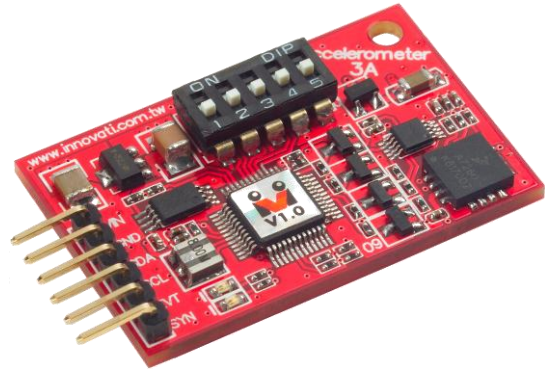


Accelerometer 3A

三軸加速度感測模組

版本: V2.0



產品介紹: 利基 Accelerometer 3A 模組是一個簡單易用的高精確度三軸加速度感測模組，透過 cmdBUS 與 Arminno 連接，可以用簡單的指令取得三個軸向的加速度值，或是加速度與軸向的夾角。

應用方向:

- 靜態傾斜角度的量測，控制載具維持平衡。
- 動態加速度的量測，感測受力大小與方向。
- 靜態加速度的量測，可以偵測重力方向。

產品特色:

- 數位化量測三個軸向的加速度值。
- 提供四種精準度(1.5g, 2g, 4g, 6g)，可隨時以軟體指令切換，配合不同量測需求。
- 以度為單位量測加速度與軸向角度值。
- 提供 2D 角度回傳，並且可任意選擇所要量測的兩個軸向。
- 提供 3D 角度回傳。
- 偵測角度可以精準到以度為單位。
- 能偵測範圍達 ± 6 g 的加速度值。
- 感測頻率能選擇五種更新速率(100 Hz, 50 Hz, 25 Hz, 10 Hz, 1 Hz)，並能隨時以指令動態更換更新頻率。
- 可以直接以指令讀取各軸向的數位電壓值。
- 可透過 cmdBUS，下達指令。

連接方式: 直接將 ID 開關撥至欲設定的編號，再將 cmdBUS 連接至 Arminno 上對應的腳位，就可透過 Arminno 執行操作。

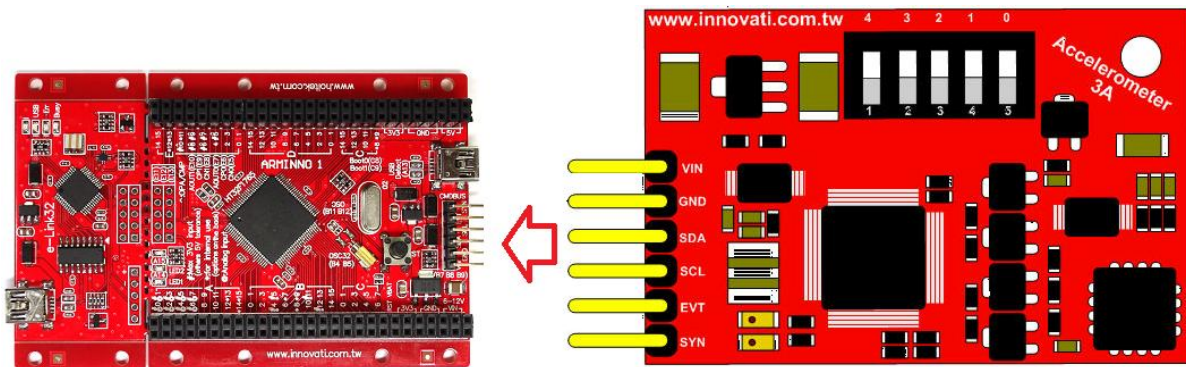
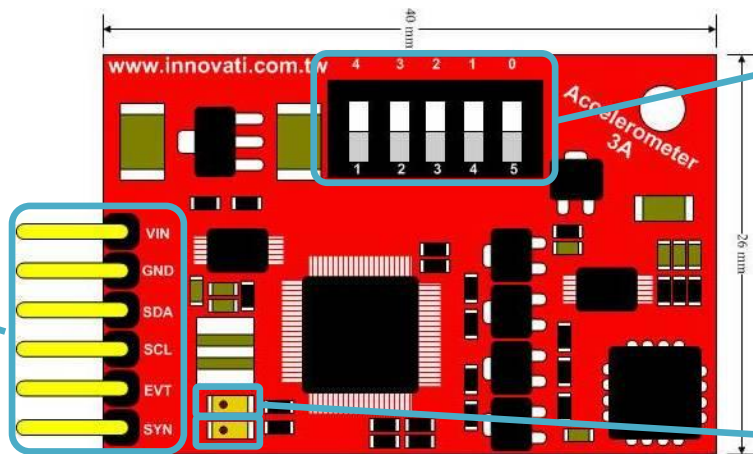


圖 1: 與 Arminno 連接

產品規格:

cmdBUS 接腳，將此處腳位與 Arminno 對應腳位相接，即可透過 Arminno 操控 Accelerometer 3A 模組(連接時請注意腳位對應，將 Vin 對接 Arminno 上的 Vin 腳位，若是腳位錯誤可能造成模組損毀)。

模組編號設定開關，由右至左以二進制設定 Accelerometer 3A 模組的模組編號，編號可以讓 Arminno 操控時，判斷想要控制的模組(請參考附錄 2)



橘色指令指示燈，閃爍代表模組與 Arminno 正在收送資

圖 2: 模組腳位與開關介紹

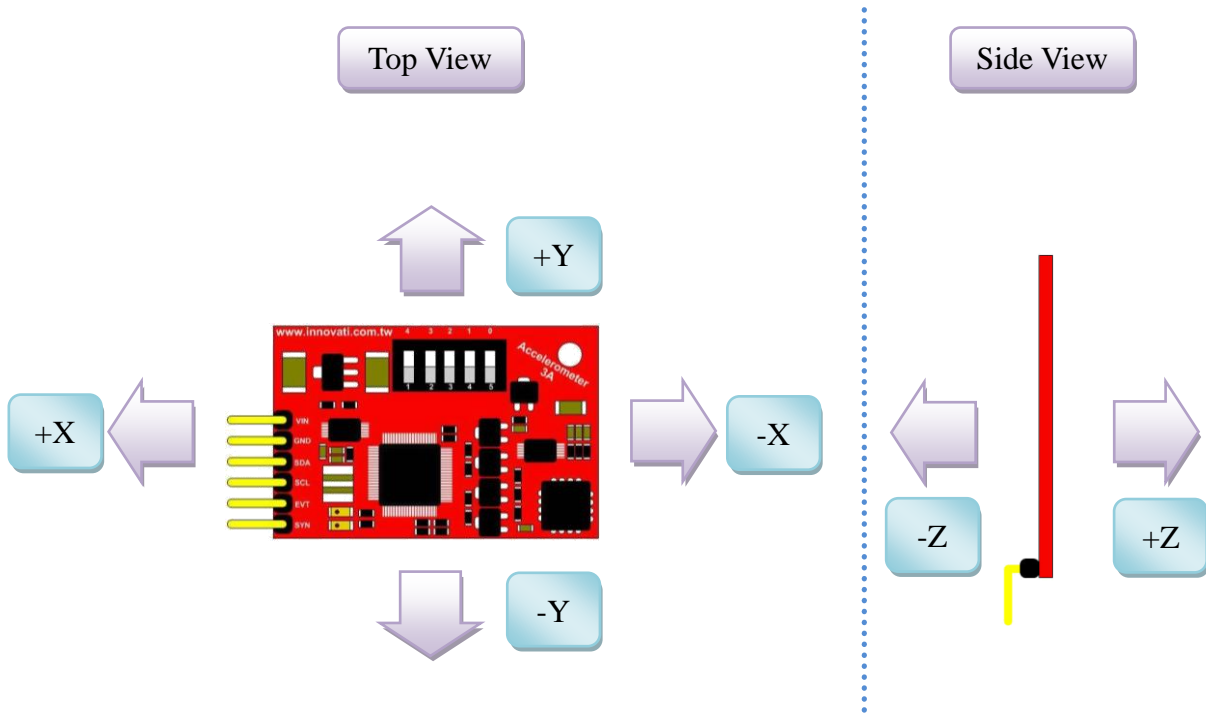


圖 3: 動態加速各軸變化值

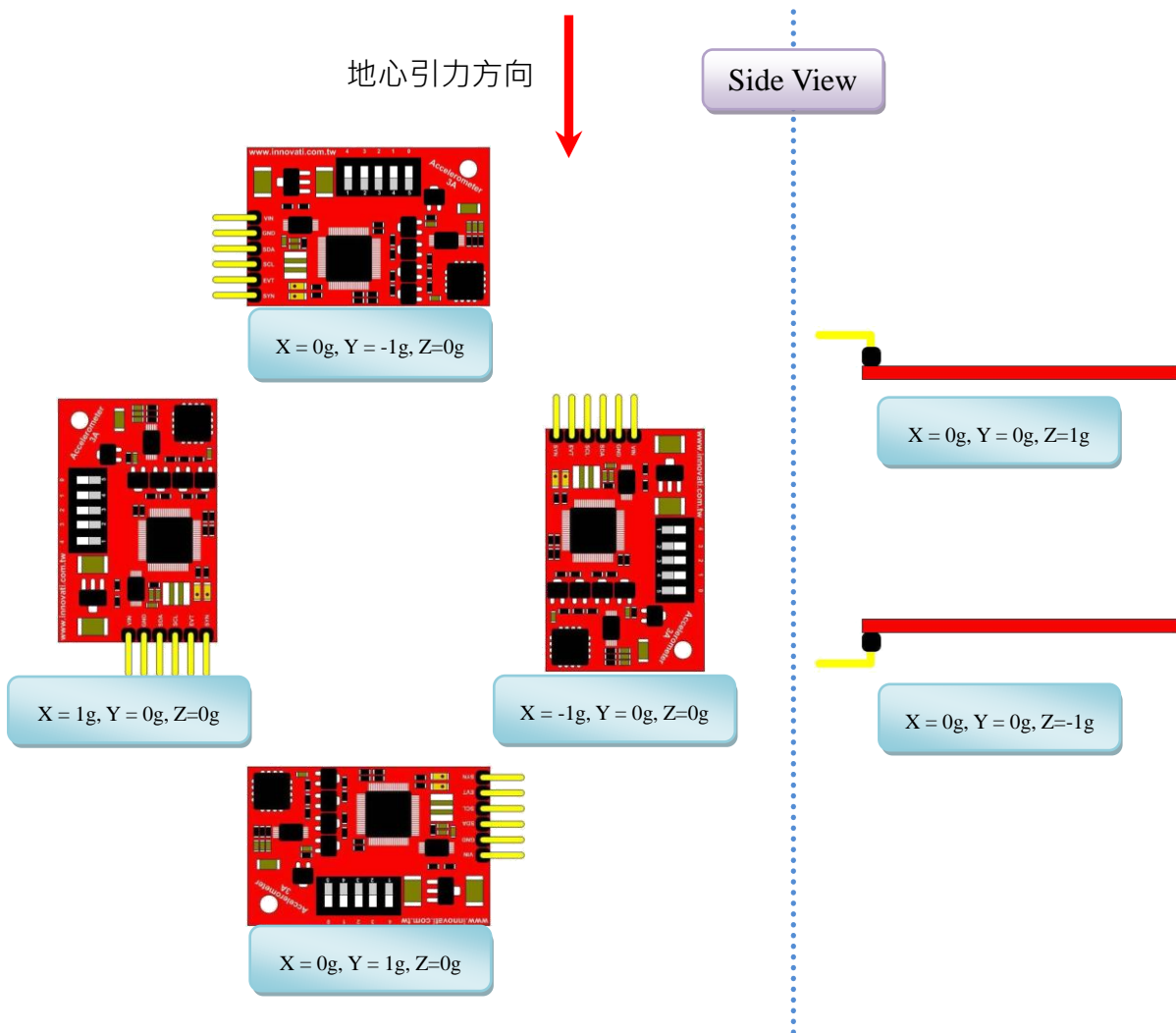


圖 4: 靜態各軸顯示值

操作注意事項:

- 請儘量將模組平放，以獲得較佳的量測值。

模組操作溫度 0 °C ~ 70 °C

模組儲存溫度 -40 °C ~ 125 °C

指令格式	指令功能
加速度與角度偵測相關指令	
GetXYZForce(int16_t &ForceX, int16_t &ForceY, int16_t &ForceZ)	取得 X, Y, Z 軸向的力量值，X 軸向存在 <i>ForceX</i> ，Y 軸向存在 <i>ForceY</i> ，Z 軸向存在 <i>ForceZ</i> ，單位會根據設定模式不同，請參考 SetMode 指令中，各個設定的單位值
GetAngle2D(uint16_t &Angle)	取得 2D 平面的合成力與主軸向的夾角存在 <i>Angle</i> ，單位為度，主軸向的設定可以用 SetAxis2D 指令更改， <i>Angle</i> 會回傳 0~359 間的整數
GetAngle3D(uint16_t &Angle1, uint8_t &Angle2)	取得 3D 的合成力於 XY 平面的分力與 X 軸向的夾角存在 <i>Angle1</i> ，單位為度，力與 Z 軸的夾角存在 <i>Angle2</i> ，單位為度， <i>Angle1</i> 會回傳 0~359 間的整數， <i>Angle2</i> 會回傳 0~179 間的整數
GetXADVal(uint16_t &Value)	取得各軸向(X, Y, Z 軸)受力的數位化電壓值，存於 <i>Value</i> 中， <i>Value</i> 會回傳 0~65535 間的整數值
GetYADVal(uint16_t &Value)	
GetZADVal(uint16_t &Value)	
各項設定相關指令	
SetMode(uint8_t Mode)	根據輸入的 <i>Mode</i> 值，設定加速度感測靈敏度，預設值為 0。 <i>Mode</i> 可以輸入 0~3: <i>Mode</i> = 0 → 當量測到 Force = 800 等於 1g，適合量測±1.5 g 內的加速度值 <i>Mode</i> = 1 → 當量測到 Force = 600 等於 1g，適合量測±2 g 內的加速度值 <i>Mode</i> = 2 → 當量測到 Force = 400 等於 1g，適合量測±4 g 內的加速度值 <i>Mode</i> = 3 → 當量測到 Force = 300 等於 1g，適合量測±6 g 內的加速度值
SetRefreshFreq(uint8_t Rate)	根據輸入的 <i>Rate</i> 值，設定模組量測值的更新速率，預設值為 1。 <i>Rate</i> 可以輸入為 0~4 等五種速率: <i>Rate</i> = 0 → 每 10 ms 更新角度值 (100 Hz) <i>Rate</i> = 1 → 每 20ms 更新角度值 (50 Hz) <i>Rate</i> = 2 → 每 40ms 更新角度值 (25Hz) <i>Rate</i> = 3 → 每 100ms 更新角度值 (10 Hz)

	Rate = 4 → 每 1000 ms 更新角度值 (1 Hz)
uint8_t Status = GetRefreshStatus(void)	讀取更新狀態，當 Status 回傳值為 1 時，代表量測值已經被更新過，此時會把內建的狀態值設定為 0，直到模組再次更新量測值，才會設定為 1
SetAxis2D(uint8_t Type)	以 Type 設定 2D 判斷時的設定軸， Type 可以輸入 0~5，分別代表： 0→ X 為 0 度軸，Y 為 90 度軸 1→ Y 為 0 度軸，X 為 90 度軸 2→ X 為 0 度軸，Z 為 90 度軸 3→ Z 為 0 度軸，X 為 90 度軸 4→ Y 為 0 度軸，Z 為 90 度軸 5→ Z 為 0 度軸，Y 為 90 度軸

範例程式:

```
#include "arminno.h"

Accelerometer3A myG(2);      // 設定模組編號為 2

unsigned short g_wAngle1;
unsigned char g_bAngle2;
short g_iFx, g_iFy, g_iFz;

int main(void)
{
    myG.SetMode(0);          // 設定靈敏度模式為 0，適合量測正負 1.5g 的加速度值
    myG.SetRefreshFreq(3);  // 設定更新速率為每秒更新十次

































    myG.SetAxis2D(0);
    while(1)
    {
        myG.GetXYZForce(g_iFx, g_iFy, g_iFz);    // 取得 XYZ 三軸的加速度值
        myG.GetAngle3D(g_wAngle1, g_bAngle2);    // 取得 3D 和 2D 角度值
        printf("fx= %d\r\n", g_iFx);
        printf("fy= %d\r\n", g_iFy);
        printf("fz= %d\r\n", g_iFz);
        printf("3D Angle=%d\r\n", g_wAngle1);
        printf("2D Angle= %d\r\n", g_bAngle2);
        Pause(2000);
    }
}
```

附錄

1. 已知問題:

- v1.1: **GetAngle3D** 取得的 3D 角度，在 $Z>0$ ，XY 接近 0 時，會誤判為 90 度，當 $XY=0$ ， $Z<0$ 會回傳錯誤值。

2. 模組編號開關對應編號表:

	0		8		16		24
	1		9		17		25
	2		10		18		26
	3		11		19		27
	4		12		20		28
	5		13		21		29
	6		14		22		30
	7		15		23		31