遠端控制手/自動駕駛車

組員：

106361624鄭勝中(隊長) 做事占比：25%

106361603張浩維 做事占比：25%

106361606黃培堯 做事占比：25%

106361609張凱傑 做事占比：25%

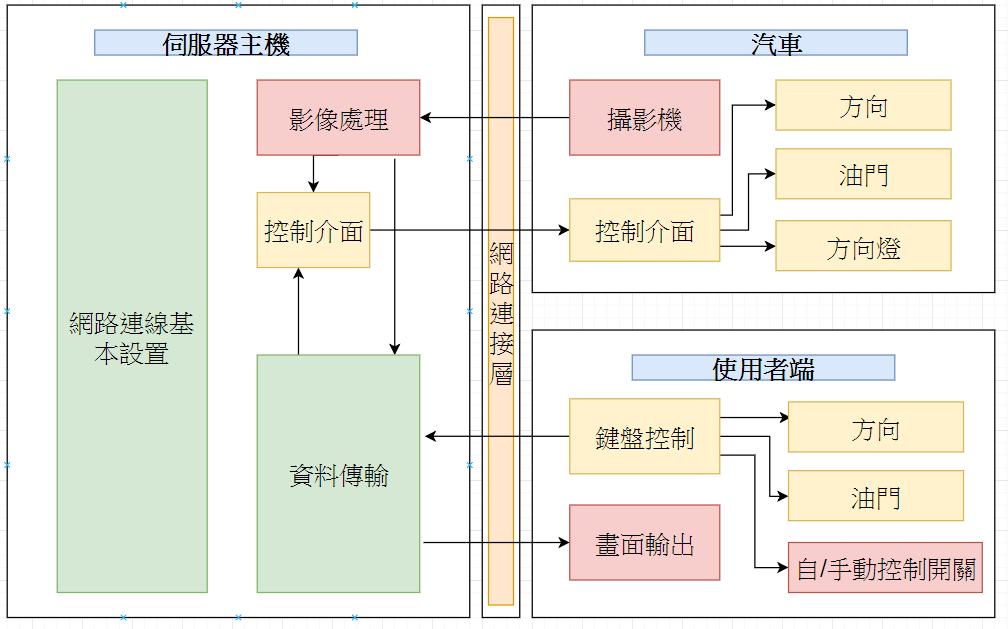
功能簡介：

建立一個伺服器來控管所連線的車子，再利用使用者端連線伺服器來對車子下達命令，車子可以透過指令的請求來執行前後左右的控制。

材料：

|  |  |
| --- | --- |
| 名稱 | 備註 |
| 樹莓派 | Raspberry Pi 3B+ |
| 電源 | 電池盒、樹莓派電源供應模塊(含UPS) |
| 車體 | 兩段變速，轉向 |
| 開關 |  |
| 可變電阻 | 調整電壓大小以控制馬達的扭力及速度 |
| 伺服馬達 | SG-90 |

功能區塊圖(預計)：



功能介紹：

伺服器：

簡介：  
提供汽車跟使用者的中控端，這個可以延伸往下做，只要將汽車連入後都加上編號，將他的連線資訊存檔，即可實現多台汽車的同時連線控制。

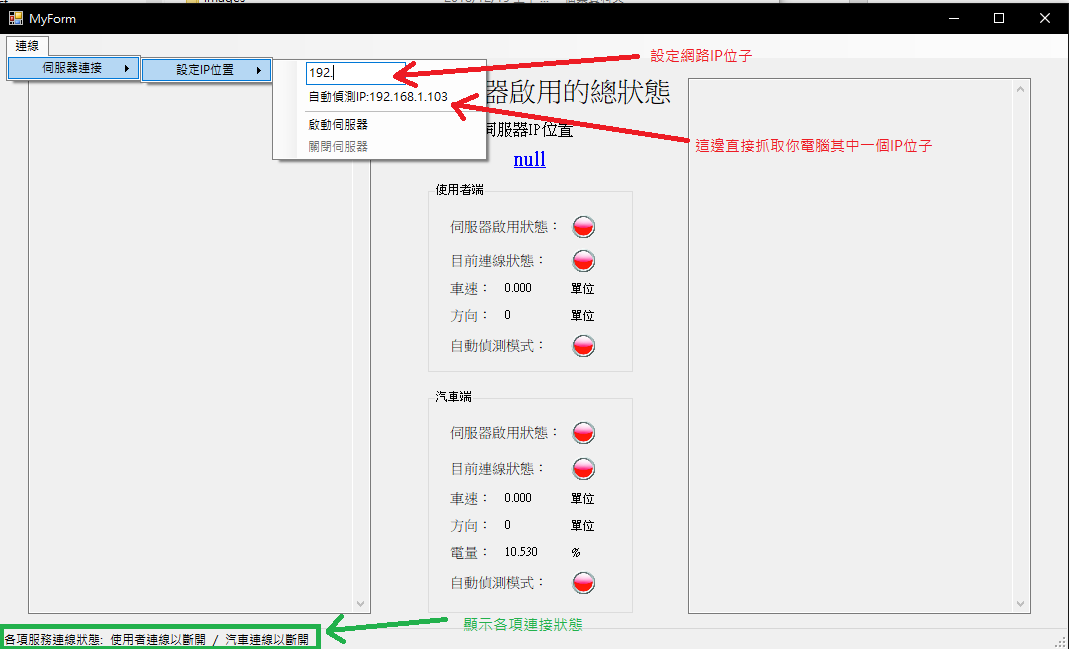
製作負責人：鄭勝中

遇到問題：

1. Vc++ 2015 年後規定不能使用sockaddr\_in.sin\_addr.s\_addr = inet\_addr(“127.0.0.1”)來設定IP位址，必須改成使用inet\_pton(AF\_INET, “127.0.0.1”,&sockaddr\_in)來設定IP，原因是因為有IPv6 及 IPv4的網路協定，微軟為了要區分差別所以強制要求程式設計師一定要使用函式來做轉換(補充:AF\_INET = IPv4, AF\_INET6 = IPv6)。
2. gcnew此指令跟new的指令很像我會用到是因為，vc++的textbox的文字存取式此用到一個叫做string^的變數我需要透過gcnew string(放入一串字元陣列) 來轉換，他會造成記憶體慢慢增加，導致我無法正常控管記憶體的使用量，所以我利用GC::collect();來強制釋放未使用的記憶體給系統。

心得(鄭勝中):  
在做這份專題時雖然遇到的問題很多導致無法達成跟當初預計的成果，但是我在這次的專題的製作過程中有學到很多內容，像是利用vc++ 來製作一個圖形化界面來控管伺服器以及使用者操作的介面，也了解到樹莓派的一些強大的功能，未來可以的話我們也會嘗試製將其他元素加進來。

展示圖：





參考資料：

* <https://www.youtube.com/watch?v=nIpOR6FX4RA&t=204s> (用vc++ 寫clr圖像界面)
* <https://awei791129.pixnet.net/blog/post/27855564-%5Bc%2B%2B-cli%5D-%E5%A4%9A%E5%9F%B7%E8%A1%8C%E7%B7%92%E7%9A%84%E5%AF%A6%E4%BD%9C-%E4%BD%BF%E7%94%A8-system%3A%3Athreading>(clr多執行續使用)
* <https://blog.csdn.net/eagle51/article/details/53157643> (inet\_ntop函數使用)
* <http://beej-zhtw.netdpi.net/09-man-manual/9-14-inet_ntop-inet_pton> (inet\_ntop函數使用)
* <https://bbs.csdn.net/topics/220077286> (gcnew釋放問題)

使用者介面：

* 簡介：  
  製作出使用者介面，並使用IP位址連接伺服端，左側暫無顯示功能，右側顯示目前控制方向以及操控控制，藉由使用者介面將指令傳進伺服端，至終控制車輛前進。

製作負責人：張凱傑

遇到問題：

我們在操作車子時希望可以使用複合鍵的方式實現前進轉彎跟倒退轉彎這樣的功能，一開始我以為使用VC的事件觸發就能輕易做到，發現好像要使用qkeyevent才能做到，而那需要使用另外一個叫做Qt Creator的IDE才能使用這個函式庫，礙於時間的關係，目前使用數字鍵代替方向。

心得(張凱傑)：  
這次製作專題我負責的是client端的程式和light sensor的製作，light sensor原本是為了要做距離感測的一個量測模組，但因為時間不構的關係，來不及完成那部分的功能，程式的部分為了使我們的專題擁有使用者介面，我們這組選擇使用VC來製作server跟client端的程式，這對我來說是很大的技術障礙，因為我從來沒有打過VC，不過最終在google大神和同學的協助下還是讓這次的專題有了基本的功能，控制端跟server間傳輸的工具我們使用以前課堂上教的winsock來製作，相對來說簡單一些，不過在VC的打法上還是有些不同，總之這次的專題在程式編輯上我也有了很大的進步。

展示圖：



參考資料：

* <https://stackoverflow.com/questions/7176951/how-to-get-multiple-key-presses-in-single-event> (抓取按鍵)
* <https://my.oschina.net/jannn/blog/515291> (QDemo之按鍵常按)

硬體端：

* 簡介：  
  伺服端資訊傳送至硬體，藉由程式內的架構，包含WIFI連接、轉彎角度、馬達輸出，控制汽車轉向以及前進。

製作負責人：黃培堯及張浩維

遇到問題：

1. 中後期發生了樹莓派無法使用Wifi連線的功能，且最後連Wifi Icon設定等都消失，第一次想盡辦法，藉由網上的資料，從Driver到自己去文件檔打內容以及最下下策的重新安裝都是了一遍，Wifi仍然無法開啟或驅動，最後只能使用USB無線網卡的硬體設備做排除，中間有想過使用藍芽開藍芽網路分享，但考慮到短距離會無法連線的問題，最後退一步使用網卡。
2. 在樹梅派上用C寫socket要使用Linux專用的標頭檔，上網找到Linux要引入”sys/socket.h”。
3. 要控制樹梅派的GPIO，找到使用C控制樹梅派GPIO的方法有bcm2835及WiringPi函式庫。
4. 伺服馬達所需要的頻率比使用WiringPi所控制的最低PWM頻率還來的低，自己寫一個控制Digital GPIO高低電位來產生可調DC的PWM脈波。

心得(張浩維)：  
期末專題大家分工合作完成一下專題報告，而我大致上負責車子本體，也簡單的寫一些Code，雖然無法像其他隊友Code打的如此強，藉由短短的一個月完成一項從無到有的車體，並能連線且無線操控，這次的專題，也對於未來大三時的專題課程很有幫助，未來在實作專題上，更能有效的利用時間完成一項實作。

心得(黃培堯)：  
這次製作遙控車的部分對我來說是難得的一次經驗，在過去幾年的經驗告訴我自己寫出來的東西要與外界溝通是非常麻煩的，多虧有組長在一開始就先規範好要寫哪些變數是要公開給server使用的，才不會浪費一堆時間與server sync socket要傳送的格式。我覺得整台遙控車在機構、server、client(控制端&車子端)整合起來後可以正常的工作實在是挺有成就感的，而且在大二的時候就能夠做這種小專題出來，對大三要做的大專題來說是個很不錯的預先經驗累積。

參考資料：

* <http://zake7749.github.io/2015/03/17/SocketProgramming/> (socket 教學)
* <https://coldnew.github.io/f7349436/> (GPIO控制)
* <http://cheng-min-i-taiwan.blogspot.com/2013/04/raspberry-pi-c.html?m=1> (樹梅派控制)
* <https://blog.csdn.net/wu_lian_nan/article/details/72510395> (伺服馬達控制)

其他相關資料：

* <https://www.youtube.com/watch?v=tdJrd75_068> (樹莓派3b+ 安裝ubuntu Mate)
* <https://blog.csdn.net/huhongtao1989/article/details/79342280> (opencv影像處理)
* <https://blog.csdn.net/u012201326/article/details/43937035> (opencv影像處理)
* <https://blog.csdn.net/huhongtao1989/article/details/79342280> (opencv影像處理)