



---

**BMduino-Shield**

語音播放擴充板

**BMV31T001**

**Arduino Library V1.0.1 說明**

版本：V1.10 日期：2024-01-24

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

## 目錄

簡介 .....	3
Arduino Library 函式 .....	3
Arduino Lib 下載及安裝 .....	6
Arduino 範例 .....	7
範例 : voiceUpdateAndPlayback .....	7

## 簡介

BMV31T001 是倍創推出的一款語音播放擴充板，採用單線通訊進行播放控制。本文檔對 BMV31T001 的 Arduino Lib 函式、Arduino Lib 安裝方式進行說明；範例演示了線上音源更新，語音播放操作。

## Arduino Library 函式

Arduino Lib 名稱：BMV31T001		Lib 版本：V1.0.1
構造函式 & 初始化		
1	BMV31T001()	
	描述	構造函式
	參數	—
	返回值	—
	備註	—
2	void begin(void)	
	描述	擴充板初始化
	參數	void
	返回值	void
	備註	—
功能函式		
3	void setVolume(uint8_t volume)	
	描述	設定音量
	參數	volume：音量參數，範圍 0~11
	返回值	void
	備註	0 是最小音量（靜音）；總共有 12 級音量調節；內部預設音量是 8
4	void playVoice(uint8_t num, uint8_t loop=0)	
	描述	播放語音
	參數	num: 燒錄音源時由燒錄 PC 工具產生的語音序號 loop : 循環播放控制 0 : 不循環（預設） 1 : 循環
	返回值	void
	備註	—
5	void playSentence(uint8_t num, uint8_t loop=0)	
	描述	播放語句
	參數	num : 燒錄音源時由 PC tool 載入的語句順序 loop : 循環播放控制 0 : 不循環（預設） 1 : 循環
	返回值	void
	備註	—

	void playStop(void)
6	描述   停止播放當前語音 / 語句
	參數   void
	返回值   void
	備註   —
	void playPause(void)
7	描述   暫停播放當前語音 / 語句
	參數   void
	返回值   void
	備註   —
	void playContinue(void)
8	描述   如果有語音 / 句子處於暫停中，則繼續播放，沒有則無效
	參數   void
	返回值   void
	備註   —
	void playRepeat(void)
9	描述   循環播放當前語音 / 語句
	參數   void
	返回值   void
	備註   —
	bool isPlaying(void)
10	描述   檢測是否在播放狀態
	參數   —
	返回值   播放狀態： true : 播放中 false : 不在播放中
	備註   —
	void scanKey(void)
11	描述   掃描按鍵，在 loop 裡面執行
	參數   void
	返回值   void
	備註   —
	bool isKeyAction(void)
12	描述   判斷按鍵狀態有沒有變化
	參數   —
	返回值   按鍵狀態 true : 有變化 false : 沒變化
	備註   按鍵狀態有變化：按下 – 鬆開，鬆開 – 按下

13	<code>uint8_t readKeyValue(void)</code>	
	描述	讀取按鍵狀態
	參數	void
	返回值	按鍵的狀態： 0x00 : 沒按鍵按下 0x01 (BMV31T001_KEY_MIDDLE) : 中鍵有按下 0x02 (BMV31T001_KEY_UP) : 上鍵有按下 0x04 (BMV31T001_KEY_DOWN) : 下鍵有按下 0x08 (BMV31T001_KEY_LEFT) : 左鍵有按下 0x10 (BMV31T001_KEY_RIGHT) : 右鍵按下
	備註	—
	<code>void setPower(uint8_t status)</code>	
14	描述	設定擴充板的語音播放電路電源狀態
	參數	status : 開關狀態 0 (BMV31T001_POWER_DISABLE) : 關閉 1 (BMV31T001_POWER_ENABLE) : 打開
	返回值	void
	備註	—
15	<code>void setLED(uint8_t status)</code>	
	描述	設定狀態指示燈 ( 綠色 LED ) 狀態
	參數	status : LED 狀態 0 : 暗 1 : 亮
	返回值	void
	備註	—
<b>線上音源更新函式</b>		
16	<code>void initAudioUpdate(unsigned long baudrate=256000)</code>	
	描述	初始化線上更新音源
	參數	baudrate : 鮑率 . 預設 256000
	返回值	void
	備註	如果用戶搭配 Arduino UNO 是通過 Serial 更新音源；如果搭配 BMduino 是通過 NATIVE USB 虛擬序列埠更新音源。詳情請參考使用手冊 Arduino 相容性章節
17	<code>bool isUpdateBegin(void)</code>	
	描述	是否有音源需要更新
	參數	void
	返回值	是否有音源需要更新： true : 有音源需要更新 false : 無音源需要更新
	備註	BMduino Voice Widge 上點擊  下載後，此函式可判斷是否開發板有接收到 PC 端的資料。

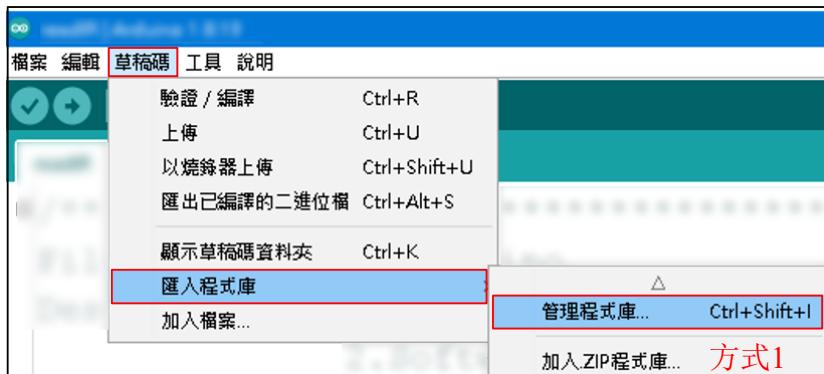
	bool executeUpdate(void)
	描述    更新音源
	參數    void
18	返回值    音源更新情況 true : 更新完成 false : 更新失敗
	備註    在 bool isUpdateBegin(void) 函式後使用 此函式會將接收到的 PC 端資料，通過 SPI 通訊燒錄到擴充板 此函式會執行直到音源更新完成

## ■ Arduino Lib 下載及安裝

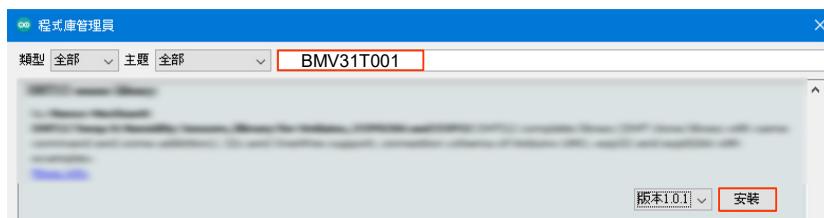
BMV31T001 Library：可參考下面兩種方法安裝 BMV31T001 的 Arduino Library

### 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 BMV31T001 → 安裝



搜索安裝流程 1

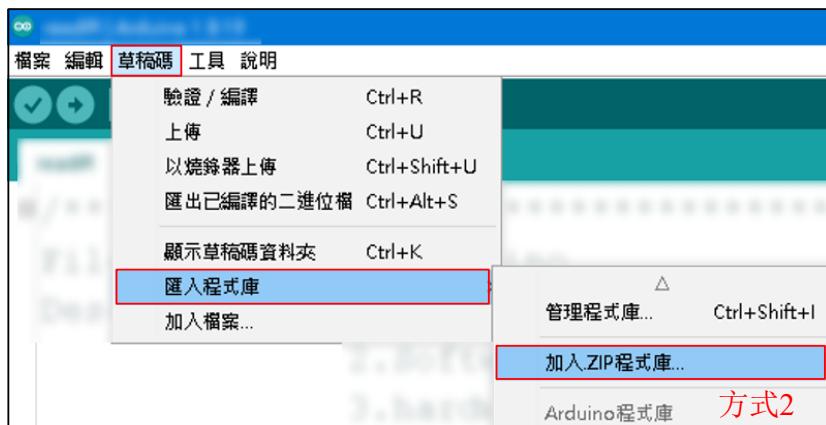


搜索安裝流程 2

### 方式 2：加入 .ZIP 程式庫，需提前下載 .ZIP 程式庫

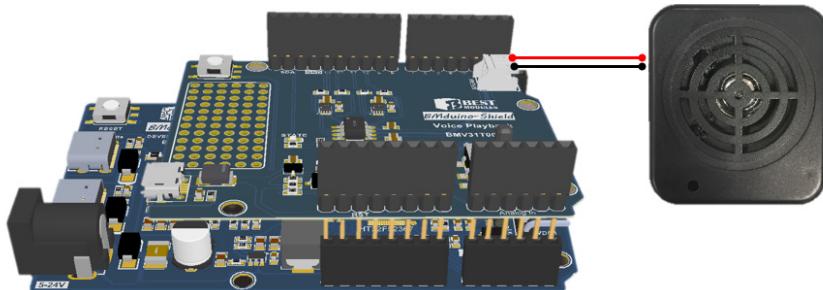
下載方法：打開倍創官方網站 (<https://www.bestmodulescorp.com/bmv31t001.html>) 文件目錄下的 Arduino 範例程式 (BMV31T001 Library)。

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



## II | Arduino 範例

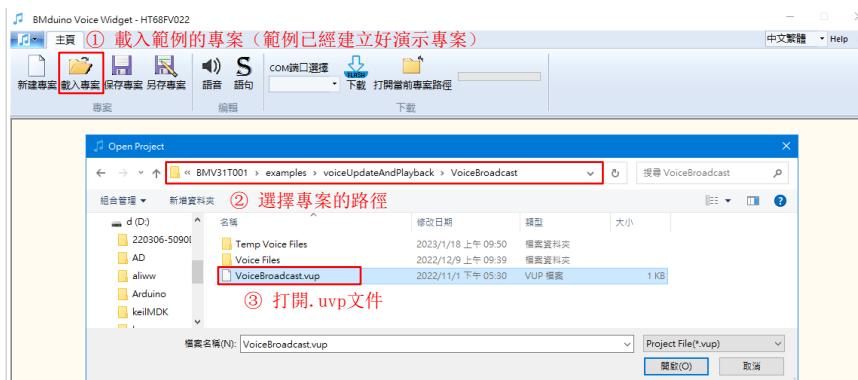
### 範例：voiceUpdateAndPlayback



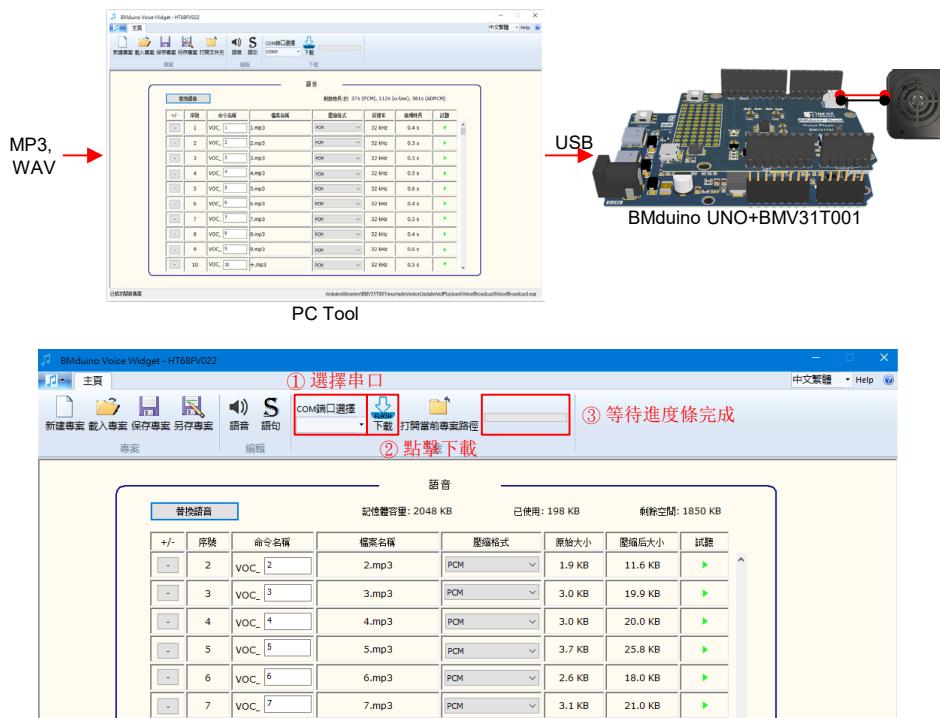
範例實現功能：線上更新音源，並可以通過按鍵控制播放燒錄的音源。

操作步驟：

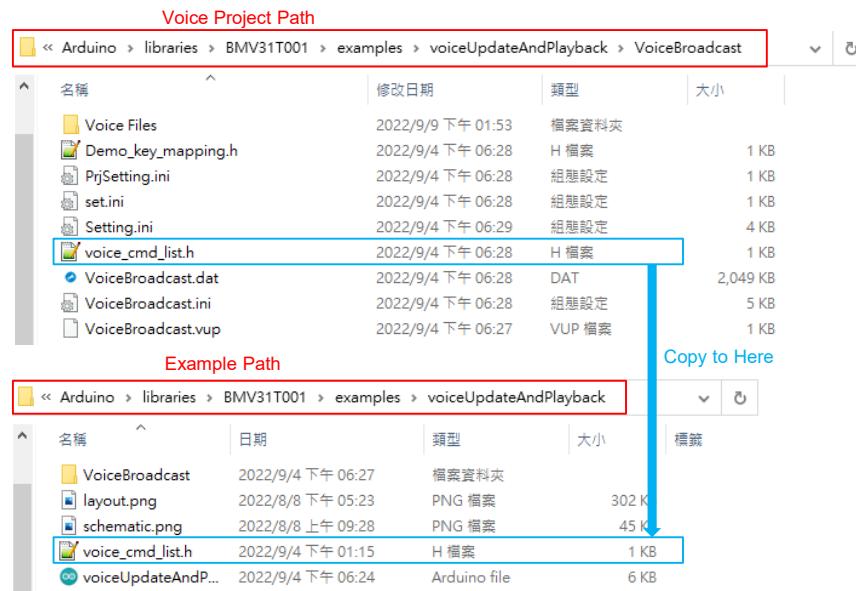
- 範例打開方式：Arduino IDE → 檔案 → 範例 → Lib 選擇 (BMV31T001) → 選擇範例 (voiceUpdateAndPlayback)，並下載到 BMduino UNO 開發板中。
- 打開燒錄音源小工具 BMduino Voice Widget.exe。範例已經建立好語音工程專案，直接選擇範例文件夾下的 .vup，載入本範例的音源處理工程。



3. 將 USB 接入 NATIVE 標識的介面，並在 BMduino Voice Widget.exe 選擇 com 口下載音源。



4. 拷貝頭文件到範例。



5. 操作完以上步驟就可以根據按鍵控制語音播放，範例程式定義的按鍵功能如下表：

按鍵	功能
中	播放 / 停止
上	音量 +

下	音量 -
左	上一首
右	下一首

範例詳解：

● 構建對象 & 初始化擴充板

```
#include "BMV31T001.h"           // 加入擴充板
#include "voice_cmd_list.h"       // 加入平台燒錄的語音資訊列表
BMV31T001 myBMV31T001;          // 建立擴充板對象
#define VOICE_TOTAL_NUMBER 10      // 本次範例測試 10 個語音資料
// 從 "voice_cmd_list.h" 拿出語音名稱建立語音列表
const uint8_t voice_table[VOICE_TOTAL_NUMBER]={
VOC_1,VOC_2,VOC_3,VOC_4,VOC_5,VOC_6,VOC_7,VOC_8,VOC_9,VOC_10};
uint8_t keyStatus;               // 按鍵的狀態
uint8_t lastKeyStatus;           // 按鍵的狀態
uint8_t playStatus;              // 當前播放狀態
uint8_t playNum;                 // 當前播放曲目
#define DEFAULT_VOLUME 6          // 預設音量
uint8_t volume = DEFAULT_VOLUME; // 當前音量
uint8_t keycode = 0;              // 觸發的按鍵編號
void setup() {
    myBMV31T001.begin();          // 初始化擴充板
    myBMV31T001.setPower(BMV31T001_POWER_ENABLE); // 啟用擴充板上電
//=====
// 如果想要線上更新音源，請加入音源更新的序列埠初始化程式
// 支援arduino的serial和BMduinoUNO的USB_NATIVE介面
    myBMV31T001.initAudioUpdate();
//=====
    delay(100);                  // 延時等待擴充板上電完成
    myBMV31T001.setVolume(DEFAULT_VOLUME); // 初始化預設音量 volume
}
```

● 在 loop 中執行音源更新、按鍵掃描、按鍵處理以及播放狀態燈顯示函式

```
void loop() {
//=====
// 如果想要線上更新音源，請加入這段程式
if(myBMV31T001.isUpdateBegin() == BMV31T001_UPDATA_BEGIN)
{// 檢測更新訊號
    myBMV31T001.executeUpdate(); // 執行音源更新
}
//=====
//----- 播放指示燈控制 -----
if(myBMV31T001.isPlaying() == BMV31T001_BUSY)
{ // 檢測播放狀態
    playStatus = BMV31T001_BUSY;
    myBMV31T001.setLED(BMV31T001_LED_ON); // LED 打開
}
else
{
    playStatus = BMV31T001_NOBUSY;
    myBMV31T001.setLED(BMV31T001_LED_OFF); // LED 關閉
}
```

```
//----- 按鍵掃描和處理 -----
myBMV31T001.scanKey();
if(myBMV31T001. isKeyAction() != BMV31T001_NO_KEY) // 判斷是否有按鍵按下
{
    keyStatus = myBMV31T001.readKeyValue(); // 讀取按鍵電位
    if(lastKeyStatus == 0) // 如果上一次按鍵的電位為 0
    {
        if((keyStatus & BMV31T001_KEY_MIDDLE) != 0){ // 中鍵按下
            keycode = BMV31T001_KEY_MIDDLE; // 記錄按下的按鍵
        }
        else if((keyStatus & BMV31T001_KEY_UP) != 0){ // 上鍵按下
            keycode = BMV31T001_KEY_UP; // 記錄按下的按鍵
        }
        else if((keyStatus & BMV31T001_KEY_DOWN) != 0){ // 下鍵按下
            keycode = BMV31T001_KEY_DOWN; // 記錄按下的按鍵
        }
        else if((keyStatus & BMV31T001_KEY_LEFT) != 0){ // 左鍵按下
            keycode = BMV31T001_KEY_LEFT; // 記錄按下的按鍵
        }
        else {
            keycode = BMV31T001_KEY_RIGHT; // 右鍵按下，記錄按下的按鍵
        }
    }
    lastKeyStatus = keyStatus; // 保存上次的按鍵狀態
    switch(keycode) // 判斷按下的按鍵做對應的動作
    {
        case BMV31T001_KEY_MIDDLE: // 中鍵按下
        if(playStatus == BMV31T001_BUSY) // 是否處於播放狀態
        {
            myBMV31T001.playStop(); // 播放中則停止播放
        }
        else
        {
            myBMV31T001.playVoice(voice_table[playNum]); // 未在播放中則進入播放
        }
        keycode = 0;
        break;
        case BMV31T001_KEY_UP: // 上鍵按下
        if(volume < BMV31T001_VOLUME_MAX) // 不處於最大音量
        {
            volume++; // 音量增
            myBMV31T001.setVolume(volume);
        }
        keycode = 0;
        break;
        case BMV31T001_KEY_DOWN: // 下鍵按下
        if(volume > BMV31T001_VOLUME_MIN) // 不處於最小音量
        {
            volume--; // 音量減
            myBMV31T001.setVolume(volume);
        }
    }
}
```

```
        keycode = 0;
        break;
        case BMV31T001_KEY_LEFT:           // 左鍵按下
        if(playNum > 0)                  // 不處於第一首
        {
            playNum--;
            // 上一首
        }
        myBMV31T001.playVoice(voice_table[playNum]); // 播放
        keycode = 0;
        break;
        case BMV31T001_KEY_RIGHT:          // 右鍵按下
        if(playNum < VOICE_TOTAL_NUMBER - 1) // 不處於最大音量
        {
            playNum++;
            // 下一首
        }
        myBMV31T001.playVoice(voice_table[playNum]); // 播放
        keycode = 0;
        break;
        default:
        keycode = 0;
        break;
    }
}
}
```

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）擁有本文件所提供之資訊（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之權利。如欲取得最新的資訊，請與我們聯繫。