AN2278
Application note

Temperature sensor application using ST LM135

Introduction

This application note describes how to develop a temperature sensor application using the ST LM135 temperature sensor. Application development was done on a Raisonance REva board. The Raisonance REva board is a universal board with several useful features for application development. It has been designed for quick and easy evaluation of a wide range of microcontrollers. This application uses the ST72325 microcontroller, which is interfaced to REva using a daughter board.

The main components required for the development of this application are:

■ ST LM135 temperature sensor

■ ST72325 microcontroller

■ ST24C02 I^{2}C EEPROM

■ ST3232 communication interface

Some of the useful features of the REva board are:

■ Digital and analog I/O evaluation features including on-board LEDs, buttons, switches, external analog connector, temperature sensor and potentiometer

■ On-board I^{2}C EEPROM and bus extension connector

■ On-board RS232 driver and DB9 connector

■ SPI, CAN and USB connections (depending on the target device)

■ Embedded RLink for in-circuit debugging and in-circuit programming V$_{DD}$ settings for 1.8 V, 3.3 V, 5 V microcontrollers

■ USB powered, no external power required

Section 1 highlights the important features of the LM135 sensor and explains the temperature calculation used in the application. *Section 2* explains the temperature sensor management and *Section 3* focusses on explaining the application flow. *Section 4* gives an overview of the hardware setup required to implement the application.

Contents

1   LM135 temperature sensor

LM135 is a precision temperature sensor and can be easily calibrated. It operates as a 2-terminal zener and the break-down voltage is directly proportional to the absolute temperature at 10 mV/K. The temperature measurement range of the LM135 is -55 °C to +150 °C.

Temperature calculation in Celsius:

The ST72325 microcontroller has a 10-bit ADC, so the maximum digital value it is capable of holding is 0x3FF. The maximum voltage available at the V+ terminal of the LM135 (refer to *Figure 4*) is approximately 4800 mV (after taking into account a voltage drop of 200 mV). Lets say x is the ADC value corresponding to an unknown temperature value.

Therefore the temperature value in kelvin is k= [(x * 4800) / (0x3FF)]/10; as zener breakdown voltage for ST LM135 changes by 10 mV for each degree Kelvin change. The temperature value in Celsius is C=(k - 273) / 100.

2   Temperature sensor application

This temperature monitoring application senses and logs the temperature value into an EEPROM. If the sensed temperature is beyond the threshold limits, specified by the user, an alarm signal is indicated by toggling the LED.

The application flow is as follows:

1.   On starting the application, it enters configuration mode and the user is prompted to set the lower and upper bounds for the normal temperature values.

2.   Once the threshold values are set, the application enters normal mode. An average of 16 sample values, taken over a period of 1 second, is displayed by the Windows Hyperterminal application via the on-chip SCI interface.

Note: *It is advisable that number of samples to be recorded should be of order 2^n, where n is an integer greater than 0. This will help in calculating the average temperature value using a 'right shift(>>)' operator.*

3.   When the temperature crosses the threshold values, the application shows a 'Critical value' message in the hyperterminal window and the LEDs on the board toggle.

4.   The application keeps account of the minimum and maximum temperature values over a span of 1 hour. At the end of an hour these minimum and maximum temperature values are stored in the on-board I^2C EEPROM.

5.   You can re-enter configuration mode and change the threshold values by pressing Button5 on the board. Since Button5 is not connected to one of the microcontroller interrupt pins, you have to keep the button pressed until the firmware polls it.

6.   To view the log of temperature values stored in the EEPROM, you have to press Button6 on the board. The temperature values (in Celsius) are displayed in the hyperterminal window. Since Button6 is not connected to one of the microcontroller interrupt pins, you have to keep the button pressed until the firmware polls it.

3 Software configuration

All the source files are in 'C' language and the application uses ST7 software library functions.

The application works in two modes:

1. Configuration mode (refer to the flowchart in *Figure 1*)
2. Normal mode (refer to the flowcharts in *Figure 2* and *Figure 3*)

Since Button5 and Button6 are not connected to the microcontroller interrupt I/Os, polling mode is used to read the request status resulting from pressing these buttons.

The main function is organized as follows:

```
while(1)
{
----------------------------
----------------------------
code=1;
switch (code)
{
    case 1:
//Configuration Mode
break;
case 2:
//Read temperature log from EEPROM and display on hyperterminal
break;
case 3:
// Sense temperature in 16 samples (refer to the flowchart in Figure
3.)
// Display average temperature every 1 sec on hyperterminal
// if temperature is out of bounds(set in configuration mode)
  // then toggle LED
  // else continue
// Record max and min temperature over 1 hr
// Write max and min temperature to EEPROM
break;
}
----------------------------
----------------------------
if(!(IO_Read(IO_PORT_C) & 0x20)) //Poll for Button 5
      code=1;
    else if(!(IO_Read(IO_PORT_C) & 0x80)) //Poll for Button 6
      code=2;
----------------------------
----------------------------
}
```
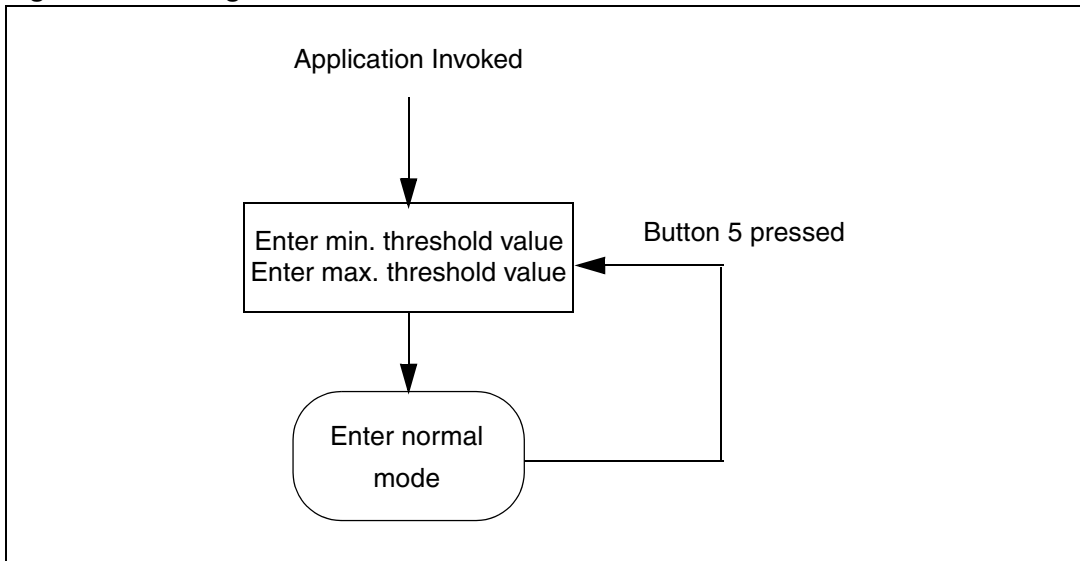
Figure 1. Configuration mode flowchart
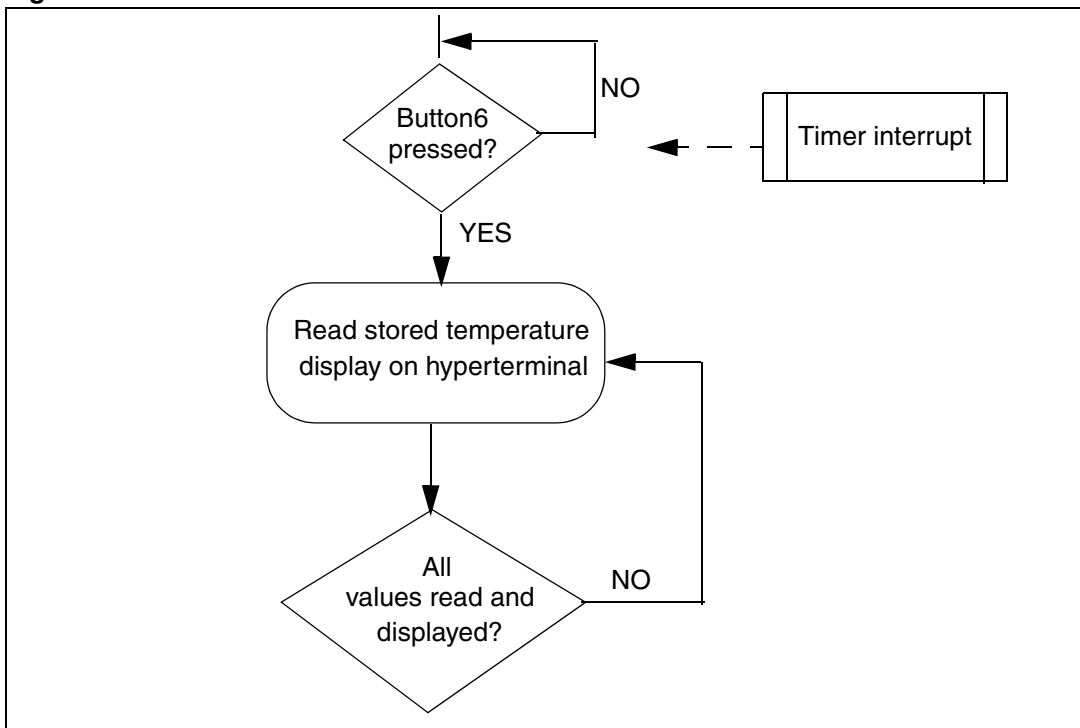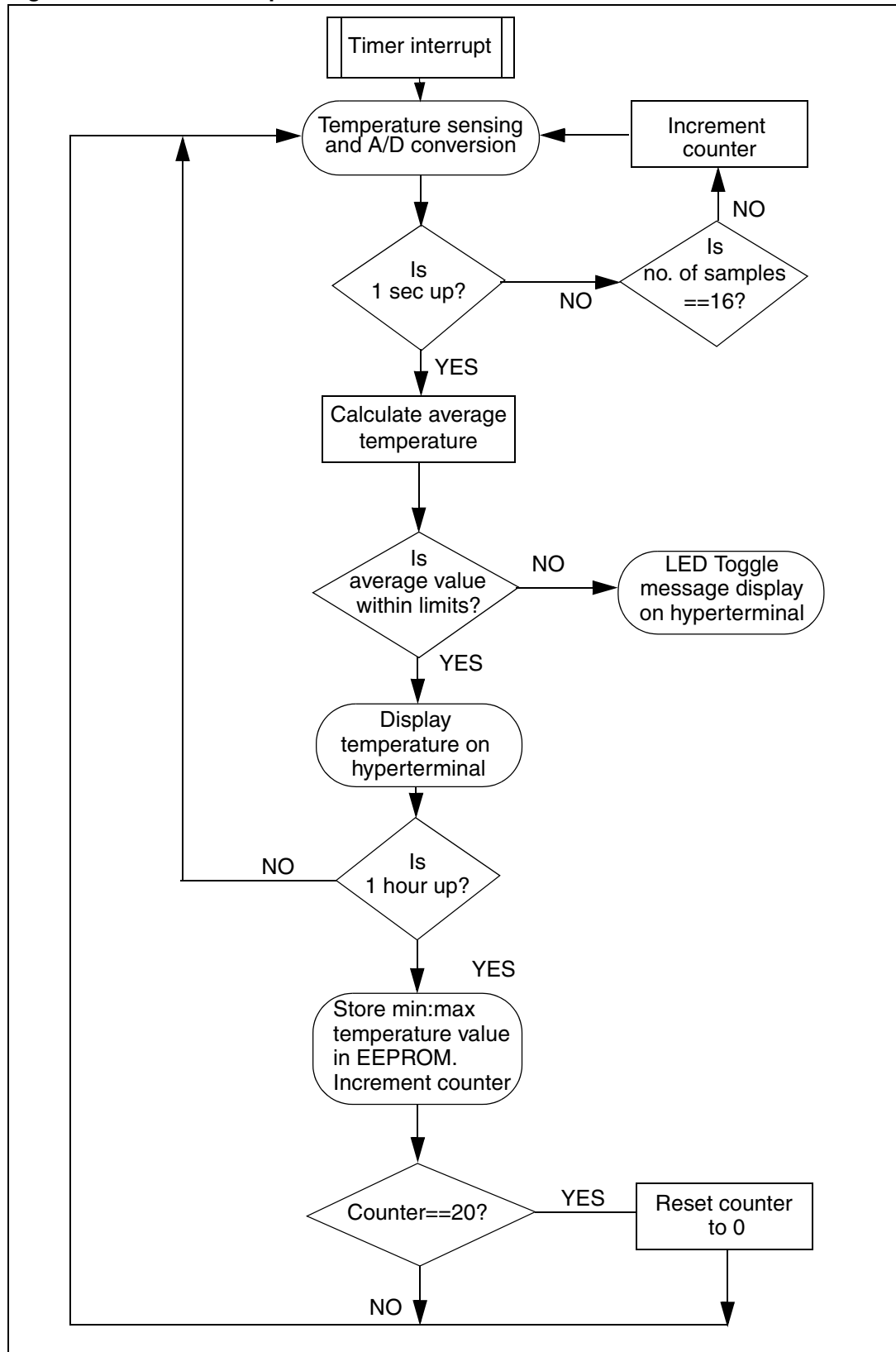

Figure 2. Normal mode flowchart

Figure 3. Timer interrupt flowchart

4 Hardware configuration

This application has been developed and tested by interfacing the ST72325 daughter board with the main REva motherboard. The LM135 temperature sensor, connects to channel 2 of ADC and the temperature is stored in the on-board I^{2}C EEPROM. *Figure 4* shows the hardware setup that you can use to develop this application if you are not using the REva board.

Figure 4.   Hardware setup
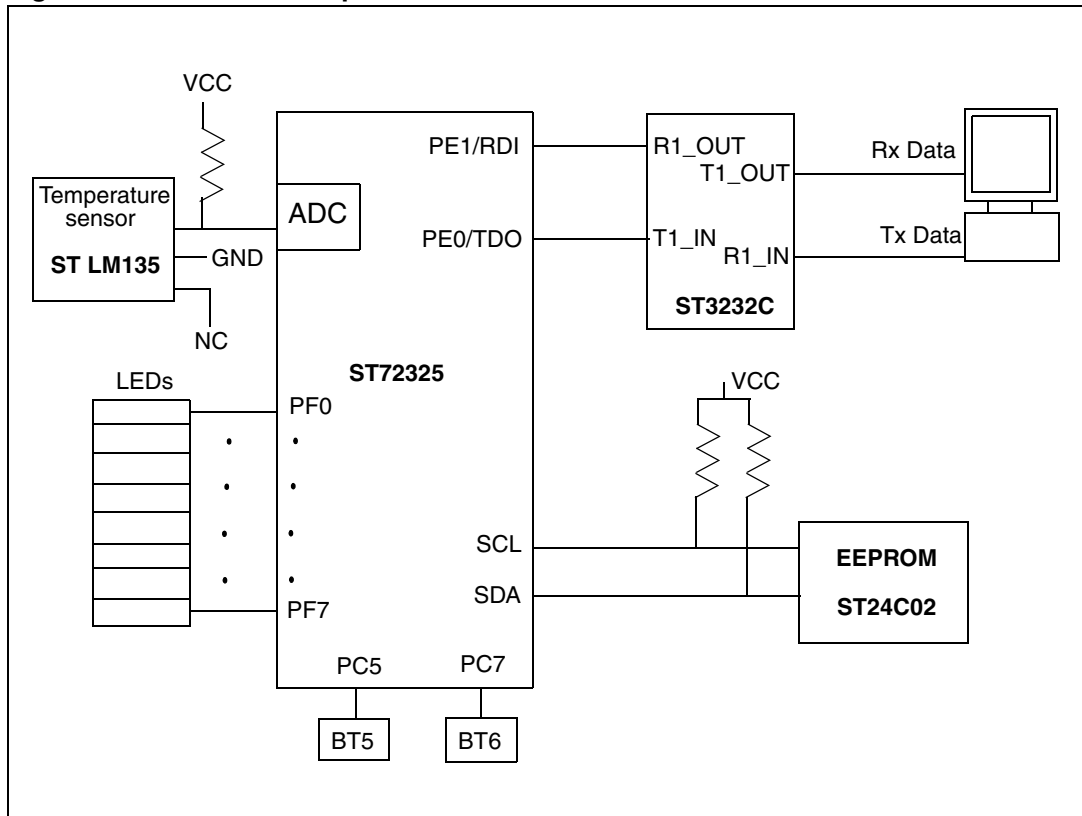
5 Software

All the source files are in 'C' language and application uses ST7 software library functions. Raisonance RIDE is used as the application development tool. The source files are given in a zip file with this application note.

The source files are for guidance only. STMicroelectronics shall not be held liable for any direct, indirect or consequential damages with respect to any claims arising from use of this software.

6 References

1. REva board Hardware Description and User's Guide: *Raisonance Universal Evaluation Board with RLink for Microcontrollers*.
2. LM135 datasheet: *LM135, LM235-LM335,A: Precision Temperature Sensors*.
3. ST72325 datasheet
4. ST M24C02 EEPROM datasheet
5. ST7 Software Manual User Library
6. AN974: Real-time clock with ST7 timer output compare

7 Revision history

Table 1.   Document revision history

Date	Revision	Changes
27-Feb-2005	1	Initial release.
29-Aug-2007	2	Temperature formula changed (see *Section 1*)
07-Sep-2007	3	Minor text changes

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST's terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST'S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZE REPRESENTATIVE OF ST, ST PRODUCTS ARE NOT DESIGNED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS, WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.