AN10806

How to use the P89LPC9251 temperature sensor

Rev. 01 — 17 April 2009 Application note

Document information

Info	Content
Keywords	P89LPC9251, Temperature sensor
Abstract	This application note describes how to use the P89LPC9251 temperature sensor. Demo code is also provided.

Revision history

Rev	Date	Description
01	20090417	Initial version.

Contact information

For additional information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

1. Introduction

The P89LPC9251 has two analog-to-digital converter modules: ADC0 and ADC1. ADC1 is an 8-bit, 4-channel multiplexed successive approximation analog-to-digital converter. ADC0 is dedicated for on-chip wide range temperature sensor. The temperature sensor provides temperature sensing capability of −40°C ~ 85°C.

This application note provides example code, which enables the user to get a jump-start into using the on-chip temperature sensor. The code was tested on the KEIL MCB900 evaluation board with a TSSOP28-DIP28 adapter. For more information about MCB900, please refer to: http://www.nxp.com/redirect/keil.com/mcb900.

2. Temperature sensor

2.1 ADC block diagram

A block diagram of the A/D converter is shown in Fig 1.

The temperature sensor (V_{sen}) is measured through Anin03. The other three channels, Anin00, Anin01 and Anin02 are unused. The on-chip temperature sensor is integrated with the ADC0 module. The Temperature sensor and the internal reference voltage $V_{ref(bg)}$ (1.23 V ± 10 %) are multiplexed on the same input channel Anin03. Selecting the temperature sensor or the internal reference voltage is achieved by configuring the TSEL1 and TSEL0 bits in the register TPSCON.

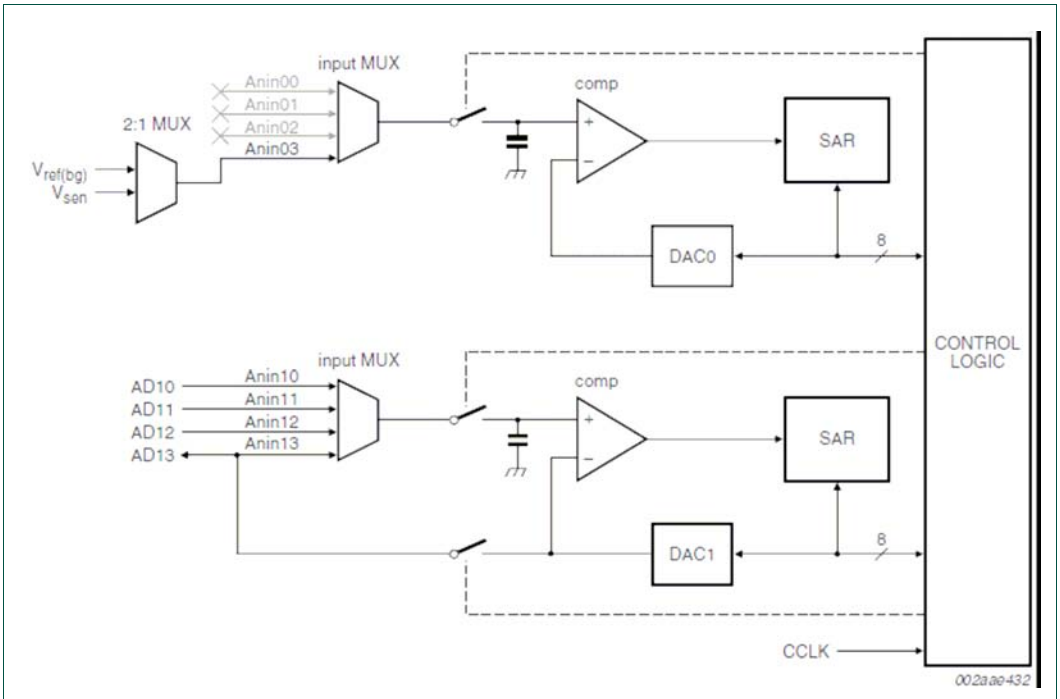

Fig 1. ADC block diagram

2.2 Temperature sensor usage steps

In order to accurately measure a temperature value, it is necessary to sense the supply voltage by measuring the internal reference voltage $V_{ref(bg)}$ first. The Temperature sensor voltage can be calculated using the following formula:

$$V_{sen} = A_{sen} * V_{ref(bg)} / A_{ref(bg)} \quad (1)$$

In the formula (1), $A_{ref(bg)}$ is the A/D converting result of $V_{ref(bg)}$ and A_{sen} is the A/D converting result of V_{sen}.

The Temperature Sensor transfer function is shown in the following formula:

$$V_{sen} = m * Temp + b \text{ (where } m=11.3mV/°C, b=890mV) \quad (2)$$

Temperature Sensor usage steps:

1. Configure TSEL1 and TSEL0 as “01” to select the internal reference voltage.
2. Use ADC to get converting result as A_{ref}.
3. Configure TSEL1 and TSEL0 as “10” to select temperature sensor.
4. Wait at least 200µs to allow the sensor to be stable. Then use the ADC to measure A_{sen}.
5. Calculate V_{sen} with the formula (1).
6. Calculate Temperature with the formula (2).

2.3 Demo introduction

In this demo, the temperature is measured and the calculated temperature result is sent to UART0.

ADC0 is configured as below.

```
1    void ad03_init(void)
2    {
3         // select ADC03
4         ADINS = 0x08;
5         // single conversion mode
6         ADMODA  = 0x01;
7         // configure clock divider
8         ADMODB |= 0x40;
9    }
```

According to the aforementioned temperature sensor usage steps, the internal reference voltage $V_{ref(bg)}$ is measured first.

```
10   ……
11   // measure internal reference voltage
12        TPSCON = 0x04 ;
13        delay (100);
14
15        temp = 0 ;
16
17        // read VREF
18        for(i=0;i<N;i++)
19        {
20             temp += get_ad();
```

```
21          }
22          aref = temp / N ;
23    ……
```

Every time configure TSEL1 and TSEL0 as “10” to choose temperature sensor, settling time of 200µs is required before getting ADC conversion result.

```
24    ……
25    // choose temperature sensor
26          TPSCON = 0x08 ;
27          // Wait sometime to let the sensor work stably
28          delay (200);
29    ……
30
```

The temperature is calculated as below.

```
31    #define VREFBG              12300L                  // Vrefbg * 10000
32    #define VT(at,ar)           ((at)*VREFBG/(ar))
33
34    #define M                   113L                    // M * 10000
35    #define B                   8900L                   // b * 10000
36    #define T(v)                (((v)-B) / M)
37
38    ……
39    // Calculate the real temperature
40    temperature = T(VT(atemp,aref));
41    ……
42
```

2.3.1 Demo setup

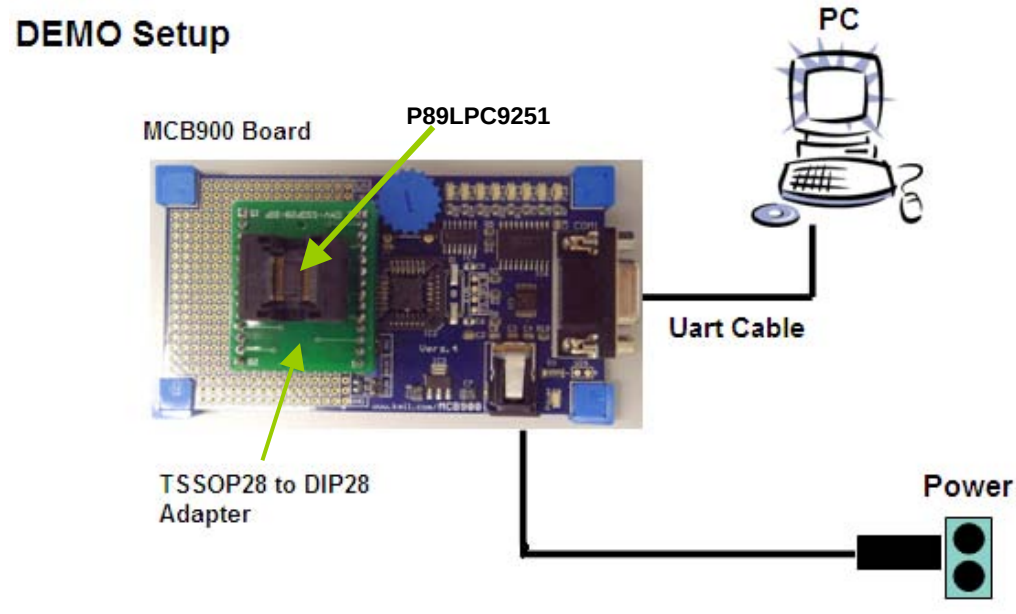

Fig 2. Demo setup

The software of PC terminal here we used is Tera Term. The setting is shown in Fig 3.

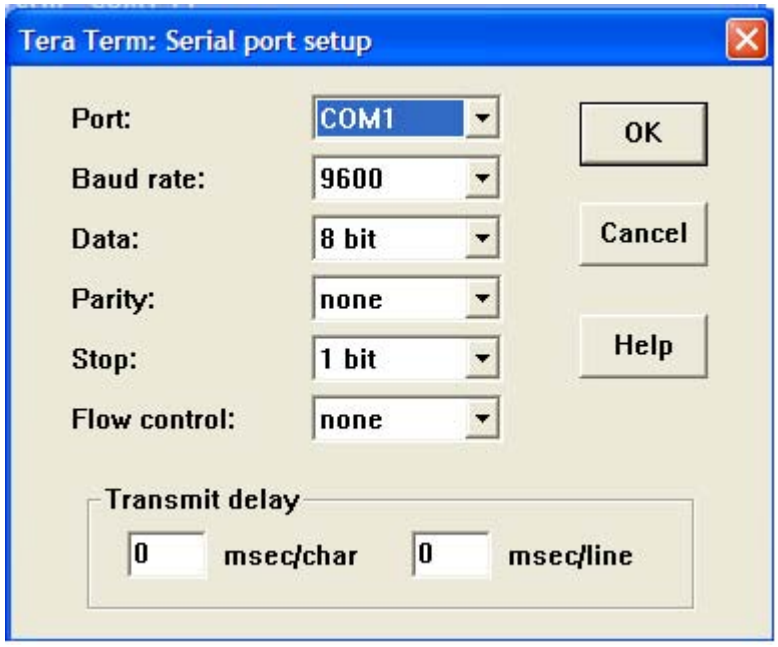

Fig 3. Tera Term setup

2.3.2 Output information using P89LPC9251 – Temperature sensor

The temperature is measured at regular intervals and the calculated result will be sent to UART0.

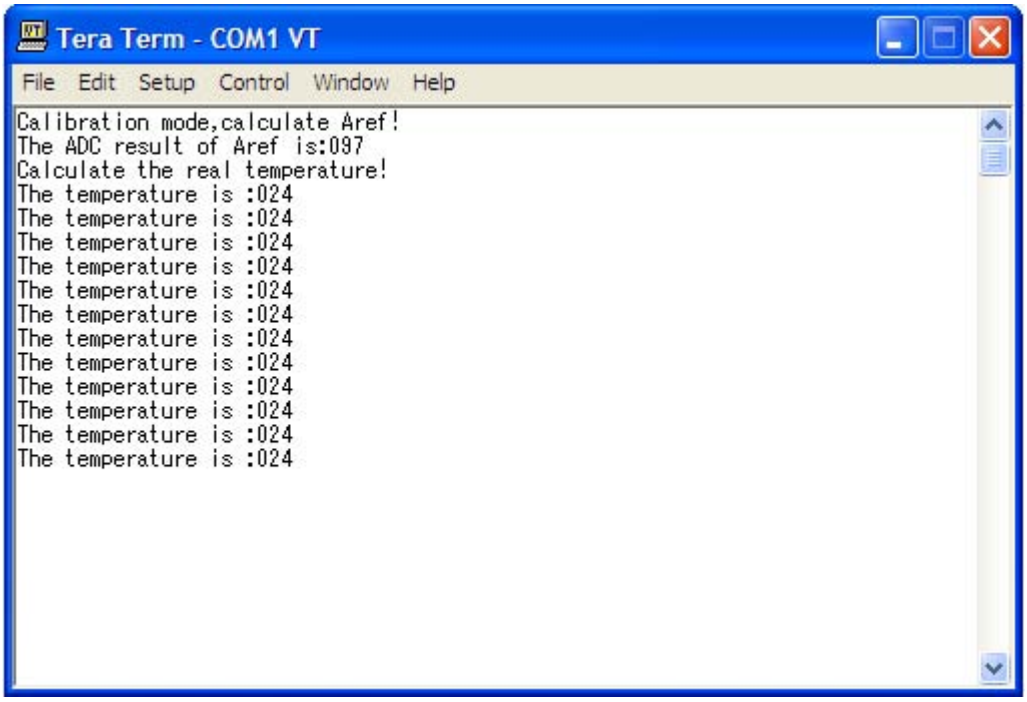

Fig 4. Print information

3. Reference

[1] UM10336 (P89LPC9201/9211/922A1/9241/9251) User Manual – Initial Version

4. Legal information

4.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

4.2 Disclaimers

General — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of a NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is for the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from national authorities.

4.3 Trademarks

Notice: All referenced brands, product names, service names and trademarks are property of their respective owners.

5. Contents

For more information, please visit: http://www.nxp.com
For sales office addresses, email to: salesaddresses@nxp.com

Date of release: 17 April 2009
Document identifier: AN10806_1