



---

陀螺儀 & 加速度計模組

# **BMS56M605**

## **Arduino Library V1.0.1 說明**

版本：V1.01 日期：2023-12-13

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

## 目錄

簡介 .....	3
Arduino Lib 函式 .....	3
Arduino Lib 下載及安裝 .....	10
Arduino 範例 .....	11
範例 1：readAccelerationAndGyroscopeAndTemperature .....	11
範例 2：getMotionStatus .....	13

## 簡介

BMS56M605 是倍創推出的陀螺儀 & 加速度計模組，使用 I<sup>2</sup>C 通訊方式。本文檔對 BMS56M605 的 Arduino Lib 函式、Arduino Lib 安裝方式進行說明；範例演示了讀取感測器 6 軸資料、運動檢測等功能。

## Arduino Lib 函式

Arduino Lib 名称：BMS56M605		Lib 版本：V1.0.1
<b>構造函式 &amp; 初始化</b>		
1	BMS56M605(uint8_t intPin=8, TwoWire *theWire=&Wire)	
	描述	構造函式
	參數	intPin：INT 腳位 *theWire：I <sup>2</sup> C 通訊介面選擇
	返回值	—
	備註	—
2	void begin(uint8_t addr=BMS56M605_IICADDR)	
	描述	模組初始化
	參數	addr：從機位址，預設 0x68
	返回值	void
	備註	—
<b>功能函式</b>		
3	void getEvent()	
	描述	一次性獲取資料，包括溫度值和 6 軸的資料
	參數	—
	返回值	void
	備註	分別儲存於類公有變數 float temperature; float accX, accY, accZ; float gyroX, gyroY, gyroZ
4	float readTemperature()	
	描述	獲取溫度值
	參數	—
	返回值	溫度值，單位 °C
	備註	—
5	float readAccelerationX()	
	描述	獲取 X 軸加速度值
	參數	—
	返回值	X 軸加速度值，單位 g
	備註	—

6	float readAccelerationY()	
	描述	獲取 Y 軸加速度值
	參數	—
	返回值	Y 軸加速度值 · 單位 g
	備註	—
7	float readAccelerationZ()	
	描述	獲取 Z 軸加速度值
	參數	—
	返回值	Z 軸加速度值 · 單位 g
	備註	—
8	float readGyroscopeX()	
	描述	獲取 X 軸陀螺儀值
	參數	—
	返回值	X 軸陀螺儀值 · 單位 °/s
	備註	—
9	float readGyroscopeY()	
	描述	獲取 Y 軸陀螺儀值
	參數	—
	返回值	Y 軸陀螺儀值 · 單位 °/s
	備註	—
10	float readGyroscopeZ()	
	描述	獲取 Z 軸陀螺儀值
	參數	—
	返回值	Z 軸陀螺儀值 · 單位 °/s
	備註	—
11	void setInterruptPinPolarity(uint8_t active_level)	
	描述	設定中斷腳位極性
	參數	active_level : 0(ACTIVE_HIGH) : 高準位有效 1(ACTIVE_LOW) : 低準位有效
	返回值	void
	備註	—
12	void setINT(uint8_t mode, uint8_t isEnabled=false)	
	描述	設定中斷模式
	參數	mode : 5(ZERO_MOTION_MODE) : ZERO MOTION 模式 6(MOTION_MODE) : MOTION MODE 模式 7(FREE_FALL_MODE) : FREE FALL 模式 isEnabled : true : 啟用 false : 停用 ( 預設 )
	返回值	void
	備註	—

13	uint8_t getINT()	
	描述	獲取 INT 腳位準位
	參數	—
	返回值	INT 腳位準位 1 : 高準位 0 : 低準位
	備註	—
14	void writeReg(uint8_t addr, uint8_t data)	
	描述	寫暫存器
	參數	addr : 暫存器位址 data : 想寫入的資料
	返回值	void
	備註	—
15	uint8_t readReg(uint8_t addr)	
	描述	讀暫存器 · 1 位元組
	參數	addr : 暫存器位址
	返回值	暫存器值
	備註	—
16	void readReg(uint8_t addr, uint8_t rBuf[], uint8_t rLen)	
	描述	連讀暫存器 · rLen 位元組 · 存於陣列 rBuf[]
	參數	addr : 暫存器位址 rBuf[] : 用於儲存讀取的資料 rLen : 連讀的位元組數
	返回值	void
	備註	—
17	void enableSleep(bool isEnabled=false)	
	描述	啟用進入休眠模式
	參數	isEnabled : true – 啟用 false – 停用 ( 預設 )
	返回值	void
	備註	—
18	void enableCycle(bool isEnabled=false)	
	描述	啟用循環喚醒模式
	參數	isEnabled : true – 啟用 false – 停用 ( 預設 )
	返回值	void
	備註	在進入休眠模式停用、循環喚醒模式啟用時，模組將每隔一段時間喚醒一次，喚醒速率通過 setCycleRate 函式設定

19	void reset()	
	描述	重置
	參數	—
	返回值	void
	備註	恢復所有暫存器為預設值
<b>參數讀取 &amp; 設定</b>		
20	uint8_t getAccelerometerRange()	
	描述	獲取加速度計量程
	參數	—
	返回值	加速度計量程 0 : ±2g 1 : ±4g 2 : ±8g 3 : ±16g
	備註	—
21	uint8_t getGyroRange()	
	描述	獲取陀螺儀量程
	參數	—
	返回值	陀螺儀量程 0 : ±250°/s 1 : ±500°/s 2 : ±1000°/s 3 : ±2000°/s
	備註	—
22	uint8_t getFreefallThreshold()	
	描述	獲取自由落體閾值
	參數	—
	返回值	自由落體閾值，單位 mg
	備註	—
23	uint8_t getFreefallDuration()	
	描述	獲取自由落體持續時間
	參數	—
	返回值	自由落體持續時間，單位 ms
	備註	—
24	uint8_t getMotionThreshold()	
	描述	獲取運動檢測閾值
	參數	—
	返回值	運動檢測閾值，單位 mg
	備註	—
25	uint8_t getMotionDuration()	
	描述	獲取運動檢測持續時間
	參數	—
	返回值	運動檢測持續時間，單位 ms
	備註	—

26	uint8_t getZeroMotionThreshold()	
	描述	獲取靜止檢測閾值
	參數	—
	返回值	靜止檢測閾值 · 單位 mg
	備註	—
27	uint8_t getZeroMotionDuration()	
	描述	獲取靜止檢測持續時間
	參數	—
	返回值	靜止檢測持續時間 · 單位 ms
	備註	—
28	uint8_t getClock()	
	描述	獲取時鐘源
	參數	—
	返回值	時鐘源 0 : 內建 8MHz 振盪器 1 : 帶 X 軸陀螺儀參考的 PLL 2 : 帶 Y 軸陀螺儀參考的 PLL 3 : 帶 Z 軸陀螺儀參考的 PLL 4 : 帶外部 32.768kHz 參考的 PLL 5 : 帶外部 19.2MHz 參考的 PLL 6 : 預留 7 : 停止時鐘並保持定時發生器重置
	備註	—
29	uint8_t getFilterBandwidth()	
	描述	獲取帶通濾波範圍
	參數	—
	返回值	帶通濾波範圍 0 : ACC 頻寬 260Hz · GYRO 頻寬 256Hz 1 : ACC 頻寬 184Hz · GYRO 頻寬 188Hz 2 : ACC 頻寬 96Hz · GYRO 頻寬 98Hz 3 : ACC 頻寬 44Hz · GYRO 頻寬 42Hz 4 : ACC 頻寬 21Hz · GYRO 頻寬 20Hz 5 : ACC 頻寬 10Hz · GYRO 頻寬 10Hz 6 : ACC 頻寬 5Hz · GYRO 頻寬 5Hz
	備註	—
30	uint8_t getSampleRateDivisor()	
	描述	獲取採樣速率除數
	參數	—
	返回值	採樣速率除數
	備註	採樣速率 = Gyroscope Output Rate/(1+divisor) 採樣速率也可通過寫 Configuration CONFIG 暫存器進行設定

31	uint8_t getCycleRate()	
	描述	獲取循環喚醒速率值
	參數	—
	返回值	循環喚醒速率值 0 : 1.25Hz 1 : 2.5Hz 2 : 5Hz 3 : 10Hz
	備註	—
32	void setAccelerometerRange(uint8_t range)	
	描述	設定加速度計量程
	參數	range : 0(ACC_RANGE_2G) : ±2g 1(ACC_RANGE_4G) : ±4g 2(ACC_RANGE_8G) : ±8g 3(ACC_RANGE_16G) : ±16g
	返回值	void
	備註	—
33	void setGyroRange(uint8_t range)	
	描述	設定陀螺儀量程
	參數	range : 0(GYRO_RANGE_250) : ±250°/s 1(GYRO_RANGE_500) : ±500°/s 2(GYRO_RANGE_1000) : ±1000°/s 3(GYRO_RANGE_2000) : ±2000°/s
	返回值	void
	備註	—
34	void setFreefallThreshold(uint8_t threshold)	
	描述	設定自由落體閾值
	參數	threshold : 自由落體閾值 · 1 LSB=1mg
	返回值	void
	備註	—
35	void setFreefallDuration(uint8_t duration)	
	描述	設定自由落體持續時間
	參數	duration : 自由落體持續時間 · 1 LSB=1ms
	返回值	void
	備註	—
36	void setMotionThreshold(uint8_t threshold)	
	描述	設定運動檢測閾值
	參數	threshold : 運動檢測閾值 · 1 LSB=1mg
	返回值	void
	備註	—

37	void setMotionDuration(uint8_t duration)	
	描述	設定運動檢測持續時間
	參數	duration：運動檢測持續時間，1 LSB=1ms
	返回值	void
	備註	—
38	void setZeroMotionThreshold(uint8_t threshold)	
	描述	設定靜止檢測閾值
	參數	threshold：靜止檢測閾值，1 LSB=1mg
	返回值	void
	備註	—
39	void setZeroMotionDuration(uint8_t duration)	
	描述	設定靜止檢測持續時間
	參數	duration：靜止檢測持續時間，1 LSB=1ms
	返回值	void
	備註	—
40	void setClock(uint8_t clock)	
	描述	設定時鐘源
	參數	clock： 0(INTERNAL_8MHZ)：內建 8MHz 振盪器 1(PLL_X_GYRO)：帶 X 軸陀螺儀參考的 PLL 2(PLL_Y_GYRO)：帶 Y 軸陀螺儀參考的 PLL 3(PLL_Z_GYRO)：帶 Z 軸陀螺儀參考的 PLL 4(PLL_EXTERNAL_32_768kHz)：帶外部 32.768kHz 參考的 PLL 5(PLL_EXTERNAL_19_2MHZ)：帶外部 19.2MHz 參考的 PLL 6：預留 7(STOP_CLOCK)：停止時鐘並保持定時發生器重置
	返回值	void
	備註	—
41	void setFilterBandwidth(uint8_t band)	
	描述	設定帶通濾波範圍
	參數	band： 0(ACC_260HZ_GYRO_256HZ)：ACC 頻寬 260Hz，GYRO 頻寬 256Hz 1(ACC_184HZ_GYRO_188HZ)：ACC 頻寬 184Hz，GYRO 頻寬 188Hz 2(ACC_96HZ_GYRO_98HZ)：ACC 頻寬 96Hz，GYRO 頻寬 98Hz 3(ACC_44HZ_GYRO_42HZ)：ACC 頻寬 44Hz，GYRO 頻寬 42Hz 4(ACC_21HZ_GYRO_20HZ)：ACC 頻寬 21Hz，GYRO 頻寬 20Hz 5(ACC_10HZ_GYRO_10HZ)：ACC 頻寬 10Hz，GYRO 頻寬 10Hz 6(ACC_5HZ_GYRO_5HZ)：ACC 頻寬 5Hz，GYRO 頻寬 5Hz
	返回值	void
	備註	—
42	void setSampleRateDivisor(uint8_t divisor)	
	描述	設定採樣速率除數
	參數	divisor：採樣速率除數
	返回值	void
	備註	採樣速率 = Gyroscope Output Rate/(1+divisor)

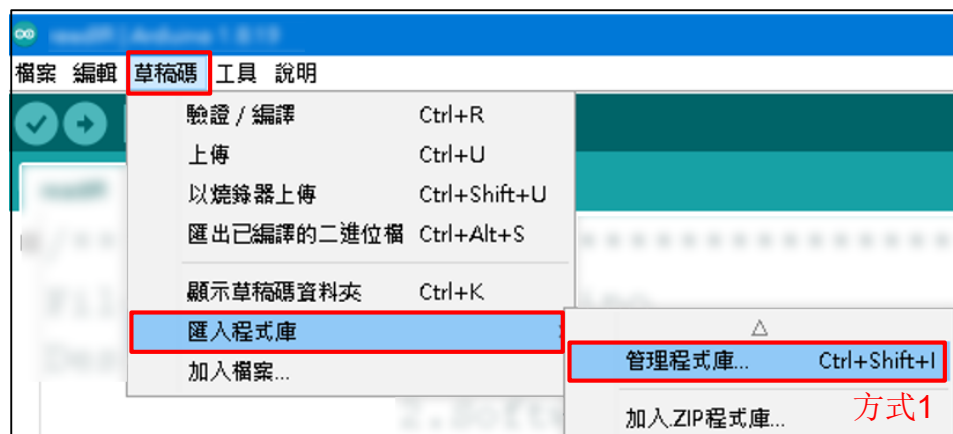
	void setCycleRate(uint8_t rate)	
	描述	設定循環喚醒速率
43	參數	rate : 0(F_1_25HZ) : 1.25Hz 1(F_2_5HZ) : 2.5Hz 2(F_5HZ) : 5Hz 3(F_10HZ) : 10Hz
	返回值	void
	備註	—

## Arduino Lib 下載及安裝

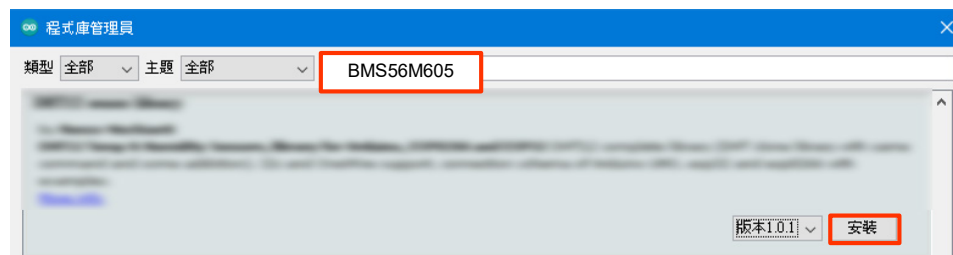
BMS56M605 Library：可參考下面兩種方法安裝 BMS56M605 的 Arduino Library

### 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 BMS56M605 → 安裝



搜索安裝流程 1

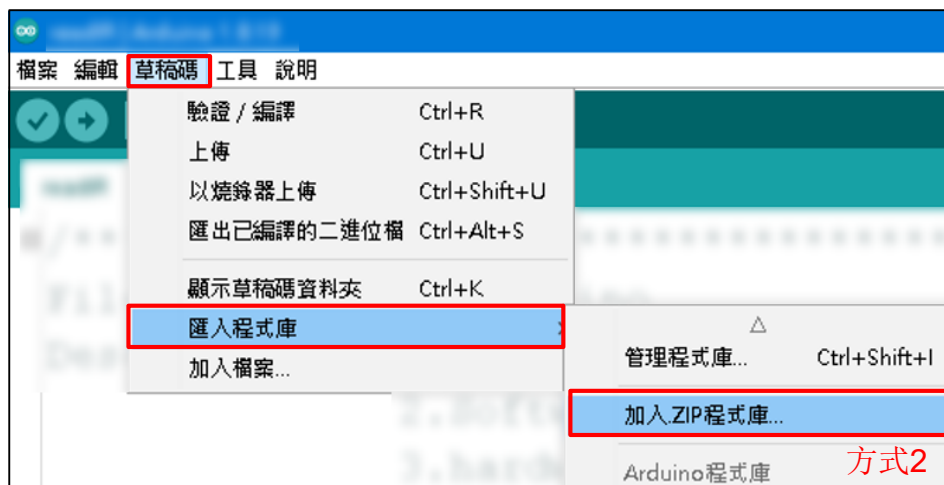


搜索安裝流程 2

## 方式 2：添加 .ZIP 程式庫 · 需提前下載 .ZIP 程式庫

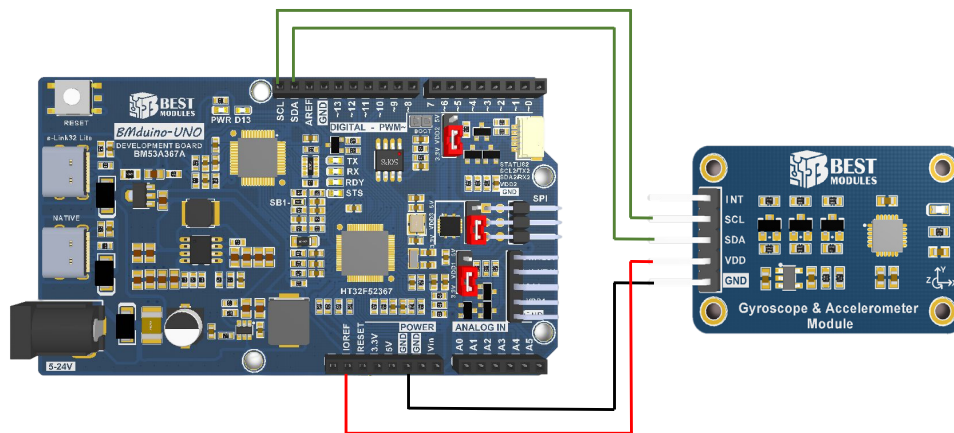
下載方法：打開倍創官方網站 (<https://www.bestmodulescorp.com/bms56M605.html>) 檔目錄下的 Arduino 範例程式 (BMS56M605 Library)。

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



## Arduino 範例

### 範例 1：readAccelerationAndGyroscopeAndTemperature



實物連接示意圖

範例實現功能：以 I<sup>2</sup>C 方式與模組通訊，獲取感測器 6 軸資料以及溫度值，並在序列埠監視視窗上顯示。

1. 範例打開：檔案 → 範例 → Lib 選擇 (BMS56M605) → 選擇範例 (readAccelerationAndGyroscopeAndTemperature)

## 2. 範例說明：

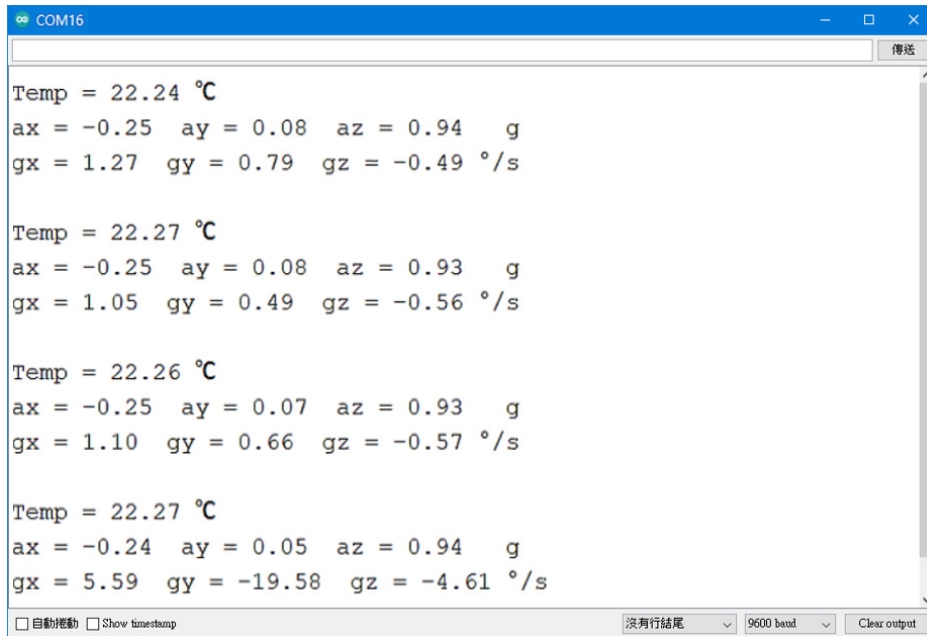
## a. 構建 &amp; 初始化對象

```
#include "BMS56M605.h"
BMS56M605 Mpu(8);      // INT Pin 連接 D8
void setup()
{
  Mpu.begin();         // 模組初始化
  Serial.begin(9600);  // 序列埠監視視窗初始化，鮑率為 9600
}
```

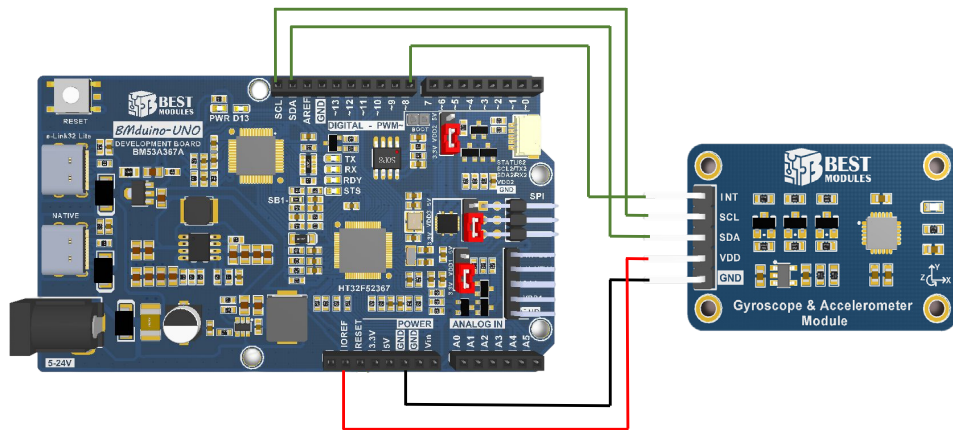
## b. 獲取感測器 6 軸資料以及溫度值並在序列埠監視視窗中顯示

```
void loop()
{
  Mpu.getEvent();      // 獲取三軸加速度、角速度以及環境溫度
  /* 溫度輸出，單位 °C */
  Serial.print("Temp = ");
  Serial.print(Mpu.temperature);
  Serial.println(" °C");
  /* 三軸加速度輸出，單位：g */
  Serial.print("ax = ");
  Serial.print(Mpu.accX);
  Serial.print("ay = ");
  Serial.print(Mpu.accY);
  Serial.print("az = ");
  Serial.print(Mpu.accZ);
  Serial.println(" g");
  /* 三軸角速度輸出，單位：°/s */
  Serial.print("gx = ");
  Serial.print(Mpu.gyroX);
  Serial.print("gy = ");
  Serial.print(Mpu.gyroY);
  Serial.print("gz = ");
  Serial.print(Mpu.gyroZ);
  Serial.println(" °/s");
  Serial.println();
  delay(1000);
}
```

3. 打開序列埠監視視窗，鮑率選擇 9600；序列埠監視視窗顯示資料如下：



## 範例 2 : getMotionStatus



實物連接示意圖

範例實現功能：以 I<sup>2</sup>C 方式與模組通訊，獲取運動檢測，每晃動一次模組，序列埠就會輸出一一次次數，累加至 0xffff 後清零，並在序列埠監視視窗上顯示。

1. 範例打開：檔案 → 範例 → Lib 選擇 (BMS56M605) → 選擇範例 (getMotionStatus)

2. 範例說明：

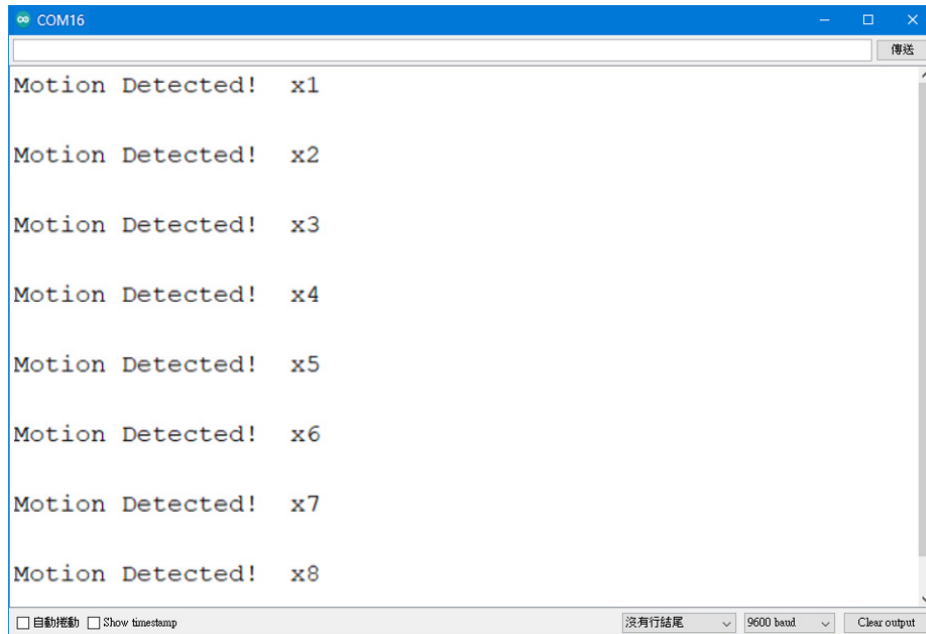
a. 構建 & 初始化對象

```
#include "BMS56M605.h"
BMS56M605 Mpu(8); // INT Pin 連接 D8
uint16_t cnt = 0;
void setup()
{
  Mpu.begin(); // 模組初始化
  Serial.begin(9600); // 序列埠監視視窗初始化·鮑率為 9600
  Mpu.setINT(MOTION_MODE,true); // 選擇中斷方式為運動模式
  Mpu.setInterruptPinPolarity(active_LOW); // 選擇低準位有效
  Mpu.setMotionThreshold(1); // 設定運動閾值為 1mg
  Mpu.setMotionDuration(30); // 設定持續運動時間為 30ms 時觸發
}
```

b. 獲取感測器 6 軸資料以及溫度值並在序列埠監視視窗中顯示

```
void loop()
{
  if( Mpu.getINT () == 0) // 獲取中斷狀態
  {
    cnt++;
    if(cnt >= 0xffff) cnt = 0;
    Serial.print( "Motion Detected! x" );
    Serial.println(cnt);
    Serial.println();
  }
}
```

3. 打開序列埠監視視窗·鮑率選擇 9600；序列埠監視視窗顯示資料如下：



The screenshot shows a serial monitor window titled "COM16". The output consists of eight lines, each displaying "Motion Detected! x" followed by a number from 1 to 8. At the bottom of the window, there are control options: "自動捲動" (checked), "Show timestamp" (unchecked), "沒有行結尾" (dropdown), "9600 baud" (dropdown), and "Clear output" (button).

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。