

STEM_4

自學教材

編著：劉偉成 (Lau Wai Shing)

樣本

專案一：組裝智能遙控爬行機械

目錄

1. 主要硬件清單	2
2. 智能遙控爬行機械	3
3. Arduino UNO R3 兼容主板	4
4. Arduino UNO R3 感應器擴展板 5.0	5
5. 伺服電動機	6
6. 超聲波感應器	8
7. 組裝輪腿	10
8. 組裝爬行機械主體	15
9. 修飾爬行機械	22
10. 作業 1	23

5. 伺服電動機

《電子零件與擴展板》基礎知識小冊中提及訊號線的作用是傳送 PWM 脈衝來控制輸出軸的停止位置旋轉角度。我們要注意訊號脈衝必須為每秒重複 50 次(即是 50Hz)，而脈衝寬度的範圍從 1.0ms 到 2.0ms(millisecond，毫秒，千分之一秒)。

脈衝持續時間的長度代表電動機輸出軸需要停在甚麼角度位置。我們可以理解 1.0ms 為輸出軸在 0 度位置停下，1.5ms 是在 90 度位置停下，而 2.0ms 則是在輸出軸轉動角度範圍的極限 180 度位置停下。因此，如果我們想電動機輸出軸在置中位置停下，脈衝寬度要設定為 1.5ms。

不過需要留意的是電動機有可能被反轉擺放，所以脈衝寬度所代表的輸出軸停轉角度會相反(即 1.0ms 表示輸出軸在 180 度位置停下；1.5ms 表示輸出軸在 90 度位置停下；2.0ms 則表示輸出軸在 0 度位置停下)。

了解訊號脈衝寬度與伺服電動機輸出軸轉動角度的關係後，我們可以利用 mBlock 編程，控制伺服電動機輸出軸轉動。例如我們選擇用鍵盤控制伺服電動機，需要在程式中設定各款按鈕分別控制輸出軸轉動的角度和速度。




6. 超聲波感應器

HC-SR04 超音波感應器腳位從左至右分別為 Vcc、Trig、Echo 與 GND。Vcc 腳位和 GND 腳位分別連接 5V 電源和 GND 線；Trig 腳位為輸入腳位，用來發射超音波訊號，Echo 腳位為輸出腳位，用來接收訊號。



如果你送出 10 微秒(μs)的 5V 高電位訊號給 Trig 腳位，就會觸發超音波的發射，同時，Echo 腳位會從 0V 低電位態轉為 5V 高電位狀態。如果接收到反射的超音波訊號，Echo 腳位就會回到 0V 低電位狀態。

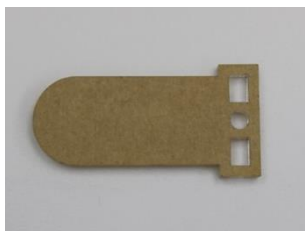
設計這個應用時，要留意超音波感應器的主要參數¹。有關超聲波感應器的介紹和如何連接超聲波感應器與感應器擴展板，可參閱基礎知識小冊子《電子零件與擴展板》。

接著，我們會用 mBlock 創建以下程式，測量出物件距離數值。上載程式步驟跟伺服電動機一樣，用滑鼠按下  按鈕，程式便會執行。當我們把物件與感應器的距離改變時，Panda 熊會讀出並不斷更新以厘米 (cm) 作單位的距離讀數。

¹感測距離：2cm - 450 cm；工作電壓/電流：5V DC / 30mA；有效感測角度：少於 15 度；被偵測物件的面積不要少於 0.5 m²，並且盡量平整。

7. 組裝輪腿

輪腿是爬行機械的主要部份，用來移動機械。我們利用隨套件工具箱附送、已預製的亞加力膠輪腿裝嵌組件(A、B 兩款，見下圖)，進行輪腿的組裝。



A 款輪腿組件



B 款輪腿組件

首先，我們把所有貼在亞加力膠片上的保護膠貼除去。



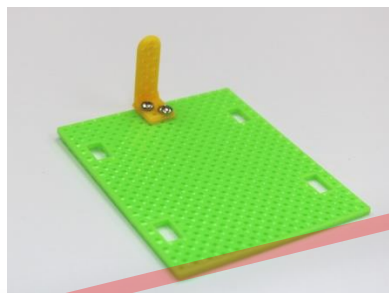
把 M3 螺母插入 B 款組件其中一端的螺母座內，並用 M3 螺絲把 B 款組件與 A 款組件連接並鎖緊如下圖。可一面用手鉗固定螺母，一面用螺絲批把螺絲扭進去。



8. 組裝爬行機械主體

組裝輪腿後，我們要把輪腿和其他電子零件裝嵌成爬行機械主體。

我們先用兩顆 6mm 長的 M3 尖頭螺絲把多孔角碼裝嵌到多孔固定萬用膠板上。



然後，把電池盒的底部固定在萬用膠板上，再用 M3 尖頭螺絲把它鎖緊。



之後，在超聲波感應器的背面塗上少量熱熔膠，並把它貼在多孔角碼上。



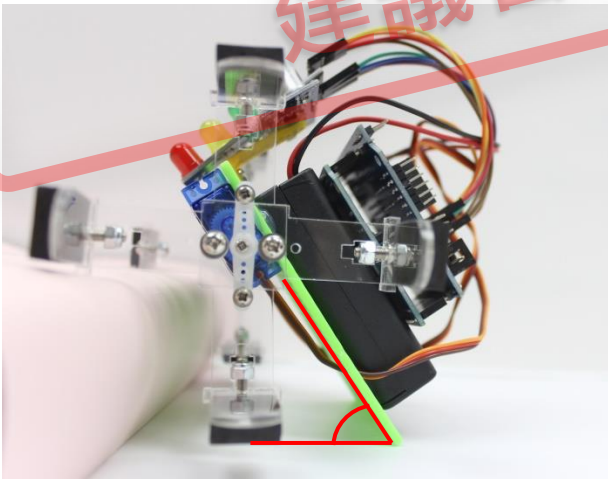
10. 作業 1

1. 按照本專案和相關基礎知識小冊的資料，裝嵌一隻能爬行、跨越障礙，甚至上樓梯的智能爬行機械。
2. 機械尾部有甚麼作用？它是否可有可無？試解釋原因。

專案 1 作業 1 建議答案

問題 2：

當機械上樓梯或進行攀爬時，身體便向上傾斜並與地面形成角度。如果它沒有尾巴，攀爬時它與地面的角度會不斷增加。當角度越近 90 度，機械就越難向上攀爬。



STEM_4

自學教材

編著：劉偉成 (Lau Wai Shing)

樣本

專案二：透過 **MBLOCK** 編寫 爬行機械避障程式

目錄

1. 控制爬行機械繞過障礙物	2
2. 作業 1	11
3. 爬行機械遇障時的隨機反應	12
4. 作業 2	16

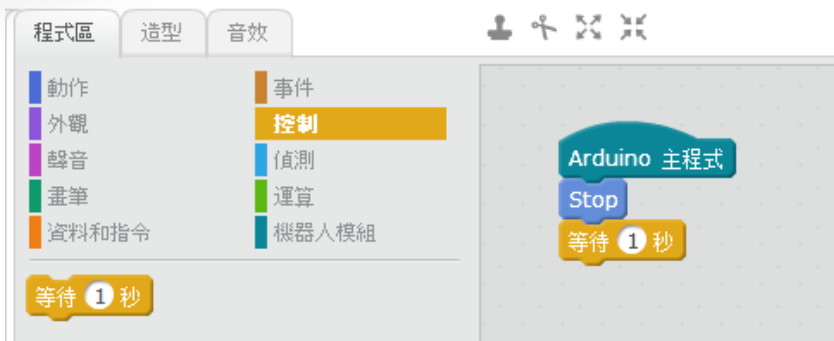
1. 控制爬行機械繞過障礙物

我們在《透過 MBLOCK 控制爬行機械》基礎知識小冊中學了利用 mBlock 編程控制爬行機械向前、向後、向左、向右和停止移動，以及控制它閃燈、「說話」和「跳舞」。現在我們會嘗試編寫程式，令爬行機械遇到障礙物時避開。

在機械人模組類別，把「Arduino 主程式」積木拖到程式區。



把已設定的「Stop」積木加在「Arduino 主程式」積木下，再從控制類別把「等待(1)秒」積木加到「Stop」下，使爬行機械在剛通電後先靜止一秒，然後再執行下一步指令。



我們要設定超聲波感應器感應距離的變數。在資料和指令類別中按下做一個變數，並在出現對話框中輸入變數名稱「distance」後按確認，就建立了一個代表距離的變數積木。



把「將變數(distance)的值設為(0)」積木加在程式區中「等待(1)秒」積木的下方。



2. 作業 1

為智能機械編程，使它每次距離障礙物少於 35cm 時就停下來並閃燈，向後退，然後向轉左。當它避過障礙物後，繼續向前移動。

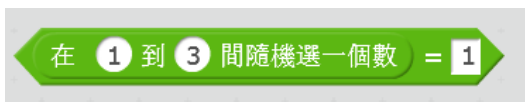
3. 爬行機械遇障時的隨機反應

在上一部份，我們學會編程使機械自動避過障礙物的程式。如果大家覺得這樣編程太單調乏味，我們可以嘗試編寫多個機械遇障時的反應指令，讓它隨機選擇，執行不同動作。

首先在程式區中加一個「如果...就...否則」積木，並在運算類別取出一個「 $() = ()$ 」積木和一個「在(1)到(10)間隨機選一個數」積木。

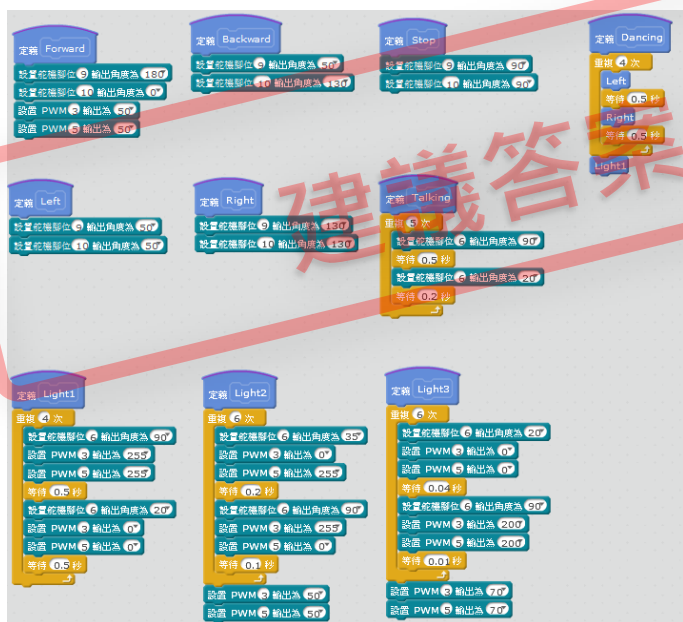


把隨機數積木的範圍參數設為「1」和「3」，然後把這個積木放入「() = ()」左方的空格內，右方的空格則輸入「1」，整組積木的用意是：在三個代表不同遇障反應的隨機數字中，「1」被選中了。



問題 1：

首先，我們要編寫各款定義方塊，編程如下：



然後，我們要建立變數 distance 並把定義方塊放在 Arduino 主程式 (Arduino Program) 方塊之下。要符合題目要求機械人與障礙物距離 35cm 時作出停止，我們必需編寫「如果距離少於 35 就停止移動 (或加入其他反應)，否則前進」的程式並把此程式方塊放在不停重複方塊中。

STEM_4

Self-learning Guide

Author: Lau Wai Shing (劉偉成)



Sample

PROJECT 3 REMOTE CONTROL OF CLIMBOT BY BLUETOOTH

Contents

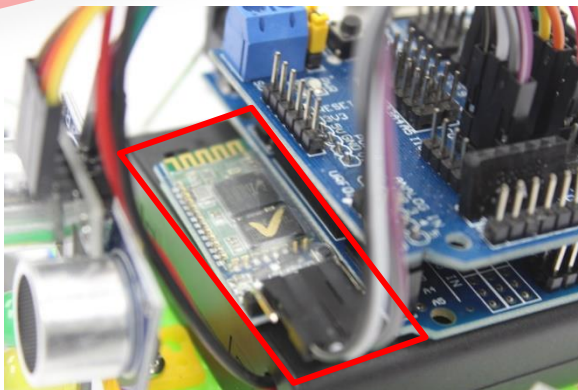
1. Hardware List	2
2. HC-05 Bluetooth Module	2
3. Connecting Bluetooth Module to Shield.....	4
4. Remote Control mBlock Programming	5
5. Assignment 1.....	14
6. Remote Control with an App.....	15
7. Assignment 2.....	19

3. Connecting Bluetooth Module to Shield

In order for us to control Climbot remotely through Bluetooth, we must connect the HC-05 Bluetooth Module to the shield with 4 jumper wires. Connect the VCC pin of the module to the 5V pin of the shield, the GND pin of the module to the GND pin of the shield, the TXD pin of the module to the RXD pin of the shield, and lastly, the RXD pin of the module to the RX pin of the shield.



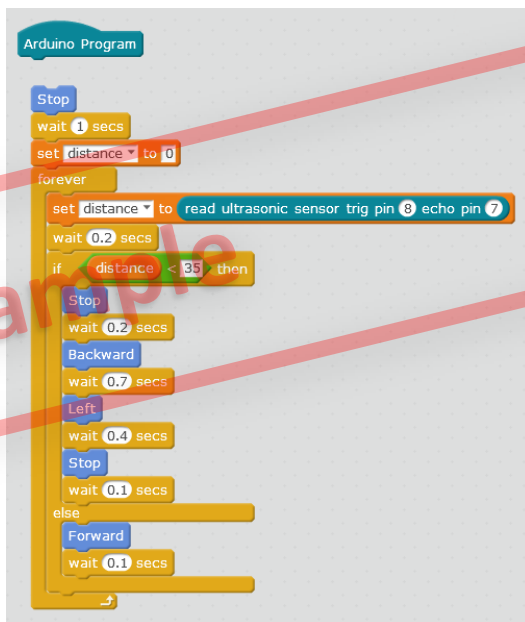
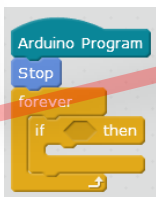
When the connections are made, we attach the Bluetooth module to the battery holder with Blu-Tack.



4. Remote Control mBlock Programming

In the previous project, we have coded the response of Climbot to obstacles. Now we will modify the main program in order to control Climbot remotely with Bluetooth.

First, remove the “Arduino Program” from the set of blocks below it temporarily. We will take care of the latter later. Next, from **Data&Blocks** category, add the “Stop” block under the “Arduino Program” block. Then **Control** category, add the “forever” and “if ... then” blocks under “Stop” as shown below:

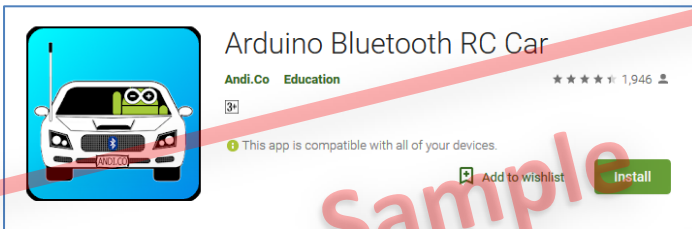


Next, from **Operators** category, place the “() > ()” block into the condition hole of the “if ... then” block. From **Robots** category, place the “serial available bytes” block in the left box of the “() > ()” block and enter “0” in the right box.

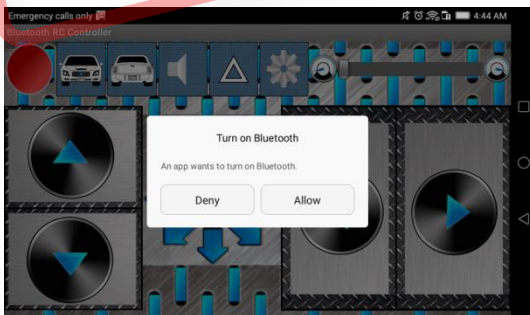
6. Remote Control with an App

Having coded the Climbot and its remote control, now we are going to download an app onto a mobile device. After installing it and selecting the remote control mode, we can control Climbot's motion so that it could avoid obstacles and even go up or down stairs.

Download the app “Arduino Bluetooth RC Car” to an Android tablet computer or smartphone and install it.



As the app is run, a dialog box for enabling Bluetooth will open. (For a first-time execution of the app, choose “Deny”.)



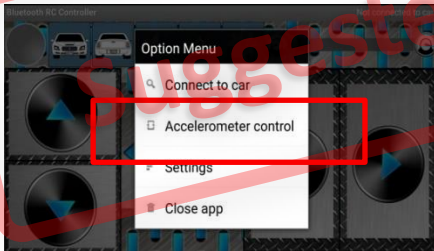
Afterward, the main interface of the app will show up with a host of control buttons, such as Upward, Downward, Leftward, Rightward, Front Lights On/Off, Back Lights On/Off.

7. Assignment 2

1. How should the app be set up to control the flashing, turning on and off the light, and dancing of Climbot?
2. Can we enable the auto mode and remote control mode of Climbot at the same time? If so, what modifications to the program should be made?
3. Besides buttons, are there other means by which to control Climbot?

Question 1:

Besides of accelerometer control mode, we can set the app into another mode to control the Climbot. We need to click Option Menu icon and Accelerometer control icon.



Then, the setting of the app changes into Gesture mode. When we rotate our mobile, the Climbot moves correspondingly.

