PSoC® 3 and PSoC 5 Temperature Measurement with Thermistor

AN66477

Author: Archana Yarlagadda
Associated Project: Yes
Associated Part Family: CY8C3xxx,CY8C5xxx
Software Version: PSoC® Creator™ 1.0
Associated Application Notes: AN2017, AN57724

Abstract

PSoC® 3 and PSoC 5 vastly simplifies temperature measurement with a thermistor, by providing a component. This implementation is based upon the use of a constant voltage method of thermistor resistance measurement. The application note project gives you practical implementation of temperature sensing using the thermistor component for PSoC® 3 and PSoC 5.

Introduction

A thermistor is a temperature-sensitive resistor in which resistance varies with temperature change. There are two types of thermistors: positive temperature coefficient (PTC) thermistors and negative temperature coefficient (NTC) thermistors. Because they are more commonly used, this application note describes NTC thermistors, in which resistance decreases with increase in temperature. Based on this principle, temperature is calculated by measuring the resistance.

This application node describes the thermistor component that is compatible with PSoC 3 and PSoC 5. This component simplifies the implementation of temperature measurement with a thermistor. It calculates the coefficients required for a thermistor and generates the code required to obtain the temperature. The component customizer and firmware provide the interface to the component. The parameters of the thermistor are set using the customizer, and the output (temperature) is obtained through a function call. This application note describes an example project that shows how to set up the thermistor calculation component, along with other components to obtain the temperature. The other components required to measure the temperature are provided below:

Essential

- Thermistor (Provided with this application note)
- Analog to digital convertor (ADC)
- Multiplexer (MUX)

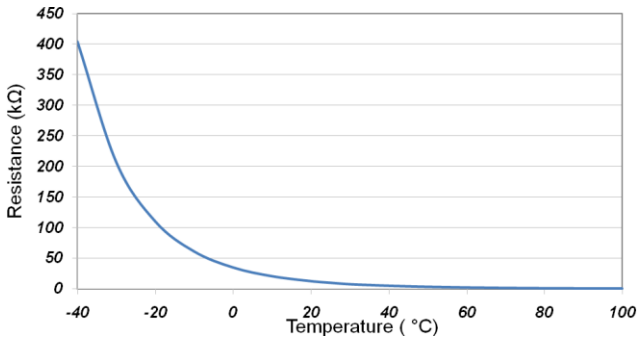
Optional

- Voltage digital to analog convertor (VDAC)
- Display (LCD)

Thermistor Parameters

The variation of resistance with temperature for a thermistor is non-linear. Figure 1 shows a typical resistance versus temperature of a thermistor.

Figure 1. Resistance vs. Temperature Curve of Thermistor



Because of this non-linear response, calculation of temperature from the resistance requires an empirical equation called the Steinhart-Hart equation, shown in Equation 1.

$$\frac{1}{T_K} = A + B * ln(R_T) + C * (ln(R_T))^3 \qquad \text{Equation 1}$$

Where:

T_K = Temperature in kelvins

A, B, and C = Steinhart coefficients

R_T = Thermistor resistance in ohms

Some datasheets provide the three Steinhart coefficients (A, B, and C). Other datasheets provide "Temperature coefficient" (Alpha) values, "Sensitive index" (Beta) values, or both. Although the Alpha or Beta coefficients can determine temperature, they are limited to a specific temperature range for which they are specified. The Steinhart-Hart equation does not have this limitation.

Because the parameters provided for thermistors can vary, their usage and interchangeability in an application can be complicated. To address this issue, the thermistor component calculates the required A, B, C coefficients, based on the resistance versus temperature table or curve available in datasheets. If these values are not provided, users can calculate them on a test bench.

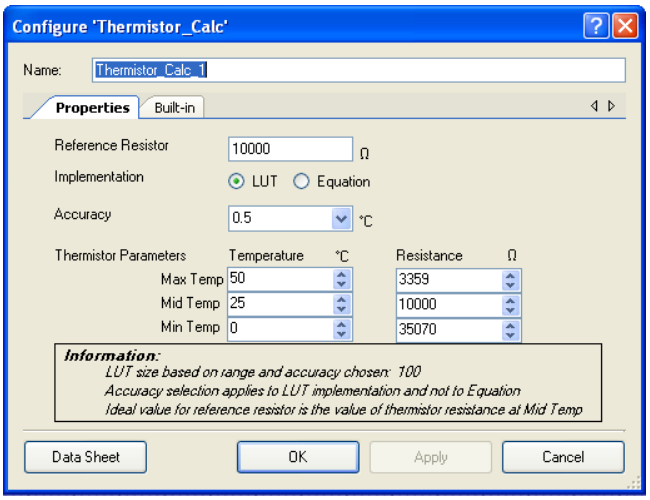
Thermistor Calculation Component

The “Thermistor_Calc” component is a software component with no hardware input or output. The customizer and a firmware function call provide the interface to the component. The component uses the temperature and resistance of the thermistor, calculates the Steinhart coefficients, and generates the code required for the temperature measurement.

The customizer gives users the option to specify the external reference resistor value. This value should be close to the resistance of the thermistor at the middle of the temperature range, to get the maximum accuracy at either extremes of temperature.

Figure 2 shows the customizer for the component. The section marked **Thermistor Parameters** contains the temperature and resistance of the thermistor, and the **Reference Resistor** field contains the optional external reference resistor value.

Figure 2. Thermistor_Calc Component Customizer



The customizer also provides the **Implementation** option of using either the equation method or look-up-table (LUT) method of generating the required code. Table 1 shows the tradeoffs between the two methods.

Table 1. Tradeoffs Between Equation and LUT Methods

	Equation	LUT	Comments
Accuracy (+/-) °C	> 0.01	≤ 0.01	Accuracy shown is the accuracy of calculation alone. This excludes the accuracy of the thermistor, reference resistor, voltage reference, or ADC. Though the accuracy of equation is greater than ±0.01 °C, the output is limited to resolution of ± 0.01 °C because it is scaled by 100.
Memory usage	Higher if FP* not included	Lower if FP* not included	The memory usage of the equation method is fixed because of the use of the floating-point library. If other components or functions already use the floating-point library, the equation method is considered efficient. The memory usage of LUT depends on the range and accuracy chosen.
	Lower if FP* included	Higher if FP* included	
Range	Wider than specified	Limited to specified	In the equation method, the temperature can be measured outside of the range specified (lower accuracy). In LUT, temperature values outside the range specified are not measured.

*Floating-Point Library (FP)

Because the accuracy and temperature range provided by the user generate the LUT, the measured temperature range and the accuracy in the LUT method are limited to the user-supplied parameters. The highest and the lowest temperatures measured are equal to the highest and lowest temperature values provided in the customizer. Therefore, the measured temperatures saturates at these temperature values.

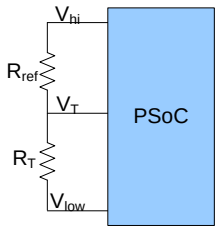
Because the accuracy of the equation method is based on the accuracy that obtained with the Steinhart-Hart equation, accuracy selection is not applicable to this method. The range of temperature that can be measured is also not limited to the range selected. This method also allows the measurement of temperatures outside of the specified range. The only difference is that the accuracy is lower for the temperatures measured outside of the range specified because the coefficients are calculated for the temperature specified and the extrapolation provides lower accuracy. Therefore, if another component or function uses the floating-point library, or if memory usage is not a constraint, user should use the equation method.

For information about other customizer options, see the component datasheet.

Resistance Measurement

The measurement of resistance in the component is based on the application note *"Sensing - A Thermistor-Based Thermometer, PSoC Style - AN2017"*. Figure 3 shows the external connections for constant voltage.

Figure 3. Thermistor Connection to PSoC in Constant Voltage



Where V_{hi} and V_{low} can be sourced by PSoC or can be external. Most common connections for V_{hi} and V_{low} are the V_{dd} and V_{ss} of the device.

Equation 2 shows the resistance of the thermistor calculated from the connections shown in Figure 3.

$$R_T = R_{ref}\left(\frac{V_T - V_{low}}{V_{hi} - V_T}\right) \qquad \text{Equation 2}$$

(V_t-V_{low}) and *(V_{hi}-V_t)* are distinct differential measurements determined by the ADC.

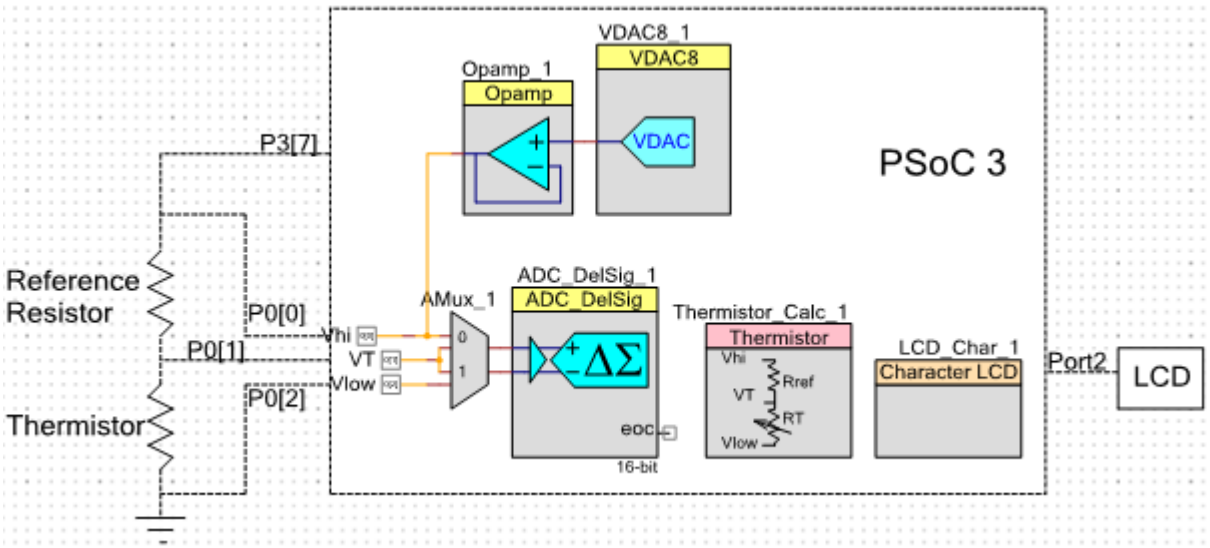
In this method, the differential measurement (V_{hi} - V_T and V_T - V_{low}) cancels the offset, and the ratio between the two cancels the gain error.

The resistor and the thermistor can be used interchangeably in connection. When the thermistor is connected between V_T and V_{low}, the voltage V_T decreases when temperature increases. When the thermistor is connected between V_{hi} and $Vl_{ow,}$, voltage V_T increases when temperature increases. The thermistor component handles both cases, because it depends on the voltage drop and not the absolute voltage.

Interface with PSoC 3

This application note provides the thermistor component. Application note *"Component Hierarchy in PSoC Creator – AN57724"* describes the addition of a component into a project. After you add the component to a library, you can add it to the workspace as any other component. Figure 2 on page 2 shows the other components that are required for this project. It also shows a schematic of external connections to PSoC 3.

Figure 4. Schematic for Thermistor Measurement



The following sections describe the function of each component.

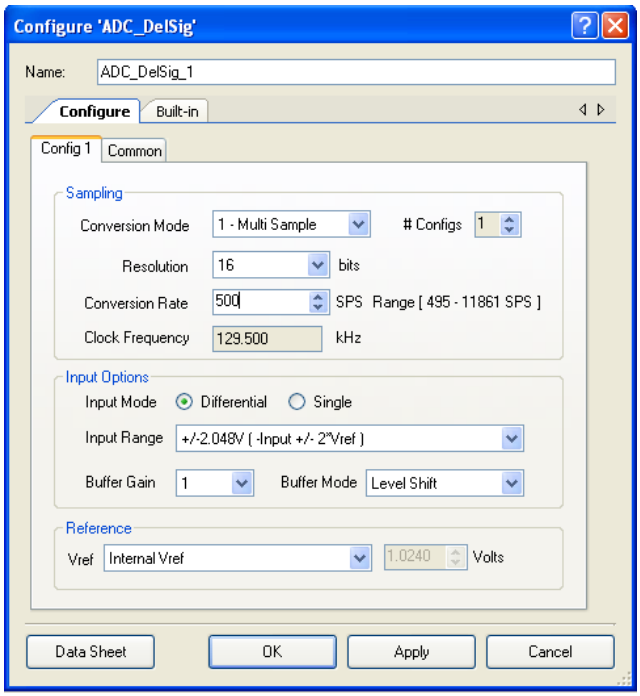
AMux

The AMux multiplexes the inputs into the ADC. The three voltages required are V_t, V_{hi} and $V_{low,}$ as shown in the Equation 2. If the ADC is a single-ended input, the AMUX must also be single-ended with three channels. If the ADC is differential, it can measure the difference voltages V_t–V_{low} and V_{hi}–V_t. Therefore, the setting for the AMux is differential and has two channels, as shown in Figure 4.

DelSig ADC

The ADC converts the analog voltages into digital values. The voltages required are V_t, V_{hi}, and $V_{low,}$ as shown in Equation 2. If the ADC is single-ended, three ADC samples corresponding to the three voltages are required. If the ADC is differential as shown in Figure 4, two ADC samples are required. The two samples provide the differential voltages shown in the numerator and denominator of Equation 2. The thermistor component is adaptable to both the single-ended and differential type of ADC. Figure 5 on page 4 shows the configuration of ADC in differential mode in the associated project.

Figure 5. ADC Configuration for Thermistor Project



The two options for the ADC buffer are **Level-Shift** and **Rail-Rail**. If you choose the level-shift option, the V_{ssa} can be measured accurately because it is in the range of ADC. This is based on the DelSig ADC datasheet. The VDAC can then be used to choose the V_{hi} such that it is within the ADC range (between V_{dda} – 650mV and V_{ssa} – 100mV for the level-shift option).

Because the voltage ratio in Equation 2 on page 3 becomes very small at the extremes of the temperature range, the ADC resolution must be based on the resolution required at the extreme temperatures in the range you want to use. For example, if the reference voltage V_{ref} is 2 V, resolution is 0.01 °C and the range of temperature is 0 to 100 °C. The V_T voltage variation for the 0.01 °C variation at 100 °C is approximately 70 µV. Therefore, a 16-bit ADC is required for these conditions. Similarly, based on the reference voltage, range, resolution; accuracy of temperature measurement, and the ADC range, the resolution of ADC must be determined for the specific application. This selection is important only when a wide range of temperature with very high accuracy is required. For a range of 50 °C with accuracy of 0.1 °C, an 8-bit ADC is sufficient.

The conversion rate of the ADC is dependent on the speed at which the temperature changes. In most applications, the conversion rate can be set very low, because the change in temperature is very slow.

LCD

LCD displays the output of the measurement. This is for the display purposes only; it can be replaced as required in the target application.

VDAC

VDAC generates the reference voltage V_{hi}. The reference voltages V_{hi} and V_{low} can be external or internal voltages. Use of VDAC is just one way of obtaining the reference voltages. In the schematic shown in Figure 4 on page 3 , V_{hi} is internal and is generated using a VDAC whereas V_{low} is connected to ground (V_{ssa}). The VDAC is set such that V_{hi} is lower than V_{dda} – 650mV. These selections are made such that the voltages are within the range of ADC. For example, the output of VDAC in the example project is 1.9 V.

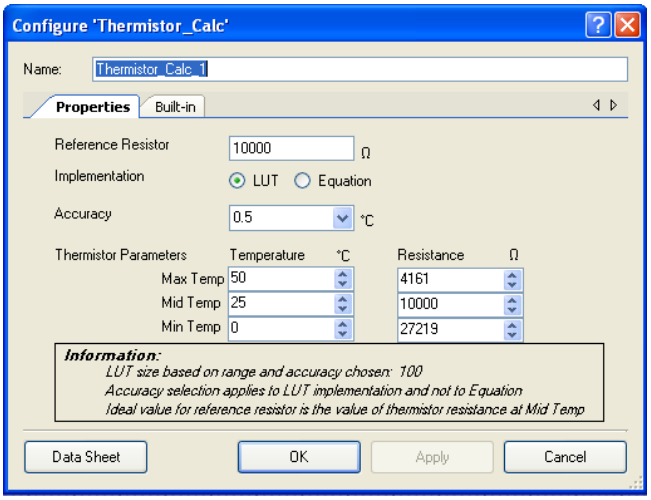
OPAMP

Opamp buffers the output of the VDAC to prevent loading. The loading can occur because of the comparable internal resistance of VDAC and the load (reference resistor and thermistor).

Thermistor component

The thermistor component converts voltage to corresponding temperature, as this application note explained earlier. The thermistor parameters are set in the thermistor customizer, and the firmware provides the actual interface. The thermistor used on the Cy8CKIT – 025 is NCP18XH103F03RB. The resistance of this thermistor is 10 kΩ. Thus, the reference resistor was choosen as 10 kΩ. The resistances vs. temperature values were readily available from the datasheets and we entered as shown in Figure 6.

Figure 6. Thermistor Component Settings in Example Project



Firmware

The interface to the thermistor component occurs through a function call. The function is shown in the following code:

```
fTemp=Thermistor_1_GetTemperature(iVtherm, iVref);
```

Where *iVtherm* and *iVref* are voltages across the thermistor and the reference resistor, respectively, as Equation 3 shows:

$$iVtherm = V_T - V_{low}$$

$$iVref = V_{hi} - V_T \qquad \text{Equation 3}$$

Figure 4 on page 3 shows the circuit connections with differential ADC. Two ADC samples corresponding to *iVtherm* and *iVref* are obtained directly. In the case of single-ended ADC, three ADC samples corresponding to V_{hi}, V_T and V_{low} are obtained. The parameters for the function call,

iVtherm and *iVref*, are then obtained from these samples, based on Equation 3.

In the example project, the differential mode for ADC is used. As mentioned earlier, the thermistor and the reference resistor can be interchanged. The Thermistor is connected between V_{hi} and V_T in Cy8CKIT – 025. Thus the code in main.c of example project, for setting the Mux, is as follows:

```
/*AMUX channel selections*/
#define AMUX_1_VT 0
#define AMUX_1_VREF 1
```

The return value of the function call is the required temperature. The conversion from voltage to temperature is based on the Steinhart-Hart equation, using either the equation or LUT method depending on what the user selects.

The returned temperature value is scaled up by 100 and passed as a 32-bit integer value. For example, if the temperature is 23.45, the return value is 2345 in both the equation and LUT method. The target application can process this value as user chooses.

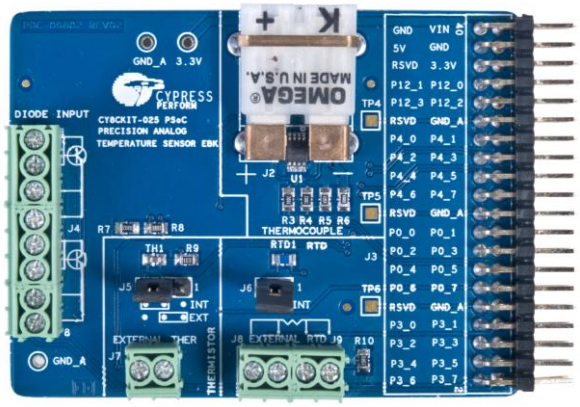
CY8CKIT-025 for Thermistor Interface:

CY8CKIT-025 is a temperature sensor interface kit that has interface for diode, thermistor, thermocouple and an RTD, as shown in Figure 7. This kit has been used for the example project of this application note. It has options of both internal and external thermistor usage on the board. The settings in the project are for internal (on-board) thermistor NCP18XH103F03RB. This board plugs into Cy8CKIT -030 and can be programmed with the example project attached with this application note.

Note In the example project V_{ref} and V_{hi} are connected internally, to avoid any external connection requirement. Connecting these two nodes to pins and connecting them externally gives more accurate results. By doing this, the voltage measured on V_{hi} pin will be the actual voltage on the board.

More details about the kit can be found in Cypress website.

Figure 7. CY8CKIT-025 Temperature Sensor EBK



Summary

This application note describes the NTC thermistor component for measuring temperature, which can be used with PSoC 3 and PSoC 5. The code generated is adaptable to any NTC thermistor. The user can choose from two options for generating the code. The project also requires other components, as shown in this application note.

About the Author

Name: Archana Yarlagadda

Title: Senior Applications Engineer

Background: Archana has a Master of Science, Electrical Engineering degree from the University of Tennessee and is interested in analog and mixed-signal systems.

Contact: yara@cypress.com

Document History

Document Title: PSoC® 3 and PSoC 5 Temperature Measurement with Thermistor

Document Number: 001-66477

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	3148830	YARA	01/20/11	New application note.
*A	3216577	YARA	06/06/2011	1. In the library project associated with the AN, The Thermistor_v1_0 has been changed to Thermistor_Calc_v1_0. 2. Text in the AN has been modified to support the same. 3. The Library for the component has been changed from "CY_Ref" to "CYRef" as per the change in Spec:001-58801

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone: 408-943-2600
Fax: 408-943-4730
http://www.cypress.com/