目錄
摘要 ........................................................................................................................................... 1
第一章 緒論 ............................................................................................................................... 2
  1.1 研究目的 ..................................................................................................................... 2
  1.2 準備工作 ................................................................................................................... 2
  1.3 研究方法 ................................................................................................................... 3
第二章 硬體架構 .................................................................................................................... 4
  2.1 Arduino 控制板 ......................................................................................................... 4
  2.2 馬達驅動板 ............................................................................................................... 5
  2.3 藍芽模組 ................................................................................................................... 6
  2.4 紅外線循線陣列感測器 ............................................................................................ 8
  2.5 車子主體 .................................................................................................................. 10
第三章 系統設計 .................................................................................................................. 11
  3.1 系統開發環境 .......................................................................................................... 11
  3.2 藍芽無線傳輸 .......................................................................................................... 14
  3.3 硬體架構 .................................................................................................................. 15
  3.4 自走車硬體架構 ...................................................................................................... 16
  3.5 軟體程式架構 .......................................................................................................... 17
  3.6 Arduino 程式碼 ......................................................................................................... 18
  3.6 Android 程式碼 ......................................................................................................... 21
第四章 實驗過程 ................................................................................................................... 22
  4.1 甘特圖 ....................................................................................................................... 22
  4.2 手機操作介面 .......................................................................................................... 23
  4.3 程式設計平台 .......................................................................................................... 26
第五章 結論 .......................................................................................................................... 27
  5.1 結論 ........................................................................................................................... 27
  5.2 未來展望 ................................................................................................................... 28
參考文獻 .................................................................................................................................. 29
圖目錄

圖 2.1 Arduino 控制板 ................................................. 4
圖 2.2 馬達驅動板 L298P ........................................... 5
圖 2.3.1 藍芽模組外觀 .............................................. 7
圖 2.3.2 藍芽模組功能及腳位示意圖 .......................... 7
圖 2.4 Infrared tracksensor IR-08H 紅外線避障感測器外觀 .. 9
圖 2.5 車子主體外觀 ................................................. 10
圖 3.1 Arduino 開發軟體示意圖 .................................. 13
圖 3.2 藍芽傳輸指令流程圖 ...................................... 14
圖 3.3 硬體電路圖 .................................................. 15
圖 3.4 自走車外觀硬體架構圖 ................................. 16
圖 3.5 軟體程式架構圖 .............................................. 17
圖 4.1 計畫甘特圖 .................................................. 22
圖 4.2.1 手機操作主畫面 ........................................ 23
圖 4.2.2 手機操作畫面(藍芽選單) ........................... 24
圖 4.2.3 手機操作主畫面(手動模式) .......................... 24
圖 4.2.4 手機操作主畫面(自走模式) .......................... 25
圖 4.3 App inventor 開發介面 .................................. 26
摘要

本次專題我們製作 Arduino 遙控履帶自走車，硬體方面，核心控制端選擇了 Arduino UNO 控制板為基礎，搭配 L298P 馬達驅動版驅動兩顆 DC 馬達，避障方面使用 紅外壁障傳感器（InfraRed track sensor）IR-08h，手機與 Arduino 溝通是使用藍芽的方式，將整體製作成類似戰車的外觀，使用履帶式考量到他能比傳統輪子更能適合不同的地形；搭配自主開發的手機 APP，控制自走車的行動。

關鍵詞：Arduino、藍牙、自走車、行動裝置
第一章 緒論

1.1 研究目的

使用者可以藉由智慧型裝置經藍芽模組與智慧避障遙控坦克車連接後，自由的無線操控車體，包括前進、轉彎、後退等相關功能，且因為車體為履帶車底盤套件組，比一般輪型車更能適應較複雜的地形。

由於現今高科技的社會中，無人載具及遠端操控將是現在高度使用的科技，像是無人飛機的偵查行動、深海探測及天氣預測等等，都是現今社會上常使用的技術，避障自走可以與固定地區配送信件或物品做為結合，節省人力，以這些相關技術，做出簡易的車體，從中激發自我的想法與創意，應用於不同的層面，就是本次研究所要達到的目的。

1.2 準備工作

Arduino的開發環境可以使用Arduino語言，app的部分是用由Google開發的App Inventor（直譯為「App發明者」）免費工具。結合電子元件，馬達驅動版L298P，紅外線避障感應器IR-08H，藍芽模組HC-05，做出互動作品，而達到具有能自走避障、手機遙控等多項功能，更深入研究更可以增添許多功能，最後能達到實際運用在現實生活上。

撰寫Application軟體方面使用下列工具：

1. App Inventor 2.0

2. Arduino

智慧避障方面，可以應用於閃避障礙物方面，軟體的主要目的是在任何時候保持對硬體的控制，及決定如何移動繞過障礙物。軟體控制硬體的
範圍包括理解感測器偵測得到的資訊，設定馬達的轉向，以及與外在周邊設備的通訊。

1.3 研究方法

一、本研究主要包括 Arduino 控制板、馬達驅動版 L298P，紅外線避障傳感器，藍芽模組 HC-05。

二、智慧型行動裝置上 Application 發出的控制訊號使用藍芽模組來進行傳輸。

三、手動遙控方式：利用手機畫面上的方向鍵下達指令，使遙控車朝指定方向前進。

四、避障模式：
經由紅外線避障傳感器感測與前方障礙物距離，當距離達到設定數值後即改變方向，將讀入的訊息傳入 Arduino 控制板解讀過後，再將訊號傳至馬達驅動版 L293P 使之移動。

五、夜行模式：運用 LED 模組，在手動遙控時可使車子仍可以在光線不足的隧道中正常移動。
第二章 硬體架構

2.1 Arduino 控制板

硬體部分是一個約手掌大小的控制板(寬 70mm X 高 54mm)，核心使用八位元 ATMEGA8 系列的微控制器，提供 14 個數位輸出/入端，6 個類比式輸出/入端，支援 USB 資料傳輸，可以使用自備電源(5V~9V)或是直接使用 USB 電源，如圖 2.1 顯示，使用者可以在數位式輸出/入端上接上不同的電子裝置，例如 LED 燈、喇叭、馬達，然後再由控制器來驅動燈的亮滅、喇叭發聲、馬達運轉。Arduino 控制板採用開放式源碼設計的概念，電路設計圖、韌體都可以在網絡上下載，稍具電子知識就可以自行製作。
2.2 馬達驅動板

馬達驅動板L298P外觀，如圖2.2，具有驅動能力強，發熱量低，抗干擾能力強的特點，電機輸出端採用8指高速肖特基二極管做為保護，該電路線路佈線合理均採用貼元件片，疊成設計可以直接插到Arduino上。

(1). 主控晶片：ST L298P
(2). 邏輯電壓：5V
(3). 驅動電壓：6.5V - 12V
(4). 邏輯電流：0mA - 36mA
(5). 驅動電流：2A (Max. 單橋)
(6). 存儲溫度：-20℃ 到 +130℃
(7). 最大功率：25W
(8). 尺寸：68mm × 53mm
(9). 驅動：雙鹿大功率H橋驅動

圖2.2 馬達驅動板L298P
### 2.3 藍芽模組

藍芽模組可以讓你在 10 米範圍內實現無線傳輸通信。該模組無需瞭解複雜的藍芽底層協定，只要簡單的幾個步驟即可享受到無線通訊的便捷。

#### 描述

1. 採用 CSR 主流藍牙晶片，藍牙 V2.0 協定標準
2. 串口模組工作電壓 3.3V
3. 串列傳輸速率默認出廠 9600，用戶可設置
4. 核心模組尺寸大小為：28mm x 15 mm x 2.35mm。
5. 工作電流：
   - i. 配對中：30～40mA
   - ii. 配對完畢未通信：2～8mA
   - iii. 通訊中：8mA
6. 休眠電流：不休眠
7. 用於 GPS 導航系統，水電煤氣抄表系統，工業現場採控系統
8. 可以與藍牙筆記本電腦，電腦加藍牙適配器，PDA 等設備進行無線連接。

#### 特性

1. LED 引腳指示藍牙連接狀態，輸出脈衝表示沒有藍牙連接，輸出高表示藍牙已連接並打開了埠
2. 電源電壓 3.3V，未配對時電流約 30mA，配對後約 8mA，絕不接反電源！電源絕對禁止超過 4.2V！
3. 介面電位 3.3V，可以直接連接各種單晶片（51, AVR, PIC, ARM, MSP430 等），5V 單晶片也可直接連接，無需 MAX232 也不能經過 MAX232！
(4). 空曠地有效距離 10 米，超過 10 米也是可能的，但不對此距離的連接品質做保證

(5). 配對以後當全雙工串口使用，無需瞭解任何藍牙協定，但僅支援 8 位元資料位元、1 位元停止位、無奇偶校驗的通信格式，這也是最常用的通信格式，不支持其他格式

(6). 在未建立藍牙連接時支援通過 AT Command 設置串列傳輸速率、名稱、配對密碼，設置的參數，可保存參數。藍牙連接以後自動切換到傳輸模式

(7). 體積小巧 (2.7cm*1.3cm)，工廠貼片生產，保證貼片品質

(8). 主機或從機，可通過 AT Command 做設定，從機之間不能配對。
2.4 紅外線避障感測器

Infrared tracksensor IR-08H，紅外線避障感測器，感應距離 2 到 30
釐米距離遠抗干擾能力好，速度快，適合智能小車避障，頻率調節電位器
是用來調節紅外線射管的載波頻率，因為一體化接收頭要在特定的 38khz
載波頻率才是最靈敏的。

特點：
(1). 可直接與 3v 和 5v 單片機 IO 相連
(2). 本傳感器 3 至 6v 供電，範圍廣
(3). 帶使能端，EN 端子等於 1 時不工作，等於 0 時工作
應用:

(1). 避障
(2). 迷宮導航
(3). 機器人比賽
(4). 障礙距離監測
(5). 邊緣檢測

圖 2.4 Infrared tracksensor IR-08II 紅外線避障感測器外觀
2.5 車子主體

履帶車底盤套件組，是採用履帶進行行走，就像給車體鋪了一道無限延長的軌道一樣，使它能夠平穩、迅速、安全地通過各種複雜路況，由於接地面積大，所以增大了坦克在鬆軟、泥濘路面上的通過能力，降低了下陷量。在較崎嶇的路面上能牢牢地抓住地面，不會滑轉。由於履帶接地長度較長，誘導輪中心位置較，所以通過上下坡的能力較強。履帶還有一個特殊功能，如果車體增加防水功能，採取潛下水面的方式，可以在河底行走；而若是像船一樣浮於水面，履帶可以像螺旋槳一樣產生推進力，驅使車輛前進。

履帶車重量大，用履帶的話可以減小單位面積壓力，對路面的要求可以小點，且越野性能比輪式要好，輪式結構比履帶簡單，一般都用於輕型車輛，速度比履帶車輛好，維護方面也比履帶簡單得多。
第三章 系統設計

3.1 系統開發環境

Microsoft Windows 7

Windows 7 是微軟公司推出的電腦作業系統，供個人、家庭及商業使用，一般安裝於筆記型電腦、平板電腦、多媒體中心等，此次使用於程式撰寫方面電腦之作業系統。

Android Inventor

最初由 Google 開發的 App Inventor（直譯為「App 發明者」）免費工具。

Android 4.4.2

是一個以 Linux 為基礎的開放原始碼行動裝置作業系統，主要用於智慧型手機和平板電腦，Android 作業系統的核心屬於 Linux 核心的一個分支，具有典型的 Linux 排程和功能，除此之外，Google為了能讓 Linux 在行動裝置上良好的運行，對其進行了修改和擴充。

2013 年 9 月 3 日，Google 公布 Android 4.3 的後續版本為 4.4，代號為奇巧巧克力（KitKat）。

2013 年 10 月 31 日，Google 正式發表 Android4.4（KitKat）版本，以及 Nexus 5。主要更新如下：

(1) 支援語音開啓 Google Now
(2) 支援全螢幕模式 Immersive Mode
(3) 優化記憶體使用
(4) 新的電話通訊功能
(5) 低電耗音樂播放
(6) 新的 NFC 付費整合
(7) 增加 ART (AndroidRuntime) 模式
(8) 移除 4.3 中的 app ops 功能

Arduino 1.5.8

Arduino 的開發環境是基於 Java 與 Processing 程式語言為基礎，以 AVR-GCC 編譯器與 avr-libc、AVRlib 函式庫所構成的開放原始碼開發環境，它讓使用者可以很輕易的上手編寫程式並將程式上傳到 I/O 板上，它可以運行於市面上各種作業軟體如：Windows，Mac OS X，及 Linux 等...

特色
(1). 基於創用 CC 開放原始碼的電路圖設計。
(2). 免費下載，也可依需求自己修改，但需遵照姓名標示。您必須按照作者或授權人所指定的方式，表彰其姓名。
(3). 依相同方式分享，若您改變或轉變著作，當散布該衍生著作時，您需採用與本著作相同或類似的授權條款。
(4). Arduino 可使用 ICSP 線上燒入器，將 Bootloader 燒入新的 IC 晶片。
(5). 可依據 Arduino 官方網站，取得硬體的設計檔，加以調整電路板及元件，以符合自己實際設計的需求。
(6). 可簡單地與感測器，各式各樣的電子元件連接，如 紅外線、超
音波、熱敏電阻、光敏電阻、伺服馬達…等。

(7). 支援多樣的互動程式，如 AdobeFlash，ax/MSP，VVV，PureData，C，Processing…等。

(8). 使用低價格的微處理控制器(Atmel AVR) (ATMEGA 8，168，328等)。

(9). USB介面，不需外接電源。另外有提供直流(DC)電源輸入。

圖 3.1 Arduino 開發軟體示意圖
3.2 藍芽無線傳輸

將電源開啟後，藍芽模組燈號便開始閃爍。

將智慧型行動装置開啟 Application，進入畫面後將藍芽開啟，便開始偵測，
確定連接上後，藍芽模組燈號便停止閃爍方式。

智慧型行動装置 Application 上主畫面共有 2 種模式及方向鍵，按下後，
指 令經由藍芽模組進行判斷，確認後便立即執行指令。

![流程圖](圖 3.2 藍芽傳輸指令流程圖)
3.3 硬體架構

以下為硬體電路圖：

圖 3.3 硬體電路圖
3.4 自走車硬體架構

![自走車外觀硬體架構圖](image)

圖 3.4 自走車外觀硬體架構圖
3.5 軟體程式架構

圖 3.5 軟體程式架構圖
3.6 Arduino 程式碼

```cpp
char temp[1];
int buttonpin=5;  // 定義避障感測器介面
int button=6;
int val=0, oldStat=0;
void setup() {
  for(int i=10; i<=13; i++)
    pinMode(i, OUTPUT);
  digitalWrite(i, HIGH);
}
digitalWrite(11, LOW); // left
digitalWrite(10, LOW); // right
Serial.begin(9600);
}

void loop() {
  performCommand();
}

void performCommand()
{
  if(Serial.available())  // 接收串列口資料
    {
    temp[0]=Serial.read();
  }
  if(temp[0] == 'f')
    {go_forward(); // 前進}
  else if(temp[0] == 'r')
    {go_right_F(); // 右轉}
  else if(temp[0] == 'l')
    {go_left_F(); // 左轉}
  else if (temp[0] == 'z')
    {stop_go(); // 停止}
  else if(temp[0] == 'b')
    {
```
go_reverse(); // 倒退
}
else if (temp[0] == 't')
{
    go_turbo(); // 開燈
}
else if (temp[0] == 'x')
{
    stop_go_turbo(); // 關燈
}
if(temp[0] == 'a') // 自動
{
    automatic();
}

void automatic()
{
    val=digitalRead(buttonpin);
    oldStat=digitalRead(button);
    if(val == LOW && oldStat==HIGH) // 右邊偵測到
    {
        digitalWrite(12,HIGH);
        digitalWrite(13,LOW);
        digitalWrite(11,LOW);
        digitalWrite(10,HIGH);
    }
    if(oldStat == LOW && val == HIGH)
    {
        digitalWrite(12,LOW);
        digitalWrite(13,HIGH);
        digitalWrite(11,HIGH);
        digitalWrite(10,LOW);
    }
    if(oldStat == HIGH && val == HIGH)
    {
        digitalWrite(12,HIGH);
        digitalWrite(13,HIGH);
        digitalWrite(11,HIGH);
        digitalWrite(10,HIGH);
    }
    if(oldStat == LOW && val == LOW)
    {
        digitalWrite(12,HIGH);
        digitalWrite(13,HIGH);
        digitalWrite(11,LOW);
        digitalWrite(10,LOW);
    }
void go_forward()  // 前進動作
{
digitalWrite(12,HIGH);
digitalWrite(13,HIGH);
digitalWrite(10,HIGH);
digitalWrite(11,HIGH);
}
void go_reverse()  // 倒退動作
{
digitalWrite(12,LOW);
digitalWrite(13,LOW);
digitalWrite(10,HIGH);
digitalWrite(11,HIGH);
}
void go_right_F()  // 右轉
{
digitalWrite(12,LOW);
digitalWrite(13,HIGH);
digitalWrite(11,HIGH);
digitalWrite(10,LOW);
}
void go_left_F()  // 左轉
{
digitalWrite(12,HIGH);
digitalWrite(13,LOW);
digitalWrite(11,LOW);
digitalWrite(10,HIGH);
}
void stop_go()  // 停止
{
digitalWrite(10,LOW);
digitalWrite(11,LOW);
}
void go_turbo()  // 開燈
{
digitalWrite(3, HIGH);
}
void stop_go_turbo()  // 關燈
{
digitalWrite(3, LOW);
}
3.6 Android 程式
第四章 實驗過程

4.1 甘特圖

從一開始大二下學期確定指導老師之後，便開始與老師有密切的聯繫，除了經過多次討論之後，以及了解老師上一屆所帶學長的專題，從眾多題目之中確定了專題走向，便開始分片協調工作，從外觀硬體到軟體撰寫以及書面文書相關的資料，接著就是收集與我們題目相關的文　資料，從中獲取靈感和想法，老師也提醒我們，要去選修與專題相關的課程，讓自己更加了解相關的技術。

學長專題成果發表時，我們小組進行了討論詢問內容後，前往觀看學長的成果，並且針對不了解的地方加以詢問，最後確定了相關的外型設計以及硬體設備，便開始著手採買的過程，過程中也經過多次修改以及外型上與功能的配合，軟體上從一開始簡易的程式也慢慢地經過多次的修改，尤其是紅外線程式與邏輯的配合，更是經過多次的測試，當車體做到一定程度時，便開始著手在相關文獻資料的處理，成果報告書、技術報告和報告用的簡報等等，經過詳細的分工合作達到最高的效率，最後也在專題展當天有了成功的展出。

### 圖 4.1 計畫甘特圖

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>確定指導老師</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>確定專題走向</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>分配協調工作</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>購買硬體設備</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>軟體撰寫測試</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>收集相關文獻</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>報告報告製作</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.2 手機操作介面

手機介面上第一步，點取我們預設好的 APP 軟體，會出現圖示的介面。

圖 4.2.1 手機操作主畫面
一開始選擇要連線的藍芽裝置。

點選藍芽並且完成連線後，手動模式按鈕就會呈現被按下狀態，沒有開啟藍芽是不能連線至車體。
接著按下自走模式，自走模式按鈕呈現被按下狀態

![手機操作主畫面](image)

圖 4.2.4 手機操作主畫面(自走模式)

連線成功後進入控制面板，就會進入預設模式，也就是手動模式，就可以點取前後左右控制車子，還附加大燈按鈕，在光線不足的地方可以開啟。自走模式啟動後，車子會自行移動，透過裝在車子前端的紅外線感測器搭配程式設定，能使車子自動避開障礙物完成自走任務。
4.3 程式設計平台

圖 4.3 App inventor 開發介面
第五章 結論

5.1 結論

本研究目標是開發出一台擁有兩種功能的智能車，為此我們針對這兩項功能進行深入的探討，第一種功能為智能避障的方式，第二種功能則是利用智慧型裝置上自行撰寫的 Application，經由藍芽控制自走車，達到能夠前進後退轉彎的功能。車體本身結合這兩種功能，進行多次的測試與實驗，進而達到本次研究的目標。

從一開始簡易的將車體架構起來後，將硬體結合，寫出簡易的程式燒入 Arduino 控制板後，測試所需的車體架構以及互相搭配配合，從中激發出與過去不一樣的想法，也決定將以以往不同的履帶車底盤套件組作為車體行進輪，考量到白天與夜間光線問題，因此在前端加上 LED 驅動模組，讓智慧車能夠在光線不足的地方還能有良好的操作視線。在一開始寫入 Arduino 控制板上連結藍芽模組的程式連不上，經過相當多的實驗，從網路上尋找，最後終於解決了此項問題。

在經過將近兩個學期的研究及撰寫之後，我們完成了屬於自己的畢業專題，並且將大學四年內所學到的理論實際呈現出來，當我們在實際撰寫程式時才發現理想狀況與現實狀況的落差，也因為如此才讓我們深刻了解到學以致用之重要性；資訊產業是一門日新月異的行業，除了學校教導我們的基本知識外，我們更需要良好的學習態度及熱誠，才可以在就業後達到優秀的表現。雖然我們現在開發的功能沒有很多，但期待在未來能夠更佳的精進，嘗試開發其他有趣的功能，也很感謝這次的團隊，大家一起合作、分配、討論
等在這過程中的是難能可貴的，而這次的專題也讓我們獲益良多，在未來研究的路上，這次將是一次相當重要的經驗。

5.2 未來展望

加裝攝影機，用視訊的方式來觀察自走車行走以及遇到的狀況。 增加 GPS 功能，可以定位車子位置並做導航。 更新更先進的遠距離無線遙控系統，舉例：Wi-Fi、433MHZ 無線傳輸模組。

增加定位巡邏功能，可用於區域巡邏。 加強感測系統，使自走車的避障更精準。 舉例：雷射感測。 增設對環境的感測裝置，如溫濕度計。

使用 RFID 製作相關延伸功能。
參考文獻

Arduino 介面控制板簡介

DIY - 智能小車：L298N 電機驅動板模組
http://bugworkshop.blogspot.tw/2013/02/diy-l298n.html

KSRobotKSM008 藍芽模組含底板(HC-05 版本)
httpes=120&pageName=/Arduino%A4%FC%EA%BD

Arduino 基本百科，http://zh.wikipedia.org/wiki/Arduino

吳志文(2013)，適合初學者的結合行動裝置與單晶片互動設計之自走車平台開發
林玉鵬(2011)，Android 平台之行動保全機器人 符信茂(2013)，多功能輪型機器人
實作 趙師葦(2012)，智慧輪型機器人設計與實作
趙英傑，超圖解 Arduino 互動設計入門，旗標出版股份有限公司
ArduinoInfo.InfoWIKI，MotorDrivers，

http://arduino-
info.wikispaces.com/MotorDriversPulseWidth

Modulation(PWM )，

http://zh.wikipedia.org/wiki/%E8%84%88%E8%A1%9D%E5%AF%AC%E5%BA%
A6%E8%AA%BF%E8%AE%8A

學習日誌，C++的位移方式，

http://blog.xuite.net/webrsb/study/66756029-C%2B%2B%E7%9A%84%E4%BD%8D%E7%BB%89%96%B9%E5%BC%8F