Electro-Phoretic Display Module Base on ML51

Example Code Introduction for 8-bit NuMicro® Family

Information

Application	This sample code is based on the SPI and GPIO of ML51 to realize the electronic paper module that controls the micro-snow electronics. 2.9inch e-Paper Module (B)
BSP Version	ML51_BSP_Keil_C51_V1.0.0
Hardware	NuTiny-SDK-ML51 V1.1

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 Function Description

1.1 Introduction

Electronic paper is a reflective display based on Electro-Phoretic Display (EPD) device. The principle of display is that the charged ions move in a transparent or colored liquid with an applied electric field, which is turned or flowed to brighten or darken the pixels, and can be fabricated on a glass, metal or plastic substrate.

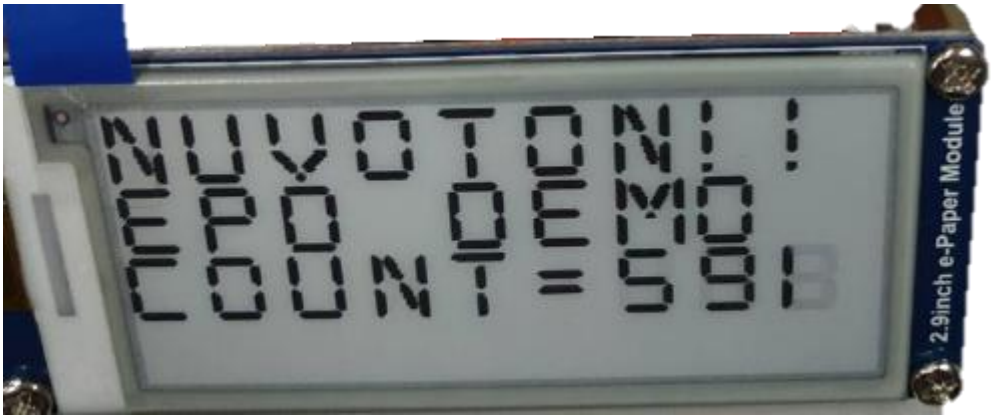
Advantages of electrophoresis technology include low energy consumption, reflective display with good effects under the sunlight or in dark environments and the low power consumption when changing images.

Please refer to this website for module related information:

http://www.waveshare.net/wiki/2.9inch_e-Paper_Module_(B)

1.2 Demo Result

Based on the SPI and GPIO functions of the ML51, this sample code implements the 2.9inch e-Paper Module (B) for controlling the electronic paper module of the micro-snow electronics.



2 Code Description

(1) Initialize the GPIO to PUSHPULL mode which connected to the EPD module and enable the SPI module function.
(2) Initialize the EPD module through the SPI command, and then transfer the display content.

```
void main(void)
{
    //EPD GPIO INIT
    P25_INPUT_MODE;
    P24 = 1;
    P23 = 1;
    P22 = 1;
    P24_PUSHPULL_MODE;
    P23_PUSHPULL_MODE;
    P22_PUSHPULL_MODE;
    //SPI GPIO initial
    MFP_P00_SPI0_MOSI;
    P00_QUASI_MODE;
    MFP_P02_SPI0_CLK;
    P02_QUASI_MODE;
    //Initial SPI0 for EPD
    SPI_Open(SPI0, SPI_MASTER, 10, SPI_MODE_3, MSB_FIRST);
    EPD_Init(&epd, lut_full_update);

    EPD_SetFrameMemory(&epd, 0, 0); //clear ram
    EPD_DisplayFrame();

    EPD_SetFrameMemory(&epd, 0, 0); //clear ram
    EPD_DisplayFrame();

    //inital EPD function
    EPD_Init(&epd, lut_partial_update);

    while(1)
    {
        print_lcd(0, "Nuvoton!!!");
        print_lcd(32, " EPD Demo");
        sprintf(value, "count=%d", count_t);
        print_lcd(64, value);
        EPD_DisplayFrame();
        EPD_DelayMs(2000);
        count_t++;
    }
}
```

Display command subroutine

```
/**
*  @brief: update the display
*          there are 2 memory areas embedded in the e-paper display
*          but once this function is called,
*          the the next action of SetFrameMemory or ClearFrame will
*          set the other memory area.
*/
void EPD_DisplayFrame(void)
{
    EPD_SendCommand(DISPLAY_UPDATE_CONTROL_2);
```

```
    EPD_SendData(0xC4);
    EPD_SendCommand(MASTER_ACTIVATION);
    EPD_SendCommand(TERMINATE_FRAME_READ_WRITE);
    EPD_WaitUntilIdle();
}
```

3 Hardware and Software Environment

Software Environment

- BSP Version
  - ML51_BSP_Keil_C51_V1.0.0
- Project Name
  - ML51_WS_EPD_Module.uvproj

Hardware Environment

- Circuit components
  - NuTiny-SDK-ML51
  - 2.9inch e-Paper Module (B)
- Diagram

NuTiny-SDK-ML51 V1.1		2.9inch e-Paper Module (B)
VCC	↔	VCC
(GPIO INPUT)P25	↔	EPD_BUSY
(GPIO OUTPUT)P24	↔	EPD_RESET
(GPIO OUTPUT)P23	↔	EPD_DC
(GPIO OUTPUT)P22	↔	EPD_CS
(SPI CLK)P02	↔	EPD_CLK
(SPI MOSI)P00	↔	EPD_MOSI
GND	↔	GND

4 Directory Information

- EC_ML51_WS_EPD_Module_V1.0
  - Library: Sample code header and source files.
    - Device: ML51 device header files.
    - Startup: Keil51 startup files
    - StdDriver: All peripheral driver header and source files.
  - Sample_Code
    - ExampleCode: Source file of example code folder.

5 How to Execute a Sample Code

1. The project can be run in Keil uVision4 PK51 Development Kit V9.52 and above version.
2. Enter the ExampleCode\WS_EPD_Module\KEIL folder according to the Directory Information section and double-click ML51_WS_EPD_Module.uvproj to open the project.
3. Enter Keil compiler mode
   a. Build
   b. Download
   c. Enter / Exit Debug Mode
4. When in under Debug mode
   a. Free run
5. Check the EPD module display.

6 Revision History

Date	Revision	Description
Sep. 24, 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.

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.