LCD1602 Panel Controlled on NUC240

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	LCD1602 Panel 8-Bit Mode Controlled on NuMicro NUC240 Series
BSP Version	NUC230_240 Series BSP CMSIS V3.01.001
Hardware	NuTiny-NUC240V

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

www.nuvoton.com

1 Function Description

1.1 Introduction

In this example code, the NuMicro NUC240 series is used as an example to implement the 8-bit mode control on LCD1602.

SMC1602(LCD1602) is a LCD display panel. 1602 means there are 16 characters and 2 lines for displaying the message. Besides, there are 8 bit control mode and 4 bit control mode on SMC1602. For more detailed SMC1602 information and instructions, please refers to the SMC1602 datasheet.

1.2 SMC1602 Pin Description

Pin num	Symbol	Description	Pin num	Symbol	Description
1	VSS	Power Ground	**9**	D3	LCD DB2
2	VDD	Power	**10**	D4	LCD DB3
3	VL	Panel Bias Voltage	**11**	D5	LCD DB4
4	RS	Register Select RS = 0 : Cmd register RS = 1 : Data register	**12**	D6	LCD DB5
5	R/W	LCD Read/Write R/W = 0 : Write R/W = 1 : Read	**13**	D7	LCD DB6
6	E	LCD Enable(Enable) E = 0 : LCD disable E = 1 : LCD enable	**14**	D8	LCD DB7
7	D0	LCD DB0 (Data Bus)	**15**	BLA	Back Light Anode
8	D1	LCD DB1	**16**	BLK	Back Light Kathode

2 Code Description

2.1 Function Description

The following code describes the GPIO setting and macro defines. For the detailed description, please refer to the following comment.

```
/*---------------------------------------------------------------------------------------------*/
/* 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 Write Data into LCD

This function writes the displaying data into LCD module. For the detailed description, please refer to the following comment.

```
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 Write Command into LCD

This function writes the command into LCD module. For the detailed description, please refer to the following comment.

```
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 Initialize the LCD Module

This function sends some command into LCD module for initializing. For the detailed description, please refer to the following comment.

```
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 Select the Displaying Line of LCD

Select the displaying line of LCD and send the command into LCD module.

```
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 Write the Character

Write the displaying character into LCD module.

```
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 Write the String

Write the displaying strings into LCD module.

```
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 Main Loop

Main loop will set the GPIOA, initialize the LCD module, and input 2 strings for displaying.

```
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 Software and Hardware Environment

Software Environment

- BSP version
  - NUC230_240 Series BSP CMSIS V3.01.001
- IDE version
  - Keil uVersion 5.26

Hardware Environment

- Circuit components
  - NuTiny-NUC240V * 1
  - Nu-Bridge * 1
  - SMC1602 * 1

■ Diagram

1. Follow the diagram to connect the SMC1602, Nu-Bridge and NUC240.

2. For printing the debug message, user needs to connect two wires, the UART_RX(PB.0) and UART_TX (PB.1) of NUC240 to Nu-Bridge. Then set the COM Port number and Baudrate in terminal tool. User can find the COM Port number in Device Manager of PC and it will display “NuBridge Virtual Com Port (COMX)”. The Baudrate should be 115200.

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 Directory Information

- 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 How to Execute Example Code

1. This project supports Keil uVersion 5.26 and higher version
2. According to the chapter 4 Directory Information, enter the path EC_NUC240_LCD1602_V1.00\SampleCode\ExampleCode\1602 LCD 8BIT folder and double click the 1602 LCD 8BIT.uvproj
3. Enter the Keil compile mode window
    a. Build the project
    b. Download
    c. Start / Stop the debug mode
4. Enter debug mode
    a. Run

6 Revision History

Date	Revision	Description
Nov. 11, 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.