



---

**BMduino-UNO 開發板**

**BM53A367A  
使用手冊**

版本：V1.30 日期：2024-08-22

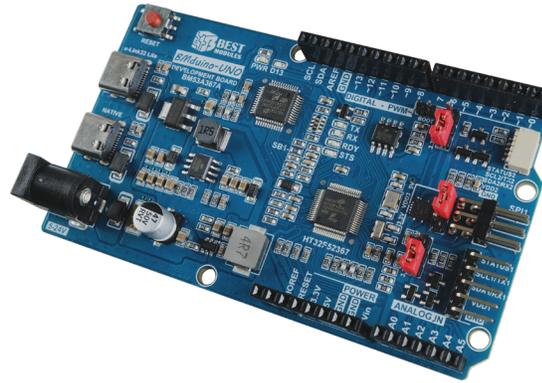
[www.bestmodulescorp.com](http://www.bestmodulescorp.com)

## 目錄

簡介 .....	3
特性 .....	3
方框圖 .....	4
腳位說明 .....	5
技術規格 .....	8
極限參數 .....	8
建議工作條件 .....	9
硬體概述 .....	9
電源 .....	10
LED 指示燈 .....	10
USB 介面電路 .....	11
UART, I <sup>2</sup> C, SPI .....	14
復位電路 .....	17
BM53A367A vs Arduino UNO R3 .....	18
燒錄方法 .....	18
Arduino IDE 軟體 .....	19
Arduino IDE 下載及安裝 .....	19
驅動及其他下載 .....	20
IDE 設置 .....	20
Arduino Library 參考 .....	23
範例 .....	26
Keil IDE 軟體 .....	26
Keil IDE 下載及安裝 .....	26
IDE 設置 .....	26
運行範例 .....	26
問題排查 .....	27
串口未出現在序列埠菜單中 .....	27
範例上傳失敗或凍結 .....	28
尺寸規格 .....	30

## 簡介

BMduino-UNO BM53A367A 是 Holtek 32-bit MCU HT32F52367 的開發板，特別設計成與 Arduino UNO R3 開發板腳位兼容，可支援 Arduino IDE 開發平台和 Keil IDE 開發平台，幫助初學者更容易地學習程式設計。BM53A367A 採用 3.3V 系列 Arm® Cortex®-M0+ 內核的 HT32F52367 為主控 MCU，支援多種供電方式，具備額外的 I<sup>2</sup>C、SPI、UART、USB 等常用通訊介面。

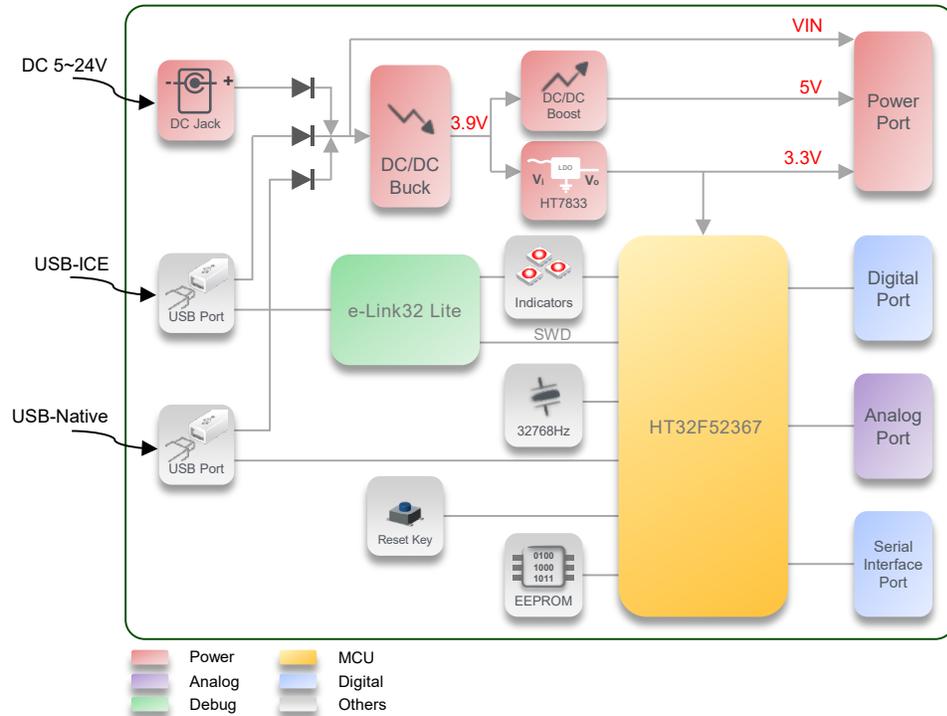


## 特性

- 主控 MCU：HT32F52367 (64-pin LQFP)
  - ◆ Cortex®-M0+，60MHz
  - ◆ Flash 存儲器：256KB
  - ◆ SRAM：32KB
- 31 個數位 I/O 腳位 (其中 17 個可作為 PWM 輸出)
- 7 個類比輸入，A/D 轉換器解析度為 12-bit
- 14 個外部中斷
- EEPROM：4KB
- 通訊介面：UART · SPI · I<sup>2</sup>C · USB
- 電源輸入：USB 介面 ×2 · DC 介面 · 外部 Vin
  - ◆ e-Link32 Lite USB：Type-C USB 介面
  - ◆ Native USB：Type-C USB 介面，支援 BC 1.2 與快充 QC 2.0，快充預設升壓為 12V
  - ◆ DC 介面：圓形介面，連接外部電源 (例如變壓器)，電壓範圍 DC 5~24V
  - ◆ 外部 Vin：Vin 腳位，電壓範圍 DC 5~24V
- 除錯工具：e-Link32 Lite (ICE)，通過開發板上的 e-Link32 Lite 電路進行程式除錯
- 燒錄方式：
  - ◆ ICP (In-Circuit Programming)，通過開發板上的 e-Link32 Lite 電路進行燒錄
  - ◆ IAP (In Application Programming)，通過 COM 序列埠進行燒錄
- 開發環境：支援 Arduino IDE、Keil IDE 兩種開發平台

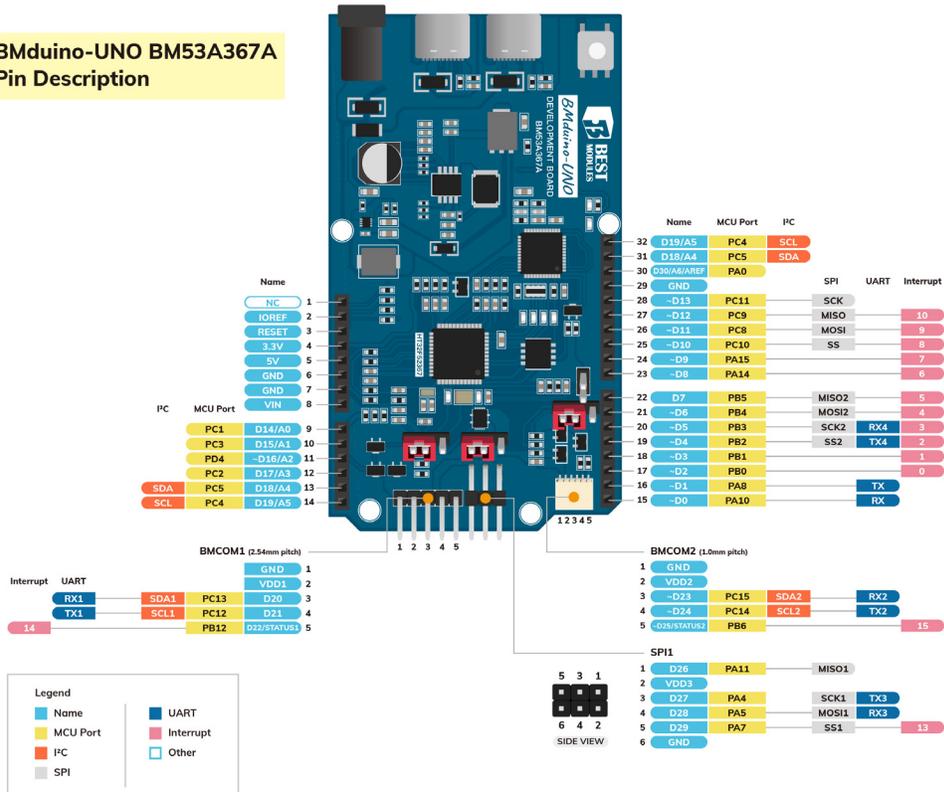
- 介面資源：BMCOM1 ( 腳距 2.54mm · 3.3V/5V 可選 ) · BMCOM2 ( 腳距 1.0mm · 3.3V/5V 可選 ) · SPI1 ( 腳距 2.54mm · 3.3V/5V 可選 )
- 開發板尺寸：53.4mm×93.221mm×14.2mm · 與 Arduino UNO R3 開發板腳位相容
- 開發板重量：28.1g ( 淨重 )

## 方框圖



## 腳位說明

BMduino-UNO BM53A367A  
Pin Description

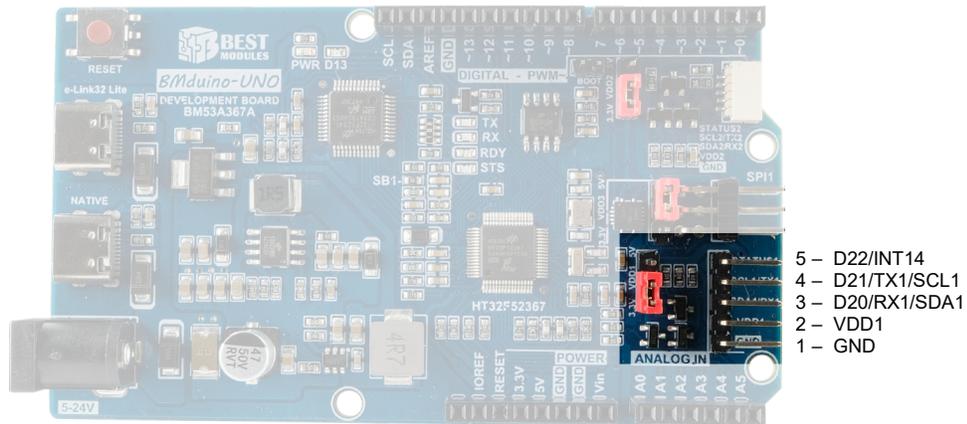


腳位編號	功能	描述
1	NC	未連接
2	IOREF	I/O 邏輯參考電壓 · +3.3V
3	RESET	復位腳位
4	+3V3	+3.3V 電源輸出
5	+5V	+5V 電源輸出
6	GND	電源地
7	GND	電源地
8	Vin	電源輸入 5~24V
9	A0/D14	類比輸入腳位 A0 / 數位腳位 D14
10	A1/D15	類比輸入腳位 A1 / 數位腳位 D15
11	A2/~D16	類比輸入腳位 A2 / 數位腳位 D16 帶 PWM 功能
12	A3/D17	類比輸入腳位 A3 / 數位腳位 D17
13	A4/D18/SDA	類比輸入腳位 A4 / 數位腳位 D18 / I <sup>2</sup> C0 介面的 SDA 腳位
14	A5/D19/SCL	類比輸入腳位 A5 / 數位腳位 D19 / I <sup>2</sup> C0 介面的 SCL 腳位
15	~D0/RX	數位腳位 D0 帶 PWM 功能 / UART0 接收腳位
16	~D1/TX	數位腳位 D1 帶 PWM 功能 / UART0 發送腳位
17	~D2/INT0	數位腳位 D2 帶 PWM 功能 / 外部中斷 INT0
18	~D3/INT1	數位腳位 D3 帶 PWM 功能 / 外部中斷 INT1

腳位編號	功能	描述
19	~D4/TX4/SS2/ INT2	數位腳位 D4 帶 PWM 功能 / UART4 發送腳位 / SPI2 介面的 SS2 腳位 / 外部中斷 INT2
20	~D5/RX4/ SCK2/INT3	數位腳位 D5 帶 PWM 功能 / UART4 接收腳位 / SPI2 介面的 SCK2 腳位 / 外部中斷 INT3
21	~D6/MOSI2/ INT4	數位腳位 D6 帶 PWM 功能 / SPI2 介面的 MOSI2 腳位 / 外部中斷 INT4
22	D7/MISO2/ INT5	數位腳位 D7 / SPI2 介面的 MISO2 腳位 / 外部中斷 INT5
23	~D8/INT6	數位腳位 D8 帶 PWM 功能 / 外部中斷 INT6
24	~D9/INT7	數位腳位 D9 帶 PWM 功能 / 外部中斷 INT7
25	~D10/SS/ INT8	數位腳位 D10 帶 PWM 功能 / SPI0 介面的 SS 腳位 / 外部中斷 INT8
26	~D11/MOSI/ INT9	數位腳位 D11 帶 PWM 功能 / SPI0 介面的 MOSI 腳位 / 外部中斷 INT9
27	~D12/MISO/ INT10	數位腳位 D12 帶 PWM 功能 / SPI0 介面的 MISO 腳位 / 外部中斷 INT10
28	~D13/SCK	數位腳位 D13 帶 PWM 功能 / SPI0 介面的 SCK 腳位
29	GND	電源地
30	AREF/A6/D30	類比參考電壓 / 類比輸入腳位 A6 / 數位腳位 D30
31	A4/D18/SDA	類比輸入腳位 A4 / 數位腳位 D18 / I <sup>2</sup> C0 介面的 SDA 腳位
32	A5/D19/SCL	類比輸入腳位 A5 / 數位腳位 D19 / I <sup>2</sup> C0 介面的 SCL 腳位

Pin 13 與 Pin 31 共用腳位，Pin 14 與 Pin 32 共用腳位。

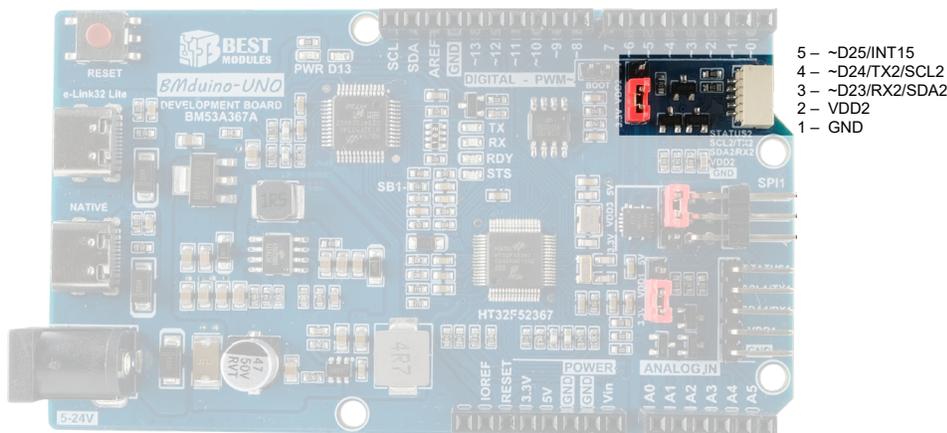
BMCOM1：可作為 I<sup>2</sup>C 或 UART 介面 (I<sup>2</sup>C1、Serial1)



腳位編號	功能	描述
1	GND	電源地
2	VDD1	3.3V 或 5V 電源輸出，由相鄰 Jumper 設定
3	D20/RX1/SDA1	數位腳位 D20 / UART1 接收腳位 / I <sup>2</sup> C1 介面的 SDA1 腳位
4	D21/TX1/SCL1	數位腳位 D21 / UART1 發送腳位 / I <sup>2</sup> C1 介面的 SCL1 腳位
5	D22/INT14	數位腳位 D22 / 外部中斷 INT14

Pin 3、Pin 4、Pin 5 的電壓準位 (3.3V 或 5V) 由相鄰 Jumper 設定。

BMCOM2：可作為 I<sup>2</sup>C 或 UART 介面 (I<sup>2</sup>C2、Serial2)

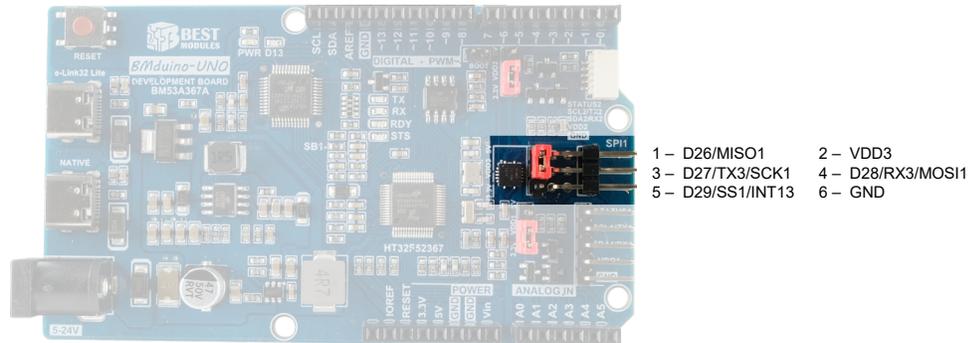


腳位編號	功能	描述
1	GND	電源地
2	VDD2	3.3V 或 5V 電源輸出，由相鄰 Jumper 設定
3	~D23/RX2/SDA2	數位腳位 D23 帶 PWM 功能 / UART2 接收腳位 / I <sup>2</sup> C2 介面的 SDA2 腳位
4	~D24/TX2/SCL2	數位腳位 D24 帶 PWM 功能 / UART2 發送腳位 / I <sup>2</sup> C2 介面的 SCL2 腳位
5	~D25/INT15	數位腳位 D25 帶 PWM 功能 / 外部中斷 INT15

Pin 3、Pin 4、Pin 5 的電壓準位 (3.3V 或 5V) 由相鄰 Jumper 設定。

注意：I<sup>2</sup>C1 與 I<sup>2</sup>C2 共用 MCU 上的同一個實體 I<sup>2</sup>C，BMCOM1 的 I<sup>2</sup>C 與 BMCOM2 的 I<sup>2</sup>C 無法同時使用。

SPI1：可作為 SPI 或 UART 介面 (SPI1、Serial3)



腳位編號	功能	描述
1	D26/MISO1	數位腳位 D26 / SPI1 介面的 MISO1 腳位
2	VDD3	3.3V 或 5V 電源輸出，由相鄰 Jumper 設定
3	D27/TX3/SCK1	數位腳位 D27 / UART3 發送腳位 / SPI1 介面的 SCK1 腳位
4	D28/RX3/MOSI1	數位腳位 D28 / UART3 接收腳位 / SPI1 介面的 MOSI1 腳位
5	D29/SS1/INT13	數位腳位 D29 / SPI1 介面的 SS1 腳位 / 外部中斷 INT13
6	GND	電源地

Pin 1、Pin 3、Pin 4、Pin 5 的電壓準位 (3.3V 或 5V) 由相鄰 Jumper 設定。

## 技術規格

### 極限參數

- 通過 e-Link32 Lite USB 介面 / Native USB 介面給 MCU 供電。

注意：USB 供電時電壓不可小於 4.5V。

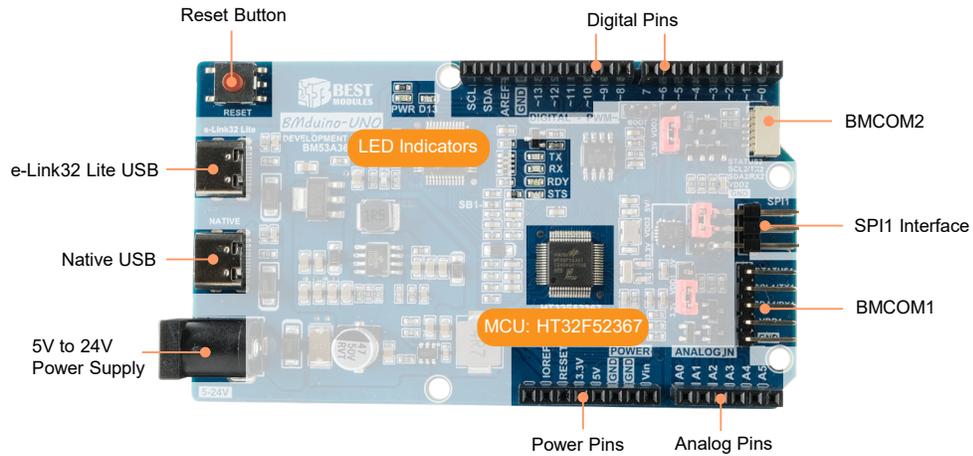
- 通過 DC Jack 輸入 5~24V，由降壓電路降壓為 5V 及 3.3V 輸出並給 MCU 供電。
- 通過 Vin 輸入 5~24V 電壓，由降壓電路降壓為 5V 及 3.3V 輸出並給 MCU 供電。

符號	參數	最小	典型	最大	單位
V <sub>inMAX</sub>	Vin 腳位輸入電壓	24	—	26	V
V <sub>USBMAX</sub>	USB 連接器輸入電壓	4.5	5.5	—	V

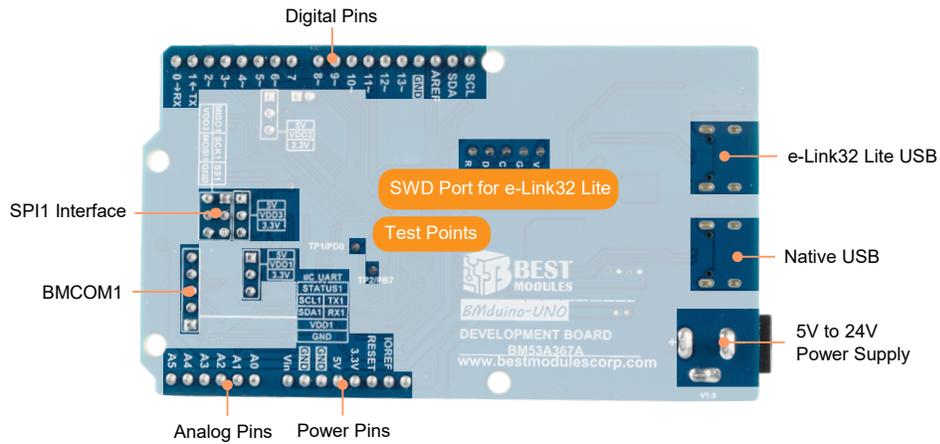
## 建議工作條件

符號	參數	測試條件	最小	典型	最大	單位
$V_{in}$	輸入電壓	—	5	—	24	V
$I_{OUT}$	5V 輸出電流	$V_{in} > 11V$	—	—	1.3	A
		$V_{in} = 4.6V$	—	—	800	mA
	3.3V 輸出電流	—	—	—	500	mA
	I/O 輸出電流	—	—	—	16	mA

## 硬體概述



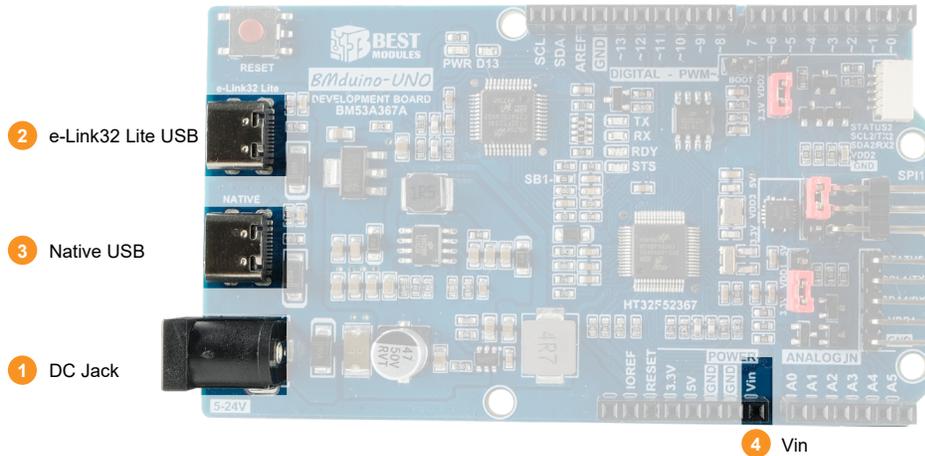
PCBA 正面圖



PCBA 反面圖

## 電源

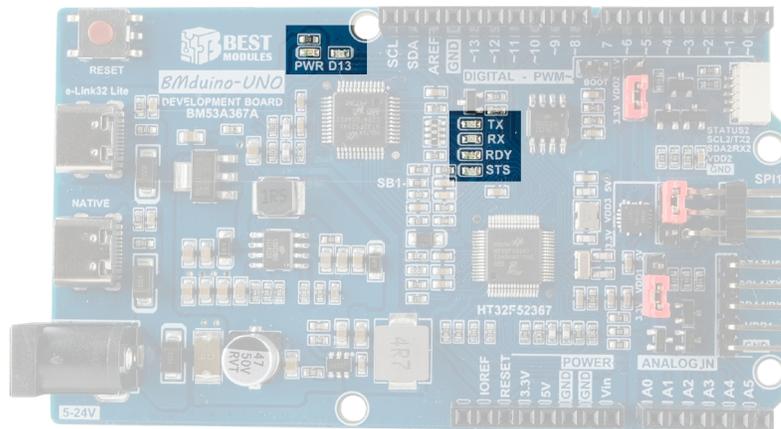
- 通過 e-Link32 Lite USB 介面輸入 5V，由降壓電路降壓為 5V 及 3.3V 輸出並給 MCU 供電。
- 通過 Native USB 介面輸入 5V 或 12V（支援 QC 2.0 接快充會升壓到 12V），由降壓電路降壓為 5V 及 3.3V 輸出並給 MCU 供電。
- 通過 DC Jack 輸入 5~24V，由降壓電路降壓為 5V 及 3.3V 輸出並給 MCU 供電。
- 通過 Vin 輸入 5~24V 電壓，由降壓電路降壓為 5V 及 3.3V 輸出並給 MCU 供電。



電源介面圖

注意：當 DC Jack / Native USB / e-Link32 Lite USB 與 Vin 同時供電時，為了防止電壓反灌問題，Vin 輸入電壓必須大於或等於 DC Jack / Native USB / e-Link32 Lite USB。

## LED 指示燈

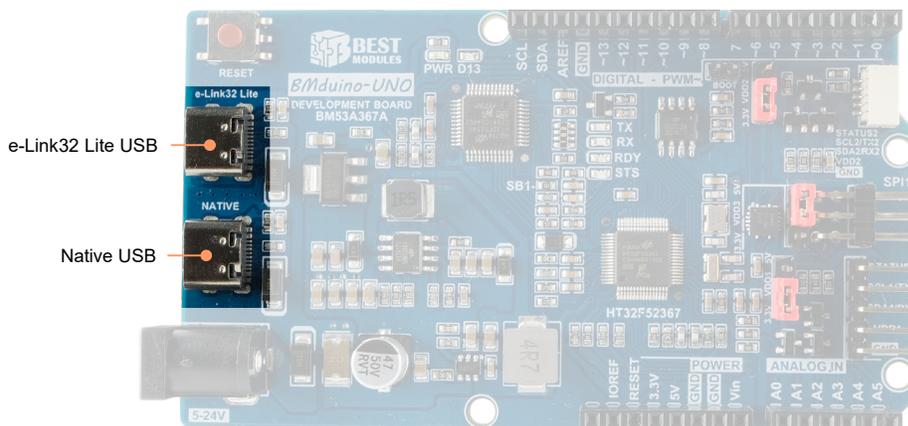


LED 配置圖

- PWR：電源 LED。
- D13：與數位腳位 13 相連，用於範例程式，觀察程式狀態用。

- TX 和 RX：指示 UART 的 TX / RX 總線傳輸狀態，數據傳輸時閃爍。
- RDY：指示 e-link32 Lite 的 USB 連線狀態，當與電腦完成連線時點亮 LED。
- STS：指示 e-link32 Lite 的燒錄狀態，當燒錄程式時會閃爍。

## USB 介面電路



BMduino-UNO BM53A36A 開發板上有 2 個 USB 介面，分別是 e-Link32 Lite 及 NATIVE。

- e-Link32 Lite：e-Link32 Lite 主要提供幾個功能：
  1. Arduino IDE 或 Keil IDE 做程式開發時，此 USB 介面連接電腦實現對主控 MCU 燒錄。
  2. 支援 VCP (Virtual COM Port) 連接到主控 MCU 的 TX 和 RX。
  3. 在 Keil IDE 的開發環境下提供程式除錯的功能 (例如，設置斷點)。

BMduino-UNO BM53A367A 預設是通過 e-Link32 Lite 電路做燒錄，燒錄時不會使用到 TX 及 RX 腳位。

- Native USB：Native USB 主要提供幾個功能：
  1. 供電介面  
當 QC 2.0 的充電器通過 Native USB 給 BMduino-UNO BM53A36A 開發板供電時，Native USB 介面預設升壓為 12V。
  2. USB 通訊  
USB 介面預設功能為 VCP，通過 Arduino API 的 SerialUSB 進行通訊。使用 BMduino-UNO BM53A36A 開發板來開發具有 USB 的產品時 (例如滑鼠或鍵盤)，此 USB 介面即是產品的 USB 介面。若要使用 Keyboard / Mouse 必須將 Head 檔加入 Sketch 中才可使用，例如 `#include <Mouse.h>`。

Obj	SerialUSB (USB CDC)	Keyboard (USB HID)	Mouse (USB HID)
腳位	僅 Native USB 介面	僅 Native USB 介面	僅 Native USB 介面

例如，BMduino-UNO 使用 Native USB 介面連接 PC 作為 USB 鍵盤或滑鼠時，只需先宣告 Keyboard.h 及 Mouse.h 並在程式中使用 Keyboard 及 Mouse 物件。

範例程式：

```
#include "Mouse.h"
#include "Keyboard.h"

// 為五個按鈕設置腳位編號：
const int upButton = 2;
const int downButton = 3;
const int leftButton = 4;
const int rightButton = 5;
const int mouseButton = 6;

int range = 5;           // X 或 Y 移動的輸出範圍；影響移動速度
int responseDelay = 20; // 滑鼠應答延時時間，單位為 ms

void setup() {
  // 按鈕輸入初始化：
  pinMode(upButton, INPUT);
  pinMode(downButton, INPUT);
  pinMode(leftButton, INPUT);
  pinMode(rightButton, INPUT);
  pinMode(mouseButton, INPUT);

  // 滑鼠控制初始化：
  Mouse.begin();
  // 鍵盤控制初始化：
  Keyboard.begin();
  // SerialUSB 初始化
  SerialUSB.begin(9600);
}

void loop() {
  // 讀取按鈕狀態：
  int upState = digitalRead(upButton);
  int downState = digitalRead(downButton);
  int rightState = digitalRead(rightButton);
  int leftState = digitalRead(leftButton);
  int clickState = digitalRead(mouseButton);

  // 根据按钮状态计算移动距离：
  int xDistance = (leftState - rightState) * range;
  int yDistance = (upState - downState) * range;

  // 如果 X 或 Y 非零，則移動：
  if ((xDistance != 0) || (yDistance != 0)) {
    // 移動滑鼠光標
    Mouse.move(xDistance, yDistance, 0);

    // 鍵盤輸出滑鼠的坐標
    Keyboard.print("Move: ");
    Keyboard.print(xDistance);
    Keyboard.print(',');
    Keyboard.println(yDistance);
    Keyboard.flush();
  }
}
```

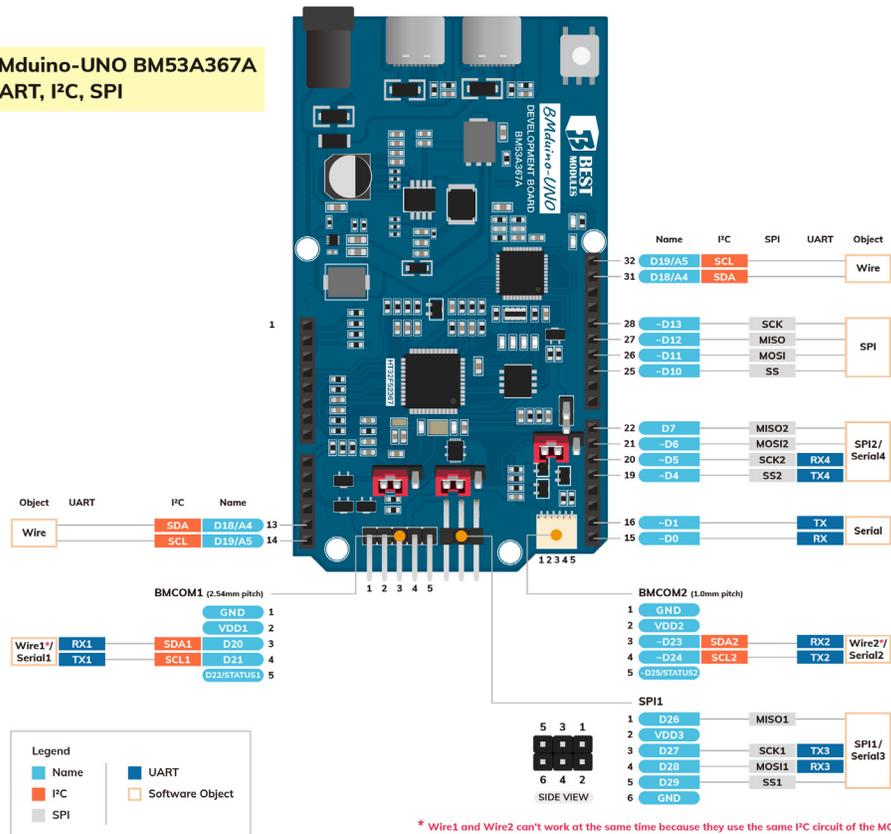
```
// SerialUSB 輸出滑鼠的坐標
SerialUSB.print("Move: ");
SerialUSB.print(xDistance);
SerialUSB.print(',');
SerialUSB.println(yDistance);
SerialUSB.flush();
}

// 如果滑鼠按鈕按下：
if (clickState == HIGH) {
  // 如果滑鼠未按下，則按下：
  if (!Mouse.isPressed(MOUSE_LEFT)) {
    Mouse.press(MOUSE_LEFT);
    // 如果鍵盤 println 未注釋，則滑鼠光標改變會導致字串被覆蓋
    // Keyboard.println("Press MOUSE_LEFT");
    // Keyboard.flush();
    SerialUSB.println("Press MOUSE_LEFT");
    SerialUSB.flush();
  }
}
// 否則滑鼠按鈕未按下：
else {
  // 如果按下滑鼠，則釋放滑鼠：
  if (Mouse.isPressed(MOUSE_LEFT)) {
    Mouse.release(MOUSE_LEFT);
    // 如果鍵盤 println 未注釋，則滑鼠光標改變會導致字串被覆蓋
    // Keyboard.println("Release MOUSE_LEFT");
    // Keyboard.flush();
    SerialUSB.println("Release MOUSE_LEFT");
    SerialUSB.flush();
  }
}

// 延時使滑鼠不會移動太快：
delay(responseDelay);
}
```

## UART, I<sup>2</sup>C, SPI

BMduino-UNO BM53A367A  
UART, I<sup>2</sup>C, SPI



\* Wire1 and Wire2 can't work at the same time because they use the same I<sup>2</sup>C circuit of the MCU.

- 支援 5 組 UART，物件名稱分別為 Serial、Serial1 ~ Serial4。

Obj	Serial	Serial1	Serial2	Serial3	Serial4
腳位	RX (D0) / TX (D1)	RX1 (D20) / TX1 (D21) (BMCOM1)	RX2 (D23) / TX2 (D24) (BMCOM2)	RX3 (D28) / TX3 (D27) (SPI1)	RX4 (D5) / TX4 (D4)

例如，要使用 BMCOM1 的 UART 介面時，只需在程式中使用 Serial1 物件。  
範例程式：

```
void setup() {
  // 串列通訊初始化：
  Serial1.begin(9600);
}

void loop() {
  // 發送類比輸入 0 的值：
  Serial1.println(analogRead(A0));
  // 等待一段時間使 A/D 轉換器在最后一次读取后穩定下來：
  delay(20);
}
```

在 BMduino-UNO BM53A367A 開發板使用 Arduino Serial Library 時，數據位可以是 7~9 位，而在 Arduino IDE 預設則是 5~8 位。

- 支援 3 組 I<sup>2</sup>C，物件名稱 Wire 其中 Wire1 與 Wire2 共用實體 I<sup>2</sup>C，同時只能有一組通訊。

Obj	Wire	Wire1	Wire2
腳位	SDA / SCL 或 A4 / A5 (共用 I/O)	SDA1 (D20) / SCL1 (D21) (BMC0M1)	SDA2 (D23) / SCL2 (D24) (BMC0M2)

注意：I<sup>2</sup>C1 與 I<sup>2</sup>C2 共用 MCU 上的同一個實體 I<sup>2</sup>C，鏈接庫 Wire1 與 Wire2 無法同時工作，只能選擇使用其一 (Wire1 或 Wire2)。如果程式中同時存在 Wire1 及 Wire2，請參考下方做法：

1. 通過 Wire1.begin() 或 Wire2.begin() 進行初始化。
2. 需要切換當前 Wire1 或 Wire2，請通過 Wire1.end() 或 Wire2.end() 關閉當前的 Wire1 或 Wire2。
3. 再通過 Wire1.begin() 或 Wire2.begin() 初始化另外一個 Wire1 或 Wire2。

若 Wire1 與 Wire2 都進行初始化，以先初始化的 Wire1 或 Wire2 為有效。

例如，要使用 BMC0M1 的 I<sup>2</sup>C 介面時，只需先宣告 Wire.h 並在程式中使用 Wire1 物件。

範例程式：

```
#include <Wire.h>

void setup()
{
  Wire1.begin();          // 加入 I2C 總線 (主機位址可選)
  Serial.begin(9600);    // 啟動串列輸出
}

void loop()
{
  Wire1.requestFrom(2, 6); // 請求從機設備 2 發送 6 字節

  while(Wire1.available()) // 從機發送的字節可能小於請求數量
  {
    char c = Wire1.read(); // 接收 1 字節的字元
    Serial.print(c);      // 打印該字元
  }

  delay(500);
}
```

- 支援 3 組 SPI。

Obj	SPI	SPI1	SPI2
腳位	SS (D10) / MOSI (D11) / MISO (D12) / SCK (D13)	SS1 (D29) / MOSI1 (D28) / MISO1 (D27) / SCK1 (D26) (SPI1)	SS2 (D4) / MOSI2 (D6) / MISO2 (D7) / SCK2 (D5)

例如，要使用 6-pin 排針的 SPI1 介面時，只需先宣告 SPI.h 並在程式中使用 SPI1 物件。

範例程式：

```
// 包含 SPI library:
#include <SPI.h>

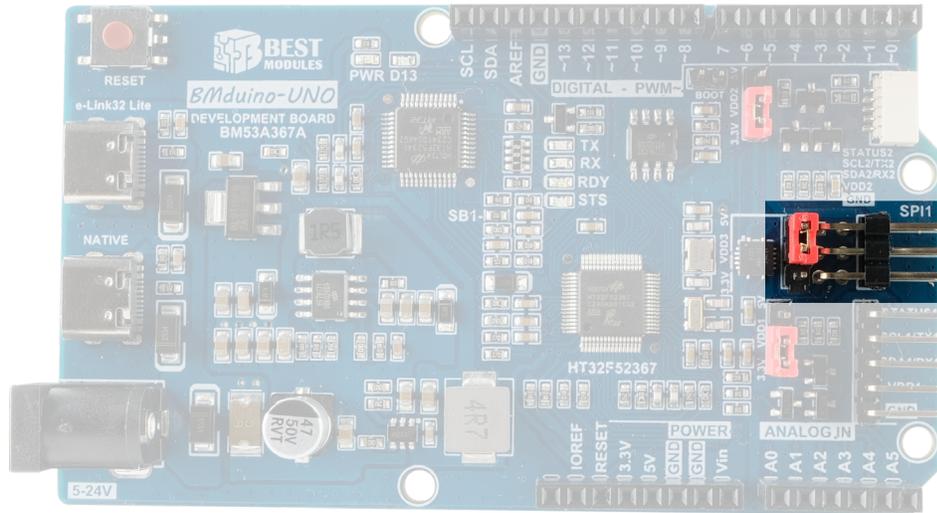
// 設置 pin 10 為數位序列埠的從機選擇腳位 :
const int slaveSelectPin = 10;

void setup() {
  // 設置 slaveSelectPin 為輸出 :
  pinMode(slaveSelectPin, OUTPUT);
  // SPI1 初始化 :
  SPI1.begin();
}

void loop() {
  // 通過數位序列埠的 6 個通道 :
  for (int channel = 0; channel < 6; channel++) {
    // 將通道上的電阻從最小值改到最大值 :
    for (int level = 0; level < 255; level++) {
      digitalPotWrite(channel, level);
      delay(10);
    }
    // 在最大值時等待 1 秒 :
    delay(100);
    // 將通道上的電阻從最大值改到最小值 :
    for (int level = 0; level < 255; level++) {
      digitalPotWrite(channel, 255 - level);
      delay(10);
    }
  }
}

void digitalPotWrite(int address, int value) {
  // SS 腳位為低時選擇芯片 :
  digitalWrite(slaveSelectPin, LOW);
  // 通過 SPI1 發送位址和值 :
  SPI1.transfer(address);
  SPI1.transfer(value);
  // SS 腳位為高時取消選擇芯片 :
  digitalWrite(slaveSelectPin, HIGH);
}
```

開發板上的 6-pin 排針 (3-pin × 2 排 · 90 度) :

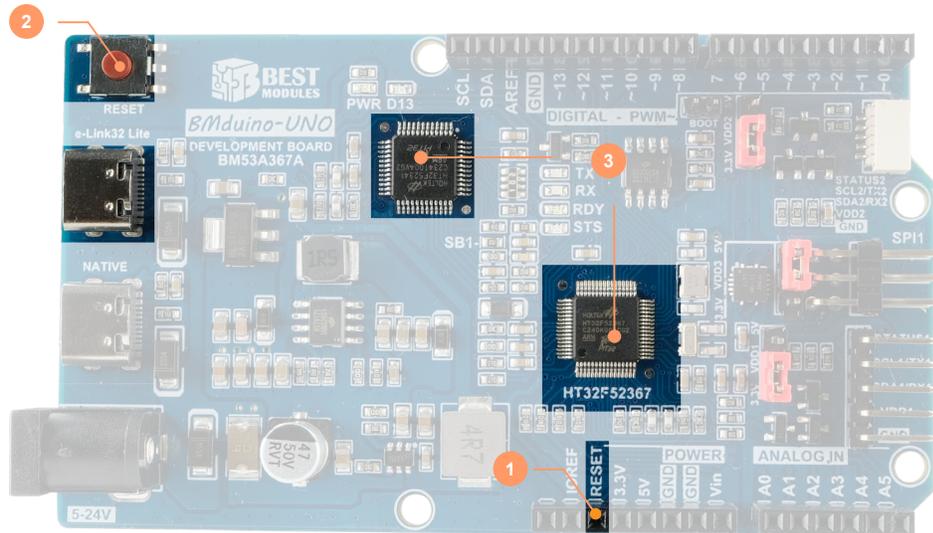


在 Arduino UNO R3 開發板，此 6-pin 排針稱為 ICSP，腳位與開發板的 Pin 11、Pin 12、Pin 13 共用。

在 BMduino-UNO BM53A367A 開發板，此 6-pin 排針為 SPI1，並沒有與開發板的 Pin 11、Pin 12、Pin 13 共用腳位，是一個可單獨使用的 SPI 介面。

Holtek 32-bit MCU 為 Cortex®-M0+ 架構，使用 SWD 介面燒錄，不是使用 SPI 介面燒錄。SWD 說明鏈接：<https://developer.arm.com/documentation/100956/0529/Arm-DSTREAM-Target-Interface-Connections/Signal-descriptions/Serial-Wire-Debug?lang=en>

## 復位電路



- 可通過將 Pin 3 (RESET) 設置為低準位 1ms 觸發系統復位。
- 可通過 RESET 按鈕進行 MCU 復位。

- RESET 電路連接 e-Link32 Lite (VCP) 的 DTR，可通過開啟 COM 序列埠進行 MCU 復位。

## BM53A367A vs Arduino UNO R3

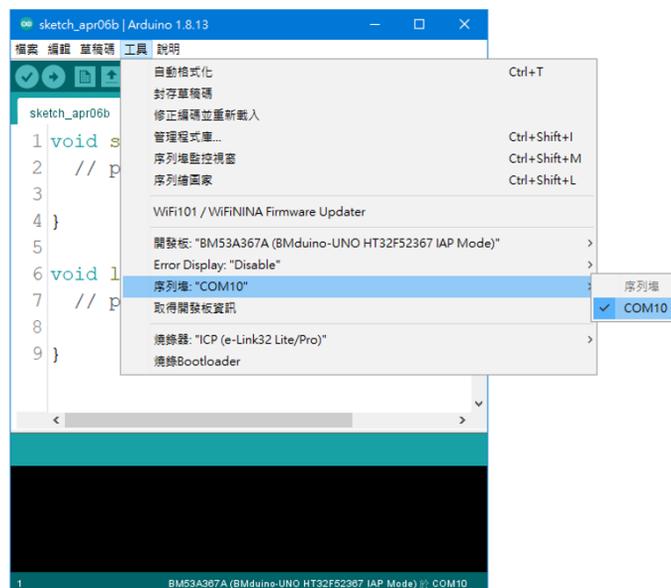
Board	BMduino-UNO BM53A367A	Arduino UNO R3
內核	Cortex®-M0+ · 60MHz	AVR 8-bit · 16MHz
Flash / EEPROM / SRAM	256KB / 4KB / 32KB	32KB / 1KB / 2KB
工作電壓	3.3V	5V
燒錄模式	ICP/IAP (Bootloader)	ICSP/ IAP (Bootloader)
燒錄介面	SWD (目標板菜單) / UART (序列埠菜單)	UART (序列埠菜單)
開發環境	Arduino、Keil	Arduino
I/O 驅動電流	16mA	20mA

## 燒錄方法

BM53A367A 在各 IDE 燒錄方法整理如下。

### 1. Arduino：

- ICP 模式 (預設模式)：點選“草稿碼 (Sketch)”下方選單的“Upload” ，Sketch 將進行程式編譯與燒錄。通過下方狀態視窗可觀察燒錄結果。詳細請參考 Arduino 官方網站：<https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics>。
- IAP 模式：與 ICP 模式方法相同，但進行“Upload”前需通過“工具 → 序列埠”選擇板子對應的 COM 序列埠 (與 Arduino UNO R3 做法相同)，如下圖。



COM 序列埠菜單

- Keil：點選 IDE 上方表單的“Build”  進行程式編譯，完成後點選表單的“Download”  進行程式燒錄。通過下方“Build Output”視窗可觀察編譯 / 燒錄結果。詳細請參考 Keil 官方網站：<https://www2.keil.com/mdk5/learn>。

## Arduino IDE 軟體

### Arduino IDE 下載及安裝

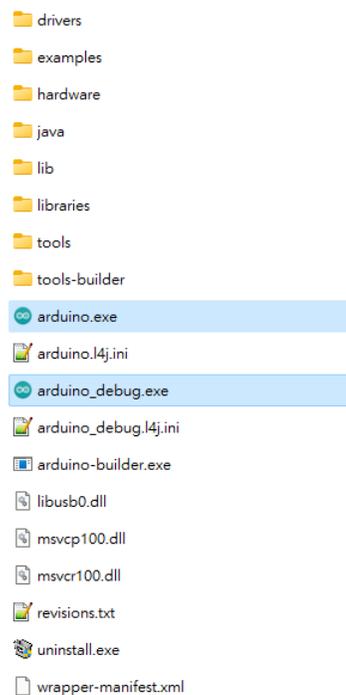
打開 Arduino 官方網站 (<http://www.arduino.cc/en/Main/Software>) 下載對應電腦作業系統的 Arduino IDE 軟體並且執行安裝。

#### Downloads



#### Arduino IDE 軟體下載

安裝完成後可以在資料夾中找到 Arduino.exe & Arduino\_debug.exe 兩個執行檔，兩者皆可以開啟程式開發環境，不一樣的是 Arduino\_debug.exe 可以開啟除錯視窗，當程式執行或編譯時都會將訊息顯示在除錯視窗上，方便用戶查看哪個環節發生問題。

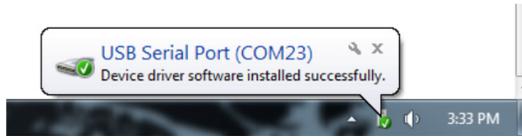


#### Arduino IDE 資料夾

## 驅動及其他下載

### USB 驅動下載

1. 在電腦 Win10 系統下，開發板與電腦相連時，電腦會自動掛載 VCP 驅動。當驅動自動下載成功，可看到提示訊息。



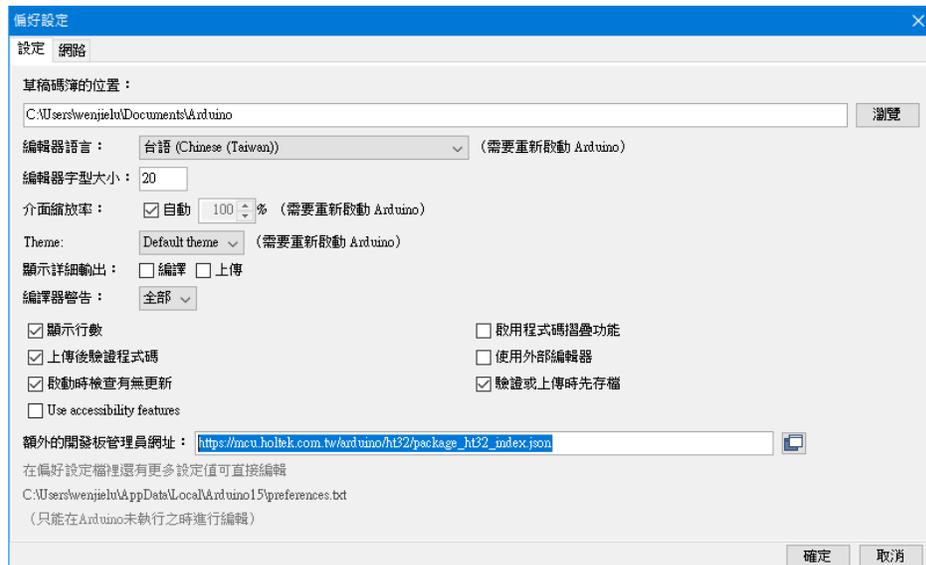
掛載 VCP

2. 在 XP 和 Win7 系統下，需手動下載 VCP 驅動。詳細做法請參考“Keil IDE 軟體”章節。

## IDE 設置

### 安裝 Holtek HT32 Boards

1. 點選“檔案→偏好設定”，選擇“設定”頁籤，在“額外的開發板管理員網址”內輸入“[https://mcu.holtek.com.tw/arduino/ht32/package\\_ht32\\_index.json](https://mcu.holtek.com.tw/arduino/ht32/package_ht32_index.json)”，完成後點選“確定”。



輸入 Json 路徑

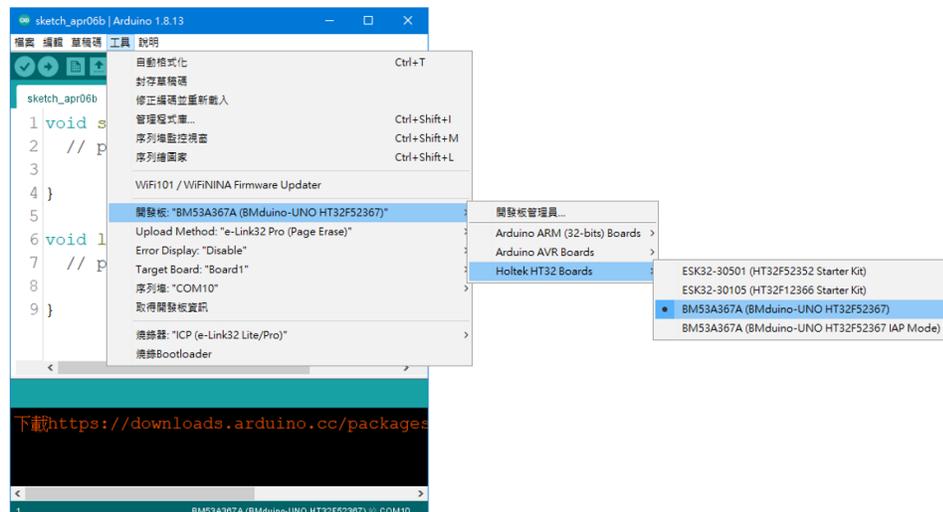
2. 點選“工具 → 開發板： “Arduino UNO” → 開發板管理員”。完成後跳出“開發板管理員”視窗，搜尋“HT32”發現 Holtek HT32 Boards 安裝視窗，請選最新版本點選“安裝”並等待安裝完成。完成後點選“關閉”。



安裝 Holtek Library

## ICP 模式設置 (建議)

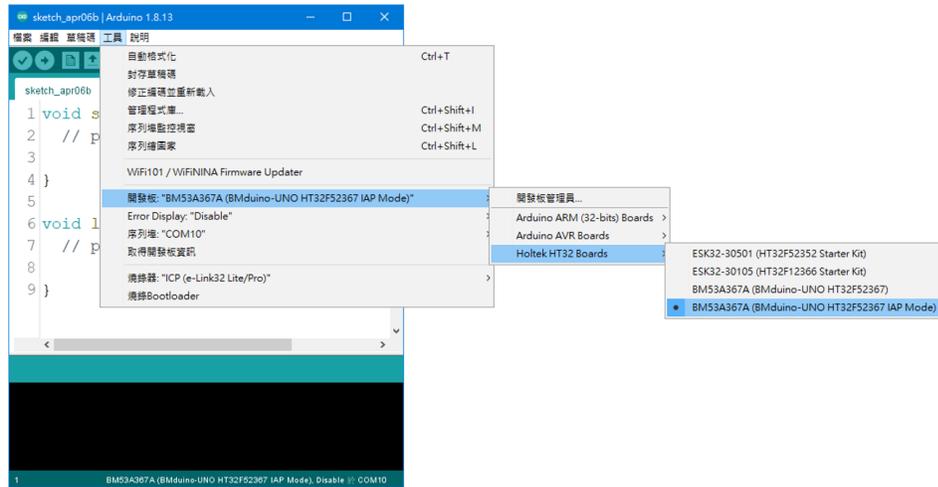
- 點選“工具 → 開發板 → Holtek HT32 Boards → BM53A367A (BMduino-UNO HT32F52367)”完成初始化設定。



選擇 BM53A367A

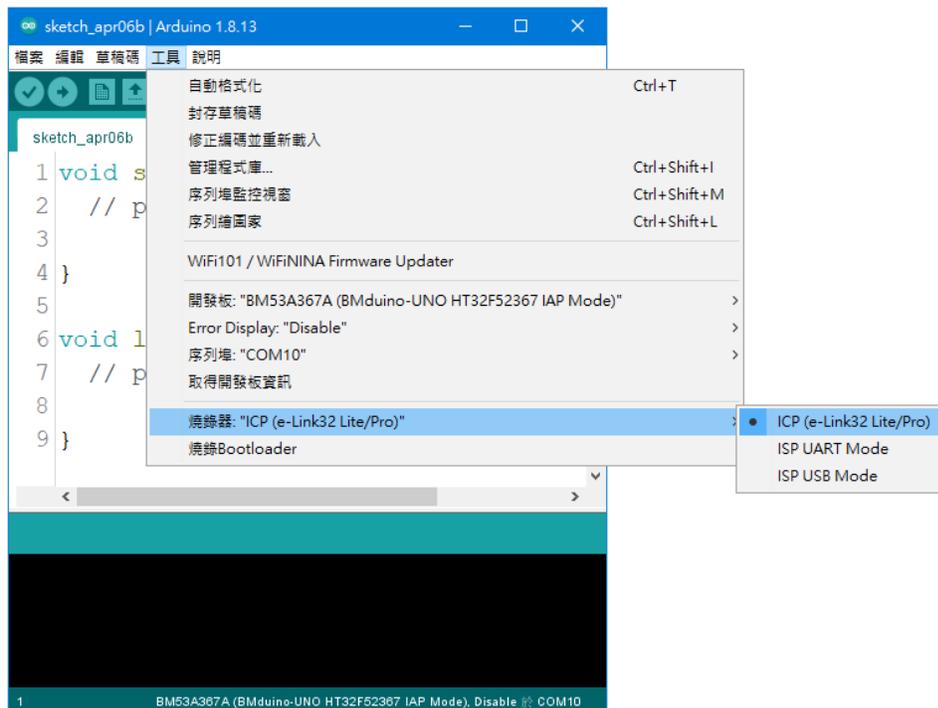
## IAP 模式設置

步驟 1. 點選 “工具 → 開發板 → Holtek HT32 Boards → BM53A367A (BMduino-UNO HT32F52367 IAP Mode)”



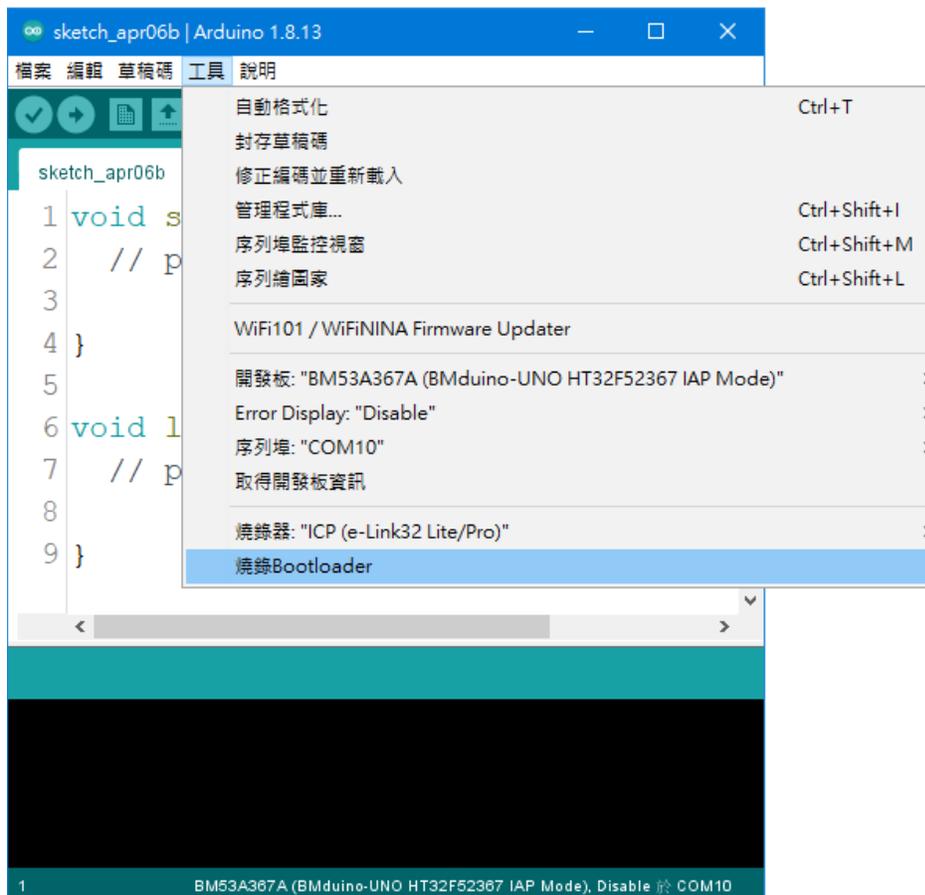
### 選擇 BM53A367A IAP 模式

步驟 2. BM53A367A 出廠未燒錄 Bootloader，因此需先進行 Bootloader 的燒錄。首先決定燒錄 Bootloader 的方法，點選 “工具 → 燒錄器 → ICP (e-Link32 Lite/Pro)”。



### 選擇 ICP (e-Link32 Lite/Pro)

步驟 3. 進行 Bootloader 燒錄，點選“工具 → 燒錄 Bootloader”。



### 選擇燒錄 Bootloader

步驟 4. 完成 Bootloader 燒錄後可發現 D13 LED 開始連線閃爍，代表完成 Bootloader 的燒錄，完成 IAP 模式的初始化。

## Arduino Library 參考

由於 BMduino-UNO BM53A367A 與 Arduino UNO R3 兩者的硬體資源不同，因此在使用 Arduino IDE 內建的 Library 時，有一部分 Library 的參數或執行方式會有不同，表列如下：

#	Library	BMduino-UNO BM53A367A	Arduino UNO R3
1	Serial	數據位：7~9 位 接收緩存器：255 字節	數據位：5~8 位 接收緩存器：64 字節
2	analogReference()	內部參考電壓：1.215V、2V、2.5V 或 2.7V	內部參考電壓：1.1V
3	SPI	setClockDivider(4) → SPI SCK=15MHz	setClockDivider(4) → SPI SCK=4MHz
4	tone()	最低頻率 1Hz	最低頻率 31Hz
5	analogWrite()	PWM 頻率：1000Hz 腳位：D0~D13	PWM 頻率：490/980Hz 腳位：D3、D5、D6、 D9、D10、D11
6	SoftwareSerial	TX：230400bps、RX：115200bps 應答延時時間：>120μs	TX/RX：57600bps 應答延時時間：>15μs
7	Servo	除能 analogWrite() on D23 (BMCOM2)	除能 analogWrite() on D9, D10
8	MsTimer2	內建 Library，為了防止名稱重複 改名為 MsTimer	第三方 Library
9	attachInterrupt()	D2~D12、D22、D25、D29	D2、D3

1. [Serial.begin\(\)](#)

Syntax: Serial.begin(speed, config)

其中 config 的有效值為：

BMduino-UNO BM53A367A	Arduino UNO R3
SERIAL_7N1	SERIAL_5N1
SERIAL_8N1 ( 默認 )	SERIAL_6N1
SERIAL_9N1	SERIAL_7N1
SERIAL_7N2	SERIAL_8N1 ( 默認 )
SERIAL_8N2	SERIAL_5N2
SERIAL_9N2	SERIAL_6N2
SERIAL_7E1：偶校驗	SERIAL_7N2
SERIAL_8E1	SERIAL_8N2
SERIAL_9E1	SERIAL_5E1：偶校驗
SERIAL_7E2	SERIAL_6E1
SERIAL_8E2	SERIAL_7E1
SERIAL_9E2	SERIAL_8E1
SERIAL_7O1：奇校驗	SERIAL_5E2
SERIAL_8O1	SERIAL_6E2
SERIAL_9O1	SERIAL_7E2
SERIAL_7O2	SERIAL_8E2
SERIAL_8O2	SERIAL_5O1：奇校驗
SERIAL_9O2	SERIAL_6O1
	SERIAL_7O1
	SERIAL_8O1
	SERIAL_5O2
	SERIAL_6O2
	SERIAL_7O2
	SERIAL_8O2

## 2. [analogReference\(\)](#)

Syntax: analogReference(type)

type 的有效值為：

BMduino-UNO BM53A367A	Arduino UNO R3
DEFAULT：默認 3.3V 類比參考電壓 INTERNAL1V215：內部 1.215V 參考電壓 INTERNAL2V0：內部 2V 參考電壓 INTERNAL2V5：內部 2.5V 參考電壓 INTERNAL2V7：內部 2.7V 參考電壓 EXTERNAL：施加在 AREF 腳位的電壓 ( 僅 0~3.3V ) 被用作參考電壓	DEFAULT：默認 5V 類比參考電壓 INTERNAL：內部 1.1V 參考電壓 EXTERNAL：施加在 AREF 腳位的電壓 ( 僅 0~5V ) 被用作參考電壓

## 3. [SPI.setClockDivider\(\)](#)

Syntax: SPI.setClockDivider(divider)

setClockDivider 為除頻的 API。因為工作頻率不同，導致參數 divider 相同但 SCK 輸出的頻率不同，BM53A367A 工作頻率 60MHz；Arduino UNO R3 工作頻率 16MHz。舉例如下：

BMduino-UNO BM53A367A	Arduino UNO R3
SPI.setClockDivider(4) → SCK = 60MHz / 4 = 15MHz	SPI.setClockDivider(4) → SCK = 16MHz / 4 = 4MHz

## 4. [tone\(\)](#)

Syntax: tone(pin, frequency)

tone(pin, frequency, duration)

BM53A367A 最低輸出頻率為 1Hz；Arduino UNO R3 最低輸出頻率 31Hz。

## 5. [analogWrite\(\)](#)

Syntax：analogWrite(pin, value)

BM53A367A 的 PWM 週期 1000Hz；Arduino UNO R3 的 PWM 週期 490Hz/980Hz。

## 6. [SoftwareSerial](#)

BM53A367A：TX 最高支援 230400bps，RX 最高 115200bps

Arduino UNO R3：TX/RX 最高支援 57600bps

## 7. [Servo](#)

BM53A367A：D23 的 analogWrite() 輸出 PWM 的功能被禁止，D23 屬 BCOM2。

Arduino UNO R3：D9、D10 的 analogWrite() 輸出 PWM 的功能被禁止。

## 8. [MsTimer2](#)

BM53A367A：內建此 Library 名稱改為 MsTimer，可在檔案 → 範例 → MsTimer 內找到相關範例。

Arduino UNO R3：第三方提供 Library 需通過 Library Manager 下載。

## 9. [attachInterrupt\(\)](#)

Syntax：attachInterrupt(interruptNum, FuncPtr callback, mode)

BM53A367A：D2~D12、D22、D25、D29 可作為外部中斷使用。

Arduino UNO R3：D2、D3 作為外部中斷使用。

## 範例

### 硬體準備

需準備開發板、Type-C USB 傳輸線、電腦。將開發板的 e-link32 Lite 通過 USB 線與電腦連接，此時 PWR LED 被點亮，還需等 e-link32 Lite 被列舉完成，列舉完成 RDY LED 會被點亮此時硬體準備完成。

### 範例代碼

請執行 Blink 範例，詳細請參考下方鏈接。完成範例燒錄後可發現 D13 LED 每秒翻轉一次。

<https://docs.arduino.cc/built-in-examples/basics/Blink>

## Keil IDE 軟體

### Keil IDE 下載及安裝

打開 Keil 官方網站 (<https://www.keil.com/demo/eval/arm.htm>) 下載 MDK-ARM 並且執行安裝。詳細安裝步驟請參考下方鏈接：

[https://www.holtek.com.tw/documents/10179/6393504/HT32\\_Keil-QuickStartv110.pdf](https://www.holtek.com.tw/documents/10179/6393504/HT32_Keil-QuickStartv110.pdf)

### IDE 設置

1. 下載 HT32 開發資源：通過下方鏈接下載最新 HT32F5 Series (Cortex<sup>®</sup>-M0+)。內包含了 HT32 開發所需的全部資源，下載完成後請解壓縮。  
<https://mcu.holtek.com.tw/ht32/resource/>
2. 安裝 HT32 Packs，執行 “\HT32\_M0p\_vxxxxxxx\Tools\Holtek.HT32\_DFP.xx.xx.xx.pack”。
3. 安裝 VCP 驅動，執行 “\HT32\_M0p\_vxxxxxxx\Tools\HT32\_VCP\_Driver\_vxxx.exe”。
4. 將 HT32 FW Lib 解壓縮，路徑 “\Firmware\_Library\HT32\_STD\_5xxxx\_FWLib\_Vx.x.x\_xxxx.zip”。

### 運行範例

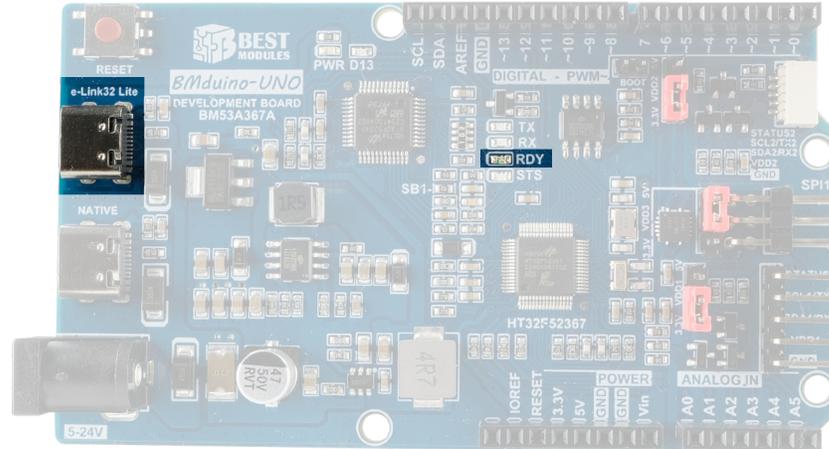
1. 開啟 HT32 FW Lib 內的 Keil 專案，路徑 “\Firmware\_Library\HT32\_STD\_5xxxx\_FWLib\_Vx.x.x\_xxxx.zip\project\_template\IP\Example\MDK\_ARMv5\Project\_53a367a.uvprojx”。
2. 通過 Keil IDE 進行編譯與燒錄，詳細請參考“燒錄方法”章節。
3. 按壓 RESET 按鈕，觀察 D13 LED 快速閃爍 5 次，完成測試。  
補充：此範例也展示 Serial (115200, 8, N, 1) 的功能，請通過終端機軟體 (例如 Tera Term，詳細內容請參考官方網站 <https://tssh2.osdn.jp/index.html.en>)，開啟 COM 序列埠觀察提示訊息。

## 問題排查

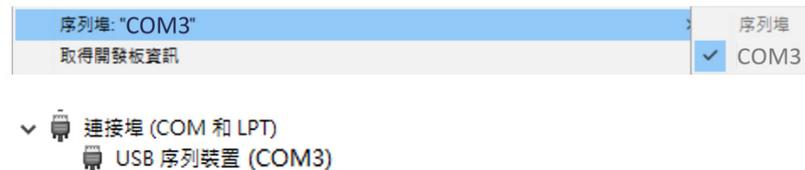
### 串口未出現在序列埠菜單中

代表 e-Link32 Lite 的 VCP 連線失敗，請依照下方步驟除錯：

1. 請確認 e-Link Lite 的 USB 是否連接 PC，以及 RDY LED 是否點亮。



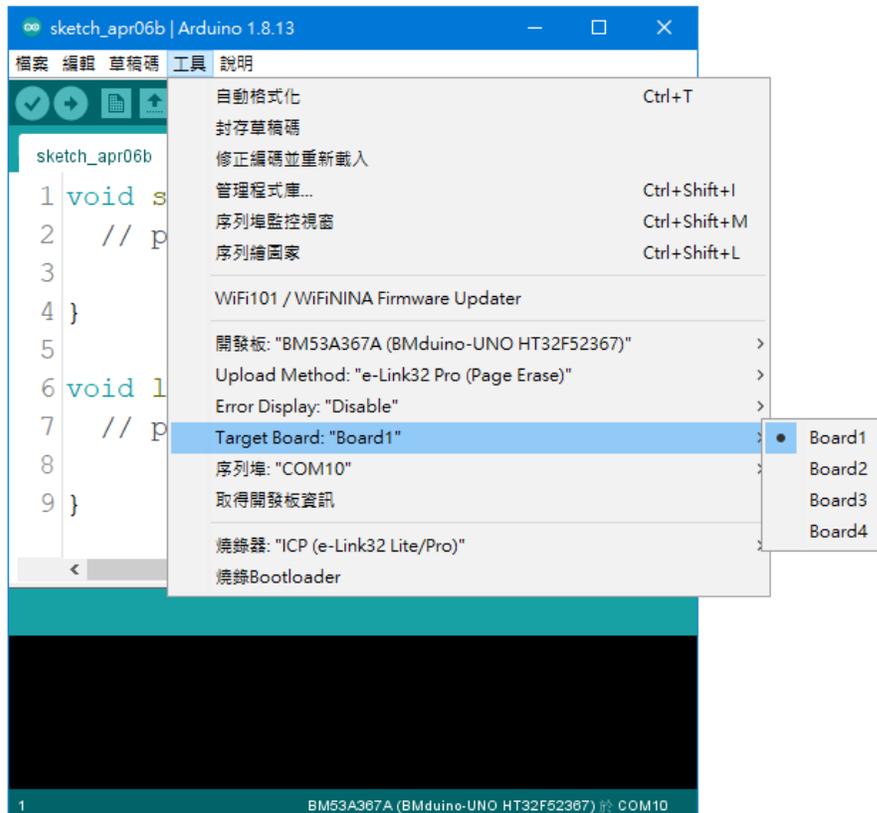
2. 若 RDY LED 未點亮請重新插拔 e-Link Lite 的 USB 並選擇 PC 上其他 USB 介面連接。
3. 10 秒後若 RDY LED 未被點亮請重新啟動 PC。若是 RDY 點亮就可在 Arduino COM 序列埠菜單內發現 BMduino 的 COM 序列埠。



4. RDY LED 已點亮但 Arduino COM 序列埠菜單未顯示 BMduino 的 COM 序列埠，請安裝 VCP 驅動程式（僅限 Windows 電腦）。  
參考 Keil IDE 軟體裡 IDE 設置的第三點安裝 VCP 驅動。

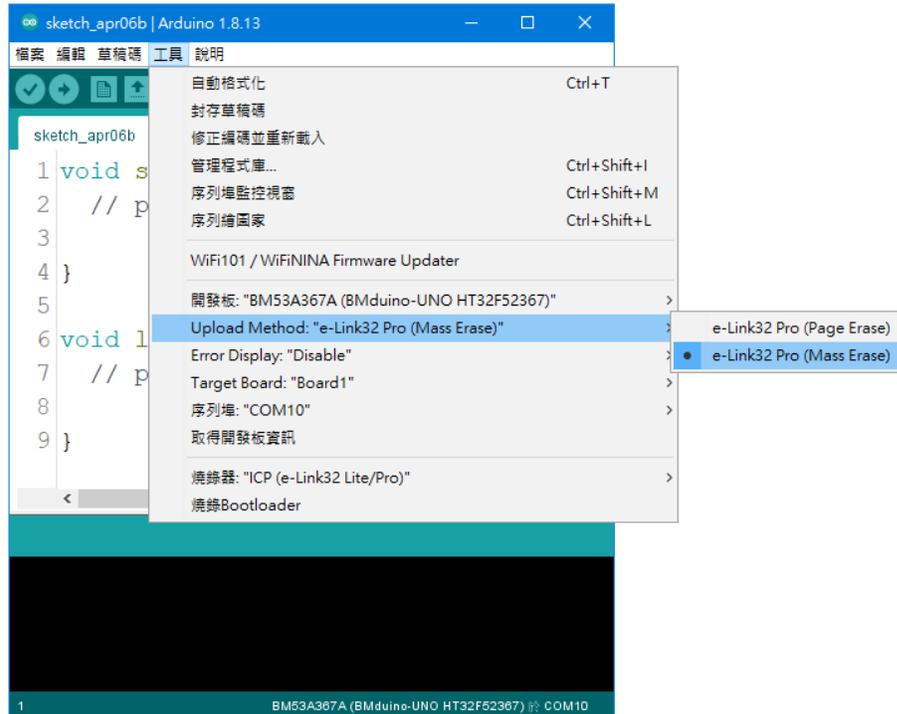
## 範例上傳失敗或凍結

1. 錯誤訊息 **“This computer can't enumerate any e-Link32 Pro/Lite. Please make sure this computer has indeed connected to e-Link32 Pro/Lite.”** 代表 e-Link32 Lite 的 CMSIS-DAP (燒錄介面) 連線失敗，請確認 e-Link Lite 的 USB 是否連接 PC 以及 RDY LED 是否點亮，若 RDY LED 未點亮請重新插拔 e-Link Lite 的 USB 並選擇 PC 上其他 USB 介面連接，10 秒後若 RDY LED 未被點亮請重新啟動 PC。
2. 錯誤訊息 **“The corresponding e-Link32 Pro/Lite can't be found by the target ID/SN, which can be in INI file or specified by users.”** 代表未找到你指定的目標板，請點選 **“工具 → Target Board: → Board1”**，再進行 **“Upload”**。



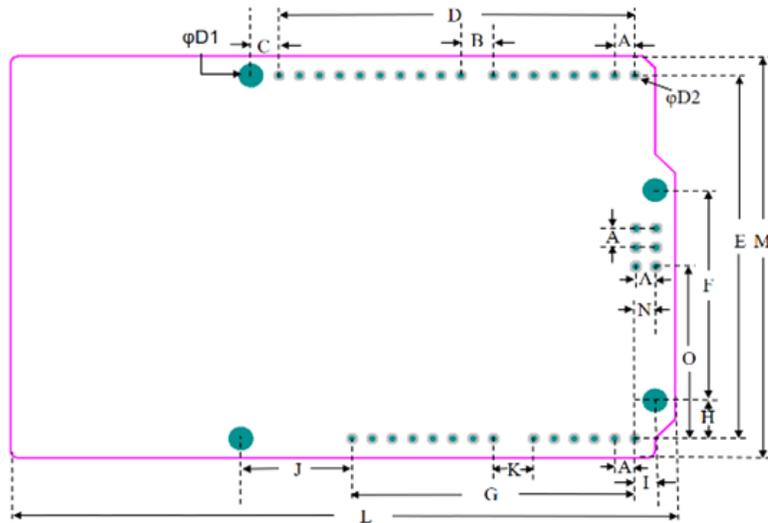
選擇 Board1

3. 若上傳失敗訊息並非上述的狀況，請通過“Mass Erase”將MCU 清空後再燒錄程式。具體做法請點選“工具 → Upload Method: → e-Link32 Pro (Mass Erase)”，再進行“Upload”。



Mass Erase

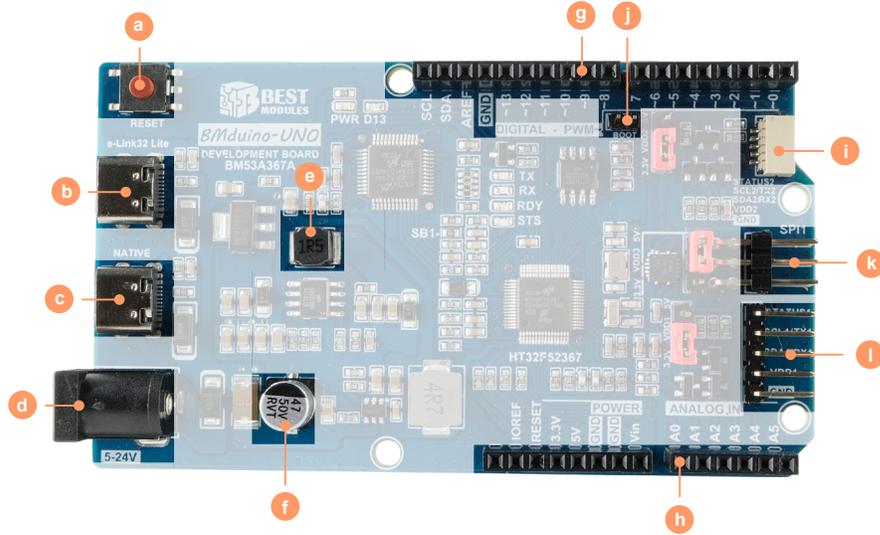
## 尺寸規格



尺寸資訊

編號	單位	mm	inch
A		2.54	0.1
B		4.064	0.16
C		3.556	0.14
D		44.704	1.76
E		48.26	1.9
F		27.94	1.1
G		36.56	1.4
H		5.08	0.2
I		2.54	0.1
J		13.97	0.55
K		5.08	0.2
L		93.221	3.67
M		53.35	2.1
N		2.667	0.105
O		22.86	0.9
D1		3.2512	0.128

尺寸列表



元件尺寸 – 高度資訊

尺寸 編號	長		寬		高	
	mm	inch	mm	inch	mm	inch
a	6.2	0.236	6.2	0.236	2	0.078
b	8	0.314	5.15	0.203	3	0.118
c	8	0.314	5.15	0.203	3	0.118
d	14.2	0.559	9	0.354	11	0.433
e	5	0.197	5	0.197	4	0.157
f	6.3	0.248	6.3	0.248	7.7	0.303
g	48	1.89	2.5	0.098	8	0.315
h	38	1.496	2.5	0.098	8	0.315
i	4.53	0.178	7.5	0.295	3.1	0.122
j	4	0.155	2	0.80	4	0.155
k	13.7	0.540	7.4	0.292	4	0.157
l	8.5	0.334	12.776	0.503	4	0.157

默認高度：11mm / 0.433 inch

元件尺寸 – 高度列表

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、示例、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。