使用NUC240控制LCD1602顯示屏

NuMicro® 32 位系列微控制器範例代碼介紹

文件資訊

代碼簡述	本範例代碼基於NUC240平台，使用GPIO功能實現LCD1602顯示屏的8位模式控制
BSP 版本	NUC230_240 Series BSP CMSIS V3.01.001
開發平台	NuTiny-NUC240V

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

本範例代碼基於NUC240平台，使用GPIO功能實現LCD1602顯示屏的8位模式控制。SMC1602是一個LCD顯示屏，型號1602的意思是每行顯示16個字符，共可顯示兩行，驅動的形式上有8位元及4位元兩種可供選擇，顯示屏詳細規格與指令內容請參考SMC1602的Datasheet。

1.2 SMC1602 引腳說明

引腳編號	符號	說明	引腳編號	符號	說明
1	VSS	電源地	**9**	D3	LCD DB2
2	VDD	電源正極	**10**	D4	LCD DB3
3	VL	液晶顯示偏壓信號	**11**	D5	LCD DB4
4	RS	暫存器選擇 RS = 0 : Cmd register RS = 1 : Data register	**12**	D6	LCD DB5
5	R/W	LCD 讀寫選擇 (Read/Write) R/W = 0 : Write R/W = 1 : Read	**13**	D7	LCD DB6
6	E	LCD 使能(Enable) E = 0 : LCD disable E = 1 : LCD enable	**14**	D8	LCD DB7
7	D0	LCD DB0 (Data Bus)	**15**	BLA	背光源正極
8	D1	LCD DB1	**16**	BLK	背光源負極

2 代碼介紹

2.1 函式描述

程式碼中的註解，說明了GPIO的設定、巨集的定義，詳細內容請參考註解。

```
/*---------------------------------------------------------------------------------------*/
/* Function Description                                                                  */
/*---------------------------------------  ----------------------------------------------*/
// GPIOE.0 -> LCD RS(Register Select)
//                RS = 0 : Cmd register
//                RS = 1 : Data register
// GPIOE.1 -> LCD R/W(Read/Write)
//                R/W = 0 : Write
//                R/W = 1 : Read
// GPIOE.2 -> LCD E(Enable)
//                E = 0 : LCD disable
//                E = 1 : LCD enable
// GPIOA.0 -> LCD DB0(Data Bus)
// GPIOA.1 -> LCD DB1
// GPIOA.2 -> LCD DB2
// GPIOA.3 -> LCD DB3
// GPIOA.4 -> LCD DB4
// GPIOA.5 -> LCD DB5
// GPIOA.6 -> LCD DB6
// GPIOA.7 -> LCD DB7

#define LCD_DATA    PA->DOUT // GPIOA data out
#define LCD_RS_SET  PB0=1 // Data register
#define LCD_RS_CLR  PB0=0 // Cmd register
#define LCD_RW_SET  PB1=1 // Write
#define LCD_RW_CLR  PB1=0 // Read
#define LCD_E_SET   PB2=1 // LCD disable
#define LCD_E_CLR   PB2=0 // LCD enable
```

2.2 寫入資料至 LCD 模組

使用此函數能夠將欲顯示的資料寫入LCD模組，詳細內容請參考下列註解。

```
void WriteDataLcd(unsigned char wdata)
{
    LCD_DATA = wdata;       // Set data out
    LCD_RS_SET;             // Select Data register
    LCD_RW_CLR;             // LCD1602 Read
    LCD_E_CLR;              // LCD disable
    CLK_SysTickDelay(1000); // Delay 1000 us

    LCD_E_SET;              // LCD enable
}
```

2.3 寫入命令至 LCD 模組

使用此函數能夠將欲顯示的命令寫入LCD模組，詳細內容請參考下列註解。

```
void WriteCommandLcd(unsigned char wdata)
{
    LCD_DATA = wdata;       // Set data out
    LCD_RS_CLR;             // Select Cmd register
    LCD_RW_CLR;             // LCD1602 Read
    LCD_E_CLR;              // LCD disable
    CLK_SysTickDelay(1000); // Delay 1000 us

    LCD_E_SET;              // LCD enable
}
```

2.4 初始化 LCD 模組

本函數發送了簡單的命令對LCD模組進行初始化，詳細內容請參考下列註解。

```
void lcd_init(void)
{
    LCD_DATA = 0;
    WriteCommandLcd(0x38);  //DL=8-bit, N=2-line, F=5x8 dots
    CLK_SysTickDelay(50000);
    WriteCommandLcd(0x08);  //D=0 Display Off, C=0 Cursor Off, B=0 Blinks Off.
    WriteCommandLcd(0x06);  //I/D=1 Increment, S=0 Display does not shift.
    WriteCommandLcd(0x0C);  //D=0 Display On, C=0 Cursor Off, B=0 Blinks Off.
}
```

2.5 LCD 模組寫入行選擇

依照欲選擇的寫入行，發送指令給LCD進行設定。

```
void display_xy(unsigned char x, unsigned char y)
{
    x += 0x80;              // For 1st line
    if (y == 1)
        x += 0x40;          // For 2nd line
    WriteCommandLcd(x);
}
```

2.6 寫入字元

寫入欲顯示的字元至LCD模組。

```
void display_char(unsigned char x, unsigned char y, unsigned char dat)
{
    display_xy(x, y);  // Select line 1 or line 2
    WriteDataLcd(dat); // Write the data to LCD
}
```

2.7 寫入字串

寫入欲顯示的字串至LCD模組。

```
void display_string(unsigned  char x, unsigned  char y, unsigned  char *string)
{
    display_xy(x, y);       // Select line 1 or line 2

    while (*string)         // If there is not end of string, sending data continuously.
    {
        WriteDataLcd(*string);  // Write data to LCD
        string++;               // Move the pointer to next character
    }
}
```

2.8 主迴圈

設定GPIOA、初始化LCD模組，輸入兩個字串進行顯示。

```
unsigned char  LcdBuf1[] = {"Welcome!"};
unsigned char  LcdBuf2[] = {"NuMicro M0"};
```

```
int main(void)
{
    GPIO_SetMode(PA, 0xFFFF, GPIO_PMD_OUTPUT);  // Set PA0 ~ PA15 to output mode.
    lcd_init();                                 // Initial LED module.
    display_string(2, 0, LcdBuf1);              // Display string 1 "Welcome!"
    display_string(0, 1, LcdBuf2);              // Display string 2 "NuMicro M0"

    while (1);
}
```

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 5.26
- **硬體環境**
  - 電路元件
    - NuTiny-NUC240V * 1
    - 牛橋 Nu-Bridge * 1
    - SMC1602 * 1

■ 接線圖

1. 將NuTiny-NUC240V的UART_RX(PB.0)、UART_TX (PB.1)連接至Nu-Bridge，打印訊息。設置終端機的COM Port與Baudrate，COM Port的編號可在裝置管理員中找到「NuBridge Virtual Com Port (COMX)」，Baudrate設置為115200。

2. 將SMC1602依照以下連接方式，與NUC240連接。

UART RX	↔	UART_TX (PB.1)
UART TX	↔	UART_RX (PB.0)
Nu-Bridge		
RS	↔	PE.0
R/W	↔	PE.1
E	↔	PE.2
D0	↔	PA.0
D1	↔	PA.1
D2	↔	PA.2
D3	↔	PA.3
D4	↔	PA.4
D5	↔	PA.5
D6	↔	PA.6
D7	↔	PA.7
SMC1602		NUC240

4 目錄資訊

- EC_NUC240_LCD1602_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 EC_NUC240_LCD1602_V1.00\SampleCode\ExampleCode\1602 LCD 8BIT，雙擊路徑中 1602 LCD 8BIT.uvproj 開啟專案。
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. 11, 2019	1.00	1. 初版

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.