

機械工程實務期末報告

風力車

第 26 組	組員姓名	學號	作業或報告分工比例
	曾兆廷	B07502068	20%
	余彥亨	B07502054	20%
	魏振宇	B07502077	20%
	石翊鵬	B07502144	20%
	丁映文	B07502065	20%
	合計		100%

2021 年 6 月 18 日

目錄

一、緒論	1
1.1 引言	1
1.1-1 設計特色概述	1
二、功能需求	2
2.1 規則與需求	2
2.1-1 測試規則	2
2.1-2 功能需求	3
2.2 車體	4
2.2-1 結構支撐	4
2.2-2 煞車系統	4
2.2-3 轉向系統	5
2.3 風扇設計	5
2.3-1 倒車	5
2.3-2 推力	5
2.4 機電系統	6
2.4-1 循跡	6
2.4-2 位置偵測	6
三、設計概念與布置	6
3.1 車體	7
3.1-1 車體主結構的設計	7
3.1-2 轉向系統	11
3.1-3 煞車系統	17
3.2 風扇設計	19
3.2-1 風扇材料、形式選用	19
3.2-2 扇葉數量	20

3.2-3 扇葉尺寸與型號.....	20
3.2-4 風扇位置.....	21
3.2-5 風扇截面與攻角.....	21
3.3 機電系統.....	23
3.3-1 馬達.....	23
3.3-2 電子變速器.....	24
3.3-3 供電系統：電池.....	25
3.3-4 控制板選擇.....	25
3.4 自動控制系統.....	27
3.4-1 紅外線循跡系統.....	28
3.4-2 編碼器測距系統.....	29
3.4-3 PID 控制系統.....	30
3.4-4 陀螺儀.....	31
四、設計分析與驗證.....	37
4.1 分析.....	37
4.1-1 車體受力分析.....	37
4.1-2 扇葉推力分析.....	45
4.1-3 扇葉應力分析.....	51
4.1-4 阻力分析.....	53
4.2 繪製工程圖.....	54
4.3 製造.....	55
4.3-1 車體部分.....	55
4.3-2 風扇部分.....	55
4.4 驗證.....	60
4.4-1 車體尺寸驗證：.....	60
4.4-2 扇葉：.....	61
4.5 實驗.....	62

4.5-1 金屬輪軸的彎曲程度測試.....	62
4.5-2 風扇推力測試.....	65
4.5-3 車體風阻測試.....	67
4.5-4 車體起動力測試.....	69
4.5-5 轉向誤差與阻力測試.....	70
4.5-6 側向支撐力測試.....	72
4.5-7 電池壓降與效能測試.....	74
4.5-8 煞車力測試.....	76
4.5-9 陀螺儀精確度測試.....	78
4.6 討論.....	79
4.6-1 規則需求.....	79
4.6-2 功能需求.....	80
五、附錄	82
5.1 組員心得.....	82
5.2 工作進度與分配	87
5.3 組員互評評分標準	89
5.4 控制系統程式碼	90
5.5 市售產品型錄.....	99
5.5-1 Arduino Uno	99
5.5-2 MG90 servo motor	100
5.5-3 TCRT5000	101
5.6 BOM 表.....	103
5.7 所有工程圖	105
六、參考資料.....	144

一、緒論

1.1 引言

機械工程實務，是一門機械系集大成之課務，整合了這三年在機械系所學到的所有東西，最初從第一堂課聽到的規則並沒有甚麼頭緒，只能試著從去年的撞球車經驗，且在第一周我們分工去查詢往年的案例與網路上的資料，已經有非常多先例可以依循，先將車體設計拆分成三個部分：車體、風扇、控制。

這次期中與期末的差異僅取決於軌道有沒有角度的差異，所以快速的分工、在有限的時間內將整台車的架構先做出來，選擇容錯率最低，實用、製作迅速為最大優先，而後，期末再去思考做精簡化，這是我們整組的共識，風扇、車體、骨架隻外型參考了學長姐報告中，數據較好的幾個規格，便列印出扇葉、車體的成品。

獨立出來的三個部分，雖然各自分工，但共同合作，每周會共同召開小組會議，將所做進度共同彙整，避免在大家做完各自的次系統，卻最後無法組裝起來，同時在明確分工之下，如果各自系統出了問題，僅需要重新設計該系統即可，可以避免最後要全部重新設計的情形發生。

1.1-1 設計特色概述

(1) 全車體組裝分為幾個大型部件，密集板之間的配合使用膠合，而於部件與部件之間由卡榫接合，兼顧結構堅固不易鬆脫及拆裝方便快速的功能。

(2) 整台車為四輪行進，在轉向時車體較為穩定，且考慮期末爬坡的穩定因素。

(3) 避免繁瑣的機構，減低耗費的金額避免超出預算，且維修時更為簡易便行。

(4) 材料選用加工便利且迅速者，多採用密集板而少用 3D 列印，降低時間成本。

(5) 風扇的設計採用順逆時鐘轉，排風不同方向，以倒車時能不同的動力方向

二、功能需求

2.1 規則與需求

2.1-1 測試規則

測試賽道俯瞰圖見圖一，位於跑道拆卸處的後半段賽道可調整仰角，圖示顏色為方便區分各區段用，非實際測試顏色。實際測試賽道只有循跡線，緩衝區框線，起跑區域框線與停止區域框線。

- I. 賽道全長 330 cm，寬 100 cm。測試時間 10 分鐘，車輛全自主循跡，以風力驅動。
- II. 測試開始時車輛需整車放置於起跑區中，感測器可與框線對齊。
- III. 測試過程中有下列情況則須返回起始區域重新開始，期間不停錶。
A：車體有任意部分離開賽道範圍，B：以外力碰觸車體，C：車體未於循跡線上超過五秒鐘(切換跑道時不計)。
- IV. 不得循跡任何區域框線，違者比照規則 III.處置。
- V. 循跡順序為起跑區，第一循跡線，第二循跡線，第一停止區，第二循跡線(倒車)，第二停止區。未照循跡順序前進比照規則 III.處置。
- VI. 變換循跡路線時需於第一與第二緩衝區內完成，違者比照規則 III.處置。
- VII. 車體需將全部投影面積靜止於停止區內超過三秒鐘，違者比照規則 III.處置。
- VIII. 仰角範圍為 10 度至 25 度，最小調整單位為 5 度。
- IX. 測試中車體明顯無法爬上斜坡時可由同學自行判斷是否要返回起跑區調整角度並/或重新測試。
- X. 於測試時間內最多可選擇 3 組測試角度。取單次最高分者為期末驗收分

數。

XI. 期末測試總分可超過 100 分 惟最終學期成績若超過 100 分則以 100 分計。

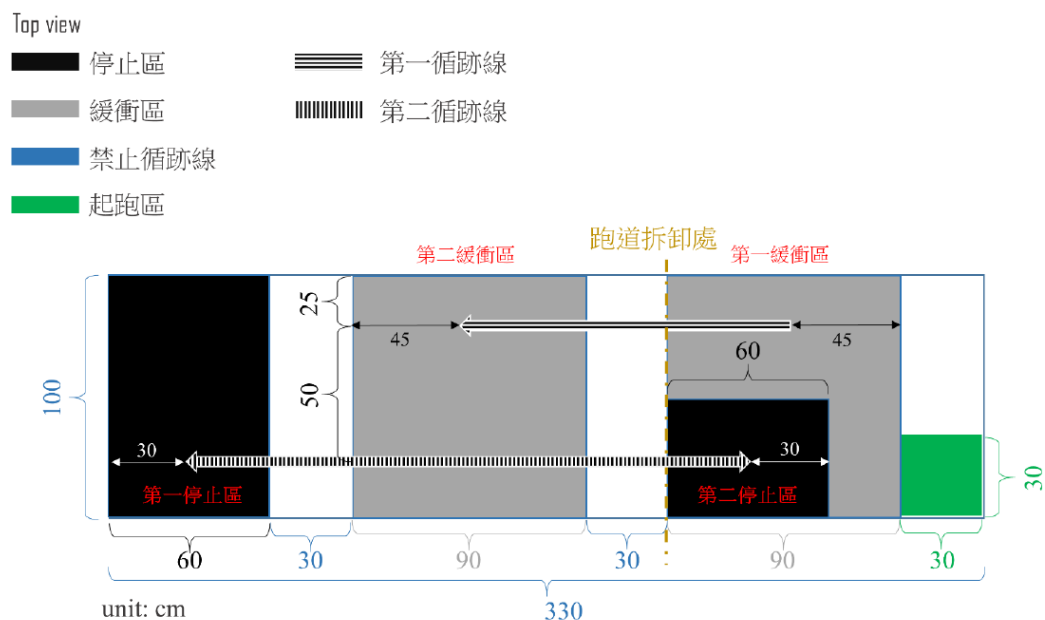


圖 1 期末賽道各區段尺寸範圍

表 1 期末車體循跡表現與累積分數表

累積分數	達成標準
15	車體投影面積完全離開第一緩衝區前正確循跡第一循跡線
20	車體投影面積到達爬坡起點
$35+8*N$	車體投影面積完全離開第二緩衝區前正確循跡第二循跡線
$35+16*N$	到達第一停止區且正確停止 3 秒（請參照規則 VII）
$50+16*N$	正確循跡第二循跡線並倒車至第二停止區
$60+16*N$	於第二停止區內正確停止 3 秒（請參照規則 VII）

2.1-2 功能需求

在作業過程中大致將工作分為三大組，即風扇、車體與控制，在規則之下，各

組別須負責完成以下功能：

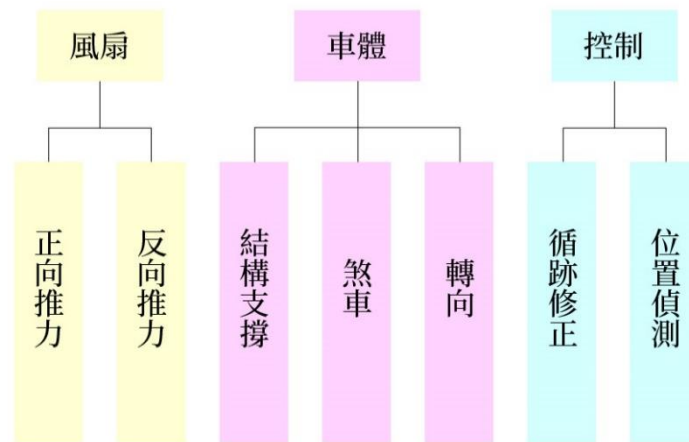


圖 2 功能需求圖

2.2 車體

車體在運行中要能確保其姿態穩定容易控制，車上之零部件與負載物須能確實固定不在運行時鬆脫、掉落。除此之外，風扇葉片若因突發狀況斷裂破損時，風扇罩須能確實將扇葉碎片擋下，對周遭人員提供良好的保護。

為了降低加工製造時間並降低製造成本，各部件之設計須加以簡化。考量功能需求的複雜程度，車體於設計初期時可能需要進行多次設計變更與更換相關零配件，因此車體各部件之模組化與易拆裝性為設計時考量的重點之一。

2.2-1 結構支撐

需能乘載結構本身、動力系統、轉向機構、煞車機構、控制電路以及額外負載(250 ml 鋁箔包)等部件的重量，並留有一定之安全係數。此外，需配備能使車子於場地內順利滑行移動之方式(輪子、氣墊、履帶...)。

2.2-2 煞車系統

需提供車子足夠的煞車力道，使其得以在規定的停止區內完全煞停，並需擁有夠快的反應速度與控制系統互相配合，來達到精準的位置控制。

2.2-3 轉向系統

須能提供車子足夠小的迴轉半徑，在規定的範圍內移動到循跡線所在位置且能在一定範圍內變更循跡線。此外，應設法減少轉向時產生的阻力，確保在不同轉向角度下系統靈敏度的一貫、易於控制。

2.3 風扇設計

風扇為主要的動力來源，在馬達運轉時，要考慮其安全性，首先，鎖住馬達的固定孔隙半徑盡可能的縮小，避免因為偏心率而導致車體震動不穩，影響控制甚至偏離軌道，扇葉本身在設計上也不可以多餘的過大，同時風扇外圍的遮罩也要大到足以包覆整個風扇的範圍。

而風扇與車體連接的龍骨之間，不能僅僅使用卡榫與三秒膠的組合，要用螺絲、電焊等元件加以固定，避免在旋轉時風扇偏離了原先圓心的位置，更甚在旋轉時鬆脫。

2.3-1 倒車

風扇在設計時需要考慮到到達終點後，整台車的行進方向必須逆轉，這有兩個方式，一則將整個風扇的輸出方向旋轉 180 度，一則讓風扇正逆轉輸出的風向為不同的，我們選擇使用其二，類似工廠內的吸排風扇，但是相對的，因為需要雙向都能出風，扇葉對於風扇平面的角度相對單向出風大的多，因此可能會導致在單向輸出時風力不夠，沒辦法造成足夠推力，尤其是期末車體還有角度的影響，故需多做考量。

2.3-2 推力

風扇的功率必須要能夠克服輪胎與地面的摩擦力，將車體運送到定點，同時，因為車體轉向時我們是採用阿克曼四輪系統，所以轉向時，前進方向的分量會隨著旋轉角度而被分散掉，假設我們每次轉向的角度不超過 45 度，車體的重量在設計階段參考歷屆報告均重約設為 1.5 kg，輪胎的材質預計採用塑膠

材料 PTFE，參考資料[1]，PTFE 最大靜摩擦係數為 0.04，動摩擦係數為 0.03。

故車體啟動時應克服最大靜摩擦力：

$$F_s = \mu_s N = 0.04 \times 1.5 \times 9.81 = 0.5886[N] \quad \text{---(1)}$$

車體行進間時應克服動摩擦力：

$$F_k = \mu_k N = 0.03 \times 1.5 \times 9.81 / 0.707 = 0.6244[N] \quad \text{---(2)}$$

故設計風扇時的重點在於如何將效率最大化、最佳化，而實際所需之推力值，將會在風扇完成之後透過軟體模擬以及實驗驗證，來做更精確的估算，並驗證推力是否足夠。

2.4 機電系統

2.4-1 循跡

控制系統最主要的任務即讓車能夠精準的沿著賽道上的黑線前進，並且大略估算車體處於賽道上的位置，以判斷變換車道、迴轉等動作的時機，黑線對光的吸收率高，其他地面部分的反射率高，因此使用紅外線感測器來達成循跡功能。

2.4-2 位置偵測

除此之外，在換車道時沒有黑線，因此必須靠位置偵測來估算車體在賽道上的位置與前進方向，好讓車順利接到下一條線上，同時位置偵測系統也能夠一定程度上在循跡上起到作用，尤其是在倒車的環節，由於所有的紅外線感測器都裝設在車體前端，透過位置偵測來適時修正車體方向是此系統的重要功能之一。

三、設計概念與布置

在本部分中同樣將所有設計依照工作性質分為車體、風扇、控制三部分，

而除此之外，為了做更深入的說明，風扇部分將再被細分為風扇本體部分以及機電部分。

在每個分類下，將會針對系統的開發前置考量、開發過程、主要目標功能與特色以及至期中測試為止所發現的問題做詳細的說明，而因為疫情升溫，期中測試之後的作業無法完成，因此對於期中測試之後的改進，將以規劃的形式報告。

3.1 車體

3.1-1 車體主結構的設計

車體主結構的主要功能為乘載煞車與轉向系統等系統，並且同時兼顧到結構的堅固性，以及在所有設備都裝上車的情況下符合課程的尺寸要求，車體的次要設計目標為最大程度的輕量化，以追求在期末的爬坡測試能夠得到最理想的結果。在車體主結構的設計上，為了充分利用空間，選擇使用長方形的車體形狀。轉向的部分，有前輪轉向以及後輪轉向兩個選項，因為在測試階段需要循跡的功能，而循跡是透過紅外線感應器即時回饋來完成，紅外線感應器以及控制板有一定的時間延遲，而在車體行進中時間延遲越長，就等於偏差值越大，為了讓車體在行進中能以較穩定的路線前進，選擇反應較快，較適於即時做出回饋的前輪轉向系統。

而在最初的發想階段，車輪的可行方案大致分為三種：三輪、四輪以及四輪以上。三輪的優點是轉向機構以及連接的設計容易，在加工上也會比較省時，四輪以上設計的自由度相當高，但是參考資料也相對較少，加上轉向系統設計不易，轉向的效率也不太好，因而在討論中遭到否決，三輪又可分為前二後一以及前一後二，但這兩個方案也都在後續討論中遭到否決，前一後二的轉向系統設計較簡單，但是若車體過重，在高速行進時有可能會遇到重心不穩的問題，造成車體向左前方或右前方傾斜，而前二後一若使用前輪轉向，在轉向系統的結構上與四輪完全相同，而增加一顆後輪雖然一定程度上增加車重以及

機構的複雜度，但能夠帶來更高的穩定度，在權衡之下決議選擇四輪的結構。

在期中測試時所使用的車體著重於空間運用，由於在開發初期尚未決定電變、馬達以及要使用的機電零件型號與數量，因此整體大方向為讓車身可以乘載的零件越多越好，因此使用了如下圖中的雙層式結構，將轉向系統、煞車系統以及鋁箔包等置於不易接觸的下層，上層與側板之間使用卡榫接合，因此隨時可以拆裝，而電池、電變與控制板等需要較高機動性的零件則置於上層。

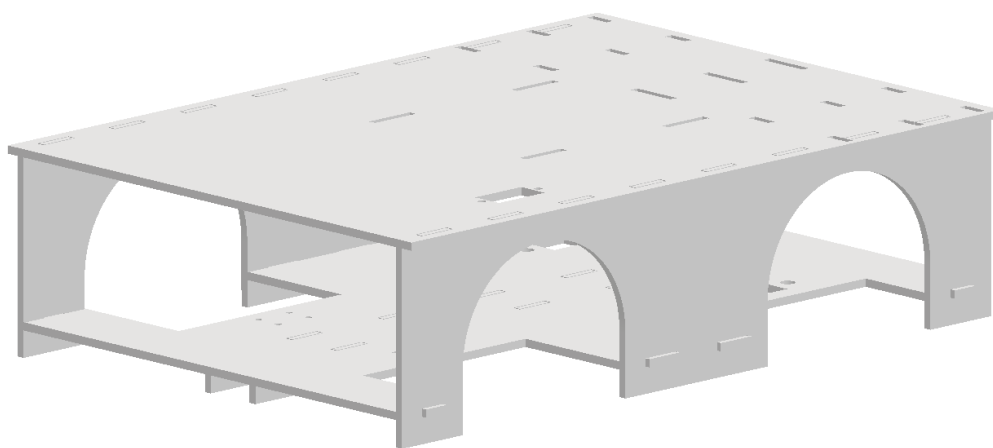


圖 3 期中車體概念設計圖

然而這套車體在期末時並不適用，主要原因為下層底板與地面的距離過近，且前、後輪分別離車體前、後端都尚有一段不小的距離，因此在從平地進入坡道段時車體會頂到地面，因此勢必要將這部分做改良。

除此之外，裝上機構時發現前輪處的空間嚴重不足，也因此轉向系統的伺服馬達必須安裝到上層，再以一個傳動機構做連接，然而因為傳動機構設計的關係，伺服馬達的拆裝工作變得相當困難，而安裝於上層的伺服馬達與安裝於下層的轉向機構互相連接，便代表著每一次要取下上層層板時，都必須經歷一段冗長的伺服馬達拆裝程序。且在測試過程中發現到轉向的動作會對伺服馬達帶來相當大的負擔，這部分將會在轉向系統的章節中詳述，而這更是證明了伺服馬達似乎不宜經常拆裝，完全地抹滅了這套設計的最大優勢。

總結來說，雙層型車體設計的缺點大至有幾個方向：

1. 下層零件拆裝調整不易
2. 無法適應上坡
3. 材料量多且笨重
4. 電池盒等零件沒有合適的位置固定

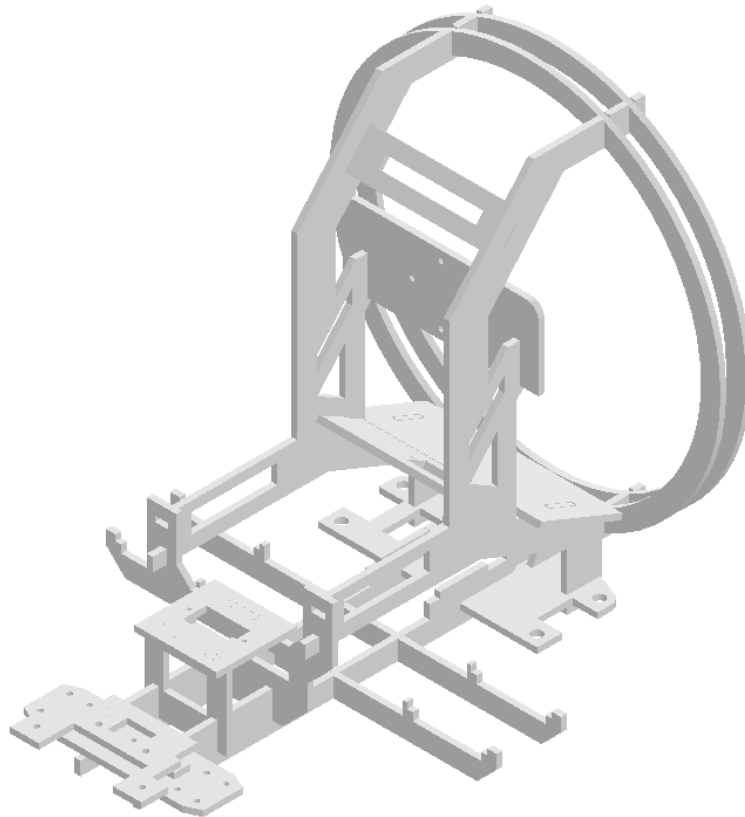


圖 4 骨架型車體概念設計圖

吸取了期中測試的經驗，在期中測試後重新設計的車體如上圖所示，主要改動為將以面為主的車體設計改為以骨為主，所有車體結構都以骨架結構結合而成，幾乎所有零件都可以在不拆除其他零件的情況下做拆除與調整，除此之外，材料量也大幅減少，以達到輕量化的目的，而在結構強度上，計畫在鄰近的結構件之間使用強力膠接合，也在設計階段將此計畫納入設計考量，如下圖所示，在同顏色的零件之間使用強力膠接合，但在需要時也能夠以部件為單位將結構從車體上拆卸。

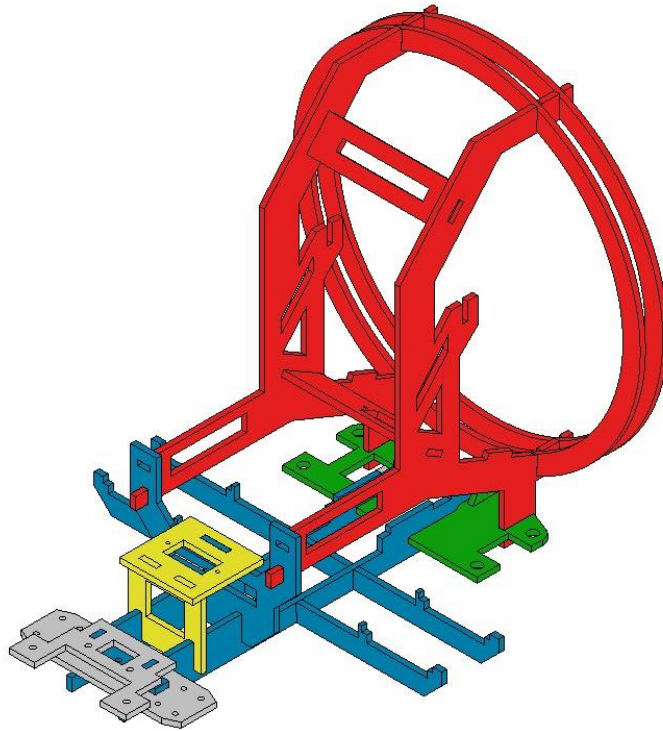


圖 5 骨架型車體可拆卸部分展示圖

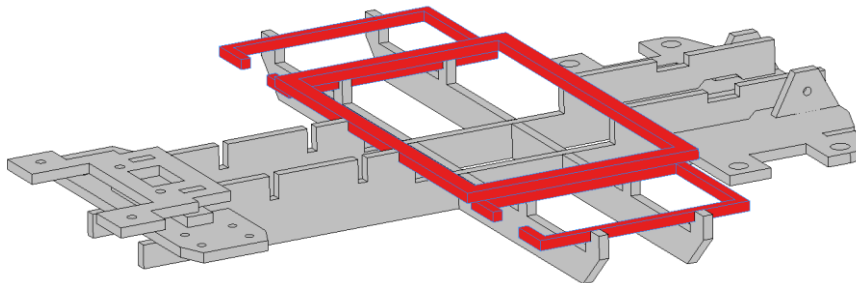


圖 6 中央電池座與鋁箔包座

除此之外，在車體中央的橫樑處也加上三個專屬的鋁箔包座以及電池座，這些結構可以在風力車行動時防止這些元件在車體上滑動，影響到車體重心，尤其期末測試包含爬坡路段，滑動的機會與影響都將會比期中來的大，而其他部件如 arduino 控制板、鋰電池等也皆有安排固定的空間。

綜合以上所述，期末的骨架型車體設計比期中來的更輕、更穩定、且確實落實了容易拆裝的目標。

3.1-2 轉向系統

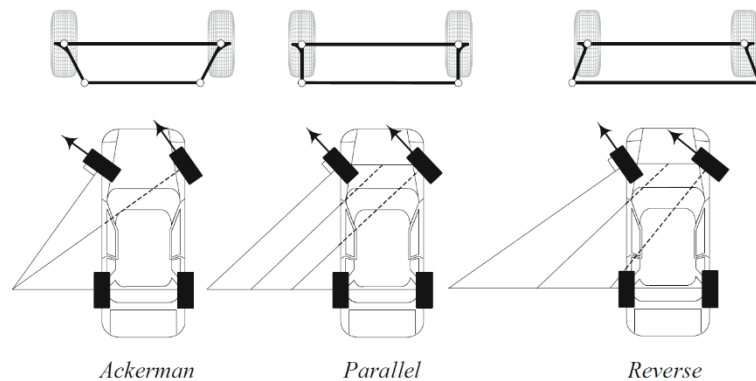


圖 7 Ackerman mechanism 與其他轉向系統比較圖

而在轉向的過程中，車的左側車輪與右側車輪可以視為兩獨立的序列，兩序列存在不同的行進軌跡，而若兩側行進軌跡相差太多，就會在至少一側（通常是兩側同時）造成滑動，進而造成轉向不精準的情況，而為了使車體轉向的軌跡趨於精準，最理想的情況是讓兩側車輪行進軌跡形成一同心圓，如上圖中所示，這種狀況叫做 Ackerman condition，是車輛轉向系統最理想的形式，而為了使轉向情形趨近於這種狀況而設計的轉向系統稱為 Ackerman mechanism，這種機構的主要目標即是讓左右兩側車輪在轉向時沿著兩同心圓軌跡前進，因此同心圓的圓心會落在後軸的延伸線上，這是最理想的轉向系統，兩側車輪完全不存在滑動。

而在這種情形下，兩側的行進距離並不會相同，在一般的汽車上經常裝上差速器，使兩側的轉速不同來解決這項問題，而在這裡我們也使用了相似的概念，即讓兩前輪之輪軸分開旋轉，如此便可以達到差速器的效果。

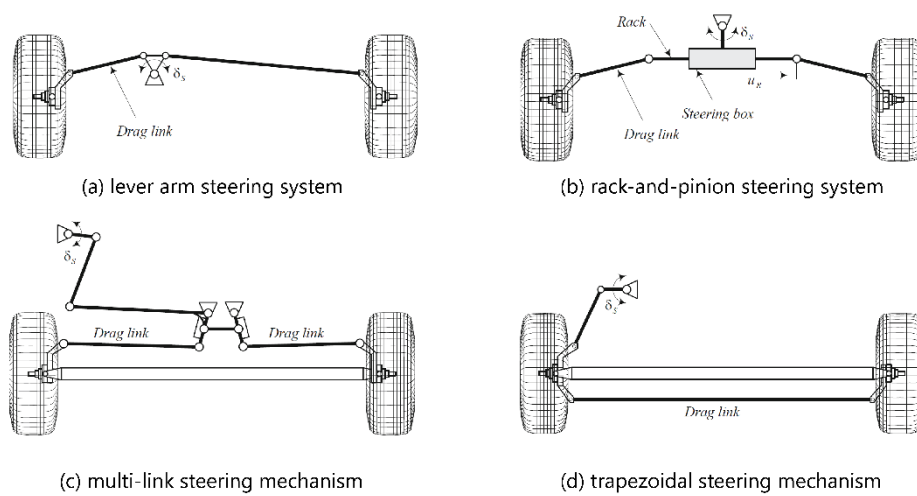


圖 8 常見的 Ackerman steering 形式

經調查之後得到上圖中的幾種常見的 Ackerman steering 形式，這裡住要考慮的點有幾個：製造加工的難易度、轉向與調校的精準度、所需的空間大小、實際應用的可靠度等，經過研究之後，大致列出四種轉向系統的優劣如下表：

表 2 四種 Ackerman steering 比較表

形式	優點	缺點
(a)	- 結構簡單，製造加工容易 - 所占空間小 - 零件配合可靠度高	- 控制系統的調校困難且不精準
(b)	- 所占空間小 - 主動機構的連接較容易	- steering box 的製造難度高 - 自製齒輪可靠度不佳易故障
(c)	- 主動機構離前輪較遠，空間運用空間大 - 轉向精準度高	- 控制系統的調校困難 - 零件數量多，實際配合易故障 - 整體機構較占空間 - 連桿長度設計困難
(d)	- 轉向精準度高 - 結構簡單，製造加工容易 - 零件配合可靠度高	- 連桿長度設計困難

經過比較考量之後，在期中時選擇使用(d)的 trapizodial steering mechanism 做為轉向機構，主要考量因素為該機構相對於其他三種，存在較少重大的缺陷，且結構也不算太複雜，且計算上的困難可以透過查詢資料與討論化解，因此選擇該機構，將伺服馬達裝在其中一邊的短連桿上使其隨伺服馬達轉動，藉此來控制角度，這種設計的缺點為車體重心會偏向其中一邊，以及在左轉與右轉時，同樣的伺服馬達舵角將會對應到不同的轉向角，因為兩邊連桿角度不同的關係。

而在期末測試中，由於以上所提及的 trapizodial steering mechanism 存在的缺點，加上使用到自製齒輪煞車系統的表現良好，經過小組會議討論之後，決議將轉向機構形式改為(b)的 rack-and-pinion steering system，將伺服馬達改為置中，用伺服馬達連接齒輪，並以齒輪帶動齒條驅動 steering box，藉此驅動連接

於 steering box 之上的連桿，而在連桿長度的部分，期中測試中的機構表現尚可接受，因此決定延用，計算之後對應到 rack-and-pinion steering system。

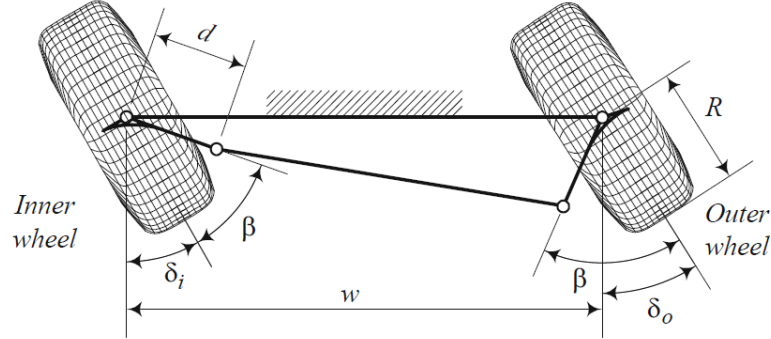


圖 9 轉向計算示意圖

為了設計接近需求的轉向機構，四根連桿的長度以及其角度都需要被精準地設計，實際上 Ackerman condition 是一種理想狀況，在實務上無法達成，但可以根據需求將其誤差降到最小，可定義實際上的轉向系統誤差值為：

$$e = \max |\delta_{D_0} - \delta_{A_0}| \quad (3)$$

其中 e 為誤差值， δ_{D_0} 為車輪實際上的轉向角， δ_{A_0} 為 Ackerman condition 的轉向角，即轉向角最理想的值，這裡經常將誤差值以方均根做定義：

$$e = \sqrt{\frac{1}{n} \sum_{i=1}^n (\delta_{D_0} - \delta_{A_0})^2} \quad (4)$$

考慮到 trapizodial steering mechanism 的內外角關係：

$$\sin(\beta + \delta_i) + \sin(\beta - \delta_o) = \frac{w}{d} - \sqrt{\left(\frac{w}{d} - 2\sin\beta\right)^2 - (\cos(\beta - \delta_o) - \cos(\beta + \delta_i))^2} \quad (5)$$

以及 Ackerman condition 的內外角關係：

$$\cot\delta_o - \cot\delta_i = \frac{w}{l} \quad (6)$$

其中 w 為車寬 l 為軸距，以下圖中的參數為例，可以找出一類似下圖的關係圖：

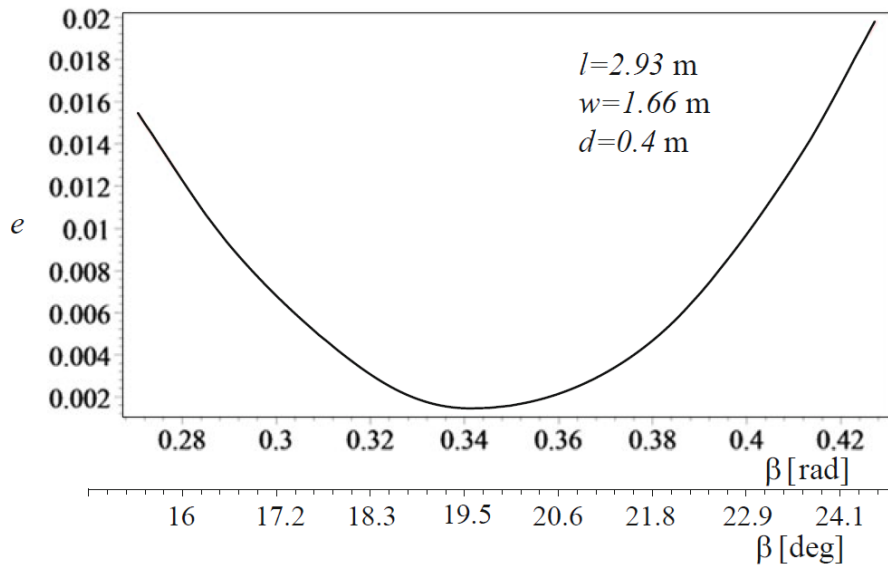


圖 10 β 角與轉向誤差關係圖

若要在 0° 時達到 Ackerman condition，連桿角度 β 應為 $\beta \approx \arctan\left(\frac{w}{l}\right)$ ，但這個角度是沒有意義的，因此最佳化的角度如下左圖所示，比 $\arctan\left(\frac{w}{l}\right)$ 略大。這裡根據尺寸要求與車輪寬度經過粗略的計算之後，在 0° 時達到 Ackerman condition 的角度為 $\beta = \arctan\left(\frac{\frac{15}{2}}{15}\right) = 26.5^\circ$ ，同樣地將這個角度稍微加大到 28.6° ，待後續實際組裝後再驗證這裡的轉向角度是否需要更改，而若考慮到車體的尺寸，連桿的長度會非常短，且非常難加工，尤其是連接在車輪的連桿可能連鎖上螺絲都有困難，因此將連桿長度 d 增加到 40mm。

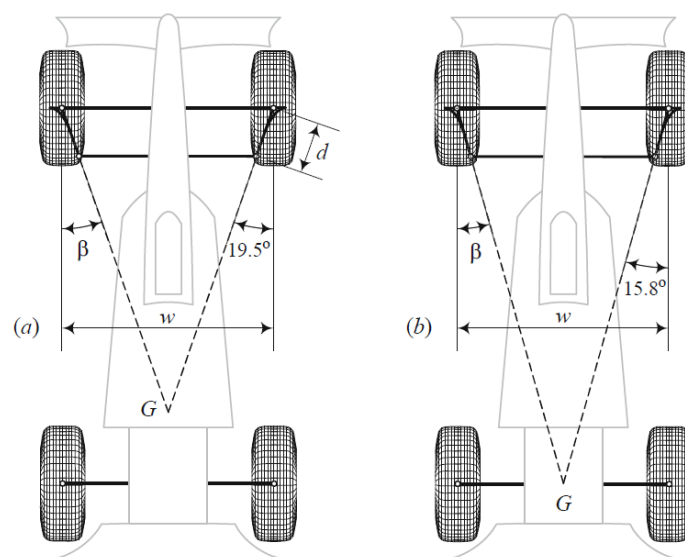


圖 11 最佳化 Ackerman steering 示意圖

除了機構的部分，驅動機構的伺服馬達也相當重要，首先說明選擇的機制，在最開始的討論中，一共列出了三個選項：SG90、MG90 以及 MG996R，三者優缺點如下表所示：

表 3 伺服馬達選用

	SG90	MG90S	MG995	MG996R
示意圖				
最大扭矩	1.8kg/cm	1.8kg/cm	13kg/cm	11kg/cm
工作速度	60°/0.12s	60°/0.12s	60°/0.13s	60°/0.13s
尺寸(mm)	23 × 12.2 × 29	22.5 × 12 × 36	41 × 20 × 43	54 × 38 × 20
重量(g)	9	13.4	55	55
價格	低	次低	次高	高
齒輪材質	尼龍	銅	銅	銅

其中價格部份由於各店家價格落差過大，因此只列出高低排列的情形以供比較

用。在選用標準上，目標為在扭矩足夠的前提下，盡可能地降低預算、空間與輕量化，比較過上述四者並經過測試之後，發現轉向馬達所需的扭矩不算太大，SG90 或 MG90 的等級就能完成，因此先行排除掉 MG995 及 MG996R 兩個選項。

期中測試時選擇 SG90 作為轉向馬達，原因其一為價格較低，其二為 MG90 的中軸所使用的螺絲為 M2.5 的規格，而永齡館所提供的螺絲並沒有此規格，由於傳動機構設計的關係，內附的螺絲也無法使用，因此選擇 SG90 作為轉向馬達，但在測試後期卻發現一個問題：由於 SG90 內部齒輪為塑膠製，推測是因為強度較低的關係，在長期使用之後，中軸開始出現鬆動的現象，嚴重影響到轉向的精準度，甚至在輸入相同 PWM 數值的情況下，做出不同的轉向動態，這種不確定因素亟需排除，因此在會議決議，期中測試之後將轉向馬達改為 MG90，雖然中軸螺絲規格特殊不易取得，但 SG90 的問題似乎更大，且更無解，在決議更改後不久即碰上新冠肺炎疫情爆發，因此沒能來得及測試 MG90 轉向馬達的性能，不過組員對其基本上持正向態度。

3.1-3 煞車系統

煞車系統的設計參考一般市售車常見的兩種煞車形式：碟煞與鼓煞，兩者都是利用摩擦力來減少車輛的動能，碟煞由煞車碟盤與卡鉗組成，煞車碟盤固定在車輪上，與車輪同速旋轉，煞車時由卡鉗夾住煞車碟盤，摩擦而減慢碟盤轉速，鼓煞則是一煞車機構被包圍在環狀的來令片之中，當煞車時煞車機構向周圍張開，向外壓住來令片來達成煞車效果。

這裡因為設計的複雜度較低選擇使用碟煞，在最初的發想中，設計出一如圖 3 的連桿機構，在後輪軸上固定一與輪軸同相轉動的煞車碟盤，將其中一齒輪連接上伺服馬達，其中一側齒輪轉動時，便會帶動另一側的齒輪跟著轉，其中連桿的設計是為了讓機構在接觸煞車碟盤時能夠保持與碟盤表面平行，並在接觸碟盤的表面以三秒膠貼上切片的橡皮擦作為煞車皮，並在控制系統中設定

兩個不同的伺服馬達角度，可分別做出輕煞與重煞的動作。

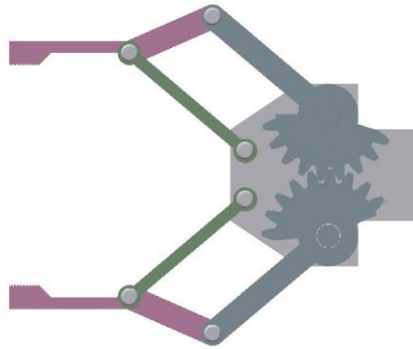


圖 12 三連桿式煞車機構示意圖

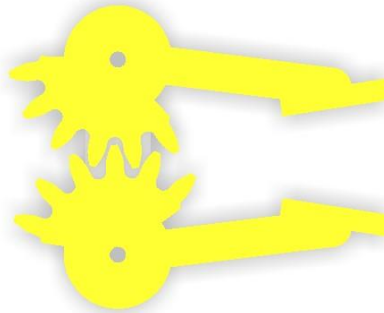


圖 13 簡化版碟煞示意圖

但這個機構占的空間過大，尤其是基於設計上的限制，其必須被擺放於前後輪之間，而設計出的機構全長已經接近整車的車長，因此後續將連桿的設計刪除，因為在齒輪轉動角度不大的情況下煞車皮的角度並不會改變太大，且作為煞車皮的橡皮擦為彈性材料，即使兩接觸面不平行，也能夠隨接觸面產生變形，因此選擇將複雜的連桿機構刪除來節省車體空間，改為如上圖的簡化機構。

而在煞車馬達的選用上，同樣具有 4 種伺服馬達的選項，詳情可以參照 3.1-2 中的表格與比較標準，對於煞車馬達的選用標準基本上與轉向相似，即在扭力值允許的前提下，盡量降低預算、尺寸以及重量，在這四者中基於同樣的理由選擇了 SG90，且在期中測試中表現尚可，因此計畫將此系統繼續延用至期

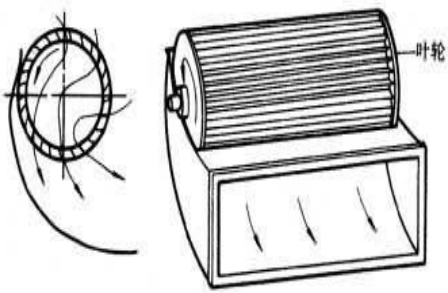
末，但有鑑於轉向馬達上中軸鬆脫的情況，組員擔心此情況在煞車馬達使用較久之後也會出現，因此也隨時預備將煞車馬達換為 MG90 的空間，如 3.1-1 中所提到的，若要更換煞車馬達，只需將煞車系統從車上卸下即可，拆裝過程的同時其他部件依然可以正常獨立運作。

3.2 風扇設計

3.2-1 風扇材料、形式選用

最初設計時曾經參考以下兩種風扇：

表 4 風扇樣式選擇比較

	 橫流扇	 直流扇
優點	1.風力較強 2.穩定性較強	1.參考資料較多 2.成本低 3.設計容易
缺點	1.輸出風力成片狀 2.設計不易	1.風力較弱 2.易斷、脆弱

在設計的初期，提供參考的資料較少，我們最後選擇採用了直流扇，一來是在開頭不曉得怎麼樣形式的風扇功率更高，所以選擇參考數據更多的直流扇，從往年的學長姐中汲取經驗，二來是呈現片狀的風力雖然更為集中，但是在倒車的設計上較為困難，必須要讓整個橫流扇旋轉 180 度，故暫定採用直流

扇，若期末推力不足再考慮橫流扇。

3.2-2 扇葉數量

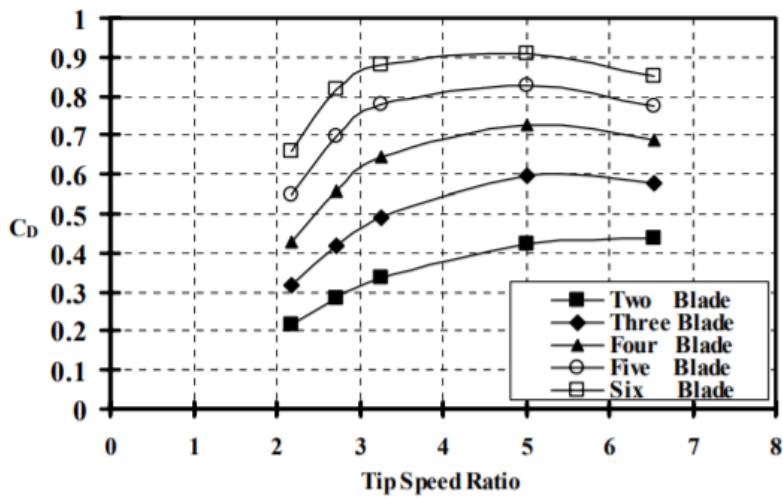


圖 14 扇葉數目的效率分析

風扇葉片的數量根據上述文獻，雖然隨著葉片數量的增加，出風量確實會得到提升，但是相對的也增加了重量，且如上圖，最大的 C_d 值並不會正比增加，所以相對來說，最平衡的數目應為三個葉片，在讓風速得到最大提升的同時，也能最大的提升風扇的功率。

3.2-3 扇葉尺寸與型號

為了符合尺寸規範，且風扇外圍必須被風罩保護，在這些限制之下我們將風扇的半徑定為 15 cm，以期在確保符合規定下能保有最大之推力。然而因為 3D 列印機本身的料槽大小限制，15 cm 的葉片大小並不可行，需要在進行縮減。

對風扇推力而言翼形的影響在於它的升阻係數比，參考前人經驗，我們決定選用較多學長姊採用的 NACA2412 進行翼形設計。



圖 15 參考 NACA 的風扇設計示意圖

3.2-4 風扇位置

風扇的位置取決於風扇的推進方式以及車身的設計，在先前的討論中，組員考慮過離心式推進的可能性，但在顧及實際製造時的器密集精度要求，最後得到的結論是使用風扇直接推進將會是較可行的做法。決定推進方式後，便是決定風扇在車體上的位置了。由於轉向機構會設置在車體前方，中段將用於乘載駕駛員（鋁箔包飲料）及電池等其他系統裝置，風扇最適宜的便在車體後方，同時也可以保障風扇後方的流場不會受到任何東西的影響，以保證風場完整不與環境互相干擾。

3.2-5 風扇截面與攻角

翼形會影響風扇升阻係數比，而我們選用的截面為 NACA 翼形，因此設計方面會先從分析 NACA 翼形的升阻比下手，現階段的風扇先採用參考學長姊的經驗，效仿他們選用的角度進行測試，再自行研究 NACA 翼型的最佳角度。惟今年的功能要求還有一項是需要能夠倒車，這點我們希望能透過風扇逆轉達到，故 NACA 翼型可能不是最佳解。在這個基礎之下，我們還繪製了另一種風扇；參考電風扇的葉片形狀，以細長的矩形為截面，透過曲線的掃略來做成風扇，以期達到逆轉時也能有無不錯的推力輸出。

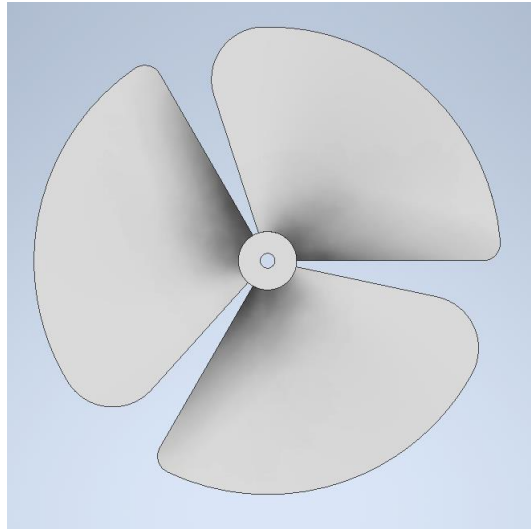


圖 16 參考電風扇設計的風扇設計

由圖 13 可以明顯觀察到，用於掃略的截面為長方形。

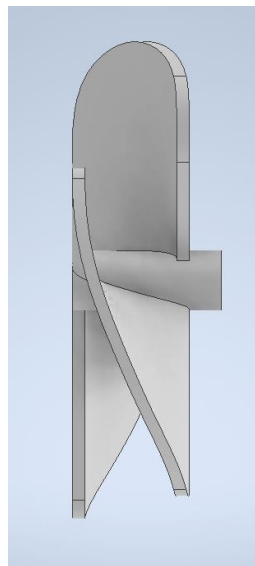


圖 17 參考電風扇設計的風扇側視圖

3.3 機電系統

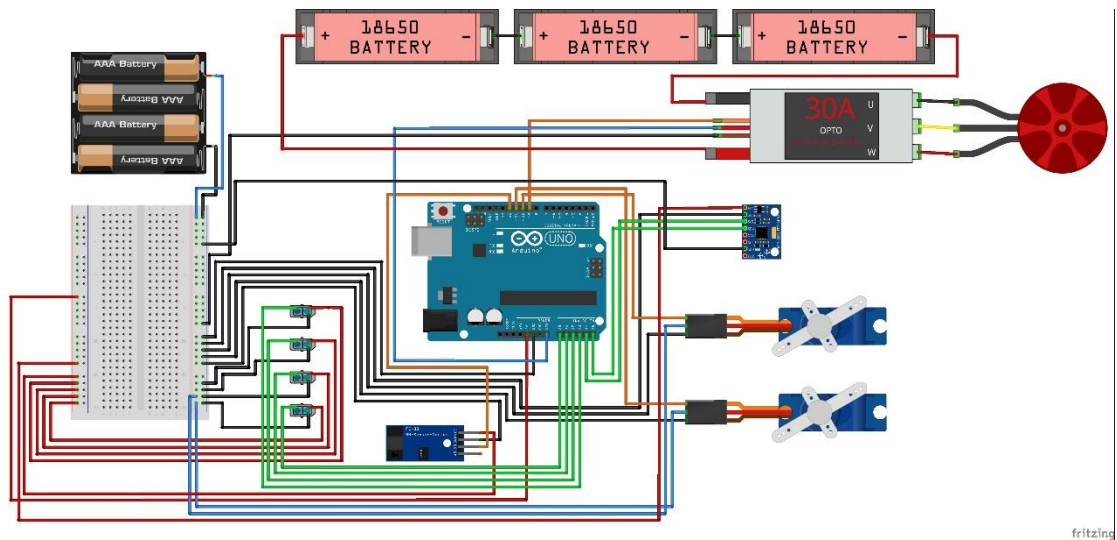


圖 18 機電系統概念圖

3.3-1 馬達

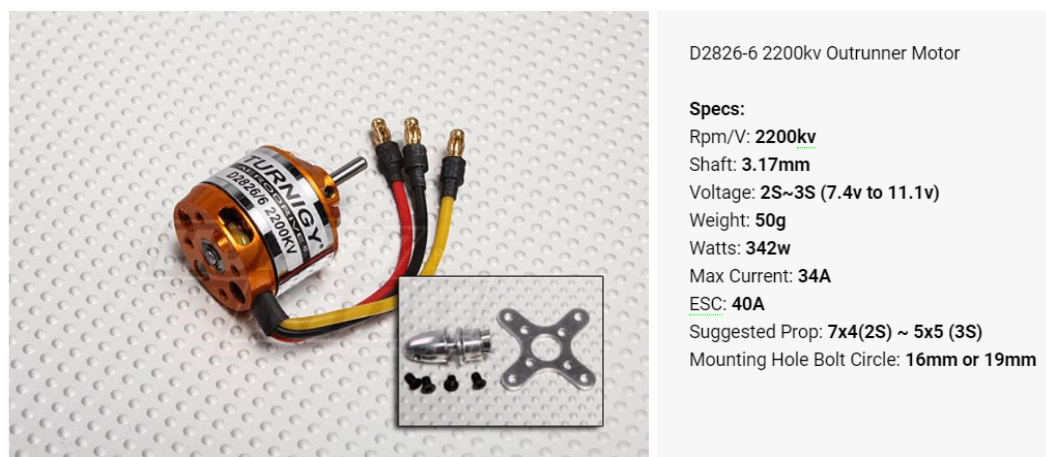


圖 19 D2826-6 2200K_v Motor 外觀及特性

由於需要以足夠的力矩讓葉片能夠給予車體足夠的前進的動力，因此挑選馬達的重點便是轉速以及力矩。最後我們選用不需要碳刷磨擦的無刷馬達，相比於碳刷馬達，其優點為馬達的溫升小，效率高。我們這組是使用從學長車上拆下來的 D2826-6 2200K_v brushless Motor。2200K_v 代表 2200 rpm/V，亦即在無負載的情況下每增加 1V 的電壓會增加 2200 rpm 的轉速。而我們是使用鋰離子聚合物電池，並會串連三顆，所以在空載的情況下，馬達可以達到

$11.1V \times 2200rpm/V = 24420rpm$ 的轉速。

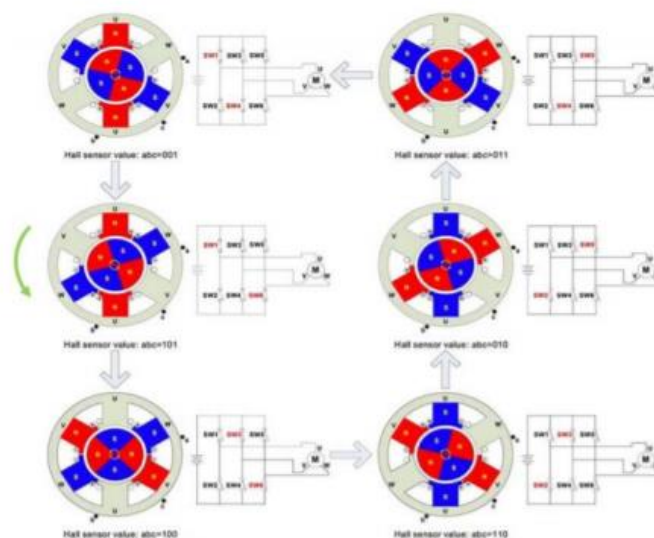


圖 20 無刷馬達作用步驟

透過三條驅動線的電流輸入改變，進而產生磁場的轉動變化驅動馬達。無刷馬達的三條驅動線會依照相位變化來依序交換其擔任角色，每一條線都是平等地位，只是隨相位來做不同的角色，相位則一共有 6 相，於 6 相輪替完後，再重複交替。

3.3-2 電子變速器



圖 21 電子變速器

根據圖一，可知此顆無刷馬達的最大功率為 342 W，在鋰聚合物電池的供

電下，最大電流為 $342\text{ W} \div 11.1\text{ V} = 30.81\text{ A}$ ，因此我們採用 Skywalker 40 A 電子變速器，確保電流在安全範圍內。這與圖一中所需要的電變一樣，都是 40 A。

從無刷馬達的工作原理可知道，無刷馬達透過三條驅動線的供電變化讓馬達內部的磁場發生改變進而驅動馬達，而電變就是利用內部的 MOSFET 來達到供電變化的功能。此外，電子變速器也可以將電池提供的直流電，通過脈波寬度調變的方式(PWM)來達到調節馬達轉速的功效。

3.3-3 供電系統：電池



圖 22 鋰聚合物電池-NCR18650B

我們所選用的 D2826-6 2200 kV Motor 需要 11V 的供給電壓及 34A 左右的最大瞬間電流，因此我們選用串聯數 3S，電容量 3400 mAh，以及放電 C 數為 2C 的鋰電池-NCR18650B，其中 S 代表串聯 3 顆 3.7V 的鋰聚合物電池，放電 C 數的定義為 1C 代表一小時放光電池電量的電流，因此該顆電池可允許的最大放電電流為 $3.4\text{ A} \times 2 = 6.8\text{ A}$ 。於實際測量後發現最大可以讓無刷馬達 Duty 到達 66%，足以應付沒有爬坡的期中測試。

3.3-4 控制板選擇

大部分組別的開發版選擇是 Arduino 系列，由於我們沒有需要使用到視覺辨識或是機器學習等功能，因此 Arduino 系列是夠用的，因此最後選擇

Arduino 系列。接著是 Arduino UNO 及 Arduino Mega 2560 的比較，在可用腳位的使用上，Arduino Mega 2560 有更多的彈性空間，且記憶體也大上許多。

計算所需腳位，總共需要的類比腳位為：紅外線循跡模組 5 個，因此 Arduino UNO 可以應付。而在程式計算速度及記憶體方面，由於程式中並沒有特別複雜的數學運算且看過學長姐的報告後，我們這組決定把 PID Control 改成 PD Control，因此不需要特別大的記憶體去紀錄 I Control 的過去誤差累積，因此控制板的選擇上，Arduino UNO 及 Arduino Mega 2560 的記憶體以及運算速度並不會是主要考量。

最後，因為手邊剛好有 Arduino UNO，再加上 Arduino UNO 比 Arduino Mega 2560 還便宜 135 元，所以最後選擇 Arduino UNO 做為控制面板。而在供給給控制板的電源方面，為了給予穩定的 5V 電壓輸出，我們使用行動電源供給電力。相較 9V 電池或是碳鋅電池，行動電源可以穩定輸出電壓，續航力佳，且可以充電重複使用。

3.4 自動控制系統



圖 23 整體控制策略流程圖

整體的控制策略如上圖所示，其特色為將整個測試賽道分為幾個不同的

state，再依照 state 的不同來更換程式碼，以達到在不同區域中完成不同任務的目標，而識別 state 的方式則是藉由車頭的紅外線感測器去偵測地面上的橫線，並利用紅外線、編碼器以及陀螺儀三種不同的感測器以達到即時動態回饋。

3.4-1 紅外線循跡系統

硬體元件選擇市面上常見的 TCRT5000 模組，它提供類比和數位兩種輸出模式，且可以利用電路板上的可變電阻來調整敏感度。類比訊號會回傳 0 到 1024 的值給控制板，可藉由調整類比訊號臨界值來判斷是否讀到黑線；而數位訊號則是回傳 0 與 1 的訊號來判斷讀到黑線與否。具體運作方式是模組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，光敏三極體一直處於關斷狀態，此時模塊的輸出端為高電平，板載指示 LED 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，光敏三極體飽和，模塊輸出低電平，板載指示 LED 被點亮。如圖 19 所示：

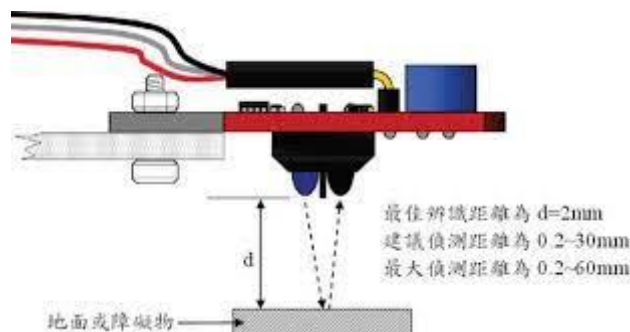


圖 24 TCRT5000 模組

在循跡策略的選擇上，紅外線探測使用 Case by Case 的策略循跡，其優點為直觀且方便編寫程式，因此我們決定使用較為單純的 case by case 循跡。具體操作方式如下：先設定四個紅外線感測器的輸入腳位並讀取類比訊號。而在架設感測器時，找出每一個感測器適合的閾值範圍，將在閾值範圍之外的數值設為 1，在閾範圍之內的數值定為 0。因為感測器間距的關係，同一時間只會有一個紅外線感測器偵測到黑線，因此總共會有五種不同的狀態。將向左向右兩

個方向分別給定兩種轉彎大小，加上遇到橫線後，也同樣有五種狀態。將此五種狀態互相應對，即可以達成循跡的目的。感測器的種狀態分別為：1000：向左大轉彎、0100：向左小轉彎、0010：向右小轉彎、0001：向右大轉彎、1111：遇到橫線。之所以選擇讀取類比訊號而不是數位訊號是因為在研究學長姐的報告時，發現如果讀取數位訊號時，會發現即使沒有移動紅外線模組，量測數值仍然會不斷的在 0、1 之間跳動。而造成這現象可能主要原因有三：

1. **物體間距**：若與黑線的間距過大就會感測到相當多的雜訊，若與測量物的間距過小，就算不在黑線上，也會顯示持續感測到黑線的狀態。
2. **背景光線與物體的表面粗糙度**：必須以讀值最小的紅外線當作是感測到背景光線，將其他紅外線讀值減去此數值來去除背景光線影響，而表面粗糙度也會影響讀值。
3. **模組間的差異**：雖然紅外線模組理論上應該要一樣，但在各種誤差下，每一個紅外線收到的讀值也都不一樣。

因此在衡量後選擇讀取類比訊號，雖然數值可能會跳動，但是在同樣顏色的區間數值會在一定的範圍內跳動，從而可以避免因為與物體間距、背景光線、模組間的差異等問題造成在沒有移動紅外線模組，量測數值仍然會不斷的在 0、1 之間跳動的情況。這樣就可以更為準確的判定是否偵測到黑線。不過缺點是類比訊號的閾值似乎經常隨著一些不可抗力因素而不斷浮動，例如不小心動到紅外線感測器，又或者是在什麼都沒有做的情況下閾值便突然改變了，且每一顆紅外線感測器的閾值也不太一樣，這裡推測是紅外線模組本身帶有的製造問題，而關於解方，目前所採用的方法在每次使用前再次調整並確認閾值是否合適，並沒有遇到太大的問題，治標不治本但尚可解決問題。

3.4-2 編碼器測距系統

如 圖 20 所示，使用 Arduino PHC 測速感測器模組用於測量車子行走速

度，得到數據後進行積分，進而得知行走距離，因此可以在特定距離切換車道。選用這款當作編碼器是因為相較於旋轉編碼器，PHC 測速感測器模組並不需要使用齒輪將轉軸的轉動帶至旋轉編碼器的軸，它只需要設計一光柵並裝於車輪轉軸即可，在使用上較為方便且所占空間較小，不會造成整體機構過多的干涉。而運作原理是當電訊號送入光電耦合器的輸入端時，發光二極體透過電流而發光，光敏元件受到光照後產生電流，CE 導通；當輸入端無訊號，發光二極體不亮，光敏三極體截止，CE 不通。對於數位量，當輸入為低電頻“0”時，光敏三極體截止，輸出為高電平“1”；當輸入為高電頻“1”時，光敏三極體飽和導通，輸出為低電頻“0”。若基極有引出線則可滿足溫度補償、檢測調製要求。

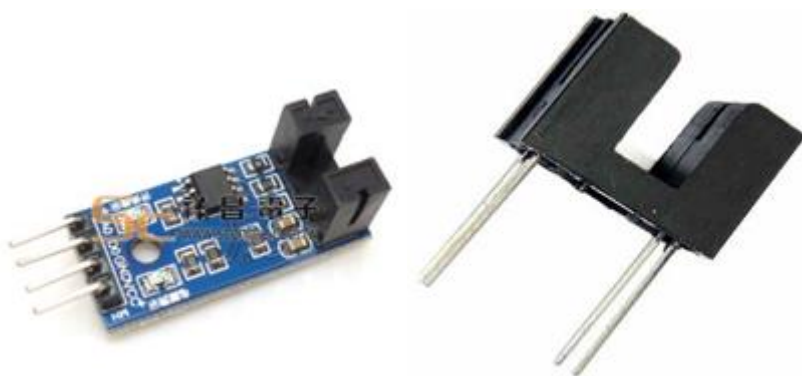


圖 25 Arduino PHC 測速感測器模組

3.4-3 PID 控制系統

使用 PID 控制來減少車子的震盪。經研究後，發現 I 控制主要是藉由累積誤差來補償轉角，因此在轉彎上比較有幫助。而我們今年的任務並不需要轉向，故最後選擇 PD 控制。同時因為紅外線每次讀取訊號之間會有時間間隔，並非連續的讀取，所以必須對先前數值加以存取來進行 PD 的控制。

下列為程式對 PD 的計算方法：

1. P 值的計算方法：P 的算法較容易，只需要乘上當下讀取的誤差即可，較沒有誤差存取的問題。

2. D 值的計算方法：D 在訊號連續的狀態下，效果如同微分值，因此在數位環境下將當下的 error 值，減去上一個狀態的 error 值，可以達到類似的效果。所以我們在程式內設定了記錄前一個 error 值的變數 D_p_error ，還有另一個變數 D_error 紀錄當前狀態的 error 值。將 D_error 減去 D_p_error 即為 D 的值。
3. K_p 的調整： K_p 是需要最先調整的參數，因為 K_p 對 PD_value 的影響最劇烈，當 K_p 值調整的好就可以有效的減少 error tracking 的時間，對於及時循跡的車子有很大的幫助。
4. K_d 的調整： K_d 可以有效的減少伺服馬達的震盪時間，對於轉彎的穩定性有很大的幫助。
5. $PD_value = K_p \times P/1000 + K_d \times D/1000$ 。 PD_value 即為整個避迴路系統的誤差值，當 PD_value 為 0 時，即代表循跡線為直線使車子直線前進。若不為 0，及代表循跡線歪斜不在車體正前方，引此使系統判斷需要進行轉彎的動作。而因為 D 的影響，所以可以有效的降低 steady state error，可以使整個循跡的過程更穩定。
6. Angle：由伺服馬達直線前進時的角度加上 PD_value ，代表當下伺服馬達應該旋轉的角度

期中測試的期間，曾經多次嘗試使用 PID 做速度以及巡跡上的控制，但有鑑於測試赛道上的循跡部分皆沒有彎道，而速度控制上的效果也沒有預期的來的好，因此在期中測試中途就捨棄掉了 PID 控制，但我們認為 PID 控制的常識依然是風力車開發過程中相當重要的一部份，因此仍選擇將其置入到報告之內。

3.4-4 陀螺儀

由於期末測試要求爬坡，為了能準確得知風力車的位置進而得到相對應的風扇轉速以及在期中測試時 PHC 編碼器模組的不穩定性，於是決定加入陀螺儀

作為主要的編碼器，原本的 PHC 編碼器模組依舊會留著，主要負責直線的距離讀取，而陀螺儀則負責讀取在地面轉向的角度、爬坡時與地面的角度以及在爬坡上轉換跑道的角度。之所以還留著 PHC 編碼器模組是因為陀螺儀在六個自由度所讀取到的值是加速度，因此需要進行積分轉換成角位移及位移。但我們組別寫不出加速度轉換成位移的程式碼，僅寫得出角加速度轉換成角位移的程式碼，故留下 PHC 編碼器模組進行位移的測量。

本組經過上網及看過學長姊的報告後，選擇 MPU6050-GY521 這型號的陀螺儀，相較於多組件方案，免除了組合陀螺儀與加速器時之軸間差的問題，減少了大量的包裝空間。下圖為其外型及規格。



- 使用晶片：MPU-6050
- 供電電源：3 ~ 5V (內部低壓差穩壓)
- 通信方式：標準IIC通信協定
- 晶片內置16bit AD轉換器 / 16位元資料輸出
- 陀螺儀範圍： ± 250 / 500 / 1000 / 2000°/s
- 加速度範圍： ± 2 / ± 4 / ± 8 / $\pm 16g$
- 引腳間距2.54mm

圖 26 MPU6050-GY521

在寫程式前，需要先下載 MPU6050 Library 函式庫，而以下為程式碼介紹。

(1)第一部分：宣告變數，分別宣告在六個自由度的加速度

```

#include <Wire.h>

long accelX, accelY, accelZ;
float gForceX, gForceY, gForceZ;

long gyroXCalli = 0, gyroYCalli = 0, gyroZCalli = 0;
long gyroXPresent = 0, gyroYPresent = 0, gyroZPresent = 0;
long gyroXPast = 0, gyroYPast = 0, gyroZPast = 0;
float rotX, rotY, rotZ;

```

圖 27 宣告陀螺儀變數程式碼

(2)第二部分：初始化參數及主程式

```

void setup() {
    Serial.begin(9600);
    Wire.begin();
    setUpMPU();
    callibrateGyroValues();
    timePresent = millis();
}

void loop() {
    Serial.print(" Accel (g)");
    readAndProcessAccelData();
    readAndProcessGyroData();
    printData();
    delay(100);
}

```

圖 28 初始化參數及主程式

(3)第三部分：設定陀螺儀

```

void setUpMPU() {
    // power management
    Wire.beginTransmission(0b1101000);          // Start the communication by using address of MPU
    Wire.write(0x6B);                            // Access the power management register
    Wire.write(0b00000000);                      // Set sleep = 0
    Wire.endTransmission();                      // End the communication

    // configure gyro
    Wire.beginTransmission(0b1101000);
    Wire.write(0x1B);                            // Access the gyro configuration register
    Wire.write(0b00000000);
    Wire.endTransmission();

    // configure accelerometer
    Wire.beginTransmission(0b1101000);
    Wire.write(0x1C);                            // Access the accelerometer configuration register
    Wire.write(0b00000000);
    Wire.endTransmission();
}

```

圖 29 設定陀螺儀程式碼

(4)第四部份：讀取數值處理

```

void calibrateGyroValues() {
    for (int i=0; i<5000; i++) {
        getGyroValues();
        gyroXCalli = gyroXCalli + gyroXPresent;
        gyroYCalli = gyroYCalli + gyroYPresent;
        gyroZCalli = gyroZCalli + gyroZPresent;
    }
    gyroXCalli = gyroXCalli/5000;
    gyroYCalli = gyroYCalli/5000;
    gyroZCalli = gyroZCalli/5000;
}

void readAndProcessAccelData() {
    Wire.beginTransmission(0b1101000);
    Wire.write(0x3B);
    Wire.endTransmission();
    Wire.requestFrom(0b1101000, 6);
    while(Wire.available() < 6);
    accelX = Wire.read() << 8 | Wire.read();
    accelY = Wire.read() << 8 | Wire.read();
    accelZ = Wire.read() << 8 | Wire.read();
    processAccelData();
}

```

圖 30 讀取數值處理程式碼

```

void processAccelData() {
    gForceX = accelX/16384.0;
    gForceY = accelY/16384.0;
    gForceZ = accelZ/16384.0;
}

void readAndProcessGyroData() {
    gyroXPast = gyroXPresent;           // Assign Present gyro reading to past gyro reading
    gyroYPast = gyroYPresent;           // Assign Present gyro reading to past gyro reading
    gyroZPast = gyroZPresent;           // Assign Present gyro reading to past gyro reading
    timePast = timePresent;             // Assign Present time to past time
    timePresent = millis();             // get the current time in milli seconds, it is the present time

    getGyroValues();                   // get gyro readings
    getAngularVelocity();              // get angular velocity
    calculateAngle();                  // calculate the angle
}

```

圖 31 讀取數值處理程式碼

```

void getGyroValues() {
    Wire.beginTransmission(0b1101000); // Start the communication by using address of MPU
    Wire.write(0x43);                   // Access the starting register of gyro readings
    Wire.endTransmission();
    Wire.requestFrom(0b1101000,6);      // Request for 6 bytes from gyro registers (43 - 48)
    while(Wire.available() < 6);        // Wait untill all 6 bytes are available
    gyroXPresent = Wire.read()<<8|Wire.read(); // Store first two bytes into gyroXPresent
    gyroYPresent = Wire.read()<<8|Wire.read(); // Store next two bytes into gyroYPresent
    gyroZPresent = Wire.read()<<8|Wire.read(); //Store last two bytes into gyroZPresent
}

void getAngularVelocity() {
    rotX = gyroXPresent / 131.0;
    rotY = gyroYPresent / 131.0;
    rotZ = gyroZPresent / 131.0;
}

```

圖 32 讀取數值處理程式碼

(5)第五部分:加速度換算

```

void calculateAngle() {
    // same equation can be written as
    // angelZ = angelZ + ((timePresentZ - timePastZ)*(gyroZPresent + gyroZPast - 2*gyroZCalli)) / (2*1000*131);
    // 1/(1000*2*131) = 0.00000382
    // 1000 --> convert milli seconds into seconds
    // 2 --> comes when calculation area of trapezium
    // substracted the callibrated result two times because there are two gyro readings
    angelX = angelX + ((timePresent - timePast)*(gyroXPresent + gyroXPast - 2*gyroXCalli)) * 0.00000382;
    angelY = angelY + ((timePresent - timePast)*(gyroYPresent + gyroYPast - 2*gyroYCalli)) * 0.00000382;
    angelZ = angelZ + ((timePresent - timePast)*(gyroZPresent + gyroZPast - 2*gyroZCalli)) * 0.00000382;
}

```

圖 33 加速度換算程式碼

(6)第六部分:輸出結果至 serial monitor

```

void printData() {
    Serial.println("Gyro (deg/sec)");
    Serial.print(" X=");
    Serial.print(rotX);
    Serial.print(" Y=");
    Serial.println(rotY);
    Serial.print("Angle (deg)");
    Serial.print(" Z=");
    Serial.println(rotZ);

    Serial.println("Angular displacement wrt started position (deg)");
    Serial.print("angel of X axis=");
    Serial.print(angelX);
    Serial.print(" angel of Y axis=");
    Serial.print(angelY);
    Serial.print(" angel of Z axis=");
    Serial.println(angelZ);

    Serial.println("Acceleration (g)");
    Serial.print(" X=");
    Serial.print(gForceX);
    Serial.print(" Y=");
    Serial.print(gForceY);
    Serial.print(" Z=");
    Serial.println(gForceZ);
}

```

圖 34 將結果輸出至 serial monitor

四、設計分析與驗證

4.1 分析

4.1-1 車體受力分析

在車體受力分析中，首先，衡量各個元件施加在車體結構上的負載，決定其大小、位置與方向，並作出合理的假設以簡化分析。接著，從文獻資料中選擇最符合實際條件的材料係數，分別指定到不同的材料上。完成材料係數的設定後，繼續完成接觸條件、邊界條件、網格等其他設定，始可進行分析。最後，對分析結果作出解讀，驗證結構的可靠與安全。

此分析中所採用的工具為 Autodesk Inventor 中的應力分析模組。由於其他分析軟體的學生版大多數皆有嚴格的網格數限制，在幾何形狀較為複雜的情況下不易使用簡化模型(如 2D 線架構)進行分析，加上 Inventor 本身完善的 CAD/CAE 整合，方便使用，因此選擇此軟體進行分析。

表 5 車體主要負載列表

Load	Weight/Unit (g)	Unit	Total weight (g)	Force (N)	Distance to head (mm)
AA battery	22.7	4	90.7	0.89	133
Fan	18.5	1	18.5	0.18	225
250ml drink	261.7	1	261.7	2.57	133
18650 battery	45.6	3	136.8	1.34	133
Speed control	41.0	1	41.0	0.40	55.4
Motor	50.0	1	50.0	0.49	205
Thrust_max	X	X	X	5.70	190

表 5 為納入分析之零組件總表，包含單位重量、組數、縱向擺放位置等資訊。車體之主結構除了要支撐車上各項設備之重量，也扮演串接懸吊/車輪模組與推

力模組的角色。為了將分析簡化，車上酬載的部分排除所有重量在 40g 以下之零組件，包括 Arduino 板、紅外線模組、Encoder 模組、伺服馬達、排線等等。由於須納入分析之主要荷重元件之擺放位置皆位於縱向之中心線上，或對稱地放置於中心線兩側，加上車體結構的對稱性，整體的重心落在中心線上。此外，分析時也會考慮風扇不產生推力以及在產生最大推力下(400rpm, 5.705N)之情形。實際施加之負載，如圖 35 車上負載示意圖圖 35 所示。

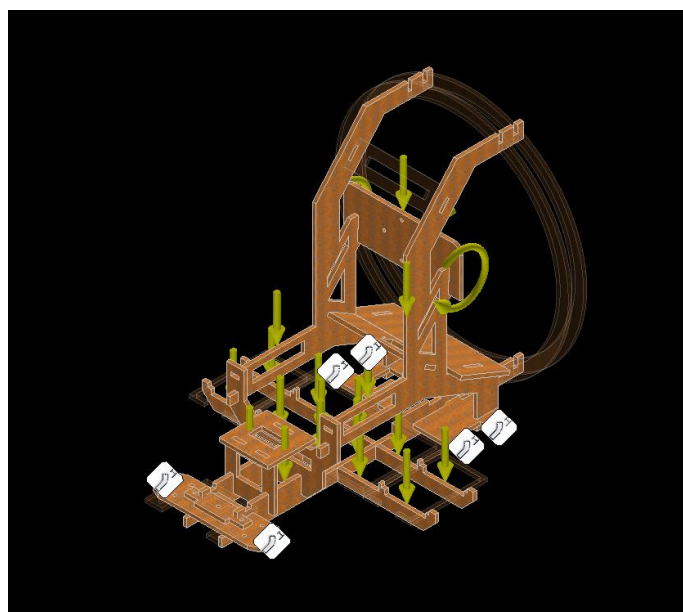


圖 35 車上負載示意圖

車體結構完全由厚度為 3mm 的密集板(MDF, Medium Density Fiberboard)所製成。根據美國農務部林業產品實驗室出版的 Wood Handbook, Wood as an Engineering Material，不同木材所製造的密集板有相當大的強度差異，且其材料性質容易受濕度與材料密度影響。在不清楚實作中心所提供的密集板其原料為何的狀況之下，決定使用測量之密度作為客觀條件尋找較為恰當的材料係數。此外，由於密集板製造公差較大，測量時挑選了不同的車體零件統一秤重以減少誤差，再由 Inventor 中的車體模型計算秤重的零件之總體積，並根據總重與總體積計算出密集板的平均密度。測量之情況如圖 36、37 中所示

$$\rho_{\text{MDF,average}} = \frac{M}{V} = \frac{0.03032 \text{ kg}}{3.71 \times 10^{-5} \text{ m}^3} = 817.2 \text{ kg/m}^3$$



圖 36、37 密集板密度測量情形

表 6 不同密度的密集板之 Young's Modulus

Density(g/cm ³)	Young's Modulus (GPa)
0.8	3.99
0.82	4.38
0.817	4.049

表 7 白松木在不同施力方向與變形方向下之 Poisson's Ratio

Orientation	μ_{LR}	μ_{LT}	μ_{RT}	μ_{TR}	Average
Poisson's Ratio	0.329	0.344	0.410	0.334	0.354

Orientation: $\mu_{Direction\ of\ applied\ stress;\ Direction\ of\ deformation}$;

L: longitudinal; R: Radial; T: Tangential

表 6 為 Wood Handbook 中不同密度之密集板的 Young's Modulus。選擇密度最相近的兩個數據進行內插，得到所使用的數據。

$$\text{Young's Modulus}_{\rho=0.817} = 3.99 + (4.38 - 3.99) \times \frac{0.817 - 0.8}{0.82 - 0.8} = 4.049$$

而根據禾暘空間設計的資料，推測使用之密集板是以松木木屑製程。表 7 為白松木在不同施力方向下的 Poisson's Ratio。由於密集板的製作使用 85% 木屑及

15%的樹脂高溫壓合製成，因此材料已不具有原木的方向性，因此將不同施力方向所測得的 Poisson's ratio 平均作為採用的數據。

由於在此分析步中只關注密集板樑架結構，因此將其與懸吊系統連接處作為支點，設定旋轉與位移六方向自由度固定之邊界條件。接觸條件的部分，則假設各零件之間皆為理想黏合狀態，接觸面之間無任何相對滑動、扭轉。而關於網格的設定，採用逐次降低網格大小的方式進行收斂性分析。在 Inventor 中，網格的設定皆為相對各個零件大小的數值。考量平均網格大小設定為 0.1 時較能充分將各零件切分，因此選擇從 0.1 的大小，以每次縮小 0.025 的方式檢驗數據的收斂性。此外，也將探討風扇是否開啟產生推力對車體所受應力之影響。

表 8 不同 Element size 之下的分析結果

Element size	Mises Stress	Maximum Displacement
0.1	Nodes:55008 Elements:27817 Type:Von Mises Stress Units:MPa 6/17/2021, 10:52 PM 3.403 Max	Nodes:55008 Elements:27817 Type:Displacement Units:mm 6/17/2021, 10:52 PM 0.1518 Max
0.075	Nodes:106222 Elements:59551 Type:Von Mises Stress Units:MPa 6/17/2021, 10:53 PM 9.521 Max	Nodes:106222 Elements:59551 Type:Displacement Units:mm 6/17/2021, 10:53 PM 0.3614 Max
0.05	Nodes:149970 Elements:82486 Type:Von Mises Stress Units:MPa 6/17/2021, 10:53 PM 9.526 Max	Nodes:149970 Elements:82486 Type:Displacement Units:mm 6/17/2021, 10:53 PM 0.3516 Max
0.05 (no fan thrust)	Nodes:149970 Elements:82486 Type:Von Mises Stress Units:MPa 6/17/2021, 12:41 PM 4.208 Max	Nodes:149970 Elements:82486 Type:Displacement Units:mm 6/17/2021, 12:42 PM 0.2206 Max

表 9 不同網格大小下的分析結果

Average element size	Number of elements	Maximum deformation (mm)	Mises stress (GPa)
0.1	27917	0.3627	8.865
0.075	59053	0.3614	9.621
0.05	82486	0.3636	9.656
0.05 (No fan thrust)	82486	0.2266	4.248

分析結果包含 Mises stress 與最大位移量列於表 X 中。從表中可以看到，不同網格大小對位移量較無影響，而 Mises stress 則是在網格數約六萬個左右趨於穩定(圖 38)。在 Mises stress 的部分，當風扇不提供任何推力時，車體所受最大應力約為 4GPa；而當風扇提供最大推力時，車體所受之最大應力約為 9.5GPa。同樣根據 Wood Handbook，密集板材的降伏應力約在 18GPa 左右，因此整個車身結構在最保守的正常情況仍有接近 2 的安全係數，若考量分析忽略所有 40g 以下影響較輕微的零件，以及實際操作時的一些外力介入(如用手將車子拿起)等情況，強度仍然非常足夠。

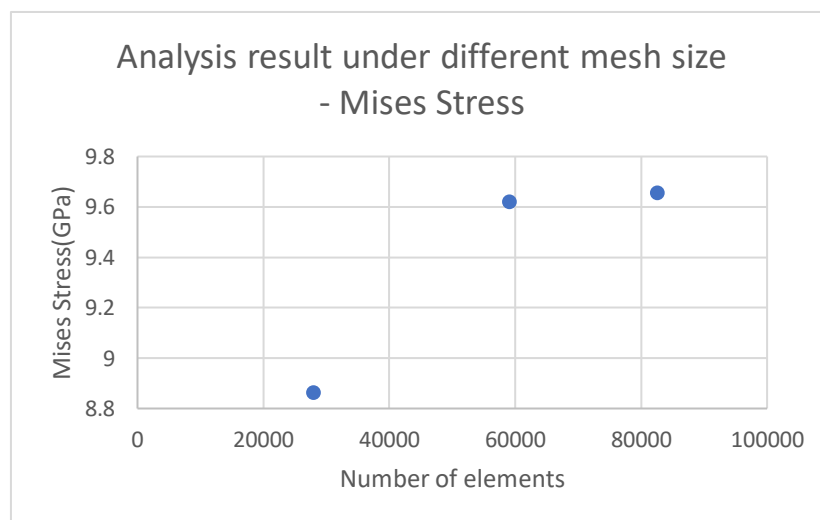


圖 38 Mises Stress 隨著網格數增加之收斂情形

由局部放大圖 X 中可以看到，在風扇產生最大推力的情況下，最大應力發

生於車體前半部用以固定轉向伺服馬達的平台以及用以承載風扇馬達和 18650 電池的馬達架之間的接合處(圖 39)。最大應力之所以會發生在此處，推測是結構下層的樑在前後輪軸之間，因為承受了位於中央的飲料、兩側的 4 顆 3 號電池與電變的重量而下沉彎曲，而位於上方的馬達架所承受的重量相對較小，彎曲的程度也較小，因此用於連接兩者的 Y 形零件(圖 40)在完全黏合的接觸條件下會受到一個彎曲力矩，加上位於此零件之前用於固定轉向伺服馬達的平面(圖 4041)提供一個後推的力量，讓 Y 型零件的彎曲更加明顯，在邊界上產生較大的應力。

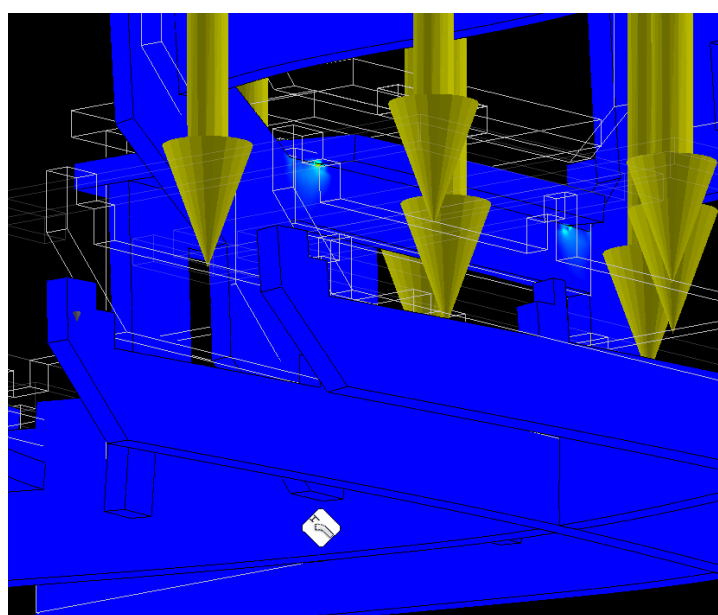


圖 39 接合處

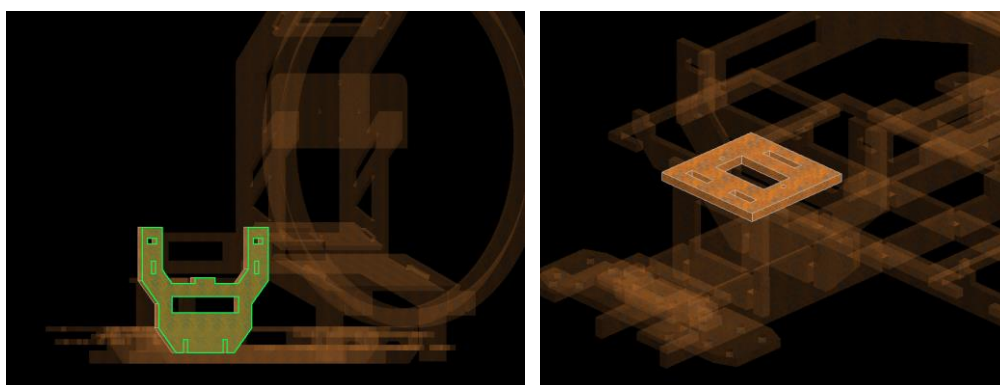


圖 40、41 Y 形零件與平面

有關於風扇是否開啟對結構所受應力的影響，在風扇不產生推力的情況下，最大應力發生的位置與風扇提供最大推力時相同，且應力有所減小(9.656GPa 降

低至 4.248GPa)，但是位於馬達架尾端底部的應力增加(0.684GPa 增加至 1.149GPa)。推測是由於馬達是以四顆螺絲水平固定於兩個馬達架之間的另一平面上，懸空的馬達會對馬達架產生一個向後仰的力矩，因此在風扇尚未開啟時對馬達架後側底部接合處產生了較大的應力。當加入風扇所產生的最大推力 5.7N 後，向前的推力舒緩了馬達重量所造成向後仰的力矩，後側底部的應力因而減小。兩種情況之馬達架尾端底部局部放大圖如圖 42、43 所示。

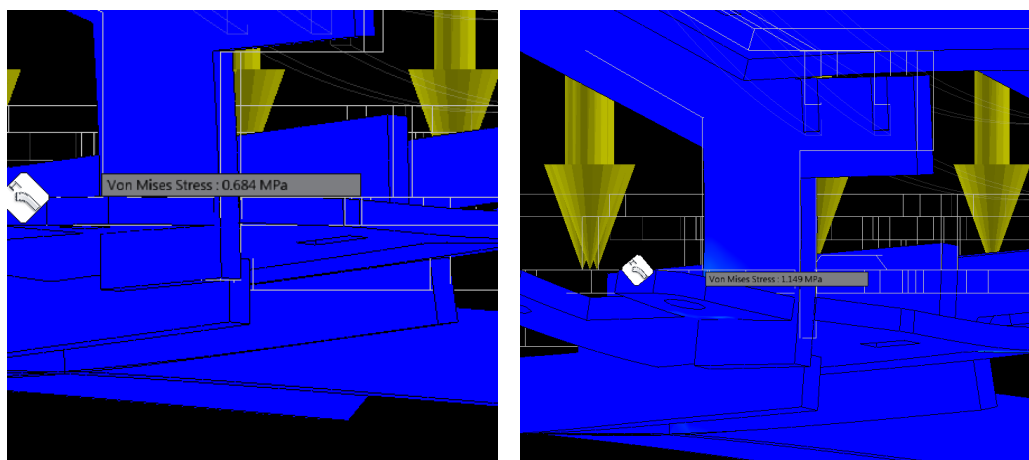


圖 42、43 馬達架尾端底部局部放大圖

4.1-2 扇葉推力分析

4.1-2.1 動量理論

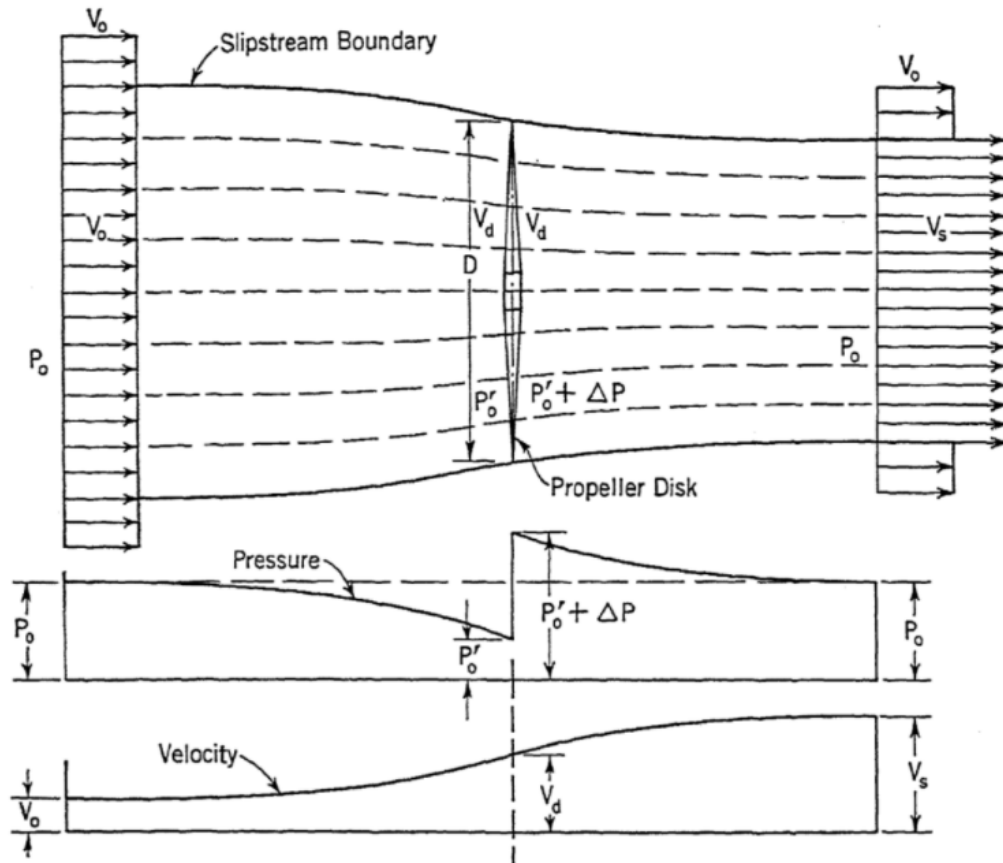


圖 44 動量理論模型[6]

理論中假設空氣是無黏性且不可壓縮的理想流體，且因為參考的軸向設置在風扇上，讓原先轉動中的風扇可以靜止狀態來考慮，故這裡假設風扇是一個圓柱狀的空間，而空氣相對此空間進行運動。

因此，我們可以直接利用 Control volume 來進行動量平衡，意即控制容積的前後邊界都設為均勻流場，上下邊界則以流線選定為容積邊界，可以將風扇直觀地，以流場前後的動量變化來表示其推力，即如下式：

$$\text{Thrust} = A \times \rho \times V_d \times (V_s - V_0) \quad (6)$$

其中 A 為圓盤的表面積， ρ 為空氣密度， V_s 、 V_0 分別是流場的頭尾流速。而因為假設流體為理想流體，因此可以藉由白努力定律將其中的速度項以壓力項替

換成下式:

$$\Delta p = \left(p_0 + \frac{1}{2} \rho V_s^2 \right) - \left(p_0 + \frac{1}{2} \rho V_0^2 \right)$$
$$\text{Thrust} = A \times \frac{1}{2} \rho (V_s^2 - V_0^2) \quad (7)$$

其中 p_0 為初始壓力，而將兩式比較就可以得出 $V_d = \frac{1}{2} (V_s - V_0)$ 。

其實，動量理論其所使用的假與現實有相當大的出入，其模型較為簡化，因此並沒有考慮到風扇的阻力(drag)、流場旋轉所造成的能量損耗、流體的壓縮能損(compressibility losses)、槳葉對流場的交互干擾,所以對於設計的分析較適合作為初步的評估。

4.1-2.2 Ansys 內部設定

為了讓車體能夠順利前進，推力必須大於靜摩擦力。推力的數據將由程式模擬分析和實際測試而得，其中程式模擬將參考學長姐的經驗，使用 ANSYS WORKBENCH 進行分析；實際測試的部分，由於在量測原理與機工實驗中，已經體驗到風洞測量出來的數據可能存在相當的誤差，故在風洞測試之後，將再採用另一種方式測量推力。使用已知彈力系數的彈簧若干，將風扇及馬達固定在一導軌上，同時以彈簧抵住一固定牆面，藉由馬達輸出讓風扇產生推力，並造成彈簧的形變，後將形變量帶入虎克定律公式，藉此計算推力。

在分析風速時，我們使用 Ansys Fluent 求解器,並且假設

1. 符合動量分析中的條件，即控制容積下，空氣為無摩擦、黏滯之連續流體。
2. 因為此模擬主要是模擬此風扇在沒有外物干擾下的運作情形，因此省略地板、車體、馬達等造成的影響。
3. 在設定之中，我們將風場的直徑設定為風扇架的直徑 0.3 m，長度假設為 1 m，並將風扇周圍的風簡化為一個 rotating zone，設定直徑為 0.2 m。
4. 不同風扇、風速的網格化均訂以統一的標準如下圖所示。

Sizing	
Use Adaptive Sizing	No
☐ Growth Rate	Default (1.2)
☐ Max Size	Default (0.10863 m)
Mesh Defeaturing	Yes
☐ Defeature Size	Default (2.7157e-004 m)
Capture Curvature	Yes
☐ Curvature Min Size	Default (5.4314e-004 m)
☐ Curvature Normal Angle	Default (18.0°)
Capture Proximity	No
Bounding Box Diagonal	1.0863 m
Average Surface Area	3.6574e-002 m ²
Minimum Edge Length	1.5e-003 m

圖 45 不同風扇、風速的網格化均訂以統一的標準

5. 而計算時的設定如下，採用暫態分析(車輛狀況多為啟動時)，計算規格為 Reynold Stress。

Run Calculation


Time Advancement

Type

Fixed

Method

User-Specified

Parameters

Number of Time Steps

20

Time Step Size [s]

0.5

Max Iterations/Time Step

10

Reporting Interval

1

Profile Update Interval

1

Options

☐ Extrapolate Variables

☐ Report Simulation Status

☐ Loosely Coupled Conjugate Heat Transfer

圖 46 計算時的設定

4.1-2.3 各風扇流場比較

在選擇風扇時，我們需要考慮不同風扇何者能夠給予更大的推力，因此，針對兩種不同的風扇，取 200rps 分別做分析如下圖。

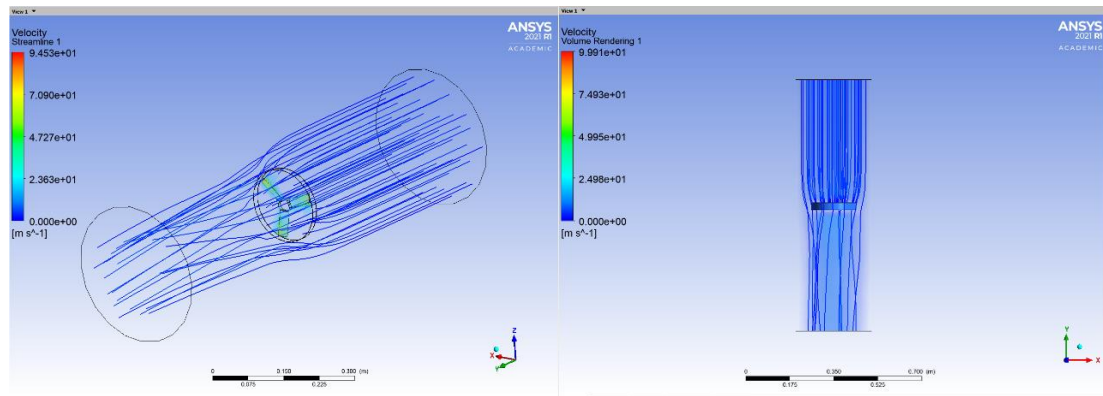


圖 47 NACA 扇葉流場圖

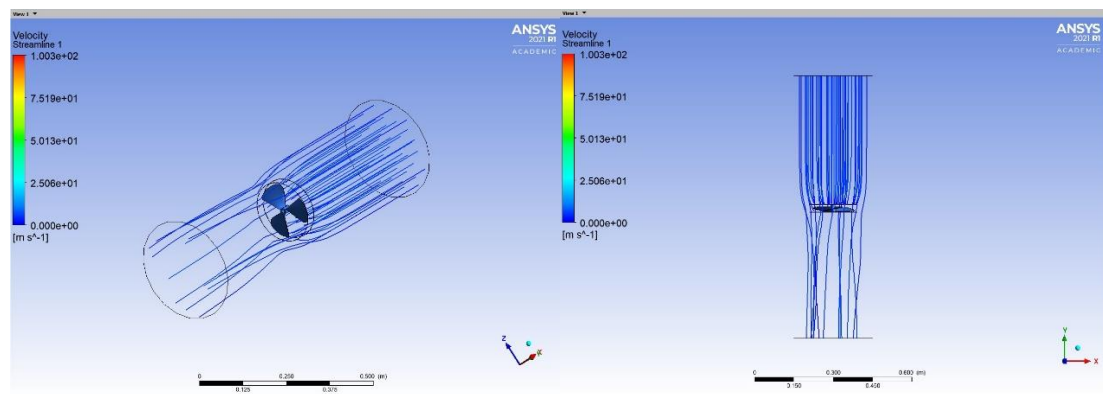


圖 48 排風扇葉流場圖

在兩者的斜視視角時，可以明顯看到 NACA 扇葉經過風扇後，流力線順著逆時鐘方向有顯著收束的趨勢，而相較之下，排風扇葉的流場收束能力相對就差的許多，不僅沒有收斂的趨勢，每根線也不全然皆是順逆時鐘同個方向，空氣繼續往後吹也很快有發散的趨勢了。

此外，兩者的推力，NACA200rps 為 2.279N，排風扇葉 200rps 為 1.384，兩者的輸出比值相差達 64.7%，而原先 NACA 只有在正面風速較強的缺點，也利用將各扇葉的傾角調大作為補償，故我們最後採用 NACA(上圖即為該版本)。

4.1-2.3NACA 風扇流場分析

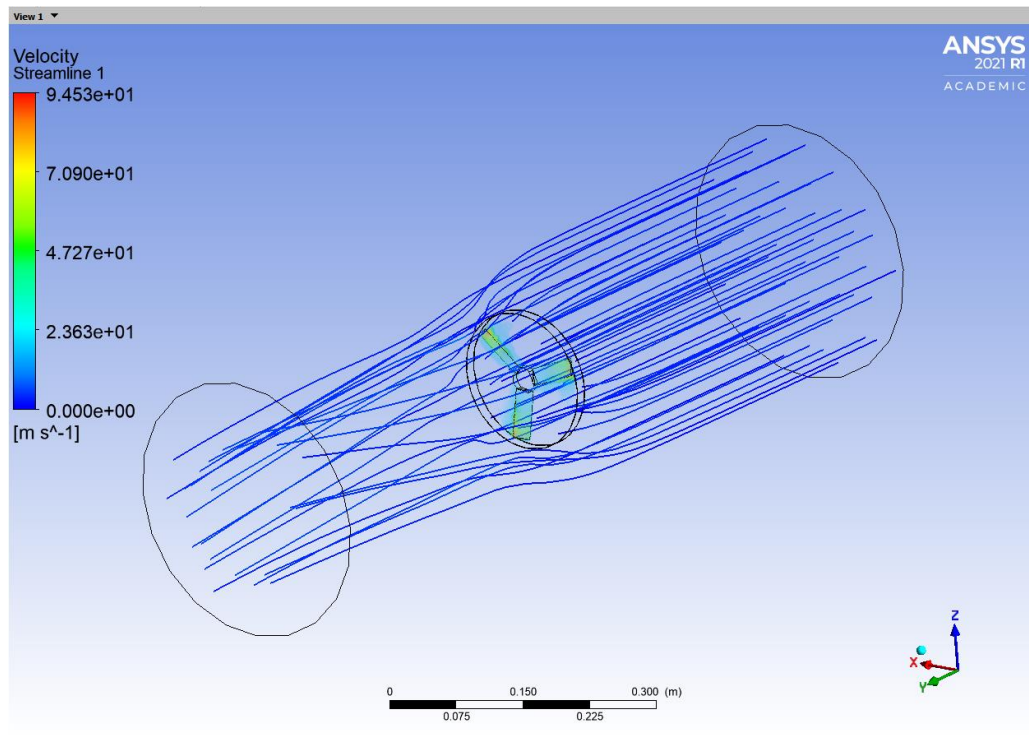


圖 49 200RPS

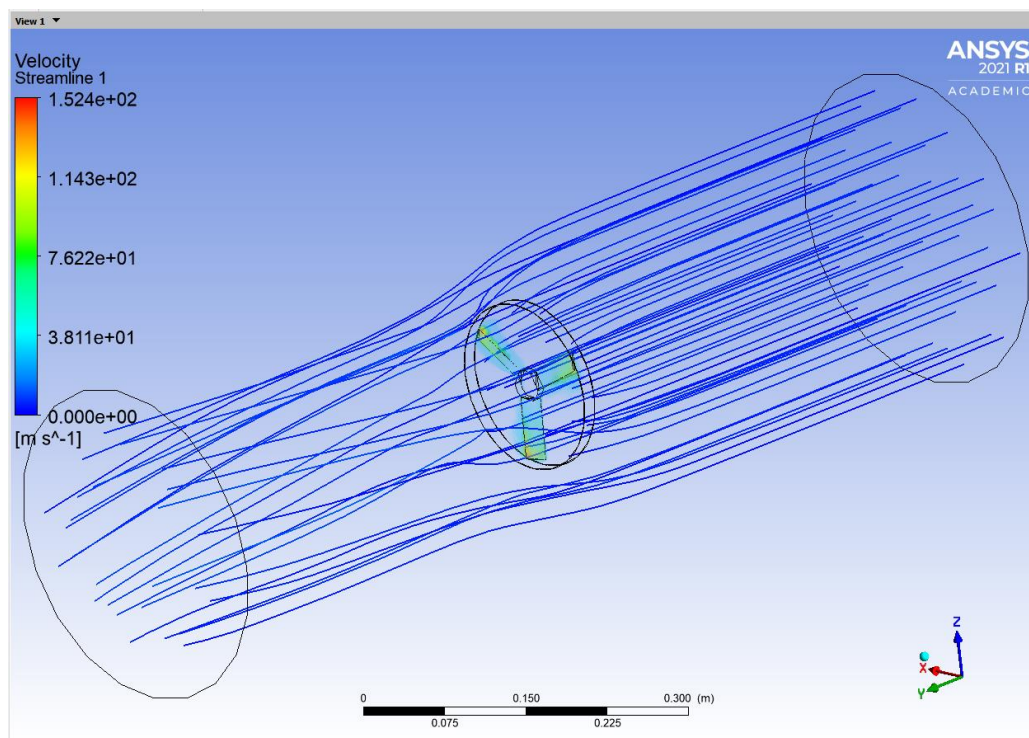


圖 50 300 RPS

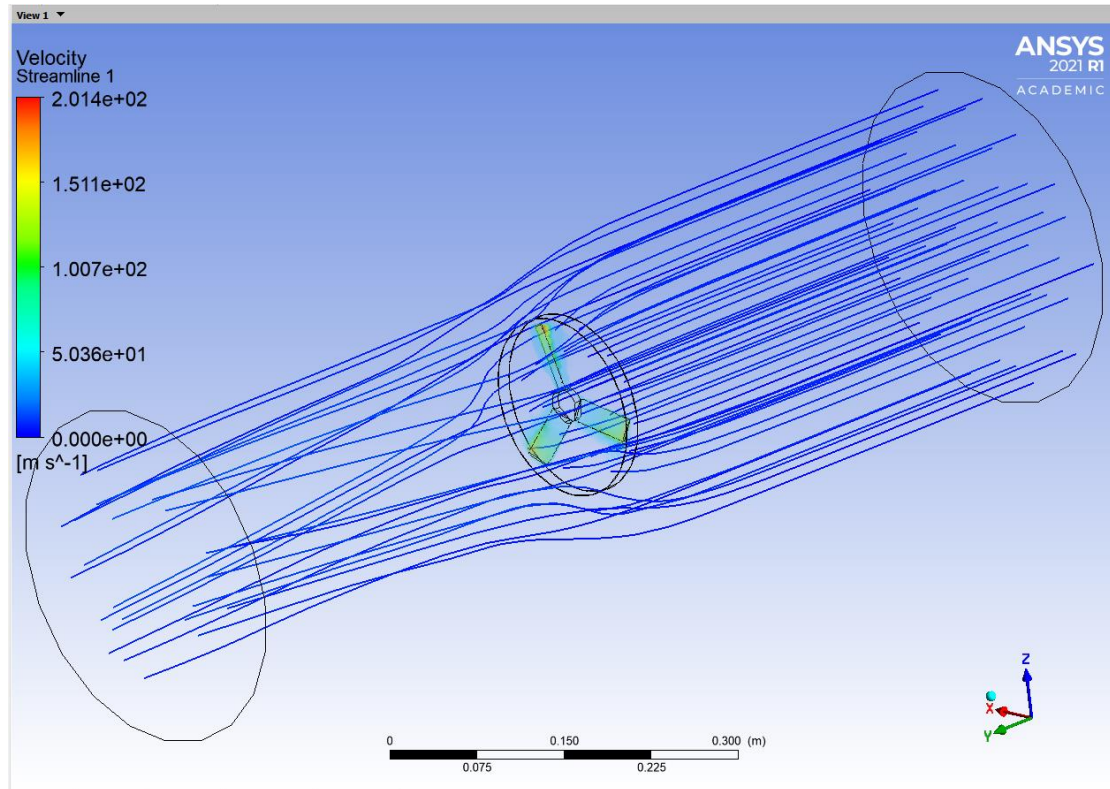


圖 51 400 RPS

三張圖所代表的三個轉速，200rps 推力為 2.279N，300rps 推力為 5.539N，400rps 推力為 8.963N。因為推力與轉速的平方成正比，如下公式，

$$F = 1/2 \rho A V^2$$

計算出回歸二次曲線，將轉速單位 w 換成 V ， $V=wr$ ，得出下公式

$$y = 0.00082x^2 + 0.285x - 3.74899$$

y 為推力， x 為轉速與半徑的乘積，為 $0.1w$ ，常數項的數字相當大，可以理解為風扇在啟動時所會面臨的空氣阻力、起始風速造成誤差等，但同時此迴歸線誤差也相當大，其一因為調整過扇葉旋角，並且 ansys 分析假設的條件為無摩擦的均勻連續流體，與現實情形不符，故而影響到扇葉面積造成誤差。

速度在三個轉速之下，隨著轉速上升，其推力應該要呈現二次曲線的加倍上升，但實際上三個點之間的關係更像是直線，隨著轉速上升，與推力之間不成通過原點之正比。

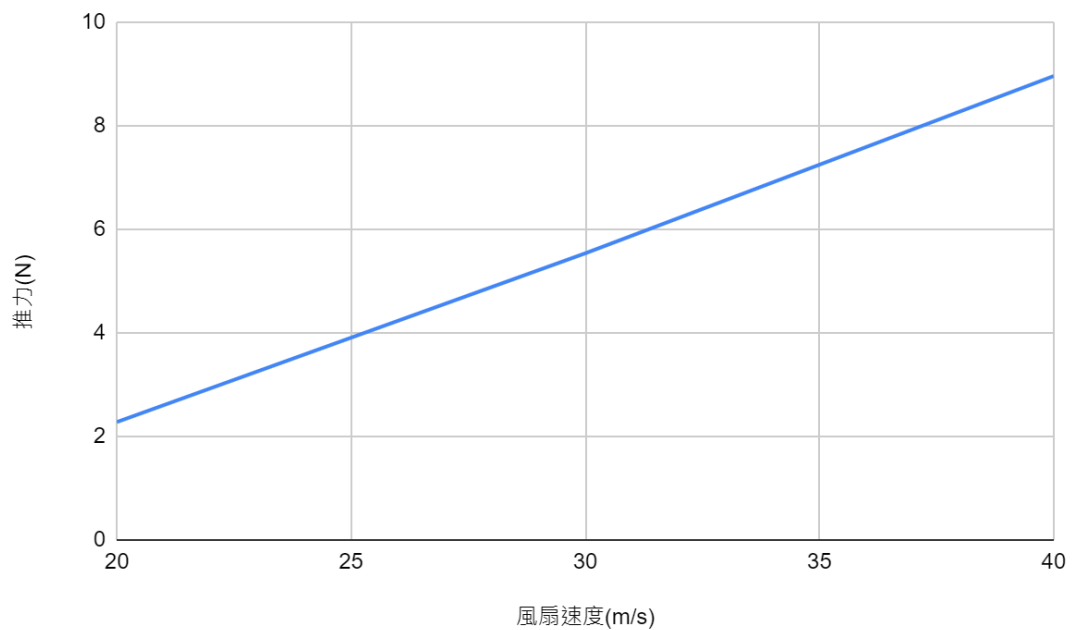


圖 52 風扇速度與推力關係圖

而這也解釋實際在測試時，切換跑道的時候車體會常常就此停住，只要風扇的轉速過低，這時推力便會急速的下降，也要考慮啟動車身的臨界電壓、車體的摩擦力等，故在實測時，若轉彎處轉不過的時候，切不可以將風扇的轉速調低，為了要使推力下降更好轉，而是要調整其他變因，如感應器、輪軸阻力等等。

此外，三張圖中流速線收束的趨勢有但不夠明顯，而期末考的版本，因為倒車為斜坡，所以不需要風扇本身再去生成倒車時的動力，可以將風扇換回扇葉的傾角並沒有調整之前的版本，讓風扇的推力獲得進一步的提升。

4.1-3 扇葉應力分析

由於 3D 列印機能夠輸出的成品尺寸取決於本身的料槽大小，15 cm 的葉片大小並不可行，需要在進行縮減，甚至是將風扇拆成分件後再組合，測試後得到葉片半徑連同中軸的最大不能超過 8 cm，故將葉片長控制在 7 cm 內，預留 2 cm 作為中軸。

下面列出分別為 200、300、400 轉 NACA 風扇所受壓力圖。

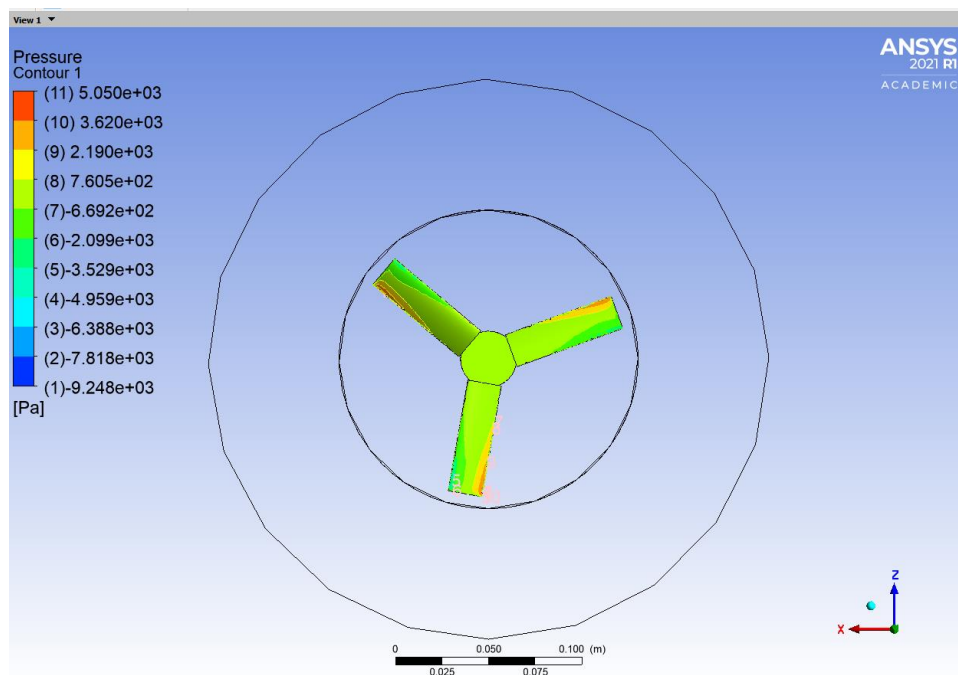


圖 53 200rps 風扇應力圖

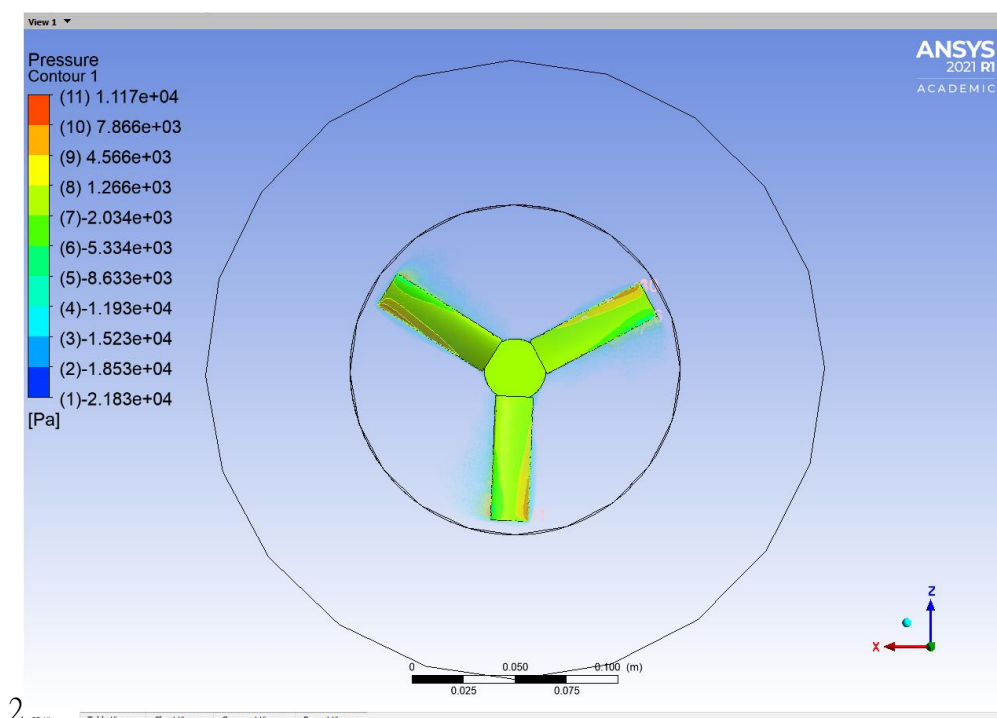


圖 54 300rp 風扇應力圖

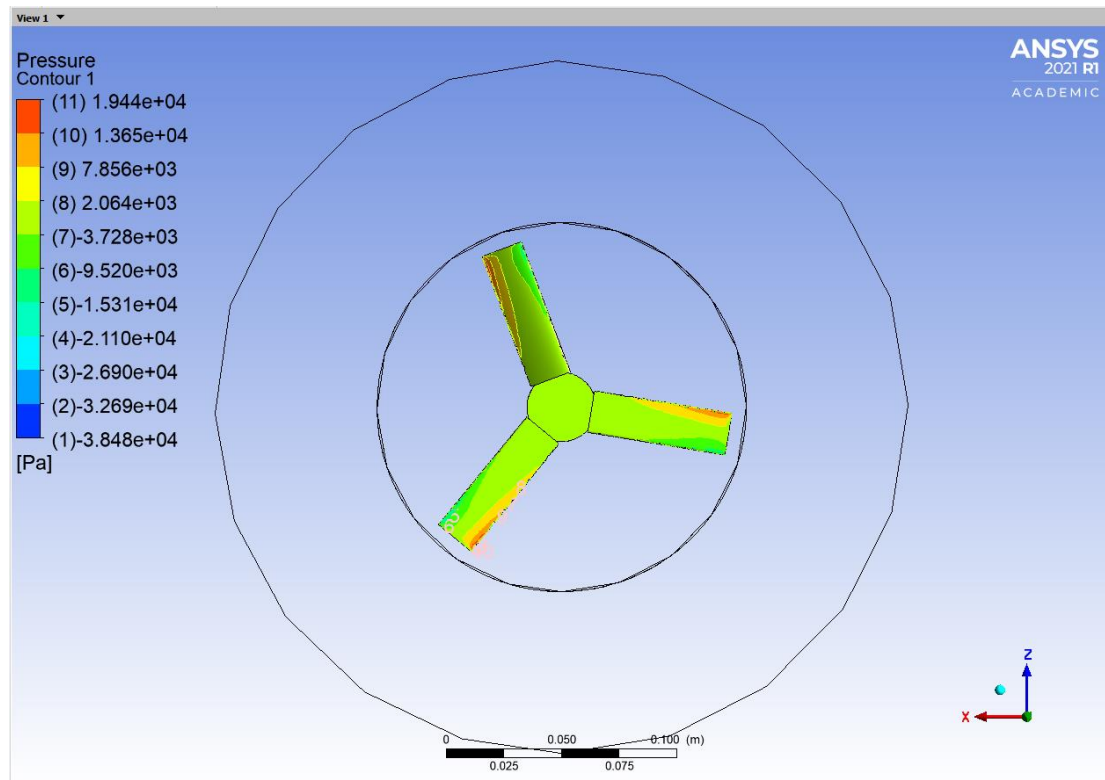


圖 55 400rps 應力圖

三張圖中的壓力，量級上升也同樣略為成正比，可以與風扇流場的分析相互呼應，推出的推力作用於風扇設上面的反作用力分布。

此外，雖然在實際測試上，扇葉本體是能夠承受這樣子的應力，但是以風扇逆時鐘轉，其壓力也多集中於迎風處，有別組的風扇在實測時時常爆炸，在風扇繪製時，也應該要將風扇的迎風面處加厚，同時為符合倒車的需求，故也須於背風面處加厚，但這麼做就會變相的影響其形成的扇葉形狀，進而影響風扇的升阻力。故如果可以，應將扇尖的迎風面，採用更為堅固的 3DP 材質，或是使

其 3DP 的密度更大等。

4.1-4 阻力分析

為簡化阻力分析計算，假設車體受風阻面積為 長 0.290 m 、寬 0.206 m 之矩形，總面積為 0.0574 m^2 ，矩形的阻力係數 $C_D = 1.05$ ，若車體在室溫 25°C

下以 1 m/s 速度等速前進，空氣密度 $\rho = 1.187\text{ kg/m}^3$ ，則車體受阻力大小為：

$$F_D = \frac{C_D \rho A v^2}{2} = 0.0357\text{ N} \quad (7)$$

4.2 繪製工程圖

其他零件的工程圖，則大多數使用 Autodesk inventor 繪製，之所以選擇 Autodesk inventor 是因為與 AutoCAD 相較之下，這套軟體對於 3D 元件的繪製非常方便，而且繪製成品可以直接轉成 3D 列印機使用的 STL 檔或者雷切機使用的 DWG 檔，且大多數組員都在系上課程「工程圖學」以及「電腦輔助工程製圖」中稍有涉獵，因此在協作的方面這套軟體更顯其優勢，因此選擇使用此軟體而非業界常用的 creo 或 solidworks。

4.3 製造

製造的部份，永齡生醫工程館備有 Universal 雷切機以及 Kingsel 3D 列印機，製造過程自動化程度高，因此大多數零件使用這兩種方式製造，缺點是加工精度較低，這裡可大致分為二維與三維元件，二維由雷切機製造，三維則使用 3D 列印機。

4.3-1 車體部分

1. **二維元件：**車體大多經由 3 mm 密集板，經 Universal 雷切機加工而成，輸出工程圖如工程圖所繪製。
2. **三維元件：**車輪、連軸機構中的六角輪座與插銷、軸承座、轉向機構連接座、前輪輪軸固定座等零件因為形狀較複雜，且精度與強度要求不高，選擇由 3D 列印機製造。
3. **特殊元件：**輪軸的部分因為受力較集中，且需要使用螺紋固定，因此選擇使用金屬製造，考量到氧化的可能性與重量、強度等性質，使用 6061 鋁合金製造，用手工鋸修整長度之後，固定於虎鉗上用鑽頭鑽孔，最後再用車床在輪軸尖端車出螺牙。

4.3-2 風扇部分

關於風扇繪製流程下列兩個風扇皆由 3D 列印製成，使用的機器為 Phrozen Sonic Mini 4K：6.1" 4K LCD 光固化 3D 列印機，使用的材料為 Phrozen 湖水模型樹脂。

1. **NACA 翼形風扇繪製：**由於 3D 列印的尺寸限制，需要對風扇進行分件處理，故先從和中軸的接點進行繪製。

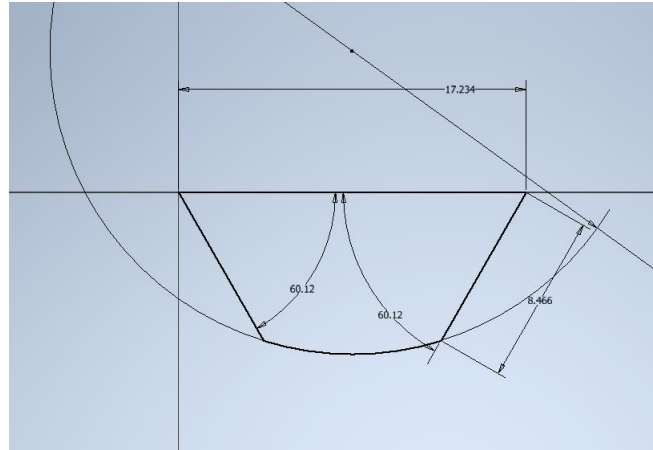


圖 56 風扇繪製流程

有了接點後，建立平面，輸入 NACA 翼形的座標參數進行雲形線繪製得到第一個截面。

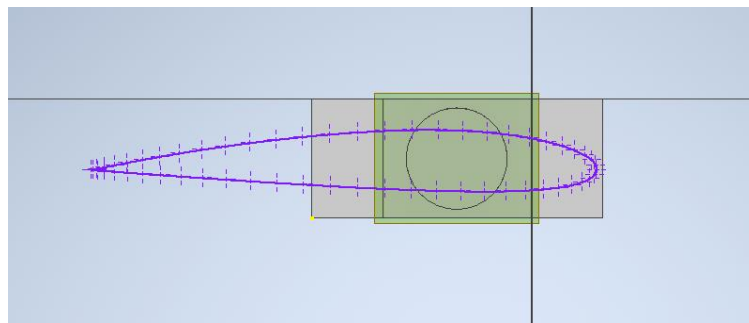


圖 57 風扇繪製流程

接著在半徑最遠端繪製一個截面。

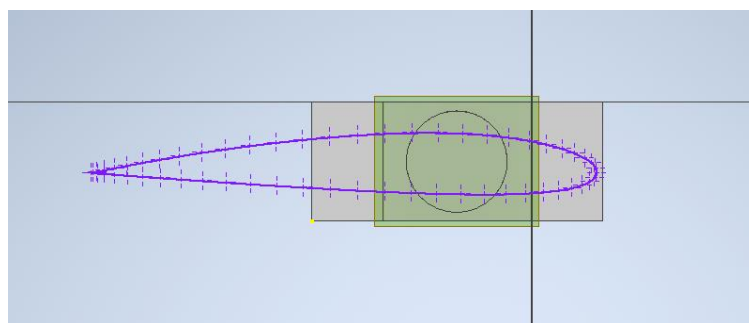


圖 58 風扇繪製流程

接著在半徑最遠端繪製一個截面。

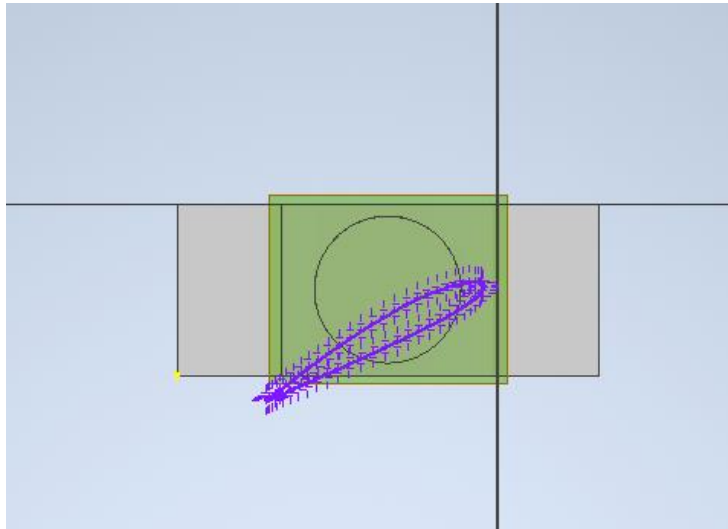


圖 59 風扇繪製流程

最後繪製 3D 草圖，連接兩截面的端點以引導斷面混成建立扇葉。其中兩條可視化的黑線即為 3D 草圖。

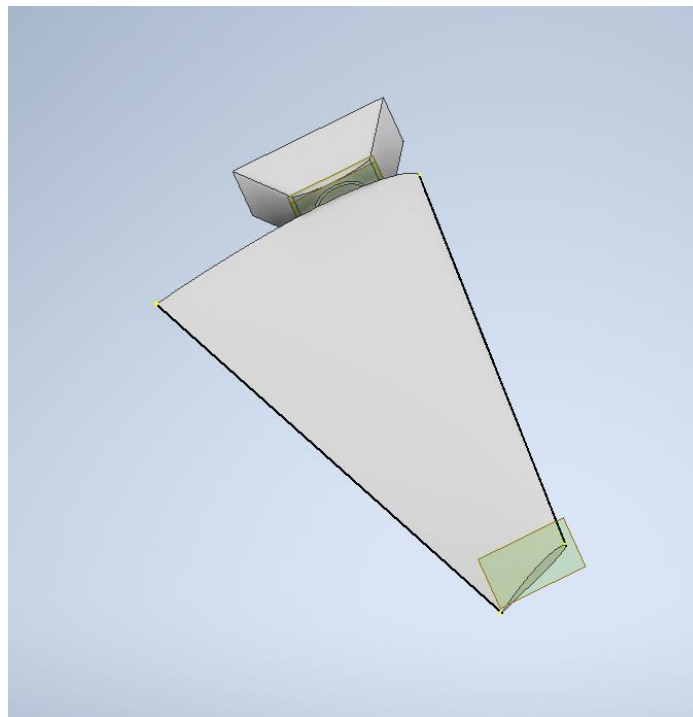


圖 60 風扇繪製流程

中軸的繪製由中軸開始建立草圖後擠出：

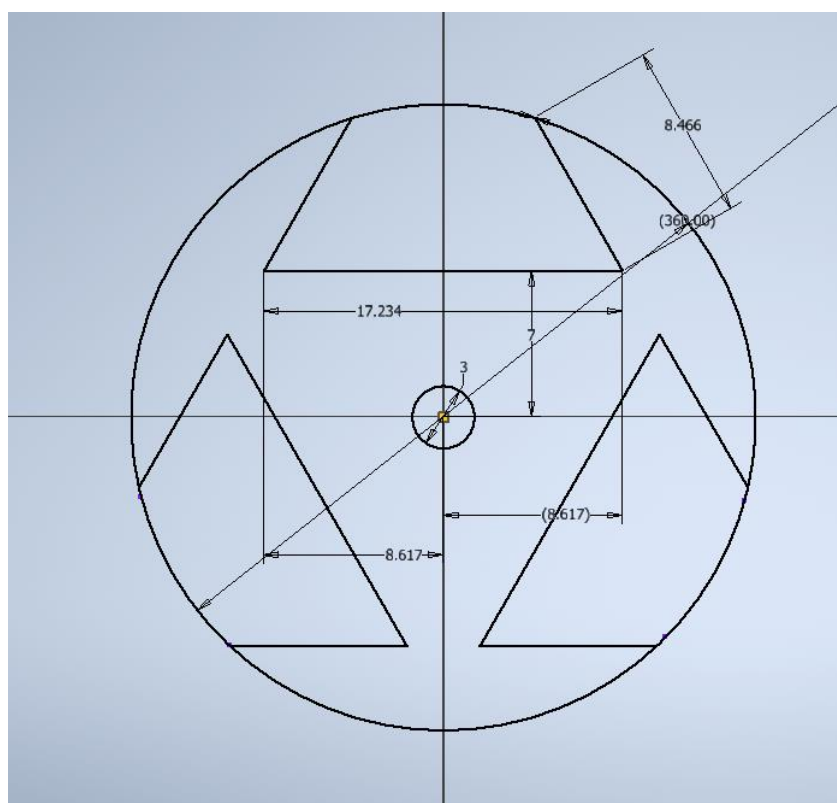


圖 61 風扇繪製流程

擠出後即可得到中軸。

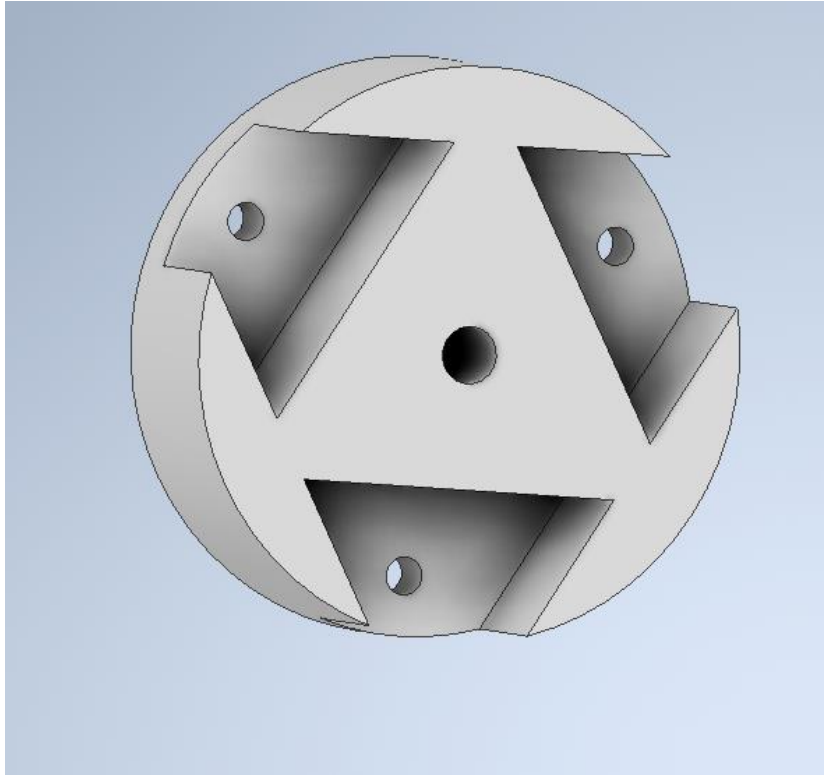


圖 62 風扇繪製流程

2. 類電風扇形風扇繪製：

第二個風扇的繪製也由中軸開始，先決定中軸尺寸：

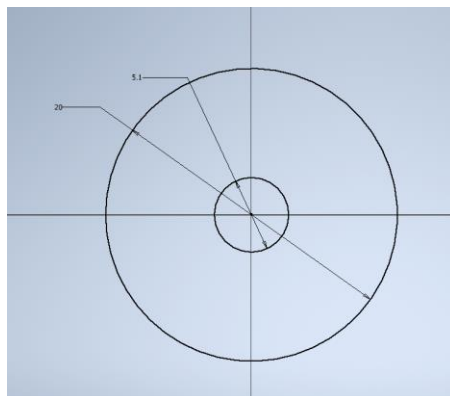


圖 63 風扇繪製流程

建立一個長方形並進行 Coil 繪製，使其產生一個葉面。

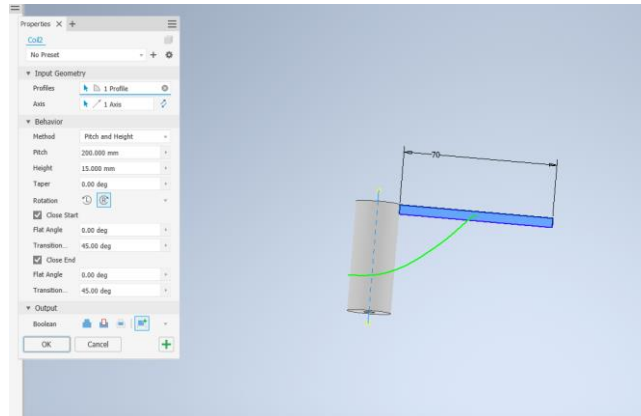


圖 64 風扇繪製流程

繪製完單一葉片後進行圓角修飾，再進行矩陣即完成風扇繪製。

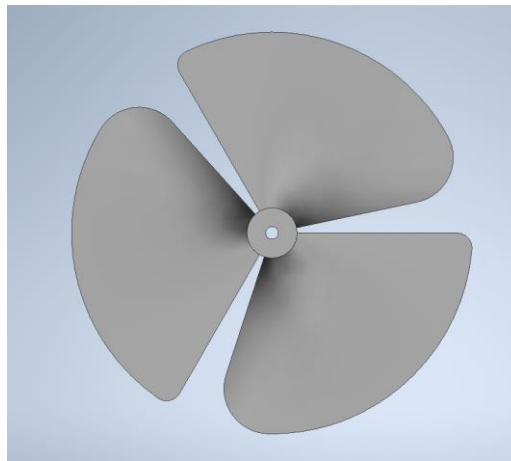


圖 65 風扇繪製流程

4.4 驗證

4.4-1 車體尺寸驗證：

車身整體多用密集板經雷切機切割精度夠高，且對於精度的要求也不大，而零件各部位的尺寸眾多，故只選用其中幾個重要的尺寸量測。

表 10 車體尺寸驗證

編號	設計值	量測值	誤差
車體總長	290.0	290.0	0%
車體總寬	190.0	190.2	0.11%
風扇中心至底座高度	100.0	99.9	0.1%
馬達固定架高度	40	39.8	0.05%
風扇架合併支架長度	88.0	88.1	0.11%
輪桿直徑	5.0	5.0	0%

4.4-2 扇葉：

風扇尺寸確認也是藉由游標尺量測而得，數據如表 N。鑒於 3D 列印品質較難掌控之因素，修整表面後僅有連接支撐材的表面仍有鋸齒狀，扇葉截面大致完好。除尾端截面因厚度偏薄，加上除料時不當操作造成損壞而誤差較大之外，其餘誤差算在接受範圍內，對本專題而言風扇功能仍能符合期待。

1. NACA 翼形風扇

表 11 風扇尺寸驗證

	理論尺寸 (mm)	實際尺寸(mm)	誤差(%)
風扇半徑	74.448	74.012	0.58
圓柱直徑	30.00	30.08	0.27
圓柱厚度	10.00	10.00	0
孔直徑	3.00	3.17	2.6
扇葉最大截面厚度	3.584	3.54	1.22
扇葉最小截面厚度	1.182	1.10	6

2. 類電風扇形風扇

表 12 風扇尺寸驗證

	理論尺寸 (mm)	實際尺寸(mm)	誤差(%)
風扇半徑	48.00	47.56	0.91
圓柱直徑	12.00	12.38	3.1
圓柱厚度	30.00	30.00	0
孔直徑	3.06	3.14	2.6

4.5 實驗

由於新冠肺炎疫情迅速升溫，大多數實驗無法如期完成，因此除金屬輪軸的彎曲程度測試以外，其餘實驗的「結果」部分更改為「數據分析規劃」，即大致說明預期的實驗結果，以及簡單說明實驗中可能出現誤差的來源。

4.5-1 金屬輪軸的彎曲程度測試

1. 目的：

金屬輪軸屬於受力較為集中的部件，除了強度需求較高以外，也較容易產生變形，本實驗旨在實際測量金屬輪軸在受力時應力與變形的關係，以利在設計上因應實驗結果而做出調整。

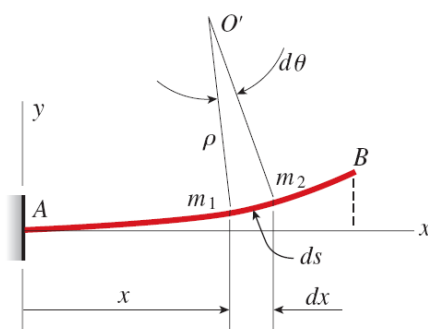


圖 66 樑受力示意圖

2. 原理：

考慮一細長結構如上圖般變形，其曲率為 κ ，曲率半徑為 ρ ，則：

$$\rho d\theta = ds \quad (8)$$

而桿件半徑遠小於曲率半徑，經推導可得：

$$\rho = \frac{M}{EI} \quad (9)$$

其中 M 為力矩， I 為轉動慣量， E 為 Young's modulus。

3. 流程：

用三個虎鉗分別固定住方格紙、手電筒及輪軸樣品，並用拉力計將樣品往上拉，利用光的直進性，分別記錄手電筒在方格紙上投影出的樣品尖端位置與拉力計的讀值。

本次實驗中虎鉗夾住樣品的位置與拉力計固定住樣品的位置相距 $30cm$ ，而轉動慣量為 $\frac{\pi d^4}{64}$ ，輪軸直徑為 $5mm$ ，因此轉動慣量為 $3.086 \times 10^{-11} kg \cdot m^2$ 。

4. 結果：

實驗結果數據如下：

表 13 實驗數據

<i>vertical displacement</i> (mm)	<i>applied force(N)</i>	<i>moment(N · m)</i>	<i>radius of curvature(mm)</i>
17.5	4.7	1.41	513.99
19	5.29	1.58	473.36
26	7.06	2.11	345.71
29.5	8.24	2.47	304.59
33.5	9.17	2.75	268.09
36	10.49	3.14	249.39
44.5	12.4	3.72	201.5
48.5	13.14	3.94	184.75
54	14.37	4.31	165.75

根據實驗結果，可以畫出力矩與位移量的關係圖如下：

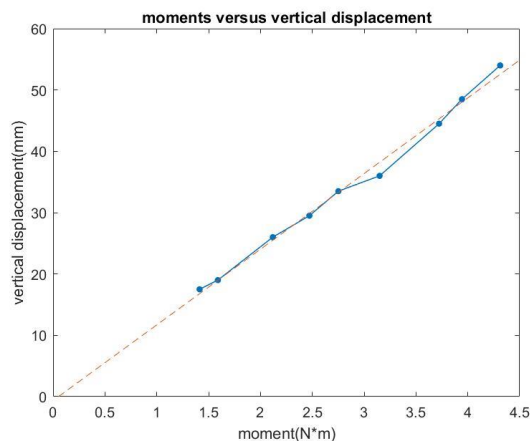


圖 67 力矩與位移量關係圖

可以看到因為角度很小，大致上呈現線性關係，因為角度非常小，因此將鋁棒彎曲形成的弧角度 θ 設為 $\theta = \arcsin(\frac{d}{300})$ ，如此便可以計算曲率半徑為 $\rho = \frac{30}{\theta}$ ，計算結果也已經示於上表中，有鑑於位移量越大，我們的計算偏差就越

大，因此以位移較小的幾組數據來計算 Young's modulus，計算所得的 Young's modulus 約為 89GPa ，與我們在網路上查到的資訊不太符合，這裡懷疑是金屬桿有經過處理，或是組內記錯鋁合金的材質，才造成 Young's modulus 比預期要來的大的情況，不過這代表變形量會比預期的要來的小，在設計上算是正面影響，加上數據分布也還算合理，因此這裡就暫不追究。

4.5-2 風扇推力測試

1. 目的：

期末測試的測試內容中包含上坡的部分，隨著坡度的不同，車體受到重力向下牽引力的大小也會不同，為了驗證風扇的設計可行，必須測試風扇的推力，並推論出其與 PWM duty 的關係

2. 原理：

本實驗牽涉到兩個流體力學常用的原理：白努力定律及連續方程式，即在穩定、非黏性、不可壓縮的前提下，流體質量、動能、位能與內能等物理量皆守恆，基於以上基礎，再以 actuator disk theorem 加以分析。

3. 流程：

大致與量測原理與機工實驗中的實驗流程相同。打開風洞底部的風扇陣列，測量風洞內因底部風扇所造成的背景風速 U_{∞} ，將風力車置於風洞內，並將車上風扇的 PWM duty 固定，在風扇前、後分別選定一固定水平位置，用熱線風速計量測不同高度下的風速，量測時取 10 秒的風速平均值，大略估計風扇推力所造成流體運動狀態改變的影響區，嘗試算出影響區內氣體總動能變化率與流量，進而求出推力。

另一方面，利用風洞內的壓力計監測量測過程中風扇前與風扇後的壓力，經由白努力定律大致計算出空氣的速度變化，並與由風速計算出的推力比較。

整體流程結束之後改變風扇的 PWM duty，重複以上流程及測出 PWM duty 與

推力之間的關係。

4. 數據分析規劃

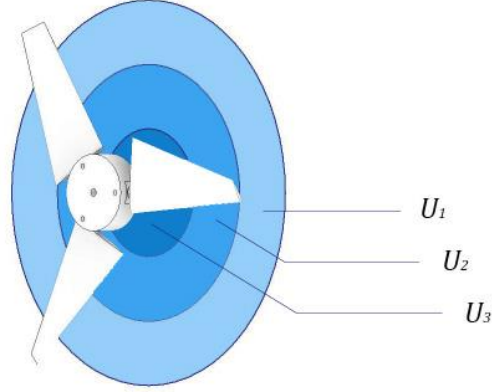


圖 68 風扇流速分區示意圖

如圖 35 中所示，由於量測風速的點為從風扇中軸開始，每向上間隔幾cm便測量一點，而風速會受到測量位置與風扇中軸的徑向距離所影響，因此在計算平均速度時，使用如上圖的假設，將氣體受到風扇影響的區域根據測量的數據點數分為數個環狀區域，每筆數據即代表其位置所對應的環狀區域中的平均流速，而後因假設整體空氣為不可壓縮且密度均勻，因此將測得流速 U_i 與該區域面積 $A_i = \pi \left[\left(r_i + \frac{d}{2} \right)^2 - \left(r_i - \frac{d}{2} \right)^2 \right]$ 相乘，平均速度則為 $U_{av} = \frac{\sum U_i A_i}{A_i}$ ，其中 r_i 為某量測點距離風扇中軸的距離，而 d 為量測點之間的間隔。

從數據中觀察，或是用煙管觀察經過風扇的流線，計算出氣體在風扇前與風扇後影響區的面積 A_1 與 A_2 ，用 A_1 與 A_2 與剛算出的 U_{av} 相乘即可算出氣體的流率，基於質量守恆定律，這裡所算出的風扇前以及風扇後的流率應該不會相差太多，若有一定程度的誤差，其來源可能是空氣被壓縮，導致質量與體積成正比的假設不成立，質量守恆因此無法保證體積守恆，或是氣體的徑向速度較大，而 A_1 、 A_2 與 U_{av} 皆為估算的結果，也難以保證其精準度。

最後將風扇兩端的 U_{av} 相減，便可以算出風扇所提供的速度 $U_{induced}$ ，根據 actuator disk theorem，速度變化量 $U_{induced}$ 與 風扇性質關係為：

$$T = \rho \cdot Q \cdot U_{induced} \quad \text{---(0)}$$

由此便可以算出風扇的推力。

而另一方面，根據白努力定律：

$$\frac{1}{2}\rho v^2 + \rho gh + p = constant \quad \text{---(0)}$$

亦即

$$\frac{1}{2}\rho v_2^2 - \frac{1}{2}\rho v_1^2 + p_2 - p_1 = 0 \quad \text{---(0)}$$

將所有壓力計讀得的壓力值以 MATLAB 做平均後，代入白努力方程式，即可算出風扇前後的流速 v_1 與 v_2 ，將其相減便可以得到另一個速度變化量 $v_{induced}$ ，同樣可以以 actuator disk theorem 算出風扇的推力，可加上風速算出的推力以及 Ansys workbench 算出的推力三者去比較，並根據實際情況去推論最合理的答案。

4.5-3 車體風阻測試

1. 目的：

車體在行進間會受到空氣阻力的作用，為了完整地釐清車體在運動過程中的動態，必須大致預測空氣阻力的量值，並釐清其與其他動態條間之間的關聯。

2. 原理：

本實驗同樣牽涉到兩個流體力學常用的原理：白努力定律及連續方程式，即在穩定、非黏性、不可壓縮的前提下，流體質量、動能、位能與內能等物理量皆守恆，基於以上基礎，再以 control volume analysis 加以分析。

3. 流程：

與量測原理與機工實驗中大致相同，打開風洞底部的風扇陣列，將風力車置於風洞內，並將熱線風速計從風洞側面的槽口放入，在風力車前端與後端，

分別沿著水平與鉛直方向每隔一間隔測量一次風速，測量時取 10 秒的風速平均值，再從前後測得的風速變化，即可求出風力車受到的空氣阻力情形。

全部的流程結束後，改變風洞底部風扇陣列的 PWM duty，再次測量風力車附近的風速，如此便可以測出風力車在不同速度行進下所受的風阻。本次實驗主要目的在於釐清風力車在行進中的動態情形，因行進過程中大部分時間將風力車控制在低速巡航，因此風扇陣列的 PWM duty 也大體以低 duty 的部分為主。

4. 數據分析規劃

與風扇推力實驗中的手法類似，將車體前方的撞風區域依照數據點的位置分成數個矩形區域，再根據測得的數據去計算車體前端與車體後端受到的空氣阻力，與計算風扇推力時不一樣的是，此處的假設較為簡單，各組數據所代表的區域面積皆相同，因此在計算平均風速時只要將各處所測得的風速加總再平均即可，如此即可算出平均上游流速 U_u 與平均下游流速 U_d 。

雖然車體撞風的區域不一定垂直於氣流方向，但若要考慮此因素會造成非常多不必要的勞力輸出，取得的結果精確度因誤差疊加也不見得較高，因此依然假設所有撞風區域與氣流平行，而氣流只有平行於車輛前進方向的分量，計算阻力的公式為：

$$\int \rho U \vec{U} \cdot \vec{n} ds = \int \rho U_d U_d(x, y) dA - \int \rho U_u U_u(x, y) dA + \int \rho U(x) V(x) dx = -F_s \text{---}()$$

在這條式子中，根據我們的假設 $\int \rho U(x) V(x) dx = 0$ ，而前兩項的積分則由面積加總代替，由於密度被假設為一致，代表其不受位置影響，因此上式將被整理為：

$$-F_s = \rho \sum U_u - U_d \cdot A \text{---}()$$

其中 A 為量測點之間的垂直間隔與水平間隔的乘積，也就是每個數據點所代表的面積大小。

如此一來便可以得到風力車所受的風阻力，改變風扇陣列的 PWM duty，求出不同風速下的空氣阻力強度，用 MATLAB 作圖，並與軟體模擬的結果比較，

並搭配實際觀察的結果，即可推論出最準確的情形。

4.5-4 車體起動力測試

1. 目的：

車體在從靜止狀態開始移動前，必須先克服某些程度的摩擦力與阻力，這些阻力可能來自車輪、輪軸或其他部件，例如摩擦力或車輪表面的不平整，這牽涉到風扇需要提供多大的推力才能使風力車起動。

2. 原理：

物體受到的摩擦力分為靜、動摩擦力兩種，其中靜摩擦力作用於物體靜止時，並存在一最大量值，若要使物體從靜止狀態開始移動，必須克服最大靜摩擦力。

除此之外，若車輪的表面有不平整，便會使車輪受到正向力與摩擦力以外的力，使車輪持續受力，此力在起動階段與行進中皆會作用，因此也影響到風力車起動所需的力量值。

3. 流程：

將車體置於一塊木板上，並固定住車體中所有可能移動的部件，將木板一端固定，並將一量角器置於固定端，隨後將木板另一端緩緩提起，使其與地面夾角慢慢增加，直到風力車開始下滑，紀錄此時的角度值

4. 數據分析規劃：

在實驗中測得開始下滑的角度之後，繪製出車體的 free body diagram，假設車體重量為 m ，傾斜角度為 θ ，且車體與木板系統只受重力影響，經由靜力學的推倒可得知車體所受到的向下牽引力為 $9.81m \cdot \sin\theta$ ，而在下滑瞬間，此牽引力的量值即代表使車體移動所需要的起動力。

4.5-5 轉向誤差與阻力測試

1. 目的：

阿克曼轉向機構存在一定的誤差，其誤差很大一部份來自車輪的側向摩擦，而這個力很難量測，但為了使換車道的過程可以更加精準且順暢，若大略得知轉向時車體轉向角與設計時所制定的理想轉向角相差多大以及需要施加多少額外力才能使車體順暢前進，便能在機構無法達成 100% 精準的情況下，從控制上做優化。

2. 原理：

如 3.1-3 所述，阿克曼轉向機構使兩個前輪產生不同的轉向角，使車輛在行進時四輪的軌跡成為一同心圓，如此便可以減少轉向時其中一輪或以上產生滑動，如下圖所示，但這種轉向系統依然存在其缺陷，亦在 3-1.3 中提及過，即不管如何改變連桿的長度與角度組合，都沒辦法使車輪達到完全不滑動，而車輪在地面上受到側向摩擦力作用，便會使車輛實際行進方向與預期產生偏差，甚至於無法預期車輛的行進方向，僅能測得大致上轉向的範圍。

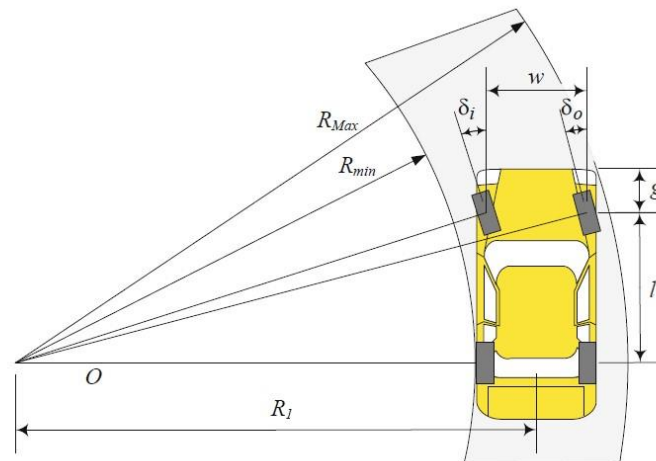


圖 69 轉向原理示意圖

除此之外，從其中測試中我們觀察到，在風力車轉向的過程中，似乎會因為動力傳輸以及車輪摩擦力的緣故，導致所需要的起動力與行進中所需的推進力都

比直走時來得大，這裡也對這些力作粗略的估計。

3. 流程：

將風力車至於平坦地面上，經由 arduino IDE 讓轉向伺服馬達的 PWM duty 固定在某一定值，並逐步增加風扇的轉速，直到風力車開始前進，紀錄此時對電變輸出的 PWM duty，並搭配風扇推力測試的結果，將此結果換算為轉向狀態下的起動力，流程結束後改變不同的轉向角，重複測試以做出轉向角與起動力的關係。

在空曠處，經由 arduino IDE 讓轉向伺服馬達的 PWM duty 固定在某一定值，並讓風力車開始移動，記錄下風力車的迴轉半徑，重複此過程數次，流程結束後改變不同的轉向角，重複測試以做出轉向角與實際迴轉半徑的關係。

4. 數據分析規劃：

經過反覆測試數次（初步暫定為 5 次，若實測後發現離散程度相當大的話，再酌量增加測試次數）之後，整理出對伺服馬達輸入的 PWM duty 與迴轉半徑與起動力之間的關係，配合場地所需的迴轉半徑設定。

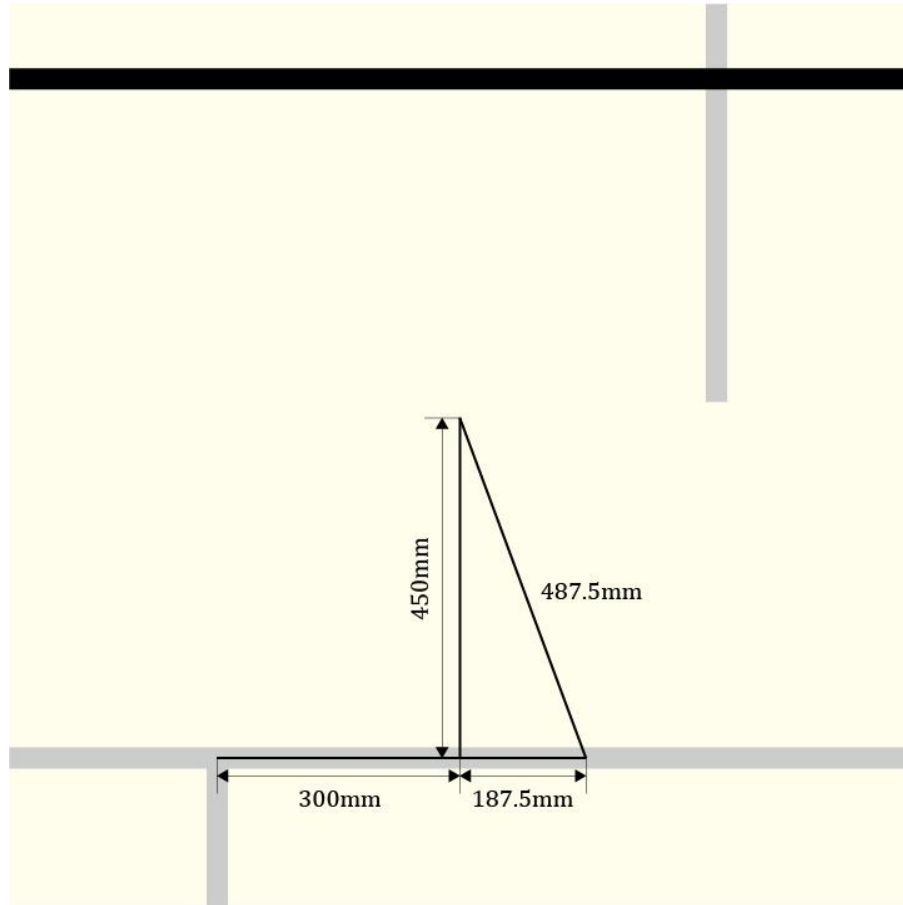


圖 70 最大迴轉半徑示意圖

舉例來說，起跑後第一緩衝區長度為900mm，而起跑時車體中心線與第一循跡線的距離為600mm，因此風力車的軌跡至少在起跑線與跑道拆卸處的中心點需要達成一半的橫向移動，否則剩餘空間不夠讓風力車在離開第一緩衝區之前打直並壓上第一循跡線，若此處的最大允許迴轉半徑為 x ，則可根據條件列出關係式：

$$x^2 = 450^2 + (x - 300)^2 \quad \text{---(1)}$$

可解出 $x = 487.5\text{mm}$ ，因此與實驗數據比對之後，找出對應到最大迴轉半徑 $< 487.5\text{mm}$ 的 PWM duty，將其設為在此處轉向馬達的 PWM duty。

4.5-6 側向支撐力測試

1. 目的：

要驗證在車身轉向時，各項條件是否足夠支撐風力車而不使其滑動或側傾。期末場地的大部分任務都在斜坡上完成，包含在第二緩衝區變換車道，這代表著風力車將會在跑道上斜向前進，風扇所提供的下壓力也會在此時減少，因此需要本實驗來做安全上的驗證。

2. 原理：

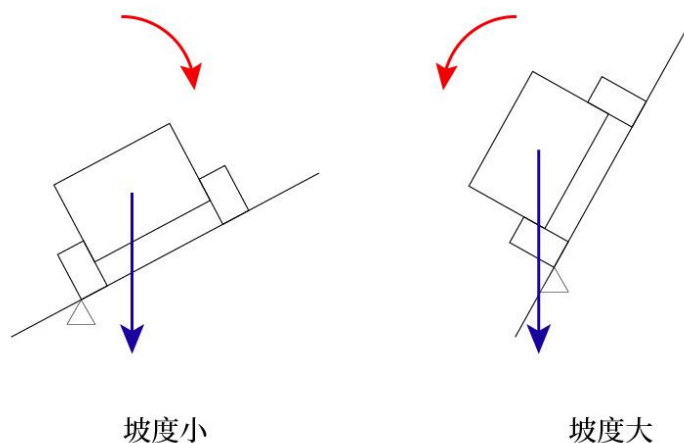


圖 71 車體側傾示意圖

根據風力車的 free body diagram，當車體處在斜坡上時，會由輪胎提供一平行於地面的摩擦力，防止車體產生側滑，而重力則會產生一力矩，較低側的車輪為中軸，若合力矩朝向外側，則車體會產生側傾。

3. 流程：

將風力車橫置於一平板上，並固定住車體中所有可能移動的部件，將木板一端固定，並將一量角器置於固定端，隨後將木板另一端緩緩提起，使其與地面夾角慢慢增加，並隨時注意車體是否有即將要滑動或是側傾的跡象，當滑動或側傾的現象出現，紀錄此時的角度值。

4. 數據分析規劃：

假設斜坡的表面與地面夾角為 θ ，風力車車頭與循跡線夾角 ϕ ，車體重量為 m ，經推導可以推得下滑力為 $mg\sin\theta$ ，其中車輪所受側向力為 $mg\sin\theta\sin\phi$ ，在坡度固定的情況下側向力最大值出現在 $\sin\phi = 90^\circ$ 時，即車身在斜坡上打橫時，如此可以確定將車身橫置於斜面上時確實可以檢驗其可耐受的最大側向滑動力。

再考慮到力矩的部分，假設重心距離外側輪 w ，高度 h ，當 $\phi + \arctan\left(\frac{h}{w}\right) \geq 90^\circ$ 時，車體會產生側傾，因為 w 大致接近車寬的一半，因此可以從 ϕ 去大致回推出 h 的數值。

測得這些數值之後，在控制的策略上便可以避開這個臨界角，確保風力車在行進過程中不會有側滑或側傾的狀況出現。

4.5-7 電池壓降與效能測試

1. 目的：

本實驗旨在觀察 NCR18650b 的放電曲線及其對於馬達效能的影響。鋰電池在使用過程中，電壓會逐漸降低，根據理論所釋放的電能也會逐漸降低，進而影響到風扇的推力，因此釐清鋰電池的壓降情況將能更準確預測風扇的效能。

2. 原理：

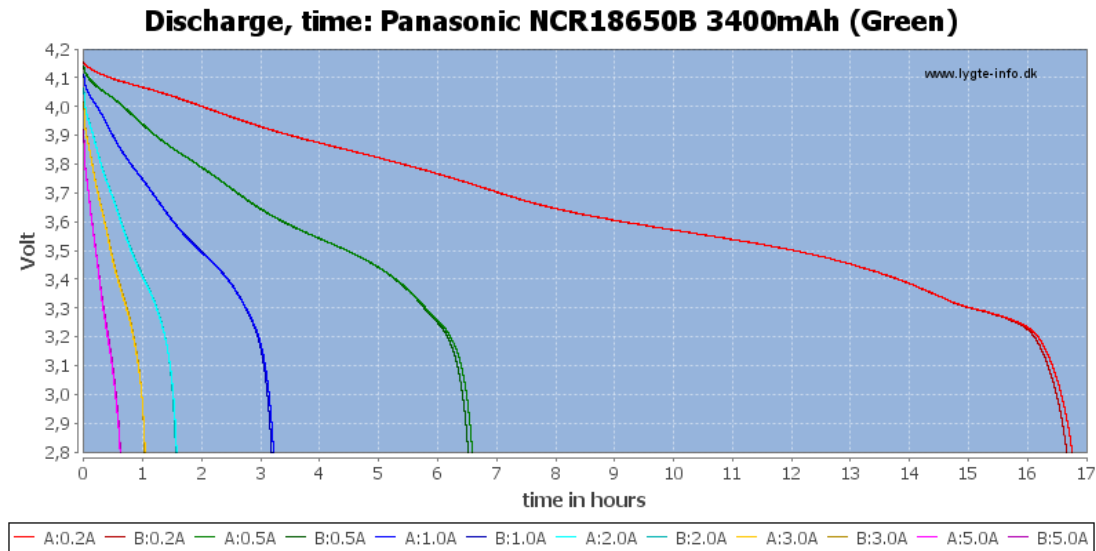


圖 72 NCR18650b 的放電曲線

電池充電時存在兩種模式，分別為 CC 與 CV，本次使用的 ncr18650b 電池充電時先用 CC 模式，在電壓達到某值後轉為 CV 模式，也因此電池充飽時電壓恆為 4.2V，而根據上圖中的放電曲線，電壓值會隨著電池不斷放電而逐漸降低，值得一提的是，降低的趨勢並不是直線而是曲線。

3. 流程：

將電池接上電變及馬達，並將馬達頭裝上一編碼器用的光柵，因轉動速度非常快，用工具將馬達固定於桌面上，並將一編碼器連接 arduino 控制板，並連接至電腦以監控編碼器的轉速，將讀頭靠近光柵，確定其位置不會干擾到光柵轉動並且可以確實讀得訊號之後，用鐵絲固定其位置。

將三用電表轉至 20V 檔位，與電池並聯以監測電壓值變化，並在此時確認電池電壓是否為充電截止電壓 4.2V，準備就緒之後打開電池開關使其開始放電，同時開始計時，記錄下每次電壓產生變化的時間直到電池電壓低至 2.8V，或者是馬達停止轉動。

```

1 int sensor = 11;
2 int steps=0;
3 bool pcb_state=0;
4 bool pcb_state_old=0;
5 unsigned int pulsesperturn = 10;
6 unsigned int rpm; // rpm reading
7 volatile byte pulses; // number of pulses
8 long starttime;
9 long duration;
10
11 void setup()
12 {
13     Serial.begin(9600);
14     pinMode(sensor, INPUT_PULLUP);
15     pulses = 0;
16     rpm = 0;
17 }
18
19 void loop()
20 {
21     if(digitalRead(sensor)==0){
22         pcb_state=0;
23     }
24     else{
25         pcb_state=1;
26         if(pcb_state_old!=pcb_state){
27             duration = millis() - starttime;
28             starttime = millis();
29             steps++;
30             Serial.print("state=");
31             Serial.println(pcb_state);
32             Serial.print("state_old=");
33             Serial.println(pcb_state_old);
34             Serial.println(duration);
35         }
36     }
37     pcb_state_old=pcb_state;
38 }
39
40 }

```

圖 73 轉速監測用程式碼

4. 數據分析規劃：

根據紀錄即可整理出時間與電壓的關係圖，再結合編碼器數據的紀錄，將 arduino IDE 中的數據擷取後做成 csv 檔，再用 MATLAB 進行整理與作圖，兩者耦合之後便可以得出電壓與轉速的關係，將這個關係作圖，以使用於之後的控制工作及測試過程。

4.5-8 煞車力測試

1. 目的：

了解煞車馬達的不同 PWM duty 對於煞車力道的影響。測試賽道的末端沒有任何障礙物，這代表著一旦風力車在第一停止區煞車不及，很有可能會衝出賽道，結果輕則測試結果泡湯，重則車體損壞甚至危害組員與助教、教授的人身安全，在倒車下坡時僅靠風扇推力控制速度效果也有限，因此需要得知精準的煞車力值以確保安全性以及最好的測試結果。

2. 原理：

動摩擦力的計算方式為：

$$f_k = \mu_k \cdot N \quad \text{---(0)}$$

其中 N 為正向力，代表在動摩擦係數不變的前提下，正向力與摩擦力成正比。

根據牛頓第二運動定律，物體受力與加速度的關係為：

$$F = ma \quad \text{---(0)}$$

3. 流程：

將風力車秤重，在空曠處，用 arduino IDE 設定風扇全速運轉 3 秒後停止，接著煞車開始做動，並同時用編碼器開始記錄前進距離，即為該煞車力對應的煞車距離，並在煞車開始作動時同時用碼表計時。

除此之外，在煞車開始做動的瞬間，也用車上的陀螺儀紀錄加速度值直到風力車停止。

流程結束之後改變不同的煞車馬達 PWM 值並重複測試。

4. 數據分析規劃：

煞車過程中車體所受的力主要有三：煞車所提供的摩擦力、其他部件之間的摩擦力與車輪不平整因而受到的作用力、空氣阻力，這裡只有空氣阻力會受到車速影響，若要討論的話難度較高，且扣除掉空氣阻力的影響後，會造成算出的煞車力比實際上要小，也就是預期的煞車距離會比實際所需的距離要來的大，對於測試結果有正向的影響，因此在這裡選擇將空氣阻力忽略。

若煞車距離為 l ，風力車重量為 m ，物體在等加速移動時其位移與加速度的關係為：

$$S = \frac{1}{2}at^2 \quad \text{---(0)}$$

將 l 代入 S ，並將碼表測得的時間代入 t ，搭配牛頓第二運動定律即可整理出煞車力為：

$$F = \frac{2mS}{t^2} \quad \text{---(1)}$$

除此之外，陀螺儀中的加速規也同時記錄下了速度變化的情況，陀螺儀所測得的加速度數據與車重相乘，同樣可以得到煞車力值，不過加速規內部或許存在誤差。將兩種方式所算出的結果比較之後，參考實際情況，推論出最可能的煞車力值。

4.5-9 陀螺儀精確度測試

1. 目的:

此測試目的為了解擺放位置對於陀螺儀讀取換算值與實際測量之角位移的誤差，進而在設計程式時考慮進去，因為這關乎到轉換跑道時的順暢度及精準度。如果誤差大過一定值，即使修改程式參數亦無法減小其誤差，則必須開始思考陀螺儀本身是否損壞以及擺設位置是否有傾斜等等其他因素。

2. 流程:

主要分成兩部分，第一部分為將陀螺儀水平放置於風力車，在風力車直行、轉彎及在坡道直行、轉彎四種情況下，記錄下陀螺儀讀取換算值與實際測量之角位移。第二部分則是將陀螺儀垂直放置於風力車，再重複第一部分的所有步驟。

3. 數據分析規劃：

計算誤差並比較。主要討論的情況有：第一種是比較陀螺儀水平及垂直擺放的誤差比較。第二種是陀螺儀在水平放置、無坡度的情況下，直行和轉彎的誤差比較。第三種是陀螺儀在垂直放置、無坡度的情況下，直行和轉彎的誤差比較。第四種是陀螺儀在水平放置、有坡度的情況下，直行和轉彎的誤差比較。第五

種是陀螺儀在垂直放置、有坡度的情況下，直行和轉彎的誤差比較。最後一種是第四種和第五種的延伸，我們會調整坡度並觀察誤差的變化。

4.6 討論

4.6-1 規則需求

規則中有車體大小不得大於 A4 尺寸(210mm*297mm)的規範，若不符合規範者，則無法通過檢錄環節，因此這裡使用 Autodesk inventor 中的測量功能，測量風力車的 3D 模型的尺寸，如下圖所示：

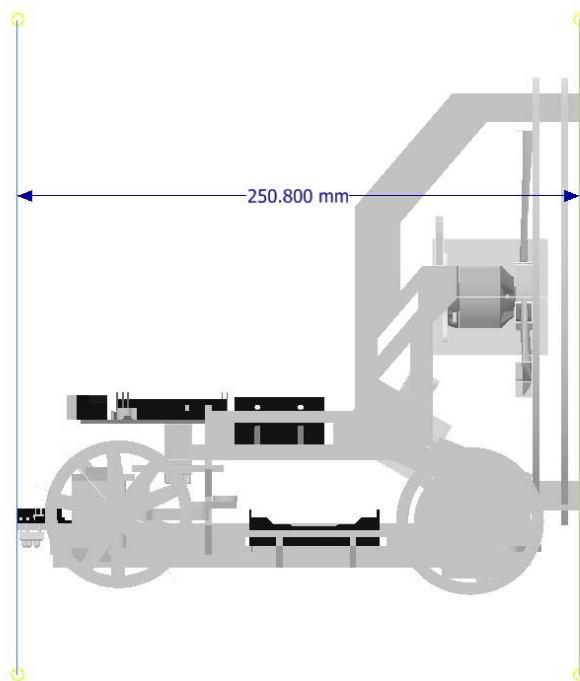


圖 74 車長的測量

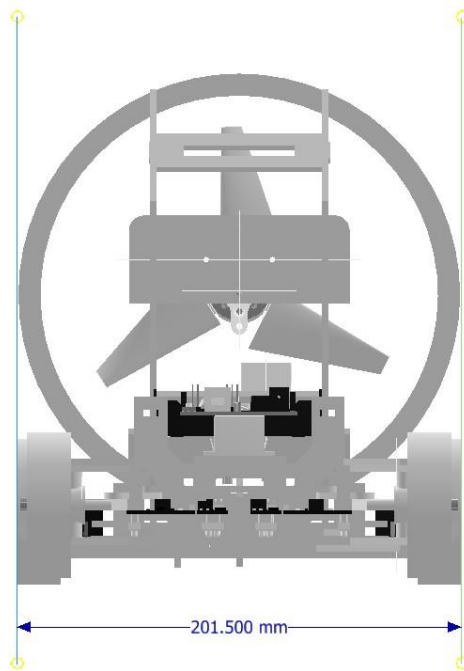


圖 75 車寬的測量

可以看到車長為 250.8mm，車寬為 201.5mm，分別小於規範的 297mm 與 210mm，因此代表風力車可以順利通過檢錄。

4.6-2 功能需求

由於無法實際將風力車放上測試場地測試，因此在這裡會將測試中的主要幾項因素化為參數的形式，與前面的分析結果做比較，推測出如果風力車順利放上測試台的話，是否能夠安全完成測試。

由於煞車、轉向、以及控制系統相關的驗證都無法完成，這裡僅就風扇推力與車體強度做討論。

風扇的部分，在 200rps 時可以輸出 2.279N 的推力，以最小坡度 10° 來討論，若假設車體重量為 m ，在這個角度時車體在斜坡上將受到 $9.81 \times m \times \sin 10^\circ$ 的向下牽引力，與推力互相比較之後，得知在 200rps 時可以推動 1.34kg 的車體，而在 300rps 時推力為 5.539N，可以推動 3.25kg 的車體，就算車體在斜坡上以 45° 的方向斜向移動，向上推力依然有 3.91N，而根據馬達數據，馬達的最高轉速為 24420rpm，即 407rps，期中測試所使用的車體重量約

為 1.5kg，而車體改款之後又做了一定程度的輕量化，由此可知風扇推力要將車體推上斜坡是綽綽有餘。

而根據車體受力分析的結果，車體在測試過程中發生破壞的機會也極低。

總和以上兩點可以得知，在其他功能正常運作的前提之下，分析結果顯示風力車能夠順利通過期末測試的考驗。

五、附錄

5.1 組員心得

我認為這門課作為統整課程，需要大家實際應用過去所學的知識，製造出一個可達到設計目標的成品並進行驗證，是學習過程中很重要的一環。在實際製造的過程中，會遇到各式各樣的現實問題，很多都不是理論課中光是用算的就可以得到答案。一開始在設計的時候時常沒有考慮到製作的時間、成本、精度、配合等問題，比較有經驗後就比較會去注重這些因素，例如盡可能避免使用 3D 列印元件(過於耗時、不易及時修正尺寸)，出工程圖的時候也會以操作機台人員的觀點，思考如何讓圖面的易讀性更好，尺寸標註更直觀。當然，這門課也訓練大家團隊合作的能力。從單純的分工、如何確保進度，到 CAD 軟體檔案的管理與整合，需要花費很多心力才能讓諸多元素不淪為混亂。很可惜這學期因為疫情的關係沒辦法將成品實際做出來，沒有享受到太多實作的樂趣。

而有關課程的建議，我認為大家應該要對安全更重視。不是所有人都接觸過遙控模型，大家不知道模型玩家都會將鋰電池放入防爆袋/防爆箱中保存。如果有人將電池帶回宿舍過夜充電，在充滿易燃物的桌面上起火，後果不堪設想。課程中雖然有提醒同學在測試風扇時，要確定風扇的護罩能確實擋下扇葉碎片，但大家似乎都不是很重視。我也不認為這件事應該是用勸導、提醒的方式，應該強硬規定若有組別在沒有風扇罩的情況下測試風扇，應予以停權。教授可能不清楚在這段期間有多少組別的風扇在高速旋轉下解體，很幸運地沒有人被打到眼睛，若真的發生意外沒有人負得起這個責任。

B07502065 丁映文

在大一第一次直屬聚餐的時候，學長姐就有提過機工實務這門課，或許是學長姊將自己的辛酸隱藏於他們故作開朗的笑容，亦或是不想嚇小大一以及本身大一還不懂事，我竟真的以為這門 2 學分的課的 loading 就像是正常的 2 學分的課。現在想想真的有些天真。我可以很肯定說這門 2 學分的機工實務真的是這學期我花最多時間的課，除了實作本身就需要很多時間製作及測試外，報告這一項作業絕對也是不遑多讓，以一門 2 學分的課來說，這 loading 真的非常大。我這學期 3 學分的數位控制都沒有這麼累。

尤其在期中測試的那一周剛好在期末考周的附近，那時候我的很多同學幾乎整晚都待在永齡。還記得那時候期中測試前一天晚上，幾乎所有同學都待在永齡加緊趕工，我們這組也不例外，就在我們已經累到兩眼發昏的時候，還因為一些小意外不得不離開永齡，那時候就覺得天啊，這堂課真折磨人。但我對這門課還是有愛的，畢竟本身就喜歡實作東西，這些阻礙雖然在當下令人覺得非常煩躁，熬過後還是覺得挺有成就感的。看到大家一起做出來的車真的能完成任務，這種狂喜是真的能讓人高興到整晚睡不著覺。不過很不幸的這個狂喜無法在期末測試時重現，因為疫情我們必須停止實作期末風力車，其實心情挺複雜的。雖然省下了很多時間，但心裡還是覺得有些空虛。而且也增加了報告的難度，畢竟沒有實際操作怎能知道究竟會遇到什麼問題，在打報告時都覺得有點心虛。

最後在這邊想感謝一起同甘共苦的組員們，很高興能跟大家一組完成這門實作課，希望沒有拖大家後腿哈哈。

B07502068 曾兆廷

整學期的課程中，主要負責風扇以及其分析的部分，在學期初的目標時，

我們希望能夠盡快做出成品，並且預留一些時間的彈性讓後期的修改能夠修改，在做出第一代成品後再和其他同學溝通過後，以拼湊的方式，一點一點地加入功能，這樣的好處是能夠不斷的替換不同的角度，讓每次臨時有地方想要修改，就只須要重新繪製一個部件，也不需要大費周章的重新修改。

在測試中遇到的困難，首先硬體方面在做設計時，未充分考慮選用材料的性質，導致最後的成品不如預期，首先，第一代的風扇因為沒有考慮到倒車的情形，以至於沒有特別去調整過風扇的斜腳，讓風扇沒有能夠後退的手段。而第二代風扇則改為使用類似吸排風扇的構造，但是發現推動車子的能力有限，所以才有了現在這一代，第三代是以第一代扇葉各自加上傾角所造成的，能夠逆轉風扇，給予其所需要的推力。而電子零件的問題也是層出不窮，由於受限於經費，不能購買太過昂貴的零件，所以導致零件的汰換率極高，例如：Encoder 常常莫名其妙地壞掉、紅外線也常常今天是好的明天卻壞掉，需要墊高或是整組換掉才能正常使用，而這些問題也讓我們在測試時格外緊張，準備了很多備料，並且模擬測試到前一天的半夜，以防哪裡有突然出現了問題，因此自掏腰包花了不少錢去採購這些材料阿(恨~~

B07502054 余彥亨

當初期中測試踉踉蹌蹌地完成，寫的不怎麼樣的期中報告也交出去了，那段時間應該是我最有動力的時候，滿腦子都是怎麼樣改良設計、怎麼樣讓控制系統寫得更好，我一下課就把自己關在房間裡面拼命的畫工程圖，就在 2021 年的 5 月 13 日，我死命畫出的風力車正式誕生在這個世界上，然後隔天台大就宣布全部改線上，把我們的風力車一腳踹回它老媽的產道裡了，氣死。那段時間則是我這學期，甚至有可能是進入大學以來的這三年裡最沒有動力的時候，但比起沒有動力，我更擔心的是在機工實務宣布改成純報告的時候，我竟然感受到了一點鬆口氣的感覺，這並不會讓這世界上的任何一個人受到影響，不過這種想法肯定是不太對的吧。

怎麼說呢？很多人對於機工實務這門課都有很多不滿，從學長姐的評價跟同學們的反應中都可見一斑，老實說我的心情真的很複雜，假設今天是要我在有很多空閒的情況下，沒有學分、沒有成績，只是單純地做出一台風力車，並幫它加上爬坡、循跡的功能的話，我應該會很開心的完成這項任務吧，畢竟我對於這類事情本來就頗有興趣，不過即使對於課程的內容有興趣，在進度的追趕之下，壓力果然還是滿大的，只能說這門課真的會是我人生中滿特別的一次經驗吧。第一次看到火災現場、第一次在一週之內通宵 3 次以上，感覺就像長輩們談起當兵的事蹟一樣，有笑有淚，但要我再來一次，我絕對是千百個不願意。也並沒有要責怪教授或系上的意思，畢竟家家有本難念的經，真的要說的話我也不討厭這門課，只是看到這樣子的現況，不由得會覺得有點難過吧。

最後來談談組員，一開始原本我們找的是另一批組員，也不知道為什麼學長姐說以前可以 6 人一組，我們就這樣放心到了開學，然後就出事了，其實我不會領導，誤打誤撞地當上組長之後，很多事情也做的一點都不好，但今天的成果真的讓我覺得很開心，感謝我的隊友們，你們就是最棒的。

B07502144 石翊鵬

這兩學分拿得非常充實呢，雖然兩個小時的課堂講座一下子就過去了，但

在課餘我們必須拿出相當多的時間來進行風力車的創作。在造車的過程中學習到了頗多課堂中學習不到的知識，舉凡繪製風扇之扇葉、3D 列印機的使用、流場分析軟體的使用都是在和組員的交流學習中習得的。特別感謝組內成員的努力，有賴於他們積極且自動自發的努力，許多問題都由他們或在他們的協助下得到很好的解決。

值得一提的是，ANYSYS 這個軟體給學生的分析功能非常實用，但要自行摸索期使用方法確實有其難度，或許系上可以考慮納入課程，在學生具有基礎軟體運用知識之下進行分析，想必可以得到更詳盡的結果吧。

B07502077 魏振宇

5.2 工作進度與分配

		周次										
類別	任務項目	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th
設計	工作分配											
	歷屆報告研讀											
	資料蒐集											
	設計發想											
	材料確認、購買											
車身	初版工程圖											
	測試修正											
	車體組裝											
風扇	初版模型											
	風扇分析											
	測試修正											
軟體	程式撰寫											
	馬達控制											
	循跡測試											
	避障測試											
整合	整合測試											
報告	設計概念、介紹											
	原型製作											
	分析測試											
	工程圖											
	報告整合											

7.

		周次						
類別	任務項目	12st	13nd	14rd	15th	16th	17th	18th
車身	期中檢討							
	分析							
	實驗設計							
風扇	期中檢討							
	分析							
	實驗設計							
軟體	程式模擬							
報告	分析測試							
	工程圖							
	報告整合							

5.3 組員互評評分標準

會議出席率 30%

溝通順暢度(ex.回訊息) 30%

分配到的工作之完成度 40%

5.4 控制系統程式碼

```
#include <Servo.h>
#include <PID_v1.h>
Servo esc;          //以 PWM 控制電變
Servo turner;       //轉向馬達
Servo breaker;      //煞車馬達
int sensor = 12;    //encoder 接到 12 號
double tset, tout, tin, Kd, Kp, Ki; //循跡變數的理想值/輸出/輸出
PID trackingPID(&tin, &tout, &tset, Kp, Ki, Kd,
DIRECT); //create PID controller
int ir_pin[]={A1, A2, A3, A4, A5}; //定義紅外線輸入腳位
int ir_threshold[5]={100, 80, 140, 155, 180}; //定義紅外線
閾值
int ir[5]={0, 0, 0, 0, 0};
int velocity=0; //速度參數
int state=0; //階段參數，找車道模式用
int v_high, v_low;
bool bdry=0; //邊界線 boolean，判斷是否在邊界線上
bool bdry_ini=0; //上一次 loop 的 bdry，判斷邊界線 bool 是否改變
int steps=0; //宣告目前行走距離
bool pcb_state=0; //用於判定 encoder 讀值是否有改變
bool pcb_state_old=0; //紀錄原先 encoder 讀值以做比較
bool break_bool;
bool no_line;
unsigned int rpm; //rpm
long starttime;
long duration;
int angle_l, angle_r;
long no_line_time, stop_time;

void setup()        //初始化函數
{
    Serial.begin(9600); //設定鮑率
    esc.attach(9);
    turner.attach(10); //連接轉向馬達
    breaker.attach(11); //連接煞車馬達
```

```

velocity=0;
Kp=300;           //設定 PD 值
Kd=300;
Ki=0;
break_bool = 0;
tset=180; //設定 PID 理想角度為 180
pinMode(sensor, INPUT_PULLUP);
state=0;
rpm = 0;
steps = 0;
breaker.write(160
               ); //給定煞車馬達初始角度
turner.write(90); //給定轉向馬達初始角度
Serial.print("Setting back to low speed! and wait 4
sec! ");
Serial.println("(hearing beep-beep-beep)");
esc.write(90);
delay(4000);
Serial.print("MOTOR IS READY! ");
Serial.println("(hearing regular beep---beep---beep--
- )");
v_low = 83;
v_high = 80;
angle_l = 50;
angle_r = 130;
}

void loop(){
  /*String ans;
  while (Serial.available() == 0) {
  }
  ans = Serial.readString();
  if (ans == "ini"){
    initialize();
  }
  else if(ans == "re"){
  }*/
}

```

```

    if(bdry_ini != bdry && bdry == 1){ //若遇到線則讓 state+1
        state++;
        Serial.print("state=");
        Serial.println(state);
    }
    bdry_ini = bdry; //更新現狀態以做比對
    navigate(); //判定是否處在需要轉彎的區間，若是則進入轉向程序，暫停循跡
    tracking();
    motor_control();
    breaking();
}

void rpm_cal(){
    if(digitalRead(sensor)==0){ //更新 encoder 狀態
        pcb_state=0;
    }
    else{
        pcb_state=1;
        if(pcb_state_old!=pcb_state){ //如果 encoder 狀態與方法末端記錄的不同，代表發生改變
            duration = millis() - starttime; //這裡我也不知道怎麼寫出來的，反正 duration 是與上一柵通過的時間差，單位 ms
            starttime = millis();
            steps++; //讀到一次增加 1
        }
    }
    pcb_state_old=pcb_state; //更新 encoder 狀態
}

void tracking() {
    //Serial.println("tracking");
    bdry = 0;
    no_line=0;
    for(int i = 0; i<5; i++){
        ir[i] = analogRead(ir_pin[i]);
    } //讀入紅外線感測值
}

```

```

    if((ir[0] > ir_threshold[0] && ir[1] > ir_threshold[1]
    && ir[3] > ir_threshold[3])||(ir[4] > ir_threshold[4] &&
    ir[1] > ir_threshold[1] && ir[3] > ir_threshold[3])){
        bdry = 1;
    } //判斷是否在橫線上，若遇到橫線則將
    else if(state == 0){
        tin = 125;
    }
    else if(ir[0] > ir_threshold[0]){
        if(ir[1] <= ir_threshold[1]){
            tin = 255;
        }
        else{
            tin = 223;
        }
    }
    else if(ir[1] > ir_threshold[1]){
        if(ir[2] <= ir_threshold[2]){
            tin = 191;
        }
        else{
            tin = 159;
        }
    }
    else if(ir[2] > ir_threshold[2]){
        if(ir[3] <= ir_threshold[3]){
            tin = 125;
        }
        else{
            tin = 84;
        }
    }
    else if(ir[3] > ir_threshold[3]){
        if(ir[4] <= ir_threshold[3]){
            tin = 56;
        }
    }
    else{

```

```

        tin = 28;
    }
}
else if(ir[4] > ir_threshold[4]){
    tin = 0;
} //依照紅外線讀值判斷角度輸入
else{
    no_line = 1;
}
tout=map(tin, 0, 255, angle_l, angle_r);
turner.write(tout); //依照 PID 運算結果轉向
}

void navigate(){
if(state==1){
    right(17);
    straight(21);
    left(17);
    state++;
    v_high=77;
}
/*else if(state==3){
    v_high = 90;
}*/
else if(state==3){
    Serial.println("accelerate!");
    v_high=80;
    motor_control();
    left(25);
    strtra(8);
    right(25);
    straight(12);
    v_high=101;
    motor_control();
    angle_l = 90;
    angle_r = 90;
    no_line_time=millis();
}
}

```

```

    delay(3000);
    v_high = 111;
    motor_control();
    state++;
}
/*else if(state==4){
    no_line_time=millis();
    state++;
}
/*else if(state==5&&millis()==stop_tim3+3000){
    motor_control();
    if(no_line == 1){
        Serial.println("6666");
        motor_control();
        no_line_time=millis();
        state++;
        breaking();
    }
}*/
/*else if(state==7&&millis(>stop_time+500){
    v_high = 96;
    motor_control();
    state++;
}*/
else if(state==7){
    v_high=83;
    motor_control();
    state++;
}
else if(state==8){
    delay(500);
    v_high=90;
    motor_control();
}
}

void left(int i){

```

```

    int dis_ini = steps;
    int dis = steps;
    Serial.println("turning left");
    while(dis<dis_ini+i){
        motor_control();
        rpm_cal();
        turner.write(60);
        dis = steps;

    }
}

void right(int i){
    int dis_ini = steps;
    int dis = steps;
    Serial.println("turning right");
    while(dis<dis_ini+i){
        rpm_cal();
        motor_control();
        turner.write(120);
        dis = steps;
    }
}

void straight(int i){
    int dis_ini = steps;
    int dis = steps;
    Serial.println("going straight");
    while(dis<dis_ini+i){
        motor_control();
        rpm_cal();
        turner.write(90);
        dis = steps;
    }
}

void strtra(int i){

```

```

int dis_ini = steps;
int dis = steps;
Serial.println("going straight");
while(dis<dis_ini+i){
    tracking();
    motor_control();
    rpm_cal();
    turner.write(90);
    dis = steps;
}
}

void motor_control(){
    /*if(rpm<100||rpm==0){
        esc.write(v_high);
        Serial.print("rpm=");
        Serial.println(rpm);
    }
    else if(rpm<200){
        esc.write(v_low);
    }
    else{
        esc.write(90);
    }*/
    esc.write(v_high);
}

void breaking(){
    if (break_bool == true){
        Serial.println("break!");
        breaker.write(143);
    }
}

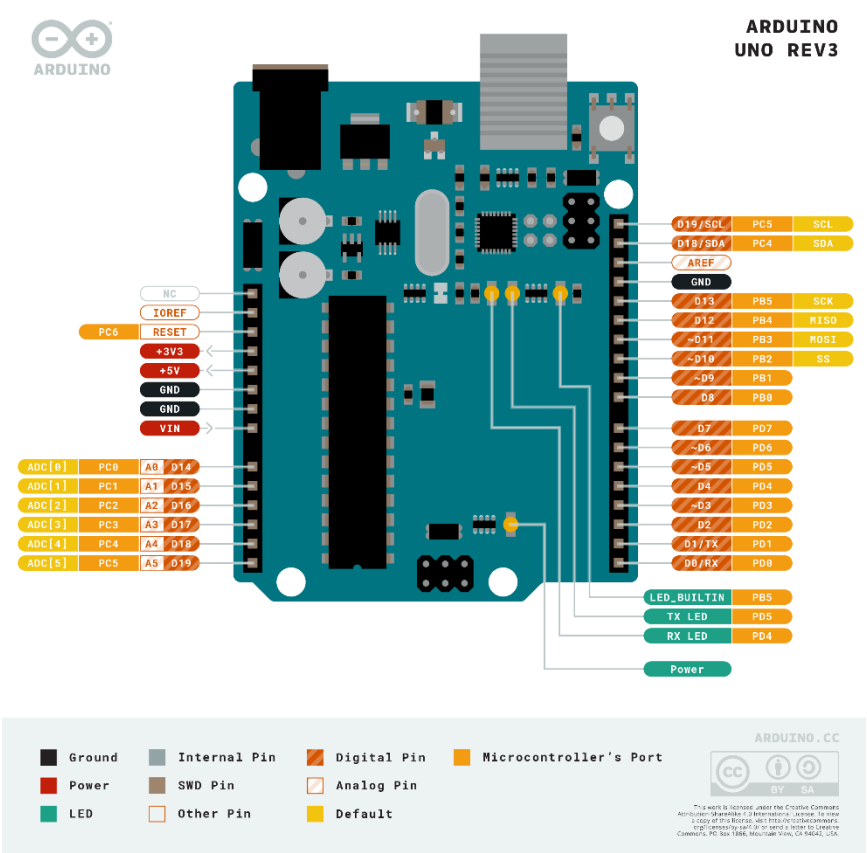
void initialize(){

}

```


5.5 市售產品型錄

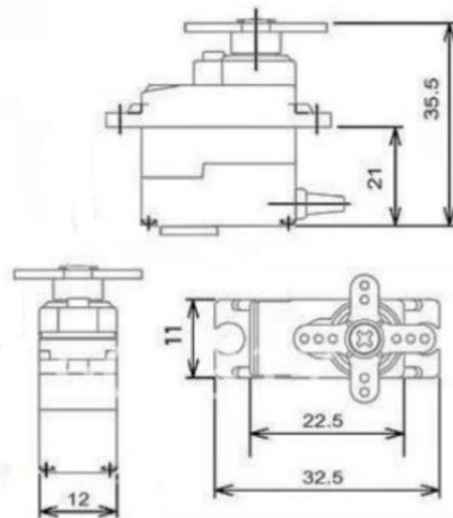
5.5-1 Arduino Uno



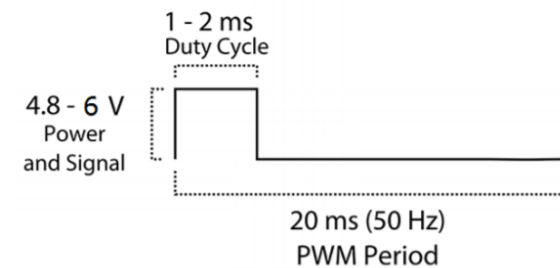
Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

5.5-2 MG90 servo motor

MG90S
Metal Gear Servo



PWM=Orange (⏏)
Vcc = Red (+)
Ground=Brown (-)

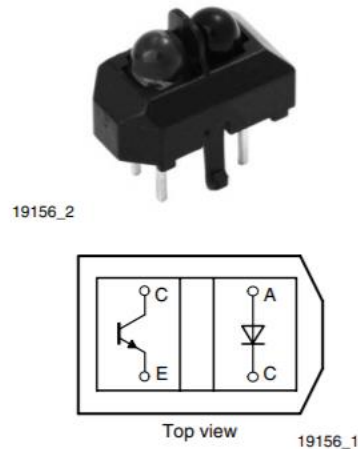


Specifications

- Weight: 13.4 g
- Dimension: 22.5 x 12 x 35.5 mm approx.
- Stall torque: 1.8 kgf·cm (4.8V), 2.2 kgf·cm (6 V)
- Operating speed: 0.1 s/60 degree (4.8 V), 0.08 s/60 degree (6 V)
- Operating voltage: 4.8 V - 6.0 V
- Dead band width: 5 μ s

5.5-3 TCRT5000

Reflective Optical Sensor with Transistor Output



DESCRIPTION

The TCRT5000 and TCRT5000L are reflective sensors which include an infrared emitter and phototransistor in a leaded package which blocks visible light. The package includes two mounting clips. TCRT5000L is the long lead version.

FEATURES

- Package type: leaded
- Detector type: phototransistor
- Dimensions (L x W x H in mm): 10.2 x 5.8 x 7
- Peak operating distance: 2.5 mm
- Operating range within > 20 % relative collector current: 0.2 mm to 15 mm
- Typical output current under test: $I_C = 1 \text{ mA}$
- Daylight blocking filter
- Emitter wavelength: 950 nm
- Lead (Pb)-free soldering released
- Compliant to RoHS directive 2002/95/EC and in accordance to WEEE 2002/96/EC

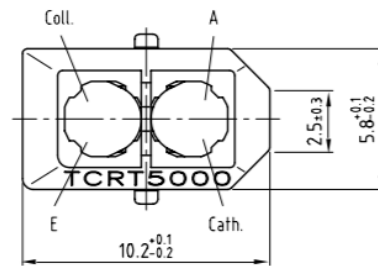
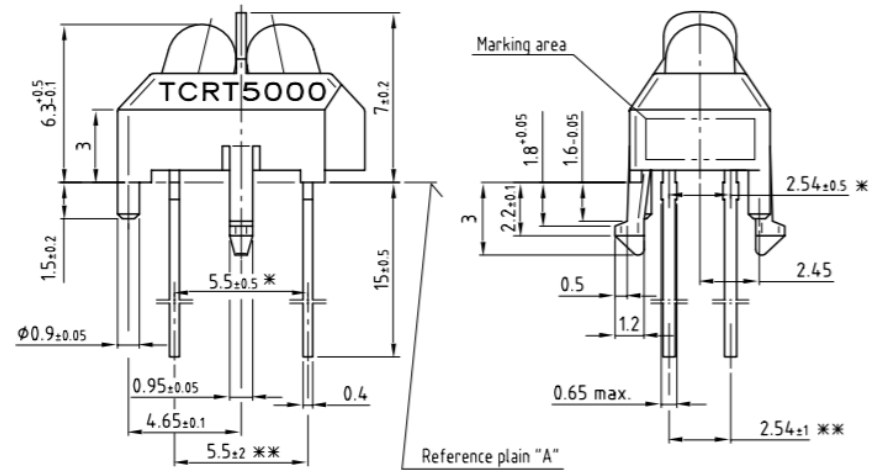


RoHS
COMPLIANT

APPLICATIONS

- Position sensor for shaft encoder
- Detection of reflective material such as paper, IBM cards, magnetic tapes etc.
- Limit switch for mechanical motions in VCR
- General purpose - wherever the space is limited

PACKAGE DIMENSIONS in millimeters, **TCRT5000L**



weight: ca. 0.23g

Drawing-No.: 6.550-5146.01-4

Issue: 4; 11.04.02

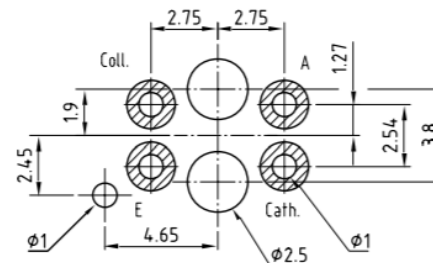
95 11267

* Tolerances related to reference plain "A"

** Tolerances related on lead end



Footprint Top View



5.6 BOM 表

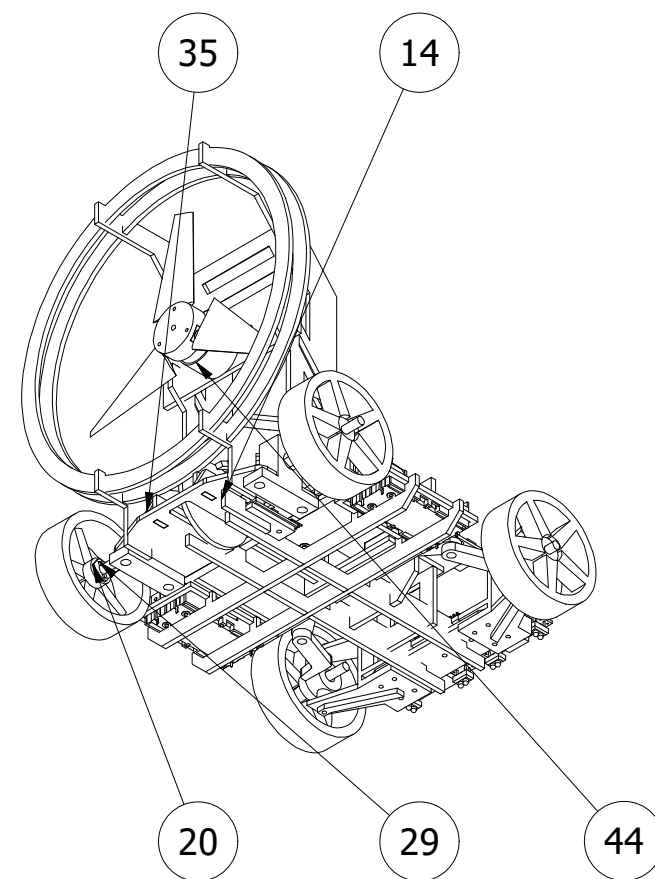
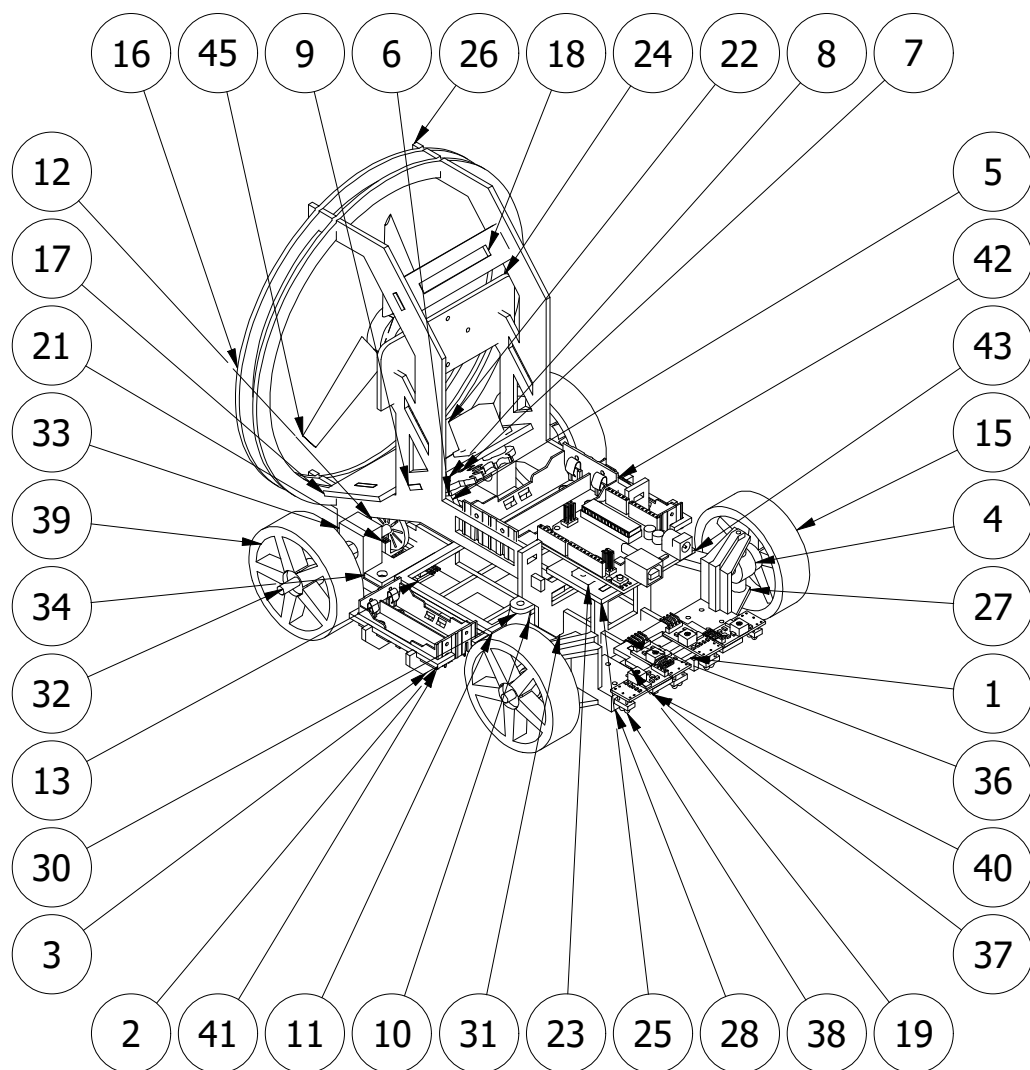
	Item	Part Number	QTY	Unit price	Sum	Material	Resource
Chassis	1	arm fixing plate	1	0	0	3mm MDF	Unversal cut
	2	battery frame	2	0	0	3mm MDF	Unversal cut
	3	battery-support beam	2	0	0	3mm MDF	Unversal cut
	4	front wheel locking plate	1	0	0	3mm MDF	Unversal cut
	5	motor fixing plate	1	0	0	3mm MDF	Unversal cut
	6	rear support fixing plate	1	0	0	3mm MDF	Unversal cut
	7	rear wheel qr	2	0	0	3mm MDF	Unversal cut
	8	servo support	1	0	0	3mm MDF	Unversal cut
	9	tcr5000 support	1	0	0	3mm MDF	Unversal cut
	10	wheel-support beam	2	0	0	3mm MDF	Unversal cut
	11	rear axle support	3	0	0	4mm MDF	Unversal cut
fan assembly	12	fan cover	2	0	0	3mm MDF	Unversal cut
	13	fan protecting board	2	0	0	3mm MDF	Unversal cut
	14	motor plate	1	0	0	3mm MDF	Unversal cut
	15	fan support rib	3	0	0	3mm MDF	Unversal cut
	16	fan	1	0	0	PLA	kingssel
wheel assembly	17	hex connector	4	0	0	PLA	kingssel
	18	pin	4	0	0	stainless steel	cut by hand
	19	phi_5mm bearing	4	15	60	bearing steel	興城街日晟軸承五金商行
	20	M5 centerlock wheel nut	4	0	0	stainless steel	永齡實作中心
	21	wheel	4	0	0	PLA	kingssel
	22	axle fastener	6	0	0	3mm MDF	Unversal cut
	23	wheel skin	1	20	20	PU	耀馬體育用品

	Item	Part Number	QTY	Unit price	Sum	Material	Resource
--	------	-------------	-----	------------	-----	----------	----------

front steering	24	connector	2	0	0	PLA	kingssel
	25	front bearing housing	2	0	0	PLA	kingssel
	26	phi_5mm front wheel axle	2	1.4	2.8	aluminum	Bought at 太原路銘丰鋁業, machined at 台大機械系實習工廠. Adjust length with a cutting machine, then drill a hole precisely using a mill equipped with a drill chuck. Finally, create external threads with a circular die.
	27	servo connector	1	0	0	3mm MDF	Universal cut
	28	arm	1	0	0	PLA	kingssel
rear wheel	29	rear bearing housing	2	50	100	aluminum	興城街日晟軸承五金商行
	30	phi_5mm rear wheel axle	1	7.3	7.3	aluminum	Bought at 太原路銘丰鋁業, machined at 台大機械系實習工廠. Adjust length with a cutting machine, then drill 2 holes precisely using a mill equipped with a drill chuck. Finally, create external threads with a circular die.
braking system	31	brake disk	1	0	0	3mm MDF	Universal cut
	32	break base	1	0	0	3mm MDF	Universal cut
	33	breaking plate	1	0	0	3mm MDF	Universal cut
	34	brake gear (right)	1	0	0	3mm MDF	Universal cut
	35	brake gear (left)	1	0	0	3mm MDF	Universal cut
encoder	36	encoder sensor support	1	0	0	3mm MDF	Universal cut
	37	Lm393 Motor Speed Measuring Sensor Module	1	57	57		Online shopping
	38	encoder sensor holder	1	0	0		
	Item	Part Number	QTY	Unit price	Sum	Material	Resource
encoder	39	encoder disk	1	0	0	3mm MDF	Universal cut

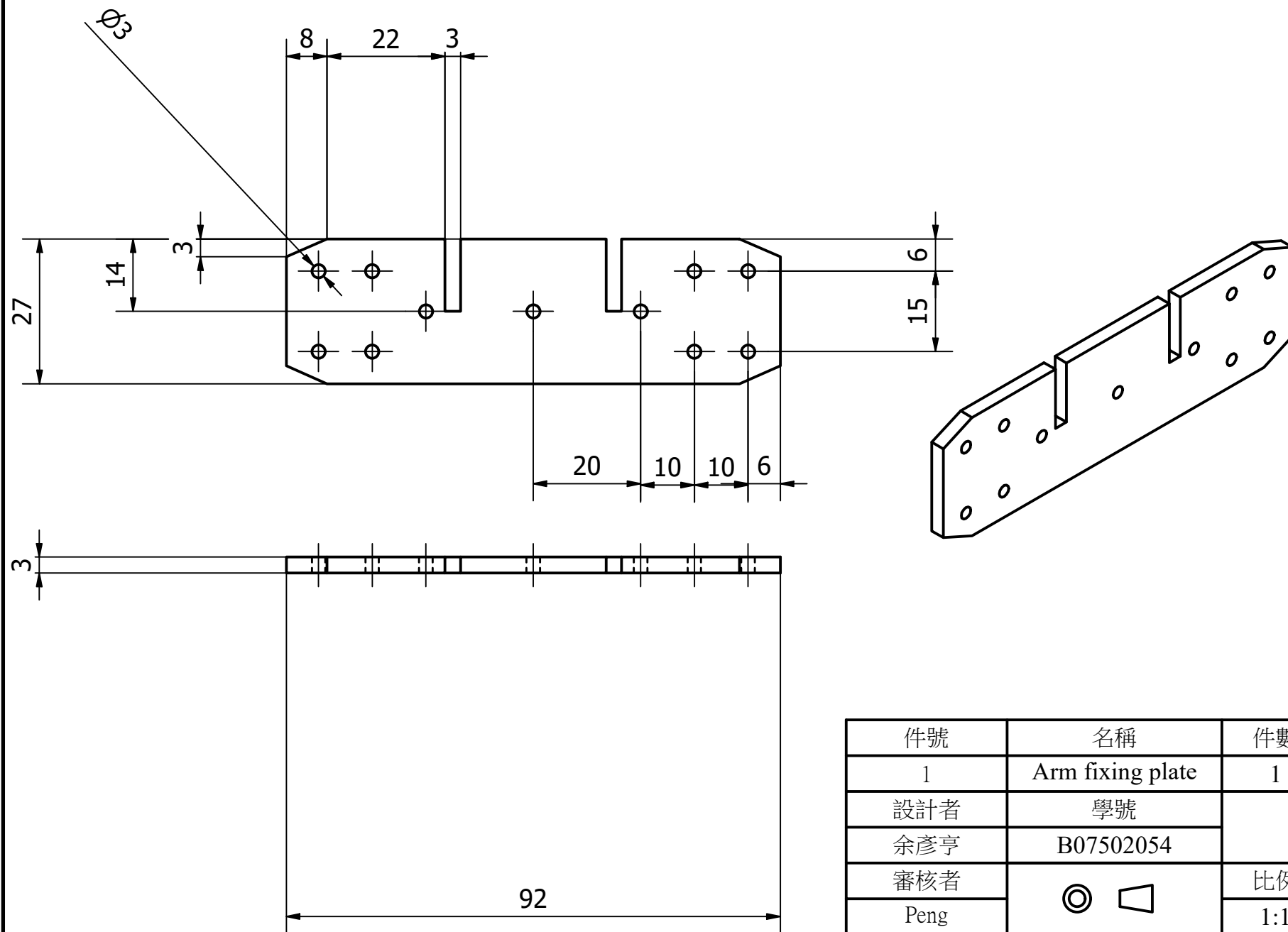
battery	40	Panasonic NCR18650B battery, 3.6V, 3200mAh	3	280	840		Online shopping
	41	18650 battery box	1	43	43		Online shopping
	42	Panasonic NEO AA battery 1.5V	4	11	44		祥昌電子
	43	1.5V battery box	2	6	12		祥昌電子
sensor	44	IRS90 infrared ray sensor	4	50	200		Online shopping
	45	MG90 servo motor	2	170	340		Online shopping
	46	T shaped plug	1	35	35		Online shopping
	47	Arduino Uno Microcontroller Board	1	302	302		Used
	48	breadboard	1	50	50		Used
	49	seaking 40A double speed control	1	300	300		Online shopping
	50	D2826 - Turnigy 2200KV brushless motor	1	360	360		Used
	51	mpu6050 motion tracker	1	50	50		祥昌電子
	sum				2823.1		

5.7 所有工程圖

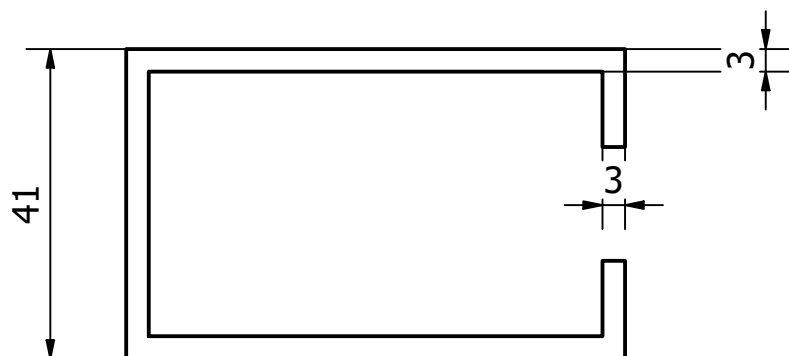
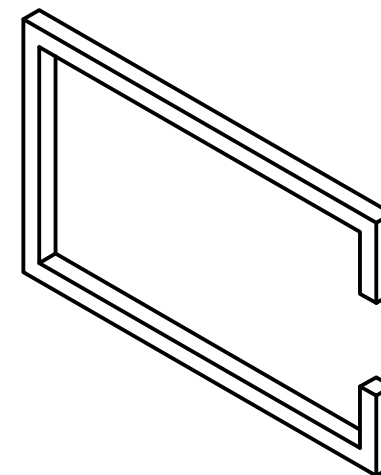
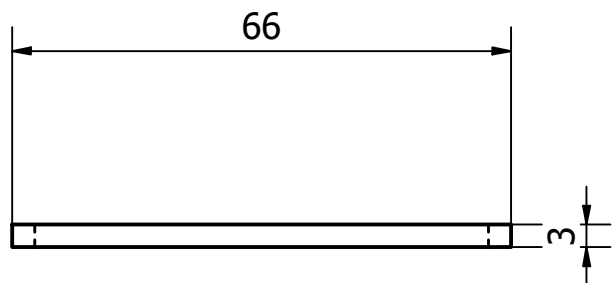


設計者	學號	國立台灣大學 機械工程學系	
第26組	B07502144		
審核者		比例	日期
石翊鵬		1:4	2021/5/11
圖名			
風力車			

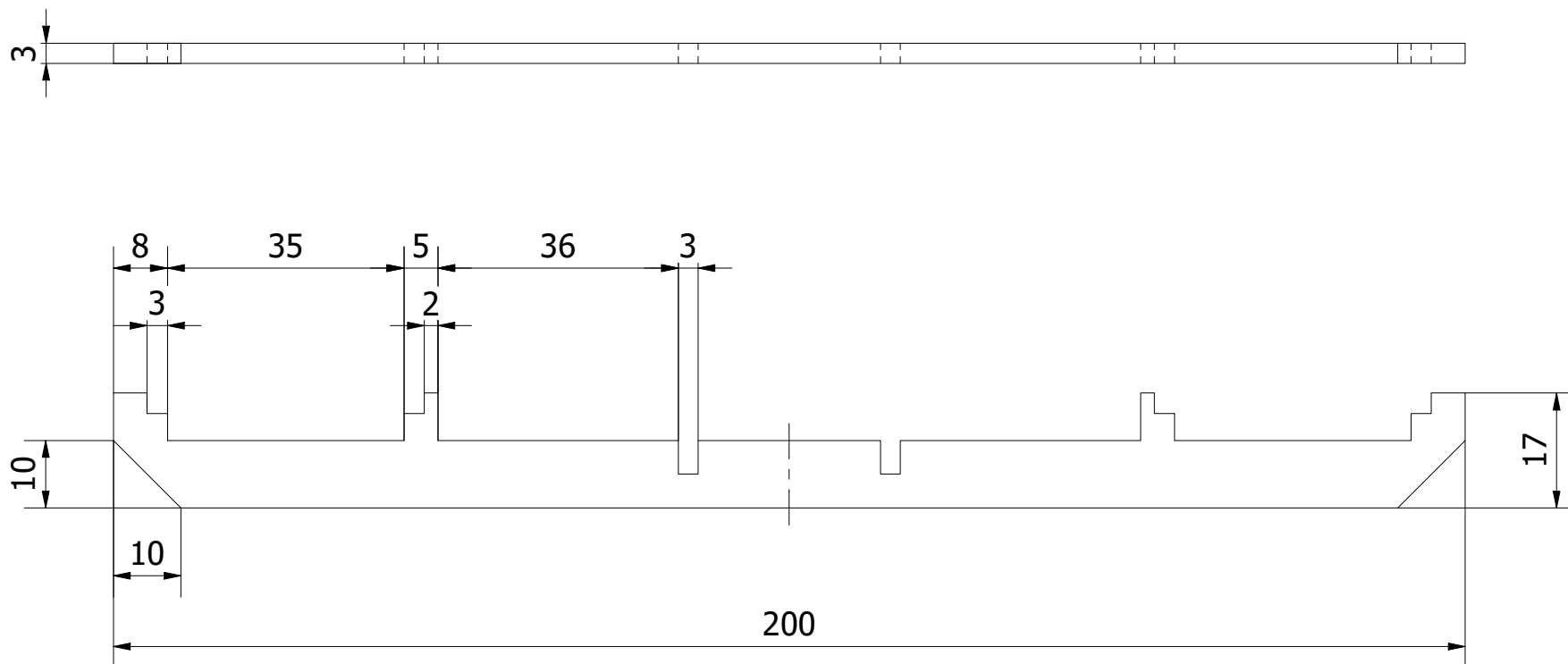
零件表				零件表				
料件	數量	零件號碼	描述	料件	數量	零件號碼	描述	
21	1	Lm393 Motor Speed Measuring Sensor Module	市售	1	1	arm fixing plate	密集板	
22	2	MG90 Servo	市售	2	2	battery frame	密集板	
23	1	motor fixing plate	密集板	3	2	battery-support beam	密集板	
24	1	motor plate	密集板	4	2	bearing housing	PLA	
25	2	motor_support	密集板	5	1	brake disk	密集板	
26	1	new arm left	PLA	6	1	brake base1	密集板	
27	1	new arm right	PLA	7	1	braking gear	密集板	
28	4	pin	不鏽鋼	8	1	braking gear2	密集板	
29	1	rack connector	密集板	9	1	braking plate	密集板	
30	1	rack holding plate	密集板	10	2	connector	密集板	
31	1	rear axle 2	6061鋁合金	11	2	driver_frame	密集板	
32	3	rear axle support	密集板	12	1	encoder disk	密集板	
33	1	rear support fixing plate	密集板	13	1	encoder holder	密集板	
34	2	rear wheel qr	密集板	14	1	encoder support	密集板	
35	1	servo support	密集板	15	2	fan cover	密集板	
36	1	tcrt5000 support	密集板	16	1	fan protecting board	密集板	
37	4	TCRT5000mod	市售	17	1	fan support rib	密集板	
38	4	wheel	PLA	18	2	front shaft	6061鋁合金	
39	2	wheel-support beam	密集板	19	1	front wheel locking plate	密集板	
40	2	AA battery case	市售	20	4	hex	PLA	
41	1	18650 battery case	市售	設計者		學號	國立台灣大學 機械工程學系	
42	1	Arduino Uno	市售	第26組		B07502144		
43	1	D2826 2200KV brushless motor	市售	審核者			比例	日期
44	1	fan	PLA	石翊鵬			1:4	2021/5/11
				圖名				
				風力車				



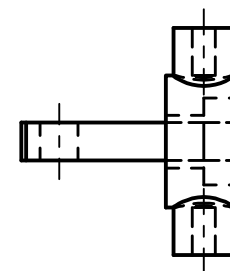
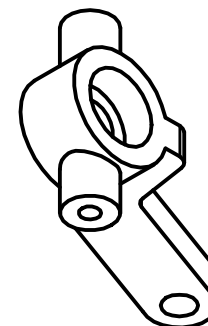
件號	名稱	件數	材料
1	Arm fixing plate	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
余彥亨	B07502054		
審核者	◎ ◻	比例	日期
Peng		1:1	2021/5/11



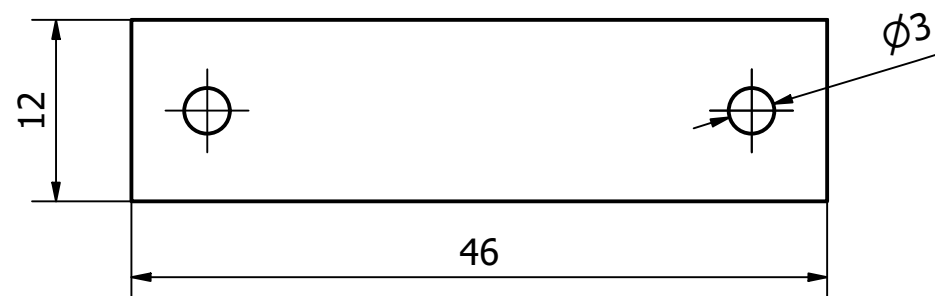
件號	名稱	件數	材料
2	battery frame	2	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	b07502077		
審核者	◎ ◻	比例	日期
魏振宇		1:1	2021/5/12



件號	名稱	件數	材料
3	battery-support beam	2	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502144		
審核者	◎ ◻	比例	日期
石翊鵬		1:1	2021/5/12

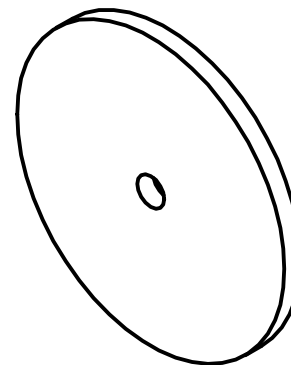
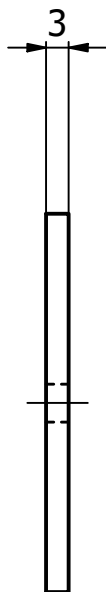
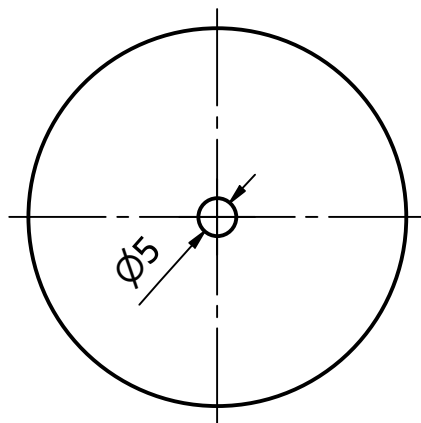
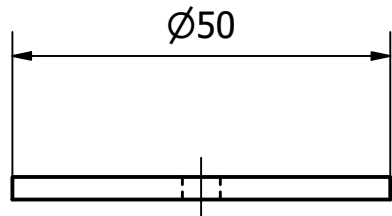


件號	名稱	件數	材料
4	bearing housing	2	PLA
設計者	學號	國立台灣大學 機械工程學系	
TingTing	B07502054		
審核者		比例	日期
余彥亨		1:1	2021/3/29

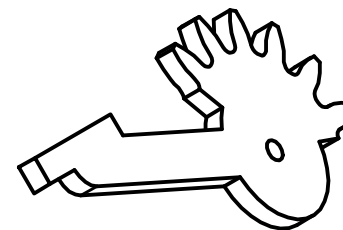


(密集板厚度為3mm)

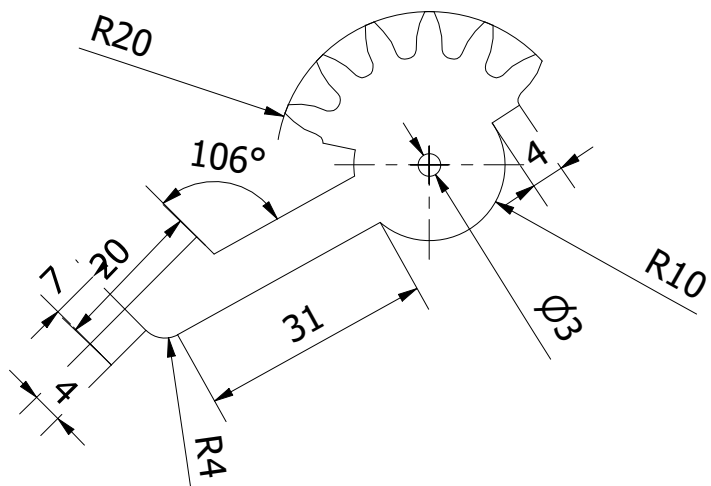
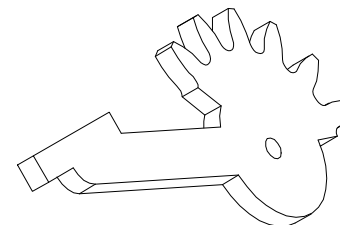
件號	名稱	件數	材料
5	Brake Base	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
丁映文	b07502065		
審核者	◎ ◻	比例	日期
丁映文		2:1	4/12/2021



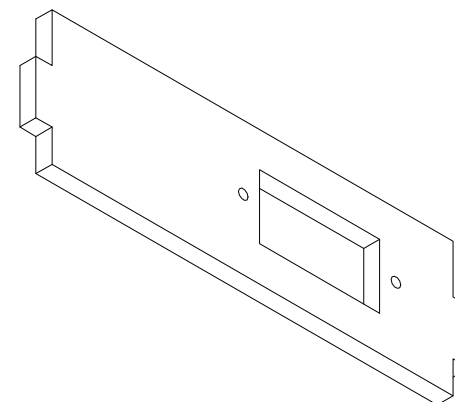
件號	名稱	件數	材料
6	brake disk	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	b07502077		
審核者	◎ ◻	比例	日期
魏振宇		1:1	2021/5/12



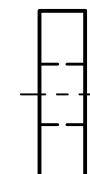
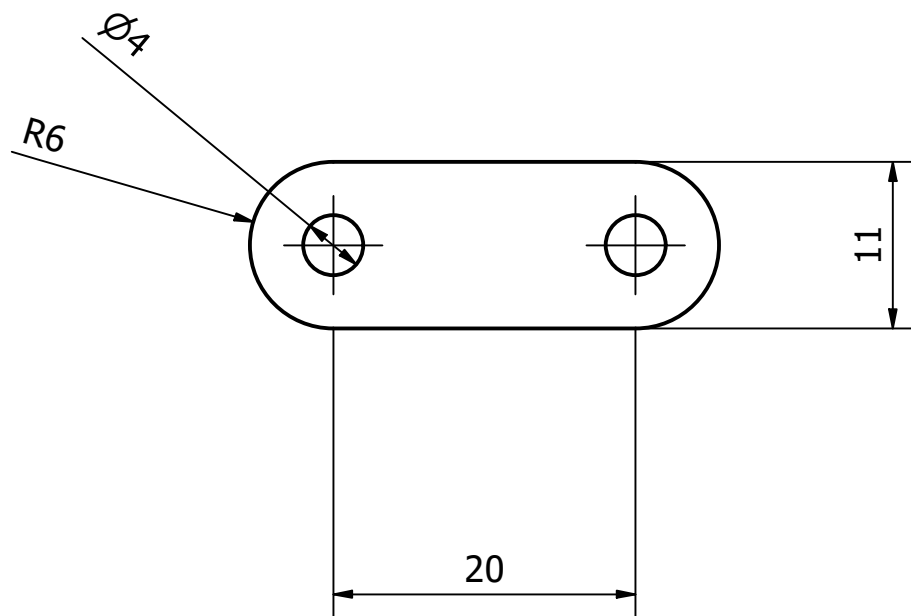
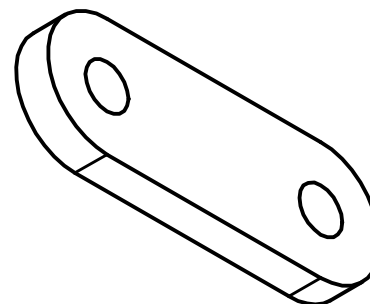
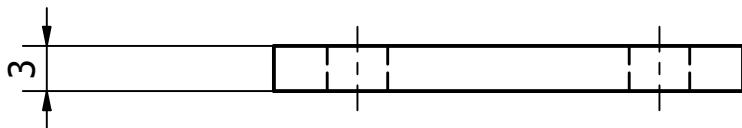
件號	名稱	件數	材料
7	braking gear	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
Peng	B07502054		
審核者		比例	日期
余彥亨		1:1	2021/3/27



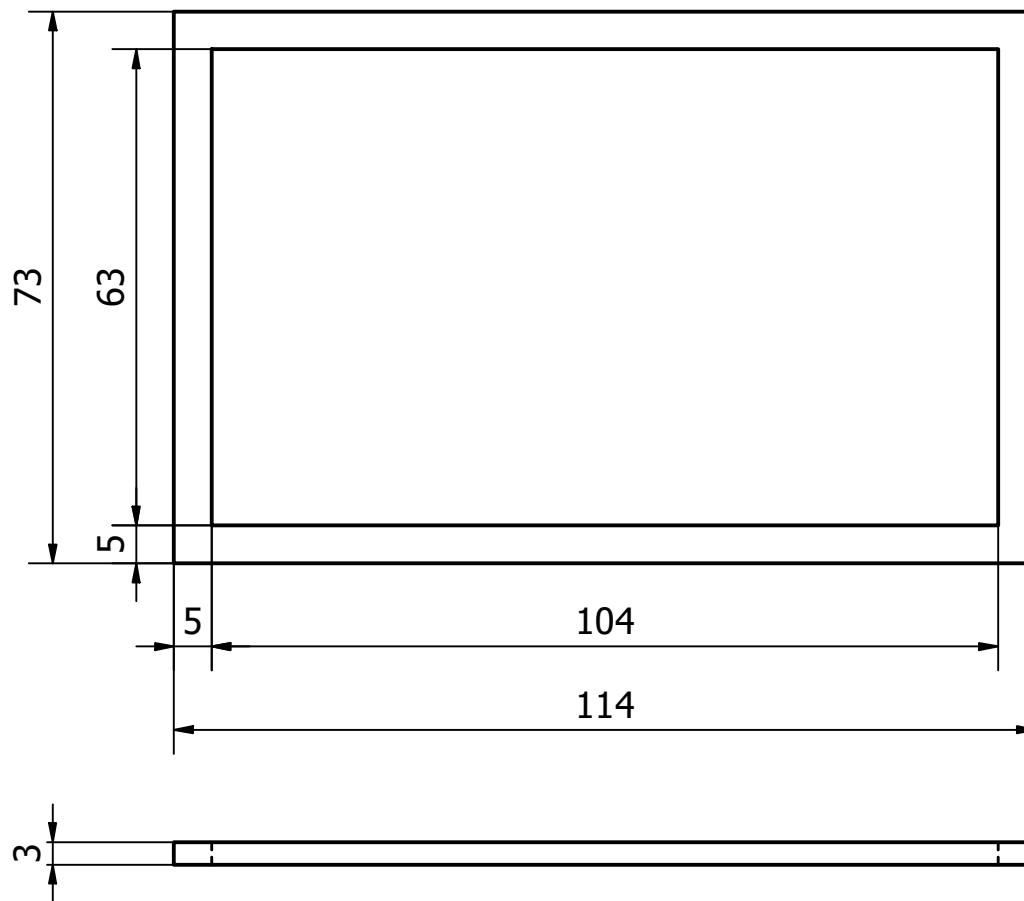
件號	名稱	件數	材料
8	breaking gear2	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502144		
審核者		比例	日期
石翊鵬		1:1	2021/3/27



件號	名稱	件數	材料
9	breaking system plate	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502144		
審核者		比例	日期
石翊鵬		1:1	2021/5/14

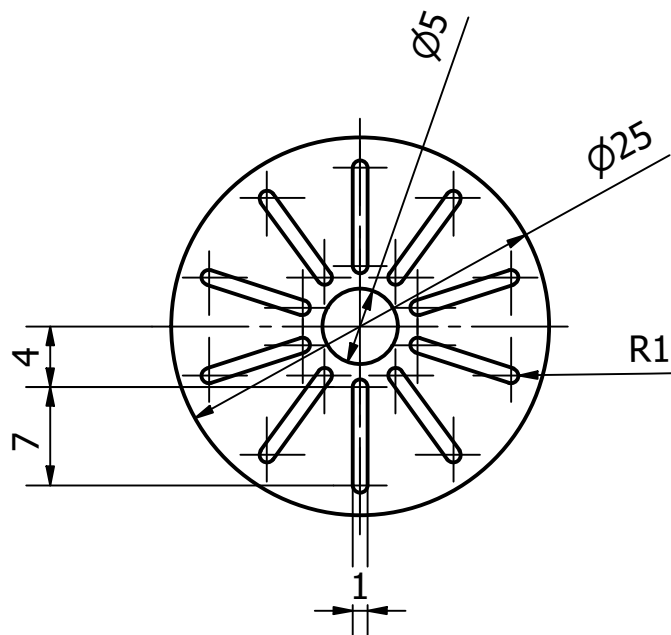


件號	名稱	件數	材料
10	connector	2	密集板
設計者	學號	國立台灣大學 機械工程學系	
Peng	B07502054		
審核者	◎ ◻	比例	日期
余彥亨		2:1	2021/5/12



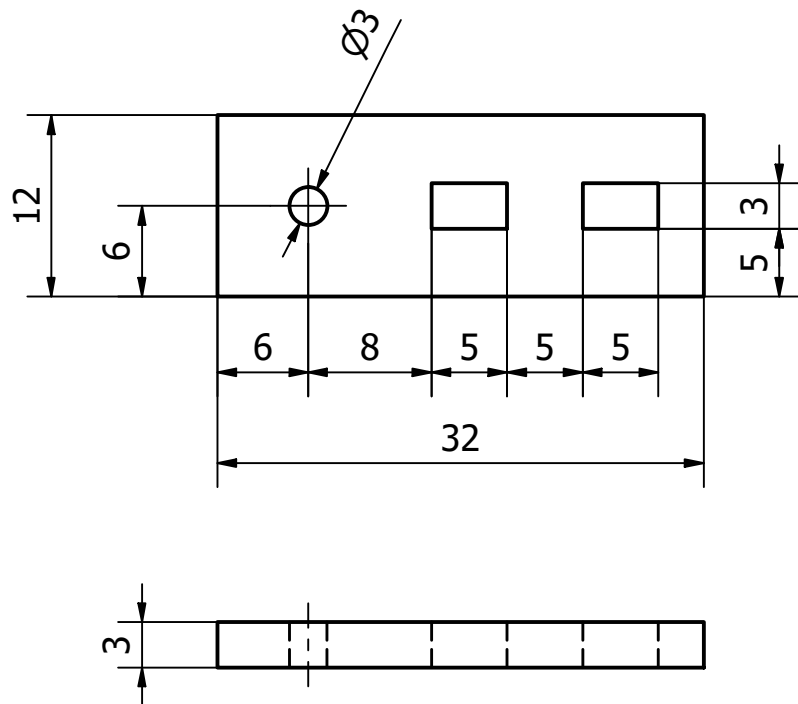
(密集板厚度為3mm)

件號	名稱	件數	材料
11	Driver Frame	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502065		
審核者	◎ ◻	比例	日期
丁映文		1:1	5/11/2021



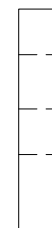
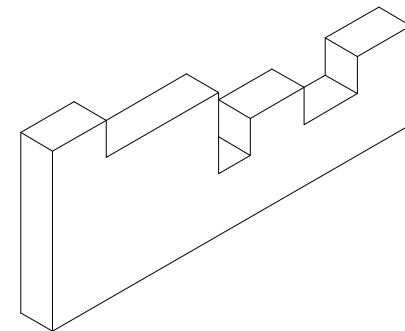
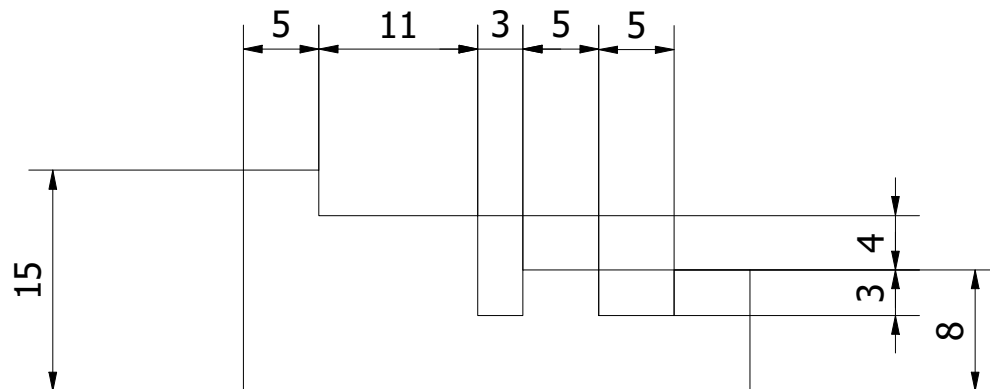
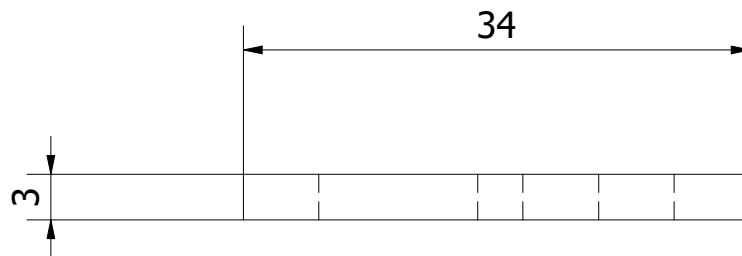
(密集板厚度為3mm，
所有溝槽皆完全切穿)

件號	名稱	件數	材料
12	Encoder Disk	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
丁映文	B07502065		
審核者	◎ ◻	比例	日期
丁映文		2:1	5/12/2021

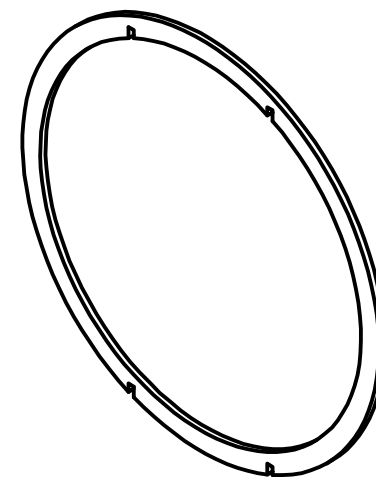
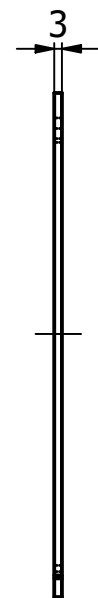
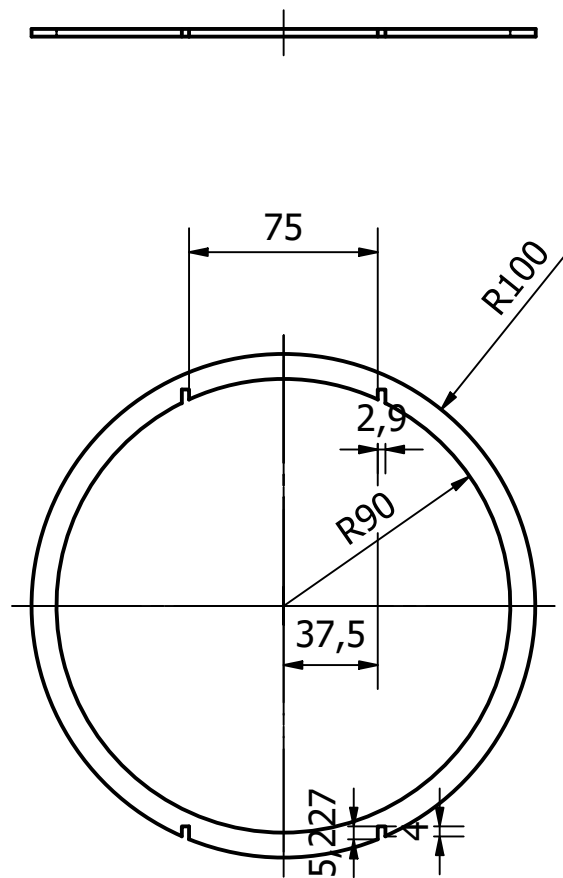


(密集板厚度為3mm)

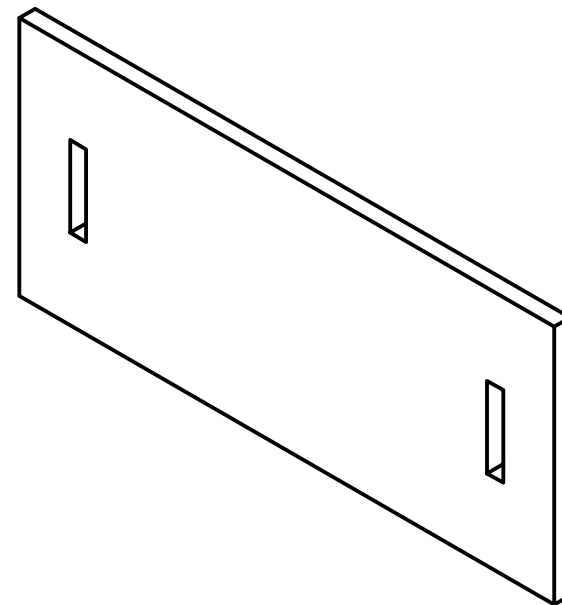
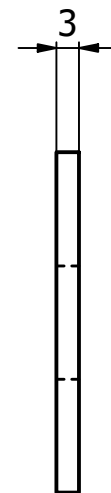
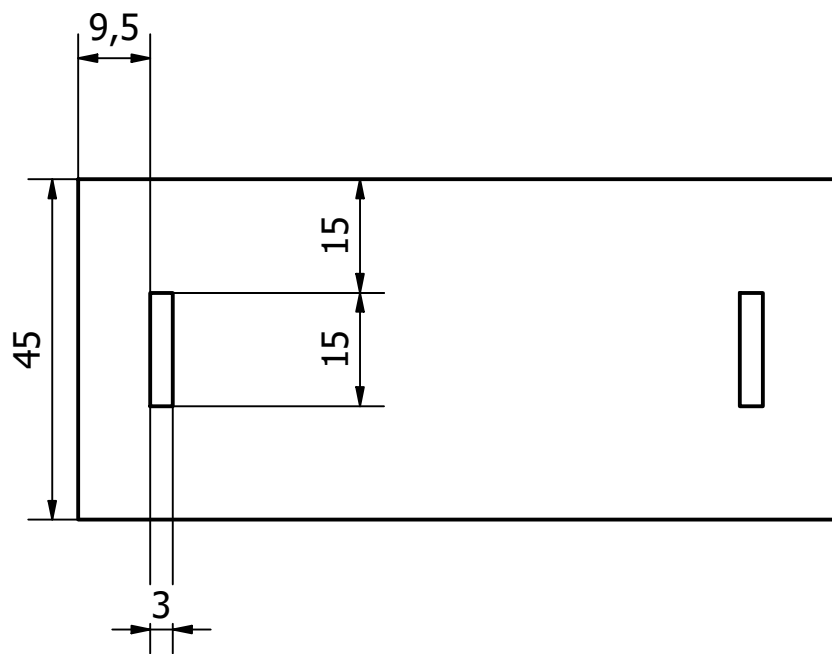
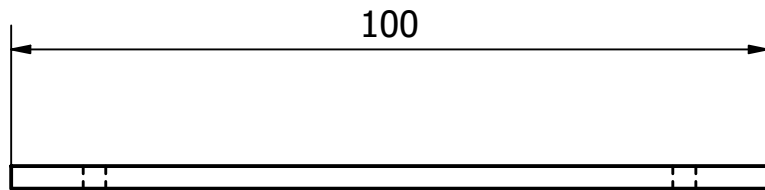
件號	名稱	件數	材料
13	Encoder holder	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502065		
審核者	◎ ◻	比例	日期
丁映文		2:1	5/12/2021



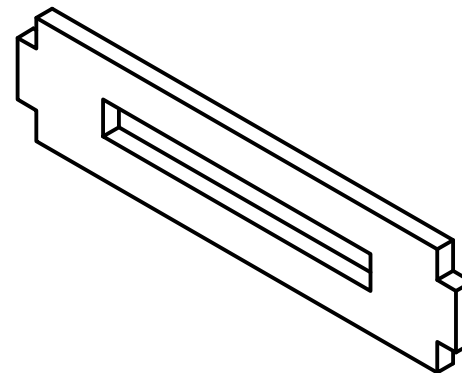
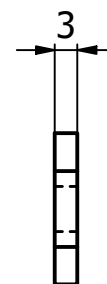
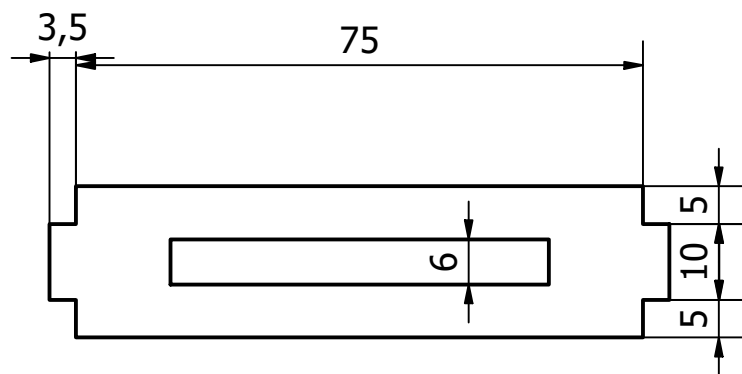
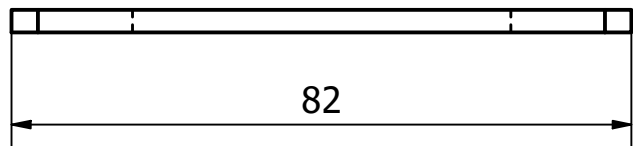
件號	名稱	件數	材料
14	encoder supporter	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502068		
審核者		比例	日期
曾兆廷		1:1	2021/5/12



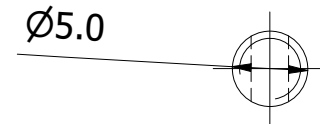
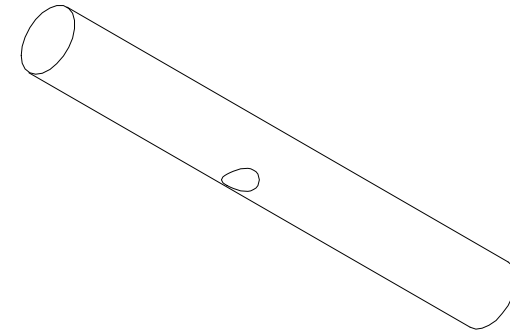
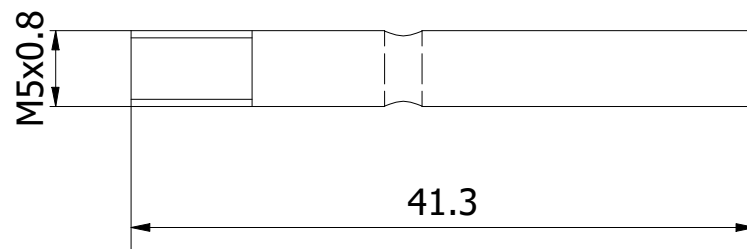
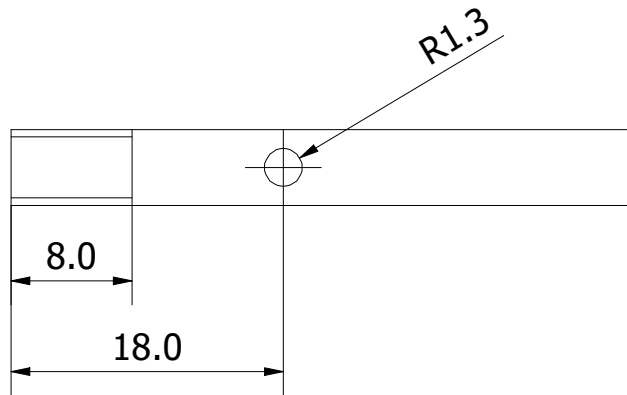
件號	名稱	件數	材料
15	fan cover	2	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	b07502077		
審核者	◎ ◻	比例	日期
魏振宇		1:3	2021/5/15



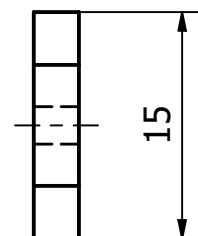
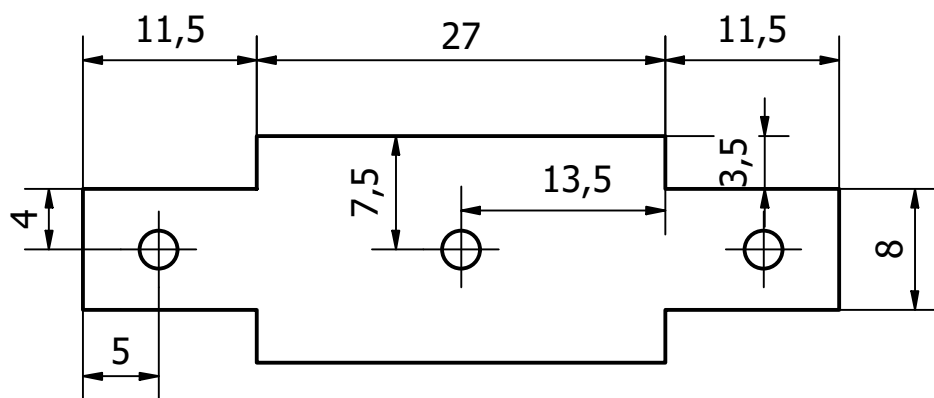
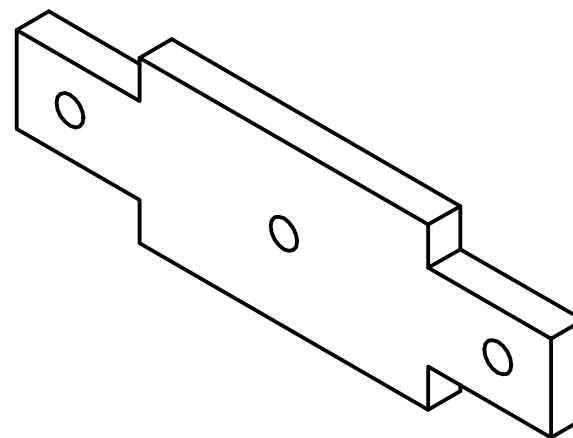
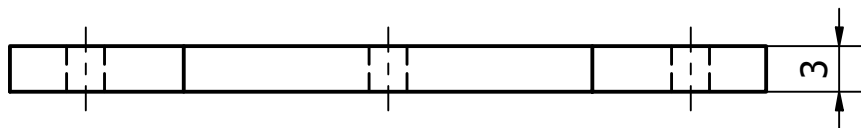
件號	名稱	件數	材料
16	fan protecting board	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502077		
審核者	◎ ◻	比例	日期
魏振宇		1:1	2021/5/15



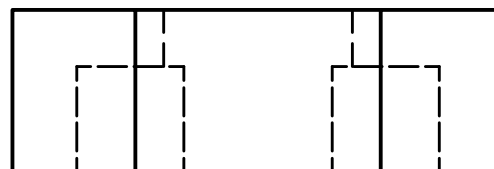
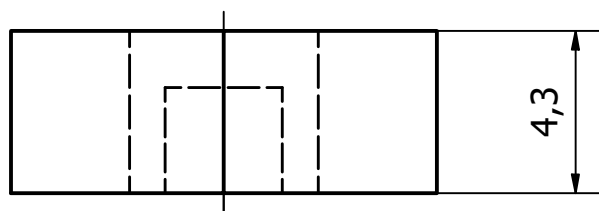
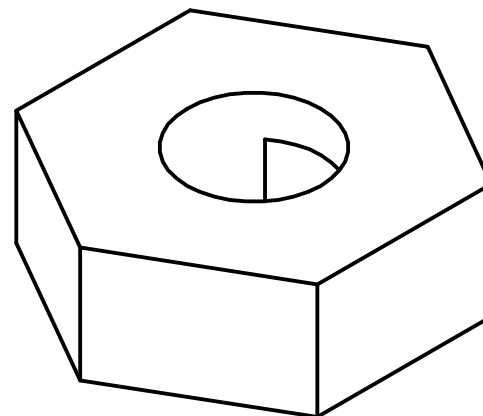
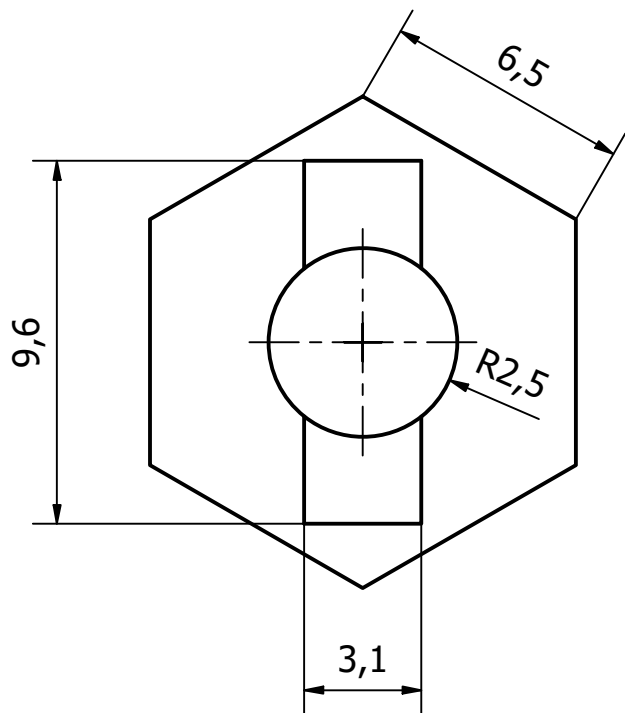
件號	名稱	件數	材料
17	fan support rib	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	b07502077		
審核者	◎ ◻	比例	日期
魏振宇		1:1	2021/5/15



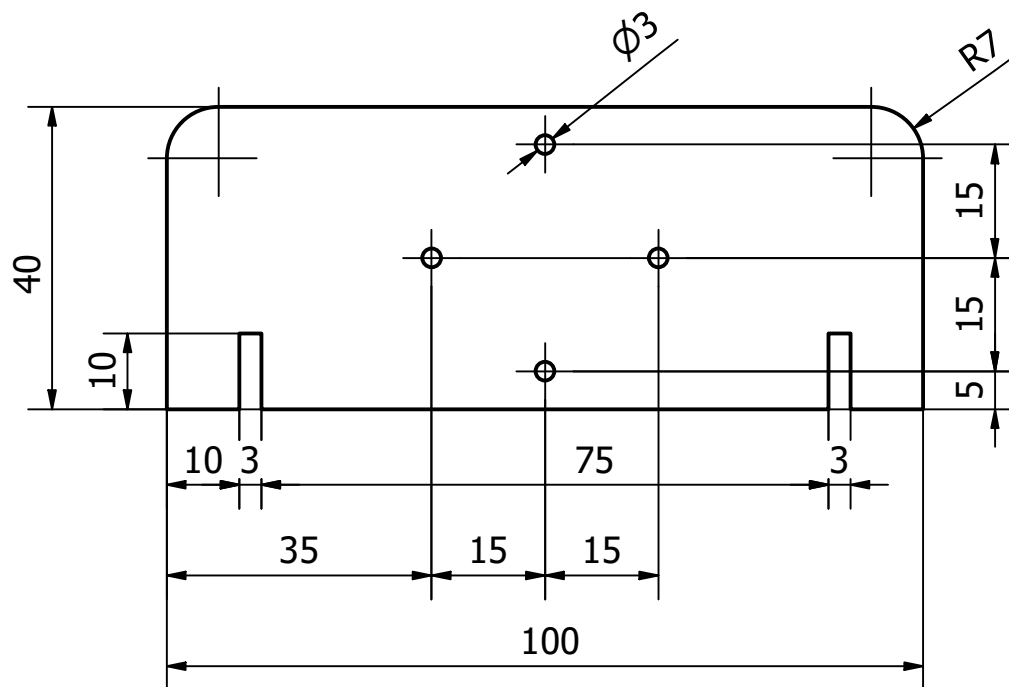
件號	名稱	件數	材料
18	front axle	2	6061鋁合金
設計者	學號	國立台灣大學 機械工程學系	
丁映文	B07502144		
審核者	 	比例	日期
石翊鵬		2:1	2021/3/27



件號	名稱	件數	材料
19	front wheel locking plate	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	b07502077		
審核者	◎ ◻	比例	日期
魏振宇		2:1	2021/5/12

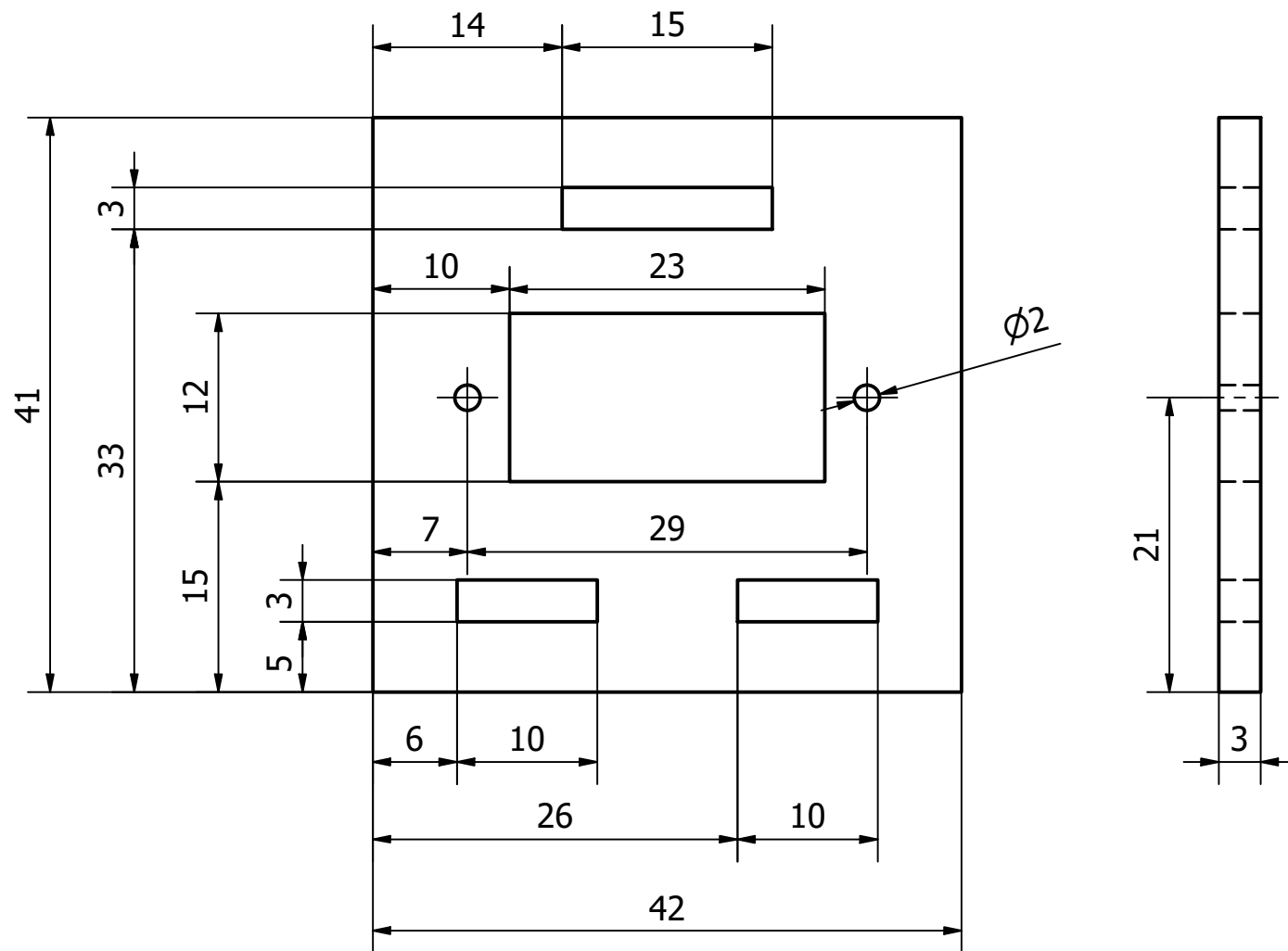


件號	名稱	件數	材料
20	hex	4	PLA
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	b07502077		
審核者	◎ ◻	比例	日期
魏振宇		5:1	2021/6/18



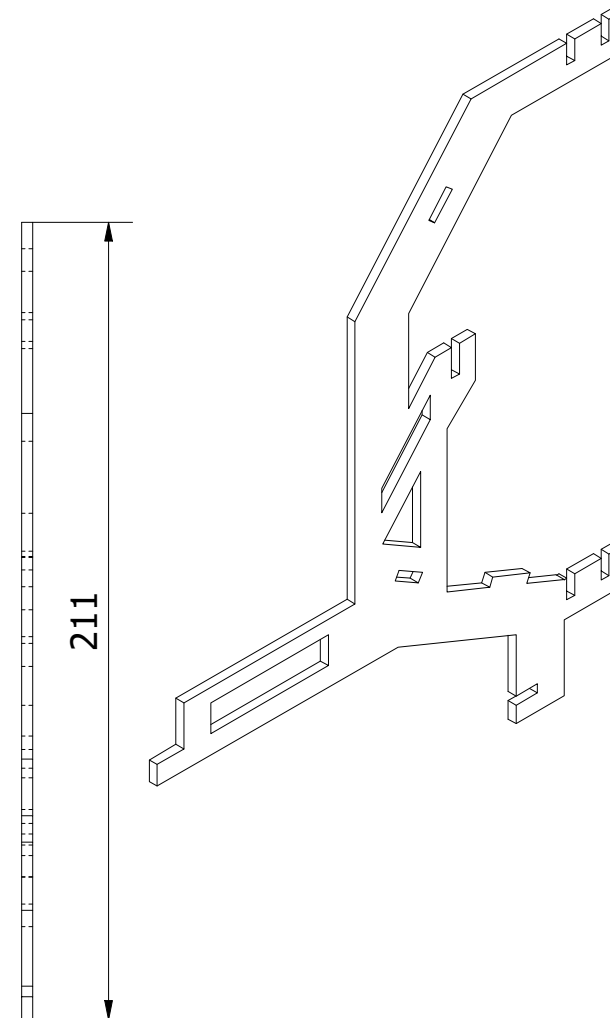
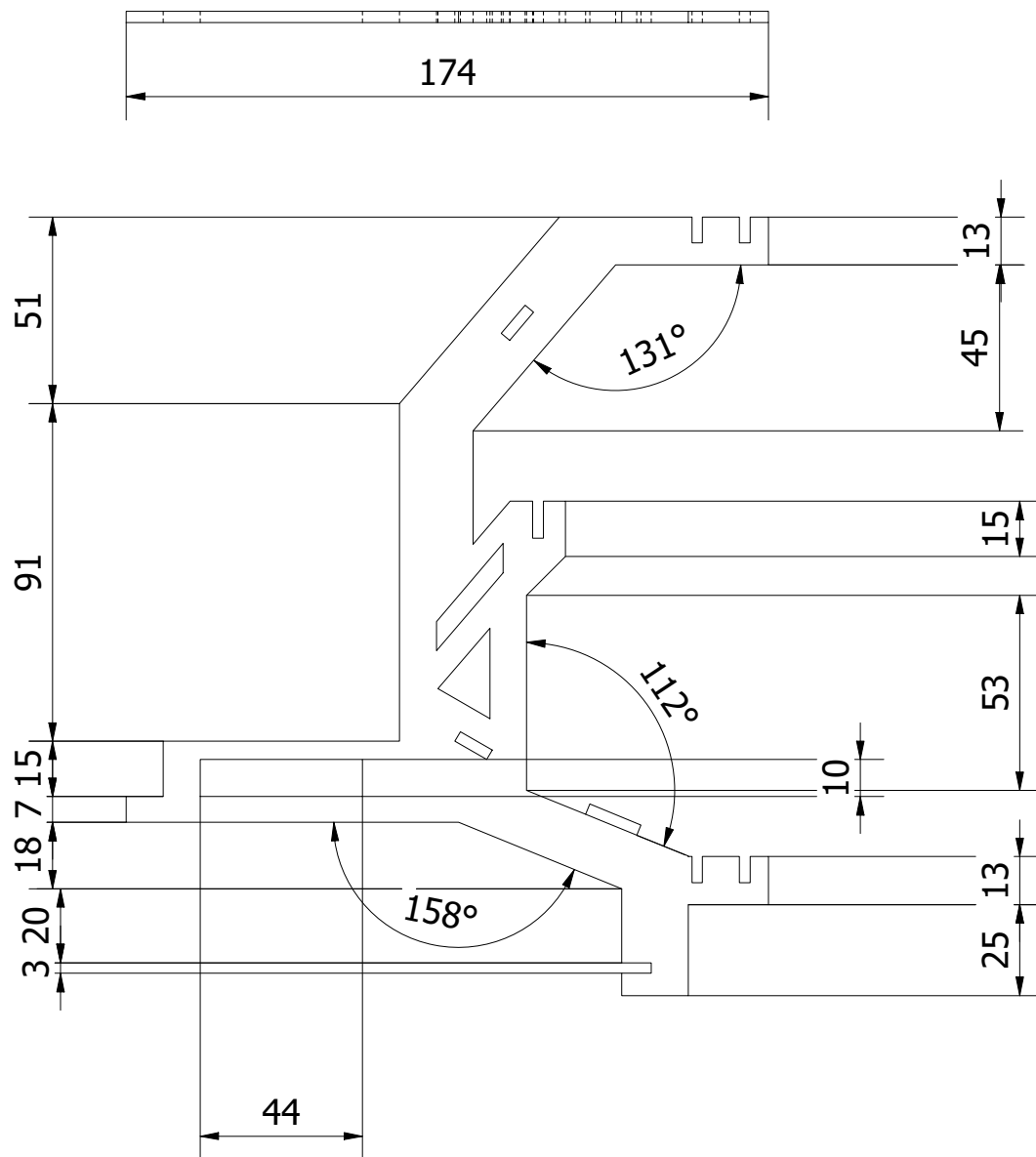
(密集板厚度為3mm)

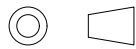
件號	名稱	件數	材料
23	Motor Fixing Plate	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
余彥亨	B07502065		
審核者	◎ ◻	比例	日期
丁映文		1:1	3/15/2021

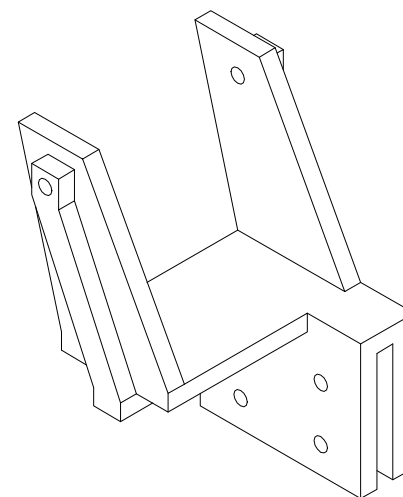
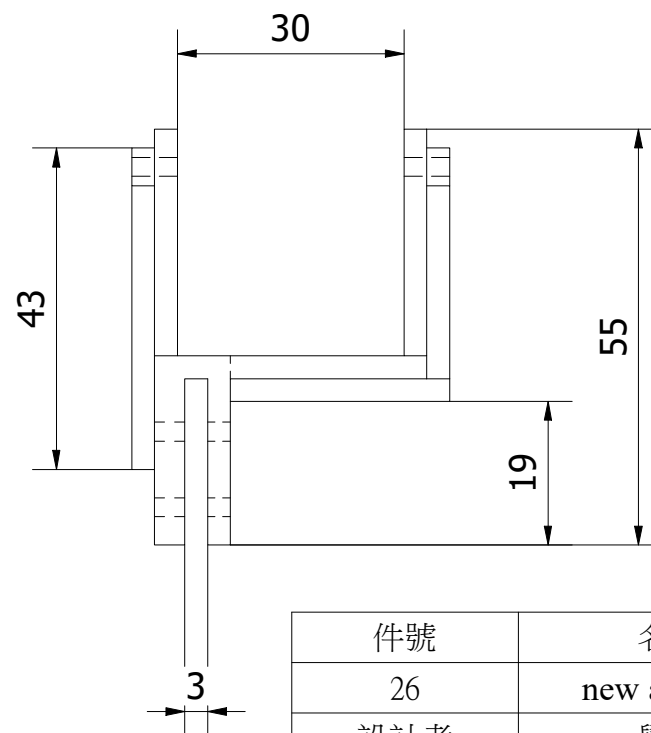
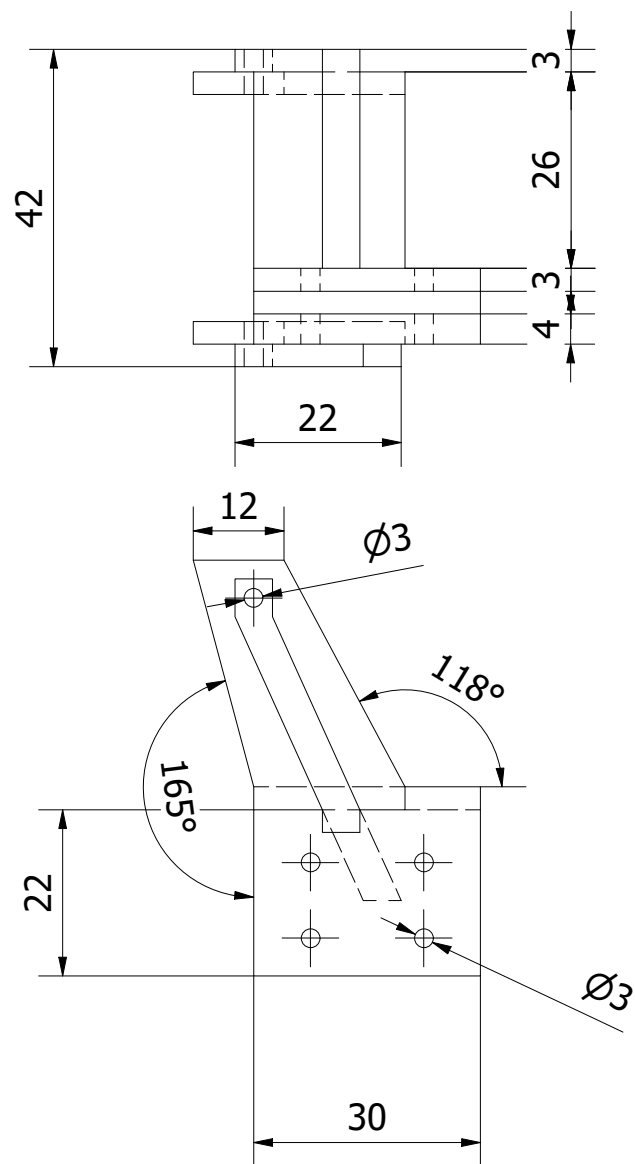


(密集板厚度皆為3mm)

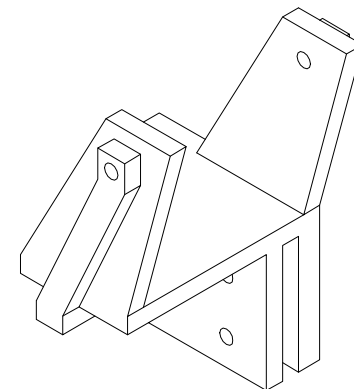
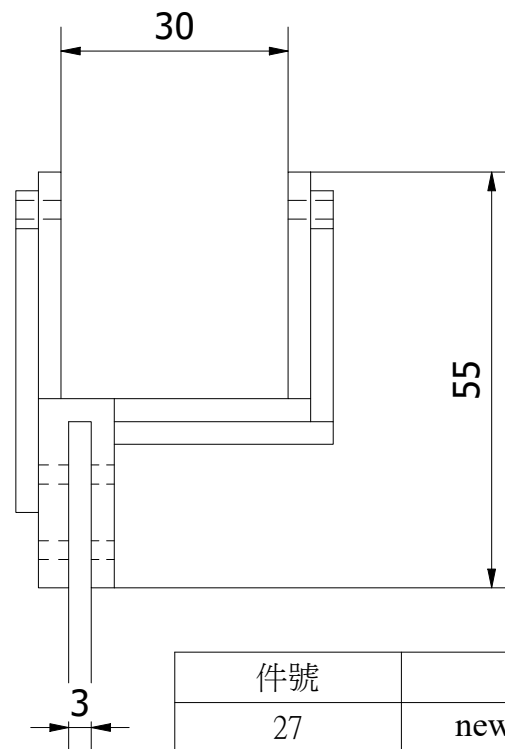
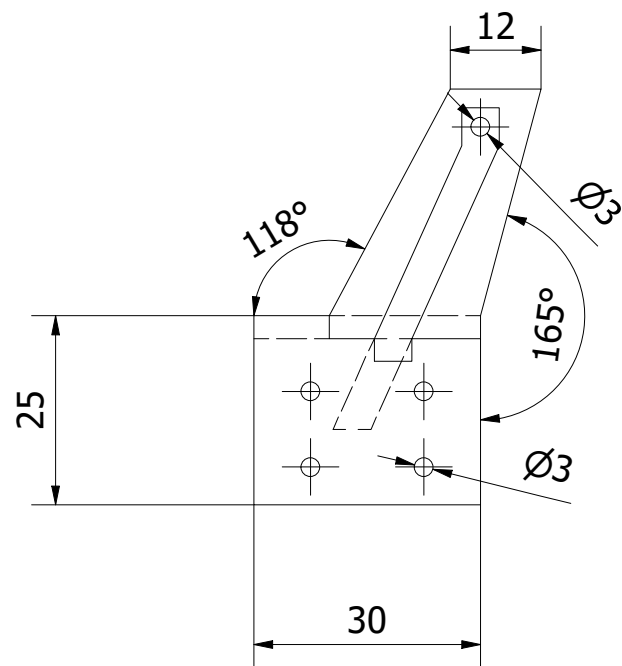
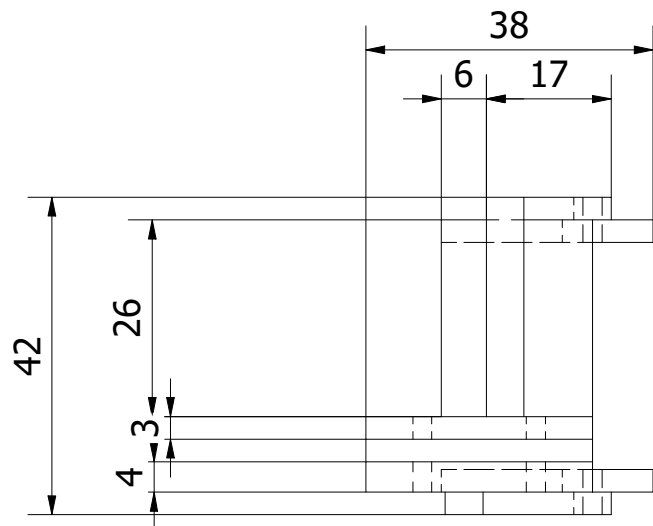
件號	名稱	件數	材料
24	Motor Plate	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502065		
審核者	◎ ◻	比例	日期
丁映文		2:1	5/14/2021

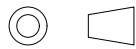


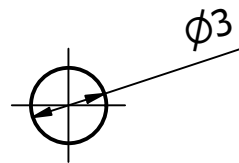
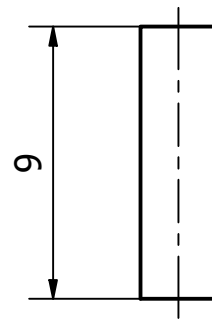
件號	名稱	件數	材料
25	motor supporter	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502068		
審核者		比例	日期
曾兆廷		1:2	2021/5/14



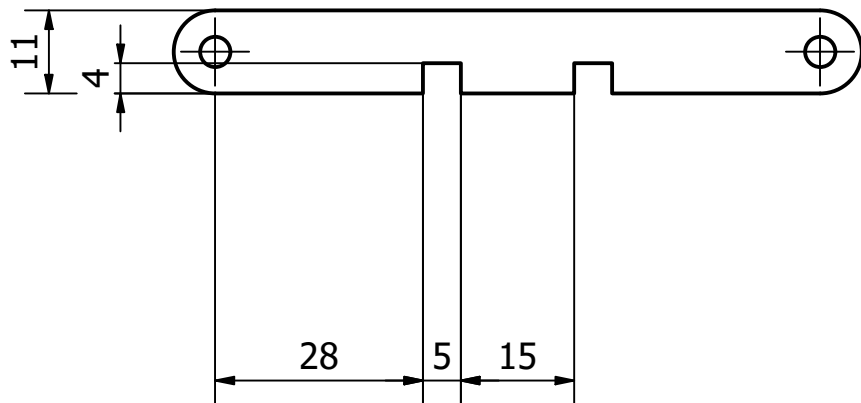
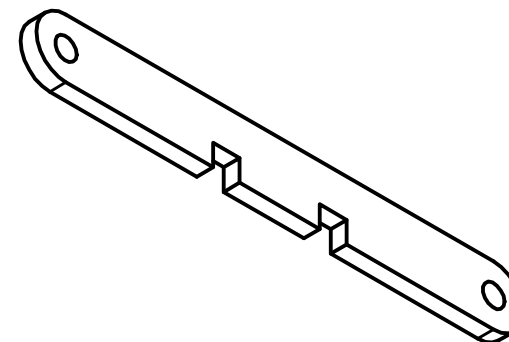
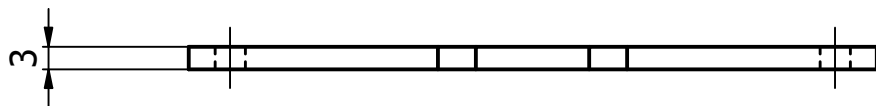
件號	名稱	件數	材料
26	new arm left	1	PLA
設計者	學號	國立台灣大學 機械工程學系	
丁映文	B07502068		
審核者	 	比例	日期
曾兆廷		1:1	2021/4/19



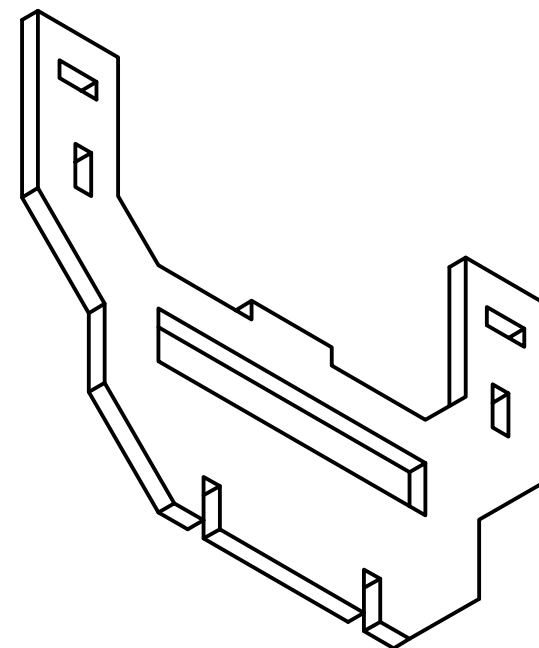
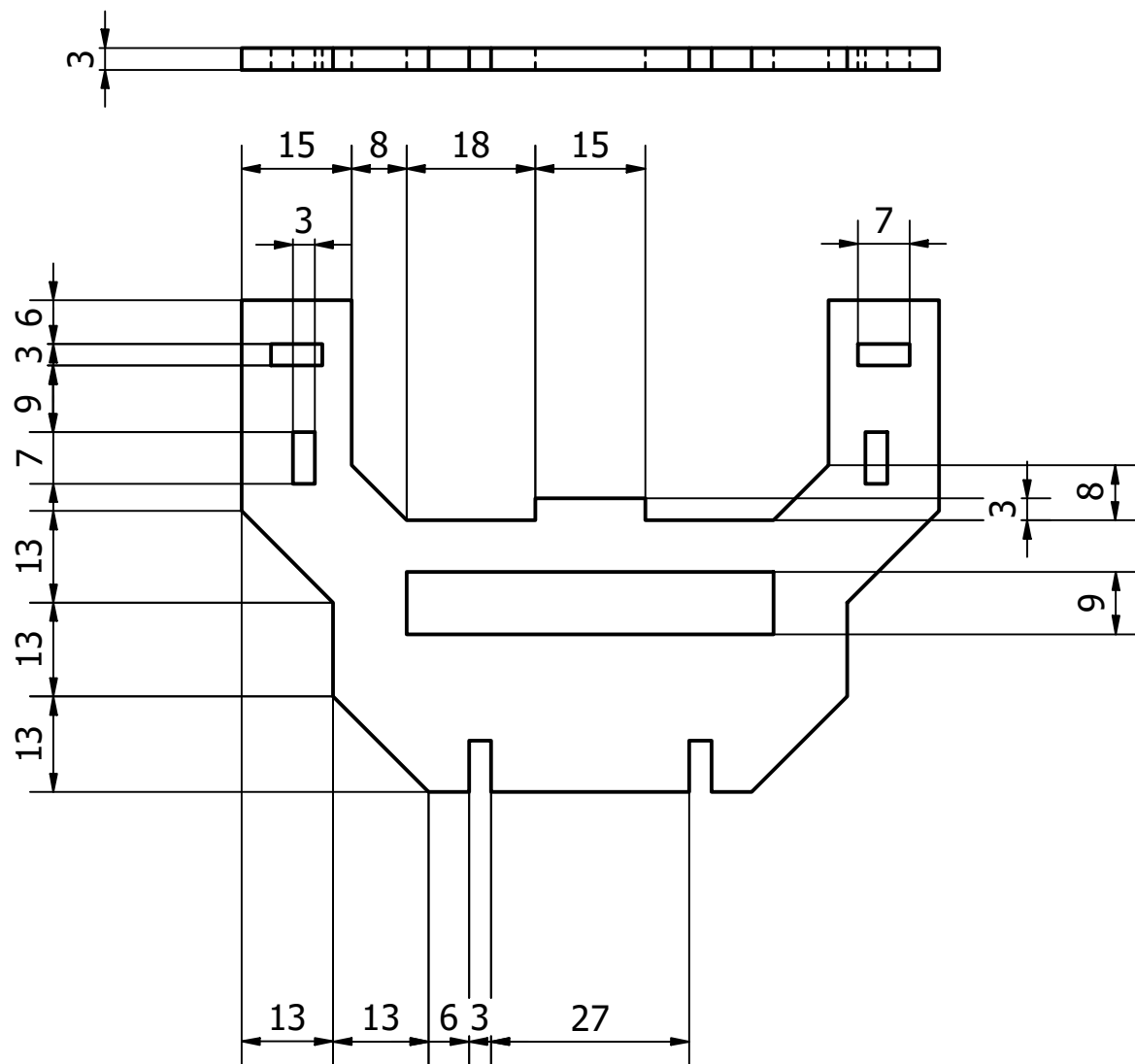
件號	名稱	件數	材料
27	new arm right	1	PLA
設計者	學號	國立台灣大學 機械工程學系	
丁映文	B07502068		
審核者		比例	日期
曾兆廷		1:1	2021/4/19



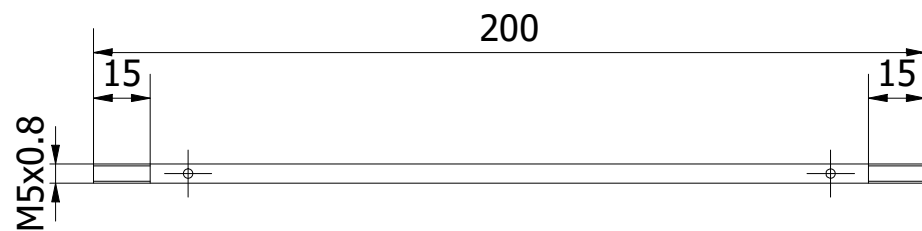
件號	名稱	件數	材料
28	Pin	4	不銹鋼
設計者	學號	國立台灣大學 機械工程學系	
丁映文	B07502065		
審核者	◎ ◻	比例	日期
丁映文		4:1	3/27/2021



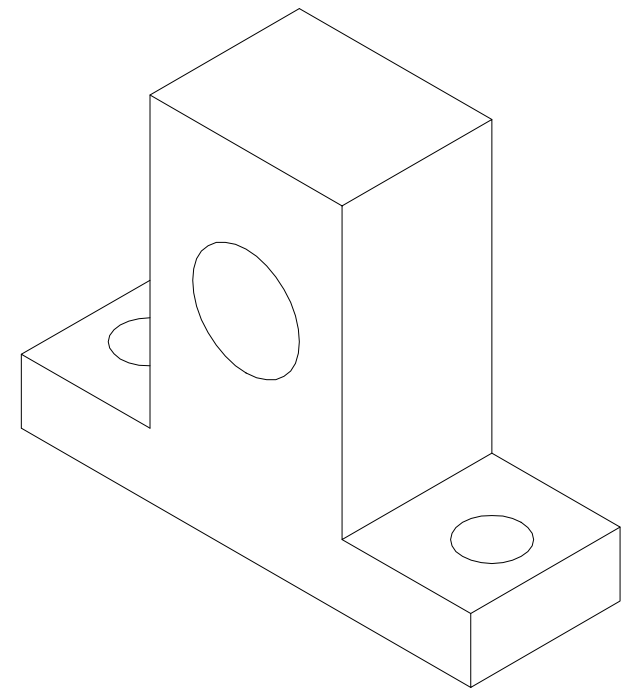
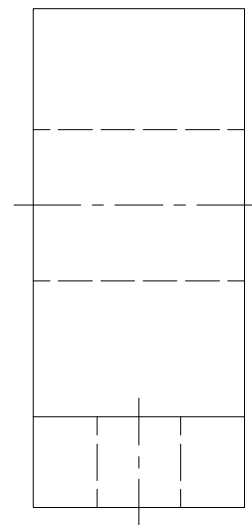
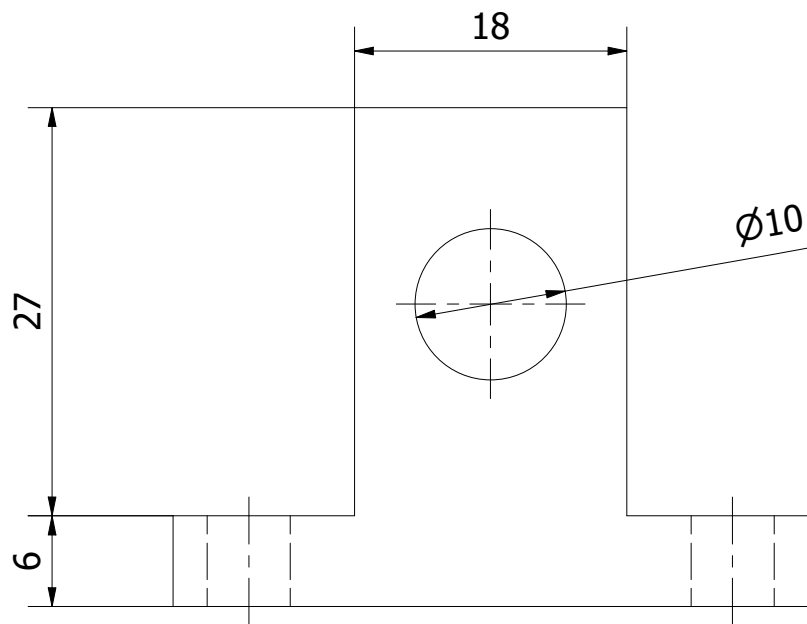
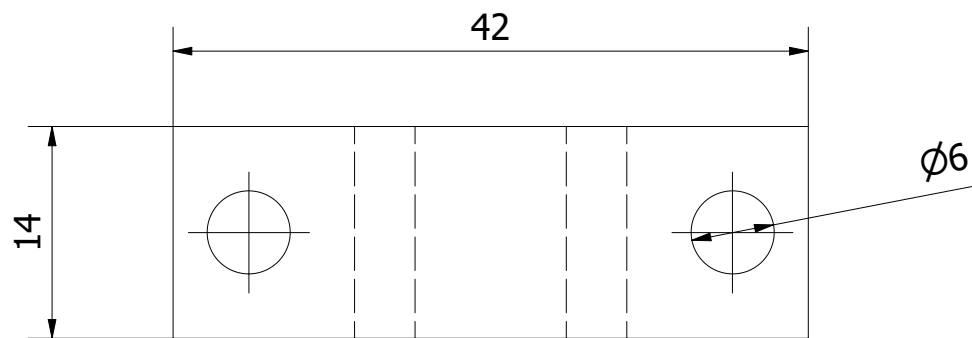
件號	名稱	件數	材料
29	rack connector	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
Peng	B07502054		
審核者	◎ ◻	比例	日期
余彥亨		1:1	2021/5/12

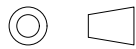


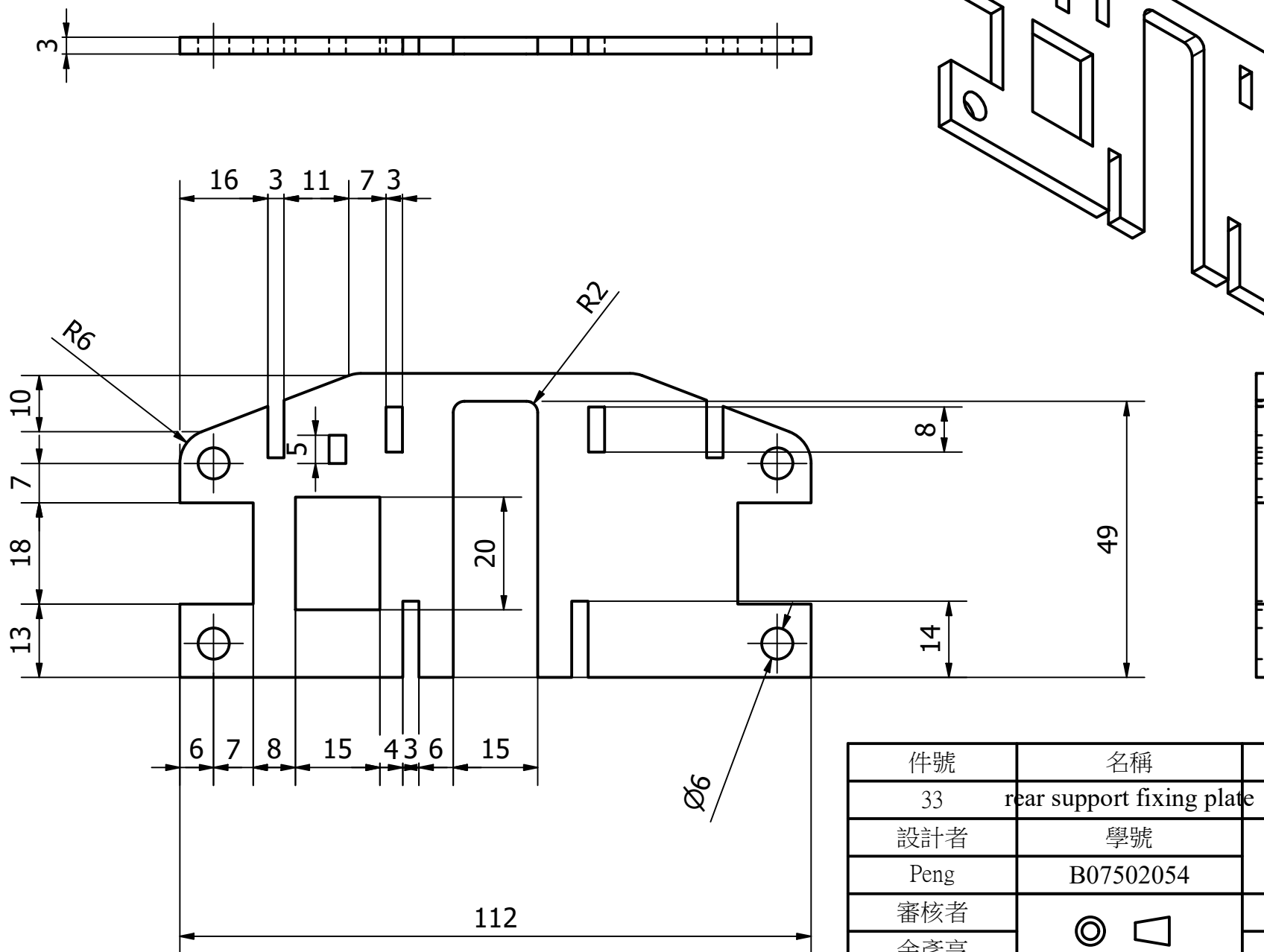
件號	名稱	件數	材料
30	rack holding plate	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
Peng	B07502054		
審核者	◎ ◻	比例	日期
余彥亨		1:1	2021/5/13



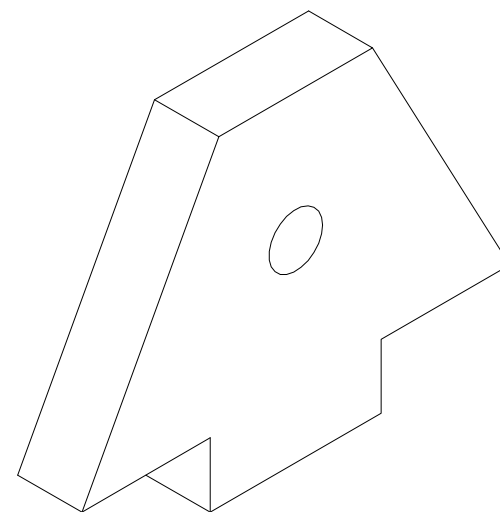
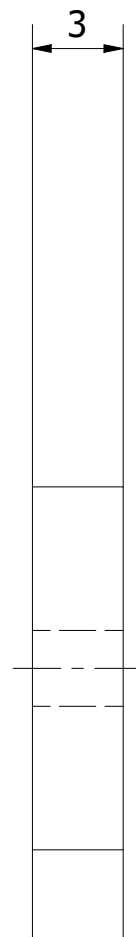
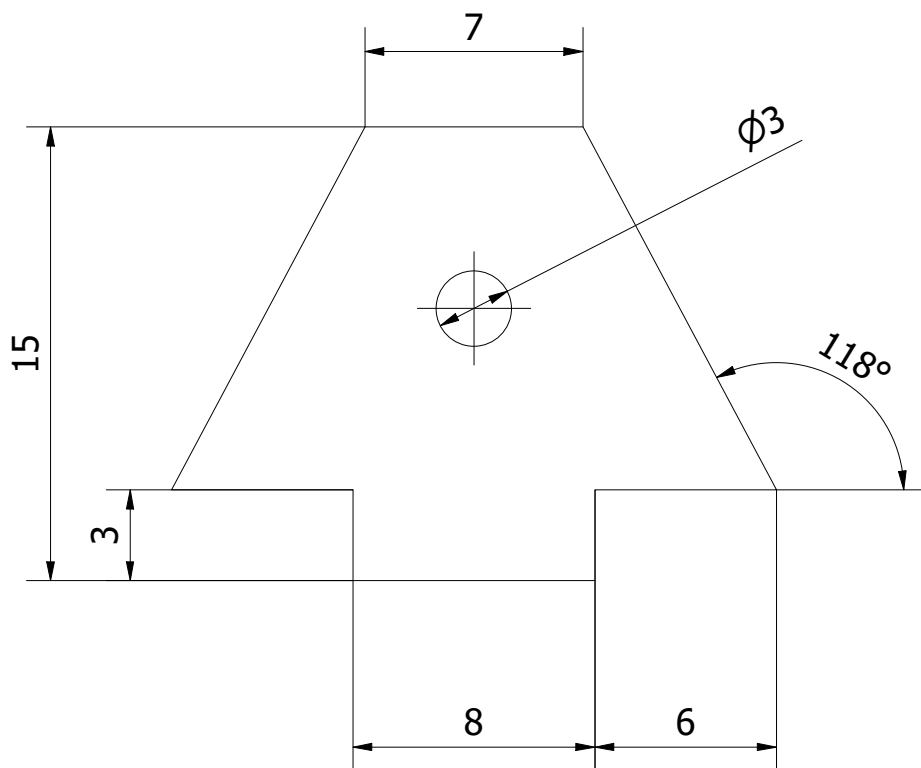
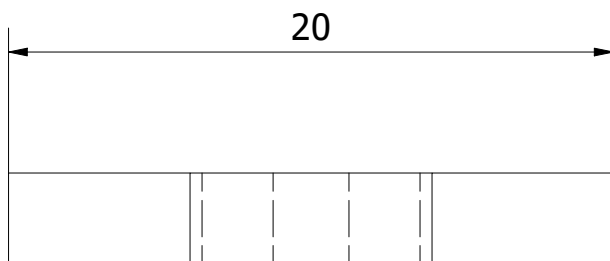
件號	名稱	件數	材料
31	rear axle	1	6061鋁合金
設計者	學號	國立台灣大學 機械工程學系	
丁映文	B07502144		
審核者		比例	日期
石翊鵬		1:2	2021/3/31



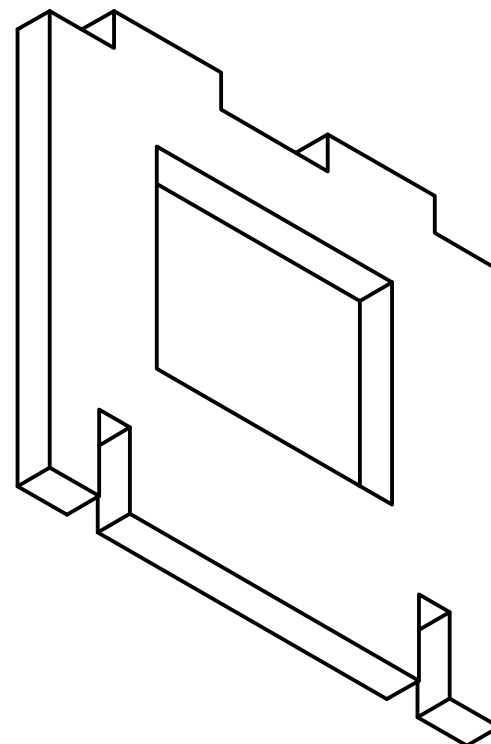
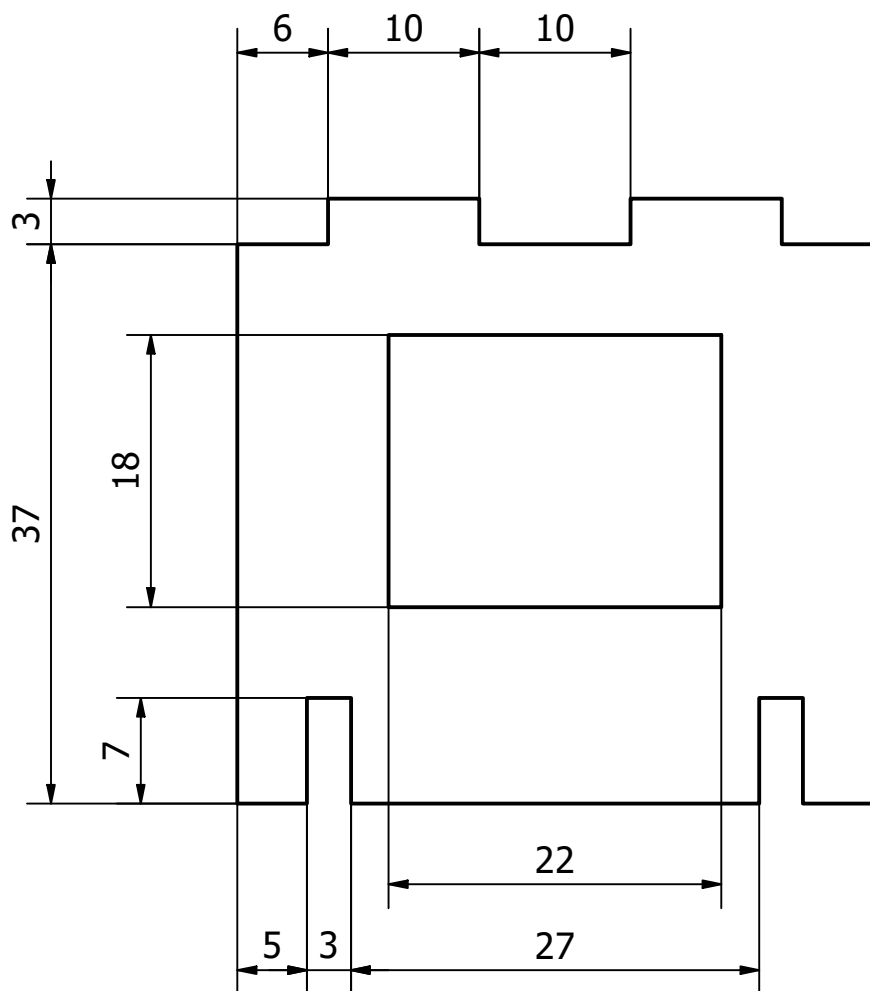
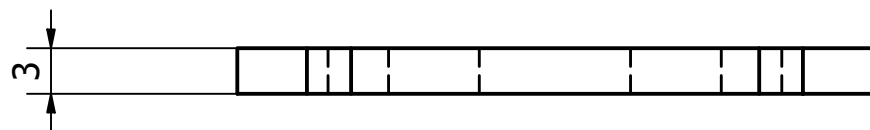
件號	名稱	件數	材料
32	rear axle support	3	PLA
設計者	學號	國立台灣大學 機械工程學系	
丁映文	B07502068		
審核者		比例	日期
曾兆廷		2:1	2021/3/28



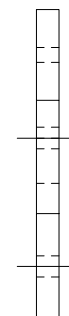
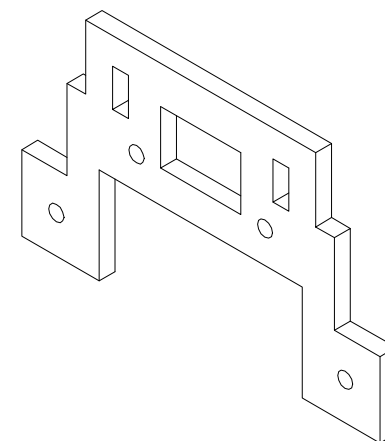
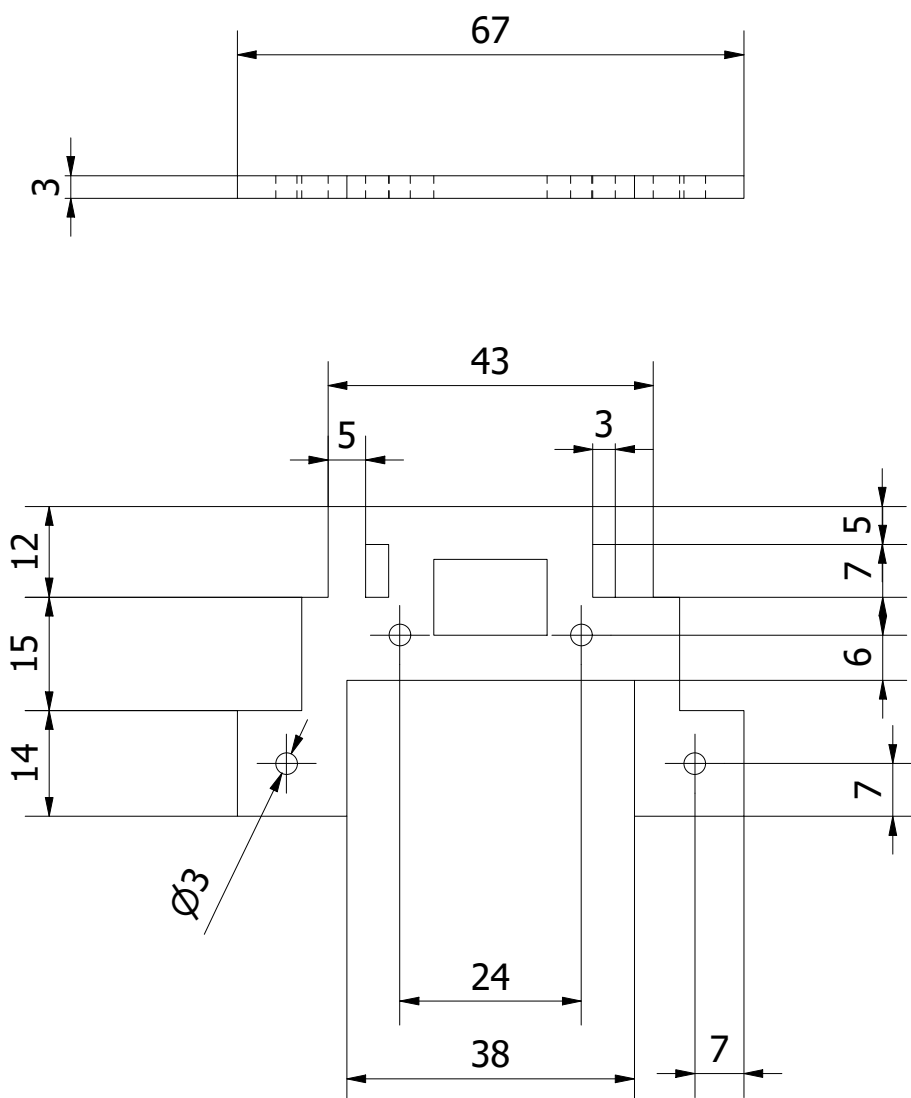
件號	名稱	件數	材料
33	rear support fixing plate	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
Peng	B07502054		
審核者	◎ ▢	比例	日期
余彥亨		1:1	2021/5/11



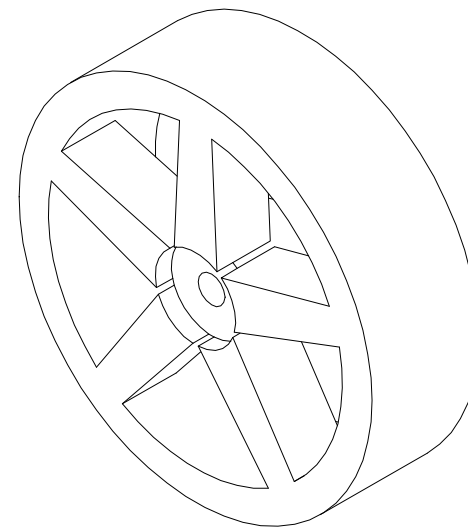
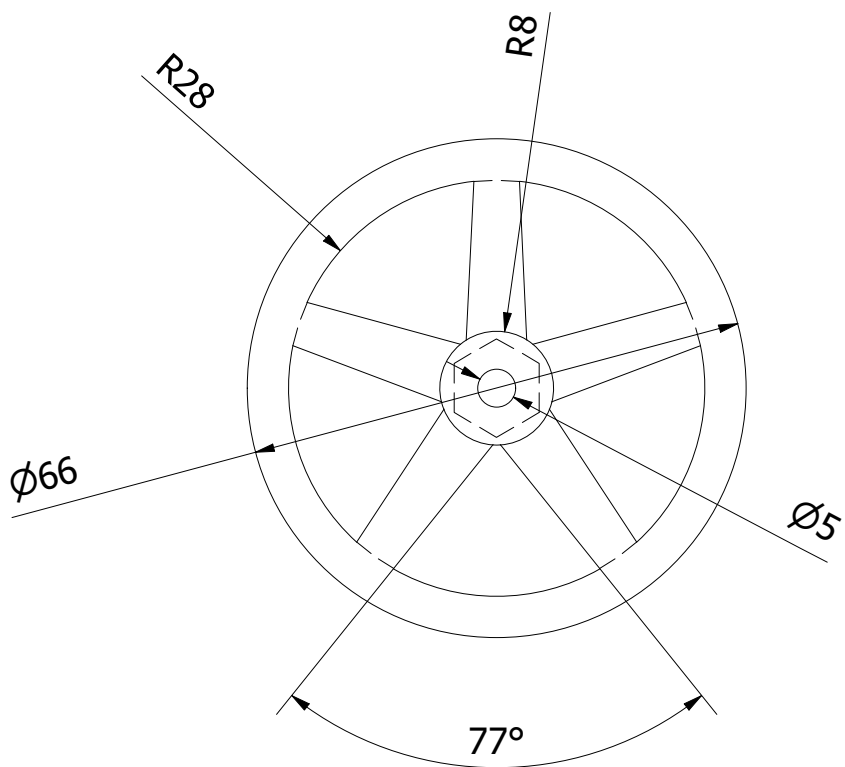
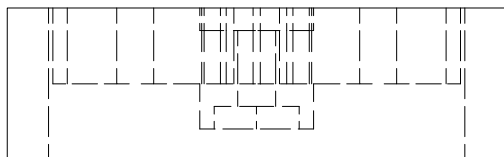
件號	名稱	件數	材料
34	rear wheel qr	2	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502068		
審核者	◎ ◁	比例	日期
曾兆廷		1:4	2021/5/14



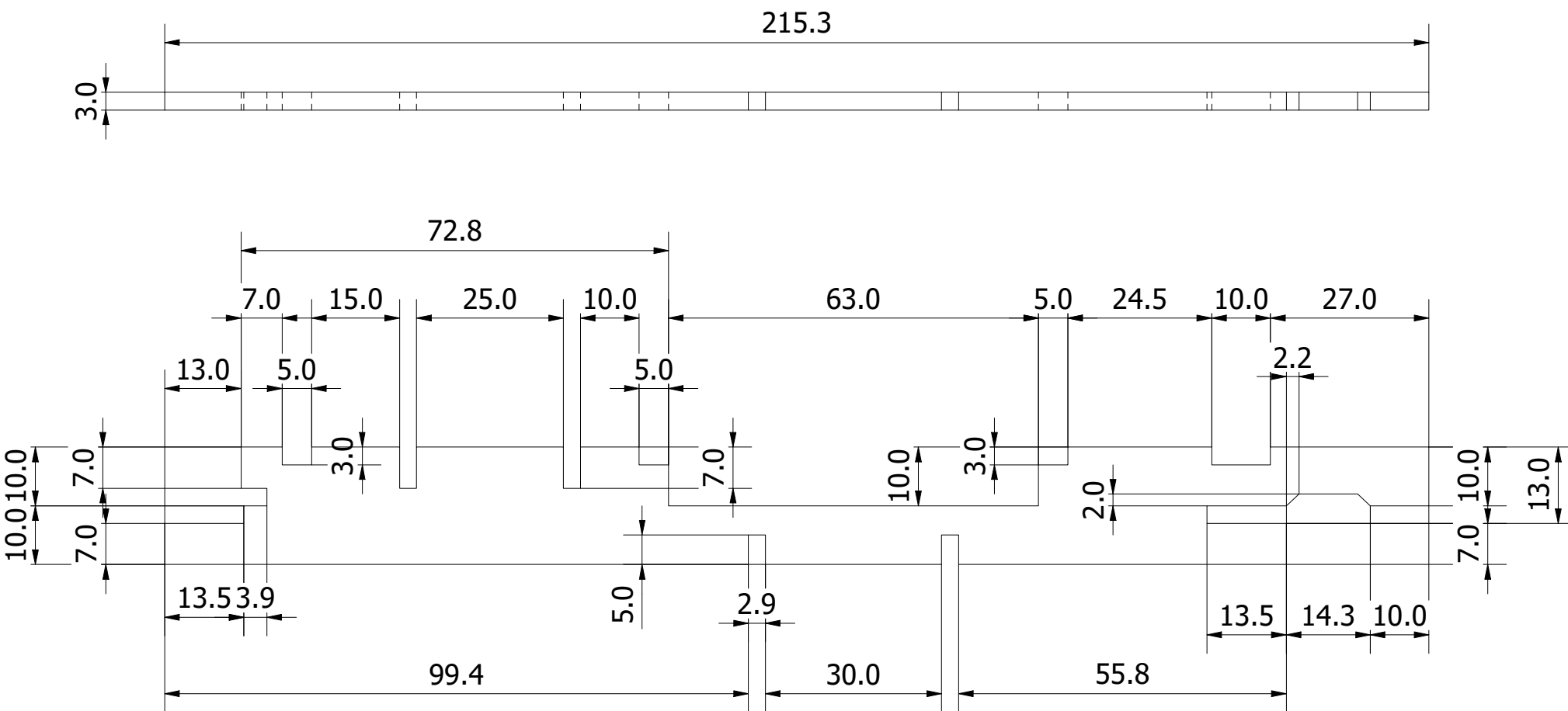
件號	名稱	件數	材料
35	servo support	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
Peng	B07502054		
審核者	◎ ◻	比例	日期
余彥亨		2:1	2021/5/14



件號	名稱	件數	材料
36	tcr5000 support	1	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	b07502068		
審核者	◎ ◁	比例	日期
曾兆廷		1:1	5/12/2021



件號	名稱	件數	材料
38	wheel	4	PLA
設計者	學號	國立台灣大學 機械工程學系	
丁映文	B07502068		
審核者	◎ ◻	比例	日期
曾兆廷		1:1	2021/3/27



件號	名稱	件數	材料
39	wheel-support beam	2	密集板
設計者	學號	國立台灣大學 機械工程學系	
石翊鵬	B07502144		
審核者		比例	日期
石翊鵬		1:1	2021/5/11

六、參考資料

- [1]<http://www.liakoptfe.com.tw/ptfe/knowledge>
- [2]<https://www.itsfun.com.tw/%E9%80%9A%E9%A2%A8%E6%A9%9F/wiki-7994495-5522375>
- [3]<https://reurl.cc/R6E9OD>
- [4]<https://reurl.cc/L0LQXX>
- [5]<https://www.azom.com/article.aspx?ArticleID=6636>
- [6] Nelson, W. (1948). Airplane propeller principles. 3rd ed. New York:John Wiley & Sons, Inc., p. 1-17
- [7] Reza N. Jazar (2014). Vehicle Dynamics: Theory and Application, Springer-Verlag New York
- [8] Drag/Fan thrust (2021) Experimental Manual of Measurement and Mechanical Engineering Laboratory. Department of Mechanical Engineering, National Taiwan University.
- [9] Transient pressure(2021) Experimental Manual of Measurement and Mechanical Engineering Laboratory. Department of Mechanical Engineering, National Taiwan University.
- [10]什麼是 塑合板、密集板、甘蔗板？
<https://hydesign.weebly.com/201604063626159-2260921512264951228923494385982649512289299763407126495653118203.html>