

NKNUBLOCK馬達與感測器課程 硬體教學手冊

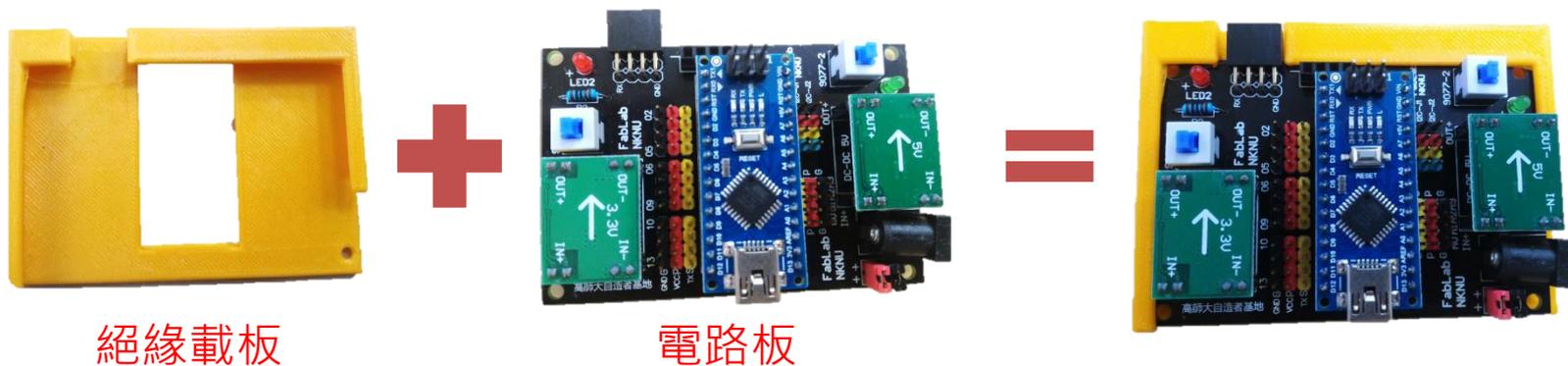
2019.11.22高師大自造者基地編輯

目錄

- (1)4060電控板
- (2)無源蜂鳴器
- (3)RGB LED模組
- (4)8*8點矩陣
- (5)搖桿
- (6)超音波感測器
- (7)伺服馬達
- (8)馬達控制模組
- (9)搖桿控制直流馬達
- (10)複合課程/電動柵欄

(1.1)4060電控板

- 注意：絕緣載板務必與電路板整合使用，避免短路危險
- 微控制器
 - 可以寫程式控制與它連接的零件(模組)
 - 透過腳位傳送/接收訊號
- FabLab NKTU開發了擴充板，解決了連接零件、電壓控制的困擾。
- 請用USB線連接電腦與板子。
 - 除了Power燈恆亮是不是還有個燈很規律的閃爍？為什麼會閃？



(1.2)腳位介紹

數位腳：2-13

輸入訊號：0與1

輸出訊號：0與1

PWM輸出：0-255

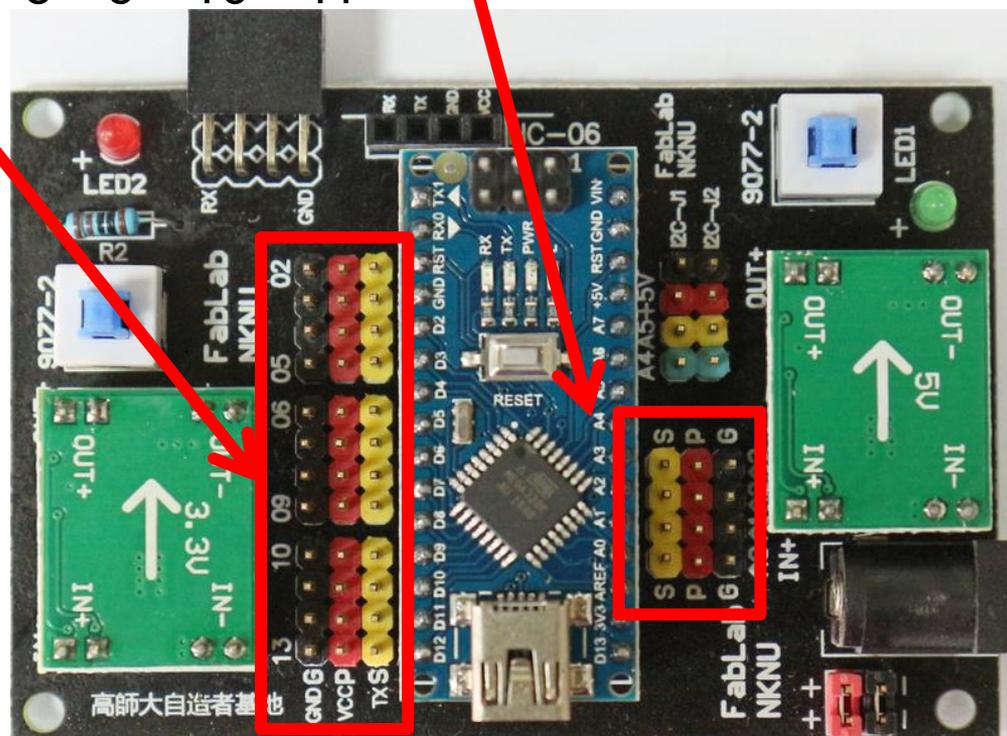
PWM腳位：3、5、6、9、10、11

類比腳：A0-A3

輸入訊號：0-1023

輸出訊號：0與1

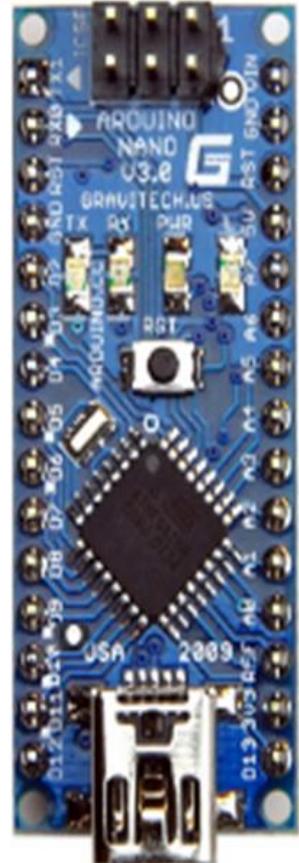
數位腳不夠用時也可當做數位腳來用



黃色為訊號腳位，
紅色為正極(5V)，
黑色為負極

補充：Arduino LIB使用Timer功能列表

Interrupt	Com	Pwm	Arduino	黑電控板腳位	黑電控板腳位	Arduino	Other	Com
	RXD		D0					
	TXD		D1			VIN		
			Reset			GND		
			GND			Reset		
INT0			D2	D02		5V		
INT1		timer2B	D3	D03		A7		
			D4	D04		A6		
		timer0B	D5	D05	A5	A5		SCL
		timer0A	D6	D06	A4	A4		SDA
			D7	D07	A3	A3		
			D8	D08	A2	A2		
	SS	timer1A	D9	D09	A1	A1		
	MOSI	timer1B	D10	D10	A0	A0		
	MISO	timer2A	D11	D11		AREF		
			D12	D12		3V3		
			D13	D13		D13	LED	SCK

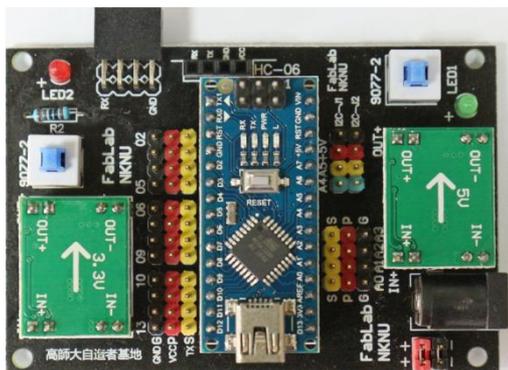


補充：Arduino LIB使用Timer功能列表

項目	Lib	Timer功能	備註														
蜂鳴器	TimerFreeTone	無															
超音波感測器	NewPing	Timer0	可使用非Timer功能的LIB,但量測距離穩定性須測試														
伺服馬達	Servo	Timer1															
溫溼度計	HT_sensor_library	Timer0	讀取時會用到delay()														
光流明計	BH1750-master	無															
顯示器	LedControl	無															
PWM	<table border="1"> <thead> <tr> <th>Timer output</th> <th>Arduino output</th> </tr> </thead> <tbody> <tr> <td>OC0A</td> <td>6</td> </tr> <tr> <td>OC0B</td> <td>5</td> </tr> <tr> <td>OC1A</td> <td>9</td> </tr> <tr> <td>OC1B</td> <td>10</td> </tr> <tr> <td>OC2A</td> <td>11</td> </tr> <tr> <td>OC2B</td> <td>3</td> </tr> </tbody> </table>	Timer output	Arduino output	OC0A	6	OC0B	5	OC1A	9	OC1B	10	OC2A	11	OC2B	3	Timer0~2	Timer0(D5、D6),Timer1(D9、D10)、Timer2(D3、D11)
Timer output	Arduino output																
OC0A	6																
OC0B	5																
OC1A	9																
OC1B	10																
OC2A	11																
OC2B	3																

電池使用注意事項(馬達與感測器教具不適用)

- 檢查外觀有沒有變形，流湯，發霉。
- 拔插電池請轉動黑色膠殼，不可拉線。
- 2顆18650串聯，充滿電時電壓可達8.4V
- 用完可隨時充電，不要把電耗盡，盡量維持在3.5V~4.0V。
- 使用有CE，UL認證的充電器。充電時避免接近火源或高於40°C時使用。
- 回收前，請用膠布包裹，避免短路自燃。
- USB與電池可同時接，不影響電壓。



電池接孔

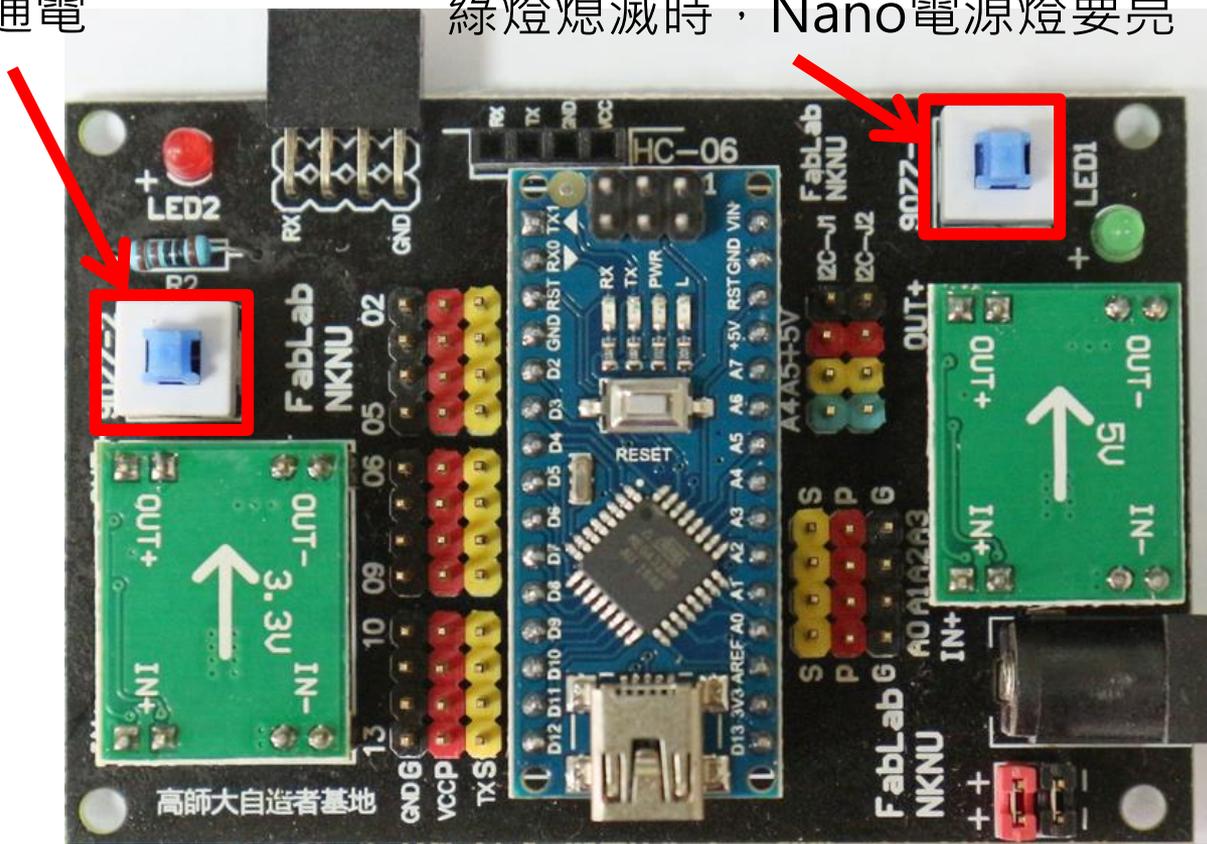


電源開關(馬達與感測器教具不適用)

WiFi電源開關，
紅燈亮時通電

主開關

綠燈亮表示電池有電，但板子沒過電。
綠燈熄滅時，Nano電源燈要亮



有用到WiFi時，先開主開關，再開WiFi開關

(1.3)寫程式前的準備

- 下載及安裝NKNUBLOCK
 - 下載位址：高師大自造者基地→總恆星基地→課程圖書館→程式及數位控制技術類
 - 請參閱NKNUBLOCK程式安裝教學說明.pdf



課程圖書館 [總恆星基地]

類別： 程式及數位控制技術類 標題： 關鍵字搜尋 搜尋

課程類別	發佈單位	標題
程式及數位控制技術類	國立高雄師範大學中心 基地	NKNUBLOCK v2.0.1 安裝檔

The screenshot shows the NKNUBLOCK web interface. The top navigation bar includes '檔案', '晶片燒錄', '網路設定', '連線', 'MIT教程', and 'NKNU教程'. The main content area displays a search filter for '程式及數位控制技術類' and a search button. Below the search results, a table lists the course details. On the right side of the interface, there is a sidebar with '舞台', 'Arduino IDE', and '腳位預設狀態' tabs, and a large orange banner for 'FabLab-NKNU' featuring a robot and the text '教育部 STEM+A 課程導向 數位自造教育扎根計畫'.

(1.4)NKNUBLOCK功能列

燒錄NKNUBLOCK韌體至Arduino nano板，只有NKNUBLOCK版本更新時才需要重新燒錄

檢查Arduino IDE程式碼

燒錄Arduino IDE code至程式碼Arduino nano

(1.5)寫程式前的準備

- 以USB線連接Nano及電腦→選擇com port→晶片燒錄→Arduino晶片→燒錄韌體



The screenshot shows the NKNUBLOCK web interface. The top navigation bar includes '檔案', '晶片燒錄', '網路設定', '連線', 'MIT教程', and 'NKNU教程'. The main interface has a 'COM : COM5' dropdown menu highlighted with a red box. Below it, a red arrow points to a text box containing the following instructions:

COM會自動偵測裝置連接的USB PORT
如果COM顯示空白，則點選USB驅動程式
→CH340 Driver，解壓縮後安裝驅動軟體，
且重新啟動NKNUBLOCK

(1.6)寫程式前的準備

- 燒錄成功後，畫面會跳出操作結束的訊息視窗。

```
C:\Windows\System32\cmd.exe
avrdude: safemode: hfuse reads as 0
avrdude: safemode: efuse reads as 0
avrdude: NOTE: "flash" memory has been specified, an erase cycle will be performed
To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "NKNUNano.hex"
avrdude: writing flash (29690 bytes):

Writing | ##### | 100% 8.57s

avrdude: 29690 bytes of flash written
avrdude: verifying flash memory against NKNUNano.hex:
avrdude: load data flash data from input file NKNUNano.hex:
avrdude: input file NKNUNano.hex contains 29690 bytes
avrdude: reading on-chip flash data:

Reading | ##### | 100% 6.48s

avrdude: verifying ...
avrdude: 29690 bytes of flash verified

avrdude: safemode: lfuse reads as 0
avrdude: safemode: hfuse reads as 0
avrdude: safemode: efuse reads as 0
avrdude: safemode: Fuses OK (E:00, H:00, L:00)

avrdude done. Thank you.
-----
燒錄成功
-----
請按任意鍵繼續
```

只有看到顯示「燒錄成功」才是成功完成燒錄韌體，燒錄動作超過20秒，未顯示「燒錄成功」字樣，即可判定燒錄失敗。NANO 晶片即可判定有問題。

(1.7)USB連線

- 連線→連線方式：USB→選擇com port→連線
- 成功連線後，連線狀態會顯示已連線(USB)

The screenshot displays the NKNUBLOCK software interface. At the top, the '連線' (Connection) menu is active, showing '連線模式: USB' (Connection Mode: USB) and 'COM: COM4'. A red box highlights the '連線' (Connect) button, with a hand cursor pointing to it. To the right, a red box highlights the '連線狀態: 已連接(USB)' (Connection Status: Connected (USB)) text. The interface also shows a '程式' (Program) editor on the left with various action blocks, and a '舞台' (Stage) area on the right displaying a 'FabLab-NKNU' logo and a robot image.

(1.8)NKNUBLOCK使用注意事項

- NKNUBLOCK 是用網頁模式運作的，執行中網頁不可切換到其他分頁，chrome 有資源節省政策，切換到別的頁面會造成背景的 NKNUBLOCK 頁面進入省電模式，會跑非常慢。

(1.9)NKNUBLOCK和Arduino

NKNUBLOCK 和Arduino 是不同國家的人，語言不通， NKNUBLOCK 的積木程式，Arduino 是讀不懂的，這時候出現NKNUBLOCK就是負責翻譯的中介軟體。她有三種任務：

- (1)翻譯
- (2)WiFi通訊
- (3)黑、灰積木。

(1) 翻譯：

要執行這個任務，就必需先在Arduino晶片植入一個高師大特製的翻譯韌體（這就是每次更新版的NKNUBLOCK出版時，就必需燒入最新版的Arduino 韌體，這樣新版的NKNUBLOCK的功能才發揮）

(1.10)NKNUBLOCK和Arduino

(2) WiFi通訊：

NKNUBLOCK的另一特色，就是可以透過WiFi通訊，這是全世界獨一無二的，一般教學軟體只能用USB線傳輸。但NKNUBLOCK要透過WiFi通訊，其通訊協定的韌體就燒入（植入）ESP8266-01s晶片中，這樣01s晶片的作業系統才讀得到NKNUBLOCK送過來訊息，再繼續傳送給Arduino晶片，此時預植在Arduino晶片內的韌體就把收到的Scratch語言翻譯成Arduino的語言，A先生就了解S小姐傳來的命令，再去命令A先生所管轄的所有小兵馬達、感測器去工作，每當小兵完成一個指令的小工作，就馬上回覆完成工作的訊息給S小姐，S小姐才會執行下一個積木的指令，這樣下指令執行指令回傳完成指令訊息一個來回，至少要33ms的時間（受限於瀏覽器更新頁面速度1秒30個頁面，所以若開啟2個瀏覽器頁面，執行一個小積木就需要至少66ms的時間，A先生的每個小兵的動作就會變遲緩）

(1.11)NKNUBLOCK和Arduino

(3) 黑、灰積木：

NKNUBLOCK第三個重要任務就是作一些本來MIT-S小姐不會的事，控制小兵的程式碼，是原本不存在於MIT-S小姐的技能，這些控制小兵的程式碼都是用Arduino 國家的話寫的，MIT-S小姐根本讀不懂。當A先生想新增各式各樣的小兵來幫他工作，就必需設計黑積木給S小姐取用。

(2.1)無源蜂鳴器

- 此「源」不是指電源。而是指震盪源。
- 無源內部不帶震盪源，所以如果用直流信號無法令其鳴叫。必須用2K~5K的方波去驅動它
- 聲音頻率可控，可以做出“多來米發索拉西”的效果。



音階	音符											
	C (Do)	C#	D (Re)	D#	E (Mi)	F (Fa)	F#	G (So)	G#	A (La)	A#	B (Si)
低音	262	277	294	311	330	349	370	392	415	440	466	494
中音	523	554	587	622	659	698	740	784	831	880	932	988
高音	1046	1109	1175	1245	1318	1397	1480	1568	1661	1760	1865	1976

(2.2)避免蜂鳴器發熱使用須知

- 使用蜂鳴器積木之前或之後，要把蜂鳴器腳位的電位拉到 High
- 因為硬體原廠設計是「低電位 Low觸發」，即使沒在跑蜂鳴器程式時，平時是處于低電位Low，即會一直送電觸發蜂鳴器，因為是DC直流，所以沒聲音，但觸發蜂鳴器即會消耗電功率，物理能量守恆原理，電功率沒轉換成聲能就轉換成熱能，蜂鳴器就會發熱發燙。
- 蜂鳴器.sb3



(2.3) NKNUBLOCK腳位預設狀態



- 程式停止的**STOP**按下後回復腳位預設狀態，蜂鳴器所在第8腳位，回至高電位避免持續觸發，產生噪音。
- USB連線成功後，步驟1 選擇公版教學版，步驟2 保存至Arduino

下列情形會將腳位設為指定狀態

1. 設備剛開機的時候會設定一次
2. 按下NKNUBLOCK停止按鈕的時候

保存到Arduino 情境模板：公版教學版 讀取

數位腳位	高電位	備註	類比腳位	數位	備註
D2	<input type="checkbox"/>	-	-	<input type="checkbox"/>	-
D3	<input type="checkbox"/>	PWM-T2	-	<input type="checkbox"/>	-
D4	<input type="checkbox"/>	-	-	<input type="checkbox"/>	-
D5	<input type="checkbox"/>	PWM-T0	-	<input type="checkbox"/>	-
D6	<input type="checkbox"/>	PWM-T0	A7	<input type="checkbox"/>	-
D7	<input type="checkbox"/>	-	A6	<input type="checkbox"/>	-
D8	<input checked="" type="checkbox"/>	-	A5	<input type="checkbox"/>	I2C
D9	<input type="checkbox"/>	PWM-T1	A4	<input type="checkbox"/>	I2C
D10	<input type="checkbox"/>	PWM-T1	A3	<input type="checkbox"/>	-
D11	<input type="checkbox"/>	PWM-T2	A2	<input type="checkbox"/>	-
D12	<input type="checkbox"/>	-	A1	<input type="checkbox"/>	-
D13	<input type="checkbox"/>	LED	A0	<input type="checkbox"/>	-

備註：

- 使用 伺服馬達 腳位 6 角度為 90 註 時D9,D10的PWM會失效
- 使用 腳位 2 播放音調,頻率為 Do,262 時D3,D11的PWM會失效
- 蜂鳴器長時間在低電位會發燙，不播放時應拉高該腳位
- EEPROM 901~1024為保留區間，如無原因請不要複寫

(2.4)無源蜂鳴器

- 電子琴-使用鍵盤來彈奏音樂
- 電子琴.sb3



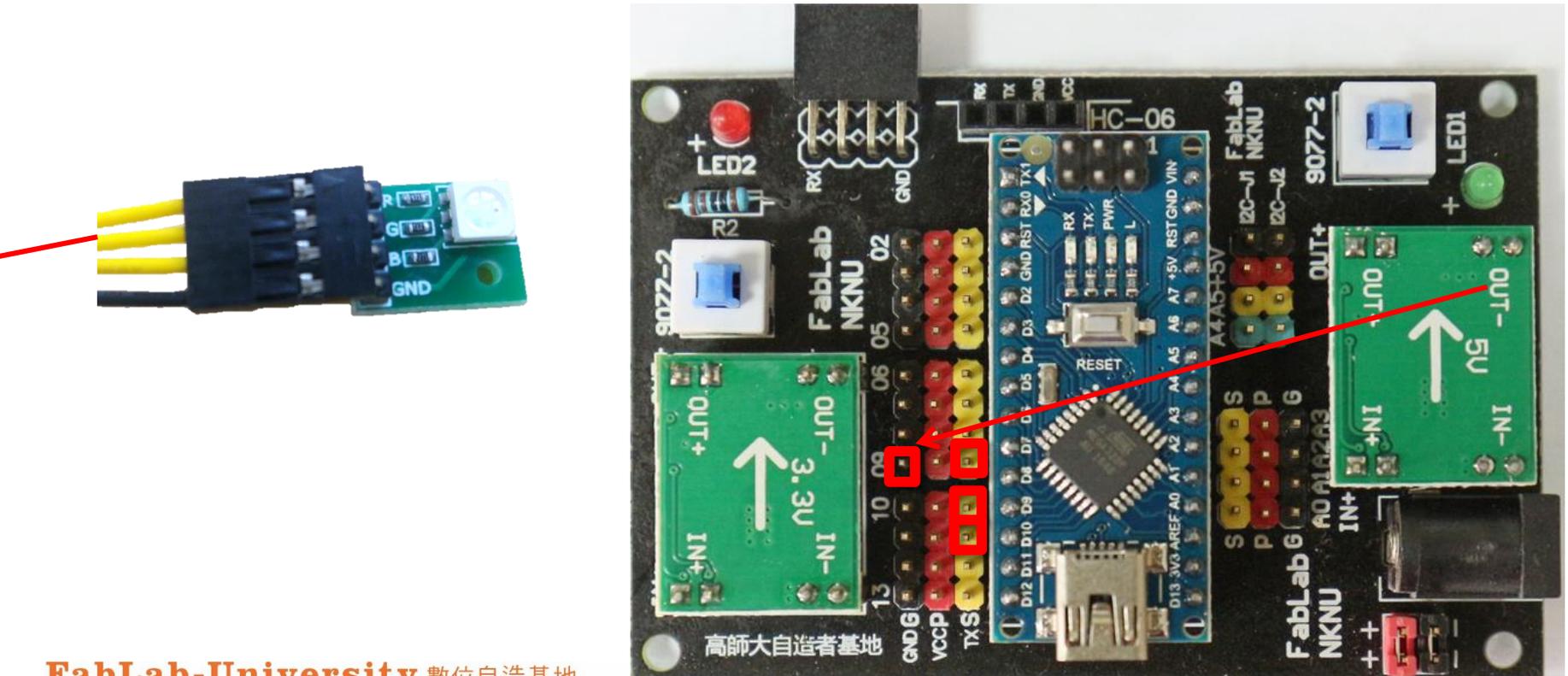
Scratch code for piano keys 1-3. The code starts with a '當被點擊' (When clicked) event block. It sets digital pin 8 to output high voltage (1). A variable 'time' is set to 500. A '重複無限次' (Repeat forever) loop contains three '如果' (If) blocks. Each '如果' block checks if a key (1, 2, or 3) is pressed. If pressed, it plays a note on pin 8 for 'time' ms. The notes are Do, 262 Hz; Re, 294 Hz; and Me, 330 Hz.



Scratch code for piano keys 4-7. It continues the '重複無限次' (Repeat forever) loop with four more '如果' (If) blocks. Each '如果' block checks if a key (4, 5, 6, or 7) is pressed. If pressed, it plays a note on pin 8 for 'time' ms. The notes are Fa, 349 Hz; So, 392 Hz; La, 440 Hz; and Si, 494 Hz. The code ends with a '設定數位腳位' (Set digital pin) block setting pin 8 to output high voltage (1).

(3.1)RGB LED模組

- 接上RGB LED模組
 - GND(G)是接地(負極)，線與接腳用黑色表示
 - VCC(P)是正極，線與接腳用紅色表示。
 - 黃色腳用來傳送、接收訊號，寫程式就是控制這隻腳。



(3.2)RGB LED模組

- 將R改接到腳位9，G接到10，B接到11，怎麼讓它亮？顏色有什麼變化？
- RGB-1.sb3
- RGB-2.sb3



怎樣讓模組重複顯示紅、綠、藍三個顏色？

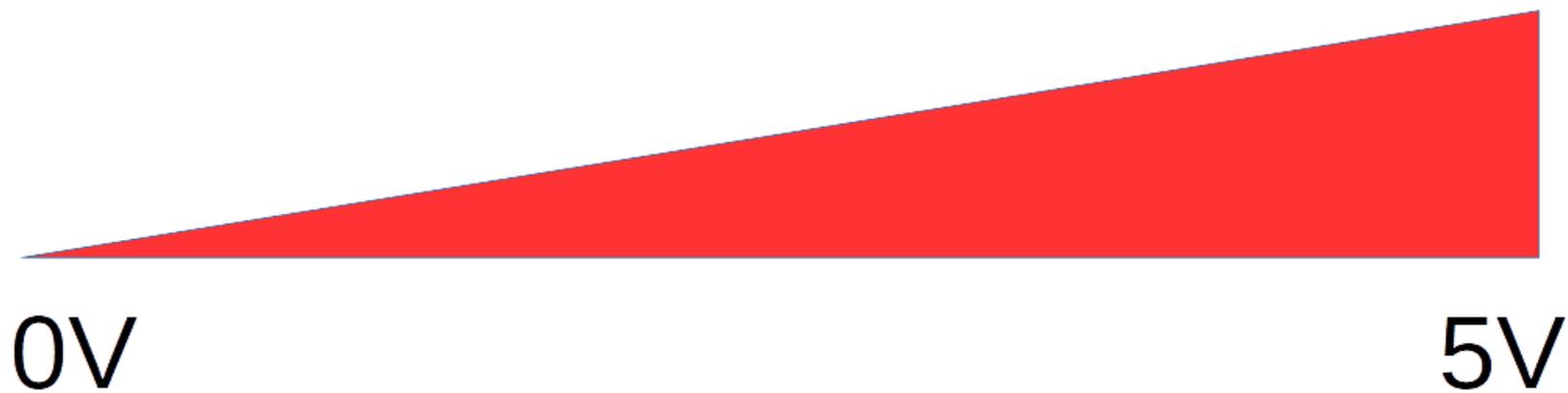
(3.3)RGB LED模組—調出彩虹的顏色

- 數位輸出只有0與1兩種訊息，要調出彩虹的顏色需要有強弱之分，所以要用到另一個指令：類比輸出。

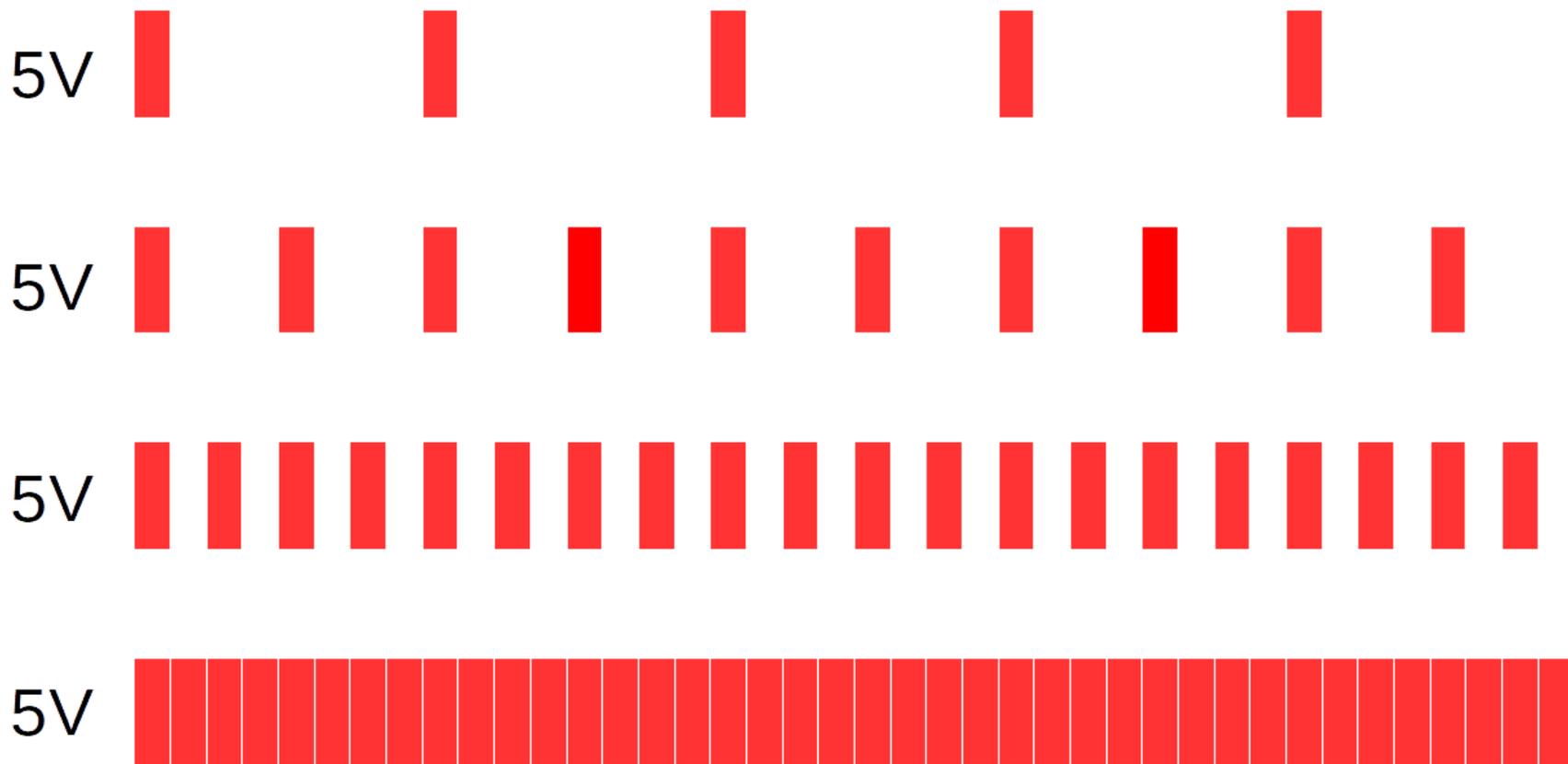


類比輸出範圍：0~255

(3.4)一般電壓控制



(3.5)PWM(類比輸出)

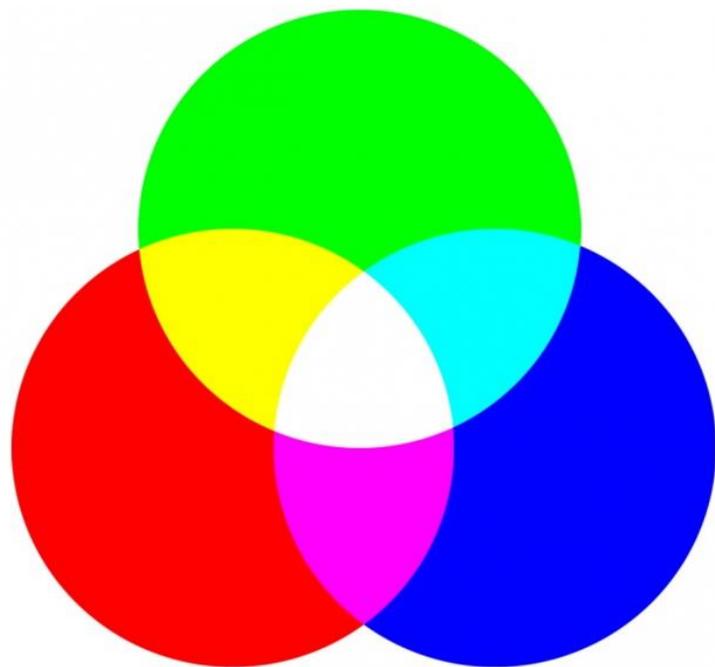


(3.6)PWM(類比輸出)實驗

- 單獨只寫底下兩個指令，然後把RGB LED 的杜邦線搖一搖，比較兩者的差異。



(3.7)RGB LED模組—調出彩虹的顏色



	R(Red)	G(Green)	B(Blue)
1紅	240	0	0
2橙	160	80	0
3黃	120	120	0
4綠	0	240	0
5藍	0	0	240
6靛(青)	0	120	120
7紫	120	0	120
8白	80	80	80

(3.8)用清單、變數記錄、呈現測試資料

- RGB-3.sb3

	R	G	B
1	210	0	0
2	180	30	0
3	150	60	0
4	0	160	0
5	0	0	255
6	0	80	155
7	105	0	155
8	70	50	90

+ 長度 8 = + 長度 8 = + 長度 8 =

i 5

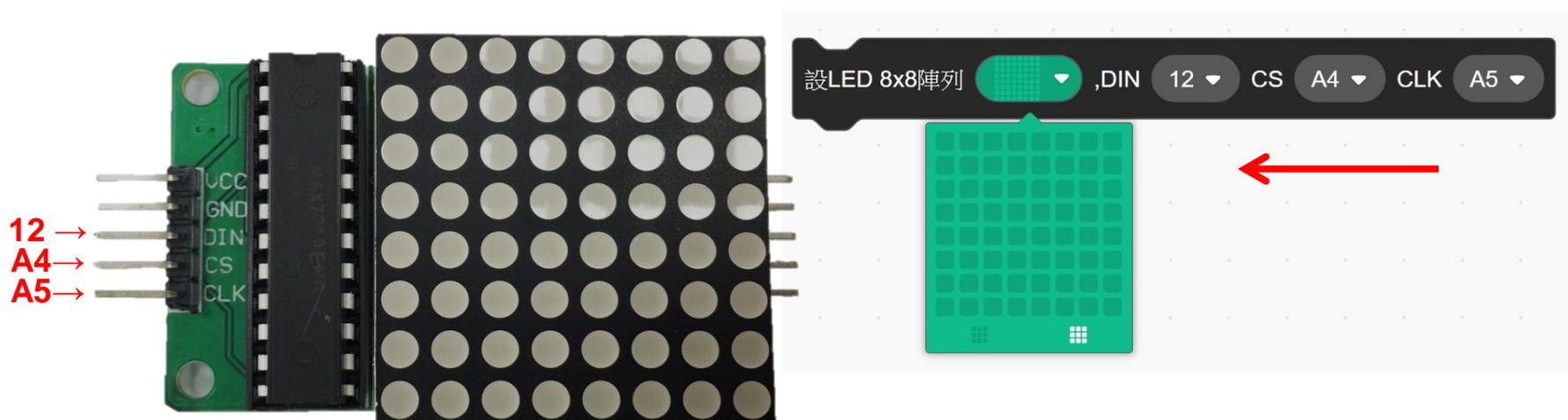
```
當 被點擊  
變數 i 設為 1  
重複無限次  
  設定PWM腳位 9 輸出為 R 的第 i 項 註 R  
  設定PWM腳位 10 輸出為 G 的第 i 項 註 G  
  設定PWM腳位 11 輸出為 B 的第 i 項 註 B  
  變數 i 設為 i 除以 8 的餘數  
  變數 i 改變 1  
  等待 1 秒
```

(3.9) 找出全部PWM腳位

- 除了9、10、11，還有3隻腳有類比輸出(PWM)的功能，請找出來。
- 將全部PWM腳位加註到程式中

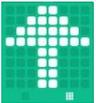
(4.1) 8*8點矩陣

- LED 8*8點矩陣，可以單獨控制1個點~64個點的亮、暗，也可創造出不同的點陣圖形，生活中常運用在號誌燈、數位刊板等。
- DIN(訊號)腳位12、CS(晶片選擇)腳位A4、CLK(時脈)腳位A5。



在點矩陣上，亮、暗所呈現的點

(4.2)用鍵盤控制8*8點矩陣

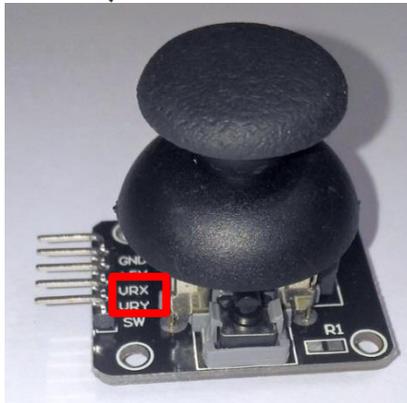
- 任務：按鍵盤中的上、下、左、右，如何出現對應的圖形。
- 如何將上面的話語轉譯成程式？
 - 換句話說：如果按鍵盤的上，8*8點矩陣就會出現  的圖案。
- 當按下字母或數字呢？
- 有看過行人穿越號誌燈嗎？
- 喜怒哀樂四種表情能用點矩陣呈現嗎？
- 點矩陣-鍵盤遙控.sb3



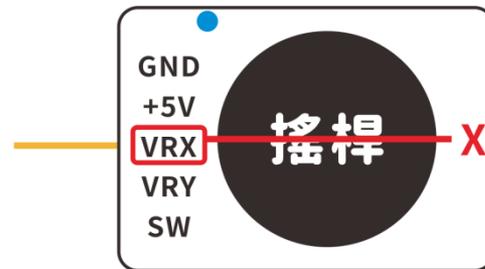
(5.1) 搖桿

- VRX與VRY可以量測電壓的強弱變化，變化範圍：0~1023。
- SW與按鈕相同，只有0與1兩種狀況。

(請依照此方向擺放搖桿)



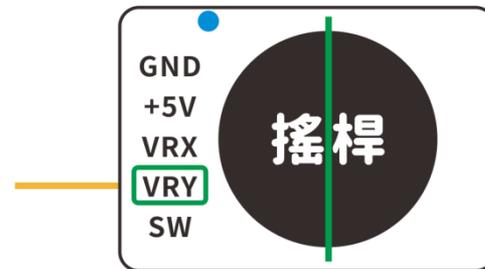
VRX線控制搖桿的X軸方向



原點位置(固定)



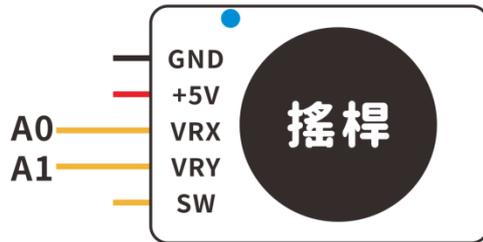
Y



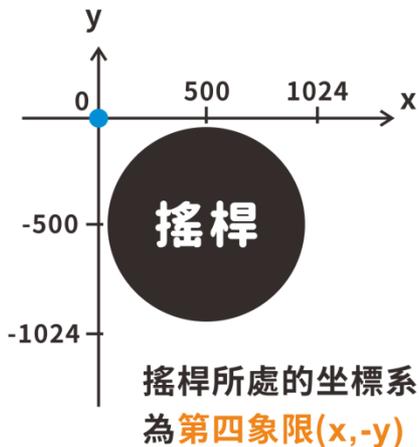
VRY線控制搖桿的Y軸方向

(5.2) 搖桿

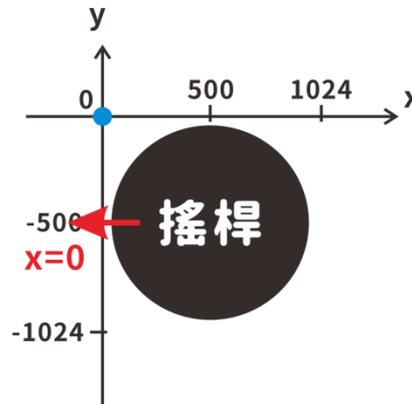
(請依照此方向擺放搖桿)



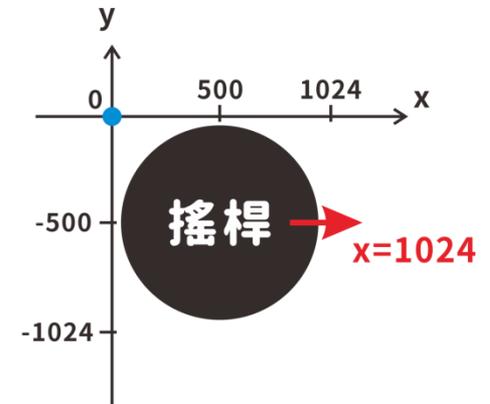
電控板上
VRX線已接A0腳位，
VRY線已接A1腳位。



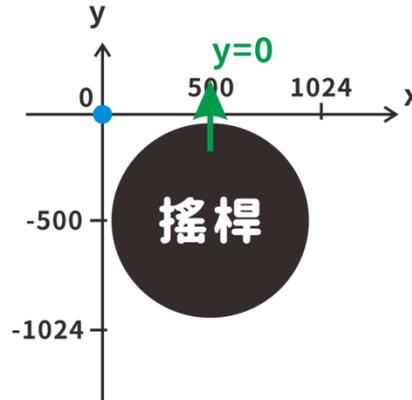
搖桿往左



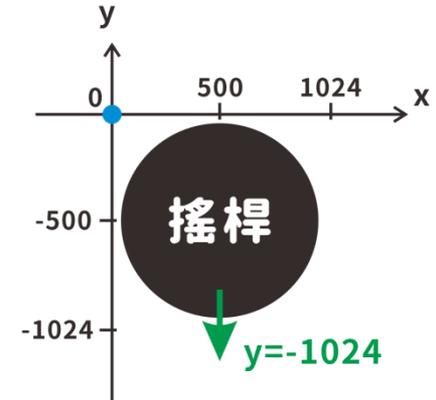
搖桿往右



搖桿往上



搖桿往下



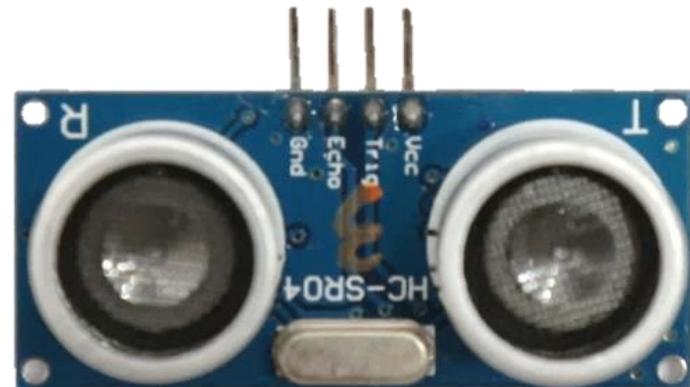
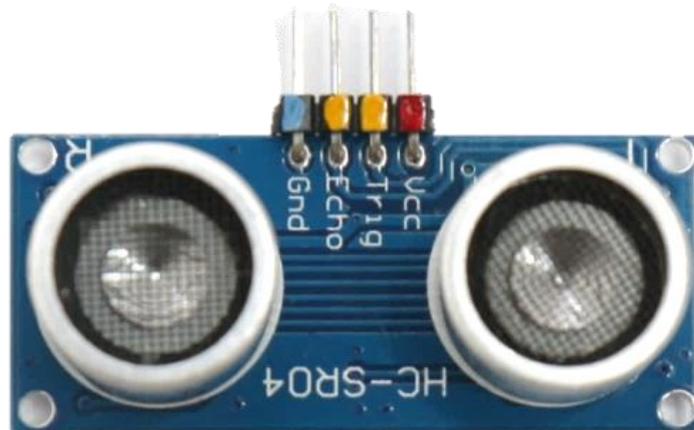
(5.3) 觀察搖桿輸入值的變化

- 將VRX接到A0；VRY接到A1，SW接到7
- 搖桿.sb3

The image shows a Scratch script for monitoring joystick input. The script starts with a 'When clicked' event block. It then sets three variables: X to 0, Y to 0, and Button to 0. A 'Repeat forever' loop contains three 'Set variable to' blocks: X is set to 'Read analog pin A0' (labeled 'Joystick X'), Y is set to 'Read analog pin A1' (labeled 'Joystick Y') multiplied by -1, and Button is set to 'Read digital pin 7' (labeled 'INPUT_PULLUP'). To the right of the script is a variable monitor showing the current values: X is 512, Y is 522, and Button is 0.

(6.1) 超音波感測器

- 由Trig發出訊號，Echo接收訊號(兩個都接黃色腳位，習慣Trig腳位在前，Echo在後，預設是Trig在A2和Echo在A3)
- 感應角度為15度，有效感測距離約2cm~400cm(?)，精度為0.3cm



(6.2)超音波感測器

- 從訊號發出到接收到的時間* 聲音傳送速度/2，就是超音波感測器與障礙物的距離。
- 當感測不出距離時會傳回0，所以寫程式時記得過濾無效資訊
- 超音波.sb3



(7.1) 伺服馬達-SG90

- 只能在0度-180度之間轉動，轉到指定位置後就會定住不動，這時不可強力扳動馬達，否則馬達裡的齒輪組會壞掉。
- 馬達耗電量大，要外接電池。



(7.2) 伺服馬達

- 使用前先歸零再固定搖臂。
 - 將SG90線接到數位腳6，寫程式讓它轉到90度
 - 伺服馬達轉到指定角度後會固定在那裡，不可強制轉開，否則馬達裡的齒輪組會損壞

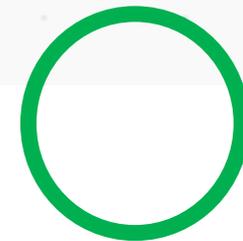


橘色為訊號線接黃色腳位
紅色接Vcc
棕色接GND



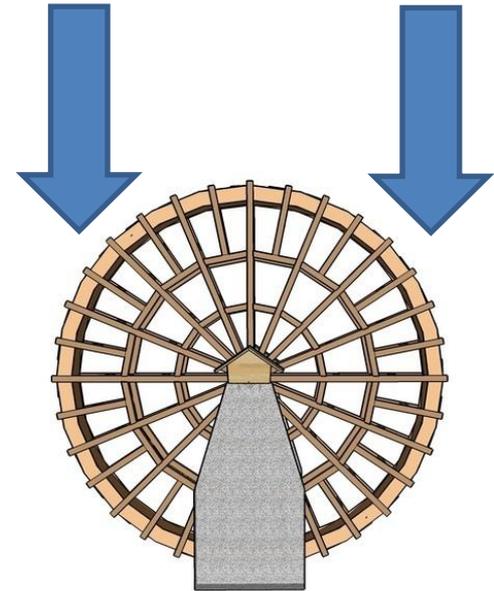
(7.3) 伺服馬達

- 分出0度與180度方向，記得加上等待時間，讓馬達有時間轉到指定的角度
- 伺服馬達.sb3



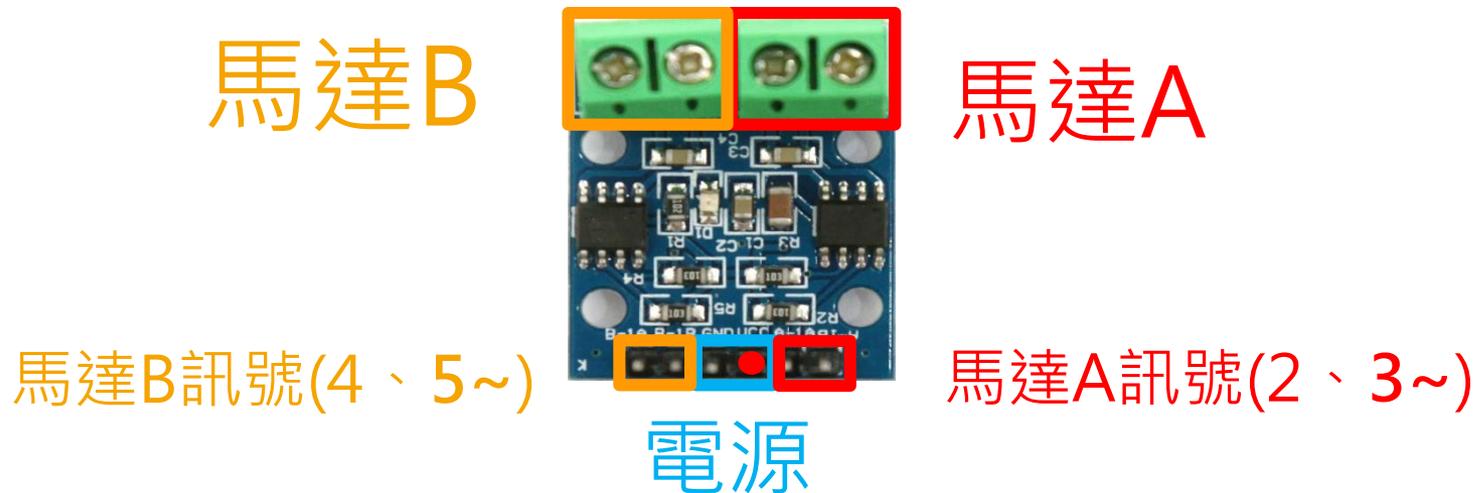
(8.1) 直流減速馬達

- N20減速馬達，採購時需指定電壓與轉速，例如6V100轉
- 利用接線兩端電位差控制轉速與轉動方向。
- 兩端電位差距越大轉速越快。
- 兩端電位相等時不會轉動。
- 將高低兩端電位對調，轉向就相反。

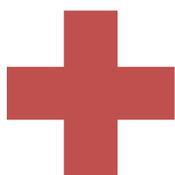
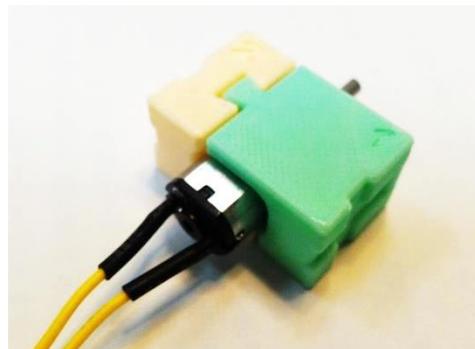


(8.2)馬達控制模組

- L9110S
- 電壓範圍2.5V~12V
- 為什麼不直接將馬達接到擴充板的IO腳位？
 - 馬達也可當做發電機，轉動時會有電流灌回Nano

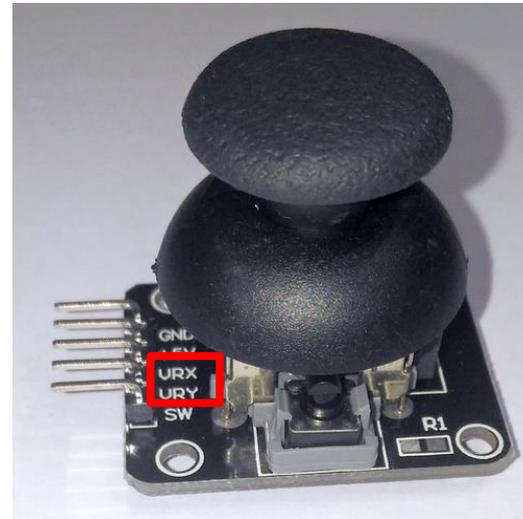


組裝轉動輪胎(參考教材)

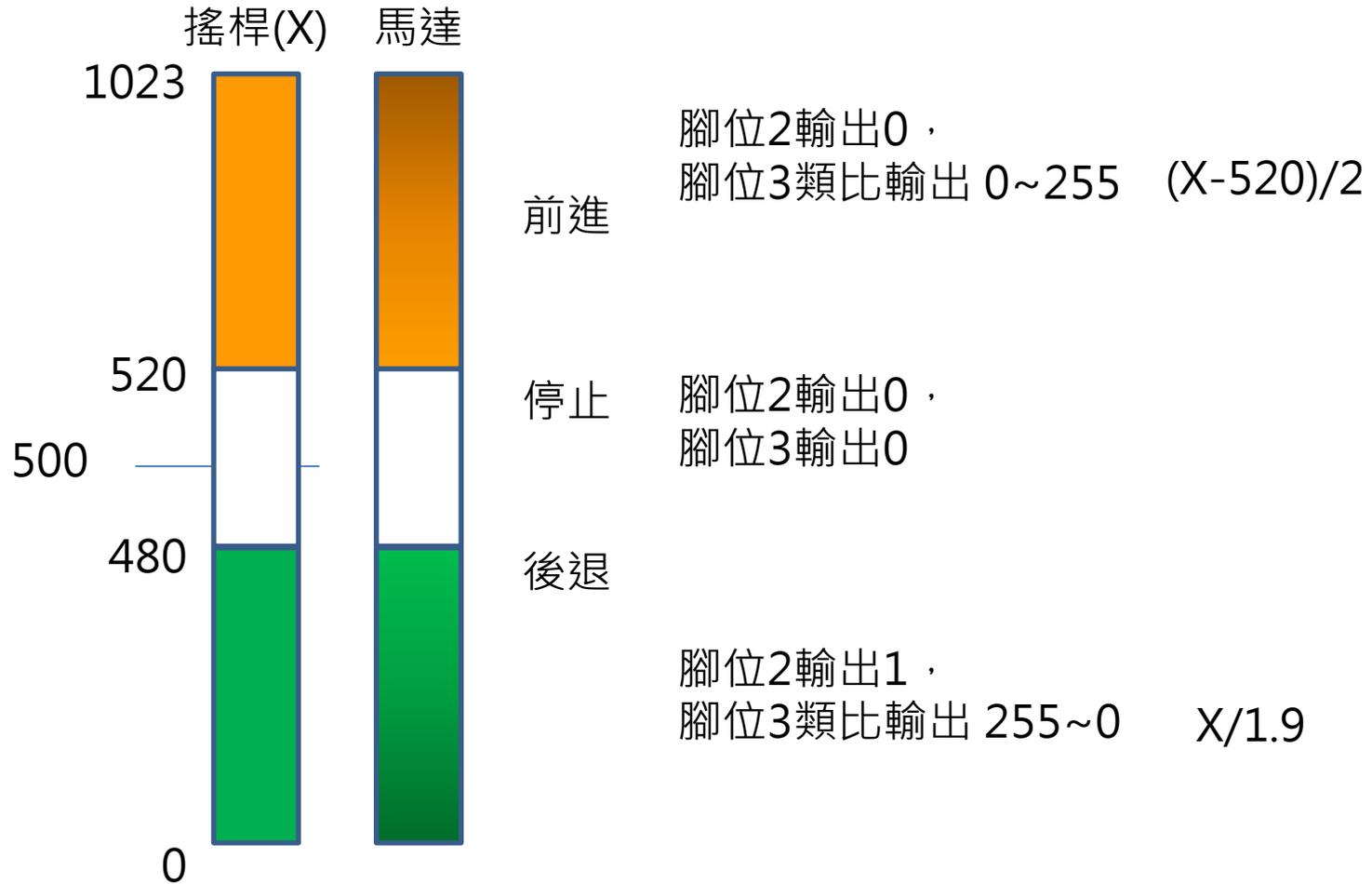


(9.1) 複合課程/搖桿控制直流馬達

- 搖動搖桿，控制馬達的轉動方向與轉速。



(9.2) 搖桿控制直流馬達

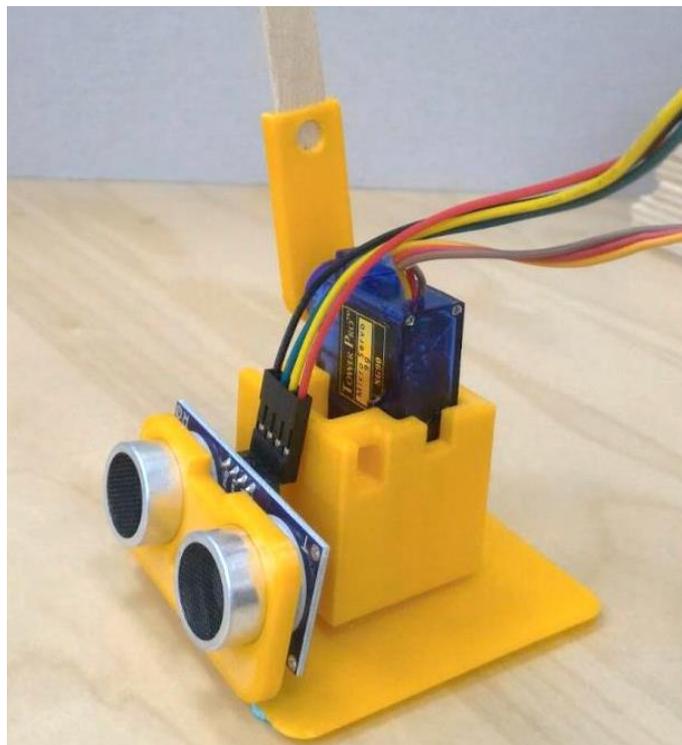


(9.3) 搖桿控制直流馬達

- 搖桿控制直流馬達.sb3



(10.1)複合課程/電動柵欄

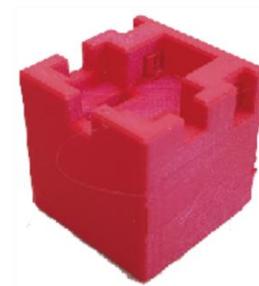
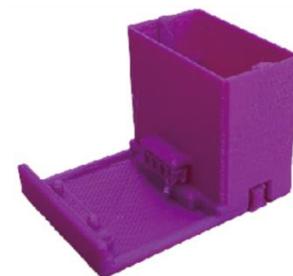


組裝前的準備(參考教材)

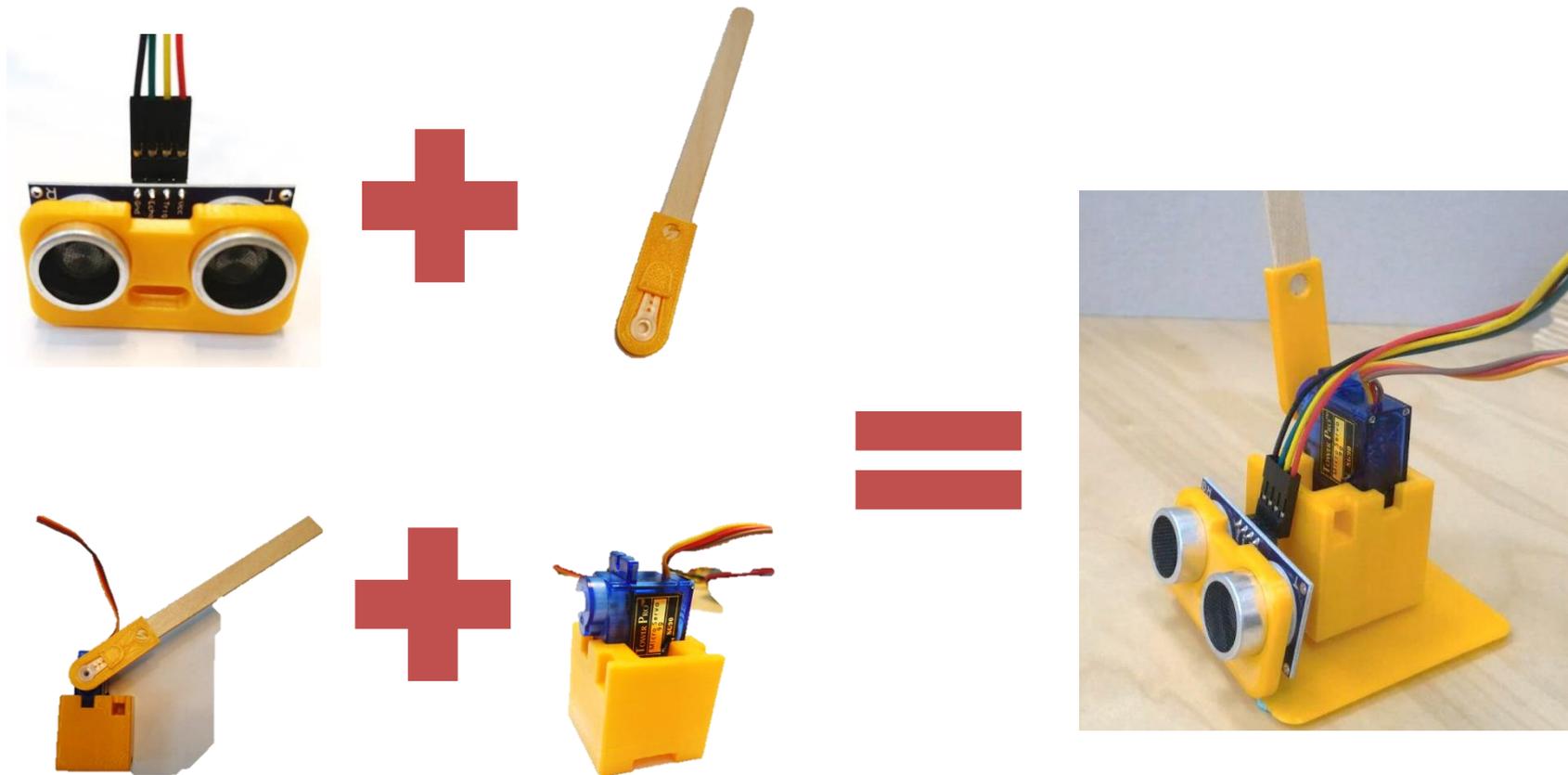
檢查零件是否齊全



(電控板底板, 與小車共用)



組裝電動柵欄(參考教材)



(10.2)複合課程/電動柵欄

- 使用超音波感測器偵測距離
- 當物體距離小10CM時，將柵欄舉起，等待1秒後，將柵欄放下。
- 電動柵欄.sb3

