

機械工程實務 第 27 組 期末報告

組員

董逸棋 B03502131

黃柏瀚 B04502075

徐敏懷 B04502119

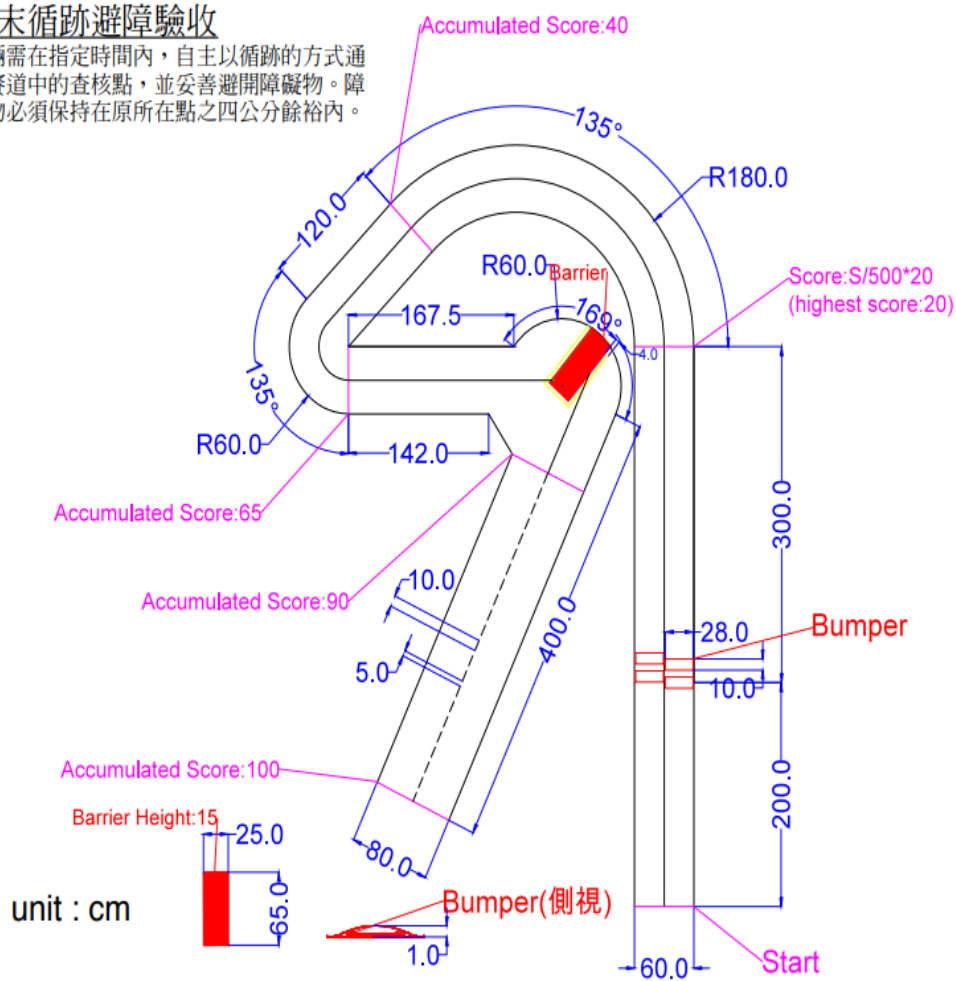
黃士瑜 B04502059

目錄

- 壹、緒論
- 貳、功能需求
- 參、車體與轉向
 - 3.1 設計介紹
 - 3.2 轉向系統設計考量
 - A.三輪車
 - B.四輪車
 - C.五輪車
 - 3.3 各版本車體與轉向的演進
 - 3.4 阿克曼轉向系統的取捨
 - 3.5 三輪車及四輪車之翻覆性比較分析
 - 3.5 摩擦力實驗
- 肆、風扇與動力
 - 4.1 設計前的評估
 - 4.2 風扇的數量與尺寸規格
 - 4.3 扇葉的數量與擺設
 - 4.4 扇葉的設計與製作
 - 4.5 扇葉的模擬分析
 - 4.6 各代扇葉的比較
 - 4.7 實驗驗證
 - A.風洞與風扇推進力實驗
 - B.風扇流場的視覺化與量測
 - 4.8 小結與期末改進方向
 - 4.9 參考資料來源
- 伍、感測系統與控制邏輯
 - 5.1 紅外線循跡
 - 5.2 超音波避障
 - 5.3 控制邏輯簡圖
- 陸、工作項目與分工
 - 6.1 甘特圖
 - 6.2 BOM 表
 - 6.3 分工表
- 柒、心得與建議
- 捌、附錄
 - 8.1 程式碼
 - 8.2 工程圖

期末循跡避障驗收

車輛需在指定時間內，自主以循跡的方式通過賽道中的查核點，並妥善避開障礙物。障礙物必須保持在原所在點之四公分餘裕內。



1. 圖二、期末測試賽道與規則

2.

考慮到這次的賽道需要有大角度的轉彎，我們選用可能轉彎半徑範圍較大的三輪轉向系統，一樣要由風扇為唯一動力來源，並選用紅外線感測器幫助車子循跡。此外，考量到需要對 barrier 進行自主避障，超音波測距儀也會加裝於車體前端，讓功能更齊全。

貳、功能需求

1. 期中查核與期末驗收過程以全自主循跡，不得以遙控控制車輛，期末繞圈競賽則以遙控完成(本學期不提供公用遙控器)。
2. 以螺旋槳產生之風力來讓車輛達到驅動與制動之所有的動力，螺旋槳需自行設計與製造，螺旋槳需有保護裝置(Helix 徑向寬度需防護，例如紗網)，避免因葉片高速旋轉時脫落崩壞導致人員受傷。除螺旋槳機組外不得有其他動力來源。
3. 整體機電系統需自行配置組裝，可採購市售機電零組件，電池與馬達數量及規格均不限制，惟需遵守 BOM 總成本規範。
4. 成本管控為工程設計的重要一環，本次競賽最終設計成品的材料(BOM)總成本不得超過新台幣 2000 元。凡規格品元件均需保留原始型號及發票(收據)，凡自製品均需有資料佐證(例如加工過程照片或影片)，自製元件之材料也須列出，嚴禁元件委外加工。
5. 車體大小不得大於 A4 尺寸(21.0cmX29.7cm)。
6. 配重 250ml 的未拆封鋁箔包飲料作為車手重量，其位置與角度可任意擺放，惟需於車內。
考慮到以上的規定，我們將設計分別以動力推進以及轉向系統分別進行設計，並且依循分析與實驗，以達到期中測試的需求。
7. 期末測試多了一開始在直線賽道的 bumper，其最大高度達 10mm，我們必須採用的紅外線感測器須於這個距離內仍可感測，否則會造成循跡不穩的狀況發生。
8. 期末賽道在最後的轉彎設置了障礙物，我們需要在車體上擺設超聲波來確定障礙物的位置並控制其轉彎，再繼續循跡。
9. 最後的直線階段改成虛線，每需線長度為 10cm，間隔為 5cm，也考驗了紅外線感測器的穩定性程度。

參、車體與轉向

3.1 設計介紹

- A. 底盤：採雙層設計，使用堅固且重量輕的密集板。
- B. 轉向系統：採用三輪轉向機構，因設計簡單、轉向靈敏度高、將伺服馬達直接接在輪子上即可、伺服馬達輸出角度即為車身轉向角度，可達到最大轉向角度。且容易計算及控制所需角度，無須複雜的幾何轉向角度運算及控制轉向，不會有輪胎打滑的問題等。
- C. 車輪：車輪皆採取金屬外框、摩擦力大、穩定性高的橡膠定向輪。因轉向的機制由輪胎控制及高速轉彎離心加速度所需的最大靜摩擦力之考量，我們傾向找摩擦力較大的膠輪材質作為轉向輪，轉向較穩固且較容易控制。

3.2 轉向系統設計考量

為了因應期中及期末不同賽道，需面臨不同的挑戰考量，原本我們傾向一次就設計決定好轉向系統，以避免中途一再的修改及變更，而浪費更多時間與金錢。所以，我們在設計轉向系統時，收集資料非常地詳細，並經過客觀的討論，較不採取一般嘗試錯誤 (trial and error) 的方式，以避免資源無效的浪費。我們在設計轉向系統時主要考慮使用三輪、四輪轉或五輪向系統，但在資料蒐集及小組討論出其優缺點如以下說明：

A. 三輪車特徵：

優點：轉向機構設計簡單、轉向靈敏度高、將伺服馬達直接連接在輪子上即可。伺服馬達輸出角度即為車身轉向角度，容易計算及控制所需角度，無須複雜的幾何轉向角度運算，亦容易控制轉向且容易操控；不會有輪胎打滑的問題等。

缺點：穩定度不佳，易因為轉彎過大或速度過快經過路面顛簸路段，而產生側翻問題。馬達轉軸直接承受前輪較大的重力所產生的反作用力，且須克服較大的摩擦力而且輸出較大的扭力以及轉矩。

B. 四輪車特徵：

優點：轉向較穩固且轉動較容易控制，底板較大的情況可以保持平衡，以四輪來移動會比三輪還要穩定。

缺點：轉向角度受到限制，迴轉半徑較大而且轉向機構設計較複雜，必須考慮轉向時輪胎可能打滑之情形。另外還需要克服齒輪及連桿傳動連接等問題，否則轉動將不如預期，也有可能會卡住。亦有可能會因為兩個前輪不同步而造成偏差，控制上較不精準。

C. 五輪車特徵：

優點：綜合三輪車及四輪車的優點。伺服馬達旋轉平面與車體平行，因此伺服馬達旋轉的角度即為前輪旋轉的角度，有利於轉向控制時參數的設定。

缺點：需要另外再外加兩個輔助輪，限制其傾斜角度，以防止高速轉彎側翻的保護裝置。

3.3 各版本缺點與轉向系統設計

第一版轉向系統設計:

經本組開會討論後，認為車體需有靈敏的轉動機制，並且考慮到可能因為轉彎過大且速度過快而產生側翻的問題，因此決定擷取三輪車及四輪車的優點，採用近似五輪車的結構設計。構造上是三輪車的特徵外，再外加兩個防側翻的緩衝輔助輪裝置。此外也考慮到轉向之靈敏度、設計簡單化、容易計算及控制轉向所需角度等。

單層車體，前輪以及後輪高度一致，規格皆為定向輪，但前輪需要連接伺服馬達的轉接板高度將比後輪多出約 10mm，故用轉接板設計來調整前輪的高度來作為改善的方法，使三輪的高度一致。輔助輪設計離地面高度約 10mm。

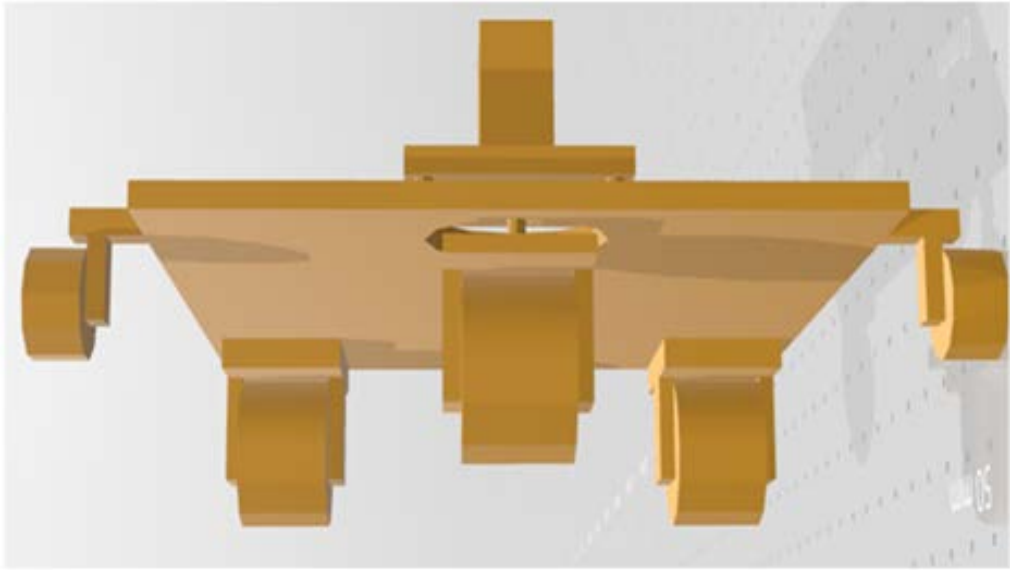


圖 1. 三輪車外加兩個輔助輪成為五輪車 (Software: Creo)

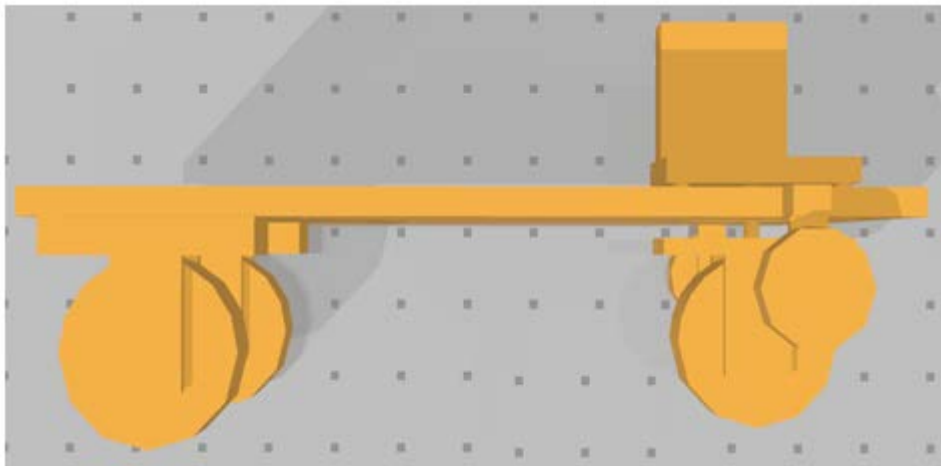


圖 2. 單層車體之五輪車設計，外加兩個防側翻之輔助輪側視圖 (Software: Creo)

但第一代五輪車設計並沒有考慮紅外線循跡感應的高度，只能離地 10mm 以內，輪子高度 70mm 可能造成重心不穩，此外期中測試並沒有要求競速或需高速轉彎等，輔助輪的助益不大。

第二版轉向系統設計:

第二版轉向系統，針對第一代版本的缺點，做了一些改良。採雙層車體結構設計，底層挖空讓輪子穿過下層板與上層板連接，上下層間由 8 根樑柱支撐。取消輔助輪設計，即五輪車之設計改成三輪車。但仍保有原來轉向系統的設計優點外，並降低車體的重心。其 3D 立體設計草圖如下:

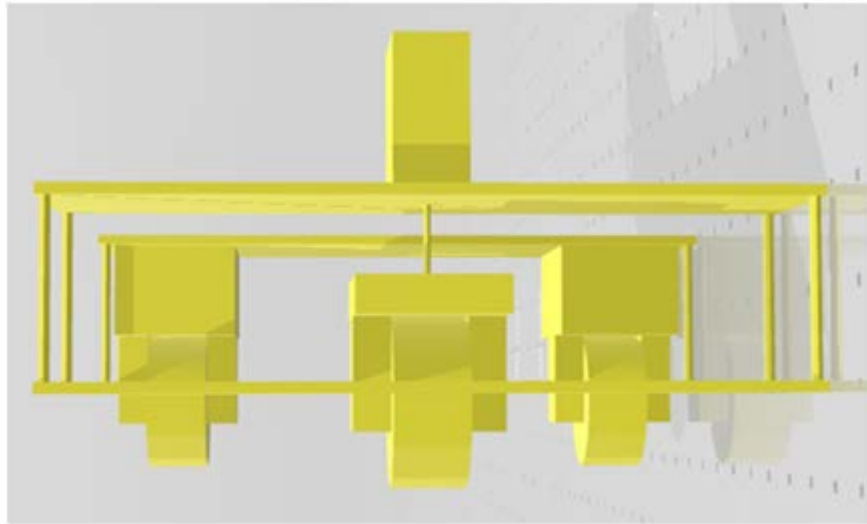


圖 3. 雙層三輪車設計圖 (Software: Creo)

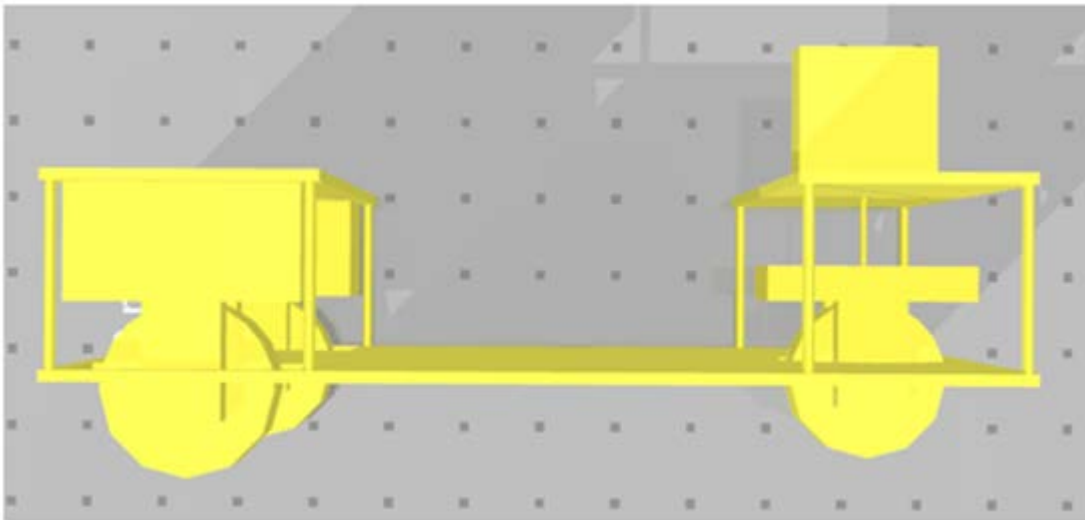


圖 4. 雙層三輪車設計圖側視 (Software: Creo)

第三版轉向系統設計:

第三版轉向系統，針對第一代及第二代版本的缺點，做了一些改良。仍採雙層車體結構設計，底層仍挖空讓輪子穿過下層板與上層板連接，樑柱改為 4 根，車體形狀改成前梯形後長方形之六邊形組合，且下層車體底板尺寸縮小。原半雙層車體結構改為真實雙層結構，以增加空間運用及電路配置。且因期中測試時循跡紅外線感應器遭到碰撞而偏移，造期中測試不甚理想。故於第三版車體改良時，於前端加上紅外線感測器保護裝置板。如下圖 5 所示:

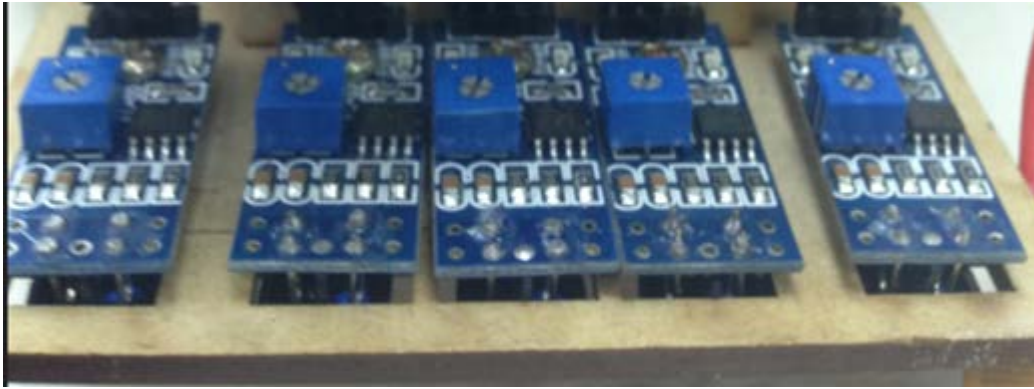


圖 5. 紅外線感測器前加裝防撞保護裝置

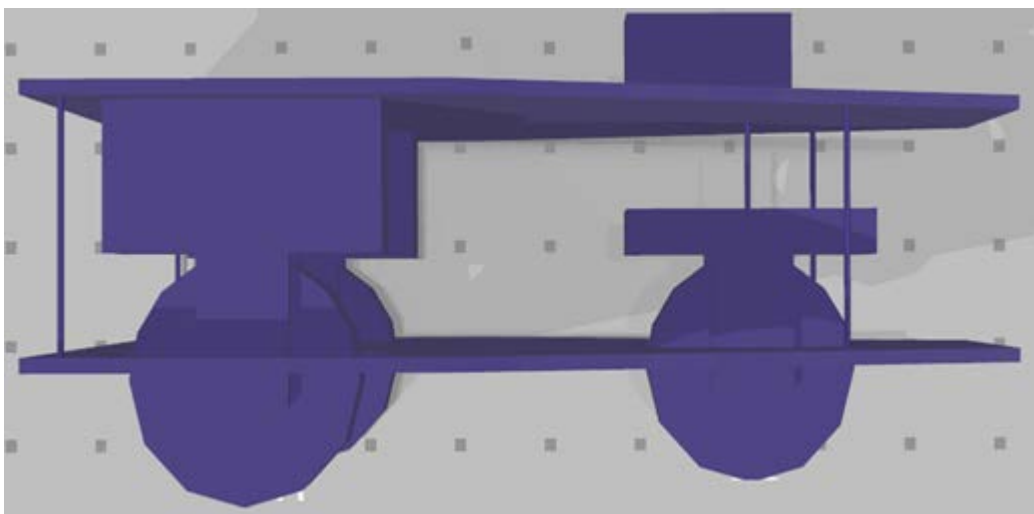


圖 6. 第三版雙層三輪車設計圖側視 (Software: Creo)

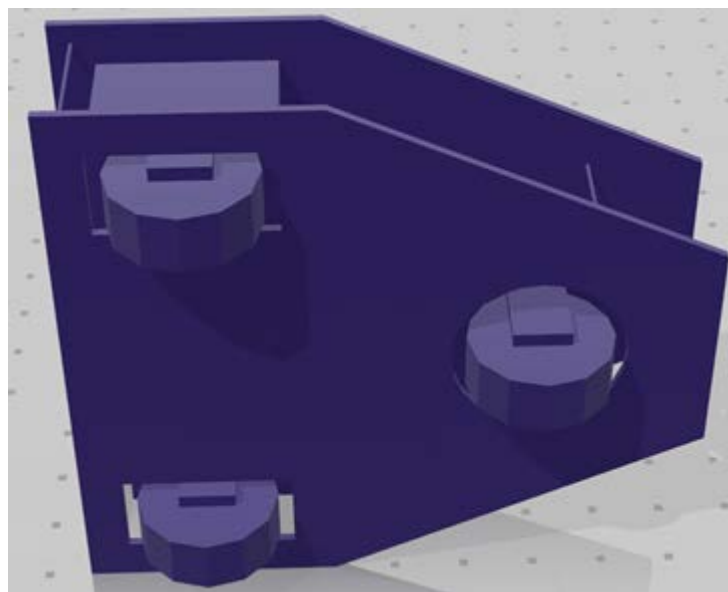


圖 7. 第三版雙層三輪車體底視圖 (Software: Creo)

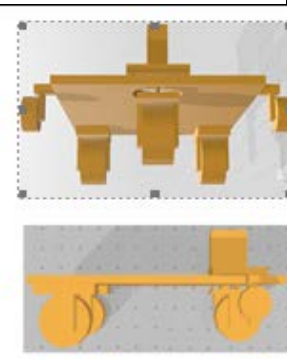
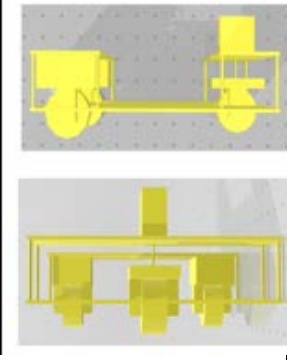

車體	圖片	優點	缺點
第一版		<ol style="list-style-type: none"> 1. 防高速轉彎之側翻。 2. 可用轉接板設計來調整前輪的高度。 3. 單層車體，重量較輕。 	<ol style="list-style-type: none"> 1. 增加車體寬度。 2. 單層車體，空間配置受限。 3. 輪子高度 70mm，直接接上下層板，可能造成重心不穩。 4. 期中循跡，非競速，輔助輪功用不大，畫蛇添足。
第二版		<ol style="list-style-type: none"> 1. 輪子接上層車體，可將重心下移，防側翻。 2. 雙層空間配置利用高。 3. 省去輔助輪之設計。 	<ol style="list-style-type: none"> 1. 設計較複雜。 2. 增加車體重量。
第三版		<ol style="list-style-type: none"> 1. 上下真實雙層車體，空間配置利用率更高。 2. 車體大小可較第二版車體再微縮。 3. 前車體加上紅外線感測器保護版，可防撞擊。 	<ol style="list-style-type: none"> 1. 車體重量又增加。 2. 競速有翻車的風險。 3. 競速時需加裝 L 型保護裝置。

圖 8. 第 1~3 版車體優缺點比較表

A. 車輪設計需求:

1. 前輪為轉向單輪，轉向部分，採用伺服馬達連結前輪的方式控制轉向，調整參數時更為簡易，能提供大幅度的轉向角度。
2. 後輪僅為從動，可以直接與螺絲鎖在上層車板上，以避免輪軸設計因承重後彎曲，輪胎脫落等問題。
3. 前後輪均使用定向輪，因為萬向輪的自由度過多，造成輪子方向亂轉，難以控制車子前進方向，因此將改用定向輪，使輪子永遠平行於車體。使用定向輪當後輪，能夠使車體本身保持穩定。

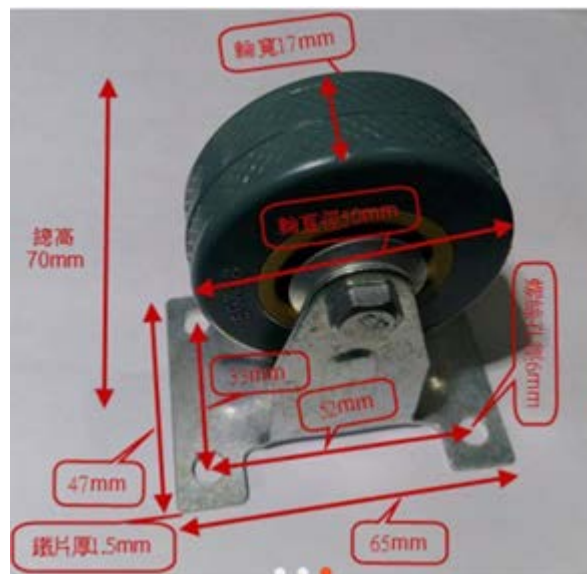


圖 9. 橡膠胎定向輪：車輪直徑 50 mm，寬度 17 mm，高度 70 mm

B. 輪距:

應考量 A4 紙張為 297mm x 210mm，且輪面寬度 17 mm，所以輪距應介於 120mm~180mm 之間，輪距越大，迴轉半徑也越大。比賽道彎道的最小旋轉半徑是 30cm，所以必須讓車子轉向的最小迴轉半徑小於 30mm。考慮到需預留上下層間由 8 根樑柱支撐的空間，輪距僅採 140mm 之短輪距設計，前輪距離後輪中心線 170mm。

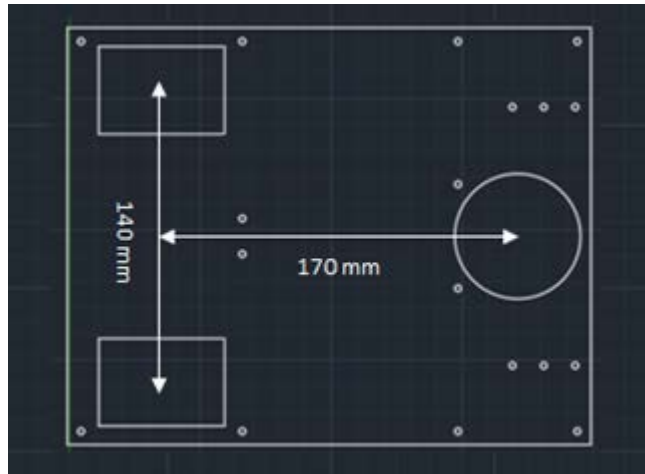


圖 10. 前後輪軸間距圖

C. 高度:

前輪及後輪高度需一致，規格皆為定向輪，但前輪需要連接伺服馬達的轉接板高度將比後輪高出許多，但也可以用調整前後輪的高度來改善或加上墊片來調整。底層高度高於輪軸，並預留紅外線循跡感應的高度。

D. 材質:

車輪皆採取金屬外框、橡膠胎之定向輪，因轉向的機制由輪胎控制，及高速轉彎離心加速度所需的最大靜摩擦力，加上期末應在煞車範圍內停車的考量，我們傾向找摩擦力較大的膠輪材質作為轉向輪。

E. 轉向伺服馬達：

用於車體的轉向機構，伺服馬達能轉 180 度，但是卻能精準的讓馬達轉到設定的角度，並維持在那裏。非常適合用來控制轉向機構，可以使用 Arduino 進行角度的數值控制。以下為伺服馬達之重要規格圖:

產品型號:MG995，

尺寸:40.7*19.7*42.9mm，

重量:55g，工作扭矩:13KG/cm，

反應轉速:53-62R/M，

轉動角度:最大 180 度，

工作電流:100mA，

使用電壓:3-7.2V 無負載

操作速度: 0.17 秒/60 度(4.8V)；0.13 秒/60 度(6.0V)



圖 11 . 超強扭力 13KG-MG995 伺服馬達

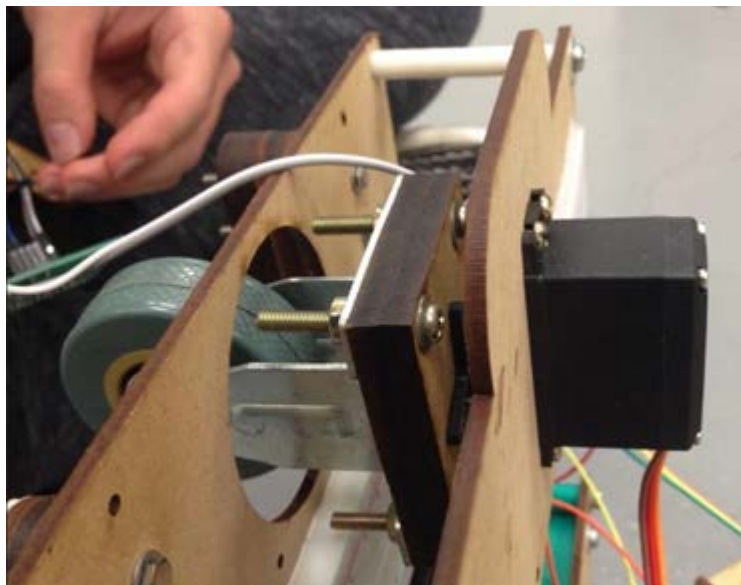


圖 12 . MG995 伺服馬達經由兩片 5mm 墊片與前輪連接作為轉向機制

3.4 阿克曼轉向系統的取捨

阿克曼轉向系統設計原理:

轉向運動時，車體是繞著旋轉中心行駛，四個輪胎的圓心落在同一點，外側車輪和內側車輪所行走的距離不一樣，在轉向時左輪和右輪的轉向角不同。阿克曼轉向角為轉向時兩前輪轉向角度之差。當轉彎時，使車輛能以後軸延伸線的瞬時中心為圓心順利過彎，彎內輪所轉的角度通常大於彎外輪。沿著彎道轉彎時，利用四連桿的曲柄，使內側輪的轉向角比外側輪多大約 2 度到 4 度，並使四個輪子路徑的圓心大致上交會於後軸的延長線上瞬時轉向中心，讓車輛可以順暢的轉彎。轉向角 δ 可由下圖表示之：

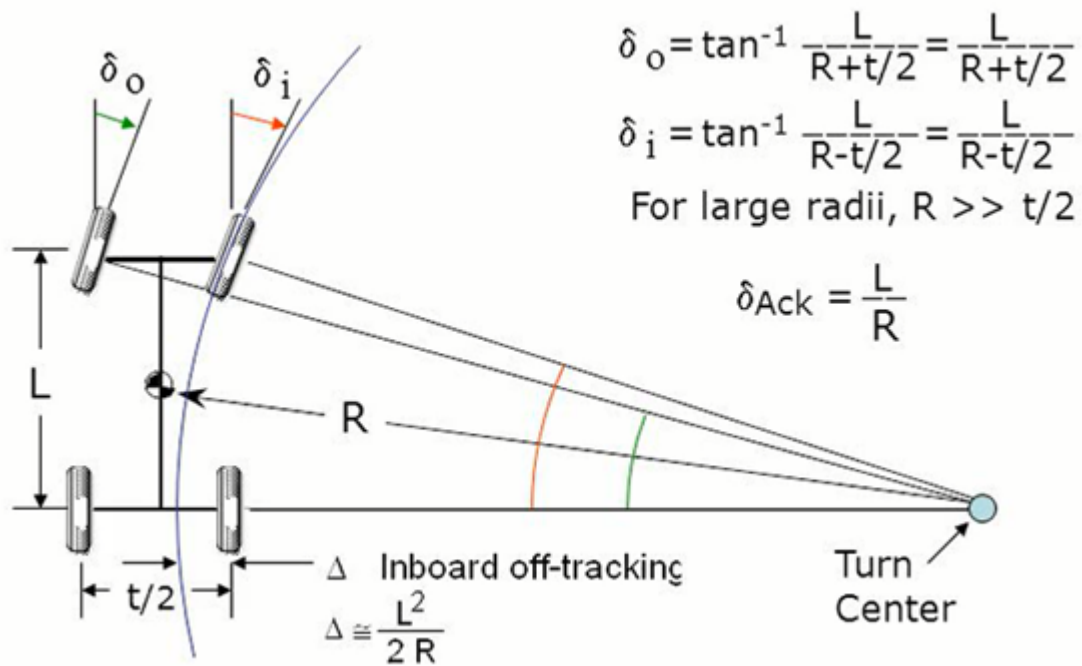


圖 13. 阿克曼轉向角 δ (Ackermann steering) 分析

阿克曼轉向系統設計分析:

以期中賽道之最小轉彎半徑 $R=30\text{ cm}$ 為例，不考慮內外輪轉角相對與地面前進方向的阻力、在低速轉彎的情況下，來估算阿克曼轉彎角度:

$\delta_{Ack} = \tan^{-1}(L/R)$ 與軸距 L 之間的關係。並可計算出伺服馬達順/逆時針轉到極限時，是否可達成期中測試之最大轉彎，並計算出適當的軸距 L 。

軸距 L 的範圍因應 A4 紙張大小計，扣掉前後預留的距離及感應系統空間後，且尚可保持適當的平衡距離估，約為 $10\text{ cm} \sim 25\text{ cm}$ 間。則可算出軸距 L 與阿克轉彎曼角度 δ 的相對應關係如下圖示：

R(cm)	L (cm)	$\theta 1$ (弧度)	θ (角度)	L
30	10	0.322	18.4	10
30	11	0.351	20.1	11
30	12	0.381	21.8	12
30	13	0.409	23.4	13
30	14	0.437	25.0	14
	15	0.464	26.6	15
	16	0.490	28.1	16
	17	0.516	29.5	17
	18	0.540	31.0	18
	19	0.565	32.3	19
	20	0.588	33.7	20
	21	0.611	35.0	21
	22	0.633	36.3	22
	23	0.654	37.5	23
	24	0.675	38.7	24
	25	0.695	39.8	25

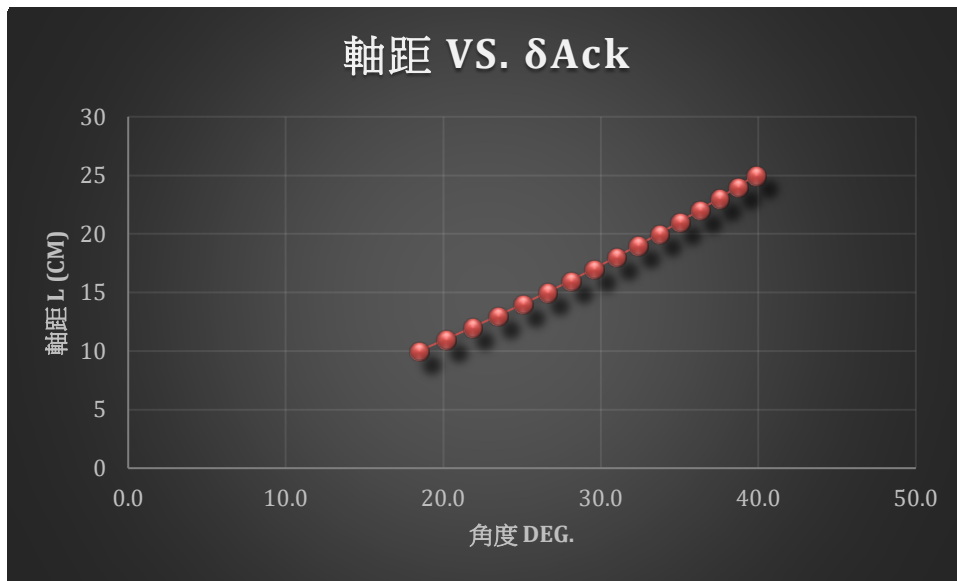


圖 14. 軸距 L 與阿克曼角度 δ 的相對應關係圖

由以上觀察得知軸距對應阿克曼轉彎角度關係，軸距越長，轉向半徑愈大，其所需的轉彎角度越大，轉彎也相對越困難。以四連桿構成的雙曲柄機構為例，如下圖示，我們可以調整控制四連桿雙曲柄的伺服馬達轉角與阿克曼轉彎角度 δ 約略相同。但仍仔細的運算與考慮的因素如：轉向角度與內輪差(轉向時內側前後輪轉彎半徑之差，稱之為內輪差)之相對應關係，轉向角度愈大，內輪差也愈大。

諸如以上各種因素，如轉向角度受到限制、迴轉半徑較大、轉向機構設計較複雜、計算及驗證的複雜度高、高速轉向時輪胎可能打滑之情形。另外還需要克服齒輪與四連桿雙曲柄及伺服馬達傳動連接等諸多問題，如齒輪鬆脫、齒輪卡住、前輪不同步而造成偏差等。否則轉動將不如預期，也不容易精準控制等。綜合以上種種原因，本組經過多次開會後決定捨棄四輪車轉向機構設計，改採三輪車系統設計。

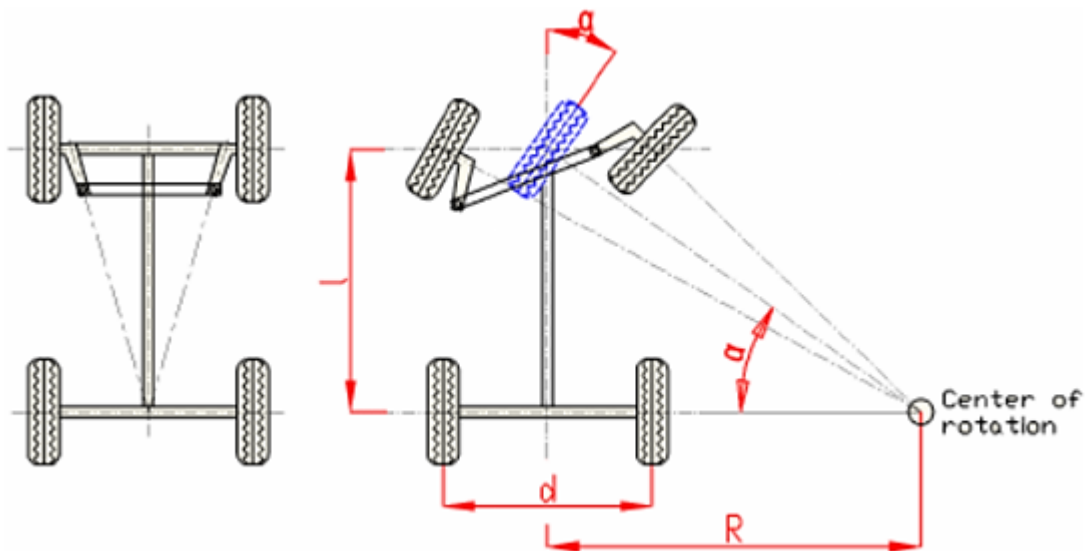


圖 15. 四連桿構成的雙曲柄機構

單輪轉向系統設計分析:

單輪轉向系統，轉彎半徑 R 是重心與後輪距連線之交點。轉向角 δ 則是後輪距連線與轉向輪垂直連線交叉點之夾角。 δ 越大則 R 越小，甚至可做 90 度之迴轉。故基本上迴轉半徑及角度均優於阿克曼轉轉向系統，靈敏度也較高，不需複雜的幾何運算。

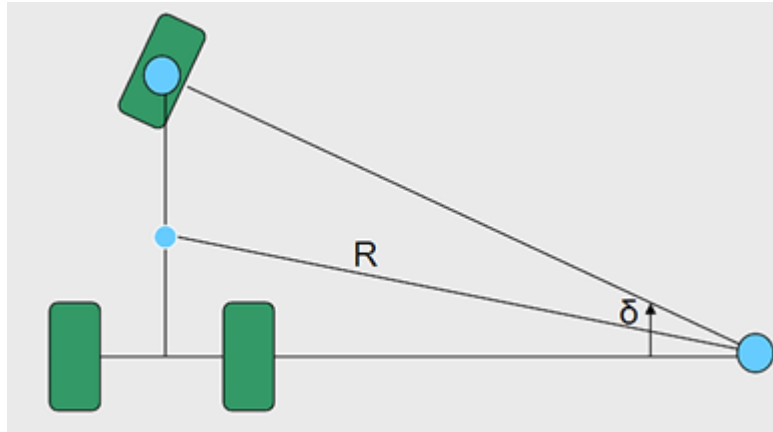


圖 16. 單輪轉向系統

但唯一的缺點就是高速轉彎會造成重心不穩，容易側翻的問題需克服。若重心過高，車速過快或是轉彎太急，即使摩擦力足夠提供轉彎時所需的向心力，但是同時對質心所產生的力矩，會使的車體旋轉翻倒。三輪車與四輪車在快速行駛的過程裡，如果碰上轉彎的狀況，三輪車會比四輪車要來得容易翻車。如下圖示：

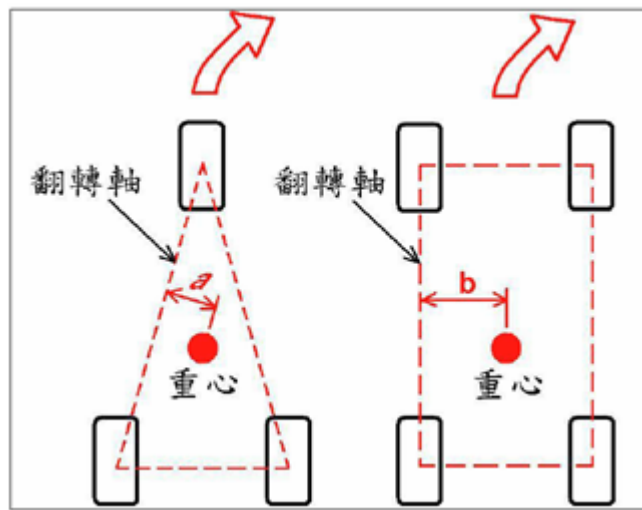


圖 17. 三輪與四輪車翻覆性比較

從上圖示，從兩個圖可以比較出三輪車的整體重心與翻轉軸間的距離為 a ，比四輪車的整體重心與翻轉軸間的距離為 b 還要短，這表示這兩種車如果在產生翻覆情況的同時，四輪車整體重心所造成的抗翻力矩會比三輪車來得大。

因此，可以知道，四輪車會比三輪車要來得穩定；倘若這兩種車體以同樣的速度前進，並且同時作出大角度轉彎動作的話，三輪車必定會比四輪車更容易有翻覆的可能。

增加車輛之抗翻覆性之改善的方式:

- A. 加裝輔助輪或緩衝裝置防止側翻。
- B. 加大三輪車的整體重心與翻轉軸間的距離為 a 。
- C. 三輪車體由四邊形改為第三版車體六邊形。
- D. 加長左右輪距。
- E. 降低車輛重心
- F. 將車子重心儘量往後輪軸移動。

考量到期中賽道並非競速，也沒限定時間內內跑完全程，則暫不考慮側翻的問題，取消加裝輔助輪或緩衝裝置防止側翻的設計。讓轉向機構設計簡單化、並提高轉向靈敏度高、本組採用三輪車之單輪轉向系統，並將伺服馬達直接連接在輪子上。伺服馬達輸出角度即為車身轉向角度，容易計算及控制所需角度，無須複雜的幾何轉向角度運算，亦容易控制轉向且容易操控。

由於物體本身的慣性，當車體轉彎時，慣性仍想依照原來的方向行進，若重心過高，車速過快或是轉彎太急，即使摩擦力足夠提供轉彎時所需的向心力，但是同時對質心所產生的力矩，會使的車體旋轉翻倒，是很危險的。

期中及期末循跡皆非競速，三輪車並沒有側翻的問題，但期末競速將高速轉彎，向心加速度加上三輪車抗翻力矩本就比四輪車會比三輪車要來得小，如不加裝 L 型輔助裝置，將有翻車的危險。

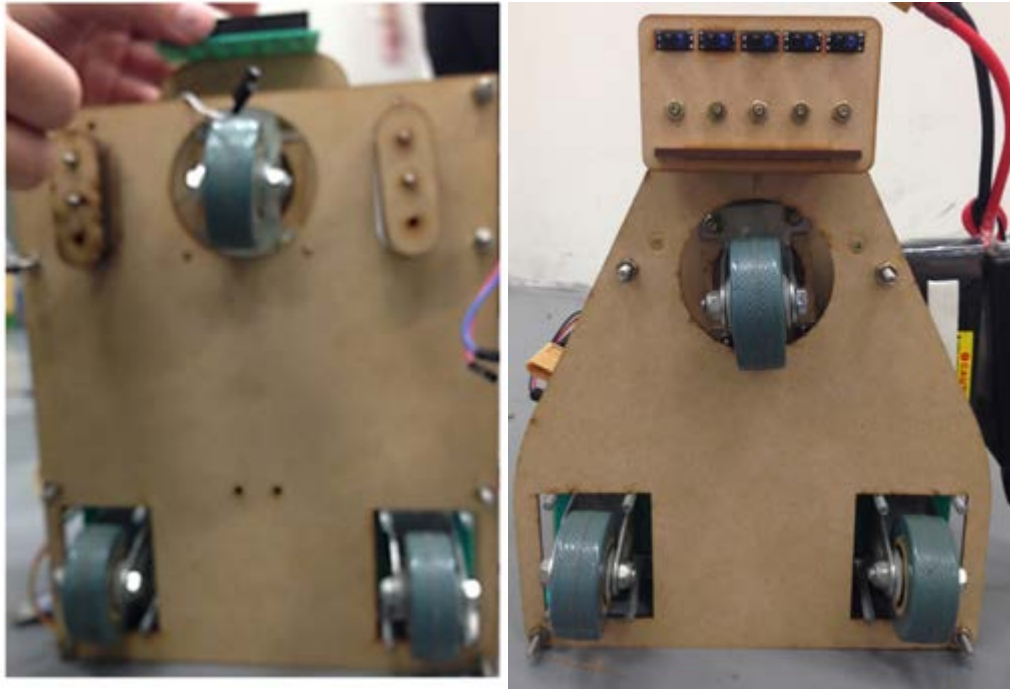


圖 18. 三輪車第二版及第三版底視圖

3.5 三輪車及四輪車之翻覆性比較分析

我們以車輛動力學的方式來分析四輪車與三輪車在進行等速轉彎、加速轉彎及減速轉彎等運動行為時之翻覆性，藉此來了解此兩種車型間彼此間翻覆性的差異。為了簡化分析，我們將車體視為剛體的分析模型，考慮車體行駛於平路上的情況，得到四輪車在左轉並煞車（ $a_x < 0$ ）時的不會翻覆條件為：

$$\frac{a_y}{g} \leq \frac{T}{2h}$$

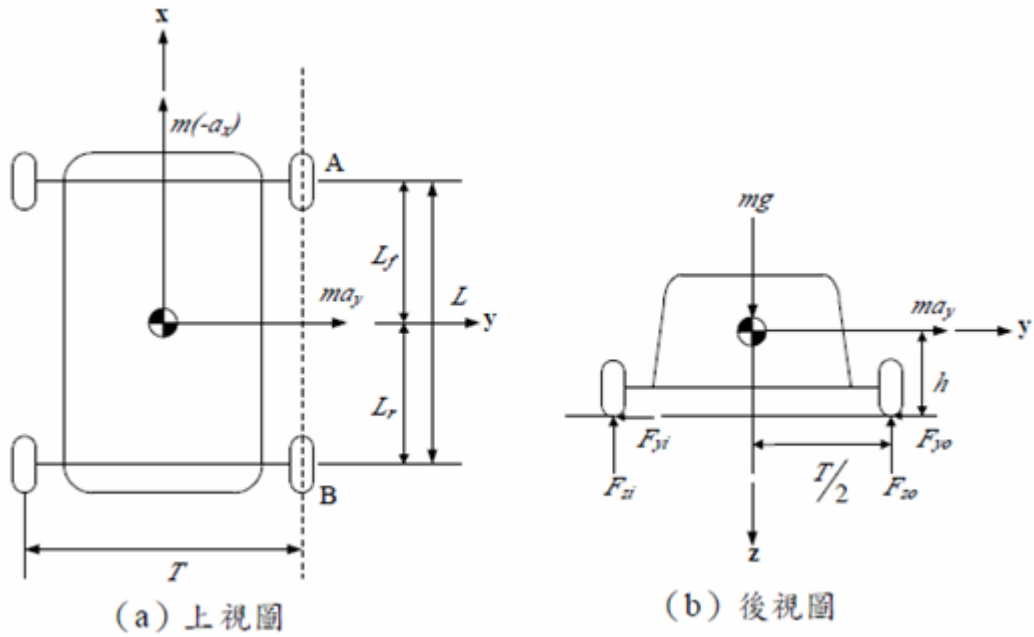


圖 19. 四輪車在左轉並煞車時之自由體圖

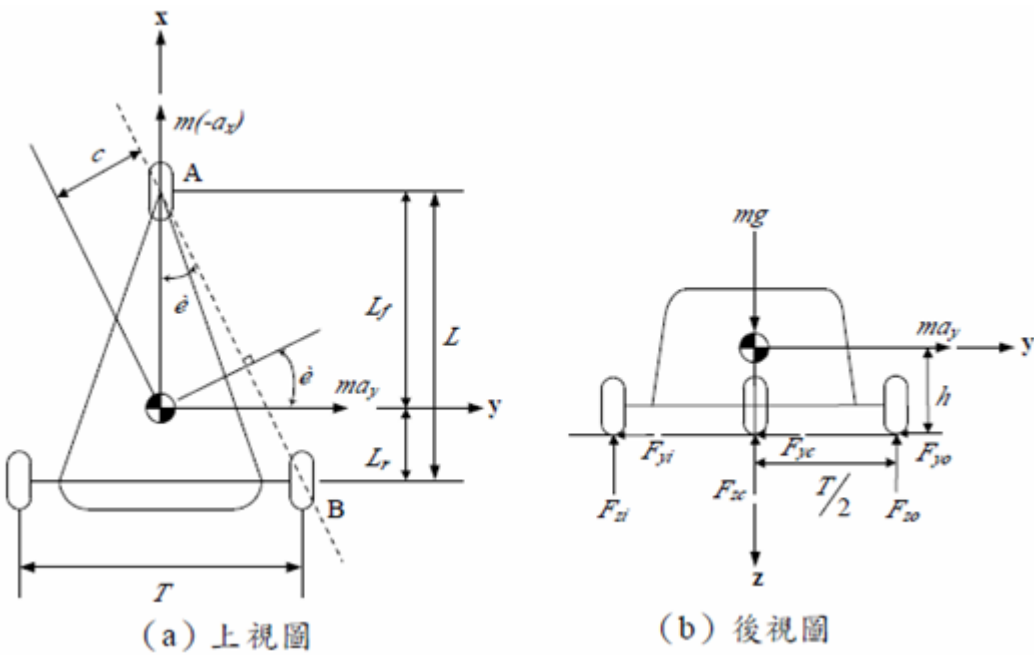


圖 20. 四輪車在左轉並煞車時之自由體圖

得出三輪車輛在左轉並同時煞車狀況下的安全條件為:

$$\frac{a_y}{g} \leq \frac{T}{2h} \left(\frac{L_f}{L} + \frac{h}{L} \frac{a_x}{g} \right)$$

m : 車輛質量	L : 前後軸距
g : 重力加速度	L_f : 車輛重心至前輪軸之距離
h : 車輛重心距離路面高度	L_r : 車輛重心至後輪軸之距離
a_x : 縱向加速度	c : 三輪車輛重心至 \overline{AB} 軸之距離
a_y : 側向加速度	θ : \overline{AG} 與 \overline{AB} 軸之夾角
T : 左右輪距 (輪幅寬度)	

上述四輪車與三輪車之翻覆極限如，等號右邊之數值愈大則車輛愈安全，亦即車輛抗翻覆性愈好。對四輪車而言，其縱向加速度與抗翻覆值無關，而對三輪車輛而言，其縱向加速度則會令抗翻覆值提高或降低。另外，若要增加車輛之抗翻覆性，可從降低車輛重心、增加左右輪距及車輛重心靠近雙輪軸等三方面著手。接著將表 2 所列之各車型參數值代入，則我們可獲得各車型之翻覆極限線，如圖 20 依公式模擬所示，若 (a_y, a_x) 點在該車型之翻覆極限線的右邊車子就會翻覆。

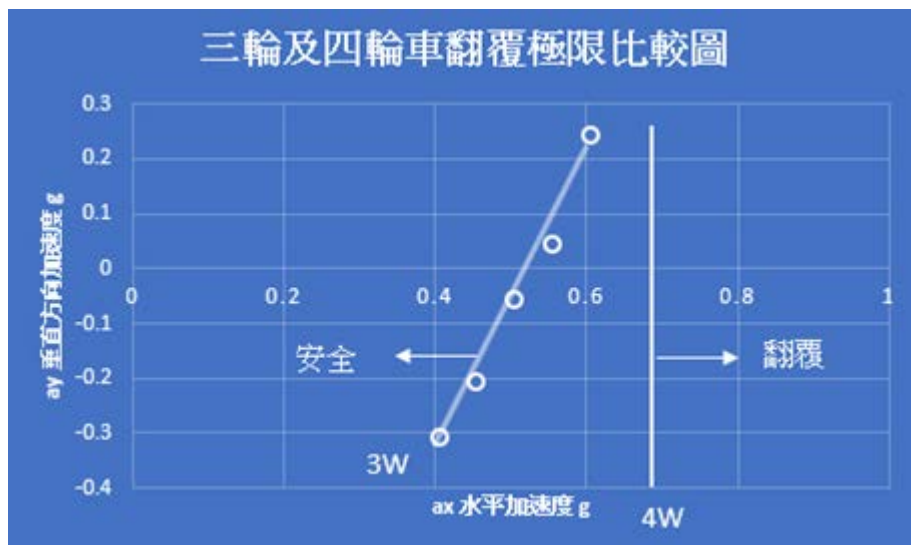


圖 21 若 (a_x, a_y) 點在該車型之翻覆極限線的右邊車子就會翻覆

由圖中，可清楚發覺四輪車輛不論是加速轉彎或煞車轉彎，翻覆時的側向加速度極限值，均比三輪車輛來得大。因三輪車翻覆極限式比四輪車輛翻覆極限多了這一項:

但若當三輪車之重心位置位於其後輪軸上時，即三輪車在加速時之翻覆極限將比四輪車輛之翻覆極限來得大，但其在煞車轉彎時之抗翻覆性仍遜於四輪車輛。因此三輪車輛之重心位置越靠近雙輪軸，則其抗翻覆性就會越好。

三輪車在煞車轉彎時比較容易翻覆，原因是煞車時車輛會受到一向前之慣性力作用，使得車身往前傾而車輛重心位置向前移，因此造成三輪車之重心位置更接近前輪之單輪軸，而增加其翻覆的機會。

3.6 實驗：摩擦力實驗

1. 實驗目的：

因我們需要計算推力要達何種大小方能讓整個系統克服最大靜摩擦力進而啟動，所以我們做了一個簡單的地板摩擦力的實驗，去推算出最大靜摩擦力係數，並算出摩擦造成的阻力為何，預測出當馬達轉速為何時，車子應有多少的速度。

2. 實驗器材：

(1) 氣動車車體 (2) 長度 60cm 密集板 (3) 量角器、尺 (4) 光滑木板

3. 實驗原理：

我們運用一個表面光滑的木板當做斜面，將車子擺置於木板上，並將木板緩緩增加傾斜角度，觀察將角度增加到幾度時，車子開始有所移動，將其記錄並計算。其中重力與斜面的水平分量 $mg\sin\theta$ 為下滑力，而重力與斜面的垂直分量 $mg\cos\theta$ 會與正向力抵消。又最大靜摩擦力 ($f_{smax} = \mu sN$) 與正向力 (N) 有關，所以我們利用調整斜面角度來測量 f_{smax} 。

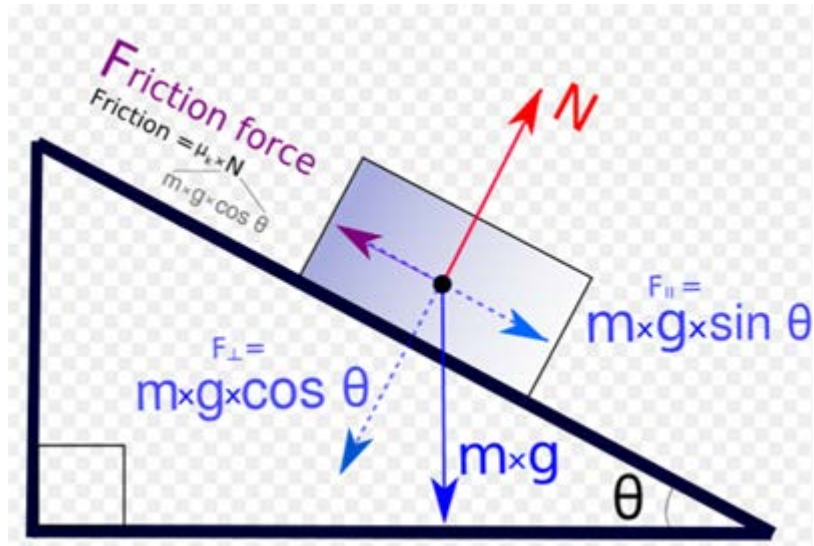


圖 22 最大靜摩擦力實驗之力平衡圖

4.實驗數據與結果：

我們進行了三次實驗，分別將木板一端抬起 1.72、1.75、1.78 公分後，車子會向下滑動。

將三組數據取平均後，我們以 1.75cm 下去計算，因密集板長度為 60cm，由次可知傾斜角度為 1.67 度。

再從實驗原理的結果去推測，可發現滾動的最大靜摩擦係數為 0.02918。

參考文獻：

1. Ackermann steering Geometry (維基百科)
2. 經濟部 96 年度科技專案技術報告 -- 自行車暨衍生系統關鍵技術開發計畫
3. 三輪式無人載具低速時第三輪轉向匹配控制之研究 (中興大學論文)
4. 四輪驅動車輛之軌跡誤差自動修正 (交通大學論文)

肆、風扇與動力

4.1 設計前的評估

因此項工程的要求動力來源為風力，所以風扇成為本次 project 中不可或缺的一項機構。本組由直流無刷馬達趨動我們設計的風扇，藉以獲得循跡車推進的動力。由於我們期中測試的車體偏大且偏重的緣故，我們必須要有足夠的推進力來克服靜摩擦力等阻力才得以讓車子前進。因此，我們希望可以讓風扇能以"高轉速、大尺寸、大攻角、最佳效率"的原則進行設計和製作。

除此之外，測試當天的一些基本參數也是要先想過的。由於期中測試的日期是在 5/4，可以稍微猜測當天的氣溫約莫在 25°C 左右，而測試地點在永齡生醫館的 124 室室內，假設在有開空調的狀況下室溫約為 23°C。得知溫度後就可以推得其他重要的參數:

(1)空氣黏滯度 $\mu = 18.5 \mu \text{ Pa}\cdot\text{s}$

(2)空氣密度 $\rho = 26.889 \text{ kg/m}^3$

(3)聲音速度 $V = 344.8 \text{ m/s}$

由上述的環境參數，代入分析軟體後，即可得到流場等等分析結果，由此可知風扇的製作成果及如何改善其運作方式。

4.2 風扇的數量與尺寸規格

首先，在風扇尺寸的方面因受限於車體大小的限制，包含風扇罩的寬度不能超過 210mm，保守扣掉風扇罩的大小後，扇葉最大的大小半徑大約可以有八至九公分。製作過程中，我們設計了許多不同大小、攻角、形狀、扇葉數目，一開始設計的小型風扇推力不足，最大原因應該是我們採用的無刷馬達轉速不高的緣故，造成負重後完全無法往前推進，最後我們直接把直徑加大到尺寸限制的邊緣去做加工，方解決推進力不足的問題。

在數量方面上，除了扇葉之外，另外一個就是風扇的數目，下面就簡略地比較各種風扇數目的優劣分析。

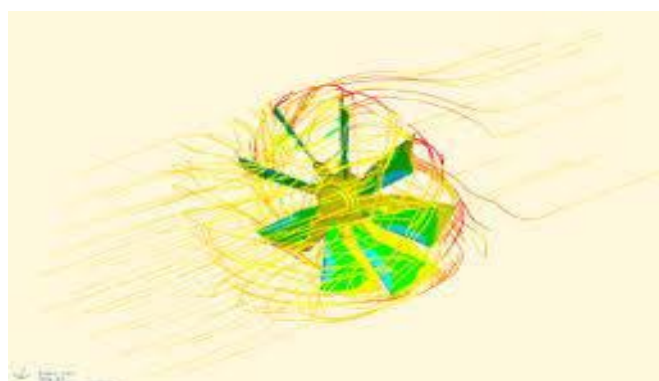
1.無扇葉：

原理來自無葉電風扇，基座中帶有電力馬達，並將空氣吸入基座內部，經由氣旋加速器加速後，即可增加空氣流通速度。再經由無葉風扇扇頭環形內唇環繞，其環繞力帶動扇頭附近的空氣隨之進入扇頭，再將空氣吹出。雖然此項選擇十分安全但製作太過於複雜，且成本過高，在往後的流場分析也更是難以理解和分析，故不予以考慮。



2.單風扇：

此種為大眾選項，因其重量輕，製作較簡易，流場也較好分析，所以成為了大家的首選。此外，更能節省空間配置，讓單一風扇可以更大，推力也更強，馬達只需要一個，耗電量也較少，缺點為可能讓流體偏向其中一方，使車體不穩定，但根據測試經驗，只要車體速度不要過快，其實影響甚小，可以忽略。



3.雙風扇：

這種選擇對於車體往前時的穩定幫助較大，但前提也是左右風扇的流量趨近一致，此外的缺點更是許多，如推力、馬達、耗電量等等，皆是較沒效率的

製作方式，另外設計上也會出現許多困難，所以即使有方向穩定的優勢也還是讓大家不予採用。

4.3 扇葉數量與擺設

至於規格方面，最明顯的外觀變化就是葉片數目，而扇葉數目選擇上也是一門重要的課題。

如數量太多的話，會導致葉片間距太小，讓氣流擾動增加，造成的磨擦會使得扇片的效率逐漸低落，也就是多餘的葉片會干擾推進的效果。反之，數量太少的話，會導致間距過大造成風壓不足，或是壓力損失增加的問題等等。

再者，參考了網路上的文獻資料以及學長姐報告，可以得知偶數個葉片可能在轉動時造成共振，影響扇葉的使用壽命。相反地，若採用奇數葉片，則會因為機構的對稱關係，就無作用力不平均的問題，在使用上較無顧慮。而因多葉片有推力高的優點，少葉片也有風阻低不易斷裂的優勢，所以本組則先以三葉及五葉的扇葉為藍圖加工製作及實驗分析。

而扇葉的擺設方面，大致可分為前置、中置、後置的三個選項，以下以表格做簡易的比較與分析。

	優點	缺點	改進方向
前置	快速地行進時穩定性較佳。	前輪負荷過重，會造成我們前輪轉向機構不夠即時，也會造成重心不穩。	需藉由其他元件去作配重，但相對地會讓車體負荷更重。
中置	重量分布較為平均，轉彎的角度較為精確。	相較之下轉彎也會過度，且風力也會因為後面擺置的東西多少造成影響。	轉彎過度的問題必須以減速的方式去克服。
後置	風力不容易受到後面擺放設置的干擾，且因慣性的關係，減速時較為平穩。	但另一方面，慣性關係也造成過彎無法加速，否則會造成嚴重的偏差。	因後置慣性問題，減速更須達到一個特定值方能順利通過彎道。

最後，考量技術層面的問題，我們將扇葉放置在車體的尾部，因主要將順利通過彎道以及行進時的穩定性為第一目標。

4.4 扇葉的設計與製作

扇葉的設計上有許多方式可以達成，而不同的方式對我們其實影響並不大。最主要的原因在於，我們身為初學者通常都是採用 NACA 所提供的扇葉模型，而我們在進行了初步的估算以及數據整理以後，也是挑選最接近的 NACA 扇葉模型來匯入繪圖軟體，因此可以想見，我們的扇葉或許會因為選擇不同的繪圖軟體而有不同的製作方式，但是本質上都是源自於 NACA 的模型，所以差異應該不大。

NACA 翼型後面皆有四位數字，而其分別代表三種參數，首位數字代表最大彎度占弦長的百分比，第二位數字代表最大彎度距機翼前緣的距離占弦長的十分之幾，而最後兩位數字代表機翼最大厚度占弦長的百分比。其中形狀方程式為下列所示：

$$y_t = 5tc \left[0.2969 \sqrt{\frac{x}{c}} + (-0.1260) \left(\frac{x}{c}\right) + (-0.3516) \left(\frac{x}{c}\right)^2 + 0.2843 \left(\frac{x}{c}\right)^3 + (-0.1015) \left(\frac{x}{c}\right)^4 \right]$$

其中 c 表示弦長、 x 表示沿弦長的距離（從 0 到 c ）、 y_t 表示 x 處厚度的一半、 t 表示最大厚度占弦長的比例（ $100t$ 即 NACA 四位數翼型的後兩位數字）。

另外，NACA 也有五位數翼型，但此通常表示較複雜的翼型，本次專題就先省略，而採用四位數的即可。[2]

我們採取的方式是先依照扇葉截面積大小，得出 chord length 等重要參數，並在 NACA 4 digit airfoil generator 這個網頁輸入對應的數值。接著透過公式可以得到理想最大攻角，在這個網頁中找取最接近的扇葉模型，在這邊我們選擇的是 NACA(XXXX)。選好模型之後，點選 send to airfoil plotter 以及 CSV file of coordinates，就會得到一個充滿點的一份 excel 檔，將其另存新檔為.txt 檔，並手動除去不必要的空白和加上逗號。

完成以上步驟之後就可以透過(scr)指令將其匯入 auto cad 中，存檔並打開 creo 叫出我們剛剛存的檔案，並使用多層平面上草繪後再混成的的技巧，就可以得到我們扇葉的雛型。

接著就是來到如何製造的問題。一般來說，金屬製的扇葉無庸置疑一定較為堅固耐用，但是由於我們的扇葉需要大的攻角且攻角的角度並不固定，金屬扇葉的堅固與不易變形的特性反而會造成加工製造時的困擾。因此，我們選用實作中心免費提供使用的 3D 列印機台來進行製造。3D 列印製作出來的扇葉，強度肯定不如金屬製的，但是他的優勢就在於方便、可變化性高且可以較便宜的價格(\$0)取得。

實作中心的 3D 列印機台主要有 UPBOX、KINGSSEL 還有 KRAFTMAKER 這三種，UPBOX 要校正蠻久但是泛用、KINGSSEL 操作較簡單但是有尺寸上的限制、KRAFTMAKER 算是裡頭較為直觀且尺寸可以做比較大的，剛好可以容下我們的大扇葉，所以我們扇葉多用 KRAFTMAKER 製作。

4.5 扇葉的模擬分析

為了選出最適合的扇葉模型，一般對於扇葉的分析皆是從風場去做模擬，但其實實際上的表現卻是十分複雜，為了可以預測流體的狀態，大家選用的軟體幾乎都是 ansys 系統的 workbench 去進行流體分析，其中也包含了大二下必修流體力學的知識，使其學以致用。但本組在 ansys 的使用方法上出了些許困難，在下列的使用步驟中會加以說明。

1. 首先把 creo 的風扇圖檔匯出成 IGES 檔
2. 點選 Ansys 左側工作列的 Fluid Flow(Fluent)，要讓六項工作的右側都成為打勾的形式
3. Geometry:建立一個長方體作為流場區域，限制範圍的設定。創造出一個 Boolean 功能中的 Subtract，產生出風旋轉的效果。
4. Mesh:流場的幾何模型建立完成後，此步驟開始要建立網格去分別計算各節點上的方程式。
5. Setup:網格方面建立完成後，就開始交由 Fluent 進行計算，在此需要許多邊界數值的設定，電腦運作的時間也主要是在因為這個步驟。
6. Solution & Results:最後的答案及結果在此呈現，像流線分布、流體向量的分布、和預測及模擬流體的動畫徒都可以在這裡進行呈現。報告中，大家採用的不外乎的就是流線以及速度場分布，再從多方比較中去推算出最適合車體的扇葉。

本組在其中的建立長方體流場中，疑似 creo 的圖檔有破裂面或是混成時有疏失，造成在 edit geometry 時不斷 generate 失敗，多方嘗試下仍然無功而返，也成為需檢討的改進點。

4.6 各代扇葉比較

第一代扇葉:

初代扇葉採用的是 5030 的螺旋槳扇葉，這四位數中前兩位數代表的是槳徑，後面兩位數代表的是槳距。

但因為我們的車體規格採用雙層的設計，且尺寸規格也皆逼近 A4 紙張大小，所以整體上的重量可能較重，造成推進力只約等於最大靜摩擦力，結果就是起步時時而成功時而失敗，這種現象更遑論說在測試時還有負重一個 250ml 的鋁箔包，所以我們趕緊再重新設計風扇，除了改變直徑大小外，葉片數目也成為我們著手改進的項目之一。



(側視圖)



(俯視圖)

第二代扇葉：

第二代的扇葉延續第一代 5030 槳的設定，加大了最大攻角與攻角變化率，視覺上看起來比第一代更為鋒利，同時也嘗試了加入更多的扇葉(3-->5)，企圖增加風扇的推進力。

而實際上跑起來卻與想像中的頗為不同，車子的速度不但沒有更快，反而感覺更慢了，原因應該是因為我們的風扇原本第一代的時候採取的最大攻角等參數可能已經接近 5030 槳的最佳參數，導致這次加大攻角反而沒有讓扇葉掃過

的流場給予更多的推進力，也產生了更多的紊流，所以造成推進效率的降低。



(側視圖)



(俯視圖)

第三代扇葉：

從這代開始我們開始使用了 8050 槳的扇葉，增加了扇葉槳徑，也讓風力大大地增加，車車的動力大大提升，馬達也只需要大約最大值一半的轉速即可在負載鋁箔包的狀況下達到我們需要的速度，經過測試過後我們決定之後所設計的扇葉，皆會從此種規格再去做微調及改良。此種翼型是採用 NACA 6409 去做規格設計，在扇葉厚度部分外觀明顯較為細小，一開始我們有擔心過在運轉時會破裂的問題，但幸好最後測試時大致都安然無事。



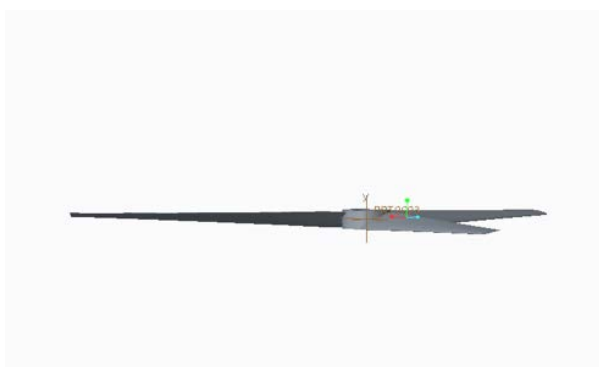
(側視圖)



(俯視圖)

第四代扇葉：

四代和五代的扇葉改採用 NACA 2412 的翼型去做繪圖及加工製作，規格方面仍採用 8050 槳，此外，後面三代的弦長(chord)也都是以相同長度(110mm)繪圖，效果與第三代並相去不遠，推力方面都是十分充足，循跡時速度亦不能過快，否則容易衝出線外，之後我們大概皆以此種模板去做測試。



(側視圖)



(俯視圖)

第五代扇葉：

其實第五代扇葉與第四代扇葉的規格十分接近，還是延續著 8050 槳的規格，但是在靠近扇葉 1/4 的部分截面積有改大，且把攻角再調大，會想要這麼調整主要是因為三四兩代的扇葉中間過為單薄，怕這麼大的槳徑容易在高速的轉速下破裂或是斷掉。除此之外，由於加大截面積的部分，可以使我們的扇葉推進能力進一步增強，成為了我們循跡車最終的版本。



(側視圖)



(俯視圖)

4.7 風洞實驗結果分析與比較

實驗一:

阻力實驗 (Drag on a model car from control volume)

1. 實驗目的: 測量在不同風速下, 車子在流場中所受到的阻力

2. 實驗器材:

- (1) 模型車
- (2) 系上自製風洞
- (3) 熱線風速儀

3. 實驗原理:

(1) control volume 的受力分析求阻力

假設在測試區域的流場是一維的, 代表與流場不平行的速度分量極小可以忽略, 此條件假設可以幫助我們進一步推測 control volume 上表面的氣流動量可以忽略不計。因此, 透過以上假設可以將 X 方向的 steady momentum equation 簡化求得 單位寬度的阻力(drag)D。

$$F_s = -D = \rho U U_{in} ds = \rho U U_{in} dy - \rho U D U_{out}(y) dy + \rho U(x) V(x) dx$$

其中, ρ 代表空氣的密度, U 代表 CV 靠近車子前緣的地方的風速, $U D$ 代表 control volume 靠近車子後緣(有風扇處)的地方的風速, $U(x)$ 、 $V(x)$ 代表著與氣流垂直的分量, 所以計算時根據之前的假設, $U(x)V(x)dx$ 這項會為 0。

4. 實驗步驟:

- (1) 測量車子的長度(L_{car})、寬度(W_{car})、高度(H_{car}), 並算出與流場垂直的截面積 $A_{car} = W_{car} H_{car}$ 。
- (2) 將車子放入風洞中, 盡量對齊中線。關上門測量兩邊門縫的距離。
- (3) 用熱線風速儀量室溫。
- (4) 將風洞開至 30% DUTY, 等抽氣風扇的噪音出現。
- (5) 將熱線風速儀從門縫中伸到車子中線處, 有孔面面向流場。
- (6) 熱線風速儀從靠近車子前緣的門縫伸入, 測量不同高度的 U (20 秒的平均) 並記錄。
- (7) 熱線風速儀從靠近車子前緣的門縫伸入, 測量不同高度的 $U D$ (20 秒的平均) 並記錄。
- (8) 分別調整風洞 DUTY 至 45% 與 60%, 並各重複一次步驟 5 和 6。

5. 實驗結果:

T=25°C	DUTY=30%		DUTY=45%		DUTY=60%	
yi	UU(m/s)	UD(m/s)	UU(m/s)	UD(m/s)	UU(m/s)	UD(m/s)
28cm	1.16	0.13	1.07	0.03	2.53	1.17
24cm	1.13	0.13	1.2	0.82	2.66	0.75
20cm	1.17	0.87	1.2	0.86	2.60	0.84
16cm	0.72	0.06	1.21	1.07	2.68	0.56
12cm	0.67	0.02	1.13	0.07	2.56	0.15
8cm	0.65	0.07	0.9	0.53	2.46	0.69
4cm	0.64	0.21	1.06	0.61	1.97	0.66

DUTY=30% UU=0.66m/s;UD=0.73m/s;

DUTY=45% UU=0.91m/s;UD=0.93m/s;

DUTY=60% UU=1.22m/s;UD=1.39m/s;

經由上表可以得知，D(duty=30%)=3.599(N/m);

D(duty=45%)=4.079(N/m);D(duty=60%)=17.726(N/m)。

實驗二:

風扇流場的視覺化與量測(Visualization and measurement of fan flows)

1.實驗目的:觀測風扇擾動流場的流線以及其分布，並用 Actuator Disk Theory 來分析。

2.實驗器材:

- (1)模型車(含風扇)
- (2)系上自製風洞
- (3)可攜式熱線風速儀
- (4)尺
- (5)超音速煙霧生成器及其導管
- (6)綠雷射片及反射鏡
- (7)轉速表

3.實驗原理

(1)Actuator Disk Theory

轉子被建模為無限薄的圓盤，沿著旋轉軸線產生恆定的速度。直升機的基本狀態正在徘徊。該光盤在轉子周圍產生流動。在流體的某些數學前提下，可以提取功率，轉子半徑，轉矩和感應速度之間的數學關係。摩擦不包括在內。

對於靜止的轉子，如懸停的直升機，產生給定推力所需的功率

$P=T^3/2A$ ，其中:T 是推進力、是介質密度(這邊是空氣)、A 是轉盤的面積(也就是扇葉繞軸旋轉所掃過的圓面積)[4]

又 $P=T*V$ ， $T=4V^2A^2$

4.實驗步驟

- (1)關燈。
- (2)打開煙霧機的風扇(其他別開)，用熱線風速儀測量 brush tube 的兩個 branch 處的流速。
- (3)啟動煙霧機，觀察流場中的煙霧。
- (4)開啟雷射，調整鏡子直到雷射片剛好通過煙霧流。
- (5)打開風洞的風扇，調整 duty 至 10-20%，讓煙霧流可以清楚觀測。
- (6)量測背景流速(U)。

(7)啟動模型車的風扇到 MODE 1，調整 brush tube 讓煙霧流通過車子風扇，觀察流線受到風扇的影響。試找出煙霧流的喉部(最細的部分)，記為 A3，而流出 brush tube 的地方為 A1。

(8)拍照，將 A1 與 A3 高度的一半標記於側牆上，並測量其高度。

(9)用熱線風速儀分別量測 A1 截面上五個點(#1-#5)以及 A3 截面上三個點(#1-#3)的速度(15 秒平均)。

(10)把模型車風扇調到 MODE2，重複步驟(7)~(9)

(11)把模型車風扇調到 MODE3，重複步驟(7)~(9)

(12)用轉速表測量模型車風扇轉速，記得在風扇上貼反光膠帶。

5.實驗結果

DUTY= % , background velocity=

#2(24cm)=0.31m/s , #4(16cm)=1.14m/s

尚未扣除環境流速的狀況下

	Fan mode 1 Propeller 0.8A		Fan mode 2 Propeller 1.1A		Fan mode 3 Propeller 1.4A	
	U1 (m/s)	U3 (m/s)	U1 (m/s)	U3 (m/s)	U1 (m/s)	U3 (m/s)
Height						
#1	1.5	6.29	1.4	6.39	1.7	6.61
#2	1.7	6.28	1.8	6.38	1.9	6.48
#3	1.6	6.07	1.2	5.97	1.2	6.36
#4	1.1		1.0		1.1	
#5	1.0		1.2		0.7	
Mean (m/s)	1.4	6.21	1.3	6.25	1.3	6.48
Throat radius(cm)	7.03		5.88		5.35	

根據實驗原理， $T=A1U1(U3-U1)$

其中， $V1$ 是未減去背景流速的進口流速， $=1.214\text{kg/m}^3$;

$T1=0.164\text{N}; T2=0.157\text{N}; T3=0.164\text{N}$

4.8 小結與改進方向

風扇是我們這台車的主要動力，從一開始的天馬行空的討論他的尺寸、數量、位置，經過參考去年學長姐的經驗以及我們動手做出來許多失敗的 prototype，我們將風扇的期中測試版做了出來。而期中測試版的大推力、高轉速不僅彌補了我們車體偏大偏重的缺點，更讓我們在調整速度程式碼的時候可以有更大的容錯和嘗試的空間，讓我們不需要額外為了紅外線 sensor 以外的東西而擔心，可以算是相當不錯。

而期末我們的車子面臨到更多的考驗，與風扇最相關的可能是如何提高推進能力來讓車子突破隆起的障礙物。尤其是當車子有可能會因為接觸到較粗糙的障礙物表面而過度減速卡在原地動彈不得，讓風扇給予車子更大的動力將會是我們接下來必須努力改善的目標。

但速度要能再過 bumper 前成功加速，並且要在進彎前給予減速，否則會讓提高紅外線高度後造成的不穩定，又因為太快造成無法循跡。但測試前五分鐘又因為轉斷了風扇，造成參數無法及時調整，讓過彎循跡穩定性大幅下降，成為我們期末測試較為不理想的原因。

突然會轉斷風扇的原因有很多，可能是因為 3D 列印機調的密度參數等等問題，又或者是列印機本身品質不穩定，也有可能是我們打磨時有不當的施力，

4.9 參考資料來源：

[1]https://www.google.com.tw/search?q=%E7%84%A1%E6%89%87%E8%91%89%E9%A2%A8%E6%89%87&rlz=1C1ASUT_enTW606TW606&source=lnms&tbm=isch&sa=X&ved=0ahUKEwimq9jRrNTbAhXIozQKHe7sDtQQ_AUICigB#imgrc=STno6QhsO6yTE M:

[2]<https://zh.wikipedia.org/wiki/NACA%E7%BF%BC%E5%9E%8B>

五、感測系統與控制邏輯

5.1 紅外線循跡：

紅外線循跡原理是利用紅外線感測器，因應反射面的反射程度，來辨識「黑線」以及「黑線以外的地方」，並回授給控制中心，經過計算後，發送命令給馬達進行轉向。

感測器訊號回授部分，可以分為類比訊號以及數位訊號兩種方式，類比訊號能因應反射面的反射程度，回傳連續變化量的值；而數位訊號，則因應使用者調整可變電阻來設定閾值，來使其回傳 0 或 1 的訊號來表示反射面反射程度強或是弱。

兩者在程式設計上會有很大的差異，類比回授的訊號優點是可以感測到壓到黑線的「程度」，缺點是容易因為地面顏色不同、訊號受到干擾而產生誤差以為有壓到一點黑線。而數位訊號優點為能很明確知道是否壓到黑線，缺點為因為二分法的方式，容易讓車體來不及反應已經壓到黑線了。

TCRT5000 紅外線反射模組(我們所採用的模組)



(一)功能介紹：

TCRT5000 感測器的紅外發射二極體不斷發射紅外線，當發射出的紅外線沒有被反射回來或被反射回來但強度不夠大時，紅外接收管一直處於關斷狀態，此時模組的輸出端為高電平，指示二極體一直處於熄滅狀態；被檢測物體出現在檢測範圍內時，紅外線被反射回來且強度足夠大，紅外接收管飽和，此時模組的輸出端為低電平，指示二極體被點亮。

(二)產品用途：

- 1、電度錶脈衝資料採樣
- 2、傳真機碎紙機紙張檢測
- 3、障礙檢測
- 4、黑白線檢測

(三)產品介紹：

- 1、採用 TCRT5000 紅外反射感測器
- 2、檢測反射距離：1mm~25mm 適用
- 3、比較器輸出，信號乾淨，波形好，驅動能力強，超過 15mA。
- 4、配電位器調節靈敏度
- 5、工作電壓 3.3V-5V
- 6、輸出形式：數位開關量輸出（0 和 1）
- 7、設有固定螺栓孔，方便安裝
- 8、小板 PCB 尺寸：3.2cm x 1.4cm
- 9、使用寬電壓 LM393 比較器

5.2 超音波避障：



超音波感測器是由超音波發射器、接收器和控制電路所組成。其原理是發出人耳聽不見的高頻超音波，測量方式也是雷同，利用反射時間與音速來算出距離。超音波受周圍環境影響大，一般來說測量距離比較短，精度較低。

超音波感測器所發出的音波範圍約為平面 60°扇形，但在邊緣處能量較不集中、會影響距離的測量，且載具主要的避障需要集中於左前方，所以我們就把此裝置擺在左前側車體並將其擺向正前方，以達到最佳的避障效果。

超聲波測距感測器的優點:

- (1) 超聲波對色彩和光照度不敏感。可用於識別透明漫反射性差的物體(如玻璃、拋光體)。
- (2) 超聲波對外界光線和電磁場不敏感。可用於黑暗、有灰塵或煙霧、電磁干擾強、有毒等惡劣環境中。
- (3) 超聲波感測器結構簡單、體積小、費用低、技術難度小、資訊處理簡單可靠、易於小型化和整合化。

5.3 Arduino uno 微控制器



Arduino 提供了簡單好用的 I/O 介面，並具可使用類似於 Java、C 語言等高階語言的開發環境。

微控制器	ATMEGA328
工作電壓	5V
輸入電壓（推薦）	7-12V
輸入電壓（限制）	6-20V
數字 I/O 接腳	14（6 個提供 PWM 輸出）
模擬輸入接腳	6 支
EEPROM	1 KB（ATMEGA328）
震盪速度	16 MHz

硬體功能說明：

數位 I/O 接腳：

14 支數位 I/O 接腳 可以當作 input 使用，也可以當作 output 使用，使用方法是透過 `pinMode()`, `digitalWrite()`, and `digitalRead()` 這幾個函式。這 14 支數位 I/O 接腳，其中幾支腳有特殊的功能：

Serial 通訊	0(RX) 和 1 (TX) 這兩支腳。用來接收 (RX)與傳輸(TX) TTL 訊號的序列資料。這兩支腳也連接到 USB Converter 晶片中。
外部中斷	2 和 3 這兩支腳。這兩支腳可以利用外部事件觸發中斷。詳細內容請參考 attachInterrupt() 函式。
PWM	3, 5, 6, 9, 10 和 11 共六支腳。透過 analogWrite() 函式可以提供 8-bit 的 PWM 輸出。
SPI	10 (SS), 11 (MOSI), 12 (MISO) 和 13 (SCK) 這四支腳。這四支腳搭配 SPI Library 可提供 SPI 序列通訊。
LED	13。內建一顆 LED，當 pin 腳為 HIGH 時，LED 打開，當 pin 腳為 LOW 時，LED 關閉。

5.3 機電系統：

如下圖所示，為我們這組的機電系統，Arduino 的電源為使用 9V 電池經由 DC plug 進行獨立供電，感測器（包含紅外線感測器、超聲波感測器）及伺服馬達皆由電變的 5V 角位進行供電，而電變的電源由 3s 鋰聚電池供電。電路上地線皆連接在一起，以確保訊號傳遞穩定，且有相同 GND 電位。

除之外，由於紅外線 sensor 總共有五個，每個都需要 5V 的電源輸入，而且紅外線位置距離其他元件較遠，因此我們自製簡易的電路，將每個紅外線 sensor 的 Vcc 以及 GND 在電路板上匯流之後接到另一塊 Vcc 及 GND 的電路板，訊號現的部分則直接以一排杜邦線接至 Arduino。這樣一來，線路將變得較整潔，要臨時更動，或是檢查都相當容易。

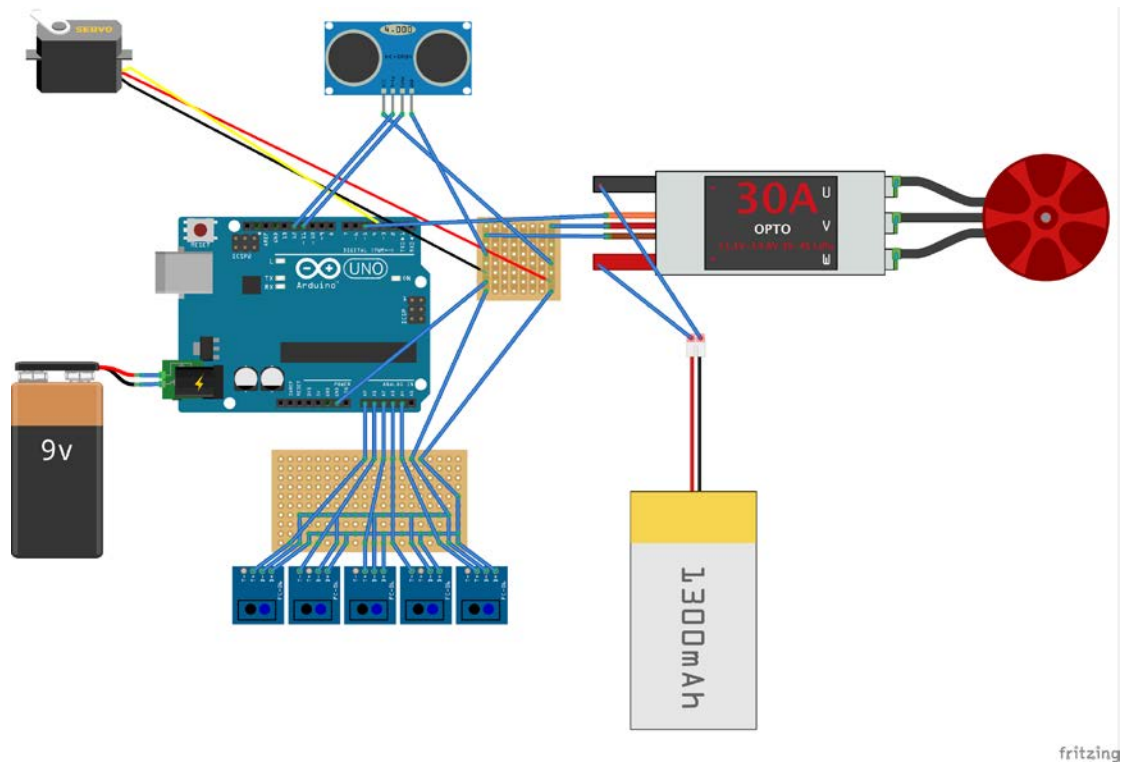


圖: 紅外線循跡感測器配線圖

5.4 控制邏輯：

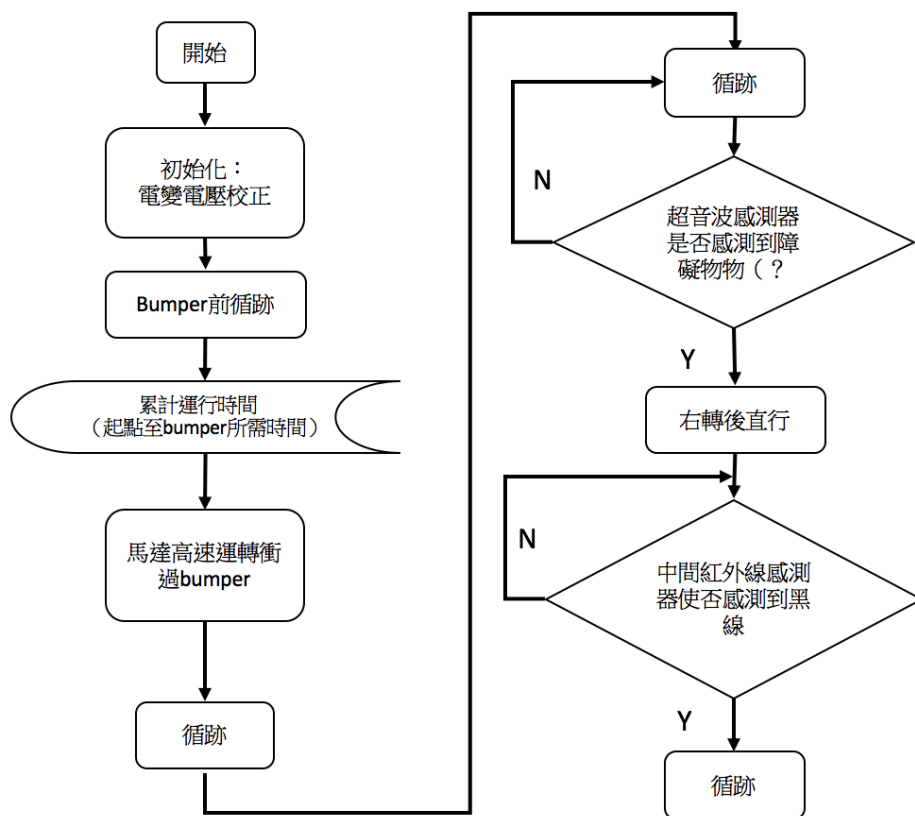
控制上如下圖所示，為控制系統架構圖，在循跡開始後，因為電變必須在使用前進行校正，校正方式為將 PWM 輸出調至最高，待約一秒後，在將其調至最低，以進行訊號上下限校正。完成後將進入第一部分，bumper 前的循跡，循跡邏輯控制將在下面細講，循跡經過一個累計的運行時間後（實驗得出），車進行衝刺越過 bumper，完成第一部分。由於第一部分在整個測試過程僅需執行一遍，因此在 Arduino 城市當中，放在 void setup 裡執行。

完成第一部分越過 bumper 後，將進入第二部分彎道的循跡，與期中測試相同，我們利用 PID 控制，調整其 K_p ， K_i ， K_d 值使在氣動車越過 bumper 後能快速穩定下來，延著黑線循跡。

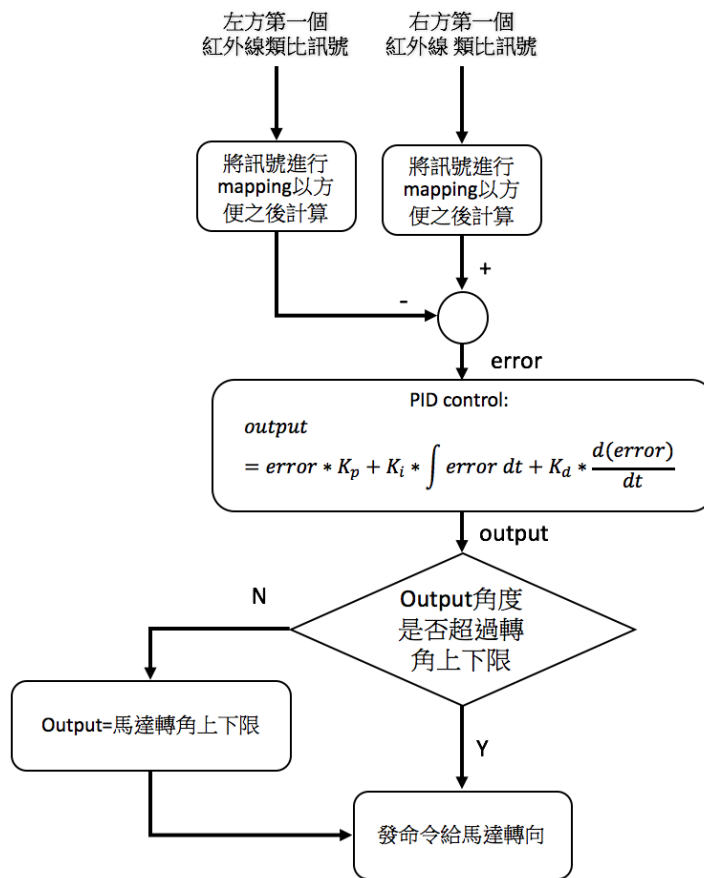
完成第二部分之後，來到第三部分的避障，由於障礙物的位置在測試前已知，因此我們採用簡單的方式進行避障，在超音波測距儀感測到障礙物在一定範圍內時，向右轉彎在轉正繼續直走，直到正中央的紅外線 sensor 再次巡好黑

線才回到循跡程式，這部分要經過多次測試才知道，車子在遇到障礙物應該轉多少，才能在規定範圍內抵達另一條黑線。

第四部分與第二部分相同，只是難度提升，考驗氣動車控制程式系統的穩定度，因此在 K_p ， K_i ， K_d 上的調校必須花更多時間，使其在循虛線時能穩定。



PID 控制邏輯在循跡的過程非常重要，其控制方法使用傳統 PID 控制器。由於紅外線訊號回傳的類比訊號值，代表紅外線感測到黑線的比例，因此可以利用這個連續變化的直進行追蹤，因此適合使用 PID 控制器。我們將循跡線兩側上方的紅外線感測器所得到的類比訊號 mapping 後進行相減，以將車子偏移黑線的程度量化，經由 PID 控制器，將需轉的角度輸出給馬達。



陸、工作項目與分工

6.1 期中以前甘特圖

	3月28日	4月2日	4月7日	4月11日	4月18日	期中考週 4月23-27日	期中驗收 5月2日	期中書面報告 5月4日	期中書面報告 5月10日
採買確定	採買確定	風扇、轉向初步	完成設計	程式、電路	完成組裝		完成測試		
設計									
風扇設計：決定規格									
風扇設計：分析比較									
風扇設計：與車體連接									
轉向設計：決定規格									
轉向設計：與車體連接									
車體設計									
機電整合：感測器閱讀									
機電整合：程式閱讀									
機電整合：電路設計									
機電整合：程式撰寫									
加工組裝									
風扇繪圖									
轉向繪圖									
車體繪圖									
製造									
電路配置									
組裝									
實驗									
轉向實驗									
風扇實驗									
期中測試、報告整理									
期中測試、報告整理									

6.2 BOM 表

BOM 表			
項目	單價	數目	總價
WH50X17-1D 定向輪	55	3	165
1-8S 鋰電池低壓警報器	50	1	50
新西達 A2212 1400KV	180	1	180
馬達電變延長線	30	3	90
40A 電變	500	1	500
3S 鋰聚電池 2250mAh	580	1	580
MG995 伺服馬達	130	1	130
MB-102 麵包版	65	1	65
Arduino mega	120	1	250
TCRT5000 紅外線模組	41	6	246
杜邦線公母頭	20	1	20
杜邦線公公頭	20	1	20
電路板	25	1	25
排針一排	12	1	12
滑動開關	2	1	2
3mm 密集板	250	(用不到一片)	100
Total			\$2435

6.3 分工表

設計	
風扇設計：決定規格	董逸棋、黃柏瀚、黃士瑜
風扇設計：分析比較	董逸棋、黃柏瀚、黃士瑜
風扇設計：與車體連接	董逸棋
轉向設計：決定規格	黃柏瀚、徐敏懷
轉向設計：與車體連接	黃柏瀚、徐敏懷
車體設計	黃柏瀚、徐敏懷、董逸棋
機電整合：感測器閱讀	董逸棋、黃柏瀚、黃士瑜、徐敏懷
機電整合：程式閱讀	董逸棋、黃柏瀚、黃士瑜、徐敏懷
機電整合：電路設計	董逸棋
機電整合：程式撰寫	董逸棋、徐敏懷
加工組裝	
風扇繪圖	董逸棋、黃柏瀚、黃士瑜
轉向繪圖	黃柏瀚、徐敏懷、董逸棋
車體繪圖	黃柏瀚、徐敏懷、董逸棋
製造	董逸棋、黃柏瀚、黃士瑜、徐敏懷
電路配置	董逸棋、徐敏懷
組裝	董逸棋、黃柏瀚、黃士瑜、徐敏懷
實驗	
轉向實驗	徐敏懷
風扇實驗	董逸棋、黃柏瀚、黃士瑜、徐敏懷
實驗整理	黃柏瀚、黃士瑜
期中期末，測試、報告整理	
期中期末測試、報告整理	董逸棋、黃柏瀚、黃士瑜、徐敏懷

柒、心得與建議：

徐敏懷:

記得剛上大二時，助教曾說“機械系以我過來人的經驗談，大二下課業最累，大三上最難”。想一想，轉眼間我已順利地度過了最累的大二下及最難的大三上了。我想助教之前應是提醒我要留意大二下及大三上的功課。那大三下呢？再兩周就要結束了，又給我留下了甚麼印象呢？

我的感覺是壓力最大的一學期，壓力大到胃痛，甚至做車體鑽孔鑽到破皮流血。記得學期剛開始的第一堂課，詹教授說希望你們最好找不同專長的人組合在一起，分工合作、解決問題、發揮整組的最大效益。我主要分配到的工作及共同協助完成的部分如：車體與轉向系統、系統組裝、期末競速第一版程式開發及競速第一版線路接線、協助工程圖繪製等。真是忙到手忙腳亂，不可開交。

可惜，測試時，臨時狀況屢屢出現，檢討主要原因除了運氣不好之外，時間緊迫匆忙完成，往往來不及做最後的修改。這也印證了詹教授期初的提醒：測試怕臨時車子有狀況，所以報告的比重會拉高，請大家好好把報告撰寫完成。這也是我傾全力專注於期中、海報、期末報告的撰寫的原因，希望留下一個被放置在 Capstone course 上的回憶。

黃士瑜：

這兩學分的課真的不只兩學分的 loading，花的時間真的比想像的還要多上很多，也真的學到了很多不同方面的東西，大致來說還是賺，除了學分、GPA 太少以外。儘管我們測試成績因為一些運氣不好還有臨時狀況發生而不盡人意，但過程中我們也的確有過許多次更好許多的結果，只能嘆息自己對於突發狀況的無能為力與不穩定性。

另外，不管在實作還是報告方面，都投入了大量時間在內，也成長了不少，真的是讓我對這門課有點又愛又恨，而且這門課還要花到自己的錢錢，希望系上的補助能夠多一點，對窮學生有所體諒和施捨 QQ。還有一個收穫是，讓我和很多機械系同學在永齡和長興一起吃了好多次消夜，又一同看到好多次日出，突然覺得日出都比日落漂亮了，漸漸地大家也習以為常。

最後也謝謝大家同心協力，各司其職，共同度過了這最重的必修課，雖然過程中可能偶有不愉快，但這些都是進步的動力，也辛苦大家的肝了。

黃柏瀚：

這學期終於來到尾聲，很慶幸學期初沒有因為看到學分數偏少就腦衝填一堆課，因為這門兩學分的課 loading 根本遠遠超過表面上的學分數阿!!

不但要經常和組員開會討論各種設計，還得經常在實作中心跟宿舍、教室、工綜地下室等等地方來回奔波，假日也不得安寧，進度上也是十分緊迫，在身心各方面都有著十足的壓力。

但是人沒有面臨到壓力和競爭就不會進步，這學期下來我真的學習到了不少。為了這台車子，我不但得回去翻流力、機動學等等已經放置一陣子的課本，還得額外上網、上圖書館自己學習關於風扇、ardunio 控制等等課堂上著墨較少甚至沒有教過的知識。除此之外，學習與人合作也是十分重要的。一直以來我都認為我是一個很好配合的組員，所以當看到別人的合作模式其實並不如自己想像的那樣時，剛開始也會覺得很生氣，直到前陣子聽到一位大師演講說道:「事情完成之後，要黑特要開地獄列車甚麼的再說。」想想也覺得還蠻有道理的，所以就算接近期末測試的時候組內有些不愉快，我們還是要把作品完成，所以很高興那時大家放下歧見，一起完成了最後的成品。

雖然期中與期末的測試可能不如預期，但是相信經過這學期的考驗之後，我們都會變得更堅強、更有能力邁向下一個挑戰！

董逸棋：

機械工程實務課程一直都是系上出了名的 loading 跟學分數不成正比的課程，除了時數上超出許多之外，也考驗到各組個人能力，與時間安排。相信一起修這門課的同學，如果有盡力參與，一定會從中學到很多，也感同身受。

在期中測試前兩天，車子都如預期的完美跑完全程，但在當天測試卻不如預期最後一個彎遲遲無法通過，除了歸類於運氣不好以外，也對於車體的穩定性，及循跡程式的穩定度發出警訊。而在期末測試時，由於原本不知道 bumper 高度會高過我們期中測試時的 sensor 高度，因此很晚才開始進行車子的改裝，測試時間也相當進迫，因此期末測試未能如期成功。而在期末競賽，在永齡進行試車時，終於在那邊看到大家的笑容，此時大家沒有測試壓力，而是沈浸在遙控車上面，那個氣氛，可能是修這門課以來感受最好的吧。

總結來說，這門課對於我們來說還是小有壓力，在時間安排，工作分配上還有許多地方是可以改進的，但相信無論是組員，抑或是自己，在結束這門課後，對於爾後遇到的各種小組專題，會更加了解自己的定位，以及自己要補強的地方。

捌、附錄

8.1 程式碼

(期中循跡)

```
#include<Servo.h>
Servo myservo; //伺服馬達
Servo blmotor; //無刷馬達
Servo readymotor;
double angleservo=82.5;
int leftmax=726;
int leftmin=391;
int rightmax=683;
int rightmin=357;
int pulse=0;
void setup() {
  myservo.attach(2);
  blmotor.attach(3);
  readymotor.attach(4);
  Serial.begin(9600);
  pinMode(A0,INPUT); // 左1紅外線
  pinMode(A1,INPUT); // 左1紅外線
  pinMode(A2,INPUT); // 右紅外線
  pinMode(A3,INPUT); // 右紅外線
  |
  pinMode(A4,INPUT); // 左1紅外線
  pinMode(A5,INPUT); // 右紅外線

  myservo.write(angleservo);
  blmotor.writeMicroseconds(2100);
  delay(8000);
  blmotor.writeMicroseconds(1000);
  delay(8000);
  blmotor.writeMicroseconds(1100);
  delay(1000);
  blmotor.writeMicroseconds(1200);
  delay(1000);

  else{
    myservo.write(angleservo);
  }
}
else if (left3==1) {
  if (angleservo-10>42.5){
    myservo.write(50);
  }
  else{
    myservo.write(angleservo);
  }
}
else if (turn+angleservo>122.5 || turn+angleservo<42.5) {
  myservo.write(angleservo);
}
else{
  myservo.write(angleservo+turn) ;
}
delay(100);

blmotor.writeMicroseconds(1300);
delay(1000);
blmotor.writeMicroseconds(1400);
delay(1000);
blmotor.writeMicroseconds(1630);
delay(3000);
}

void loop() {
  readymotor.writeMicroseconds(1000);
  blmotor.writeMicroseconds(1580);
  int left3 = digitalRead(A0);
  int left2 = digitalRead(A1);
  int left = analogRead(A2);
  int right = analogRead(A3);
  int right3 = digitalRead(A5);
  int right2 = digitalRead(A4);
  left = map(left, leftmin, leftmax, 0, 10);
  right = map(right, rightmin, rightmax, 0, 10);
  int turn= right-left;
  if (left2==1) {
    if (angleservo-10>42.5) {
      myservo.write(50);
    }
    else {
      myservo.write(angleservo);
    }
  }
  else if (right2==1) {
    if (angleservo+10<122.5){
      myservo.write(120);
    }
  }
}
```

(期末循跡版)

```
//加入library
#include<Servo.h> //無刷馬達&伺服馬達
#include<NewPing.h> //超聲波感測<

//定義腳位
#define servo_pin 3
#define wireless_pin 5
#define echo_pin 11
#define trig_pin 12
#define left2_pin A4
#define left1_pin A3
#define mid_pin A2
#define right1_pin A1
#define right2_pin A0

//定義常數及名稱
Servo mg995;
Servo blmotor;
//irsensor
const int leftmax = 800;
const int leftmin = 485;
const int rightmax = 792;
const int rightmin = 512;
const int max_distance = 200;
const double kp = 2;
const double ki = 0;
const double kd = 0;
//sonar sensor
NewPing sonar(trig_pin, echo_pin, max_distance);
//obstacle distance
const int obs_dis = 35;
//obstacle turn angle & turn time
const int turn_time = 1500;
const int obs_turn = 30;
//g995初始角度
const double initialangle = 80;
//mg995現在角度
double angleservo = initialangle;
//fastswitch angle
const int fast_turn = 5;
//無刷馬達速度&切換時間間隔
const int high_speed = 1850;
const int low_speed = 1850;
const int switch_time = 30; //指得是幾倍的followtime
//bumper前需要時間&衝刺時間&衝刺速度
const int bumpertime = 80; //指得是幾倍的followtime
const int acctime = 1500;
const int accveo = 1850;
//循跡delaytime
const int followtime = 50;
//i d control lasterror & intergral
double error = 0;
double integral = 0;
double lasterror = 0;
//else
int j=0;
int k=switch_time;
double turn_angle = 0;
```

```

//設定&初始化code
void setup() {
  Serial.begin (9600);
  //設定馬達 & sensor腳位
  mg995.attach(servo_pin);
  blmotor.attach(wireless_pin);
  pinMode(left2_pin, INPUT);
  pinMode(left1_pin, INPUT);
  pinMode(mid_pin, INPUT);
  pinMode(right1_pin, INPUT);
  pinMode(right2_pin, INPUT);
  Serial.begin(9600);
  //初始加速
  mg995.write(initialangle);
  blmotor.writeMicroseconds(2100);
  delay(8000);
  blmotor.writeMicroseconds(1000);
  delay(8000);
  blmotor.writeMicroseconds(1100);
  delay(1000);
  blmotor.writeMicroseconds(high_speed);
  delay(2000);
  //bumper前循跡
  for (int i=0; i<=bumpertime ; i++){
    followline();
    Serial.println(angleservo);
  }
  blmotor.writeMicroseconds(accveo);
  delay(acctime);
  j=0;
  k=switch_time;
}

```

```

#include <SoftwareSerial.h>
#include <Servo.h>
//循跡
void followline(){
  blmotor.writeMicroseconds(low_speed);
  //sensor reading
  int left2 = digitalRead(left2_pin);
  int left1 = analogRead(left1_pin);
  int mid = digitalRead(mid_pin);
  int right1 = analogRead(right1_pin);
  int right2 = digitalRead(right2_pin);
  //sensor mapping
  error = right1-left1;
  //pidcontrol
  integral = integral*2/3 + error;
  double derivative = error-lasterror;
  lasterror = error;
  double turn_angle = kp*error + ki*integral + kd*derivative;
  //turn
  if (left2==1){
    if (angleservo-fast_turn>initialangle-40){
      angleservo=angleservo-fast_turn;
    }
  }
  else if (right2==1){
    if (angleservo+fast_turn<initialangle+40){
      angleservo=angleservo+fast_turn;
    }
  }
  else if (turn_angle+angleservo<initialangle+40 && turn_angle+angleservo>initialangle-40){
    angleservo=turn_angle+angleservo;
  }
  mg995.write(angleservo);
  delay(followtime);

```

>

(期末競速遙控程式碼第一版)

```
#define servo_pin 5
#define blmotor_pin 9

Servo mg995; //伺服馬達
Servo blmotor; //無刷馬達

SoftwareSerial BTSerial(12,13); // RX | TX
int initialangle = 85;
int rotate = initialangle;
int velocitychange = 0;
int anglechange = 5;
char inChar,a;

void setup()
{
  Serial.begin(9600); // 訊號傳送速率
  BTSerial.begin(9600); // HC-06 current bound rate (default 9600)
  mg995.attach(servo_pin);
  mg995.write(initialangle); //預設是直進
  blmotor.attach(blmotor_pin);
}
void loop()
{
  if (BTSerial.available())
  {
    //Serial.write(BTSerial.read());
    update(BTSerial.read()); // Keep reading from HC-06 and control it
  }
}
void update(int x)
{
  switch(x)
  {
  case 97: //a 右轉
    rotate = rotate + anglechange;
    mg995.write(rotate);
    break;
```

```
case 98: //b 左轉
rotate = rotate - anglechange;
mg995.write(rotate);
break;

case 104: //h 回到直線前進
rotate = initialangle;
mg995.write(rotate);
break;

case 99: //c 正方形
blmotor.writeMicroseconds(1600+velocitychange);
break;

case 100: //d 三角形
blmotor.writeMicroseconds(1400+velocitychange);
break;

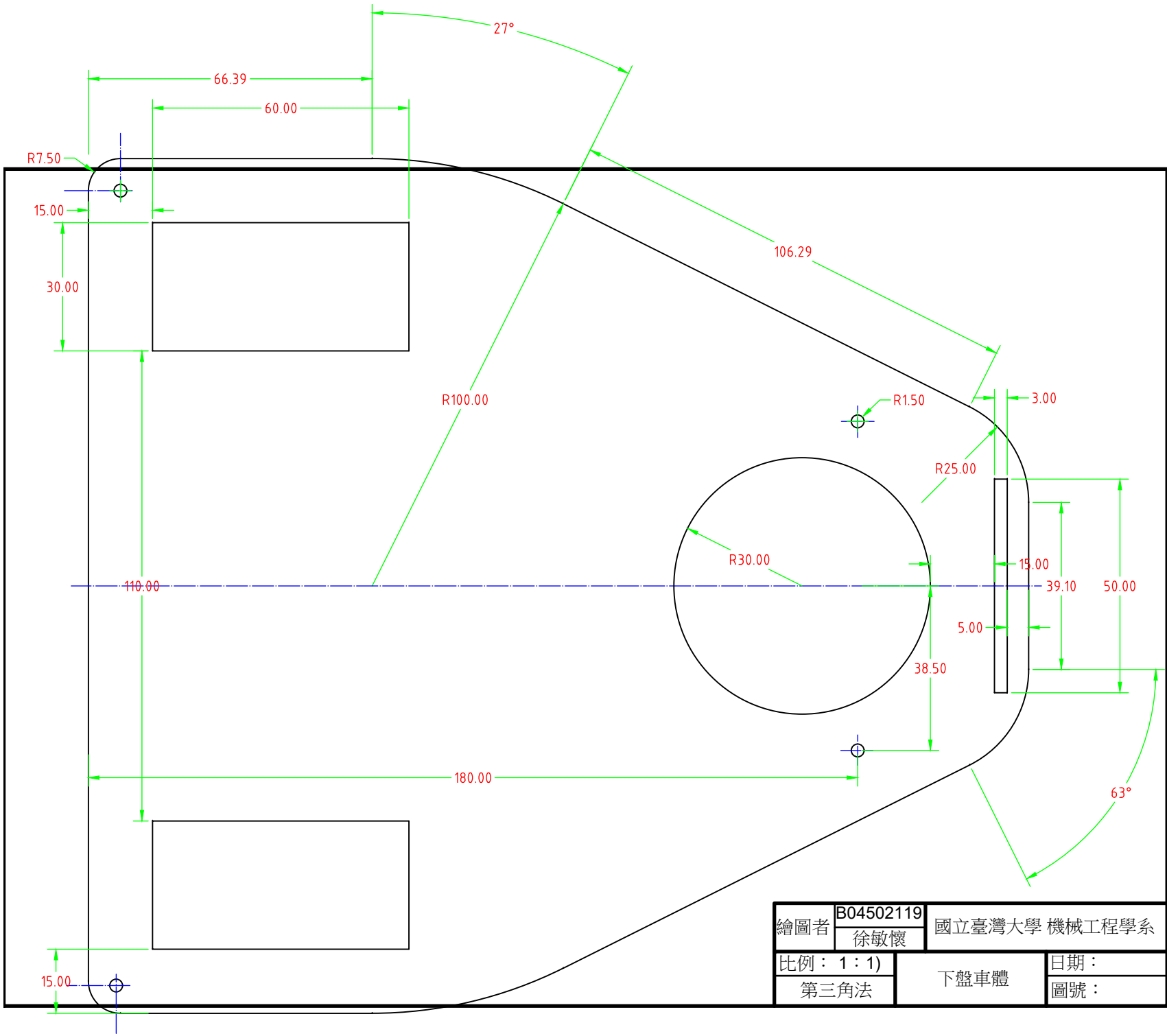
case 103: //g 圓形
blmotor.writeMicroseconds(1800+velocitychange);
break;

case 101: //e 停止 叉叉
blmotor.writeMicroseconds(1130);
break;

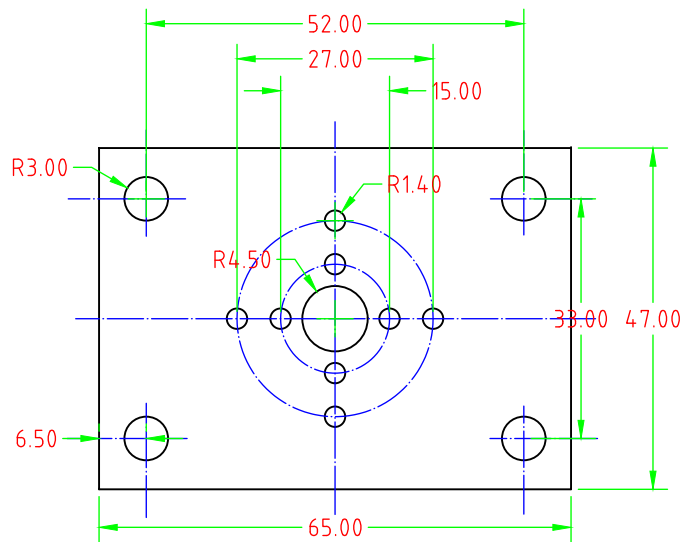
case 102: //f 預備啟動
blmotor.writeMicroseconds(1100);
break;

case 105: //i 切換檔位 select
if(velocitychange == 100)
{ velocitychange = 0;}
else
{ velocitychange = 100;}
}
}
```

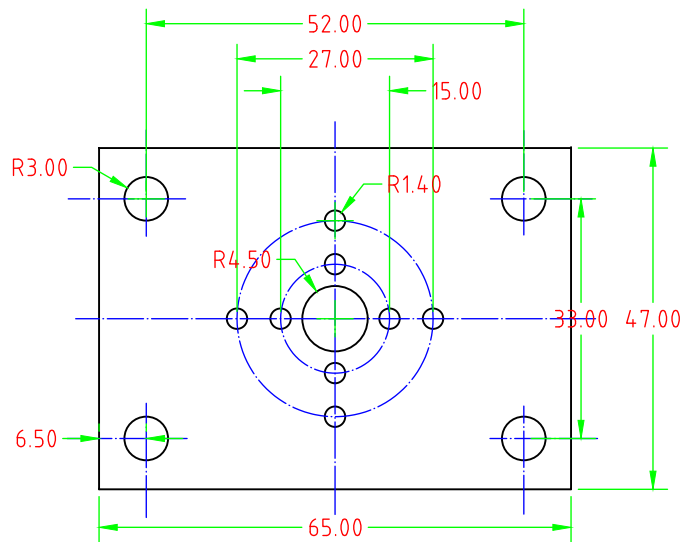
8.2 工程圖



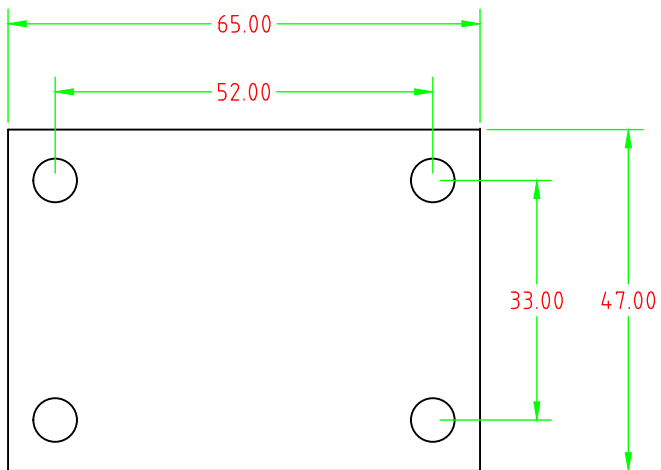
繪圖者	B04502119	國立臺灣大學 機械工程學系
	徐敏懷	
比例： 1 : 1)	下盤車體	日期：
第三角法		圖號：



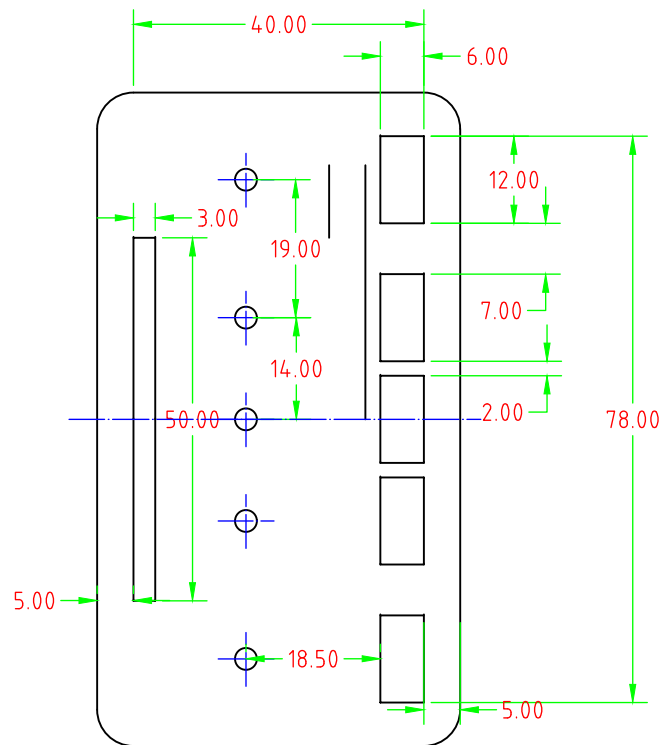
繪圖者	B04502119	國立臺灣大學 機械工程學系	
	徐敏懷		
比例： 1 : 1)	伺服馬達連接		日期：
第三角法	墊片		圖號：



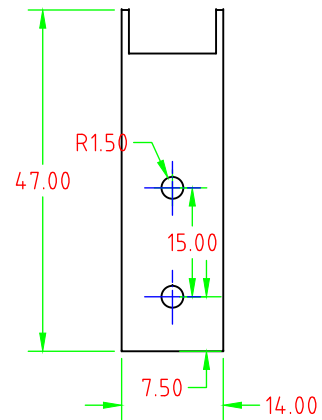
繪圖者	B04502119	國立臺灣大學 機械工程學系	
	徐敏懷		
比例： 1 : 1)	伺服馬達連接		日期：
第三角法	墊片		圖號：



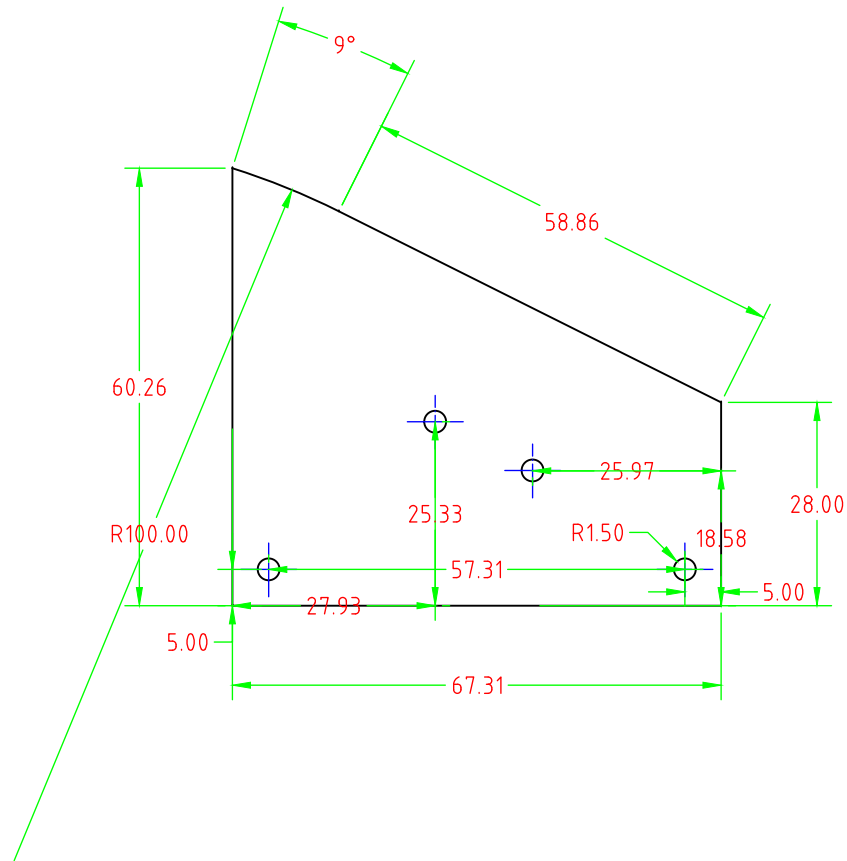
繪圖者	B04502119	國立臺灣大學 機械工程學系
	徐敏懷	
比例： 1 : 1)	後輪墊片	日期：
第三角法		圖號：



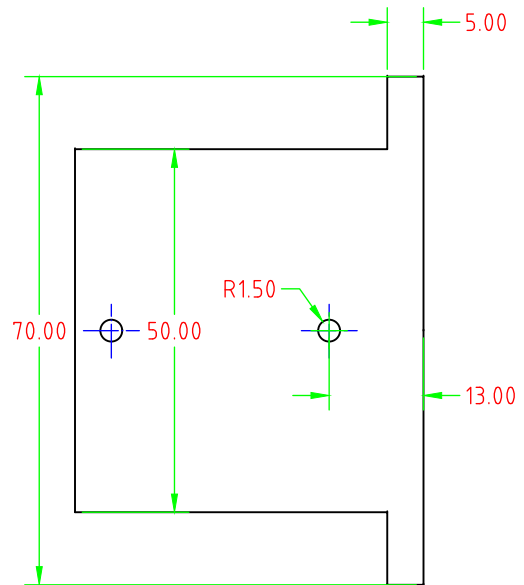
繪圖者	B04502119	國立臺灣大學 機械工程學系
	徐敏懷	
比例： 1 : 1)	紅外線支撐架	日期：
第三角法		圖號：



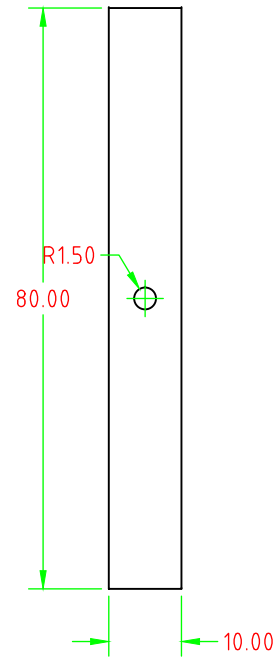
繪圖者	B04502119	國立臺灣大學 機械工程學系	
	徐敏懷		
比例： 1 : 1)	紅外線感測器	日期：	
第三角法	下層固定零件	圖號：	



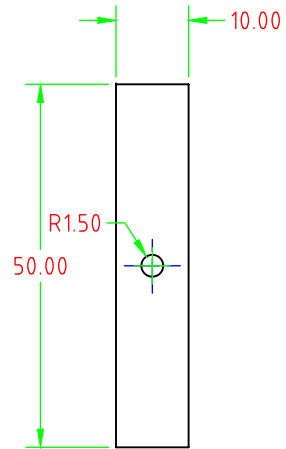
繪圖者	B04502119	國立臺灣大學 機械工程學系
	徐敏懷	
比例：1：1)	第三角法	日期：
		超音波支撐架



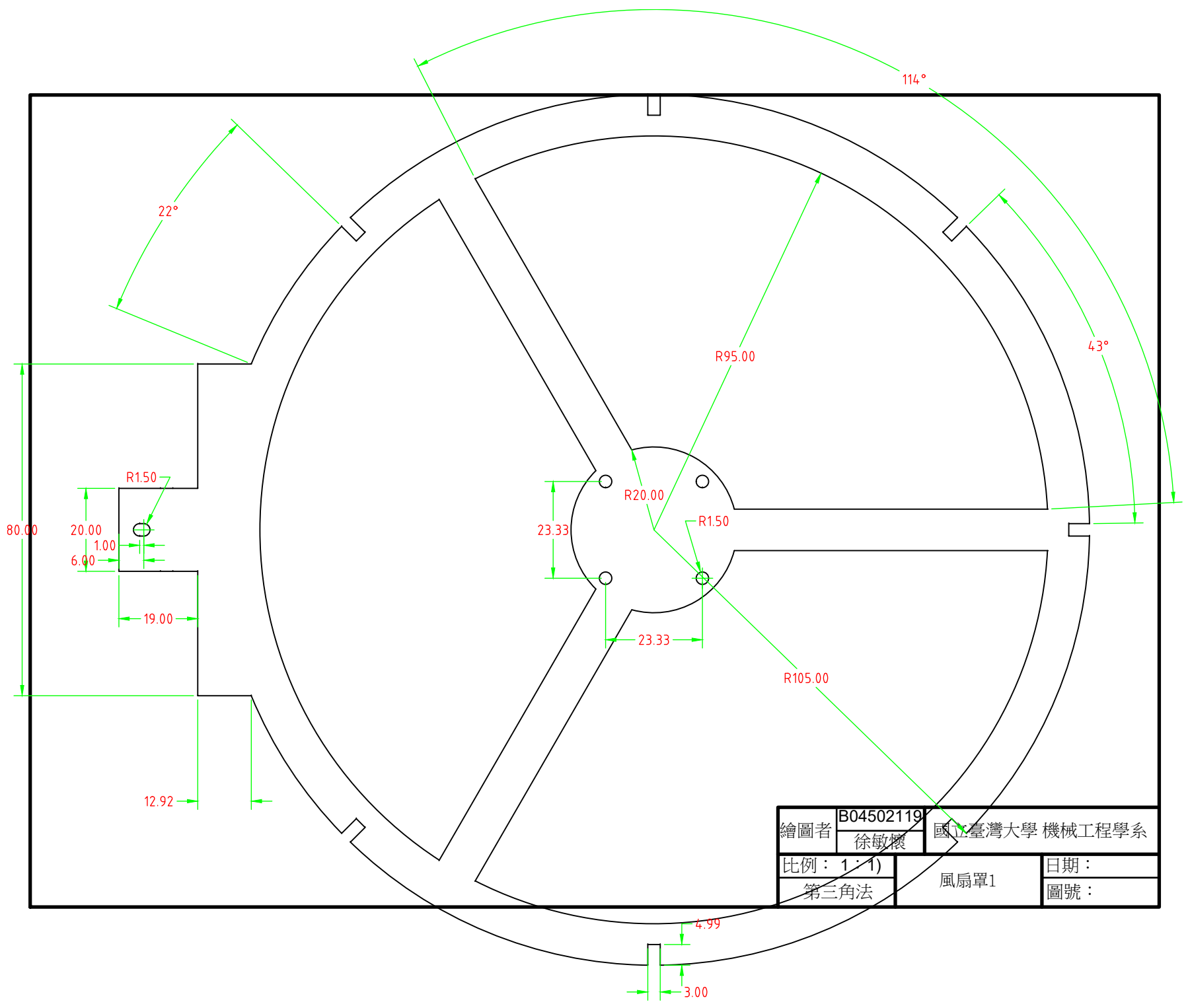
繪圖者	B04502119	國立臺灣大學 機械工程學系
	徐敏懷	
比例： 1 : 1)	電路板固定板	日期：
第三角法		圖號：



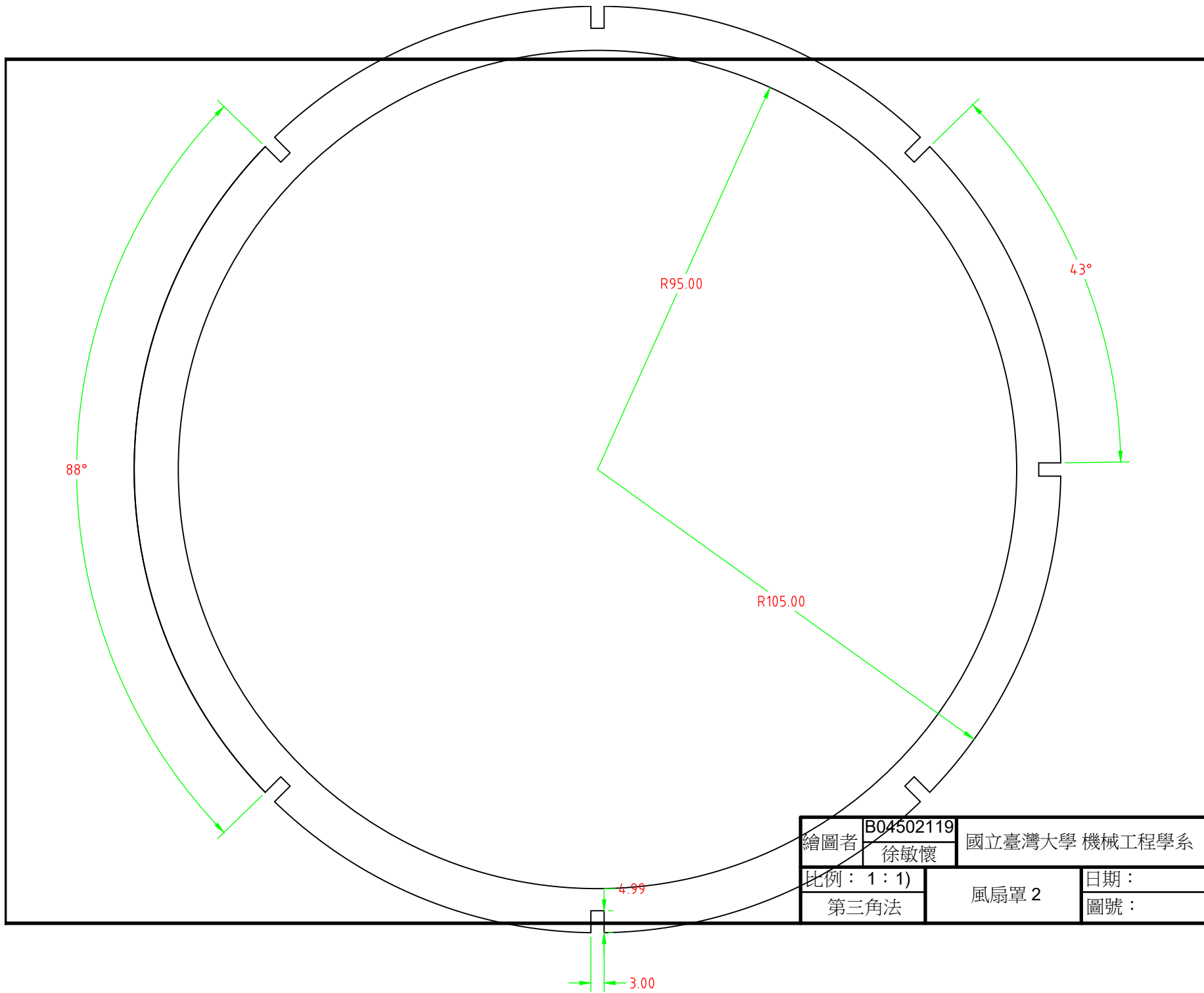
繪圖者	B04502119	國立臺灣大學 機械工程學系	
	徐敏懷		
比例：1：1)		螺絲固定板 1	日期：
第三角法			圖號：

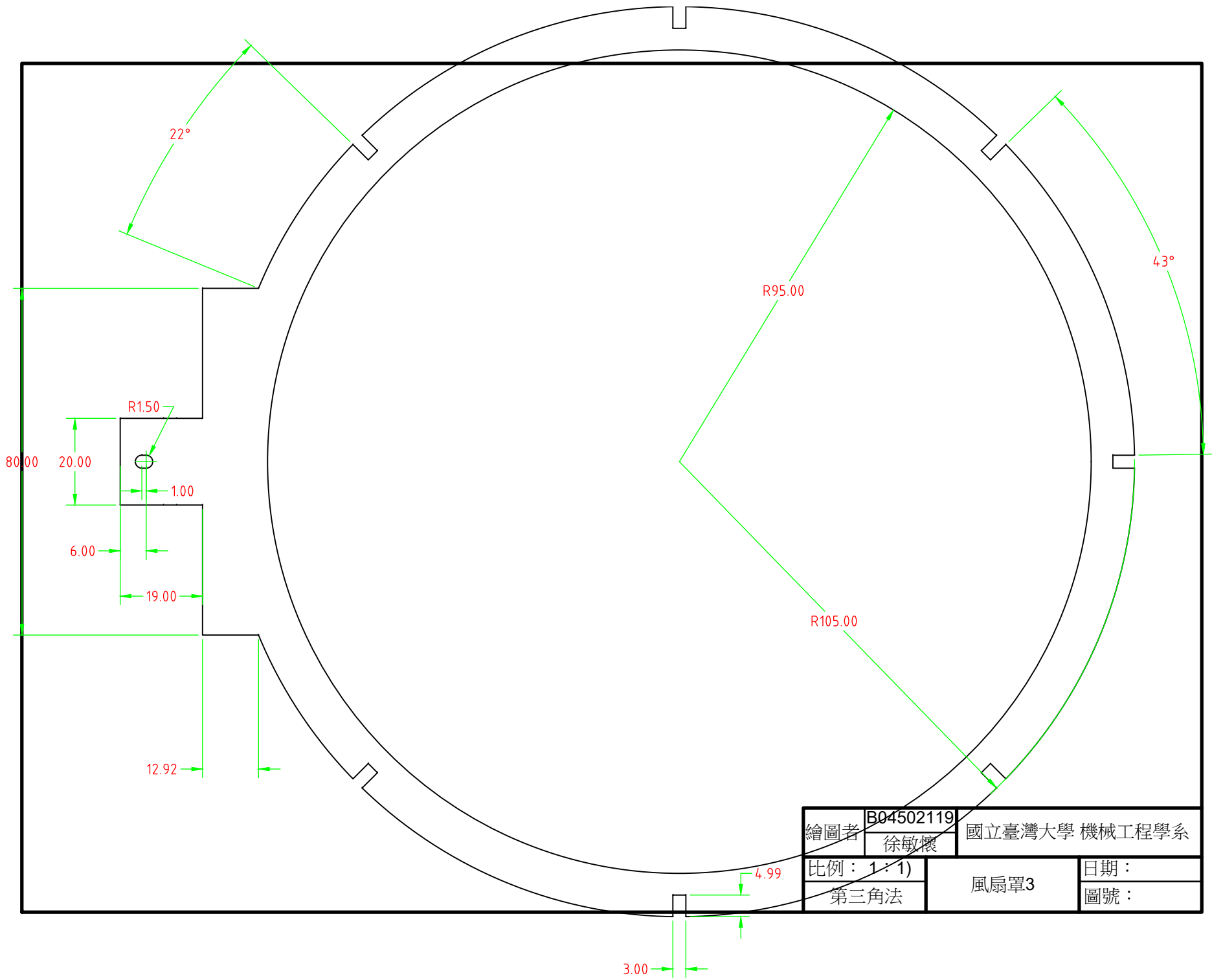


繪圖者	B04502119	國立臺灣大學 機械工程學系	
	徐敏懷		
比例： 1 : 1)		螺絲固定板 2	日期：
第三角法			圖號：

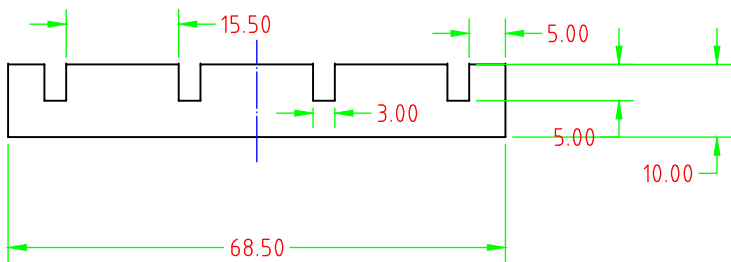


繪圖者	B04502119	國立臺灣大學 機械工程學系
	徐敏懷	
比例：1:1	風扇罩1	日期：
第三角法		圖號：

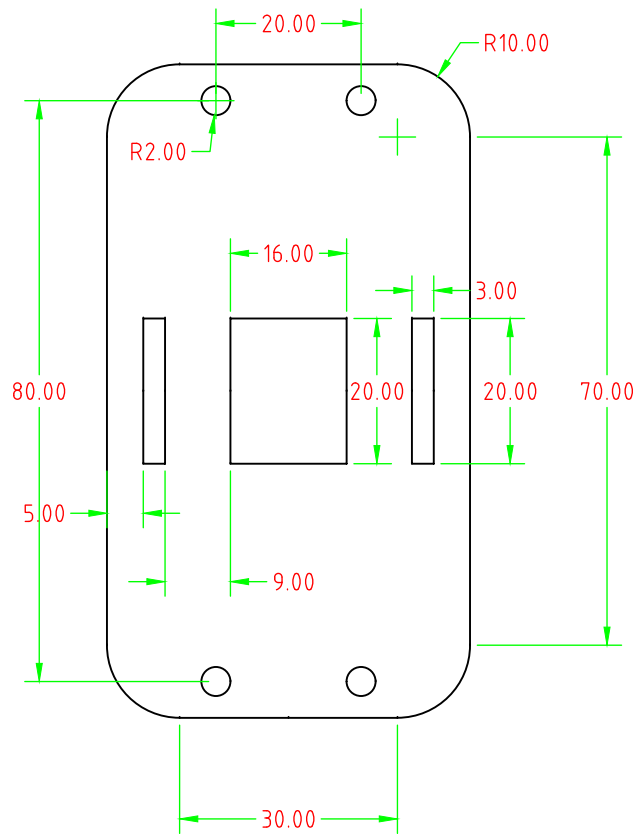




繪圖者	B04502119	國立臺灣大學 機械工程學系	
	徐敏懷	日期：	
比例：1:1)	第三角法	風扇罩3	圖號：



繪圖者	B04502119	國立臺灣大學 機械工程學系	
	徐敏懷		
比例： 1 : 1)		風扇罩固定物	日期：
第三角法			圖號：



繪圖者	B04502119	國立臺灣大學 機械工程學系
	徐敏懷	
比例： 1 : 1)	風扇罩底座	日期：
第三角法		圖號：