AN10768

How to use the P89LPC9351 Programmable Gain Amplifier (PGA)

Rev. 01 — 2 December 2008 **Application note**

Document information

Info	Content
Keywords	P89LPC9351, Programmable Gain Amplifier (PGA)
Abstract	This application note describes how to use the P89LPC9351 Programmable Gain Amplifier (PGA). In addition, demo code is provided.

founded by Philips

Revision history

Rev	Date	Description
01	20081202	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

In the P89LPC9351 an additional PGA is integrated in each ADC module to improve the effective resolution of the ADC. A single channel can be selected for amplification.

The PGA gain value can be programmable to 2, 4, 8 or 16. The PGA outputs go into the 4-input multiplexer of A/D converter, allowing the amplified signal to be converted by the ADC. **For PGA1, its outputs also pass to analog comparators.**

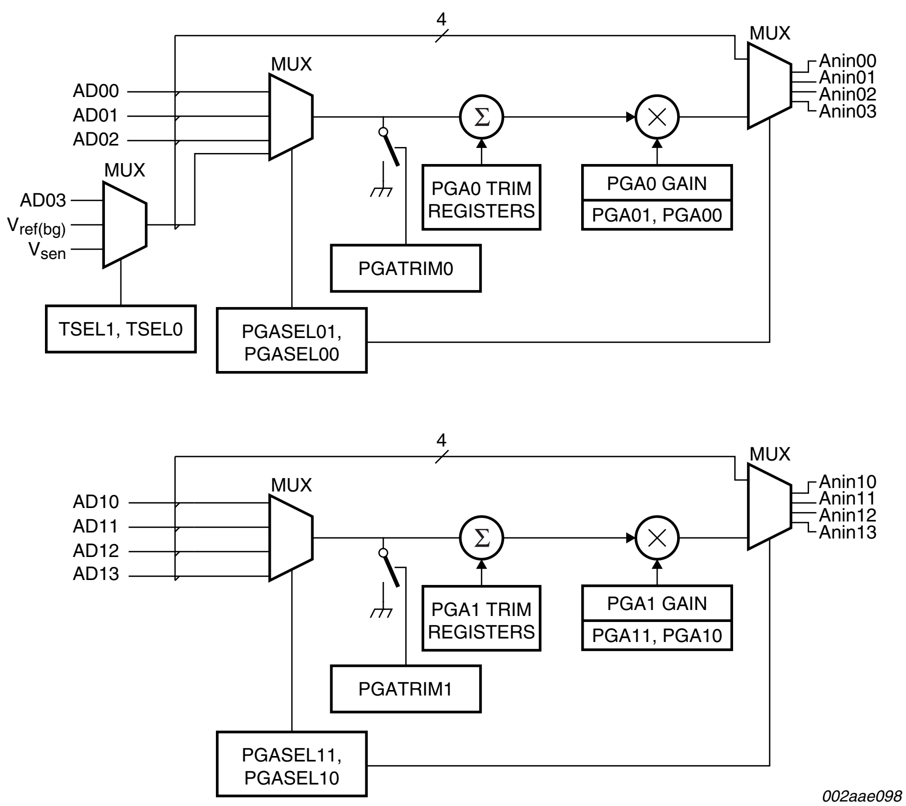

Fig 1. PGA block diagram

On the P89LPC9351, there are extra SFRs added to support the PGA function:

PGACONx — PGAx Control register

PGACONxB — PGAx Control register B

PGAxTRIM8X16X — Trim value for 8x and 16x gain value

PGAxTRIM2X4X — Trim value for 2x and 4x gain value

(x = 0, 1)

With these registers, user can configure the PGA, including enable/disable PGA, set gain of PGA and selection of the appropriate channel.

2. PGA Demo

2.1 Hardware environment

A Keil MCB900 is used as the test board for this AN. The code (.hex) can be downloaded to on chip flash via ISP using Flash Magic software.

2.2 Software environment

Keil uVision3 (PK51 ver 7.50) is the IDE and toolchain for the software.

2.3 Demo introduction

The objective of the demo is to present an example on how to use the PGA for an ADC application.

On the MCB900 board, P0.3 is connected to a potentiometer. The AD12 channel is used to measure the voltage level of the potentiometer. In the demo, PGA mode is switched on and off and different gain values are set to achieve different AD conversion data. The conversion result is sent to a PC Terminal program via the UART0.

In the demo code, the PGA functions are divided into several subroutines which are easy for use. The function setgain() is used to set PGA gain value. The function GetOffset( ) and LoadOffset( ) are defined for PGA calibration.

ADC1 is configured as follows:

```
  ADCON1 = 0x04;                  // write to ADCON1 for enabling ADC1
  ADMODB = 0x40;                  // divide clk by 3 to produce ADC clock
…
  ADCINSEL = 0x40;                // enables AD12 pin for sampling and conversion.
  ADMODA = 0x10;                  // selects single conversion mode (fixed channel)
```

The configuration of PGA1 is shown as below.

```
  setgain(PGA1,gainA);            //set gain value
  PGACON1|=0x40;
  PGACON1&=0xDF;                  // AD12 channel using PGA
  PGACON1|=0x80;                  //Enable PGA
```

For PGA calibration, the PGA input needs to be grounded and only PGA offset voltage is connected. The function GetOffset( ) gives a good example on how to get the offset voltage for different PGA gain values and stores them in the variables. If the calibration values are not required then comment out “#define _ GetOffset” in the code.

```
#define _GetOffset
…
BYTE data PGA1Offset2x,PGA1Offset4x,PGA1Offset8x,PGA1Offset16x;
…
void GetOffset()                  //Get PGA1 offset value(2/4/8/16)
{
  BYTE temp = 0;
  PGACON1B = 1;
  PGACON1|=0x10;

  setgain(PGA1,PGAGain2x);         //set gain value =2
  ADCON1 = 0x05;                  //Start conversion now, ADC1
```

```
    do
    {
        temp = ADCON1;
    }
    while(!(temp & 0x08));
    ADCON1 &= 0x08;             //clear convert-done flag clear boundry interrupt
     PGA1Offset2x=AD1DAT2;
  ……
  ……
  }
```

Get amplified value is implemented in main loop of main(). The result is sent to UART0.

```
        ……
        if(gainA==1)
        {  ADCTemp= AD1DAT2;  }
        else
        {ADCTemp= AD1DAT2-GetOffset(PGA1,gainA);}
         ……
```

2.4 Demo setup

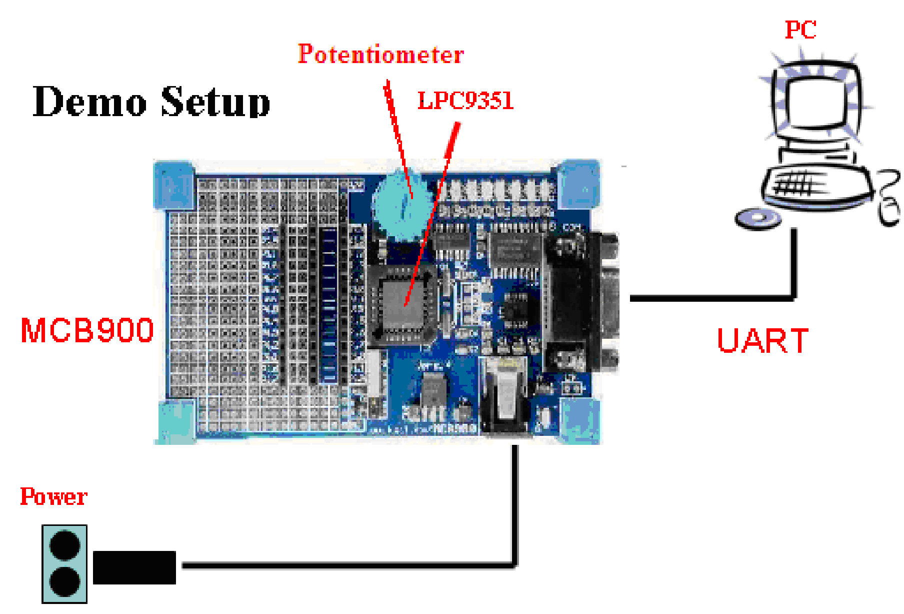

Fig 2. Demo setup

2.5 Output information using P89LPC9351 – different PGA gain value

In the demo, the PGA gain value is sequentially set to x1, x2, x4, x8, x16. Adjust the potentiometer to get the AD conversion result of different PGA gain values on a certain external voltage. All the results will be sent to UART0.

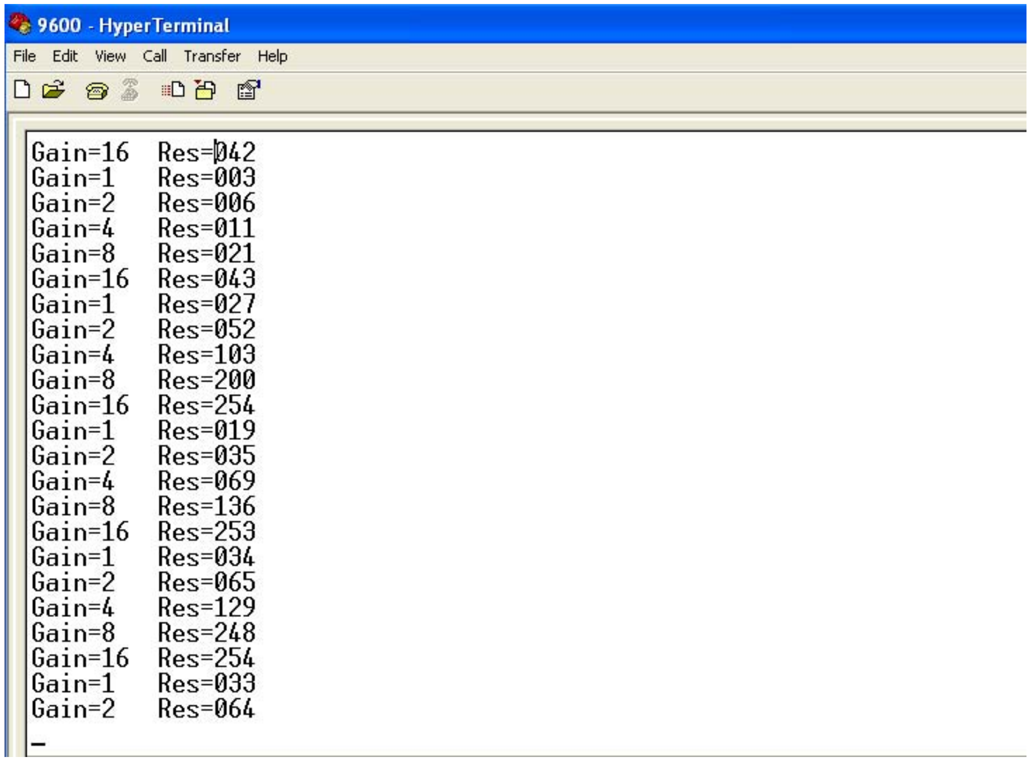

Fig 3. output information for different PGA gain value

3. Reference

[1] P89LPC9351 User manual (UM10308) – Rev. 01

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.

4.3 Trademarks

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

5. Contents

Date of release: 2 December 2008
Document identifier: AN10768_1