

讀取類比腳位 A0 ▾ 註

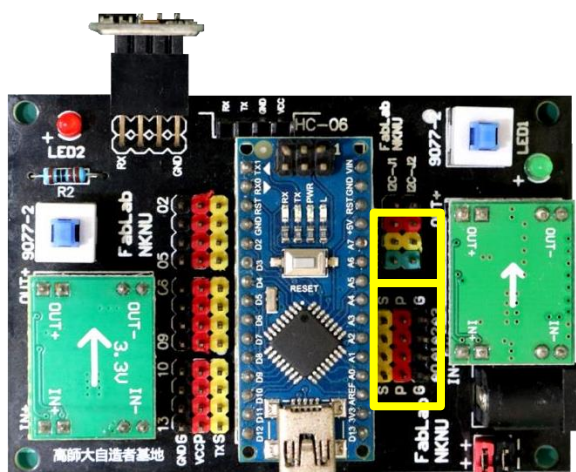
積木用途：讀取類比腳位(Analog pin, A0~A7)的傳回值，數值範圍 0~1023。

說明：

1. PWM 讓數位腳可以有 0~255 的輸出，但數位腳的輸入訊號仍然只有 0 與 1 兩種狀態，如果模組測量的對象是有強弱程度的，就要使用類比腳位。
2. Arduino Nano 有 8 個類比腳位，編號 A0~A7，也可以接續數位腳的號碼，數位腳編號是 0~13，類比腳編號就是 14~21。舊的 NKNU 電路板上的類比腳位區只有 A0~A3，新的電路板與 5012 教具平台則多出 A6 與 A7 腳位，A4 與 A5 做成 I<sup>2</sup>C 界面的排列方式以方便連接 I<sup>2</sup>C 界面模組，不過仍可做為類比輸入。
3. 當數位腳不夠用時，A0~A5 也可當作數位腳來用。
4. 想要把類比腳位當作數位腳來用，可是數位積木的腳位選單中沒有列出類比腳位，可使用底程式。



5. A6、A7 這兩支腳位只能做為類比輸入，無法當做數位腳位來用，第一批 NKNU 電路板也沒有引出這兩支腳的線路，在新的 NKNU 電路板上就可以看到這兩支腳位的插槽。



舊電路板



新電路板

程式範例一：使用搖桿來控制 LED 亮度。

這個範例使用「4060馬達與感測器教具平台」中的搖桿與 RGB LED。

程式中使用搖桿 Y 軸(前後搖，接到 A1)來控制綠燈(接到 D10)的亮度。不過類比輸入的範圍是 0~1023，PWM 的範圍只有 0~255，需要經過轉化才能將搖桿的數值拿來控制LED 的亮度。



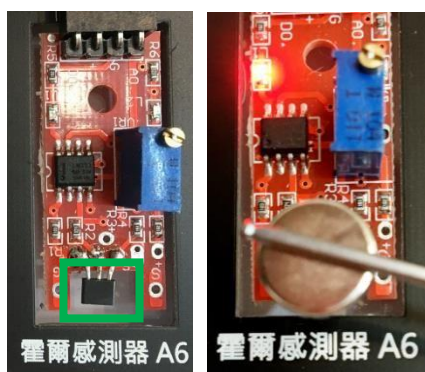
若是使用「5012智慧數控教育平台」教具務必搭配電源供應器，搖桿類比讀取最大數值才能達到1023。

程式範例二：5012 教具平台上的霍爾感測器實驗

說明：

1. 5012 教具平台的霍爾感測器接在 A6 腳位，可量測磁力大小，建議使用直徑 10mm 厚度 2mm 以上的釹鐵硼強力磁鐵來進行實驗。
2. 剛執行程式時，傳回值約在 500 左右，拿磁鐵靠近感測器(最好吸附在鐵器上比較好操作)，如果數值下降表示面向感測器的是磁鐵的 S 極，S 極的數值變化不明顯，請將磁鐵翻面，用 N 極靠近，傳回值會升高，靠得越近數值越高。

(注意：磁鐵會導電，別讓磁鐵碰觸感測器或電路板)。



程式範例三：使用霍爾感測器控制 8\*8LED 矩陣亮燈數。

說明：

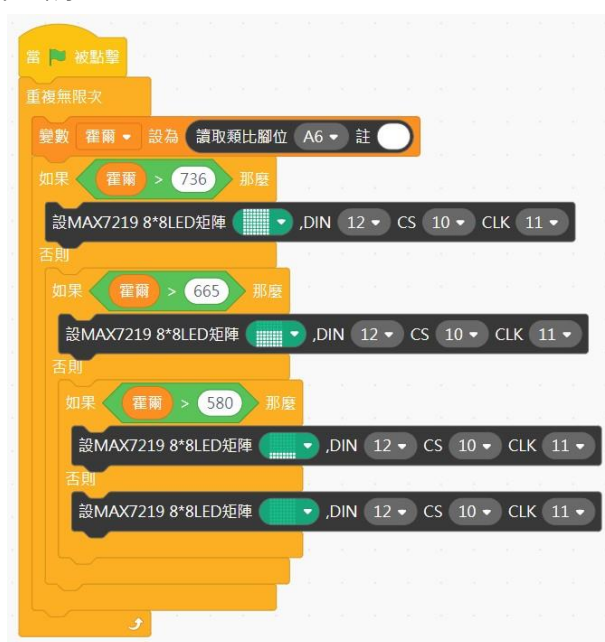
1. 將磁鐵 N 極朝下裝入滑塊中，使用上例實驗程式，將滑塊指針對準每個刻度，會發現傳回值不斷跳動，可使用底下找出在每個刻度的傳回值範圍並記錄下來。  
(每換一個刻度要重新執行)



2. 依據各刻度的範圍找出適當的分界值並設計對應的亮度。

	刻度一	刻度二	刻度三	刻度四
最大值	799	733	660	575
最小值	740	672	582	544
分界值	>736	>666	>578	
亮燈	全亮	亮 5 排	亮 2 排	不亮

3. 依照數據撰寫程式並調整。



程式範例四：使用霍爾感應器製作里程、速度計。

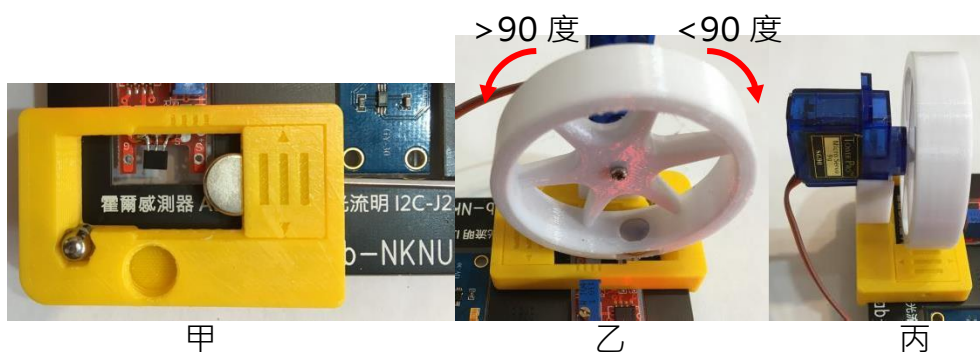
說明：

### 1. 改裝霍爾感測器配件：

甲、將滑塊固定在右方，使用大迴紋針將磁鐵戳出。

乙、將磁鐵裝在輪框上，N 極朝外，並用 SG90 附帶的螺絲將輪框鎖上伺服馬達，這裡使用的是可以連續轉動的 360 度 SG90，角度設為 90 度時馬達停止，角度 < 90 順時針轉，角度越小速度越快，角度最小為 0；角度 > 90 逆時針轉，角度越大速度越快，角度最大為 180。

丙、將伺服馬達裝在配件支柱上，並將線接到數位腳 D6，橘色線是訊號線，棕色線是 Gnd。



2. 里程 = 圈數 \* 輪框圓周長，輪框直徑為 5cm，圓周長 =  $5 \times 3.14 = 15.7\text{cm}$ 。

速率 = 周長 / 轉一圈的時間 (cm/sec)

計算里程與速率的關鍵都在「如何知道輪框轉了一圈」？當磁鐵靠近霍爾感測器時，A6 腳位的傳回值會升高，離開時傳回值會降低，我們可以訂出一個數值來做為開始算圈數的標記，例如 600。可是輪框裝上去之後磁鐵的位置要在哪裡？我們並不知道傳回值 = 600 的位置在哪裡啊！可以用底下的程式讓磁鐵轉到標記點。



馬達角度設為 70，先不要轉太快比較容易除錯。

測試看看轉到我們設定的標記區時會不會停下來。成功的話要把馬達停止的積木拿掉。

到達標記區後記錄「到達標記區」設為 1，將「跑一圈時間」與「圈數」歸零準備計算里程與速率，然後計時開始。



當感測器傳回值>600 時，我們記錄「到達標記區」=1，那麼離開標記區時(傳回值<601)就要記錄「到達標記區」=0。

那麼到達標記區後就可以把圈數加 1、算里程、速率.....了嗎？喔不！標記區(傳回值>600)的範圍有點大，不能讓磁鐵停留在標記區時就把圈數加好、加滿，然後自 hi 跑了好遠，跑得好快，只有在剛進入標記區那瞬間才可以增加圈數，而那時「到達標記區」=0，所以增加圈數的條件應該是：

當「霍爾」感測器傳回值>600 且「到達標記區」=0

而且進入標記區後要馬上把「到達標記區」設為 1，這麼一來，即使磁鐵在標記區內待到天荒地老，仍然只會記錄它跑了一圈，一定要跑出標記區再跑進來才會增加圈數。

The image shows a Scratch script for a robot's movement logic. The script starts with a 'Servo Motor' block set to position 6 and angle 70. It then sets a variable '霍爾' (Hall) to 0. A 'Repeat Until' loop is used to reach the marker area, with the condition '霍爾 > 600'. Inside this loop, the '霍爾' variable is set to the value of '讀取類比腳位 A6'. After the loop, the '到達標記區' (Reached Marker Area) variable is set to 1, '跑一圈時間' (Time to run one lap) is set to 0, and '圈數' (Lap count) is set to 0. A 'Timer Reset' block is also present. The main loop is a 'Repeat Forever' loop. Inside, the '霍爾' variable is set to '讀取類比腳位 A6'. An 'If' block checks if '霍爾 < 601', and if true, '到達標記區' is set to 0. Another 'If' block checks if '霍爾 > 600' and '到達標記區 = 0'. If true, it sets '跑一圈時間' to '計時器' (Timer), resets the timer, sets '到達標記區' to 1, increments '圈數' by 1, sets '里程' (Distance) to '四捨五入數值 (圈數 \* 15.7)', and sets '秒速' (Speed) to '四捨五入數值 (15.7 / 跑一圈時間)'. Three green callout boxes provide additional context: '已刪除馬達停止積木' (Motor stop block deleted) points to the area where a stop block was removed; '到達標記區的瞬間' (The moment of reaching the marker area) points to the condition '霍爾 > 600 且 到達標記區 = 0'; and '記錄已進入標記區，以免重複計算圈數' (Record that the marker area has been entered to avoid repeated lap calculations) points to the '到達標記區' variable being set to 1.

已刪除馬達停止積木

到達標記區的瞬間

記錄已進入標記區，以免重複計算圈數。

把訊息顯示在 OLED 上。



改變伺服馬達的角度，看看速率有什麼變化？