USBD HID To EEPROM

NuMicro® 32 位系列微控制器範例代碼介紹

文件資訊

代碼簡述	實作從 PC 端透過 USB HID 介面控制 NUC240 端的 I2C 傳輸協定以存取 EEPROM 裝置
BSP 版本	NUC230/240 Series BSP v3.01.002
開發平台	NuTiny–EVB–NUC240–LQFP100 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 功能介紹

1.1 簡介

此範例程式演示如何在PC端透過USB HID(Human Interface Device)的連接來控制NUC240的I2C資料傳輸，以進行對EEPROM裝置的存取。

1.2 原理

此範例程式的運作流程，是讓使用者可從PC端下命令，經由USB HID介面的中斷傳輸，將命令傳到NUC240，進而存取EEPROM裝置。而NUC240端則等待USB連線，以USB的端點3 (End Point 3) 來控制傳輸。當端點3發出傳輸中斷，亦即USB連線成功後，NUC240端開始接收命令並執行I2C讀寫的動作，以進行對EEPROM裝置的存取。如圖1-1所示：

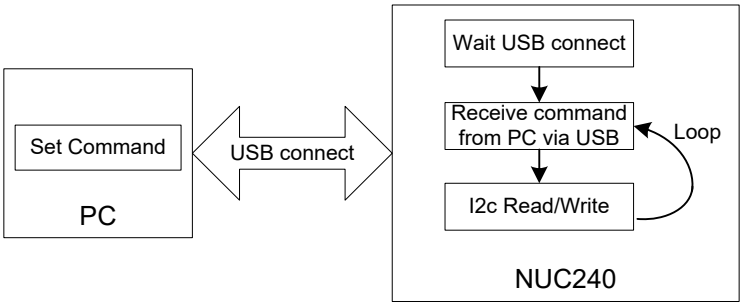

圖 1-1 PC 與 NUC240 的連線與傳輸流程控制

1.3 執行結果

在此範例程式中需使用到兩個windows的工具，分別為HIDTransferTest和HID_I2C_TOOL。以下為取得這兩個open source code的連結：

HIDTransferTest的連結：
https://github.com/OpenNuvoton/NUC472_442BSP/tree/master/SampleCode/StdDriver/USBD_HID_Transfer/WindowsTool/HIDTransferTest

HID_I2C_TOOL的連結：
http://bbs.21ic.com/icview-1103894-1-1.html

將PC與NUC240的USB Port連接後，執行此範例程式，在PC的UART視窗上會出現初始訊息。如圖1-2所示：

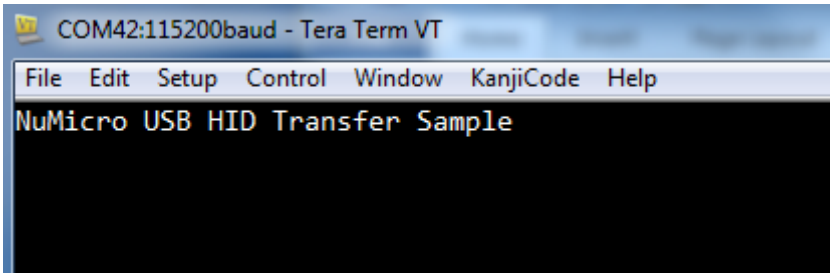

圖 1-2 執行範例程式的初始訊息

接著開啟執行檔 HIDTransferTest.exe，會出現一個視窗顯示連線成功的訊息。如圖1-3所示：

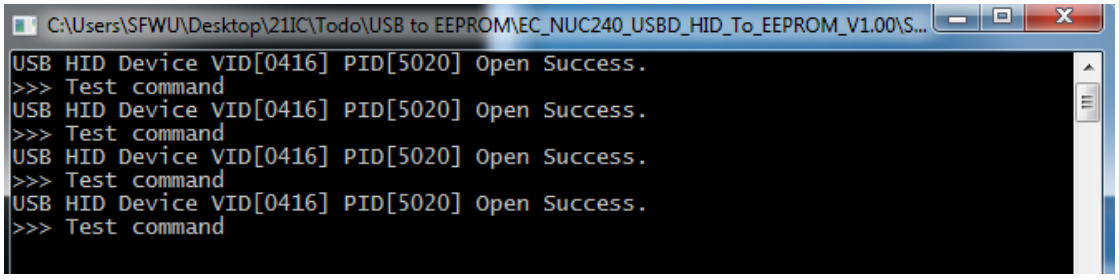

圖 1-3 偵測 USB HID 連線成功的訊息

在開啟執行檔 HIDTransferTest.exe 顯示成功訊息後，在 PC 的 UART 視窗上會顯示出 I2C 傳輸資料的訊息，表示 I2C 已啟動資料傳輸。如圖 1-4 所示：

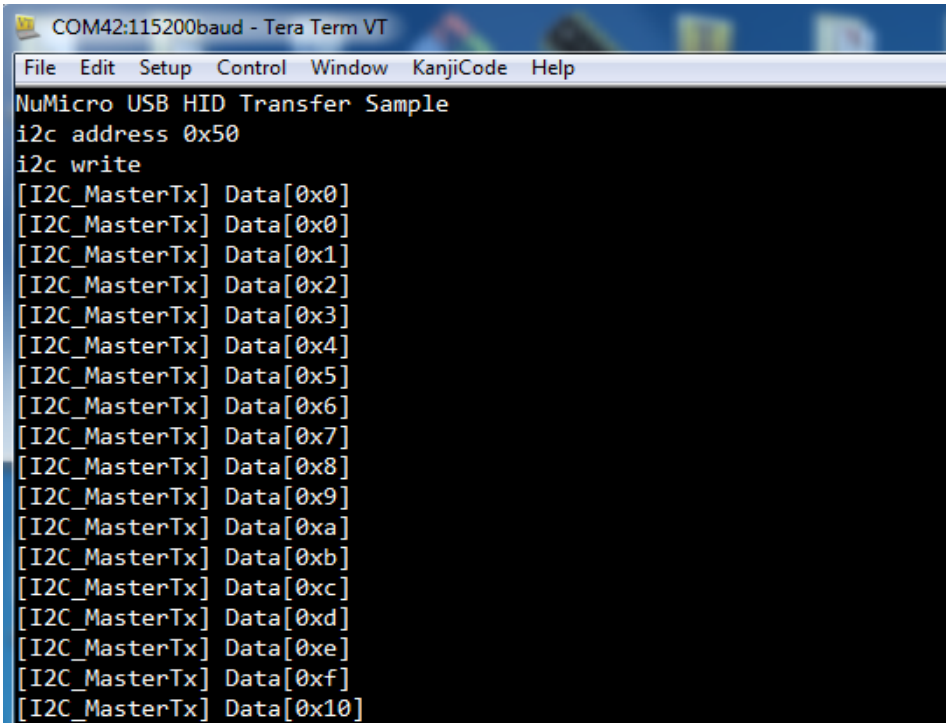

圖 1-4 USB HID 連線後，與 EEPROM 傳輸資料的訊息

接著開啟執行檔 HID_I2C_TOOL.exe，會出現一個視窗如圖 1-5 所示。使用者可在此介面上設定傳輸資料的起始位置(OFFSET 欄位)、長度(LENGTH 欄位)以及傳輸資料的值(Value 欄位)。

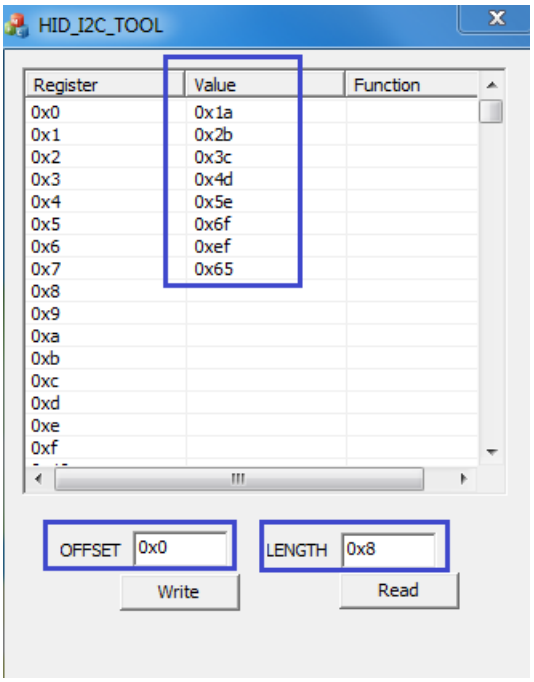

圖 1-5 HID_I2C_TOOL 介面與初始設定

初始設定完成後，可選擇執行讀取(Read按鍵)或寫入(Write按鍵)。若選擇寫入，則按下Write按鍵。如圖1-6所示：

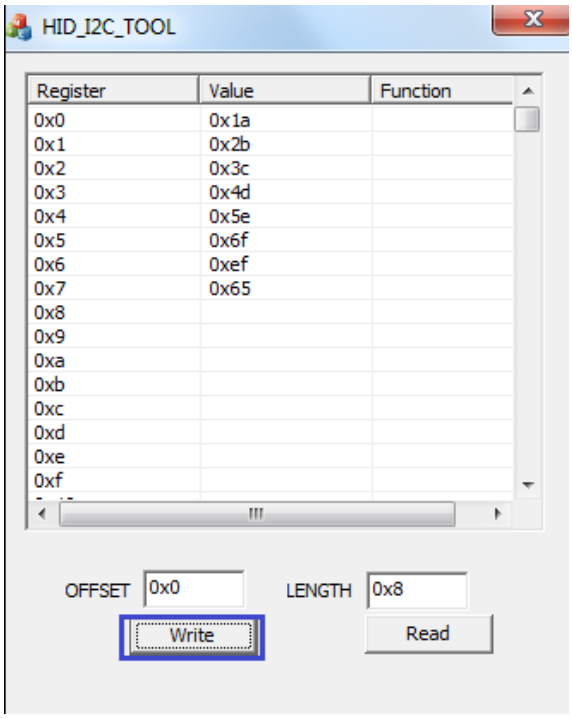

圖 1-6 在 HID_I2C_TOOL 介面上執行 I2C 寫入資料

按下Write按鍵，PC上的UART視窗則會顯示寫入EEPROM裝置的資料。如圖1-7所示：

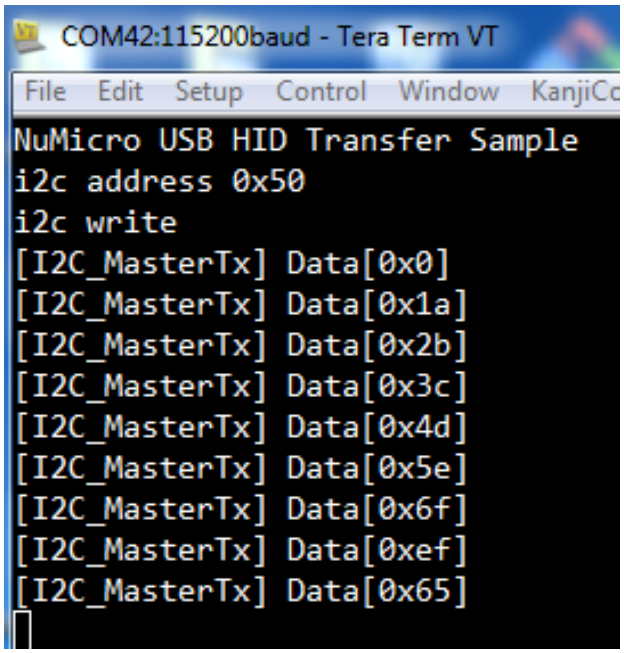

圖 1-7 在 UART 視窗上顯示 I2C 寫入的資料

若選擇讀取，則按下 Read 按鍵。如圖 1-8 所示：

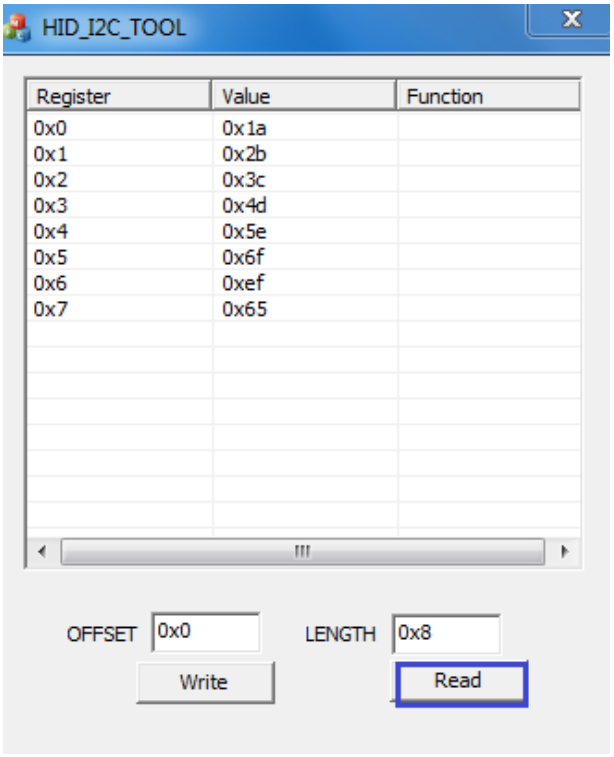

圖 1-8 在 HID_I2C_TOOL 介面上執行 I2C 讀取資料

按下Read按鍵，PC上的UART視窗則會顯示讀取EEPROM裝置的資料。如圖1-9所示：

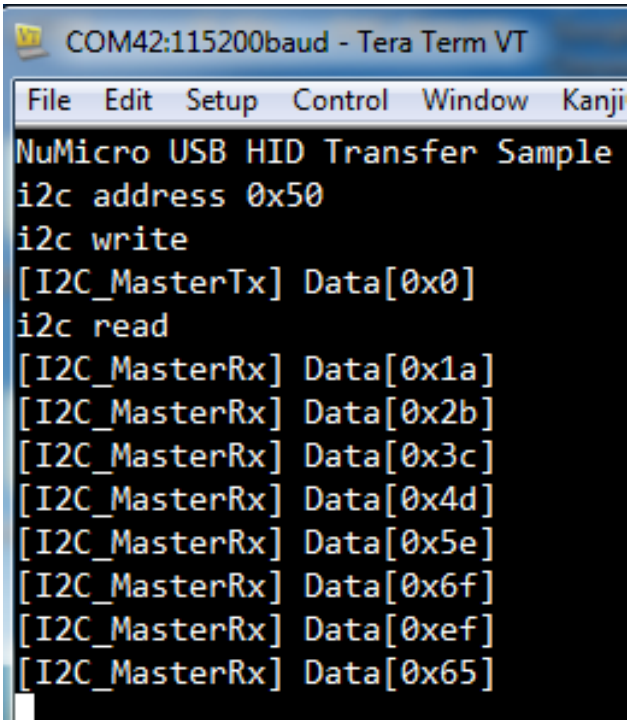

圖 1-9 在 UART 視窗上顯示 I2C 讀出的資料

2 代碼介紹

下方的函式為 EEPROM 的初始設定，其中包含設定 I2C 的 SCL、SDA 的 PIN 腳、I2C 傳輸頻率、以及 I2C 中斷的開啟：

```
/*---------------------------------------------------------------------------*/
/*  EEPROM Initial Function                                                  */
/*---------------------------------------------------------------------------*/
void I2C_EEPROM_Init(void)
{
    /* Init I2C PIN setting */
    I2C_PIN_Init();

    /* Init I2C */
    I2C_Open(I2C_EEPROM, 100000);

    /* Enable I2C interrupt and set corresponding NVIC bit */
    I2C_EEPROM->I2CON |= I2C_I2CON_EI_Msk;
    NVIC_EnableIRQ(I2C_EEPROM_IRQn);
}
```

下方的函式為 Write I2C EEPROM，其流程為設定 I2C Master TX 的 callback 函式後，啟動 I2C 的傳輸：

```
/*---------------------------------------------------------------------------*/
/*  Write I2C EEPROM                                                         */
/*---------------------------------------------------------------------------*/
void I2C_EEPROM_Write(void)
{
    g_u8DataLen = 0;
    g_u8EndFlag = 0;

    /* I2C function to write data to slave */
    s_I2CHandlerFn = (I2C_FUNC)I2C_MasterTx;

    /* I2C as master sends START signal */
    I2C_SET_CONTROL_REG(I2C_EEPROM, I2C_I2CON_STA);

    /* Wait I2C Tx Finish */
    while (g_u8EndFlag == 0);
}
```

下方的函式為 Read I2C EEPROM，其流程為設定 I2C Master RX 的 callback 函式後，啟動 I2C 的傳輸：

```
/*---------------------------------------------------------------------------*/
/*  Read I2C EEPROM                                                          */
/*---------------------------------------------------------------------------*/
uint8_t I2C_EEPROM_Read(void)
{
    g_u8DataLen = 0;
    g_u8EndFlag = 0;

    /* I2C function to write data to slave */
    s_I2CHandlerFn = (I2C_FUNC)I2C_MasterRx;

    /* I2C as master sends START signal */
    I2C_SET_CONTROL_REG(I2C_EEPROM, I2C_I2CON_STA);

    /* Wait I2C Tx Finish */
    while (g_u8EndFlag == 0);

    return g_u8RxData;
}
```

以下則為主程式，其流程為等待USB連線，以NUC240中USB的端點3來控制傳輸，當由端點3發出傳輸中斷後，EEPRON裝置端開始接收I2C傳輸命令(Read，Write)並執行：

```
/*---------------------------------------------------------------------------*/
/*  Main Function                                                            */
/*---------------------------------------------------------------------------*/
int32_t main(void)
{
    uint8_t *pu8PtrOut, *pu8PtrIn;

    /* initial settings ... */
    /*..................... */


    /* Open USB controller */
    USBD_Open(&gsInfo, HID_ClassRequest, NULL);

    /*Init Endpoint configuration for HID */
    HID_Init();
```

```
/* Start USB device */
USBD_Start();

/* Enable USB device interrupt */
NVIC_EnableIRQ(USBD_IRQn);

while(1)
{
    if(g_u8EP3Ready==1)
    {
        g_u8EP3Ready=0;

        /* Interrupt OUT */
        pu8PtrOut = (uint8_t *)(USBD_BUF_BASE + USBD_GET_EP_BUF_ADDR(EP3));
        pu8PtrIn = (uint8_t *)(USBD_BUF_BASE + USBD_GET_EP_BUF_ADDR(EP2));

        /* Get command form HID and copy to command queue */
        USBD_MemCopy((uint8_t *)&g_tCmd, pu8PtrOut, 64);

        /* Set EEPROM device address */
        if(g_tCmd.u8Command==I2C_SET_ADDRESS)
        {
            g_u8DeviceAddr=g_tCmd.u8Size;
            printf("i2c address 0x%x\n\r",g_tCmd.u8Size);
        }

        /* Write data to EEPROM device */
        if(g_tCmd.u8Command==I2C_WRITE)
        {
            printf("i2c write\n\r");
            I2C_EEPROM_Write();
        }

        /* Read data from EEPROM device */
        if(g_tCmd.u8Command==I2C_READ)
        {
            printf("i2c read\n\r");
            I2C_EEPROM_Read();
        }
        g_tCmd.u8PID+=1;
```

```
            /* Write command to HID */
            USBD_MemCopy(pu8PtrIn, (void *)&g_tCmd, 64);
            USBD_SET_PAYLOAD_LEN(EP3, EP3_MAX_PKT_SIZE);
            USBD_SET_PAYLOAD_LEN(EP2, EP2_MAX_PKT_SIZE);
        }
    }
}
```

3 軟體與硬體環境

● 軟體環境

- ■ BSP 版本
  - ◆ NUC230/240 Series BSP CMSIS v3.01.002
- ■ IDE 版本
  - ◆ Keil uVersion 5.28

● 硬體環境

- ■ 電路元件
  - ◆ NuTiny–EVB–NUC240–LQFP100 V1.0
- ■ 示意圖
  - ◆ 將 UART TX（PB.1）pin 腳連接到 PC UART RX，以顯示示例代碼的執行結果於 PC 上，並將 NUC240 USB 藉由 mini USB 傳輸線連接至 PC。
  - ◆ 將 NUC240 的 I2C1_SDA 和 I2C1_SCL 分別與 EEPROM 的 SDA 和 SCL 連接。

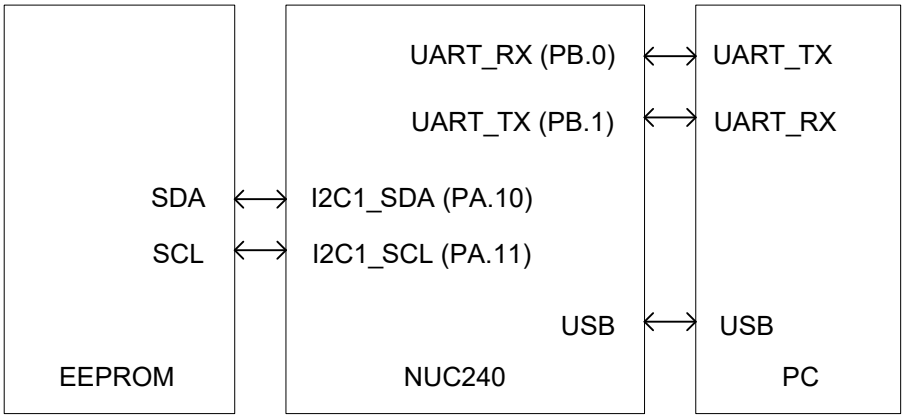

圖 3-1 NUC240 和 PC、EEPROM連接圖

4 目錄資訊

- EC_NUC240_USBD_HID_To_EEPROM_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 USBD_HID_To_EEPROM.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.