

## 4 wired Resistive Touch Panel Controlled

Example Code Introduction for 32-bit NuMicro<sup>®</sup> Family

### Information

Application	This sample code based on NUC126 for implementing the 4 wired resistive touch panel controlled.
BSP Version	NUC126Series_BSP_CMSIS_v3.00.003
Hardware	NuTiny-NUC126

*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](http://www.nuvoton.com)

## 1 Function Description

### 1.1 Introduction

Currently, there are 4-wired, 5-wired, 7-wired and 8-wired types resistive touch panel on the market. This sample code based on 4-wired resistive ADC touch panel.

The X-axis direction of touch panel is regarded as resistor and the Y-axis direction of the touch panel is regarded as another resistor, too. When user press the touch screen, we can get two sampling result from X-axis and Y-axis resistor by ADC. After sampling finished, calculating the touch point by the ADC sampling result.

For example, the resolution of touch panel is 128x160. Because the NUC126 is equipped with 12 bits ADC, the 5V will be divided into 4096 parts. The ADC sampling result on the far left point, 0 point on X-axis will map to probably 500 but 0. The ADC sampling result on the far right point, 128 point on X-axis will map to probably 3200 but 4095, too. Hence, the result of ADC sampling should be normalized.

Ex. If the ADC sampling result is (2048, 1024), what's the touch point on the 128x160 touch panel?

1. Normalized the ADC sampling result.  
Let the ADC value 500 be 0 point on X-axis and 3200 be 128 point on X-axis. So the X-axis can be cut to  $3200 - 500 = 2700$  parts.
2. Calculate the X-axis touch point.  
By the proportion formula "  $128 : 2700 = X\text{-axis touch point} : 2048 - 500$  ", we can calculate the X-axis touch point result is 73.39.
3. The calculate method of Y-axis touch point is same as the X-axis.

## 1.2 Demo Result

串口配置

接收區：已接收3449位元組，速度0位元組/秒，接收狀態[允許]，輸出文本狀態[已停止]

端口：COM69

波特率：115200

數據位：8

停止位：1

校驗：NONE

 關閉串口

線路控制

DTR     BREAK

RTS

線路狀態 (只讀)

CTS     DSR

RING     RLSD

```
(0, 16)
(1, 17)
(6, 20)
(13, 26)
(19, 32)
(25, 39)
(33, 48)
(40, 57)
(47, 64)
(53, 72)
(60, 79)
(67, 88)
(74, 95)
(80, 103)
(84, 109)
(90, 114)
(98, 121)
(105, 128)
(112, 134)
(118, 142)
(124, 147)
(127, 153)
```

## 2 Code Description

### 2.1 Main Function

In main function, it will do system initializing, UART initializing and touch panel hardware initializing. The while loop will report the touch point per second.

```
/*-----*/
/* Main Function */
/*-----*/
int32_t main(void)
{
    uint32_t u32TouchX, u32TouchY;
    SYS_Init();
    UART0_Init();

    // Init Touch Panel HW
    Drv_NuTFT_HWInit();

    // Touch Test
    while (1)
    {
        if (App_GetTouchPoint(&u32TouchX, &u32TouchY) == Get_Success)
            printf("(%d, %d) \n", u32TouchX, u32TouchY);
        CLK_SysTickDelay(1000000);
    }
}
```

### 2.2 Get the Touch Point of X-Axis

Because this is a resistive touch panel, when senses the x-axis direction touch point, it needs to set y-axis direction wires on input mode and set the x-axis direction wires (XR, XL) to output mode. Makes XR be logic high, XL be logic low and samples the voltage on YU wire by ADC to get the x-axis touch point. We can get the the touch point of y-axis direction in the same way.

```
static uint16_t _Api_Get_TP_X(void)
{
    uint32_t u32_ADC_Convert_Result_X_Axis;
    uint16_t u16_nNormalizationFactor;

    /* Init ADC for TP */
    /* Set input mode as single-end and enable the A/D converter */
    ADC_Open(ADC, ADC_ADCR_DIFFEN_SINGLE_END, ADC_ADCR_ADMD_SINGLE, BIT6);
```

```
/* Get X from ADC input */
GPIO_SetMode(PE, BIT2, GPIO_MODE_OUTPUT); // XR
GPIO_SetMode(PB, BIT11, GPIO_MODE_INPUT); // YD
GPIO_SetMode(PB, BIT10, GPIO_MODE_OUTPUT); // XL
PE2 = 1; //XR High
PB10 = 0; //XL Low

/* Configure the GPB9 ADC analog input pins. */
SYS->GPE_MFPL &= ~(SYS_GPE_MFPL_PE2MFP_Msk); // Disable ADC CH9
SYS->GPB_MFPH &= ~(SYS_GPB_MFPH_PB9MFP_Msk); // Enable ADC CH6
SYS->GPB_MFPH |= SYS_GPB_MFPH_PB9MFP_ADC0_CH6; //YU sample

/* Disable the GPB8 digital input path to avoid the leakage current. */
GPIO_DISABLE_DIGITAL_PATH(PB, BIT9); //YU

/* Power on ADC module */
ADC_POWER_ON(ADC);

/* Enable the sample module 1 interrupt. */
ADC_EnableInt(ADC, ADC_ADF_INT); //Enable sample module A/D ADINT1 interrupt.
NVIC_EnableIRQ(ADC_IRQn);
/* Clear the A/D ADINT1 interrupt flag for safe */
ADC_CLR_INT_FLAG(ADC, ADC_ADF_INT);

/* Reset the ADC interrupt indicator and trigger sample module 1 to start A/D
conversion */
g_u32AdcIntFlag_TP = 0;
ADC_START_CONV(ADC);

/* Wait ADC interrupt (g_u32AdcIntFlag_TP will be set at IRQ_Handler function) */
while (g_u32AdcIntFlag_TP == 0);

/* Get x-axis ADC convert result */
u32_ADC_Convert_Result_X_Axis = ADC_GET_CONVERSION_DATA(ADC, 6);

/* Close ADC module */
ADC_Close(ADC);

//Normalization
if (u32_ADC_Convert_Result_X_Axis <= ReportThreshold)
{
```

```
    u32_ADC_Convert_Result_X_Axis -= Init_MinBorderCaliValue_X;
    u16_nNormalizationFactor = (Init_MaxBorderCaliValue_X - Init_MinBorderCaliValue_X)
* 1000 / LCD_Resolution_X;
    u32_ADC_Convert_Result_X_Axis *= 1000;
    u32_ADC_Convert_Result_X_Axis /= u16_nNormalizationFactor;
}

return u32_ADC_Convert_Result_X_Axis;
}
```

### 3 Software and Hardware Environment

#### ● Software Environment

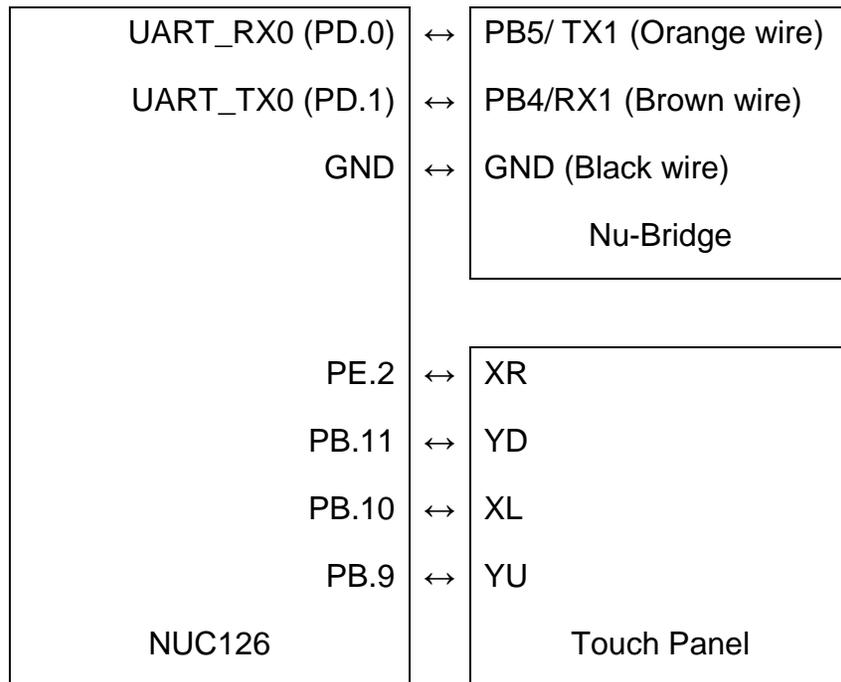
- BSP version
  - ◆ NUC126Series\_BSP\_CMSIS\_v3.00.003
- IDE version
  - ◆ Keil uVersion 5.26

#### ● Hardware Environment

- Circuit components
  - ◆ NuTiny-NUC126
  - ◆ Nu-Bridge
  - ◆ NuTFT v1.3 Touch Panel

- Diagram

For printing the debug message, user needs to connect two wire, the UART\_RX0(PD.0), UART\_TX0(PD.1) of NUC126 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. Finally, connects the NUC126 and Touch Panel accorded to the following diagram.



## 4 Directory Information

📁 EC\_NUC126\_4Wired\_Resistive\_Touch\_Panel\_V1.00

- 📁 **Library**                      Sample code header and source files
  - 📁 **CMSIS**                      Cortex<sup>®</sup> Microcontroller Software Interface Standard (CMSIS) by Arm<sup>®</sup> 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 ExampleCode/4Wired\_Resistive\_Touch\_Panel/Keil folder and double click the 4Wired\_Resistive\_Touch\_Panel.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
Jun. 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.*