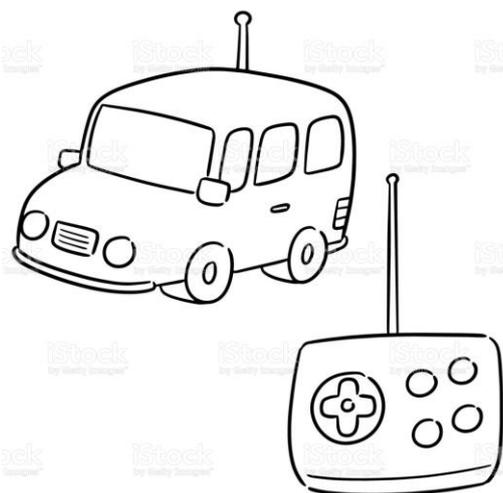


機電整合與系統設計

機電整合(四)



授課教師：周瑞仁

組別：13

B06611003 林峪台

B06611008 賴乙豪

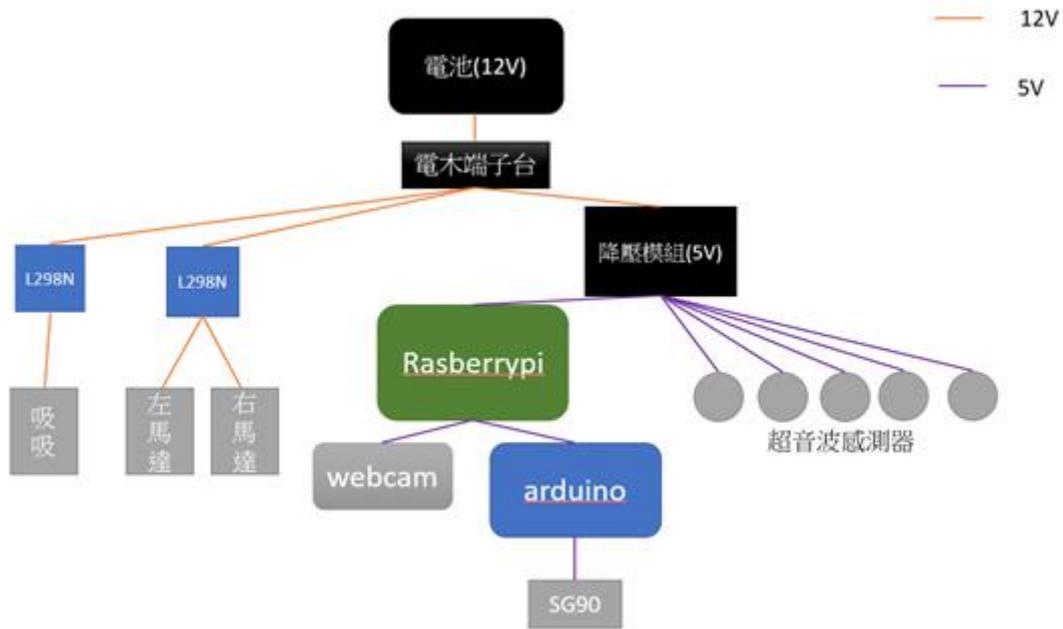
B06611032 武敬祥

一、系統說明

(一)硬體說明

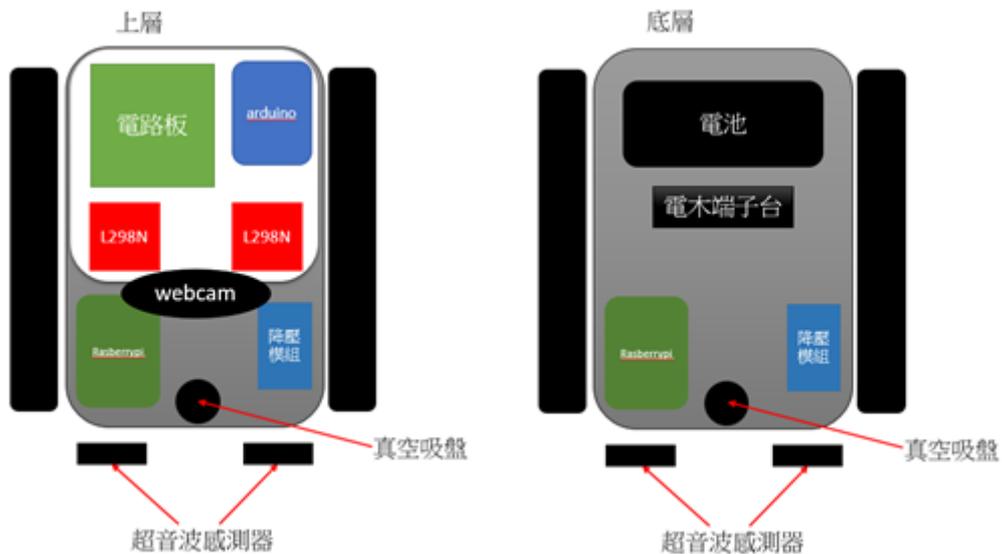
1.架構圖

電源配置:

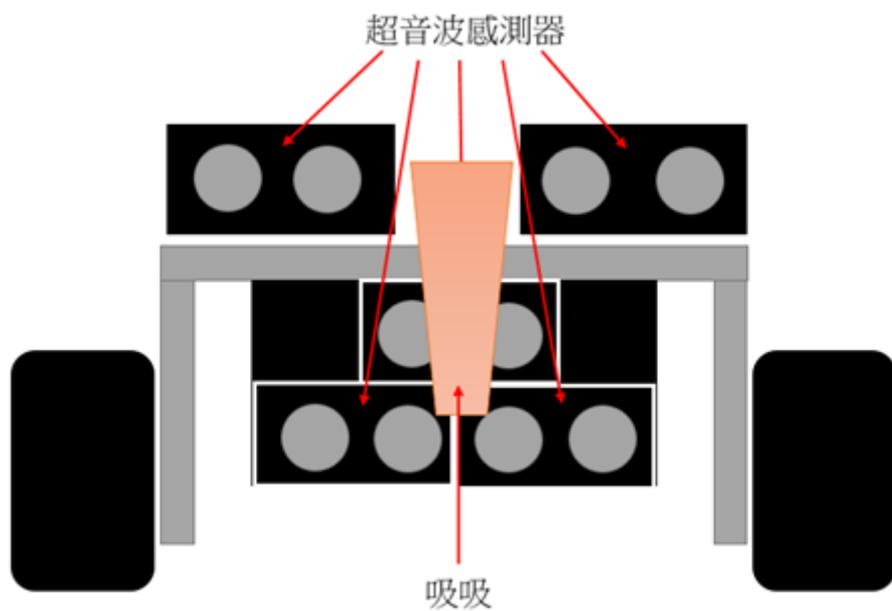


車體配置簡圖:

上視圖

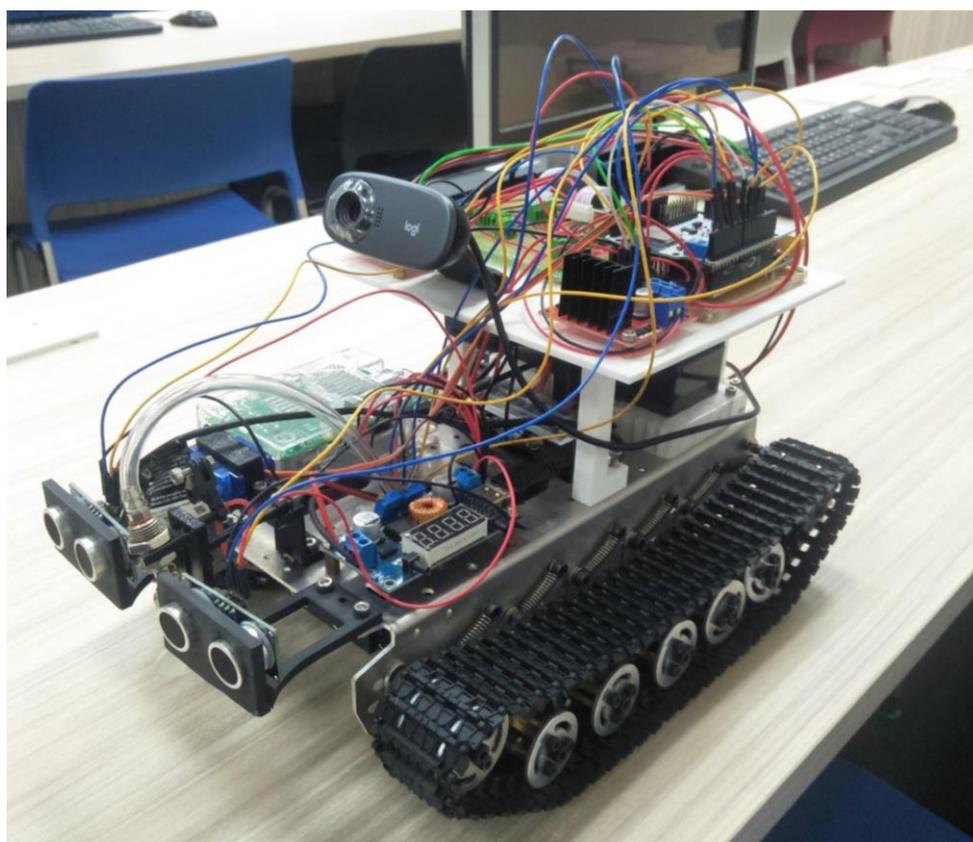


前視圖

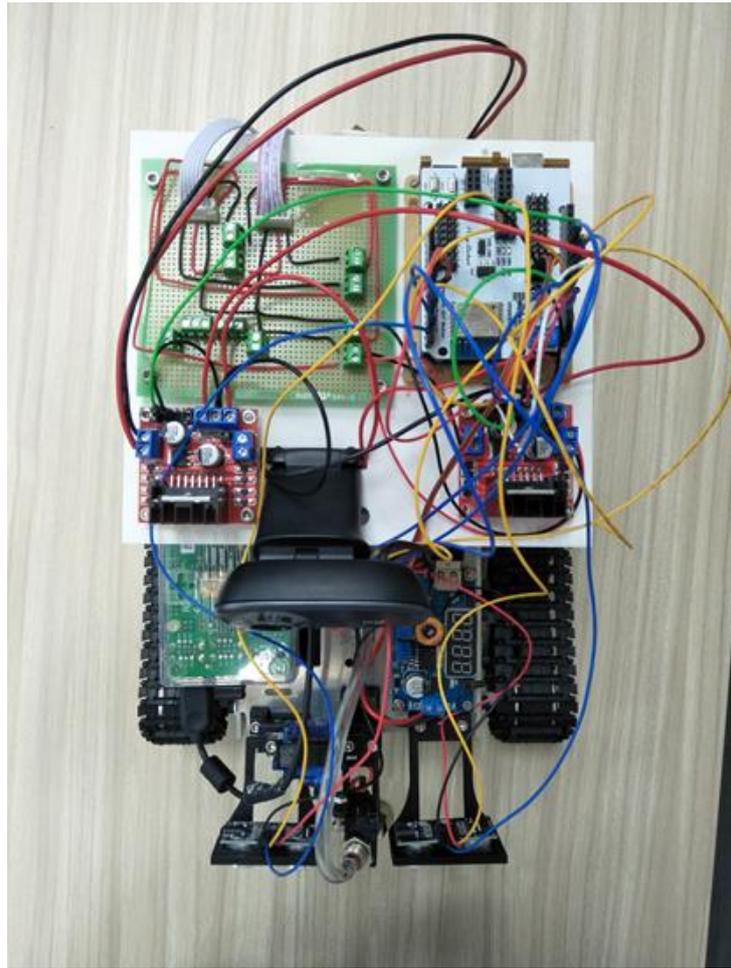


車體配置實體圖

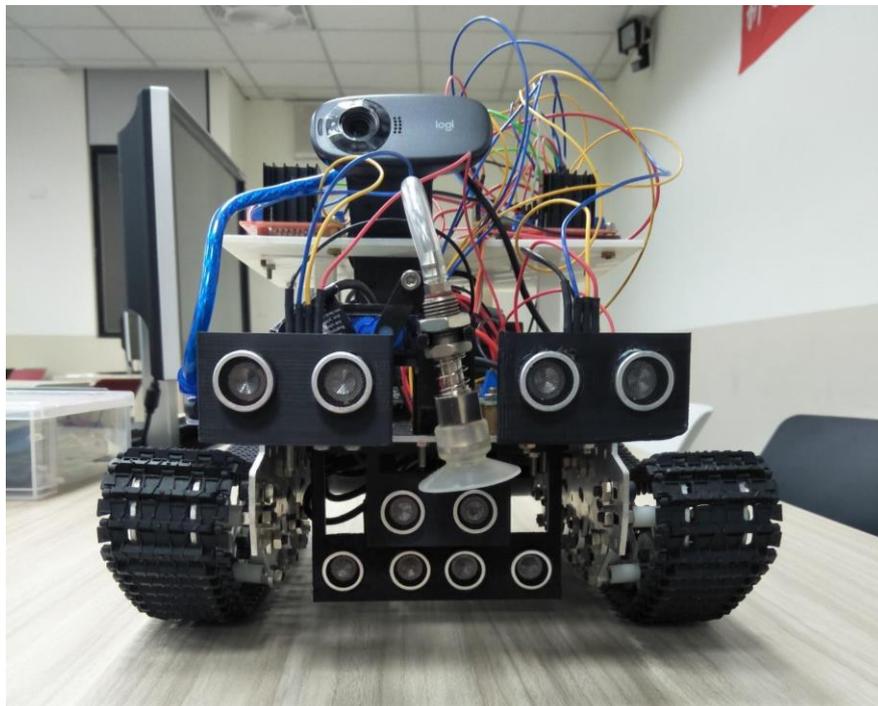
全景



上視圖



前視圖



元件

1. 車體



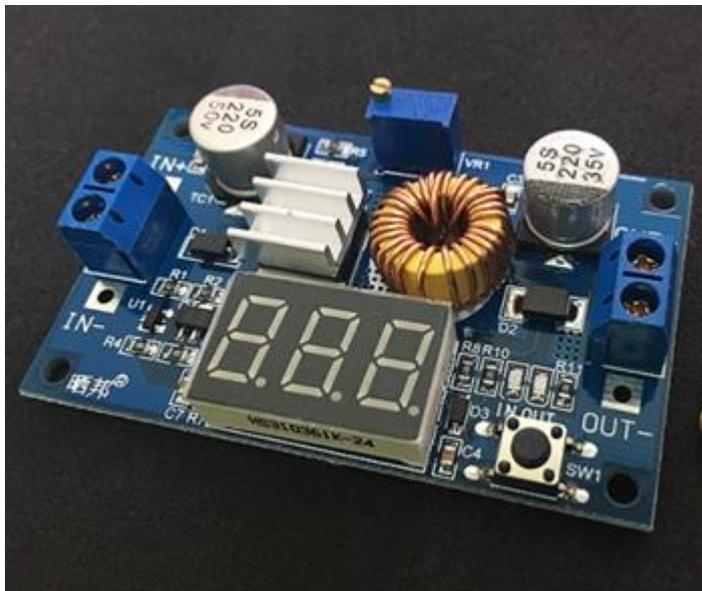
2. 鉛酸電池



3.電木端子台



4.降壓模組



5.真空吸盤



6.L298N

7.Rasberrypi

8.Arduino

9.Webcam

10.SG90

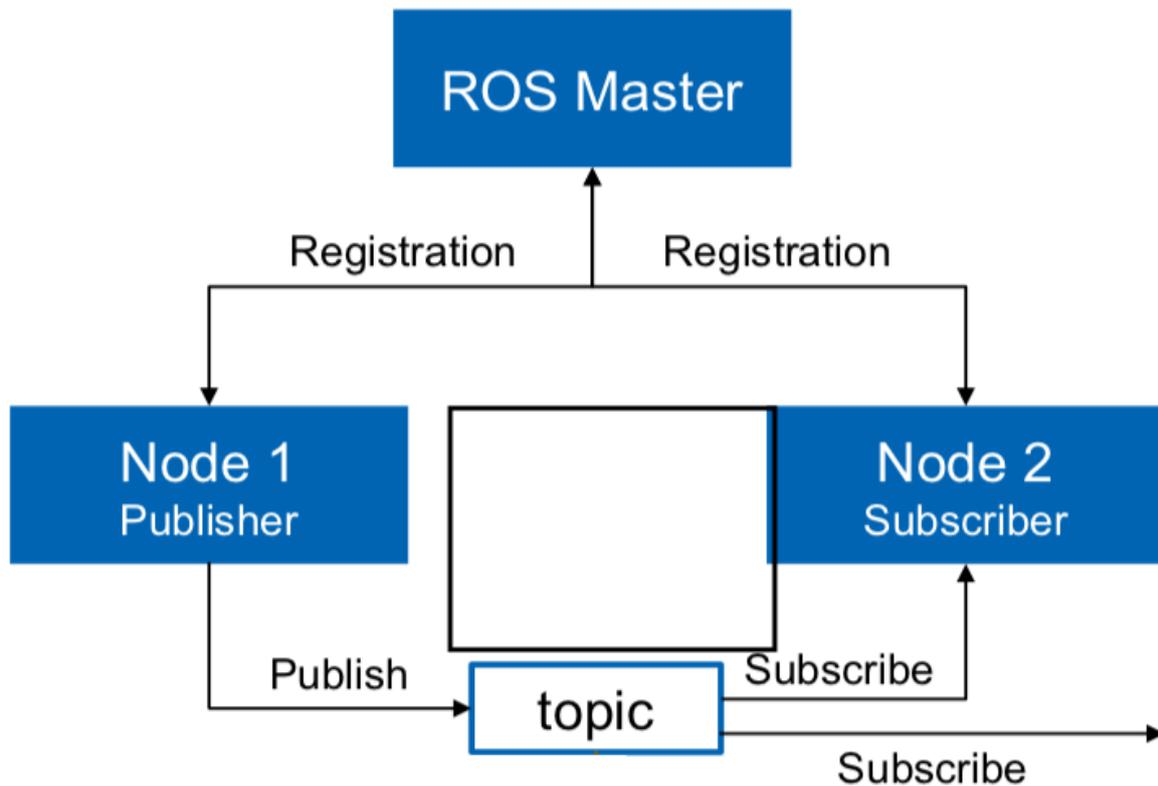
11.車用馬達

(二)軟體說明

本次的專題，我們嘗試使用了ROS(機器人作業系統，Robot Operating System)，作為系統的基底，並進行各功能模塊的整合。

ROS

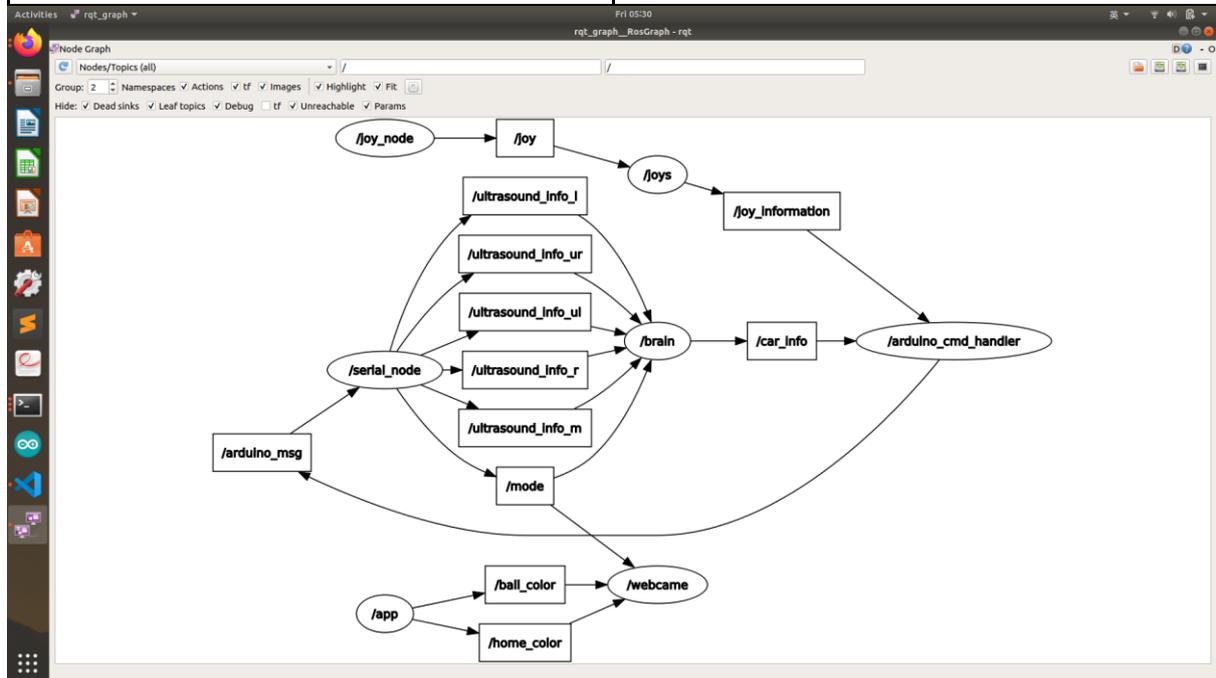
是專為機器人軟體開發所設計，提供類似於作業系統的服務，首要的設計目標是在機器人研發領域提高代碼復用率。ROS的運行架構是一種使用ROS通信模塊實現模塊間P2P的鬆耦合的網絡連接的處理架構。



簡單來說ROS架構下的每支程式都可以作為節點(Node)且單獨運行，Node可以發布或訂閱主題，以達成整個系統各程式之間的溝通。

| 程式名稱 | 功能 |
|-------------------------------|--|
| serial_node.py | 與arduino透過serial port溝通 使arduino也可以發佈node |
| ros_arduino_communication.ino | 讀取HC-SR04的資訊 輸出PWM訊號控制各馬達(車輪、氣泵、SG90) |
| webcame.py | 拍照並進行影像處理 |
| arduino_cmd_handler.py | 處理並發佈馬達的指令 |

| | |
|----------------|------------------------------|
| app.php | 接收手機訊息後執行phpexecthis.sh |
| phpexecthis.sh | 執行app_color.py |
| app_color.py | 發佈訂單需求 訂閱完成訊息 |
| brain.py | 訂閱sensor資訊後進行判斷， 並下達指令 |
| joy_node | 發佈搖桿指令 |
| joy_info.py | 處理搖桿資訊，如放大、調整 發佈頻率，並對應各馬達 |



附圖為整體系統各程式連接之架構圖

二、系統操作說明

手動模式

1. serial_node.py使arduino進入ros系統內

2. 搖桿透過joy_node發佈指令到ros
3. joy_info.py把搖桿數值remap到 [0, 100] 的區間內
4. arduino_cmd_handler.py發佈指令給arduino

自動模式

1. serial_node.py使arduino進入ros系統內，arduino即開始發佈各超音波感測器資訊並等待馬達指令
2. 使用者透過手機傳送需求給app.php，php執行app_color.py來傳送需求，系統設定當前目標為球
3. webcame.py透過指定顏色來辨識目標的位置資訊
4. brain.py分析完webcam和超音波的資訊後下達移動命令，若太遠則直走，若目標偏離中間則轉彎，若達到指定距離則下達抓取命令，夾取後系統將目標改為家並重複上述步驟。過程中左右sr04若偵測到障礙物則會進入避障路線

三、困難與解決

- ROS不適合在Debian環境
 - 重灌ubuntu mate 16.04的rpi
- 共同開發的檔案管理麻煩(google drive 很難滿足需求
 - 使用github進行版本控制
- arduino 無法進行multithreading
 - 想辦法縮短單loop執行時間/區分遙控與自動的架構
- php exec 的環境變數和terminal不太一樣
 - 執行腳本並在腳本裡設定變數
- 如果換網路，app的ip需要重灌

- 設計可變ip的app
- 搖桿的操控不夠real time
 - 改變發送指令方法：從有變化才發送修改成持續發送最後的狀態
- 吸盤的可吸範圍很小，位置需要控制的非常準(直徑六公分)
 - 使用三個超音波模組以精細地測量球的位置
- 氣泵口跟氣管大小不一樣
 - 用打火機加熱撐大氣管
- webcam角度視線被抓取機構擋住
 - 使用超音波輔助
- 電池容易沒電
 - 使用電源供應器，並準備備用電池
- 原創的杜邦線易有問題，尤其串線時
 - 使用自製線材

四、結論

雖然用了ROS來整合省去了各個程式之間溝通的問題，我們還是沒能在時間內完成自動模式。最主要的原因還是時間的掌握，就是老師在課堂上所教的專案管理，可惜未能在此次專題中良好的運用其技巧。