NUC240使用GPIO模擬I²C存取EEPROM

NuMicro® 32 位系列微控制器範例代碼介紹

文件資訊

代碼簡述	本範例代碼使用NUC240的GPIO模擬I²C存取EEPROM
BSP 版本	NUC230_240 Series BSP CMSIS v3.01.001
開發平台	NuEdu-EVB-NUC240 v1.0

1 功能介紹

1.1 簡介

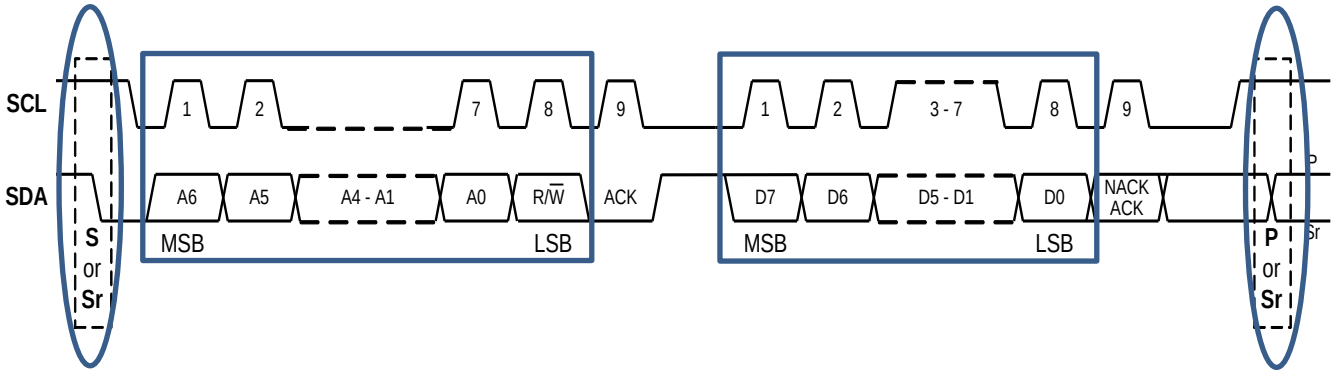
產品開發時，難免會碰到硬體I²C組數不夠用，此時便可以用GPIO產生I²C的波形。本範例用軟體操作GPIO實現I²C的功能，外部接一個EEPROM，程式控制去寫讀EEPROM。本範例提供了2個API，讓使用者呼叫：

API	功能
void I2C_GPIO_Write_Byte(uint16_t u16Addr, uint8_t u8Data)	送出字元
void I2C_GPIO_Read_Byte(uint16_t u16Addr, uint8_t* pu8Data)	接收字元

1.2 原理

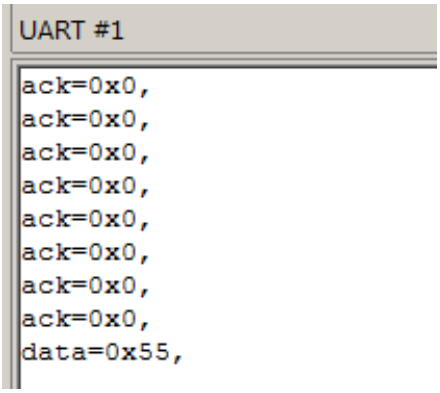
I²C協定分為四個步驟：

1. 產生 START 或 Repeated START 訊號
2. 傳輸 Slave 地址與讀/寫 bit
3. 資料傳輸
4. 產生 STOP 訊號



1.3 執行結果

NUC240開發版的PA11(SCK)連接至NuEdu-Basic01的NU6_10；PA10(SDA)連接至NuEdu-Basic01的NU6_9。編譯專案並進入除錯模式執行程式， EEPROM透過I^{2}C回應的ACK與讀寫數值會列印在Serial Window UART #1。



2 代碼介紹

2.1 軟體 I²C 初始化

I2C_Init()此函式主要工作為軟體 I²C的初始化，即將GPIO(PA.10與PA.11)腳位設定成Open-drain模式，且在空閒時I/O為高電位。

```
void I2C_Init(void)
{
       /* Configure PA10 as open-drain mode */
       GPIO_SetMode(PA, BIT10, GPIO_PMD_OPEN_DRAIN);

       /* Configure PA11 as open-drain mode */
       GPIO_SetMode(PA, BIT11, GPIO_PMD_OPEN_DRAIN);

       /* Control PA11 output status */
       I2C_CLK = 1;

       /* Control PA10 output status */
       I2C_DAT = 1;
}
```

2.2 I²C 字元傳送

I2C_GPIO_Write_Byte()此函式主要工作為傳送I²C字元，需指定EEPROM的位址與欲傳輸的字元。EEPROM協定請參考EEPROM產品規格手冊。

```
void I2C_GPIO_Write_Byte(uint16_t u16Addr, uint8_t u8Data)
{
       uint8_t u8AddrH, u8AddrL;

       u8AddrH = u16Addr >> 8;
       u8AddrL = (uint8_t)u16Addr;

       /* Send Start bit to I2C EEPROM */
       I2C_Start();
```

```
        /* Send control byte to I2C EEPROM */
        printf("ack=0x%x,\n\r",I2C_Write(I2C_ADDRESS_W));

        /* Send I2C EEPROM's High Byte Address */
        printf("ack=0x%x,\n\r",I2C_Write(u8AddrH));

        /* Send I2C EEPROM's Low Byte Address */
        printf("ack=0x%x,\n\r",I2C_Write(u8AddrL));

        /* Send data byte to I2C EEPROM */
        printf("ack=0x%x,\n\r",I2C_Write(u8Data));

        /* Send Stop bit to I2C EEPROM */
        I2C_Stop();
}
```

2.3 I²C 字元接收

I2C_GPIO_Read_Byte此函式主要工作為I²C字元，需指定EEPROM的位址。EEPROM協定請參考EEPROM產品規格手冊。

```
void I2C_GPIO_Read_Byte(uint16_t u16Addr, uint8_t* pu8Data)
{
        uint8_t u8AddrH, u8AddrL;

        u8AddrH = u16Addr >> 8;
        u8AddrL = (uint8_t)u16Addr;

        /* Send Start bit to I2C EEPROM */
        I2C_Start();

        /* Send control byte to I2C EEPROM */
        printf("ack=0x%x,\n\r",I2C_Write(I2C_ADDRESS_W));

        /* Send I2C EEPROM's High Byte Address */
        printf("ack=0x%x,\n\r",I2C_Write(u8AddrH));

        /* Send I2C EEPROM's Low Byte Address */
        printf("ack=0x%x,\n\r",I2C_Write(u8AddrL));
```

```
    /* Send Start bit to I2C EEPROM */
    I2C_Start();

    /* Send control byte to I2C EEPROM */
    printf("ack=0x%x,\n\r",I2C_Write(I2C_ADDRESS_R));

    /* Read data byte from EEPROM */
    *pu8Data = I2C_Read(NO_ACK);

    /* Send Stop bit to I2C EEPROM */
    I2C_Stop();
}
```

3 軟體與硬體環境

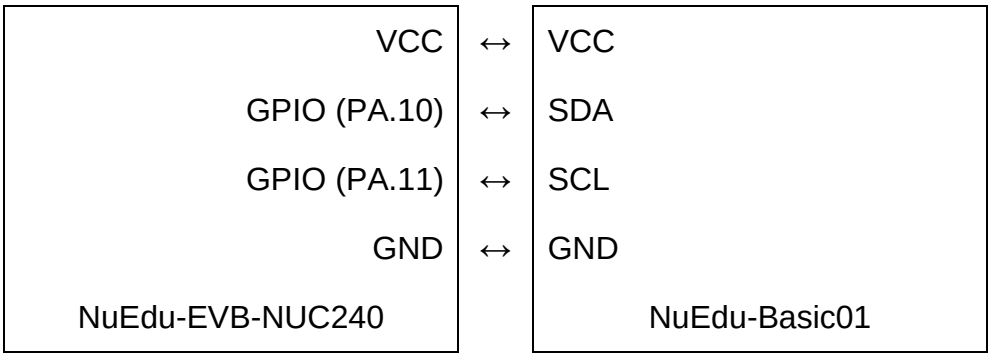
● 軟體環境

- ■ BSP 版本
  - ◆ NUC230_240 Series BSP CMSIS v3.01.001
- ■ IDE 版本
  - ◆ Keil uVersion 4.74

● 硬體環境

- ■ 電路元件
  - ◆ NuEdu-EVB-NUC240 V1.0
  - ◆ NuEdu-Basic01 V2.1
- ■ 示意圖

NuEdu-EVB-NUC240 使用 PA10 與 PA11 腳位模擬 I^2C 的 SDA 與 SCL 波型來控制在 NuEdu-Basic01 上的 EEPROM。



4 目錄資訊

- EC_NUC240_GPIO_I2C_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 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 NUC240_GPIO_I2C.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
Nov. 1, 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.