



XAPP438 (v1.0) November 29, 2004

CoolRunner-II Low Cost, Low Power Thermometer for Embedded Designs

Summary

This application note describes the implementation of a simple temperature controller in a CoolRunner™-II CPLD device. This thermometer uses a minimum of design resources and on board components (resistors, capacitor, and NTC) to measure, observe, monitor or control temperatures.

CoolRunner-II devices, the latest CPLD family from Xilinx, offer low power and high performance and come in a variety of sizes. A thermometer or temperature controller can be implemented beside additional designs and provides a low cost method of measuring temperatures.

Introduction

Xilinx CoolRunner-II devices provide features, such as Schmidt Trigger inputs, which allow users to implement an easy and simple ADC. This can be used to monitor an analog input value (in this case the temperature) and compute the result in a digital manner for further arrangements. This application consists of four different blocks:

- Analog
- Analog to Digital Converter
- Digital
- Monitoring, computing, displaying, or controlling, depending on the application

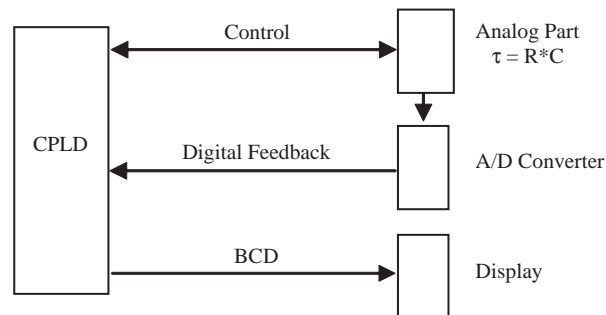


Figure 1: Thermometer Block Diagram

The analog portion of the design consists of a sensor, which changes its resistance depending on the temperature. This characteristic changes in the analog to digital converter to a time based variable, which can be handled by a CPLD. How to implement the last part is up to the application, the elicited temperature can be used for a temperature controller, for an overload and overheat protector or it just can be used to display or observe temperatures. All these features are implemented in the same design. The only limitation is the size of the design.

© 2003 Xilinx, Inc. All rights reserved. All Xilinx trademarks, registered trademarks, patents, and further disclaimers are as listed at <http://www.xilinx.com/legal.htm>. All other trademarks and registered trademarks are the property of their respective owners. All specifications are subject to change without notice.

NOTICE OF DISCLAIMER: Xilinx is providing this design, code, or information "as is." By providing the design, code, or information as one possible implementation of this feature, application, or standard, Xilinx makes no representation that this implementation is free from any claims of infringement. You are responsible for obtaining any rights you may require for your implementation. Xilinx expressly disclaims any warranty whatsoever with respect to the adequacy of the implementation, including but not limited to any warranties or representations that this implementation is free from claims of infringement and any implied warranties of merchantability or fitness for a particular purpose.

Description of Parts

Analog Circuit

Temperature Range and Reliability

The limitation of the temperature range is primarily given by the sensor. Any sensor that changes its resistance depending on temperature can be used for this application. It is up to the user to choose range, accuracy and type of sensor. In this application an NTC is used, as it is a cost efficient solution, and the range is ideal for observing the ambient temperature.

Let's look at the following example:

General Purpose Thermistors (NTC). For example, from RS Components.

- 50 Ohms at 25° C
- Resistance tolerance $\pm 10\%$ at 25° C
- Maximum ambient temperature up to 125° C

For exact details and characteristics see the data sheet: <http://rswww.com>

NTC

An NTC is used as sensor in this application. The analog circuit is based on the following term:

$$\text{Tau (time constant)} = R \text{ (NTC sensor)} * C \text{ (capacitor)}$$

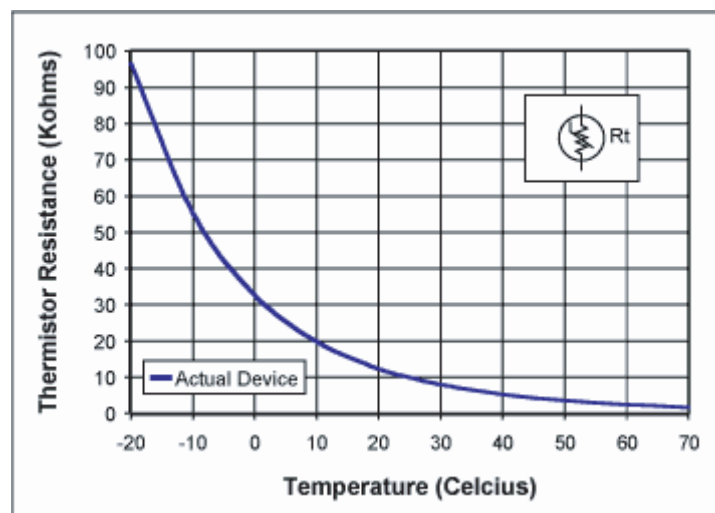


Figure 2: Resistance by Temperature

As shown in the NTC characteristics of [Figure 2](#), there is a coherence between the temperature and the resistance. See the Monitoring / Computing / Displaying for information on how to adjust the NTC curve.

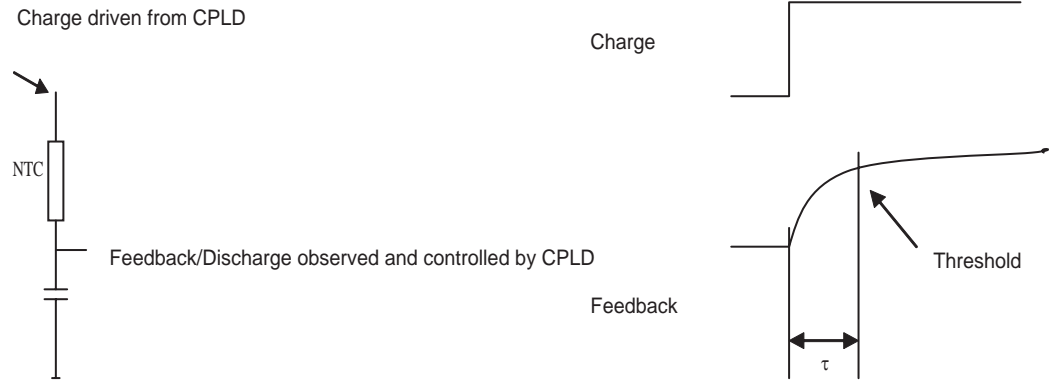


Figure 3: Analog Circuitry

The analog part ([Figure 3](#)) consists of a sensor and a capacitor. The waveforms on the right show the stimulus (Charge) given from the CPLD and the resulting response (Feedback) resulting from the analog part.

Analog to Digital Convertor

The easiest way to ascertain τ (tau) is to set the threshold ([Figure 3](#)). CoolRunner-II provides a feature (Schmitt Trigger) to elicit the threshold.

Digital Circuitry

The digital part consists of a FSM. [Figure 4](#) shows the Waveforms of the required signals.

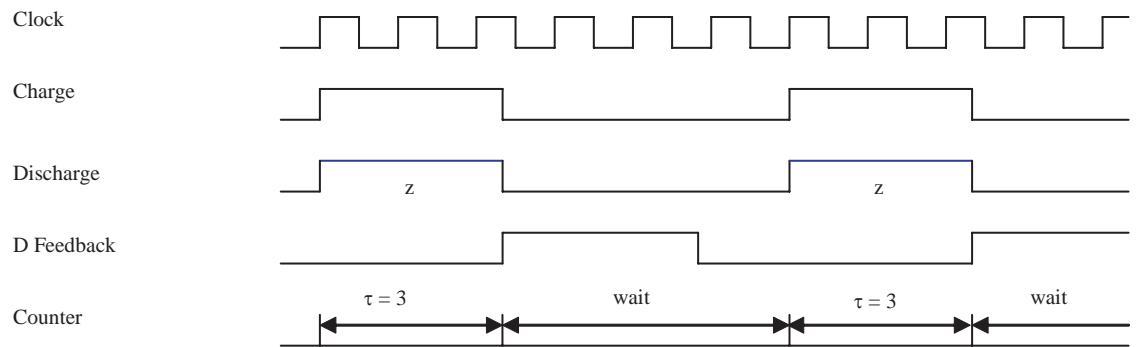


Figure 4: Required Signals

- The CPLD charges (signal Charge) the capacitor through the NTC and counts the clock cycles until the feedback (signal D Feedback) coming from the analog-to-digital converter tells the CPLD that the threshold for τ is reached.
- The number of counted clock cycles correlates to the temperature (depending on accuracy and temperature range, this can be seen as a linear coherence).
- When the threshold is reached the signal Charge changes to '0' and the signal Discharge changes from 'Z' to '0'. So the current will be discharged through the NTC and the signal Discharge into the CPLD (to protect the CPLD IOB, a resistor on the Feedback signal will limit the current).

- The FSM will stay in this state for a certain time (for a low power application this time can increase; for a fast response application this value can decrease, but the minimum should not be less than 5 times τ to make sure the capacitor is completely discharged).
- At this point the FSM begins charging the capacitor with the signal Charge. The signal Discharge changes back to 'Z' at the same time.

Monitoring / Computing / Displaying

This section of the design can be customized for your application. The design can be used for overload protection, or just for observing and/or monitoring the temperature. It is also possible to control the temperature (see "Reliability").

This example shows how to display the temperature on a 7-segment Liquid Crystal Display (LCD):

1. For the most accurate solution create a look-up table (LUT) that correlates clock cycles to temperature (Figure 2). For example:
 - ◆ If τ is between 5 and 7 clock cycles the temperature corresponds (for example) to 15°C or
 - ◆ If τ is between 7 and 9 clock cycles the temperature corresponds (for example) to 14°C...
2. Or, an arithmetic expression can be used to evaluate the temperature.

Note: To display the temperature, a simple "BIN2BCD" decoder can be used in the design.

Note: For an LCD display, a frequency is required to refresh the segments (typically 20 Hz - 50 Hz).

Note: For an LED display, see the CoolRunner-II data sheet concerning information on maximum power consumption.

Drive a LCD Display

To drive an LCD, a clock at the common pin of the display is required. The following example shows how to enable the display:

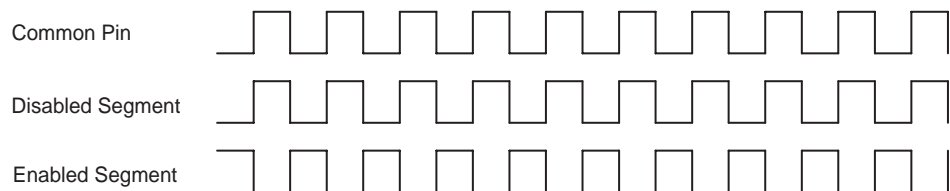


Figure 5: LCD Clocking

Calculation

The following calculations are for example purposes. The values depend on the temperature range, the specific sensor, and the application.

- Choose a value for the NTC:

$$NTC_{min} = V_{cc} / I_{max}$$

where I_{max} is from the data sheet 'IOB'.

Example: $NTC_{min} = 3.3 \text{ Volts} / 200 \text{ mAmps.} = 15.5 \text{ Ohms}$ (Choose 50 Ohms at 25°C)

- Select a minimum value for the resistor that limits the discharge current at the Feedback and Tau signal:

$$R_{discharge_{min}} = V_{DD33} / I_{max}$$

where I_{max} is from the data sheet 'IOB'.

Example: $R_{discharge_{min}} = \text{approx. } 3V / 200 \text{ mAmps.} = 15 \text{ Ohms}$ (Choose 100 Ohms)

- Define the range of t which corresponds to the temperature range ($t_{\max} - t_{\min}$):

$$t_{\text{diff}} = (\text{NTC_range_min} * \text{Capacitance}) - (\text{NTC_range_max} * \text{Capacitance})$$
 Example: $t_{\text{diff}} = (66 \text{ Ohms} * 470 \text{ uF}) - (44 \text{ Ohms} * 470 \text{ uF}) = 100 \text{ ms}$.
- Define a sample frequency depending on the range of t and accuracy (x defines resolution):

$$F_{\text{clock}} = 1/t_{\text{diff}} * x$$
 Example: $F_{\text{clock}} = 1/100 \text{ ms} * 100 = 1 \text{ kHz}$ (resolution of 100 in given range).
- Define the Time to discharge the capacitor (5 is the recommended minimum value):

$$\text{Wait_min} = 5 * (\text{NTC_range_min} * \text{Capacitance})$$
 Example: $\text{Wait_min} = 5 * (66 \text{ Ohms} * 470 \text{ uF}) = 155 \text{ ms}$.
- Define t_{25} :

$$t_{25} = \text{NTC}_{25} * \text{Capacitance}$$
 Example: $\text{Tau}_{25} = 50 \text{ Ohms} * 470 \text{ uF} = 23.5 \text{ ms}$

Why Low Power?

- Current consumption at 25° C:

$$I_{25} = (V_{\text{DD}}/\text{NTC}_{25}) * (t_{25}/\text{Wait_min})$$
 Example: $I_{25} = (3.3 \text{ Volts}/50 \text{ Ohms}) * (23.5 \text{ ms}/155 \text{ ms}) = 10 \text{ uA}$ maximum.

Note: Double the time Wait will halve the current consumption!

Note: Additionally, there is some current used by the voltage divider and the CoolRunner-II itself.

Reliability

The first step to make the system reliable is to make sure the chosen sensor can handle the required temperature range. Further deterioration can be a problem. The influence of ambient temperature (junction temperature of the CoolRunner-II) can carefully be ignored.

Test Design

Note: This design is a functional example. It can be easily optimized by simple changes, such as using vectors instead of integers, or using the "clock divider" feature. This example is written for a better understanding and therefore not "size optimized."

Behavior Simulation

The following simulation shows the detailed function of this application.

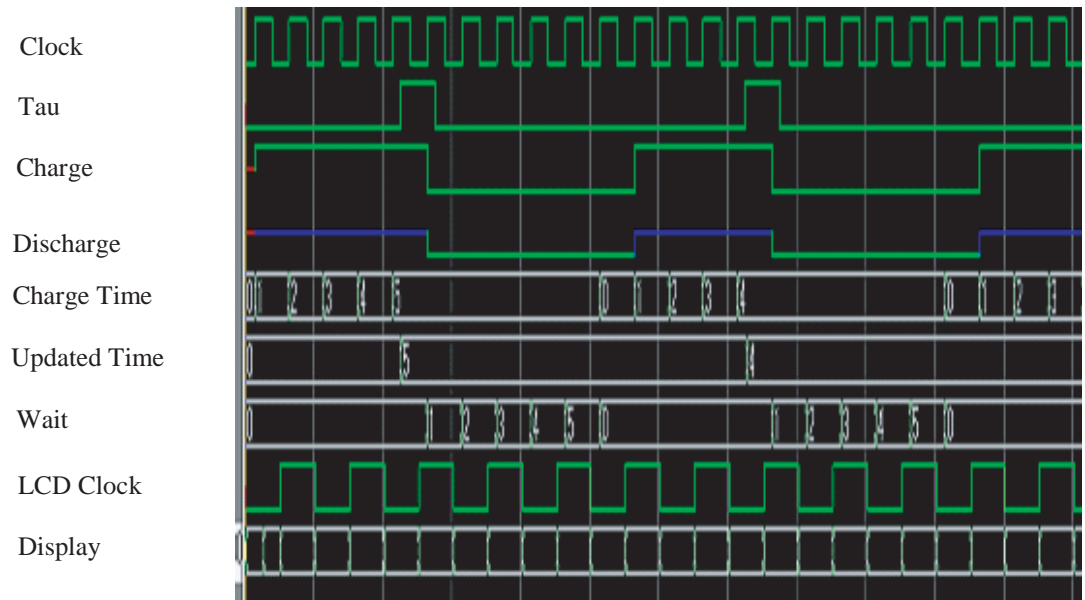


Figure 6: Behavior Simulation

VHDL Code and UCF File

THIRD PARTIES MAY HAVE PATENTS ON THE CODE PROVIDED. BY PROVIDING THIS CODE AS ONE POSSIBLE IMPLEMENTATION OF THIS DESIGN, XILINX IS MAKING NO REPRESENTATION THAT THE PROVIDED IMPLEMENTATION OF THIS DESIGN IS FREE FROM ANY CLAIMS OF INFRINGEMENT BY ANY THIRD PARTY. XILINX EXPRESSLY DISCLAIMS ANY WARRANTY OR CONDITIONS, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, AND XILINX SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE, THE ADEQUACY OF THE IMPLEMENTATION, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OR REPRESENTATION THAT THE IMPLEMENTATION IS FREE FROM CLAIMS OF ANY THIRD PARTY. FURTHERMORE, XILINX IS PROVIDING THIS REFERENCE DESIGN "AS IS" AS A COURTESY TO YOU.

XAPP438 - <http://www.xilinx.com/products/xaw/coolvhdlq.htm>

How to Connect Sensor and LCD Display to CoolRunner-II Design Kit

Figure 7 shows how to connect the Sensor and LCD Display according the .ucf File.

D Connector of CoolRunner-II DesignKit

GND	1	2	VU
VDD33	3	4	TAU
	5	6	CHARGE
	7	8	DISCHARGE
	9	10	OSCILL
	11	12	DISP<0>
	13	14	DISP<1>
	15	16	DISP<2>
	17	18	DISP<3>
	19	20	DISP<4>
	21	22	DISP<5>
	23	24	DISP<6>
	25	26	DISP<7>
	27	28	DISP<8>
	29	30	DISP<9>
	31	32	DISP<10>
	33	34	DISP<11>
	35	36	DISP<12>
	37	38	DISP<13>
	39	40	

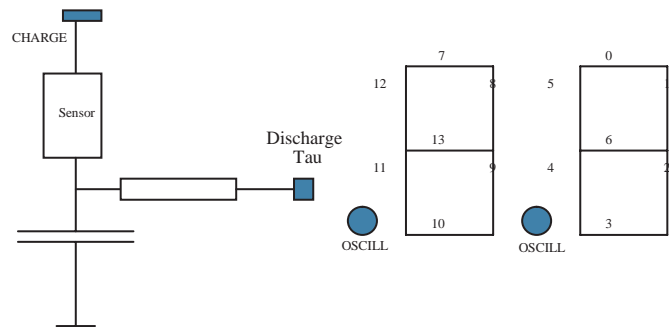


Figure 7: Connecting to CoolRunner-II Design Kit

Summary

CoolRunner-II devices, the latest CPLD family from Xilinx, offer low power and high performance and come in a variety of sizes. A thermometer or temperature controller can be implemented beside additional designs and provides a low cost method of measuring temperatures.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
11/29/04	1.0	Initial Xilinx release.