使用M483 I²C 接口驱动KA32183A 9x9 LED 矩阵

NuMicro® 32位系列微控制器范例代码介绍

文件信息

应用简述	本范例代码使用 M483 I²C 接口控制 KA32183A 驱动 9x9 LED 矩阵显示器
BSP 版本	M483_Series_BSP_CMSIS_V3.05.005
开发平台	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

1. 概述

本范例代码使用 M483 I²C 接口控制 LED 驱动 IC(KA32183A) 驱动 81 点 LED 矩阵显示器，且基于数据库管理 LED 灯效与演示时序并实现在 NuMaker-KA32183A_Extension_Board 上。

KA32183A 是一个 81 点(9x9) 矩阵 LED 驱动器芯片，可提供简单的 LED 灯效设计解决方案，在各种照明应用中实现吸引人的照明效果。 KA32183A 开发平台提供 PC-GUI 工具来生成 LED 图案，然后将其转换为 HEX 格式头文件，使用者可在 PC-GUI 工具上来检视显示效果。此 PC-GUI 工具可以帮助使用者更轻松地完成 LED 效果。欲了解更多详情，请访问新唐官网 KA32183A Website Page。

1.1 原理

接下来的章节会描述驱动 KA32183A 的 3 个阶段。

1. 使用 PC-GUI 工具(LED_Driver_Demo_Software_v1.00.exe)来编辑 LED 图案，操作步骤请参考章节 1.1.1。表 1-1 列出 LED_DRV_DEMO_SW (PC-GUI)_v1.00 文件夹中相关档案。

相关文件夹与档案		描述
文件夹	Pattern	储存创建的 LED 图形档案 (*.ptf)，同时会自动生成相对应的 (*.cmf)。
	Playlist	储存创建的LED 图案播放列表档案 (*.pls)。
	Setup	储存预设的矩阵配置文件案 (*.mts) 可用来创建新的 LED 图案。
档案	CyUSB.dll	请勿删除、重新命名和更改档案位置。
	Fx2hid.iic	请勿删除、重新命名和更改档案位置。
	LED_Driver_Demo_Software	应用程序执行文件。

表 1-1 LED_DRV_DEMO_SW (PC-GUI)_v1.00 文件夹档案列表

2. 使用 PC-GUI 工具(KA3218xPatternConverter_v1.00.exe)来产生 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. 合并 LED 图案头文件至项目中。操作步骤请参考章节 1.1.3

编译与下载程序至 MCU 后，可在 NuMaker-KA32183A_Extension_Board 上显示灯效。

1.1.1 使用 LED_Driver_Demo_Software Tool 新增与编辑 LED 图案

请依照以下步骤说明来新增与编辑自定义的 LED 图案。

1. 开启 LED_Driver_Demo_Software.exe 应用程序，如图 1-1 所示。

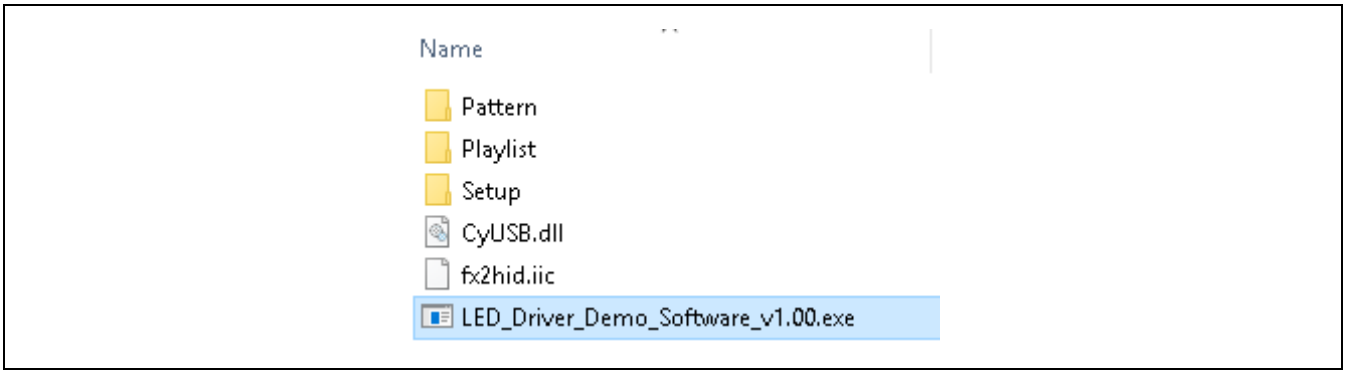

图 1-1 开启应用程序

2. 点击工具栏 [Tools] -> [Pattern Editor] ，如图 1-2 所示。

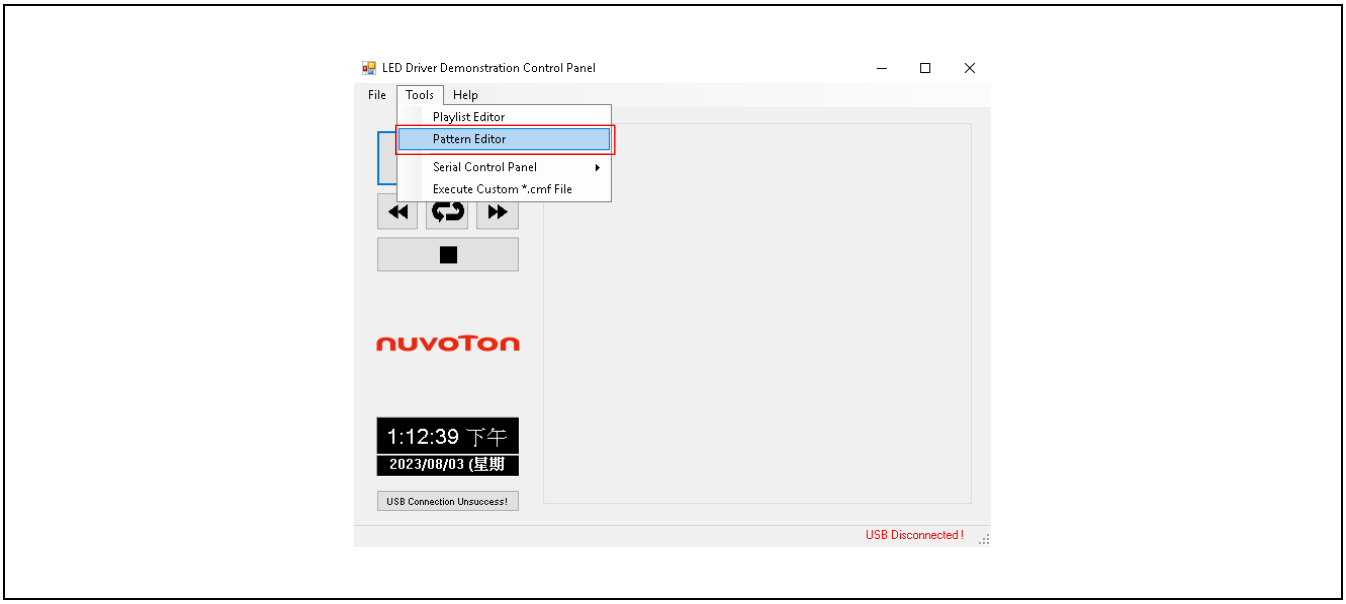

图 1-2 在主画面中开启 [Pattern Editor] 页面

3. 首次使用时可透过 32183A_default.mts，来开启 KA32183A 预设的环境设定。此档案储存在 [Setup] 文件夹中，如图 1-3 所示。

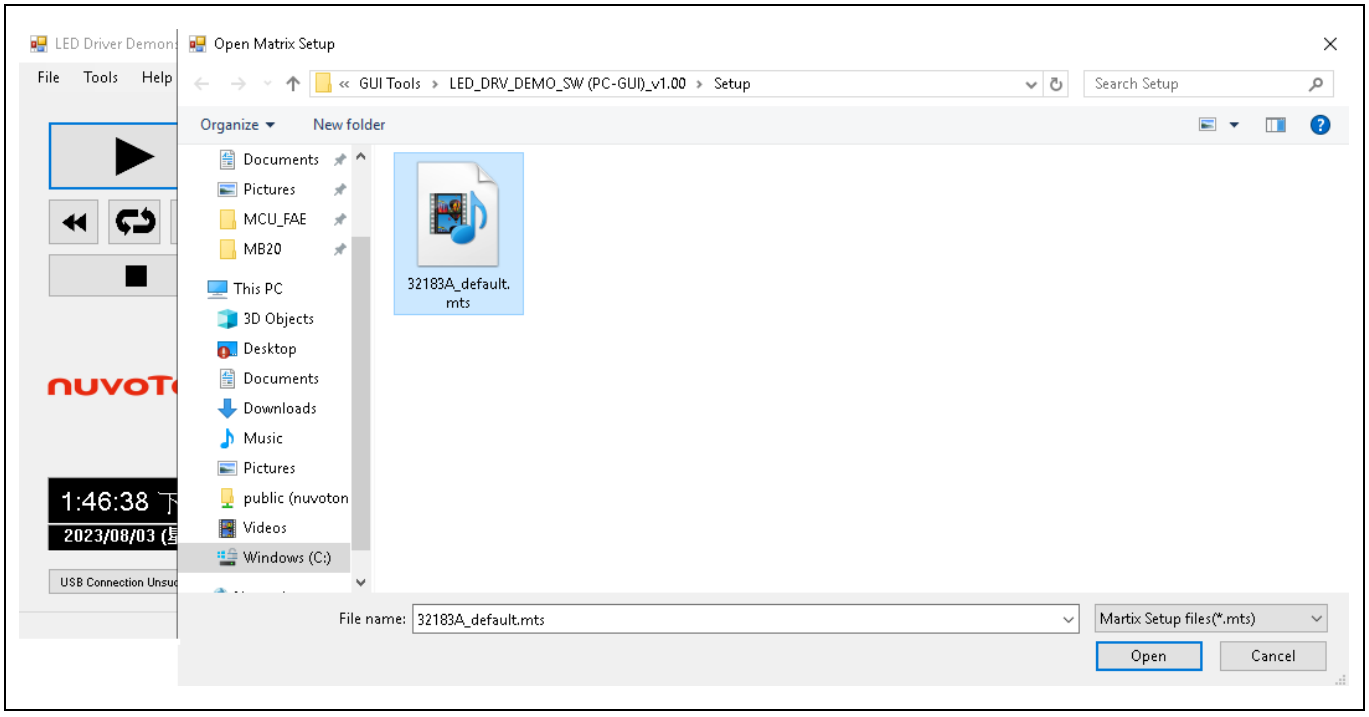

图 1-3 使用 32183A_default.mts 来开启预设环境设定

4. 在 [Pattern Editor] 页面中，点击 “+” 按钮新增帧窗口与设定每一帧的保持时间(hold time)；用户可点选红色框标记区域中的点矩阵来编辑 LED 图案；蓝色框标记区域中则可独立设定每一颗 LED 的亮度、Firefly 与 Fade-in/out 的效果，如图 1-4 所示。

[注]红色框标记区域中的点矩阵由 4 个 9x9 点矩阵组成，亦即支援 4 个 KA32183A IC；黄色框标记区域表示 NuMaker-KA32183A_Extension_Board 上的相应区域。点矩阵中的每个点代表一个独立设定的 LED， “0” 表示 LED 亮度设置为 “0”。

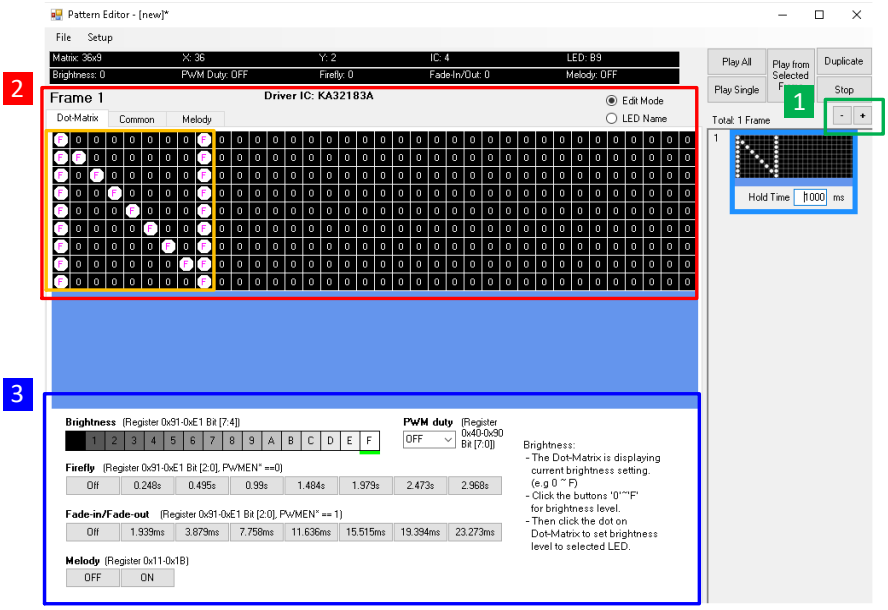

图 1-4 编辑 LED 图案

5. 当完成编辑后，储存档案(*.cmf and *.ptf) 在 [Pattern] 文件夹。

1.1.2 使用 KA3218xPatternConverter.exe 将 LED Pattern 转换成头文件

请依照图 1-5 中操作步骤说明来将 LED Pattern 转换成头文件。

1. 在 KA3218xPatternConverter.exe 中点选 [KA32183A] 标签页。
2. 选择(*.cmf) 档案存放的文件夹，並導入(*.cmf) 檔案。
3. 点击 [Convert] 按键来产生相对应的头文件，其预设名称为 demo_pattern.h。

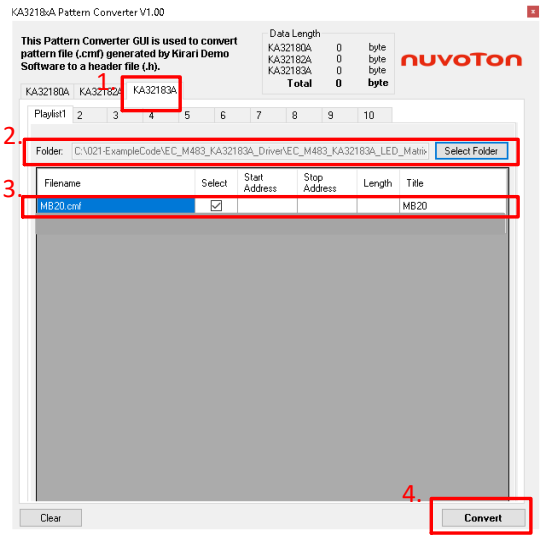

图 1-5 产生 LED 头文件

1.1.3 合并 LED 头文件至专案中

将 inc 文件夹中的 demo_pattern.h 替换为 KA3218xPatternConverter.exe 生成的 demo_pattern.h，档案路径如图 1-6 所示。

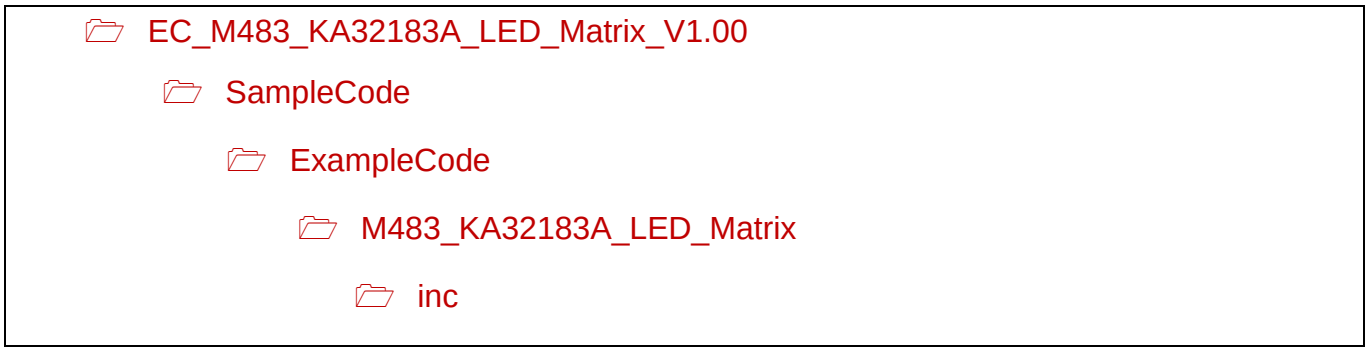

图 1-6 demo_pattern.h 檔案路徑

编译与下载程序至 MCU 后，可在 NuMaker-KA32183A_Extension_Board 上显示灯效。

1.2 执行结果

本范例代码执行的结果如图 1-7 所示。NuMaker-KA32183A_Extension_Board 上依序显示“NTC♡”。

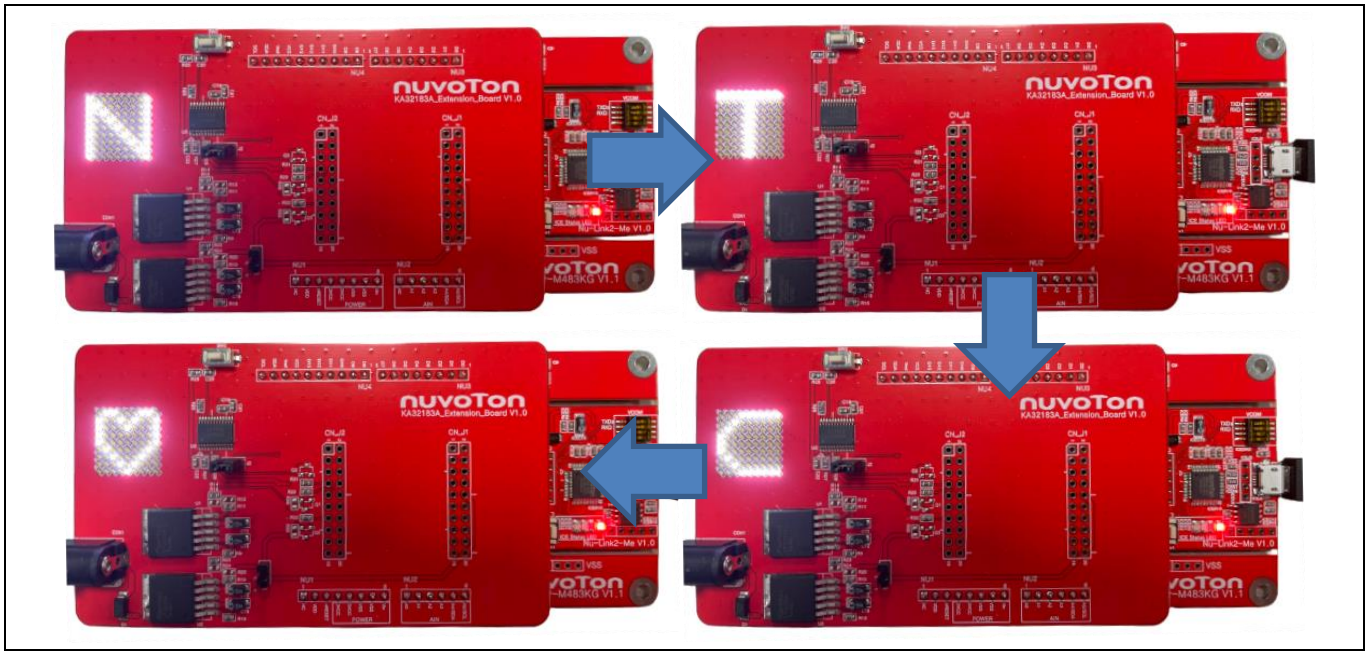

图 1-7 执行结果

2. 代码介绍

main.c，包含以下 5 个部分：

- SYS_Init(): 初始化系统时钟、外设时钟与多功能脚位。
- Timer0_Init(): 设置 Timer0 每 1ms 产生一次中断。
- I2C1_Init(): 初始化 I²C 模块与设置 I²C SCL 频率为 1 MHz。由于 KA32183A 的 SLAVESEL 脚位接至 LOW，亦即 I²C 的地址为 0b1011100x。
- LED_DeviceDetect(): 透过唯一的 I²C 地址来自动侦测 KA32183A 芯片。代码位于 *led_app.c*。
- LED_APP_MainCtrl(): 此函示代码位于 *led_app.c*，包含以下 2 個功能：
  - 管理状态机过程控制用于流程控制的状态机并选择存储在二进制模式文件中的播放列表。
  - 透过 I²C 接口来设定 KA32183A 缓存器。

```
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();
        }
    }
}
```

初始化系统后，LED_APP_DemoCtrl() 管理 M483 透过 I²C 接口来设定 KA32183A 缓存器。代码位于 *led_app.c*，包含以下 3 个功能：

- 取得播放列表地址与 LED 图案信息。
- 设定 KA32183A 缓存器。
- 控制 LED 图案显示序列。

```
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. 软件与硬件需求

3.1 软件需求

- BSP 版本
  - M480_Series_BSP_CMSIS_V3.05.005
- IDE 版本
  - Keil uVersion 5.27

3.2 硬件需求

- 电路组件
  - NuMaker-M483KG V1.1
- 线路示意图
  - 将 NuMaker-KA32183A_Extension_Board 连接至 NuMaker-M483KG (NU1, NU2, NU3 与 NU4)。

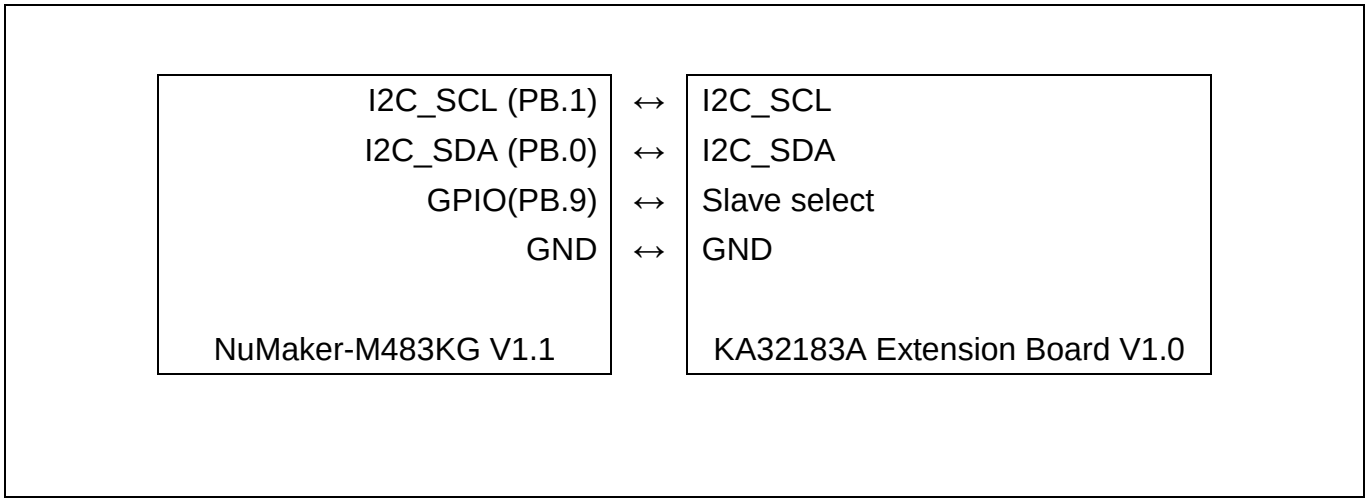

图 3-1 M483 (I^2C Master) 和 KA32183A (I^2C Slave) 连接

4. 目录信息

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

图 4-1 目录信息

5. 范例程序执行

1. 根据目录信息章节进入 ExampleCode 路径中的 KEIL 文件夹，双击 *M483_KA32183A_LED_Matrix.uvproj*。
2. 进入编译模式界面
   - 编译
   - 下载代码至内存
   - 进入 / 离开仿真模式
3. 进入仿真模式界面
   - 执行代码

6. 修订纪录

Date	Revision	Description
2023.07.31	1.00	初始发布。

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.