

## 如何使用LCDView完成Com/Segment LCD專案

NuMicro® 32位系列微控制器範例代碼介紹

### 文件資訊

應用簡述	本範例代碼使用 NuMaker-M258KG 說明如何使用 LCDView 完成 COM/SEG LCD 專案
BSP 版本	M251_M252_M254_M256_M258_Series_BSP_CMSIS_V3.02.003
開發平台	NuMaker-M258KG 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 and microprocessor 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. 概述

NuTool-LCDView 是一款用於輔助 COM/SEG LCD 開發的軟體工具，特色是能讓開發者在不需要有實體 LCD 屏幕的情況下，即可開發、除錯、顯示效果。操作步驟使用圖形化介面點選設定以及拖拉放的方式，開發者可以依照需求自定義名稱、COM/SEG 規劃，快速完成 LCD 顯示設定，並自動產生對應的定義代碼，再結合 BSP 與 API，即可直接於 LCD 上與 PC 上展現設計效果。

本範例代碼將示範如何使用 LCDView 軟體工具，開發與除錯，逐步完成 COM/SEG LCD 專案開發。

### 1.1 原理

#### 1.1.1 LCDView 功能說明

NuTool-LCDView 的功能可以分為 Create Mode 與 Emulator Mode 兩種模式。開發時需要先在 Create Mode 下完成 LCD 屏幕的圖面配置與 COM/SEG Table 的設定，開發完成並燒錄至 MCU 後，可以切換到 Emulator Mode 在 NuTool-LCDView 上查看顯示效果。LCDView 提供單機版本與嵌入在 Keil IDE 與 IAR IDE 的版本，讓開發者可以在 IDE 環境下配合使用 LCDView 單步調適與偵錯。

介面介紹與使用操作，可以參考 [NuTool-LCDView 用戶手冊](#)。

#### 1.1.2 Supported Chips

NuTool-LCDView 目前支援以 Arm® Cortex®-M23 為核心的 M254/M256/M258 系列與 M2354 系列，以及 8-bit-8051 的 ML54 系列與 ML56 系列。

## 1.2 執行步驟

使用 NuTool-LCDView 完成專案的執行步驟可分為四大階段，第一階段：準備相關硬體與軟體，第二階段：進入 Create Mode 編輯內容，第三階段：代碼編輯，第四階段：進入 Emulator Mode 模擬與除錯。

以下就各項階段詳細說明執行步驟。

### 1.2.1 第一階段：準備相關硬體與軟體

在此次範例程式中，會使用到的硬體資源為 NuMaker-M258KG 以及型號 TNLCDSUB\_M258K 的 LCD 上板。

軟體工具需要使用三個項目，分別為 NuTool-LCDView，NuTool-PinConfigure，以及 Nu-Link Keil Driver。NuTool-PinConfigure 是提供腳位配置的工具，配置完後可以匯入至 NuTool-LCDView，以直接在 COM/SEG table 上顯示對應的 Pin Number；Nu-Link Keil Driver 則是作為支援讓 NuTool-LCDView 嵌入在 Keil IDE 中，以方便調適與除錯。實際使用將於第二階段：進入 Create Mode 編輯內容與第四階段：進入 Emulator Mode 模擬與除錯分別說明。

### 1.2.2 第二階段：進入 Create Mode 編輯內容

#### 1.2.2.1 開啟 NuTool-PinConfigure 配置 LCD 相關腳位

首先開啟 NuTool-PinConfigure，選擇 MCU 所屬的系列與相對應的 MCU 型號，根據 NuMaker-M258KG 與 LCD 的電路配置，如圖 1-1 所示，在 NuTool-PinConfigure 中完成 LCD 腳位配置，如圖 1-2。

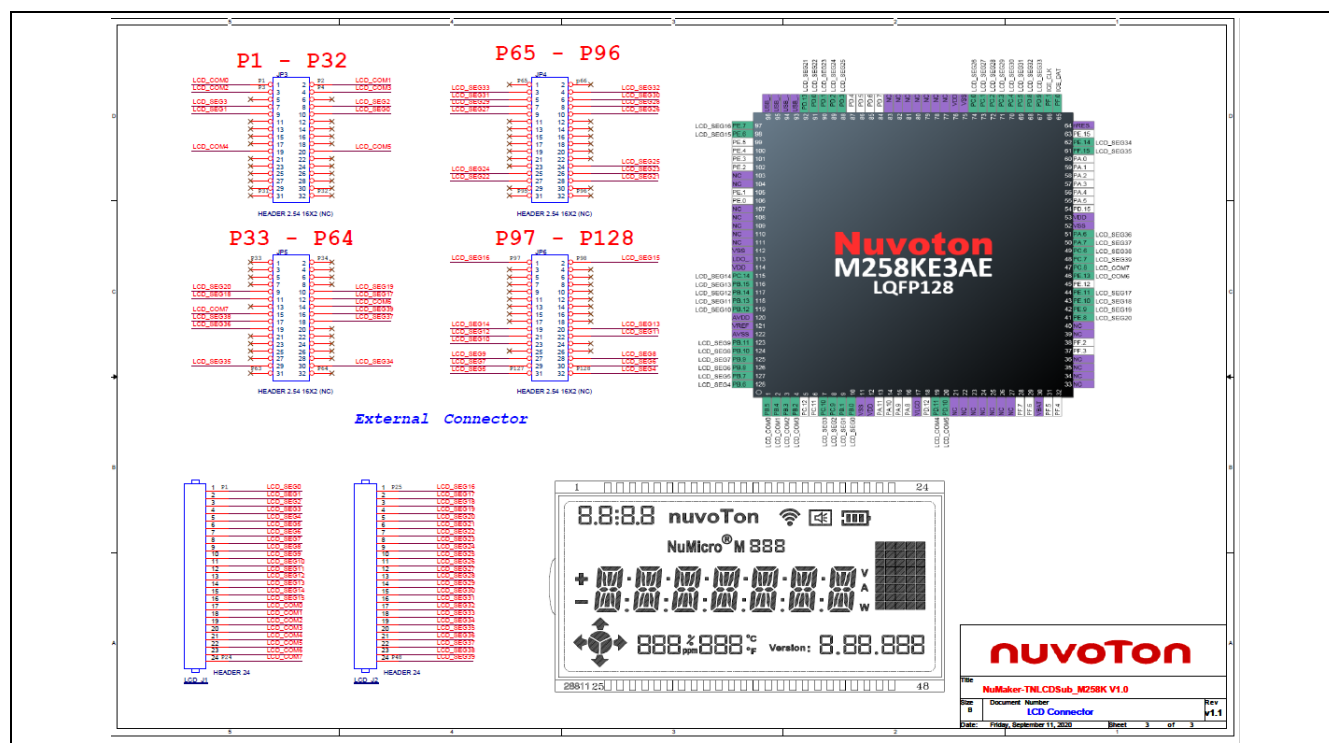


圖 1-1 NuMaker - M258KG 電路圖

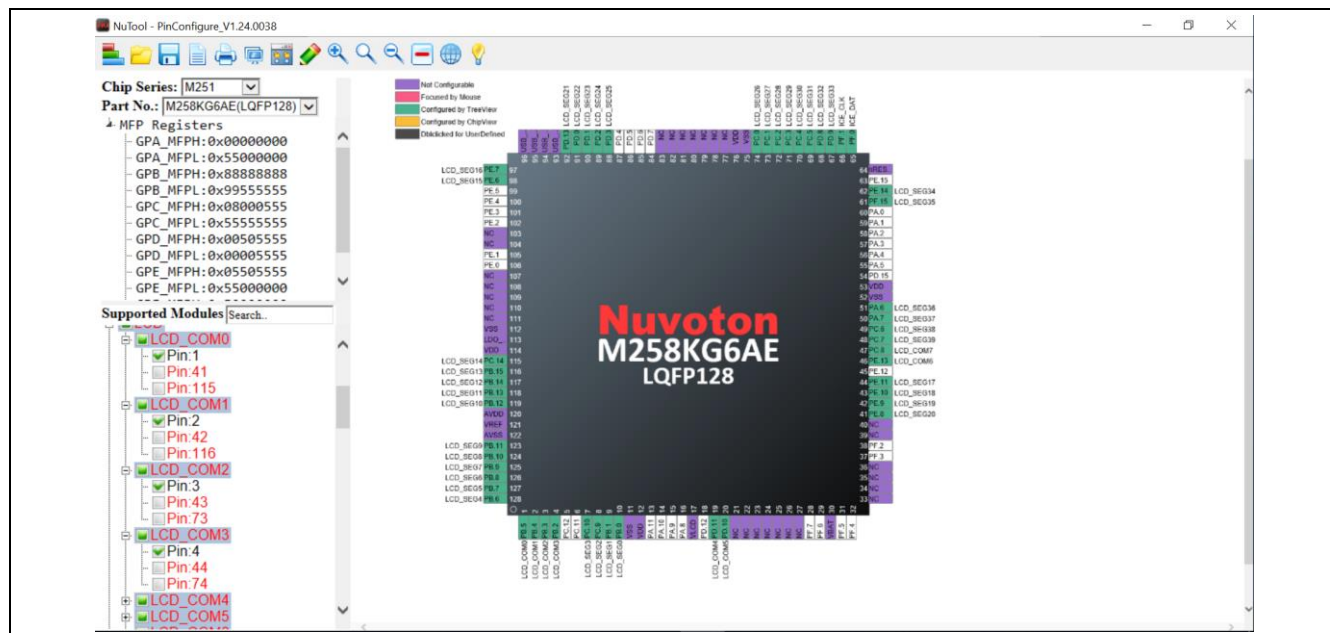


圖 1-2 在 NuTool-PinConfigure 完成 LCD 腳位配置

如圖 1-3步驟所示，選擇Generate Code，從NuTool-PinConfigure中自動生成相對應的專案檔-M258KG.cfg，此檔案將用於1.2.2.1開啟NuTool-PinConfigure配置LCD相關腳位，以及相對應的Pin Define檔M258KG.c，此檔案將於代碼介紹中的LCD屏幕初始化LCD\_Init()：初始化LCD Pin腳位步驟使用。

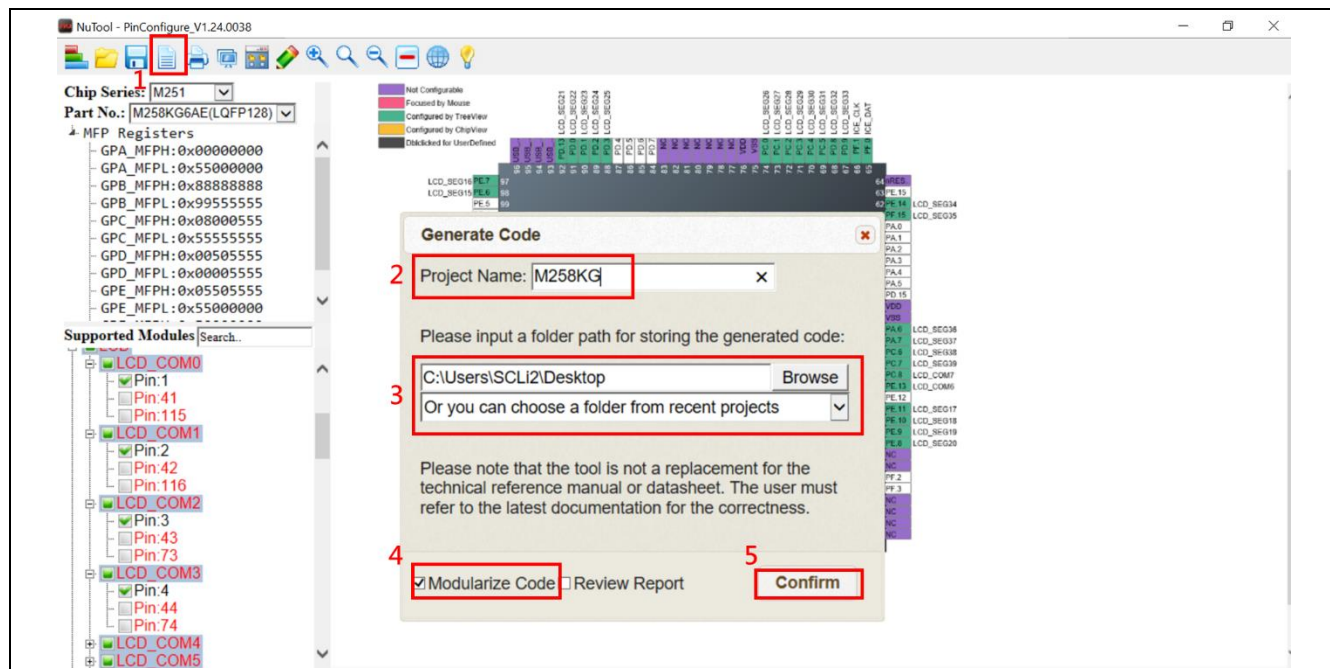



圖 1-3 NuTool-PinConfigure 操作說明

### 1.2.2.2 將 PinConfigure 檔案匯入 LCDView 中

先在 NuTool-LCDView 中選擇對應的 Chip Series，接著點選  圖示，選擇匯入剛剛完成的 NuTool-PinConfigure 檔案 M258KG.cfg，操作步驟如圖 1-4。匯入成功後，即可看到 Pin Number 顯示在相對應的 COM/SEG 上，如圖 1-5。

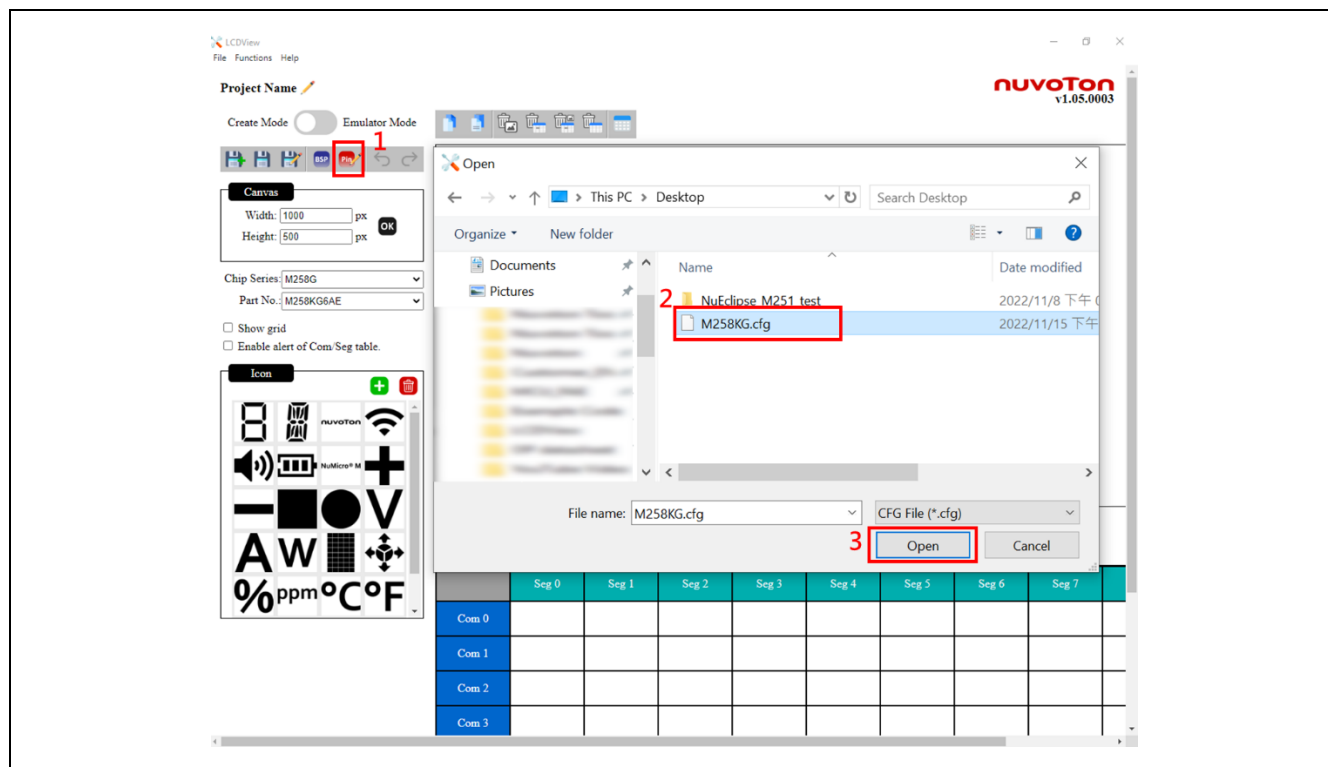


圖 1-4 將.cfg 檔案匯入 LCDView

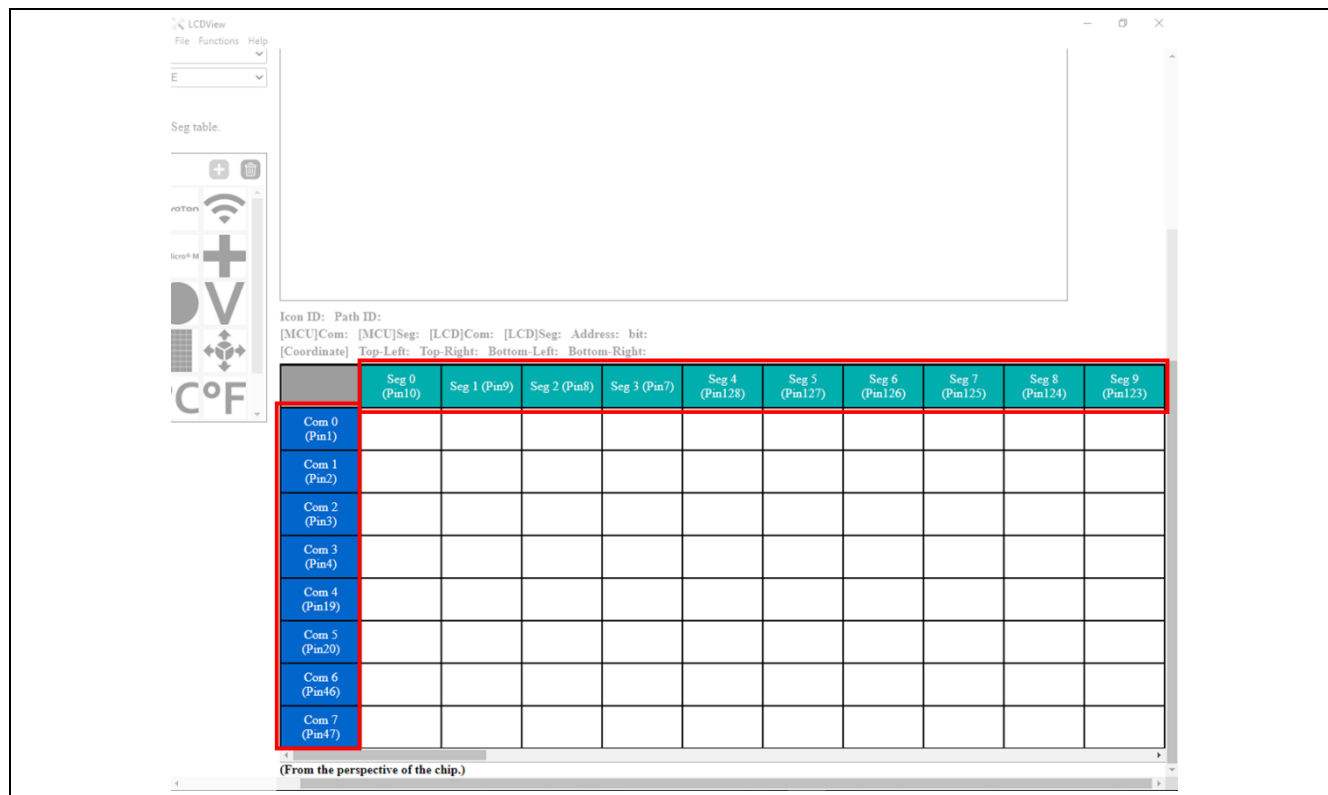


圖 1-5 匯入成功即可看到 MCU Pin Number 與 COM/SEG 對應

### 1.2.2.3 編輯圖面與 COM/SEG

根據 LCD 規格，如圖 1-6，將圖示以拖、拉、放的方式，直接放到 LCDView 的 Canvas 上，若圖示大小不符合，也可以自由放大縮小、調整位置，完成擺放後如圖 1-7；若 LCDView 的基本圖庫未能滿足需求，可以自行增加 svg 檔案型別的圖片到 LCDView 中。

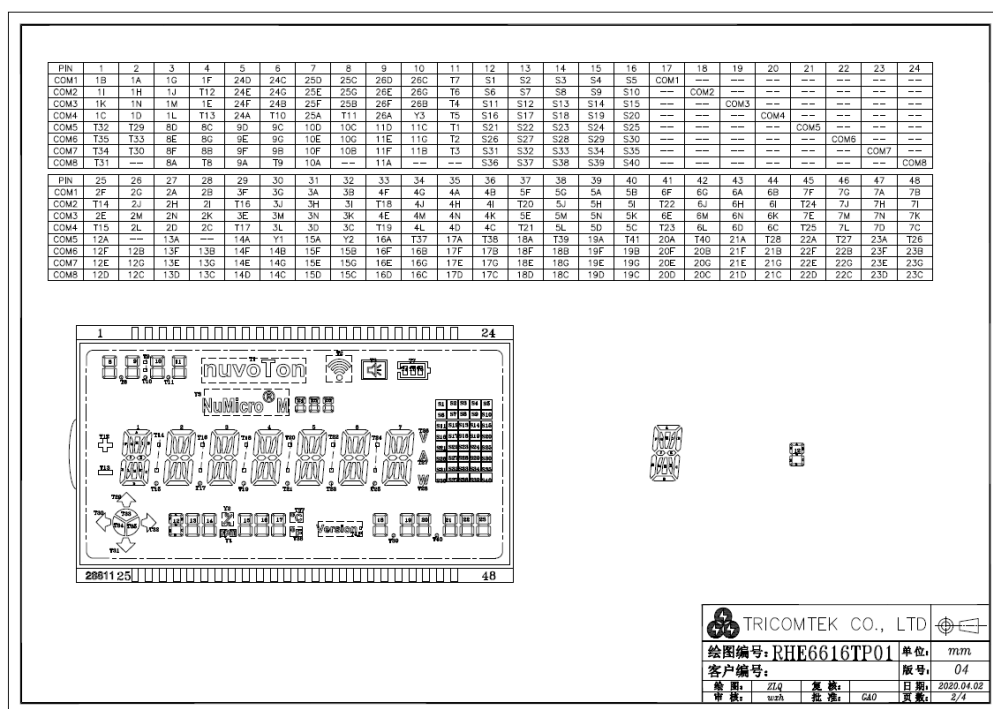


圖 1-6 LCD 規格

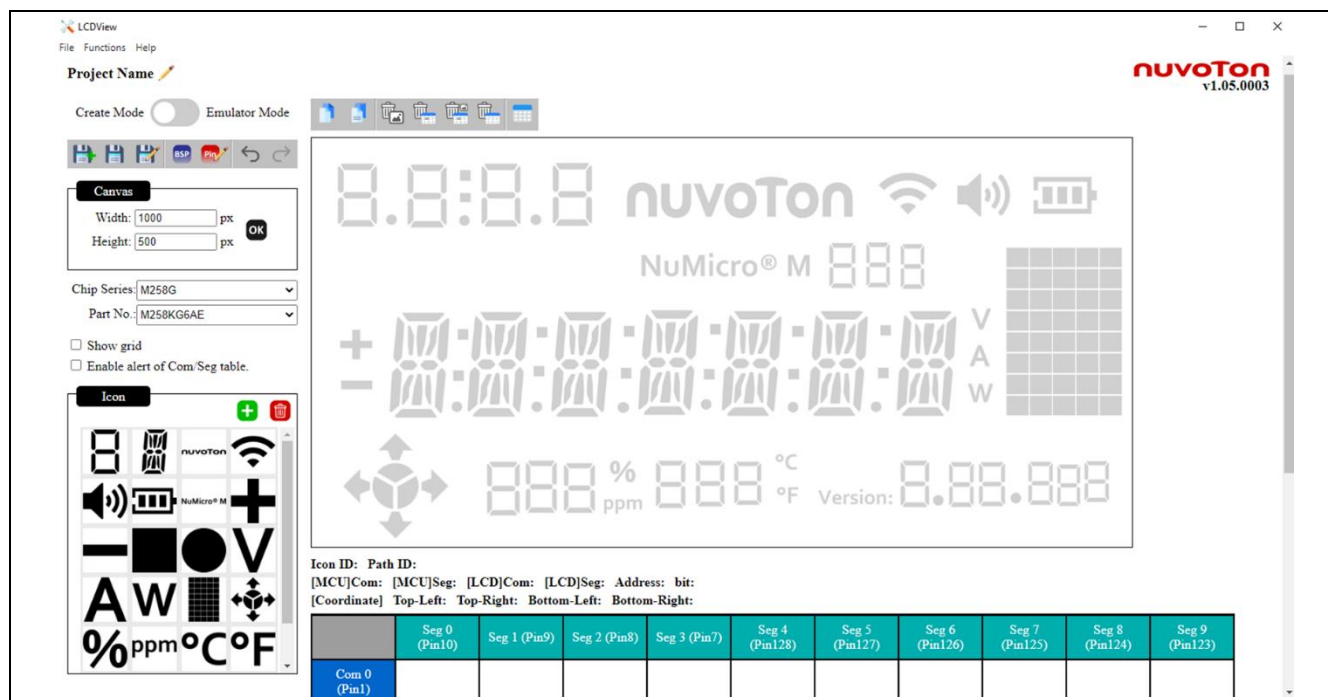


圖 1-7 根據 LCD 規格完成圖面擺放

圖面擺放完成後，將圖面依據LCD規格之定義，以點選的方式，將圖面點選至對應的COM/SEG中。七段顯示器以及十四段顯示器，則是分段依順序點選至設定的COM/SEG，如圖



1-8所示。

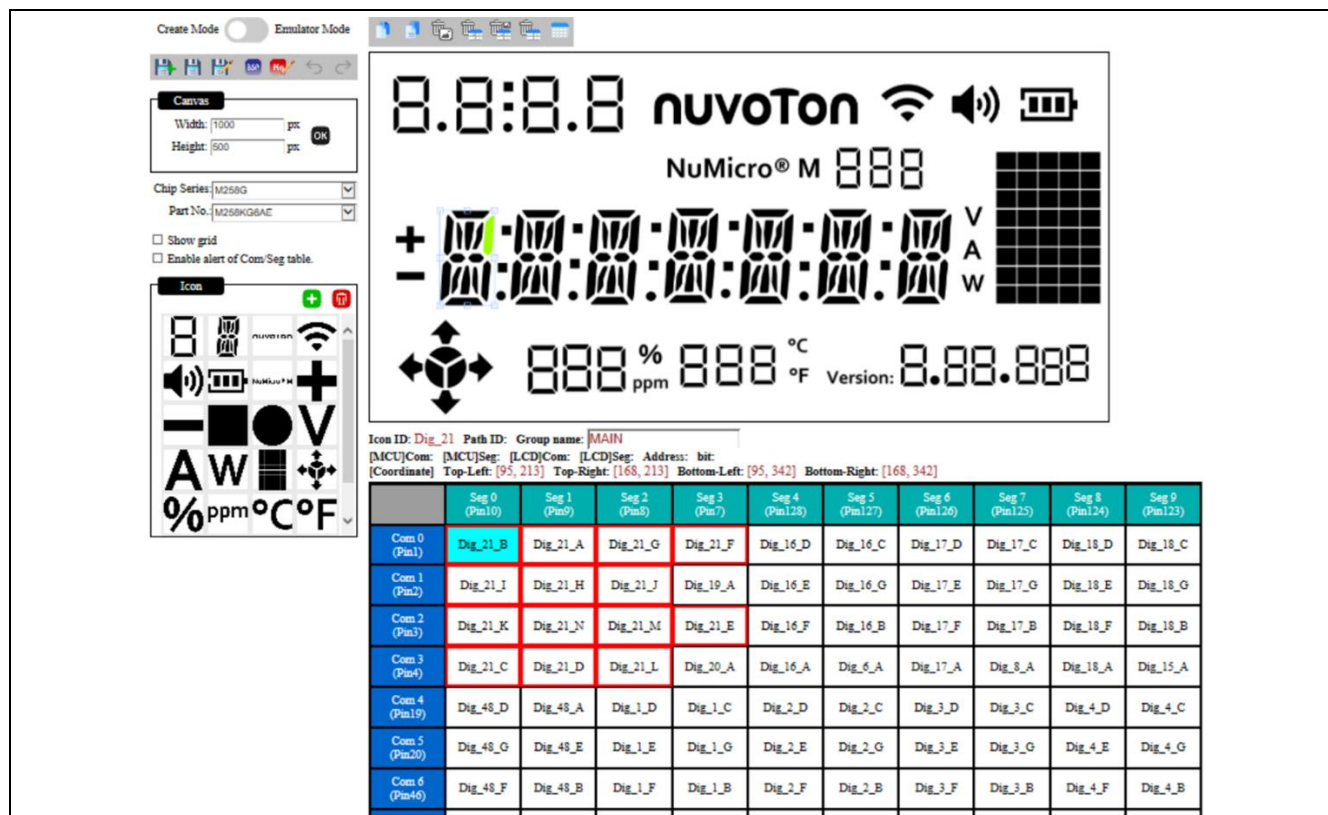


圖 1-8 依規格將對應圖面點選設定至 COM/SEG 範例

LCDView支援自定義命名圖案，所設定的名稱，亦會作為生成LCD圖面對應header file的名稱定義。例如圖 1-9所示，可以將wifi圖案名稱設定為“WIFI”。而七段顯示器與十四段顯示器則可以設定為以Group為概念，使用新唐提供的API便可以同時做動。例如圖 1-10所示，若將此七個十四段顯示器皆命名為相同名稱“MAIN”，則在使用新唐API時，便可以設定直接輸出一組數字或字串，詳細API使用方式將於第三階段：代碼編輯說明。



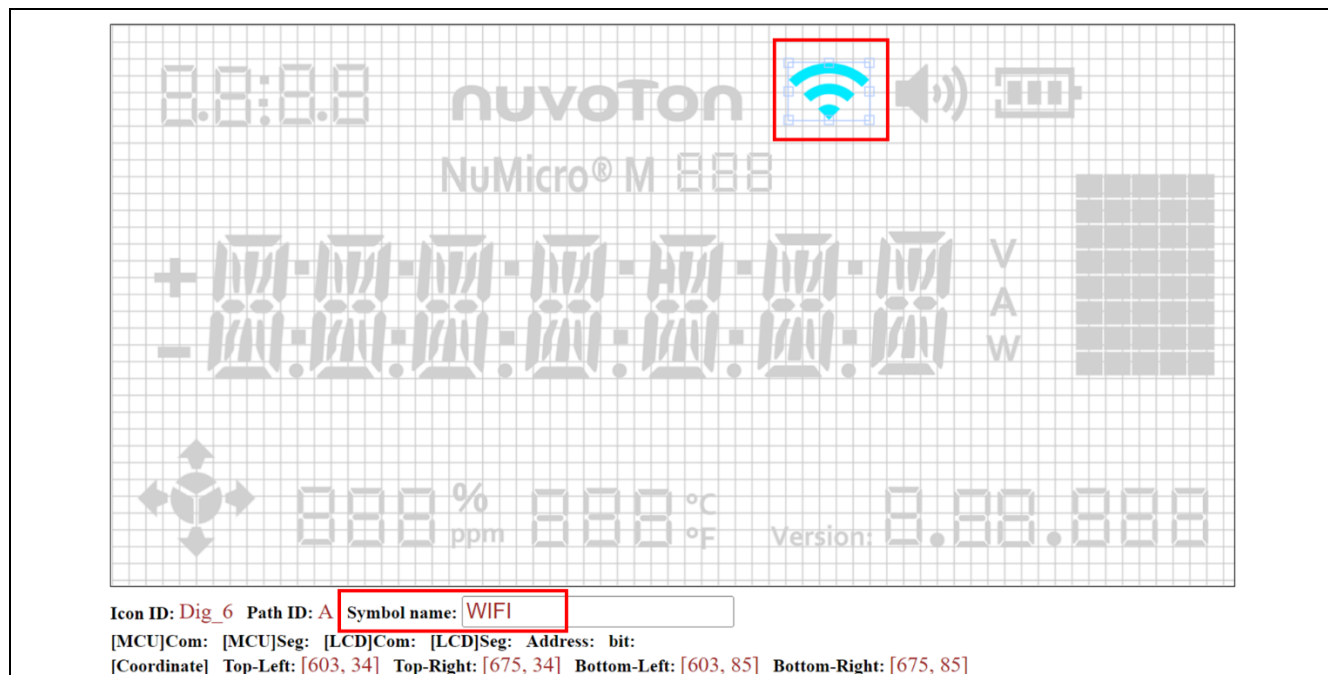


圖 1-9 自定義圖案名稱範例

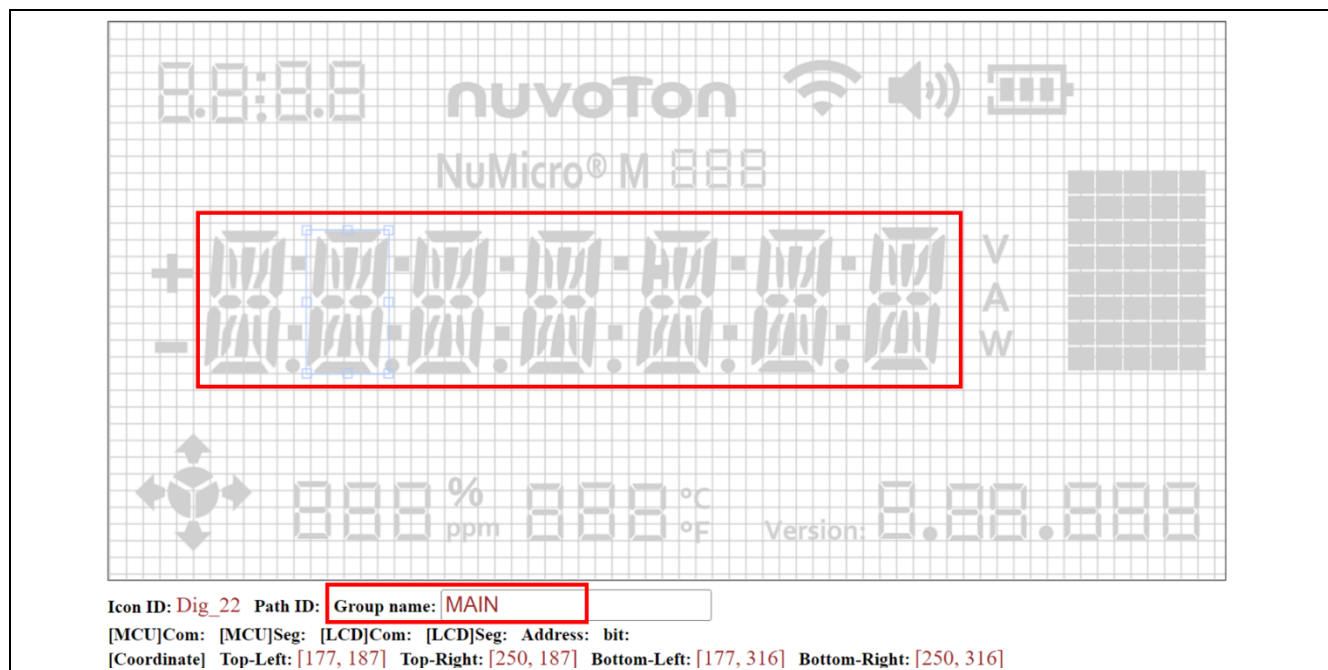



圖 1-10 自定義命名七段/十四段顯示器範例

### 1.2.3 第三階段：代碼編輯

#### 1.2.3.1 生成 LCD header file

在 LCDView 中編輯完圖面與 COM/SEG 設定後，按下 BSP 圖示，即可產出對應的 LCD header file，此檔案檔名為 lcdzone.h，如圖 1-11。在 lcdzone.h 中，可以清楚看到於章節 1.2.2.3 編輯圖面與 COM/SEG 中所自定義的圖面名稱，如圖 1-12 所示，並於第三階段：代碼編輯時，使用新唐提供的 API，即可馬上設定圖案顯示，效果如圖 1-13 所示。

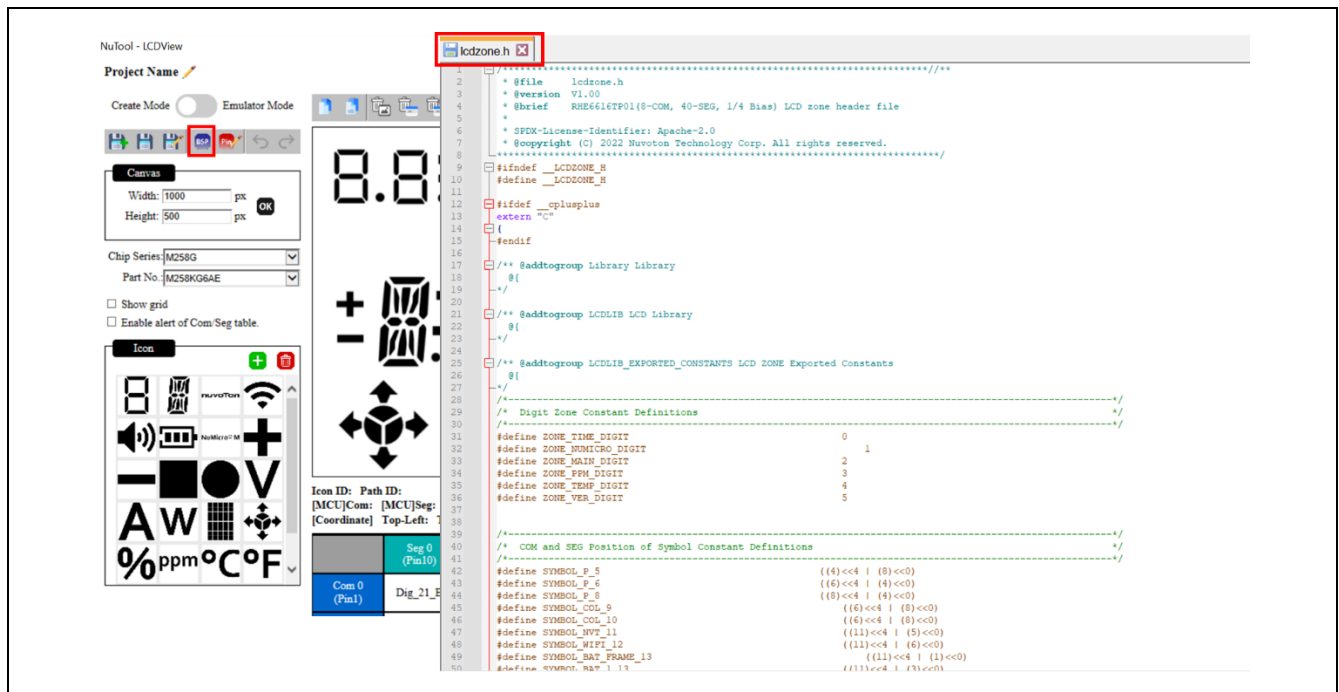


圖 1-11 按下 BSP 圖示後即可產出 lcdzone.h

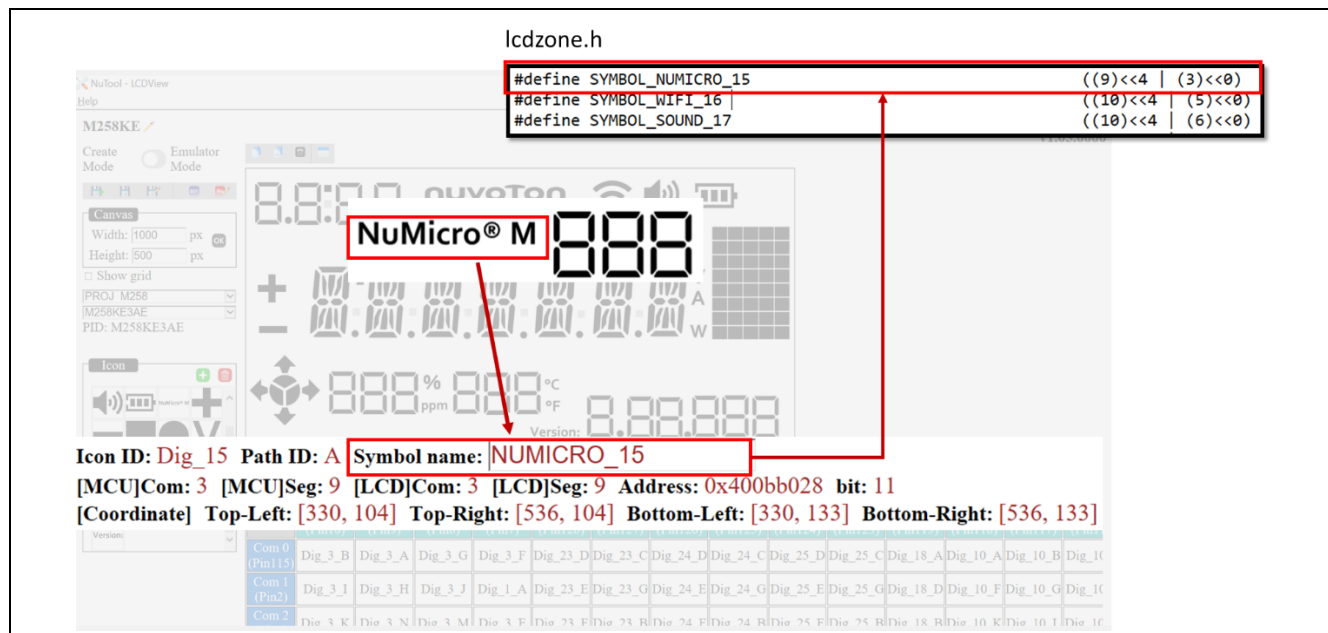


圖 1-12 自定義圖案名稱與自動生成的定義對照示意圖

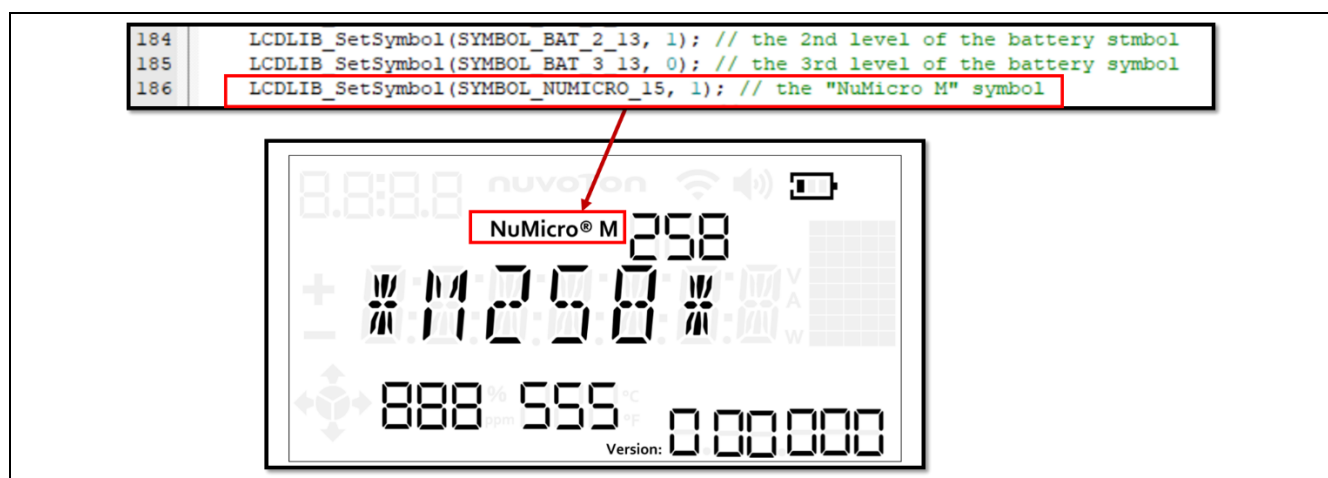


圖 1-13 使用 API 點亮指定圖案示意圖

### 1.2.3.2 將 lcdzone.h 匯入 Project

依照圖 1-14 路徑，將 lcdzone.h 放到資料夾中。

EC\_M258\_LCD\_View\_V1.00

Library

LCDLib

Include

lcdzone.h

圖 1-14 lcdzone.h 的檔案路徑

### 1.2.3.3 主程式代碼

主程式代碼請詳 2 代碼介紹。

## 1.2.4 第四階段：進入 Emulator Mode 模擬與除錯

LCDView 提供 Plug-in Keil IDE 的功能，使用 Plug-in Keil IDE 前需確保已安裝 Nu-Link Keil Driver。

### 1.2.4.1 在 Keil IDE 中單步調適

在 Keil IDE 編譯成功後，按下 debug session 圖示進入 Debug Mode，依照圖 1-15 在 Debug Mode 下開啟 LCDView 步驟，開啟 LCDView。

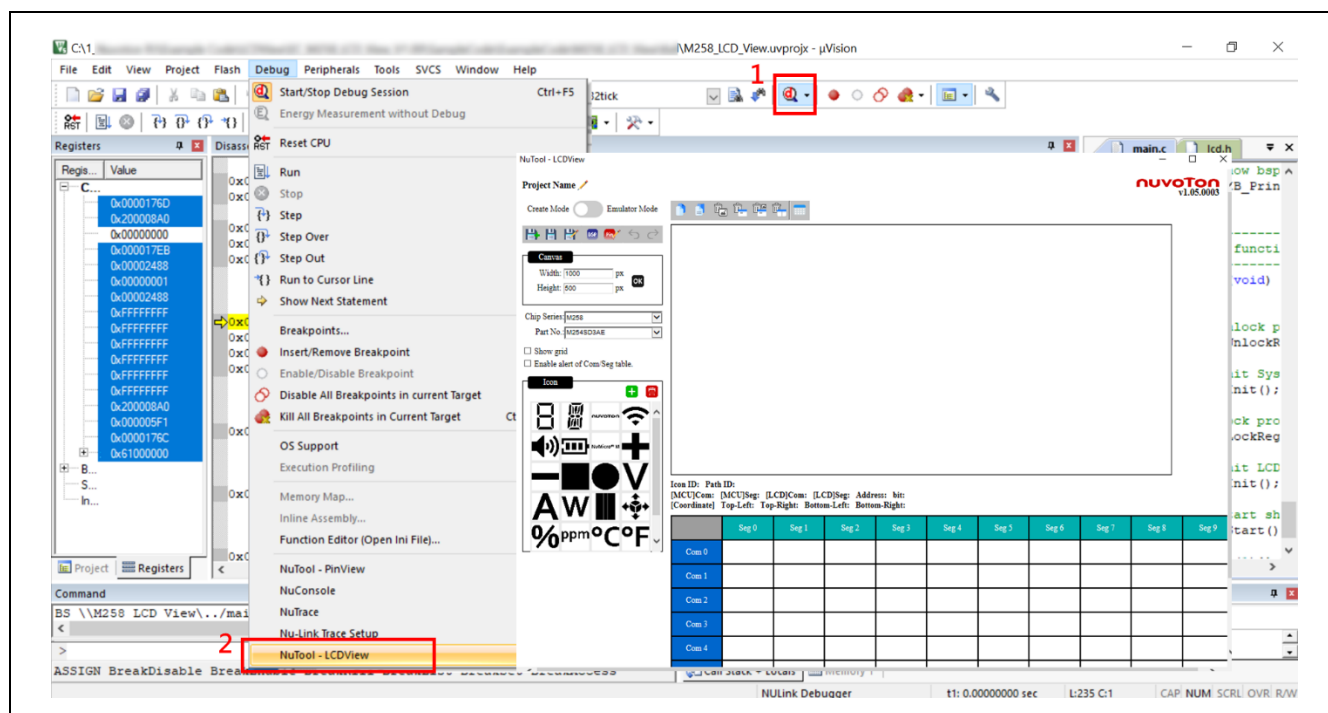


圖 1-15 在 Debug Mode 下開啟 LCDView

依照圖 1-16 步驟，將 LCDView 專案載入，並切換至 Emulator Mode，如圖 1-17。

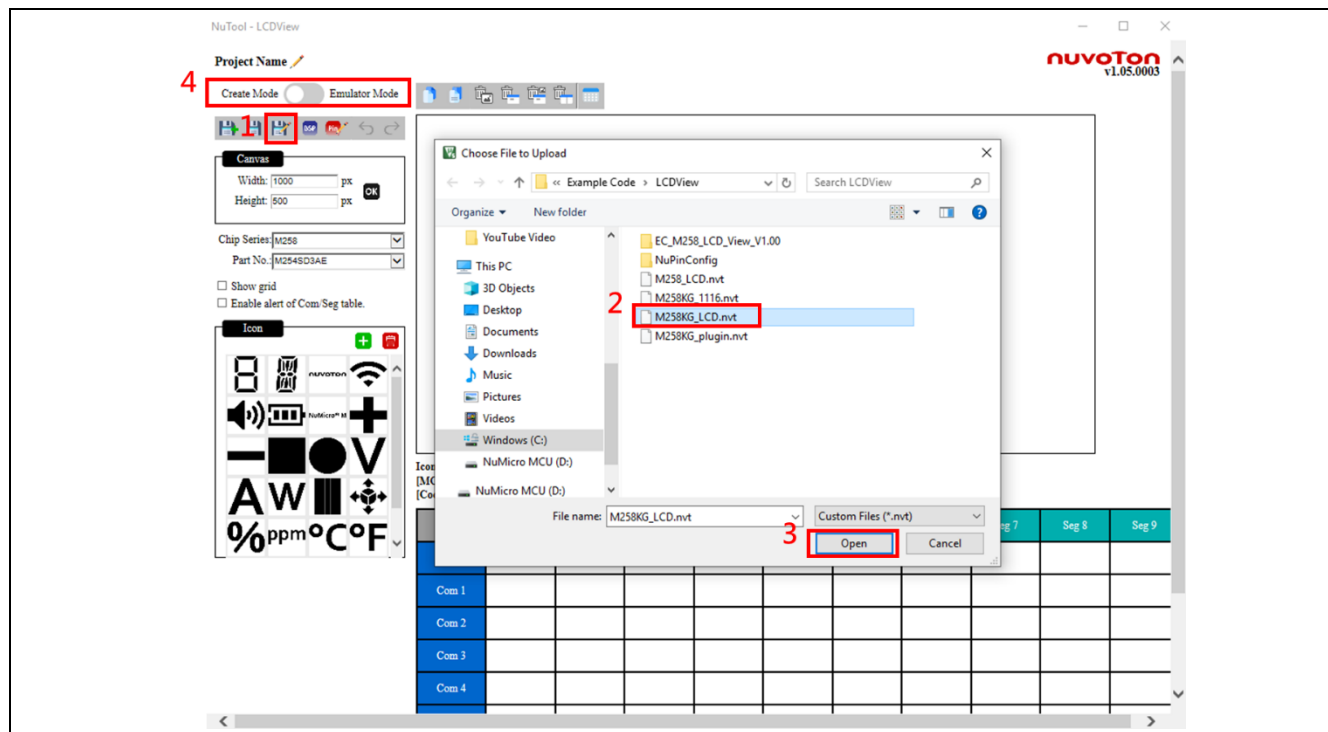


圖 1-16 載入 LCDView 專案，並切換至 Emulator Mode



圖 1-17 Emulator Mode 畫面

隨著單步執行，LCDView將會依代碼順序顯示所設定的效果，如圖 1-18。

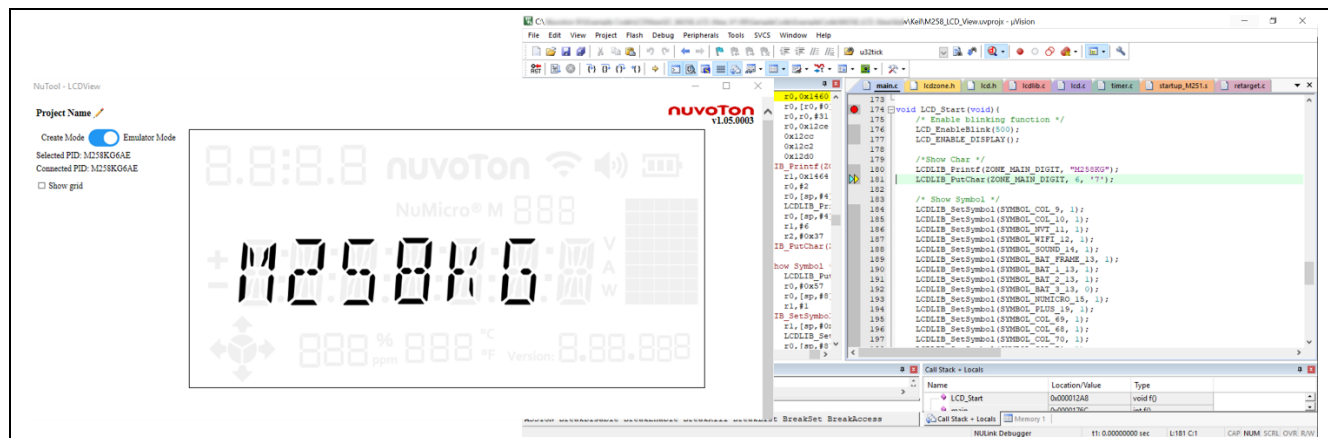


圖 1-18 單步調適與 Emulator Mode

### 1.3 執行結果

本範例代碼執行編譯和燒錄的結果如圖 1-19 所示。



圖 1-19 執行結果



## 2. 代碼介紹

主代碼於 *main.c* 中，包括 3 個部分如下：

1. 系統初始化 *SYS\_Init()*：系統與外設時鐘初始化
2. LCD 屏幕初始化 *LCD\_Init()*：初始化 LCD Pin 腳位與 LCD 相關設定
3. 點亮 LCD *LCD\_Start()*：開啟 LCD 屏幕及使用 *lcdlib.c* 中的 API 點亮屏幕內容

```
int main(void)
{
    /* Unlock protected registers */
    SYS_UnlockReg();

    /* Init System, peripheral clock and multi-function I/O */
    SYS_Init();

    /* Lock protected registers */
    SYS_LockReg();

    /* Init LCD multi-function pins and settings */
    LCD_Init();

    /* Start showing the specific text */
    LCD_Start();

    while (1) {}
}
```

系統初始化需要打開 GPIO 時鐘、系統時鐘、LCD 模組時鐘，請參考如下代碼。

```
void SYS_Init(void)
{
    /* Enable all GPIO clock */
    CLK->AHBCLK |= (CLK_AHBCLK_GPACKEN_Msk | CLK_AHBCLK_GPBCKEN_Msk |
    CLK_AHBCLK_GPCCKEN_Msk | CLK_AHBCLK_GPDCKEN_Msk |
    CLK_AHBCLK_GPECKEN_Msk | CLK_AHBCLK_GPFCKEN_Msk);

    /* Init System Clock */
    /* Enable HIRC clock */
    CLK_EnableXtalRC(CLK_PWRCTL_HIRCEN_Msk);

    /* Waiting for HIRC clock ready */
    CLK_WaitClockReady(CLK_STATUS_HIRCSTB_Msk);

    /* Enable LIRC clock */
    CLK_EnableXtalRC(CLK_PWRCTL_LIRCEN_Msk);

    /* Waiting for LIRC clock ready */
    CLK_WaitClockReady(CLK_STATUS_LIRCSTB_Msk);
}
```

```

/* Switch HCLK clock source to HIRC */
CLK_SetHCLK(CLK_CLKSEL0_HCLKSEL_HIRC, CLK_CLKDIV0_HCLK(1));

/* Set SysTick source to HCLK/2*/
CLK_SetSysTickClockSrc(CLK_CLKSEL0_STCLKSEL_HCLK_DIV2);

/* Configure LCD module clock */
CLK_EnableModuleClock(LCD_MODULE);
CLK_SetModuleClock(LCD_MODULE, CLK_CLKSEL2_LCDSEL_LIRC, 0);
}

```

LCD\_Init 需要根據 LCD 規格書，對 COM/SEG 相關引腳初始化，COM/SEG 引腳初始化可以直接使用於章節 1.2.2.1 開啟 NuTool-PinConfigure 配置 LCD 相關腳位中所自動生成的 M258KG.c 內的函數。

接著使用 LCD\_Open 與 LCD\_SET\_CP\_VOLTAGE 設定 LCD 相關參數，例如 duty、bias、frame rate 等，此次範例使用 NuMaker-TNLCDSUB\_M258K，參數規格為 1/8 duty、1/4 bias、frame rate 64Hz，請參考如下代碼。

```

static S_LCD_CFG_T g_LCDCfg =
{
    /*!< LCD clock source frequency */
    _LIRC,
    /*!< COM duty */
    LCD_COM_DUTY_1_8,
    /*!< Bias level */
    LCD_BIAS_LV_1_4,
    /*!< Operation frame rate */
    64,
    /*!< Waveform type */
    LCD_WAVEFORM_TYPE_A_NORMAL,
    /*!< Interrupt source */
    LCD_DISABLE_ALL_INT,
    /*!< Driving mode */
    LCD_HIGH_DRIVING_OFF_AND_BUF_ON,
    /*!< Voltage source */
    LCD_VOLTAGE_SOURCE_CP,
};

void LCD_Init(void)
{
    uint32_t u32ActiveFPS;
    /* The definition of multifunction pin is generated from NuTool-PinConfigure.*/
    M258KG_init_lcd();

    /* Reset LCD module */
    SYS_ResetModule(LCD_RST);

    /* Output Setting Select */
    LCD_OUTPUT_SET(LCD_OUTPUT_SEL8_TO_COM4 | LCD_OUTPUT_SEL9_TO_COM5 |
                  LCD_OUTPUT_SEL10_TO_SEG20 | LCD_OUTPUT_SEL11_TO_SEG19 |
                  LCD_OUTPUT_SEL12_TO_SEG18 | LCD_OUTPUT_SEL13_TO_SEG17 |

```

```

LCD_OUTPUT_SEL14_TO_COM6 | LCD_OUTPUT_SEL15_TO_COM7 |
LCD_OUTPUT_SEL24_TO_SEG31 | LCD_OUTPUT_SEL25_TO_SEG30 |
LCD_OUTPUT_SEL26_TO_SEG29 | LCD_OUTPUT_SEL27_TO_SEG28 |
LCD_OUTPUT_SEL28_TO_SEG27 | LCD_OUTPUT_SEL29_TO_SEG26 |
LCD_OUTPUT_SEL41_TO_SEG14 | LCD_OUTPUT_SEL42_TO_SEG13 |
LCD_OUTPUT_SEL47_TO_SEG8 | LCD_OUTPUT_SEL48_TO_SEG7 |
LCD_OUTPUT_SEL49_TO_SEG6);

```

```
/* LCD Initialize and calculate real frame rate */
```

```
u32ActiveFPS = LCD_Open(&g_LCDCfg);
```

```
/* Select output voltage level 9 for 4.8V */
```

```
LCD_SET_CP_VOLTAGE(LCD_CP_VOLTAGE_LV_9);
```

```
}
```

LCD\_Start 包括開啟 LCD 屏幕，並使用以下四種 LCD API 點亮屏幕上的內容，點亮結果請參考 1.3 執行結果。

#### 1. LCDLIB\_Printf(uint32\_t u32Zone, char \*InputStr)

- 功能說明：對在 LCDView 中設定成相同"Group name"的七段顯示器或十四段顯示器，輸出字串。
- 參數說明：
  - u32Zone：欲顯示字串區域的名稱。此名稱可在 LCDView 所生成的 lcdzone.h 中找到對應區域的名稱定義。
  - InputStr：欲顯示的字串。

#### 2. LCDLIB\_PrintNumber(uint32\_t u32Zone, uint32\_t InputNum)

- 功能說明：對在 LCDView 中設定成相同"Group name"的七段顯示器或十四段顯示器，輸出數字。
- 參數說明：
  - u32Zone：欲顯示數字區域的名稱。此名稱可在 LCDView 所生成的 lcdzone.h 中找到對應區域的名稱定義。
  - InputStr：欲顯示的數字。

#### 3. LCDLIB\_PutChar(uint32\_t u32Zone, uint32\_t u32Index, uint8\_t u8Ch)

- 功能說明：對在 LCDView 中設定成相同"Group name"的七段顯示器或十四段顯示器，指定位置輸出字元。
- 參數說明：

- u32Zone：欲顯示字串區域的名稱。此名稱可在 LCDView 所生成的 lcdzone.h 中找到對應區域的名稱定義。
- u32Index：欲指定的位置。
- u8Ch：欲輸出的字元。

#### 4. LCDLIB\_SetSymbol(uint32\_t u32Symbol, uint32\_t u32OnOff)

- 功能說明：對在 LCDView 中的單一圖案，設定亮或暗。
- 參數說明：
  - u32Symbol：圖案的名稱。此名稱可在 LCDView 所生成的 lcdzone.h 中找到對應圖案的名稱定義。
  - u32OnOff：填入 1 為顯示指定圖案，填入 0 則為不顯示。

```
void LCD_Start(void)
{
    /* Enable LCD display */
    LCD_ENABLE_DISPLAY();

    /* Set the text to show on LCD. Please refer to the NK-M258KG_EC_LCD.nvt to check the
    definition of the name. */
    /*Show text on the specific group with using LCDLIB_Printf */
    LCDLIB_Printf(ZONE_MAIN_DIGIT, "M258");

    /*Show char on the specific position of the group with using LCDLIB_PutChar() */
    LCDLIB_PutChar(ZONE_MAIN_DIGIT, 4, 'K');
    LCDLIB_PutChar(ZONE_MAIN_DIGIT, 5, 'G');

    /* Show number on the specific group with using LCDLIB_PrintNumber() */
    LCDLIB_PrintNumber(ZONE_TIME_DIGIT, 1352);
    LCDLIB_PrintNumber(ZONE_NUMICRO_DIGIT, 258);
    LCDLIB_PrintNumber(ZONE_PPM_DIGIT, 123);
    LCDLIB_PrintNumber(ZONE_TEMP_DIGIT, 36);
    LCDLIB_PrintNumber(ZONE_VER_DIGIT, 105004);

    /* Show the specific Symbol with using LCDLIB_SetSymbol() */
    LCDLIB_SetSymbol(SYMBOL_COL_9, 1);
    LCDLIB_SetSymbol(SYMBOL_COL_10, 1);
    LCDLIB_SetSymbol(SYMBOL_NVT_11, 1);
    LCDLIB_SetSymbol(SYMBOL_WIFI_12, 1);
    LCDLIB_SetSymbol(SYMBOL_SOUND_14, 1);
    LCDLIB_SetSymbol(SYMBOL_BAT_FRAME_13, 1);
    LCDLIB_SetSymbol(SYMBOL_BAT_1_13, 1);
    LCDLIB_SetSymbol(SYMBOL_BAT_2_13, 1);
    LCDLIB_SetSymbol(SYMBOL_BAT_3_13, 0);
    LCDLIB_SetSymbol(SYMBOL_NUMICRO_15, 1);
    LCDLIB_SetSymbol(SYMBOL_PLUS_19, 1);
    LCDLIB_SetSymbol(SYMBOL_COL_69, 1);
}
```

```

LCDLIB_SetSymbol(SYMBOL_COL_68, 1);
LCDLIB_SetSymbol(SYMBOL_COL_70, 1);
LCDLIB_SetSymbol(SYMBOL_COL_71, 1);
LCDLIB_SetSymbol(SYMBOL_COL_72, 1);
LCDLIB_SetSymbol(SYMBOL_COL_73, 1);
LCDLIB_SetSymbol(SYMBOL_COL_74, 1);
LCDLIB_SetSymbol(SYMBOL_COL_75, 1);
LCDLIB_SetSymbol(SYMBOL_COL_76, 1);
LCDLIB_SetSymbol(SYMBOL_COL_77, 1);
LCDLIB_SetSymbol(SYMBOL_A_29, 1);
LCDLIB_SetSymbol(SYMBOL_S01_47, 1);
LCDLIB_SetSymbol(SYMBOL_S07_47, 1);
LCDLIB_SetSymbol(SYMBOL_S13_47, 1);
LCDLIB_SetSymbol(SYMBOL_S19_47, 1);
LCDLIB_SetSymbol(SYMBOL_S25_47, 1);
LCDLIB_SetSymbol(SYMBOL_S29_47, 1);
LCDLIB_SetSymbol(SYMBOL_S33_47, 1);
LCDLIB_SetSymbol(SYMBOL_S37_47, 1);
LCDLIB_SetSymbol(SYMBOL_CIRCLE_UP_48, 1);
LCDLIB_SetSymbol(SYMBOL_ARROW_UP_48, 1);
LCDLIB_SetSymbol(SYMBOL_TEMP_C_57, 1);
LCDLIB_SetSymbol(SYMBOL_PPM_53, 1);
LCDLIB_SetSymbol(SYMBOL_VERSION_59, 1);
LCDLIB_SetSymbol(SYMBOL_P_66, 1);
LCDLIB_SetSymbol(SYMBOL_P_67, 1);

```

```

}

```

### 3. 軟體與硬體需求

#### 3.1 軟體需求

- BSP 版本
  - M251\_M252\_M254\_M256\_M258\_Series\_BSP\_CMSIS\_V3.02.003
- IDE 版本
  - Keil uVersion 5.37.0.0
- LCDView 版本 V1.05.0004
- Nu-Link Keil Driver 版本 V3.09.7443r

#### 3.2 硬體需求

- 電路元件
  - NuMaker-M258KG V1.1
  - NuMaker-TNLCDSUB\_M258K V1.0
- Nu-Link2-Me
  - Firmware Version V3.10.74xx
- 示意圖
  - 如圖 3-1 將 NuMaker-TNLCDSUB\_M258K 連接至 NuMaker-M258KG V1.1，並如圖 3-2 將 Nu-Link2-Me 以 USB 線與 PC 相連。

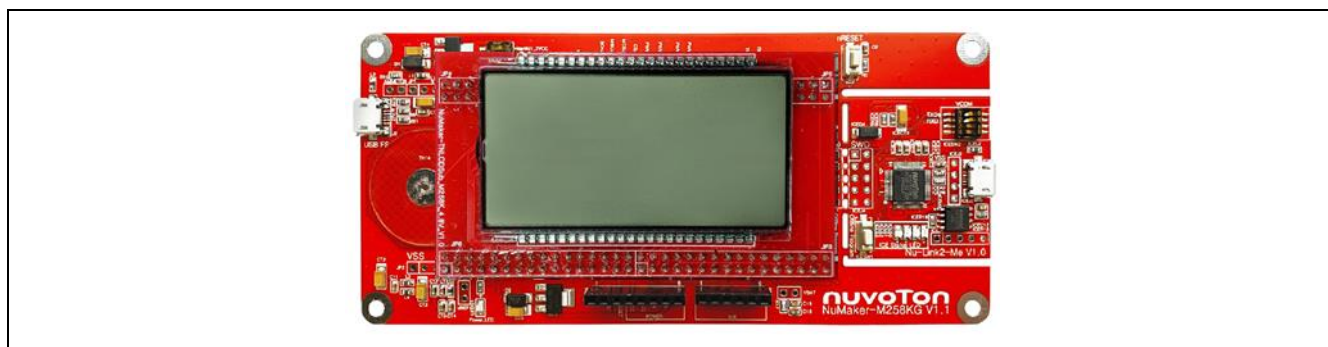


圖 3-1 NuMaker-TNLCDSUB\_M258K 和 NuMaker-M258KG V1.1 連接



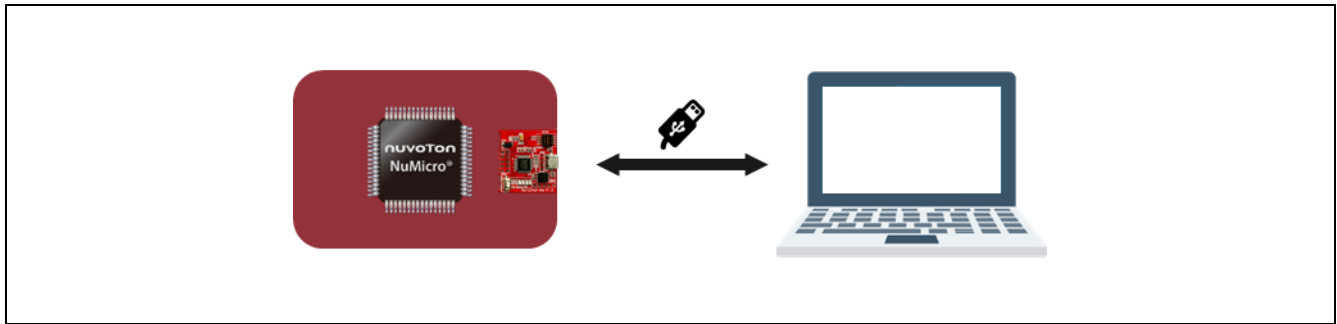


圖 3-2 將 Nu-Link2-Me 以 USB 線與 PC 相連

## 4. 目錄資訊










 EC_M258_LCDView_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
 LCDLib	Library for controlling LCD module
 StdDriver	All peripheral driver header and source files
 SampleCode	
 ExampleCode	Source file of example code
 NK-M258KG_EC_LCD.nvt	Sample file of LCDView for NuMaker-M258KG

圖 4-1 目錄資訊

## 5. 範例程式執行

1. 根據目錄資訊章節進入 ExampleCode 路徑中的 KEIL 資料夾，雙擊 M258\_LCD\_View.uvprojx。
2. 進入編譯模式介面
  - 編譯
  - 下載代碼至記憶體
  - 進入 / 離開除錯模式
3. 進入除錯模式介面
  - 執行代碼

## 6. 修訂紀錄

Date	Revision	Description
2022.11.30	1.00	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.**

---

*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.*