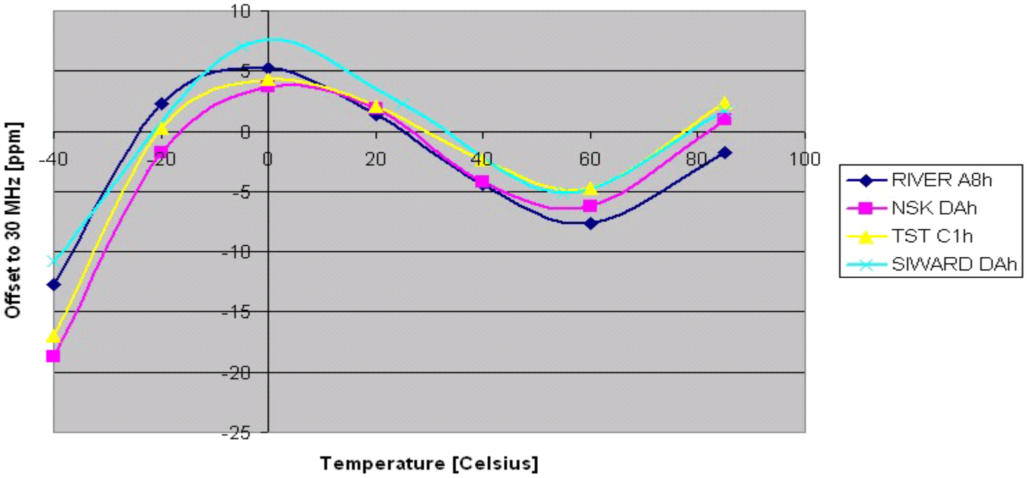

Figure 3. Crystal Frequencies vs. Temperature

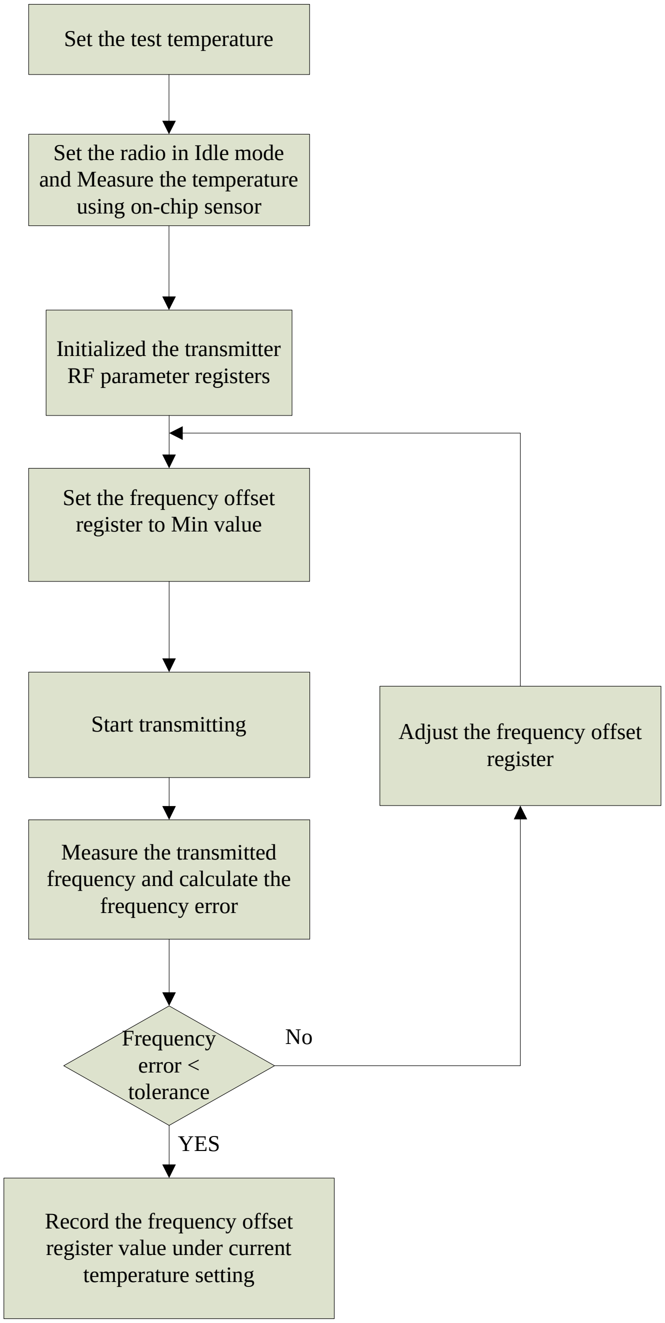
Set the test temperature
Set the radio in Idle mode and Measure the temperature using on-chip sensor
Initialized the transmitter RF parameter registers
Set the frequency offset register to Min value
Start transmitting
Adjust the frequency offset register
Measure the transmitted frequency and calculate the frequency error
Frequency error < tolerance
No
YES
Record the frequency offset register value under current temperature setting

Figure 4. Measuring the Frequency Offset Value at Given Temperature

The temperature dependency comes from the crystal itself, and the curves shown are typical for AT-cut crystals. Other cuts have parabolic frequency error vs. temperature curves and a very small frequency error gradient around room temperature, but, compared to the AT-cut, significantly larger overall frequency variation across extended temperature ranges.

The actual frequency of the crystal doesn't change linearly with temperature variation as shown on Figure 3. The actual temperature has to be periodically measured for accurate compensation. In this case, a temperature vs. frequency offset register value table should be measured and stored in the MCU in advance. After measuring the ambient temperature using the internal temperature sensor, the frequency offset value can be easily determined from the lookup table according to the measured temperature and used to compensate for frequency drift.

Each chip or board should be frequency-calibrated separately because there is a temperature measurement variation from chip to chip. To achieve 3 °C accuracy, each chip should use its own compensation offset value because, under the same temperature, the measured temperature can spread up to ±10 degrees. It is not practical to find a common compensation value for a batch of boards/chips.

The flowchart in Figure 4 can be used to help determine the frequency offset value at different temperature settings.

Simplicity Studio

One-click access to MCU tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

www.silabs.com/simplicity

MCU Portfolio
www.silabs.com/mcu

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support and Community
community.silabs.com

Disclaimer

Silicon Laboratories intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Laboratories products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Laboratories reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Laboratories shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products must not be used within any Life Support System without the specific written consent of Silicon Laboratories. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Laboratories products are generally not intended for military applications. Silicon Laboratories products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

Trademark Information

Silicon Laboratories Inc., Silicon Laboratories, Silicon Labs, SiLabs and the Silicon Labs logo, CMEMS®, EFM, EFM32, EFR, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZMac®, EZRadio®, EZRadioPRO®, DSPLL®, ISOmodem ®, Precision32®, ProSLIC®, SiPHY®, USBXpress® and others are trademarks or registered trademarks of Silicon Laboratories Inc. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.

SILICON LABS®

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

http://www.silabs.com