利用PDMA搬運EADC的轉換數據

NuMicro® 32 位系列微控制器範例代碼介紹

文件資訊

代碼簡述	本範例代碼利用 PDMA 搬運 EADC 的轉換數據到 RAM 中存放
BSP 版本	M451 Series BSP CMSIS V3.01.003
開發平台	NuTiny-EVB-M451-LQFP100 V1.3

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 功能介紹

1.1 簡介

本範例代碼利用PDMA搬運EADC的轉換數據到RAM中。範例中會示範如何用軟體觸發ADC，或用中斷觸發ADC，以及EADC如何觸發PDMA開始搬運資料。

1.2 原理

EADC有多種來源可以觸發ADC開始轉換，包括軟體觸發以及中斷觸發。本範例代碼會示範如何用軟體觸發EADC module 4，再用module 4 轉換完成的中斷來觸發module 5, 6, 7 做ADC轉換。

PDMA也有多種來源可以觸發PDMA開始搬運資料，包括EADC轉換完成的中斷。本範例代碼會示範如何設定PDMA去同時搬運 4 個EADC module的轉換數據到RAM buffer中。

1.3 執行結果

執行結果，以及打印資料。

```
System clock rate: 72000000 Hz
+------------------------------------------------------+
|    M453 Series EADC and PDMA Transfer Sample Code    |
+------------------------------------------------------+
PDMA interrupt !!
Conversion result of channel 0: 0xCB9, PDMA transfer data :0xCB9
Conversion result of channel 1: 0x7D9, PDMA transfer data :0x7D9
Conversion result of channel 2: 0x000, PDMA transfer data :0x000
Conversion result of channel 3: 0xFFF, PDMA transfer data :0xFFF
Conversion result of channel 0: 0xBC3, PDMA transfer data :0xBC3
Conversion result of channel 1: 0xA9F, PDMA transfer data :0xA9F
Conversion result of channel 2: 0x000, PDMA transfer data :0x000
Conversion result of channel 3: 0xFFF, PDMA transfer data :0xFFF
Exit EADC sample code
```

EADC得到的轉換值要符合外部的輸入電壓，而且要和PDMA搬運出來的值完全相同才對。

2 代碼介紹

設定 EADC module 4 為軟體觸發，module 5 ~ 7 為中斷觸發：

```
    /* Set the ADC internal sampling time, input mode as single-end and enable the A/D
converter */
    EADC_Open(EADC, EADC_CTL_DIFFEN_SINGLE_END);
    EADC_SetInternalSampleTime(EADC, 6);


    /* Configure the sample 4 module for analog input channel 0 and enable software trigger
source */
    EADC_ConfigSampleModule(EADC, 4, EADC_SOFTWARE_TRIGGER, 0);
    /* Configure the sample 5 module for analog input channel 1 and enable ADINT0 trigger
source */
    EADC_ConfigSampleModule(EADC, 5, EADC_ADINT0_TRIGGER, 1);
    /* Configure the sample 6 module for analog input channel 2 and enable ADINT0 trigger
source */
    EADC_ConfigSampleModule(EADC, 6, EADC_ADINT0_TRIGGER, 2);
    /* Configure the sample 7 module for analog input channel 3 and enable ADINT0 trigger
source */
    EADC_ConfigSampleModule(EADC, 7, EADC_ADINT0_TRIGGER, 3);


    /* Clear the A/D ADINT0 interrupt flag for safe */
    EADC_CLR_INT_FLAG(EADC, EADC_STATUS2_ADIF0_Msk);
    /* Enable the interrupt 0 */
    EADC_ENABLE_INT(EADC, BIT0);
    /* Enable sample module 4 to use interrupt 0 */
    EADC_ENABLE_SAMPLE_MODULE_INT(EADC, 0, (BIT0 << 4));


    /* Clear the A/D ADINT1 interrupt flag for safe */
    EADC_CLR_INT_FLAG(EADC, EADC_STATUS2_ADIF1_Msk);
    /* Enable the interrupt 1 */
    EADC_ENABLE_INT(EADC, BIT1);
    /* Enable sample module 7 to use interrupt 1 */
    EADC_ENABLE_SAMPLE_MODULE_INT(EADC, 1, (BIT0 << 7));


    NVIC_EnableIRQ(ADC01_IRQn);
```

設定 PDMA 去搬運 8 筆 EADC 的轉換數據，每筆數據有 32 bits：

```
    /* Reset PDMA module */
    SYS_ResetModule(PDMA_RST);
    /* Open PDMA channel */
    PDMA_Open(BIT0 << PDMA_CH);
    /* PDMA transfer width is one word (32 bit) and transfer count 8 */
    PDMA_SetTransferCnt(PDMA_CH, PDMA_WIDTH_32, 8);
    /* PDMA transfer data from ADC to buffer g_au32DestArray*/
    PDMA_SetTransferAddr(PDMA_CH, (uint32_t) & (EADC->CURDAT), PDMA_SAR_FIX,
(uint32_t)g_au32DestArray, PDMA_DAR_INC);
    PDMA_SetTransferMode(PDMA_CH, PDMA_ADC_RX, FALSE, 0);


    PDMA_SetBurstType(PDMA_CH, PDMA_REQ_SINGLE, PDMA_BURST_128);


    g_u32IsPDMATestOver = 0;
    PDMA_EnableInt(PDMA_CH, PDMA_INT_TRANS_DONE);
    NVIC_EnableIRQ(PDMA_IRQn);


    PDMA_Trigger(PDMA_CH);


    /* Enable PDMA transfer */
    EADC_ENABLE_PDMA(EADC);
```

用軟體觸發 EADC module 4 開始做 ADC 轉換。Module 4 轉換完成後會發出中斷觸發 module 5, 6, 7 也開始做 ADC 轉換，所以一輪會產生 4 筆轉換數據。

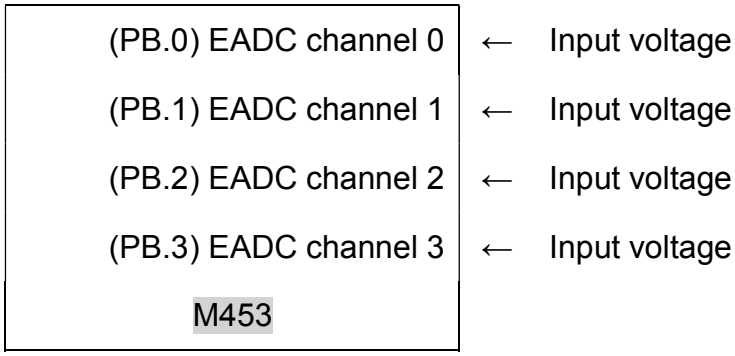
```
    /* Reset the ADC indicator and trigger sample module 4 to start A/D conversion */
    g_u32AdcIntFlag = 0;
    g_u32IsADCChannel7Over = 0;

    /* Start EADC module 4 conversion */
    EADC_START_CONV(EADC, (BIT0 << 4));

    /* Wait EADC interrupt (g_u32AdcIntFlag will be set at IRQ_Handler function) */
    while (g_u32IsADCChannel7Over == 0);

    /* Get the conversion result of the sample module 4 ~ 7 */
    for (u32ModuleCount = 4, u32SampleCount = 0;
            u32ModuleCount < 8;
            u32ModuleCount++, u32SampleCount++)
        g_au32ConversionData[u32SampleCount] = EADC_GET_CONV_DATA(EADC, u32ModuleCount);
```

因為已經設定 PDMA 要搬運 8 筆數據，所以相同的過程要做兩輪，產生 8 筆數據讓 PDMA 搬運，PDMA 才能完成任務。

3 軟體與硬體環境

- **軟體環境**
  - BSP 版本
    - M451 Series BSP CMSIS V3.01.003
  - IDE 版本
    - Keil uVersion 4.74
- **硬體環境**
  - 電路元件
    - NuTiny-EVB-M451-LQFP100 V1.3
  - 示意圖

M453 設定 EADC module 4 使用 channel 0，EADC module 5 使用 channel 1，EADC module 6 使用 channel 2，EADC module 7 使用 channel 3。硬體外部線路連接如下。



4 目錄資訊

- EC_M453_Move_EADC_Data_by_PDMA_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
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

5 如何執行範例程式

1. 根據目錄資訊章節進入 ExampleCode 路徑中的 KEIL 資料夾，雙擊 M453_Move_EADC_Data_by_PDMA.uvproj。
2. 進入編譯模式介面
   a. 編譯
   b. 下載代碼至記憶體
   c. 進入 / 離開除錯模式
3. 進入除錯模式介面
   a. 執行代碼

6 修訂紀錄

Date	Revision	Description
Sep. 16, 2019	1.00	1. 初始發布.

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.