Implements ADC Median Filter with NUC240

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	The example code implements ADC sampling noise reduction by using Median Filter.
BSP Version	NUC230_240 Series BSP CMSIS v3.01.001
Hardware	NuEdu-EVB-NUC240 v1.0

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

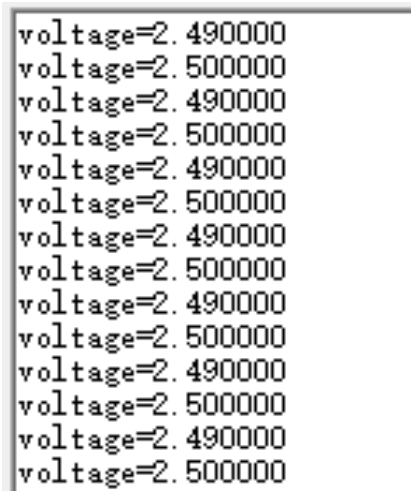
1 Function Description

1.1 Introduction

The sample code implements a simple ADC sampling noise reduction with Median Filter. Do five times ADC sampling and then sort the values. Use Median Filter to takes the number in the middle of the sorted values.

1.2 Demo Result

Connect NuEdu-EVB-NUC240 with NuEdu-Basic01. NUC240 uses ADC channel 0 to sample the voltage value and then sorts the sampling value. Use Median Filter to get the median from the sorted array. Compile the project and enter debug mode, the serial window UART #1 will show the result continuously. User can vary the resistance of the variable resistor on NuEdu-Basic01 to see the changes of the result.



2 Code Description

2.1 Median Filter Algorithm

The algorithm implements Median Filer by sorting ADC sampling table and getting the median from the sorted table.

```
while (1)
{
    /* ADC value range: 0 ~ 4095 */
    g_au32Buffer[g_u32l] = Get_ADC_Knob();

    /* select sort */
    if (g_u32l == 5)
    {
        for (g_u32i = 1; g_u32i <= 3; g_u32i++)
        {
            g_u32k = g_u32i;

            for (g_u32j = g_u32i + 1; g_u32j <= 5; g_u32j++)
            {
                if (g_au32Buffer[g_u32j] < g_au32Buffer[g_u32k])
                    g_u32k = g_u32j;
            }

            if (g_u32k != g_u32i)
            {
                g_au32Buffer[0] = g_au32Buffer[g_u32i];
                g_au32Buffer[g_u32i] = g_au32Buffer[g_u32k];
                g_au32Buffer[g_u32k] = g_au32Buffer[0];
            }
        }

        g_u32l = 0;

        /* ADC value to voltage value */
        d_newVoltage = (3.27 / 4095.0) * g_au32Buffer[3];

        /* round to two decimal places */
        d_newVoltage = (double)((int)(d_newVoltage * 100) / 100.0);
```

```
        if (d_newVoltage != d_oldVoltage)
        {
            printf("voltage=%f\n\r", d_newVoltage);
            d_oldVoltage = d_newVoltage;
        }
    }
    g_u32l++;
}
```

3 Software and Hardware Environment

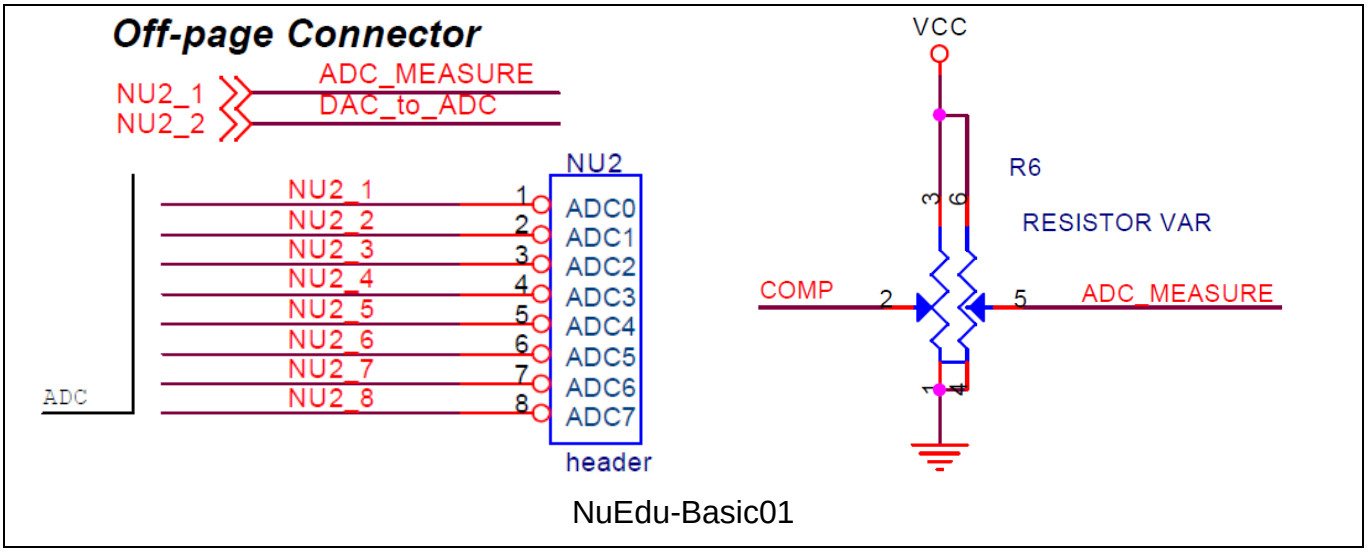
Software Environment

- BSP version
  - NUC230_240 Series BSP CMSIS v3.01.001
- IDE version
  - Keil uVersion 4.74

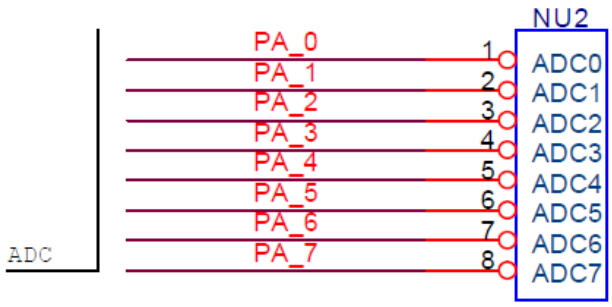
Hardware Environment

- Circuit components
  - NuEdu-EVB-NUC240 V1.0
  - NuEdu-Basic01
- Diagram

  Connect NuEdu-EVB-NUC240 with NuEdu-Basic01. The pin connections between NuEdu-Basic01 and Nu-Bridge are listed below.



NuEdu-Basic01



NuEdu-SDK-NUC240

4 Directory Information

- EC_NUC240_ADC_Median_Filter_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - NuEdu — Library for NuEdu-SDK-NUC240 board
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

5 How to Execute Example Code

1. Browsing into sample code folder by Directory Information (section 4) and double click NUC240_ADC_Median_Filter.uvproj.
2. Enter Keil compile mode
   a. Build
   b. Download
   c. Start/Stop debug session
3. Enter debug mode
   a. Run

6 Revision History

Date	Revision	Description
Nov. 1, 2019	1.00	1. Initially issued.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.