



RGB LED 8×8 模組

# **BMD26M088**

## **Arduino Library V1.0.1 說明**

版本：V1.21 日期：2023-11-27

[www.bestmodulescorp.com](http://www.bestmodulescorp.com)

## 目錄

簡介 .....	3
Arduino Library .....	3
Arduino Lib 下載及安裝.....	8
Arduino 範例 .....	10
範例 1: blink.....	10
範例 2: cascade4.....	11
範例 3: gradient.....	12
範例 4: colorful .....	13

## 簡介

BMD26M088 是倍創推出的 RGB LED 8×8 模組，使用 I<sup>2</sup>C 通訊方式。本文檔對 BMD26M088 的 Arduino Lib 函式、Arduino Lib 安裝方式進行說明；範例演示了燈板顯示功能。

## Arduino Library

Arduino Lib 名稱：BMD26M088		Lib 版本：V1.0.1
<b>構造函式 &amp; 初始化 &amp; 重置</b>		
1	BMD26M088(TwoWire *theWire=&Wire)	
	描述	構造函式，使用 Wire 通訊介面
	參數	*theWire：選擇 Wire 介面 (預設 Wire 介面)
	返回值	—
	備註	—
2	void begin(uint32_t i2c_addr=0x67, uint32_t i2c_Clock=400000)	
	描述	模組初始化
	參數	i2c_addr：I <sup>2</sup> C 從機位址，預設位址 0x67 i2c_Clock：I <sup>2</sup> C 通訊速率，預設 400kHz
	返回值	void
	備註	模組內部預設設定亮度為最亮，恒流率為 6mA
<b>功能函式</b>		
3	void reset(uint8_t i2c_addr=0x67)	
	描述	模組重置
	參數	i2c_addr：I <sup>2</sup> C 通訊位址，預設位址 0x67
	返回值	void
	備註	—
4	bool isConnected(uint8_t i2c_addr)	
	描述	判斷 I <sup>2</sup> C 通訊是否連接成功
	參數	i2c_addr：I <sup>2</sup> C 通訊位址
	返回值	連接狀態： true：連接成功 false：連接失敗
	備註	—
5	void writeCmd(uint8_t i2c_addr, uint8_t cmd, uint8_t data)	
	描述	寫命令
	參數	i2c_addr：I <sup>2</sup> C 通訊位址 cmd：命令 data：資料
	返回值	void
	備註	cmd 見 HT16D33B 規格書

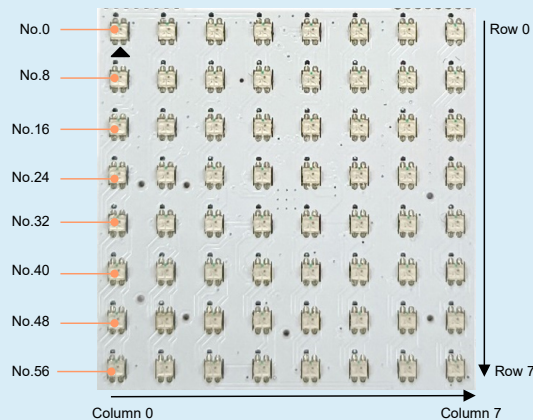
6	uint8_t readCmd(uint8_t i2c_addr,uint8_t cmd)	
	描述	讀取暫存器的命令
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 cmd : 命令
	返回值	命令下讀取的回傳值
	備註	cmd 見 HT16D33B 規格書
7	void setOverTempertureProtect(uint8_t i2c_addr,uint8_t state,bool auto_control=true)	
	描述	設定晶片內部過溫保護功能
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 state : 過溫保護開關 1 : 打開 0 : 關閉 auto_control : 觸發過溫保護後自動關閉 LED 顯示 ( 預設為 true) true : 自動關閉 false : 不自動關閉
	返回值	void
	備註	進入過溫保護的閾值為 150°C · 退出過溫保護狀態的閾值為 125°C
8	bool getOverTemperatureFlag(uint8_t i2c_addr)	
	描述	獲取過溫觸發狀態標誌位
	參數	i2c_addr : I <sup>2</sup> C 通訊位址
	返回值	過溫觸發狀態 true : 觸發 false : 未觸發
	備註	過溫高於 150°C 標誌位置高 · 低於 125°C 清零 ; 當設定為過溫不自動關閉 · 則需要讀取標誌確認是否關閉 LED 顯示
9	void setLedRAMEnable(uint8_t i2c_addr)	
	描述	啟用 LED 的 RAM 記憶體
	參數	i2c_addr : I <sup>2</sup> C 通訊位址
	返回值	void
	備註	開啟 RAM 後 · 只要往對應 RAM 位址裡寫入資料即可點亮對應的燈珠 請參考使用手冊的 RAM 位址映射章節
10	void setLedRAMDisable(uint8_t i2c_addr)	
	描述	停用 LED 的 RAM 記憶體
	參數	i2c_addr : I <sup>2</sup> C 通訊位址
	返回值	void
	備註	請參考使用手冊的 RAM 位址映射章節
11	void setBrightness(uint8_t i2c_addr, uint8_t brightness)	
	描述	設定全域亮度
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 brightness : 亮度 · 範圍 0~255 · 數值越大亮度越高
	返回值	void
	備註	—

12	void setCurrent(uint8_t i2c_addr, uint8_t level)	
	描述	設定恒流大小
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 level : 恒流率 0(BMD26M088_CCR_3MA): 3mA 1(BMD26M088_CCR_6MA): 6mA 2(BMD26M088_CCR_9MA): 9mA 3(BMD26M088_CCR_12MA): 12mA 4(BMD26M088_CCR_15MA): 15mA 5(BMD26M088_CCR_18MA): 18mA 6(BMD26M088_CCR_21MA): 21mA 7(BMD26M088_CCR_24MA): 24mA 8(BMD26M088_CCR_27MA): 27mA 9(BMD26M088_CCR_30MA): 30mA 10(BMD26M088_CCR_33MA): 33mA 11(BMD26M088_CCR_36MA): 36mA 12(BMD26M088_CCR_39MA): 36mA 13(BMD26M088_CCR_42MA): 42mA 14(BMD26M088_CCR_45MA): 45mA 15(BMD26M088_CCR_48MA): 48mA
	返回值	void
	備註	—
13	void clearAll(uint8_t i2c_addr)	
	描述	清除所有顯示
	參數	i2c_addr : I <sup>2</sup> C 通訊位址
	返回值	void
備註	—	
14	void clearRGB(uint8_t i2c_addr, uint8_t RGB_Number)	
	描述	清除指定 RGB 的顯示
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 RGB_Number : RGB 序號 <sup>(3)</sup> 0~63
	返回值	void
備註	—	
15	void clearRow(uint8_t i2c_addr, uint8_t RowIndex)	
	描述	清除一行 RGB 的顯示
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 RowIndex : 列序號 <sup>(2)</sup> 0~7
	返回值	void
備註	—	
16	void clearColumn(uint8_t i2c_addr, uint8_t ColumnIndex)	
	描述	清除一行 RGB 的顯示
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 ColumnIndex : 行序號 <sup>(1)</sup> 0~7
	返回值	void
備註	—	

17	void writeRGB(uint8_t i2c_addr, uint8_t RGB_Number, uint8_t R, uint8_t G, uint8_t B)	
	描述	點亮指定的 RGB 燈
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 RGB_Number : RGB 序號 <sup>(3)</sup> 0~63 R/G/B : 顏色數值
	返回值	void
	備註	—
18	void writeAllRGB(uint8_t i2c_addr, uint8_t R, uint8_t G, uint8_t B)	
	描述	點亮所有的 RGB 燈
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 R/G/B : 顏色數值
	返回值	void
	備註	—
19	void writeColumn(uint8_t i2c_addr, uint8_t ColumnIndex, uint8_t R, uint8_t G, uint8_t B)	
	描述	點亮一行 RGB 燈
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 ColumnIndex : 行序號 <sup>(1)</sup> 0~7 R/G/B : 顏色數值
	返回值	void
	備註	—
20	void writeRow(uint8_t i2c_addr, uint8_t RowIndex, uint8_t R, uint8_t G, uint8_t B)	
	描述	點亮一列 RGB 燈
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 RowIndex : 列序號 <sup>(2)</sup> 0~7 R/G/B : 顏色數值
	返回值	void
	備註	—
21	void DrawAsciiChar(uint8_t i2c_addr, char Ch, uint8_t R, uint8_t G, uint8_t B)	
	描述	顯示一個 ASCII 字元
	參數	i2c_addr : I <sup>2</sup> C 通訊位址 Ch : ASCII 數值或字元 R/G/B : 顏色數值
	返回值	void
	備註	字元的 ASCII 碼值 · 範圍 32~126 。如 97 對應字元 'a' 。

22	void setGradient(uint8_t i2c_addr,uint8_t mode,uint8_t T1,uint8_t T2=T2_IS_T1,uint8_t T3=T3_IS_T1,uint8_t T4=T4_IS_T1)	
	描述	設定漸變功能
	參數	<p>i2c_addr : I<sup>2</sup>C 通訊位址</p> <p>mode : 漸變的工作模式 (4)</p> <p>0(GFS_GRADIENT_GARMA) : 伽馬強度漸變的漸變模式</p> <p>1(GFS_GRADIENT_LINEAR) : 線性強度漸變的漸變模式</p> <p>2(GFS_BLINK) : 閃爍模式</p> <p>T1 : 漸變 T1 時間 (4) ( 註 : 本模組每幀的時間為 1.813ms)</p> <p>0(GFT_OFF) : 關閉</p> <p>1(GFT_256_FRAME) : 256 幀</p> <p>2(GFT_512_FRAME) : 512 幀</p> <p>3(GFT_1024_FRAME) : 1024 幀</p> <p>4(GFT_1536_FRAME) : 1536 幀</p> <p>5(GFT_2048_FRAME) : 2048 幀</p> <p>6(GFT_2560_FRAME) : 2560 幀</p> <p>7(GFT_256_FRAME) : 3072 幀</p> <p>T3 : 漸變 T3 時間 (4)</p> <p>0(T3_IS_T1) : T1 ( 預設 )</p> <p>1(T3_IS_T1x2) : T1×2</p> <p>T2 : 漸變 T2 時間 (4)</p> <p>0(T2_IS_T1x1_4) : T1×0.25</p> <p>1(T2_IS_T1x1_2) : T1×0.5</p> <p>2(T2_IS_T1) : T1 ( 預設 )</p> <p>3(T2_IS_T1x2) : T1×2</p> <p>T4 : 漸變 T4 時間 (4)</p> <p>0(T4_IS_T1x1_4) : T1×0.25</p> <p>1(T4_IS_T1x1_2) : T1×0.5</p> <p>2(T4_IS_T1) : T1 ( 預設 )</p> <p>3(T4_IS_T1x2) : T1×2</p>
	返回值	void
	備註	詳細內容請見 HT16D33A 規格書

註 : RGB 序號如下 :



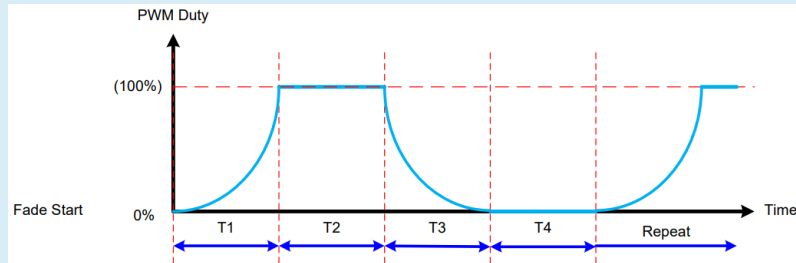
註 1 : 從左到右燈珠行號增加。行號範圍 0~7

註 2 : 從上到下燈珠列號增加。列號範圍 0~7

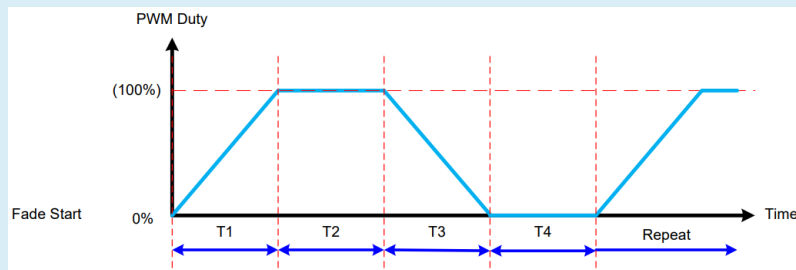
註 3：RGB 序號 = 列號 × 8 + 行號

註 4：漸變模式與時間說明

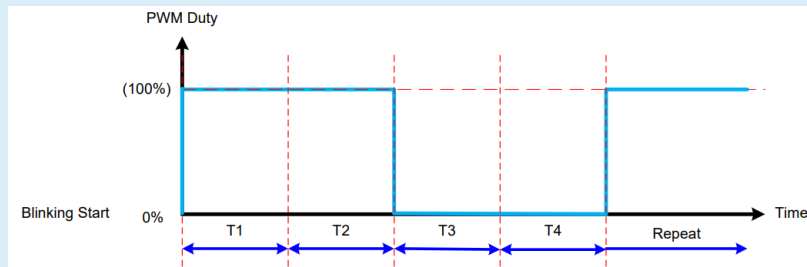
伽馬強度漸變的漸變模式



線性強度漸變的漸變模式



閃爍模式

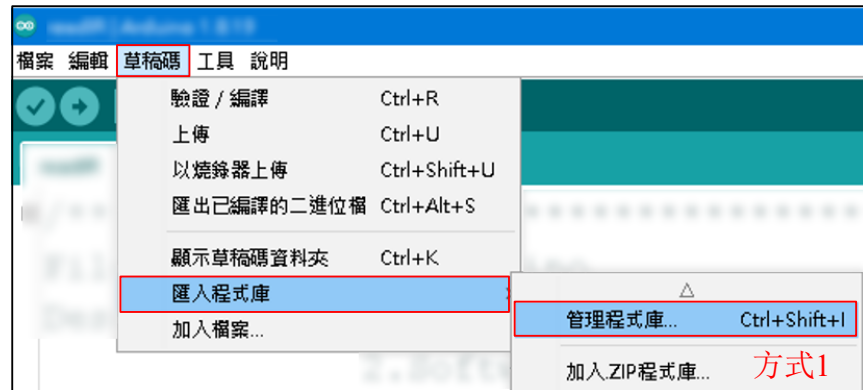


## Arduino Lib 下載及安裝

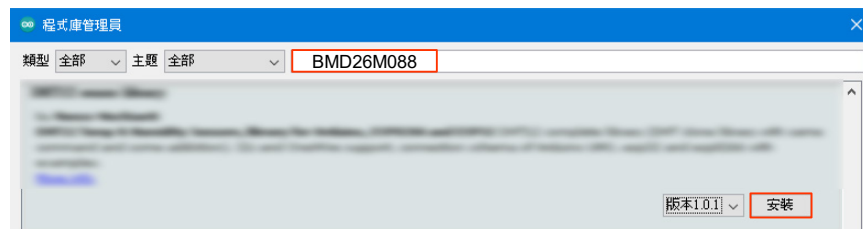
BMD26M088 Library：可參考下面兩種方法安裝 BMD26M088 的 Arduino Library

### 方式 1：搜索安裝

搜索安裝：Arduino IDE → 草稿碼 → 匯入程式庫 → 管理程式庫 ... → 搜索 BMD26M088 → 安裝



搜索安裝流程 1

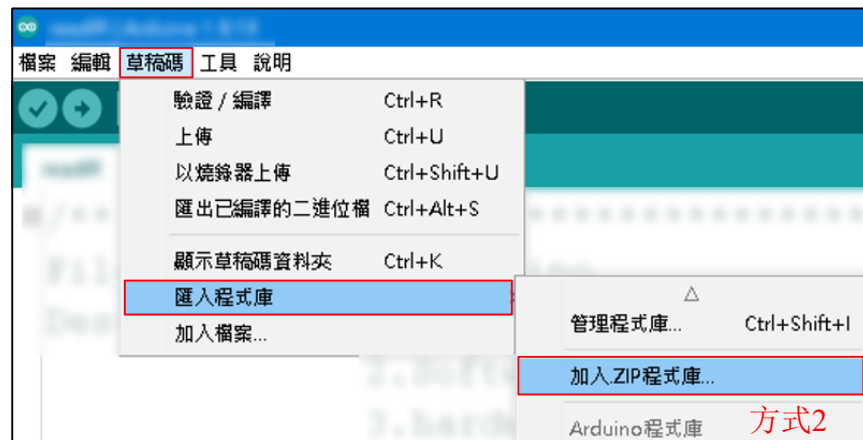


搜索安裝流程 2

### 方式二：添加 .ZIP 程式庫 · 需提前下載 .ZIP 程式庫

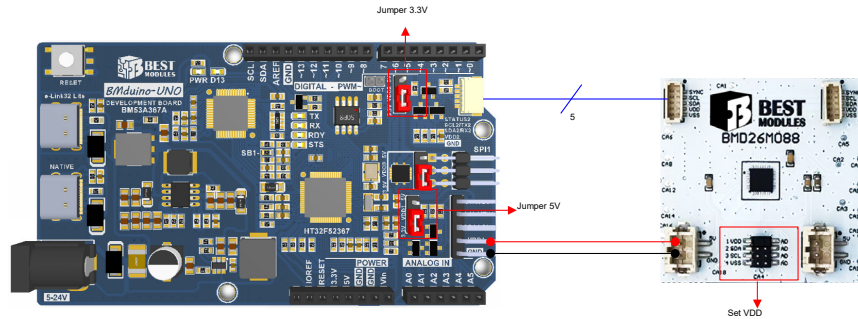
下載方法：打開倍創官方網站 (<https://www.bestmodulescorp.com/bmd26m088.html>) 文件目錄下的 Arduino 範例程式 (BMD26M088 Library)。

添加 .ZIP 程式庫：Arduino IDE → 草稿碼 → 匯入程式庫 → 加入 .ZIP 程式庫 ...



## Arduino 範例

### 範例 1: blink



實物連接示意圖

範例實現功能：所有 RGB 閃爍

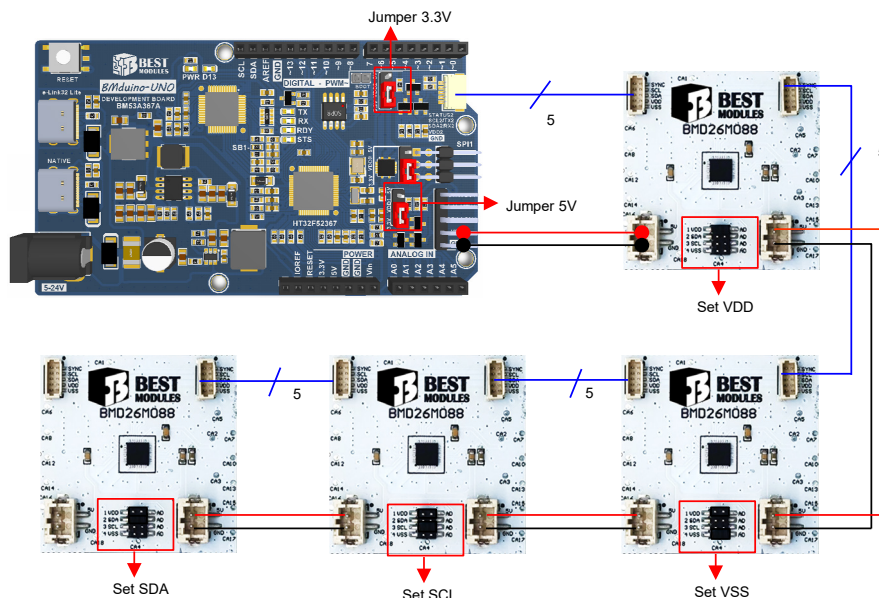
1. 範例打開方式：Arduino IDE → 檔案 → 範例 → Lib 選擇 (BMD26M088) → 選擇範例 (blink)
2. 範例說明：
  - a. 設定對象 & 模組初始化及設定

```
#include <BMD26M088.h>
// 設定 I2C 位址 · 範例採用 I2C 位址跳帽處於 VDD on 位置
BMD26M088 myBMD26M088(&Wire2);
#define ADDRESS BMD26M088_I2C_ADDRESS_VDD
void setup()
{
  myBMD26M088.begin(ADDRESS); // 模組初始化
  myBMD26M088.setBrightness(ADDRESS,0xff); // 設定全域亮度為最大
  myBMD26M088.setCurrent(ADDRESS,BMD26M088_CCR_6MA); // 設定恆流率為 6mA
}
```

- b. 使所有 RGB 閃爍

```
void loop()
{
  // 點亮所有 RGB 為紅色
  myBMD26M088.writeAllRGB(ADDRESS,0xff,0,0);
  delay(200); // 延時 200ms
  myBMD26M088.clearAll(ADDRESS); // 熄滅所有 RGB 燈珠
  delay(200); // 延時 200ms
}
```

## 範例 2: cascade4



實物連接示意圖

範例實現功能：單路 I<sup>2</sup>C 控制級聯的 4 個模組，顯示不同顏色

注意：由於 BMduino 供電能力有限，本次設計在程式中設定恆流率為 6mA。

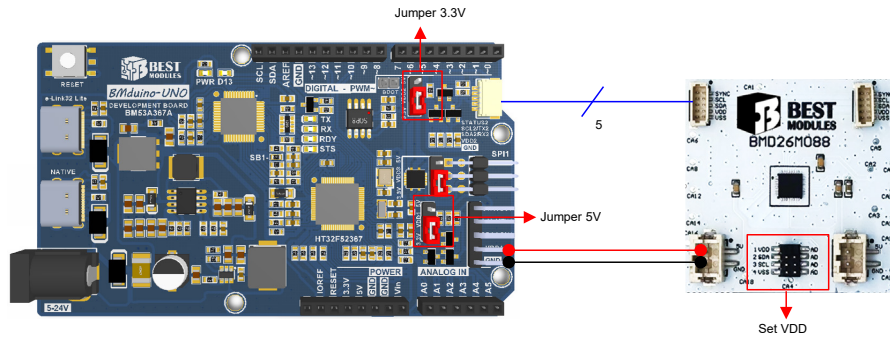
1. 範例打開方式：Arduino IDE → 文件 → 範例 → Lib 選擇 (BMD26M088) → 選擇範例 (cascade4)
2. 範例說明：
  - a. 設定對象 & 模組初始化 & 控制級聯的 4 個模組顯示不同顏色

```
#include "BMD26M088.h"
#include "BMB22T124.h"
BMD26M088 myBMD26M088(&Wire2);
#define i2c_addr_boardcast BMD26M088_I2C_ADDRESS_BOARDCAST// 廣播位址
// 利用跳帽設定 4 個模組分別為不同的 I2C 位址
#define i2c_addr_board1 BMD26M088_I2C_ADDRESS_VDD// 位址 1
#define i2c_addr_board2 BMD26M088_I2C_ADDRESS_GND// 位址 2
#define i2c_addr_board3 BMD26M088_I2C_ADDRESS_SCL// 位址 3
#define i2c_addr_board4 BMD26M088_I2C_ADDRESS_SDA// 位址 4
void setup()
{
    myBMD26M088.begin(i2c_addr_boardcast); // 模組初始化
    myBMD26M088.setBrightness((i2c_addr_boardcast,0xff); // 設定全域亮度
                                // 為最大
    myBMD26M088.setCurrent((i2c_addr_boardcast,BMD26M088_CCR_6MA); // 設定恆流率為 6mA

    // 根據第一個模組的 I2C 位址，設定第一個模組為紅色
    myBMD26M088.writeAllRGB(i2c_addr_board1, 0xff,0,0);
    // 根據第二個模組的 I2C 位址，設定第二個模組為綠色
    myBMD26M088.writeAllRGB(i2c_addr_board2, 0, 0xff,0);
```

```
// 根據第三個模組的 I2C 位址 · 設定第三個模組為藍色
myBMD26M088.writeAllRGB(i2c_addr_board3, 0,0, 0xff);
// 根據第四個模組的 I2C 位址 · 設定第四個模組為黃色
myBMD26M088.writeAllRGB(i2c_addr_board4, 0xff, 0xff,0);
}
```

### 範例 3: gradient



實物連接示意圖

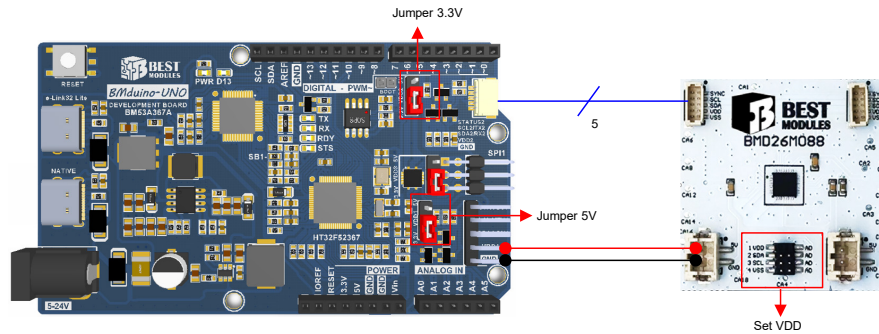
範例實現功能：亮度漸變控制

注意：由於 BMduino 供電能力有限，本次設計在程式中設定恆流率為 6mA。

1. 範例打開方式：Arduino IDE → 文件 → 範例 → Lib 選擇 (BMD26M088) → 選擇範例 (gradient)
2. 範例說明：
  - a. 設定對象 & 模組初始化 & 亮度漸變控制

```
#include <BMD26M088.h>
BMD26M088 myBMD26M088 (&Wire2);
#define ADDRESS BMD26M088_I2C_ADDRESS_VDD
void setup()
{
  myBMD26M088.begin(ADDRESS); // I2C 初始化
  myBMD26M088.setBrightness(ADDRESS,0xff); // 設定亮度
  myBMD26M088.setCurrent(ADDRESS,BMD26M088_CCR_6MA); // 設定恆流率
  // 亮度漸變 RGB 的位置和顏色設定
  //myBMD26M088.writeAllRGB(ADDRESS, 0xff, 0xff, 0xff); // 設定所有 RGB
  myBMD26M088.writeRGB(ADDRESS, 1,0xff, 0, 0); // 設定單個 RGB
  // 設定漸變參數
  myBMD26M088.setGradientControl(ADDRESS, FFEN_ENABLE, GMEN_GARMA,
  GFEN_GLOBE, FOT_T1, FET_T1x1_4, FET_T1x1_2);
  // 全域漸變設定
  myBMD26M088.setGlobeGradient(ADDRESS,GFS_GRADIENT,GFT_1024_FRAME);
}
void loop()
{
}
```

## 範例 4: colorful



實物連接示意圖

範例實現功能：七彩流光顯示

注意：由於 BMduino 供電能力有限，本次設計在程式中設定恆流率為 6mA。

1. 範例打開方式：Arduino IDE → 文件 → 範例 → Lib 選擇 (BMD26M088) → 選擇範例 (colorful)
2. 範例說明：
  - a. 設定對象 & 模組初始化 & 流光控制

```
#include <BMD26M088.h>
BMD26M088 myBMD26M088(&Wire2);
#define ADDRESS BMD26M088_I2C_ADDRESS_VDD // 模組位址

typedef struct RGB_DATA
{
    uint8_t r;
    uint8_t g;
    uint8_t b;
}rgb_def;
rgb_def get_color(int param); // 顏色獲取函式
void setup()
{
    myBMD26M088.begin(ADDRESS); // I2C 初始化
    myBMD26M088.setBrightness(ADDRESS, 0xff); // 設定全域亮度
    myBMD26M088.setCurrent(ADDRESS, BMD26M088_CCR_6MA); // 設定恆流 6mA
}
int flow = 0;
void loop()
{
    u8 i;
    rgb_def rgb;
    for(i = 0; i < 8; i++) // 更新 8 行 RGB
    {
        rgb = get_color(flow + i * (420 / 16)); // 獲取顏色
        myBMD26M088.writeRow(ADDRESS, i, rgb.r, rgb.g, rgb.b); // 更新一
        // 行 RGB
    }
    if((flow++) > 420) {flow = 0;} // 每個迴圈刷新 420 次顏色
}
```

```
// 顏色獲取函式，根據輸入的參數獲取對應的顏色
rgb_def get_color(int param) {
    rgb_def rgb;
    if (param<0) {param+=420;}
    else if (param>419) {param-=420;}
    if (param<60) //step 1:red->yellow
    {
        rgb.r=240;
        rgb.g=4 * param;
        rgb.b=0;
    }
    else if (param<120) //step 2:yellow->green
    {
        rgb.r=240-(4*(param-60));
        rgb.g=240;
        rgb.b=0;
    }
    else if (param<180) //step 3:green->cyan
    {
        rgb.r=0;
        rgb.g=240;
        rgb.b=4*(param-120);
    }
    else if (param<240) //step 4:cyan close to blue
    {
        rgb.r=0;
        rgb.g=240-(3*(param-180));
        rgb.b=240;
    }
    else if (param<300) //step 5:blue->purple
    {
        rgb.r=(param-240);
        rgb.g=60-(param-240);
        rgb.b=240-(2*(param-240));
    }
    else if (param<360) //step6:purple->pink
    {
        rgb.r=60+(param-300);
        rgb.g=0;
        rgb.b=120;
    }
    else if (param<420) //step7:pink->red
    {
        rgb.r=120+(2*(param-360));
        rgb.g=0;
        rgb.b=120-(2*(param-360));
    }
    return rgb;
}
```

Copyright© 2023 by BEST MODULES CORP. All Rights Reserved.

本文件出版時倍創已針對所載資訊為合理注意，但不保證資訊準確無誤。文中提到的資訊僅是提供作為參考，且可能被更新取代。倍創不擔保任何明示、默示或法定的，包括但不限於適合商品化、令人滿意的品質、規格、特性、功能與特定用途、不侵害第三人權利等保證責任。倍創就文中提到的資訊及該資訊之應用，不承擔任何法律責任。此外，倍創並不推薦將倍創的產品使用在會因故障或其他原因而可能會對人身安全造成危害的地方。倍創特此聲明，不授權將產品使用於救生、維生或安全關鍵零組件。在救生 / 維生或安全應用中使用倍創產品的風險完全由買方承擔，如因該等使用導致倍創遭受損害、索賠、訴訟或產生費用，買方同意出面進行辯護、賠償並使倍創免受損害。倍創 ( 及其授權方，如適用 ) 擁有本文件所提供資訊 ( 包括但不限於內容、資料、範例、材料、圖形、商標 ) 的智慧財產權，且該資訊受著作權法和其他智慧財產權法的保護。倍創在此並未明示或暗示授予任何智慧財產權。倍創擁有不事先通知而修改本文件所載資訊的權利。如欲取得最新的資訊，請與我們聯繫。