

## 使用M483 I<sup>2</sup>C 介面驅動KA32183A 9x9 LED 矩陣

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 M483 I <sup>2</sup> C 介面控制 KA32183A 驅動 9x9 LED 矩陣顯示器
BSP 版本	M480_Series_BSP_CMSIS_V3.05.005
開發平台	NuMaker-M483KG V1.1 NuMaker-KA32183A_Extension_Board V1.0

*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

本範例代碼使用 M483 I<sup>2</sup>C 介面控制 LED 驅動 IC( KA32183A) 驅動 81 點 LED 矩陣顯示器，且基於資料庫管理 LED 燈效與演示時序並實現在 NuMaker-KA32183A\_Extension\_Board 上。

KA32183A 是一個 81 點(9x9) 矩陣 LED 驅動器芯片，可提供簡單的 LED 燈效設計解決方案，在各種照明應用中實現吸引人的照明效果。 KA32183A 開發平台提供 PC-GUI 工具來生成 LED 圖案，然後將其轉換為 HEX 格式標頭檔，使用者可在 PC-GUI 工具上來檢視顯示效果。此 PC-GUI 工具可以幫助使用者更輕鬆地完成 LED 效果。欲了解更多詳情，請訪問新唐官網 [KA32183A Website Page](#)。

### 1.1 原理

接下來的章節會描述驅動 KA32183A 的 3 個階段。

1. 使用 PC-GUI 工具(LED\_Driver\_Demo\_Software\_v1.00.exe)來編輯 LED 圖案，操作步驟請參考章節 1.1.1。表 1-1 列出 LED\_DRV\_DEMO\_SW (PC-GUI)\_v1.00 資料夾中相關檔案。

相關資料夾與檔案		描述
資料夾	Pattern	儲存創建的 LED 圖形檔案 (*.ptf)，同時會自動生成相對應的 (*.cmf)。
	Playlist	儲存創建的LED 圖案播放列表檔案 (*.pls)。
	Setup	儲存預設的矩陣設定檔案 (*.mts) 可用來創建新的 LED 圖案。
檔案	CyUSB.dll	請勿刪除、重新命名和更改檔案位置。
	Fx2hid.iic	請勿刪除、重新命名和更改檔案位置。
	LED_Driver_Demo_Software	應用程式執行檔。

表 1-1 LED\_DRV\_DEMO\_SW (PC-GUI)\_v1.00 資料夾檔案列表

2. 使用 PC-GUI 工具(KA3218xPatternConverter\_v1.00.exe)來產生 LED 圖案標頭檔，其預設名稱為 demo\_pattern.h。操作步驟請參考章節 1.1.2。

3. 合併 LED 圖案標頭檔至專案中。操作步驟請參考章節 1.1.3。

編譯與下載程式至 MCU 後，可在 NuMaker-KA32183A\_Extension\_Board 上顯示燈效。

### 1.1.1 使用 LED\_Driver\_Demo\_Software Tool 新增與編輯 LED 圖案

請依照以下步驟說明來新增與編輯自定義的 LED 圖案。

1. 開啟 LED\_Driver\_Demo\_Software.exe 應用程式，如圖 1-1 所示。

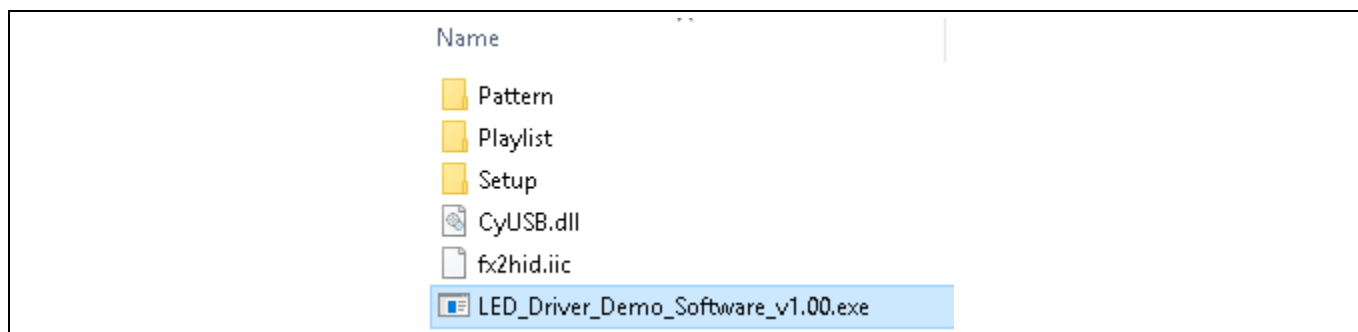


圖 1-1 開啟應用程式

2. 點擊工具列 [Tools] -> [Pattern Editor]，如圖 1-2 所示。

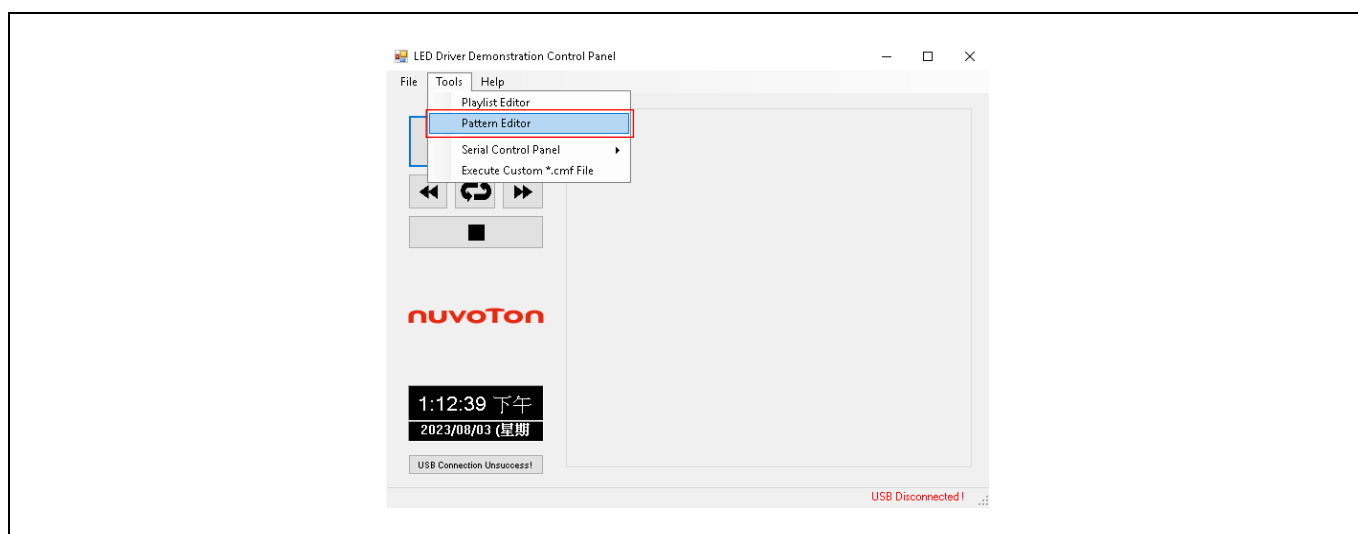


圖 1-2 在主畫面中開啟 [Pattern Editor] 頁面

3. 首次使用時可透過 32183A\_default.mts，來開啟 KA32183A 預設的環境設定。此檔案儲存在 [Setup] 資料夾中，如圖 1-3 所示。

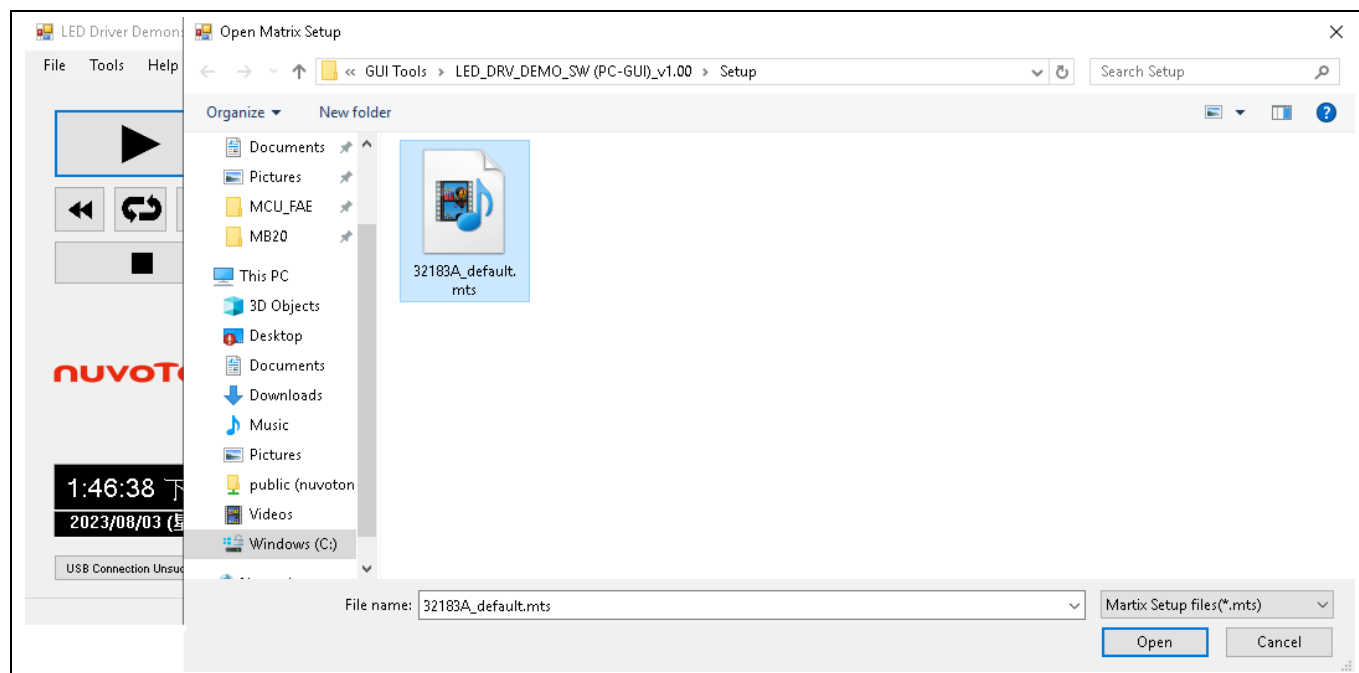


圖 1-3 使用 32183A\_default.mts 來開啟預設環境設定

4. 在 [Pattern Editor] 頁面中，點擊“+”按鈕新增幀視窗與設定每一幀的保持時間(hold time)；使用者可點選紅色框標記區域中的點矩陣來編輯 LED 圖案；藍色框標記區域中則可獨立設定每一顆 LED 的亮度、Firefly 與 Fade-in/out 的效果，如圖 1-4 所示。

[註]紅色框標記區域中的點矩陣由 4 個 9x9 點矩陣組成，亦即支援 4 個 KA32183A IC；黃色框標記區域表示 NuMaker-KA32183A\_Extension\_Board 上的相應區域。點矩陣中的每個點代表一個獨立設定的 LED，“0”表示 LED 亮度設置為“0”。

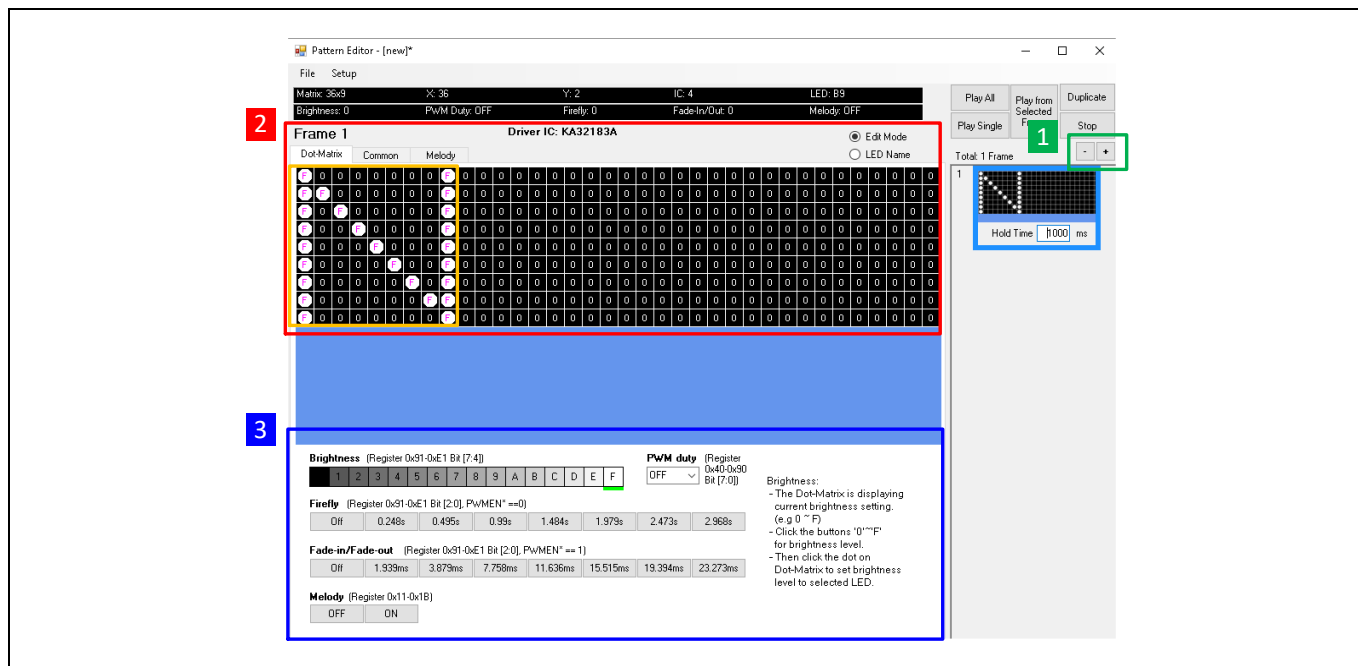


圖 1-4 編輯 LED 圖案

5. 當完成編輯後，儲存檔案(\*.cmf and \*.ptf) 在 [Pattern] 資料夾。

### 1.1.2 使用 KA3218xPatternConverter.exe 將 LED Pattern 轉換成標頭檔

請依照圖 1-5 中操作步驟說明來將 LED Pattern 轉換成標頭檔。

1. 在 KA3218xPatternConverter.exe 中點選 [KA32183A] 標籤頁。
2. 選擇(\*.cmf) 檔案所在資料夾，並導入(\*.cmf) 檔案。
3. 點擊 [Convert] 按鍵來產生相對應的標頭檔，其預設名稱為 demo\_pattern.h。

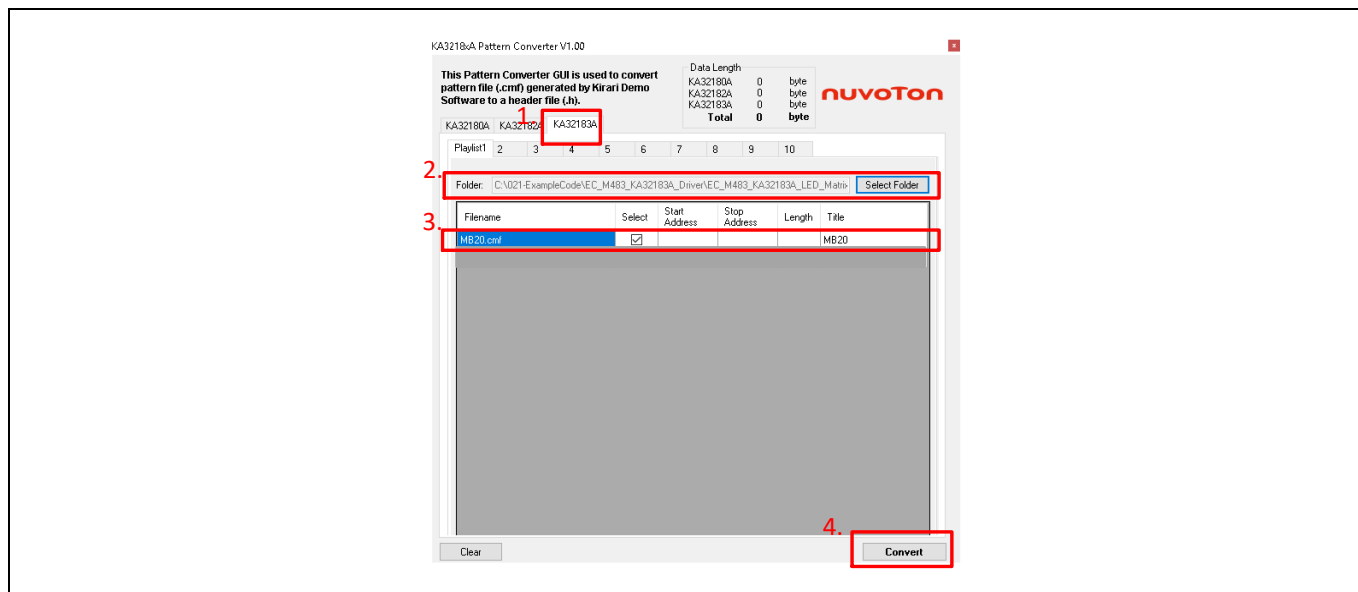


圖 1-5 產生 LED 標頭檔

### 1.1.3 合併 LED 標頭檔至專案中

將 inc 資料夾中的 demo\_pattern.h 替換為 KA3218xPatternConverter.exe 生成的 demo\_pattern.h，檔案路徑如圖 1-6 所示。

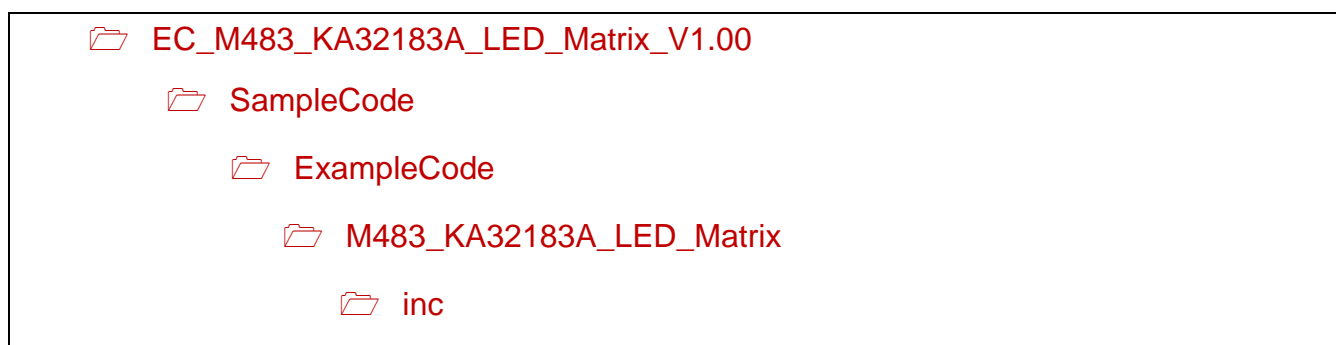


圖 1-6 demo\_pattern.h 檔案路徑

編譯與下載程式至 MCU 後，可在 NuMaker-KA32183A\_Extension\_Board 上顯示燈效。

## 1.2 執行結果

本範例代碼執行的結果如圖 1-7 所示。NuMaker-KA32183A\_Extension\_Board 上依序顯示“NTC♡”。

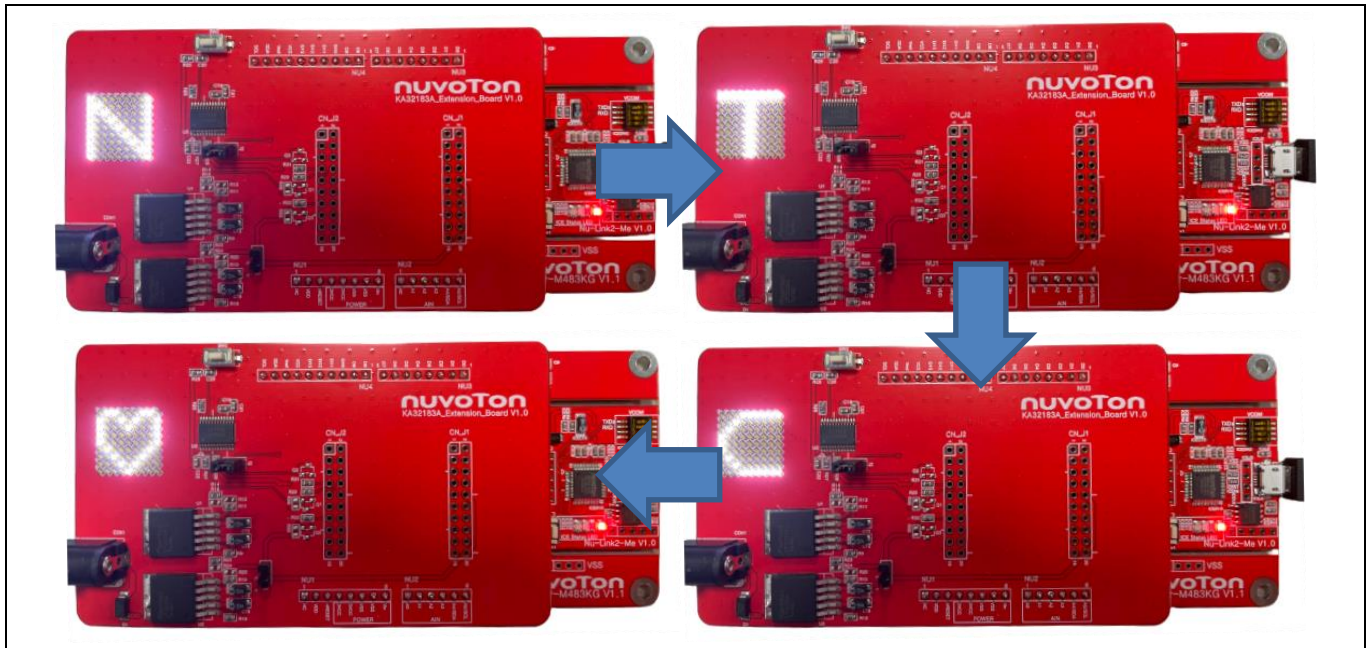


圖 1-7 執行結果

## 2. 代碼介紹

*main.c* · 包含以下 5 個部分：

- **SYS\_Init():** 初始化系統時鐘、外設時鐘與多功能腳位。
- **Timer0\_Init():** 設置 Timer0 每 1ms 產生一次中斷。
- **I2C1\_Init():** 初始化 I<sup>2</sup>C 模組與設置 I<sup>2</sup>C SCL 頻率為 1 MHz。由於 KA32183A 的 SLAVESEL 腳位接至 LOW，亦即 I<sup>2</sup>C 的位址為 0b1011100x。
- **LED\_DeviceDetect():** 透過唯一的 I<sup>2</sup>C 位址來自動偵測 KA32183A 芯片。代碼位於 *led\_app.c*。
- **LED\_APP\_MainCtrl():** 此函示代碼位於 *led\_app.c*，包含以下 2 個功能：
  - 管理狀態機程序控制用於流程控制的狀態機並選擇儲存在二進制模式文件中的播放列表。
  - 透過 I<sup>2</sup>C 介面來設定 KA32183A 暫存器。

```
int32_t main(void)
{
    ...

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

    ...

    /* System tick */
    TIMER0_Init();
    /* Using I2C to control KA32183A */
    I2C1_Init();

    /* Detect device connected base on unique Slave Address of device */
    LED_DeviceDetect();

    while (1)
    {
        /* Polling LED_APP_MainCtrl() every 20 ms */
        if (u8MainLoopFlg)
        {
            u8MainLoopFlg = 0;
            LED_APP_MainCtrl();
        }
    }
}
```



初始化系統後，LED\_APP\_DemoCtrl() 管理 M483 透過 I<sup>2</sup>C 介面來設定 KA32183A 暫存器。代碼位於 *led\_app.c*，包含以下 3 個功能：

- 取得播放清單位址與 LED 圖案資訊。
- 設定 KA32183A 暫存器。
- 控制 LED 圖案顯示序列。

```
void LED_APP_DemoCtrl(UCHAR u8PlaylistNo, UCHAR u8PatternNo)
{
    ...

    switch (u8LED_APP_DemoModeState)
    {
        /* Get the playlist address and pattern information. */
        case (STATE_START):
            LED_GetPlaylistAddr(u8PlaylistNo);
            LED_GetPatternInfo(u8PatternNo);

            u32Cyc = 0;
            u32TempAddr_demo = Demo_PatternInfo.u32TempAddress - 1;
            u32Length = Demo_PatternInfo.u32Length;
            break;

        /* Configure KA32183A register. */
        case (STATE_SINGLEWRITE):
            LED_SingleWrite();
            break;

        /* Control LED pattern sequence. */
        case (STATE_INCLOAD):
            do
            {
                LED_IncLoad();
                u32TempAddr_demo++;
                u8LED_APP_DemoModeState = *(MemoryPtr + (u32TempAddr_demo));
                u32Cyc++;
                u32TempAddr_demo++;
                u16Num = *(MemoryPtr + (u32TempAddr_demo));
                u32Cyc++;
            } while (u8LED_APP_DemoModeState == STATE_INCLOAD);

        case (STATE_INCWRITE):
            LED_IncWrite();
            break;

        /* Wait time between each frame of LED pattern display */
        case (STATE_WAIT):
            if (u8WaitFlgSet == 0)
            {
                LED_Wait();
                u8WaitFlgSet = 1;
            }
    }
}
```

```

    }

    if (u8LEDWaitLoopflg == 1)
    {
        u8LEDWaitLoopflg = 0;
        u8WaitFlgSet = 0;

        if (u32Cyc >= u32Length)
        {
            u8LED_APP_DemoModeState = STATE_END;
        }
    };

    break;

    /* End of demo sequence */
case (STATE_END):
    u8Start = 0;
    u8WaitFlgSet = 0;
    break;

default:
    u8error_status = ERR_DEMOMODESTATE;
    u8Start = 2;      /* Assign for unknown state */
    break;
}

...
}

```

### 3. 軟體與硬體需求

#### 3.1 軟體需求

- BSP 版本
  - M480\_Series\_BSP\_CMSIS\_V3.05.005
- IDE 版本
  - Keil uVersion 5.27

#### 3.2 硬體需求

- 電路元件
  - NuMaker-M483KG V1.1
- 接線示意圖
  - 將 NuMaker-KA32183A\_Extension\_Board 連接至 NuMaker-M483KG (NU1, NU2, NU3 與 NU4)。

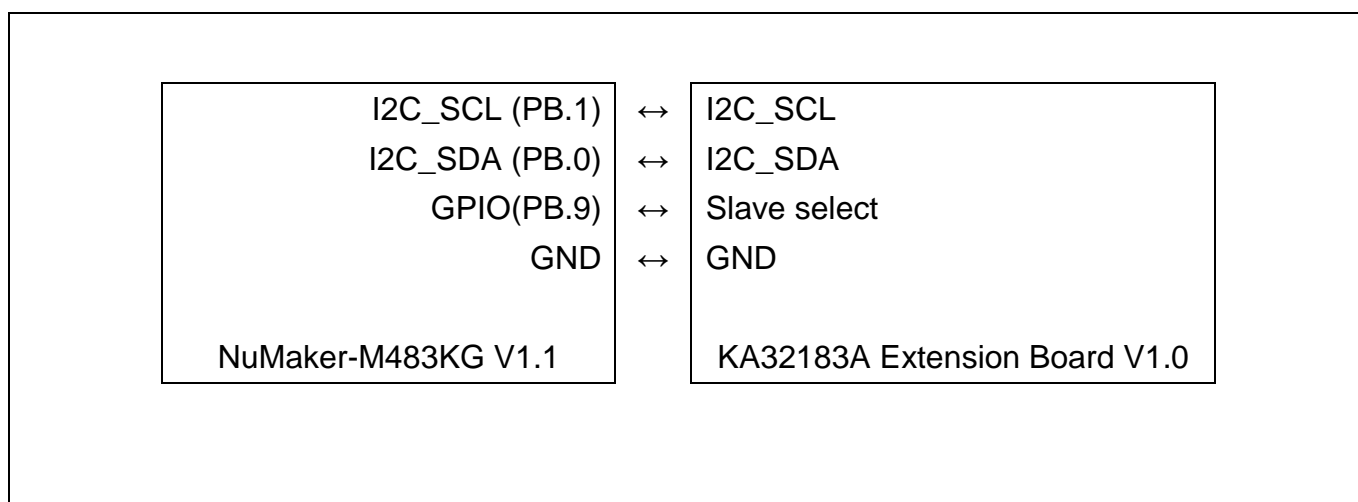


圖 3-1 M483 (I<sup>2</sup>C Master) 和 KA32183A (I<sup>2</sup>C Slave) 連接

## 4. 目錄資訊












 EC_M483_KA32183A_LED_Matrix_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
 KA3218xA Reference GUI Tools	PC-GUI tools for creating customized LED pattern
 GUI Tools	
 LED_DRV_DEMO_SW (PC-GUI)_v1.00	LED_Driver_Demo_Software.exe for creating LED pattern
 Pattern Converter GUI_v1.00	KA3218xPatternConverter.exe for generating LED pattern header file

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 *M483\_KA32183A\_LED\_Matrix.uvproj*。
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
2023.07.31	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.