**Control System Design Report**

Lab #4、5

組員：0410797陳源德

組員：0410757佘柏陞

**1. 雙輪車動態推導改寫成狀態方程式**

(1) 改寫系統動態方程式成狀態方程式

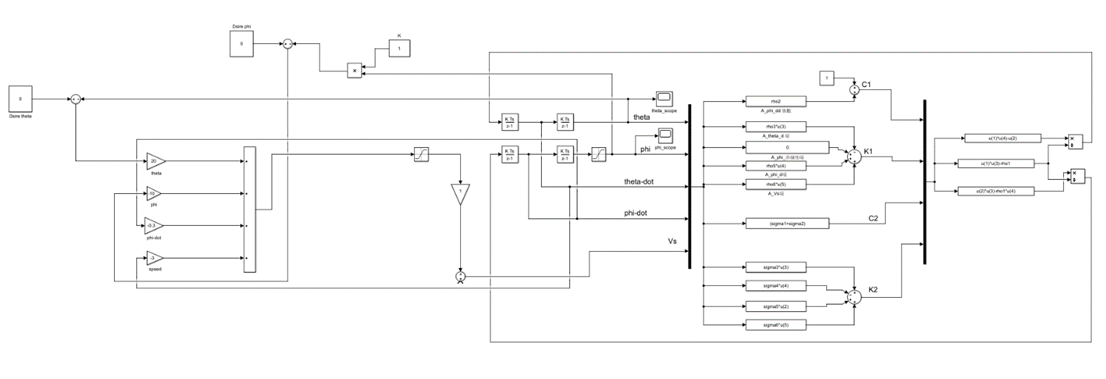
使用feedback來完成控制，首先要確定有哪些系統參數回授使控制器得以控制，其中我們推得有theta (position) , theta-dot (speed) , phi(車傾角) , phi\_dot ,之後我們將其控制器的輸入形式設計以state-feedback的方程式完成控制，而K值我們使用之前推倒的系統動態方程式：

來藉此推算K值大致上為多少。

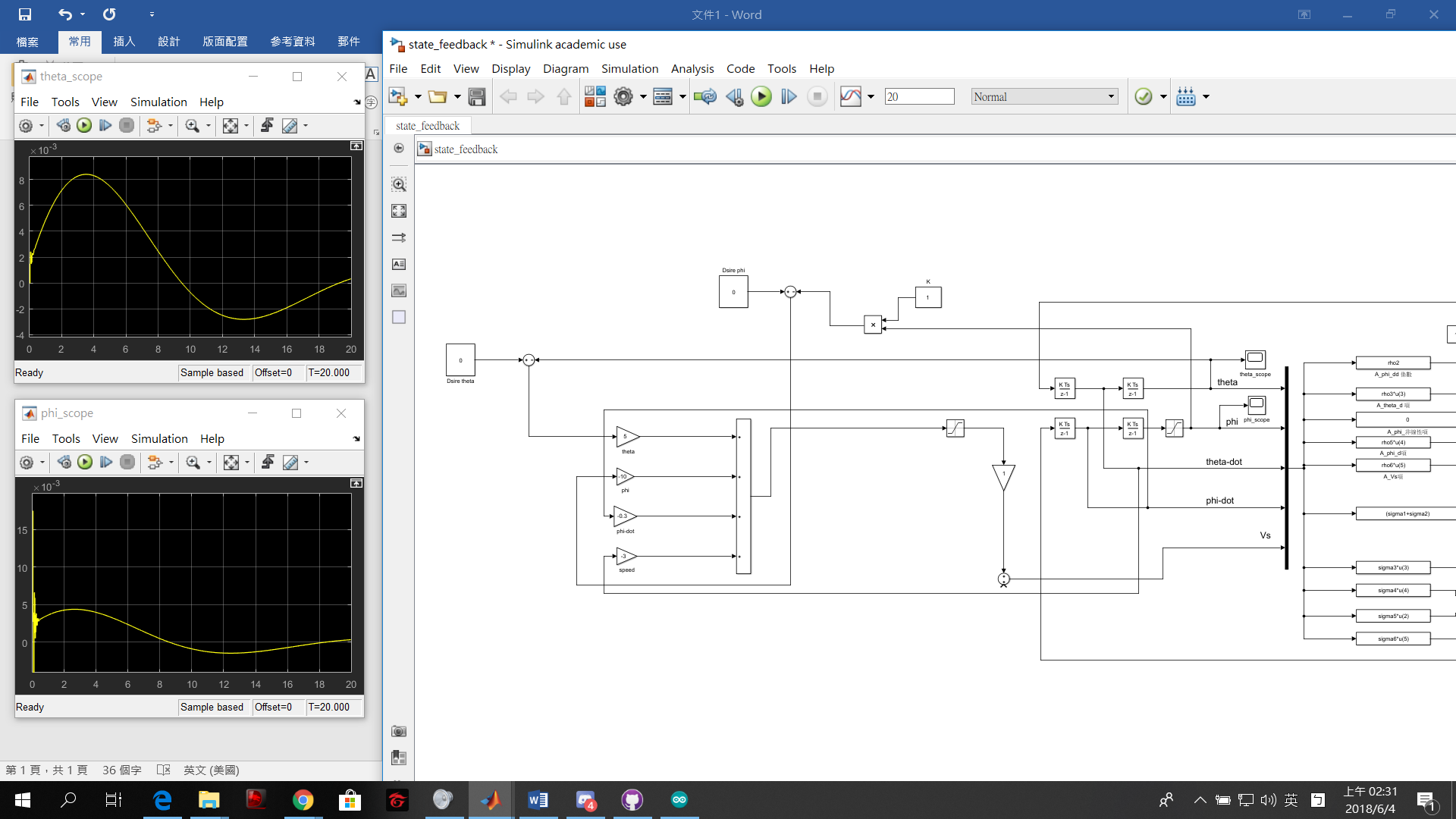
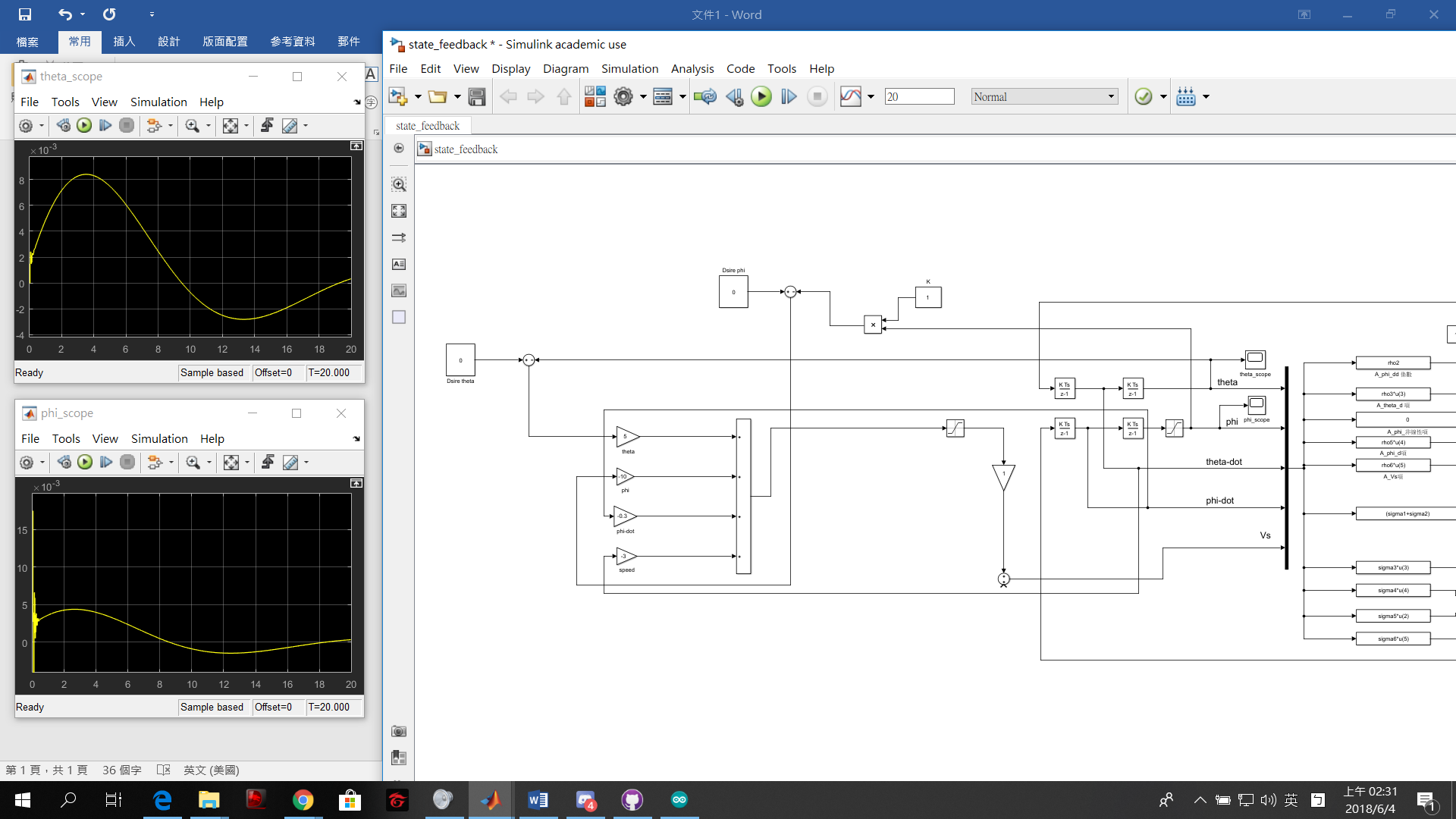
(2) 確認系統之可控性

因為state-feedback以很直觀的方是來控制系統，當車體偏離穩定值多少時，就將值乘上參數-K值，並且回授至系統去調整輸出，使系統穩定，而這也讓控制器很好設計，可控性是可行的。

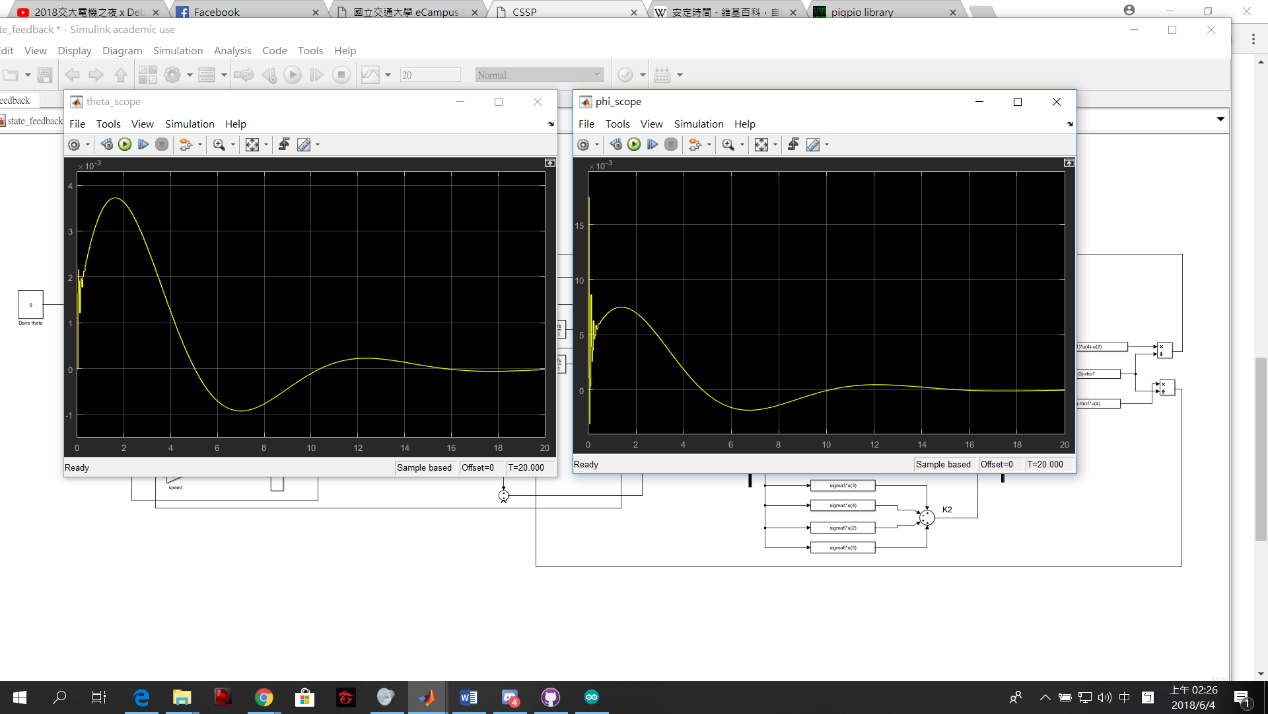
**2. 建立模擬 & 設計 K 值進行實際模擬**

**** 以下是我們用simulink模擬以state-feedback控制位置的系統圖以及結果

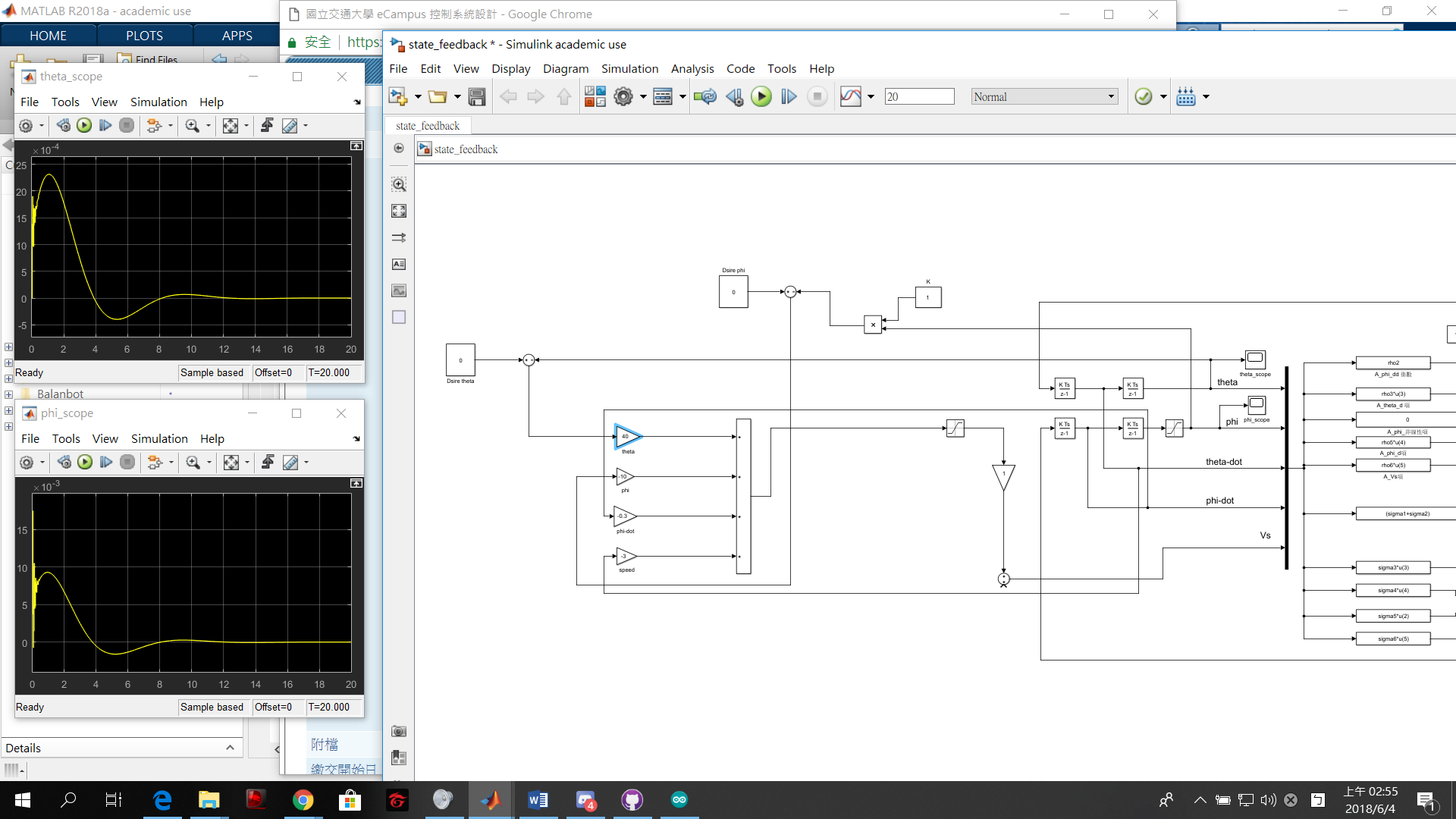
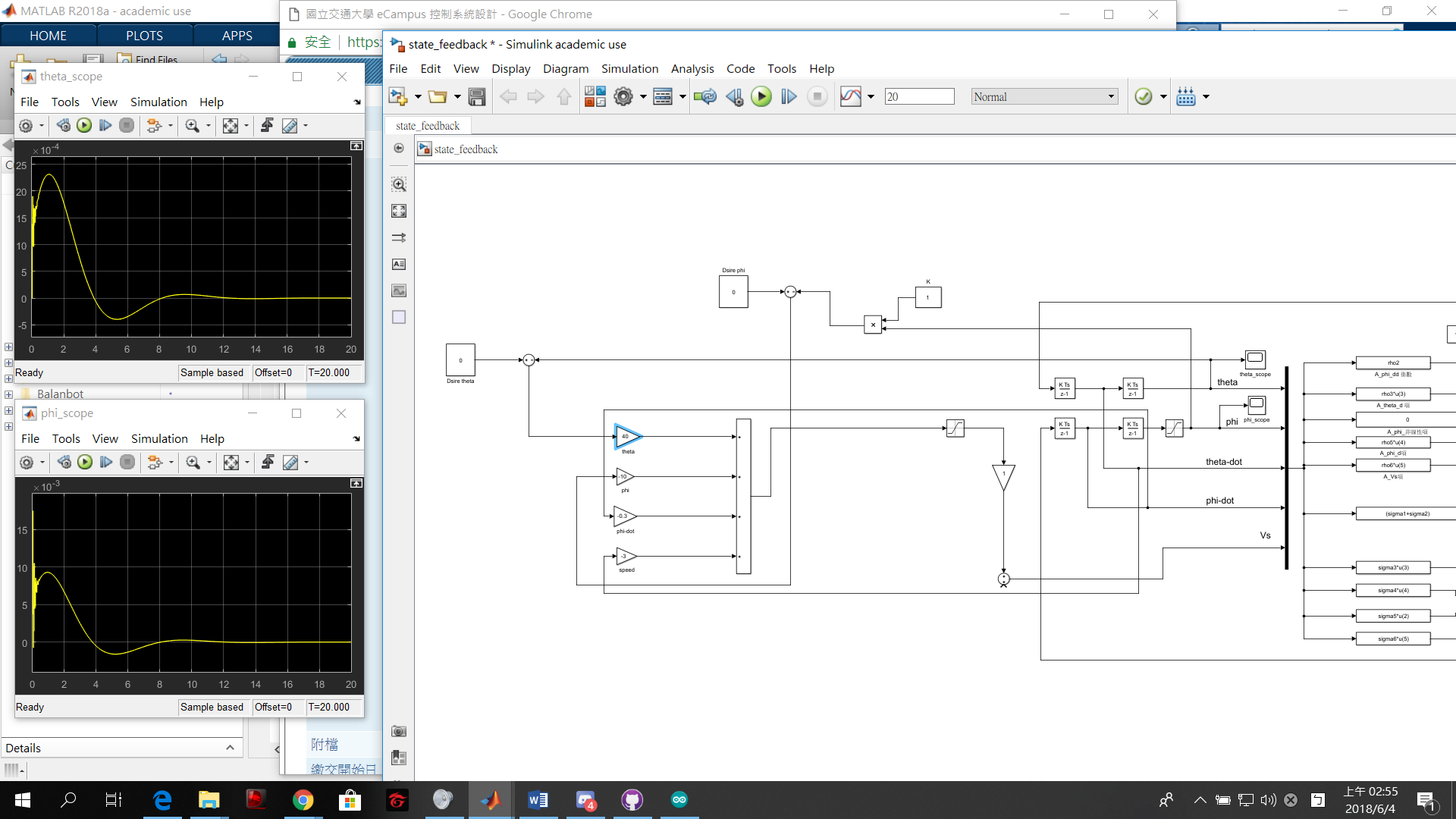
k1(theta) = 5, k2(speed) = 3, k3(phi) = 10, k4(phi\_dot) = 0.3

theta phi

k1 = 20, k2 = 3, k3 = 10, k4 = 0.3

theta phi

k1 = 40, k2 = 3, k3 = 10, k4 = 0.3

theta phi

我們試了幾種不同的數據，發現在k2 = 3, k3 = 10, k4 = 0.3時，系統較易穩定，而k1則容易會影響系統的overshoot及settling time。我們最後選用了k1 = 20, k2 = 3, k3 = 10, k4 = 0.3繼續做位置控制的部分。

以下為使用不同Desire theta時的模擬結果

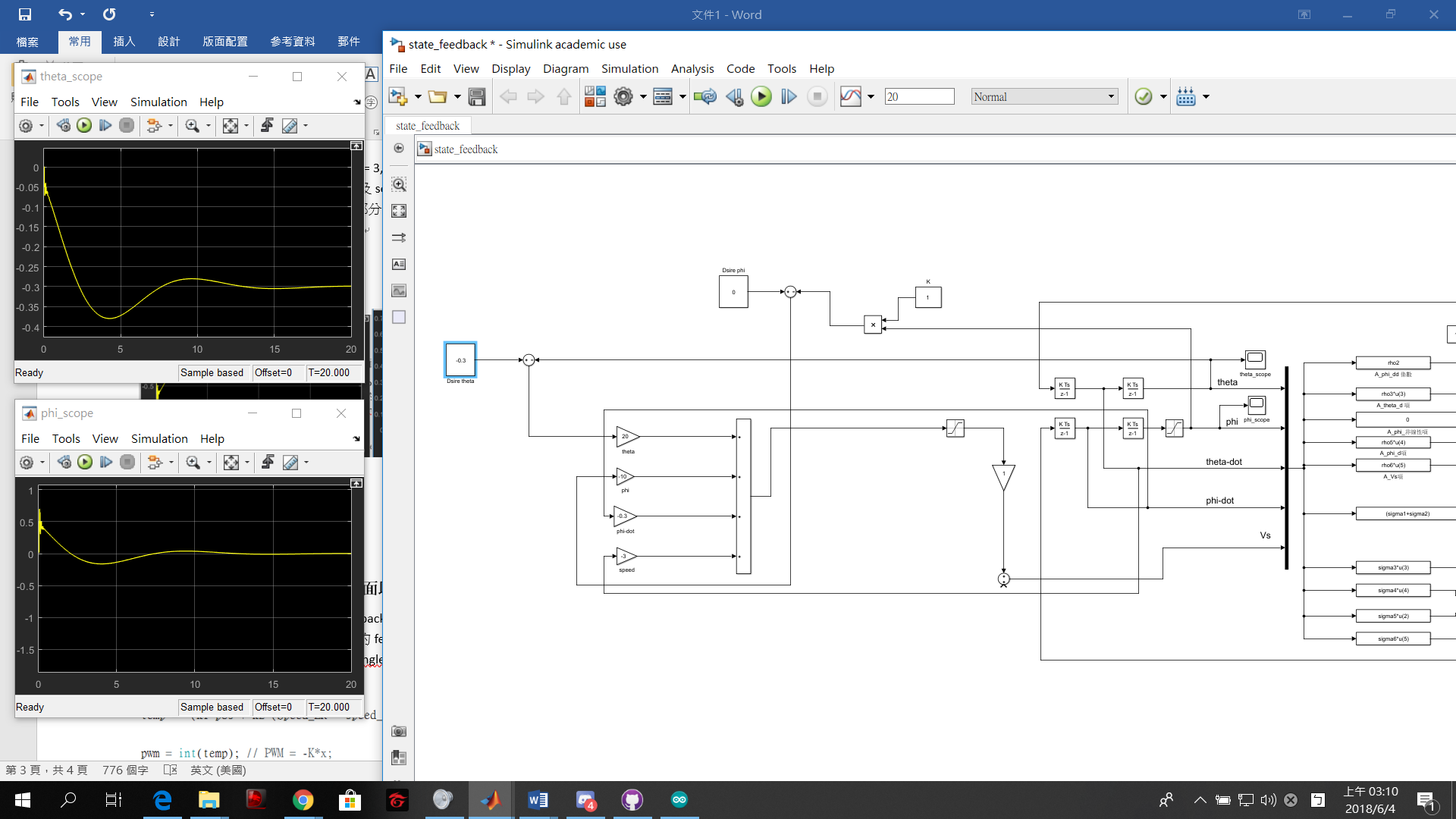
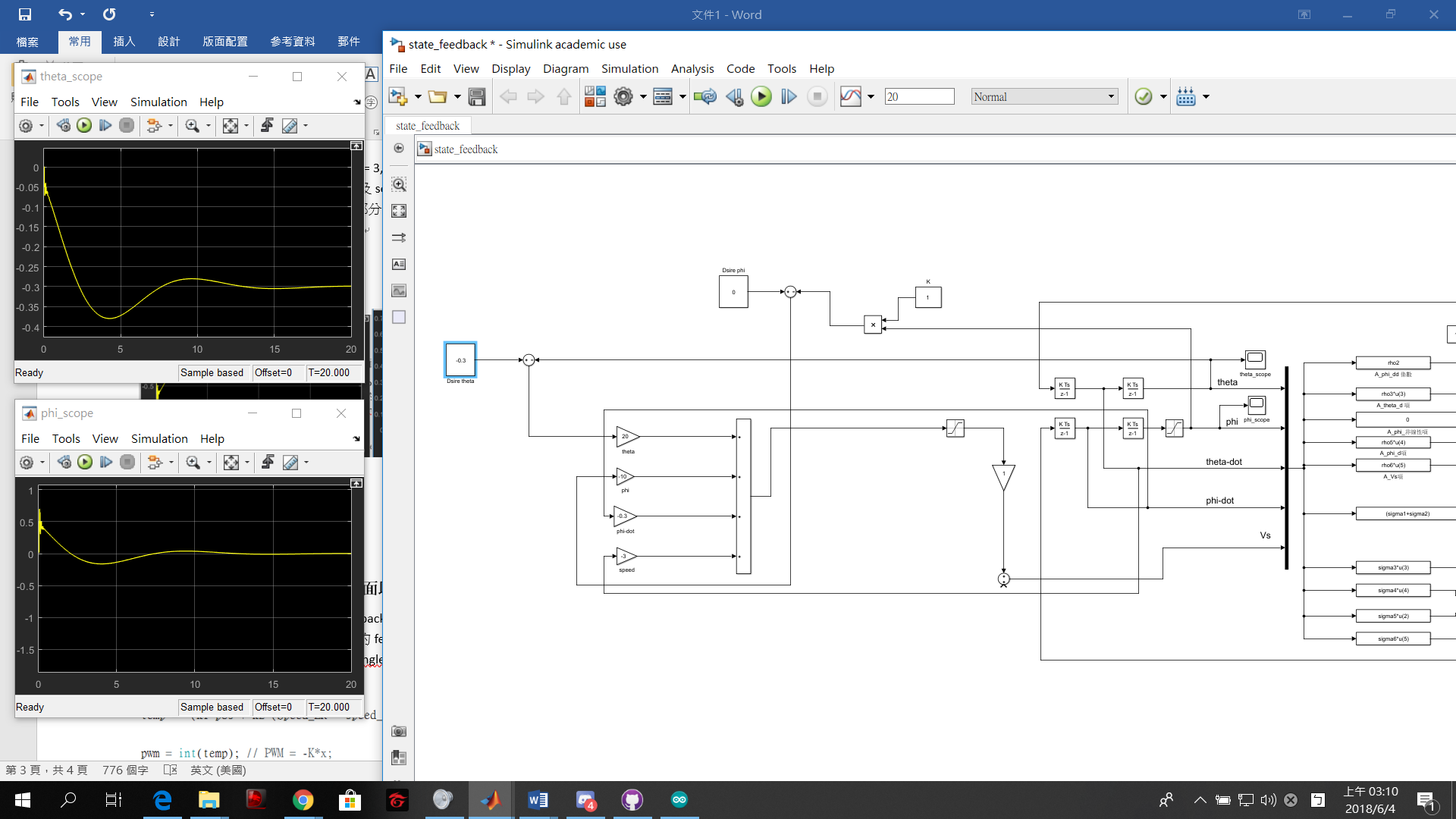
Desire theta=0.5

phi theta



Desire theta=-0.3

phi theta



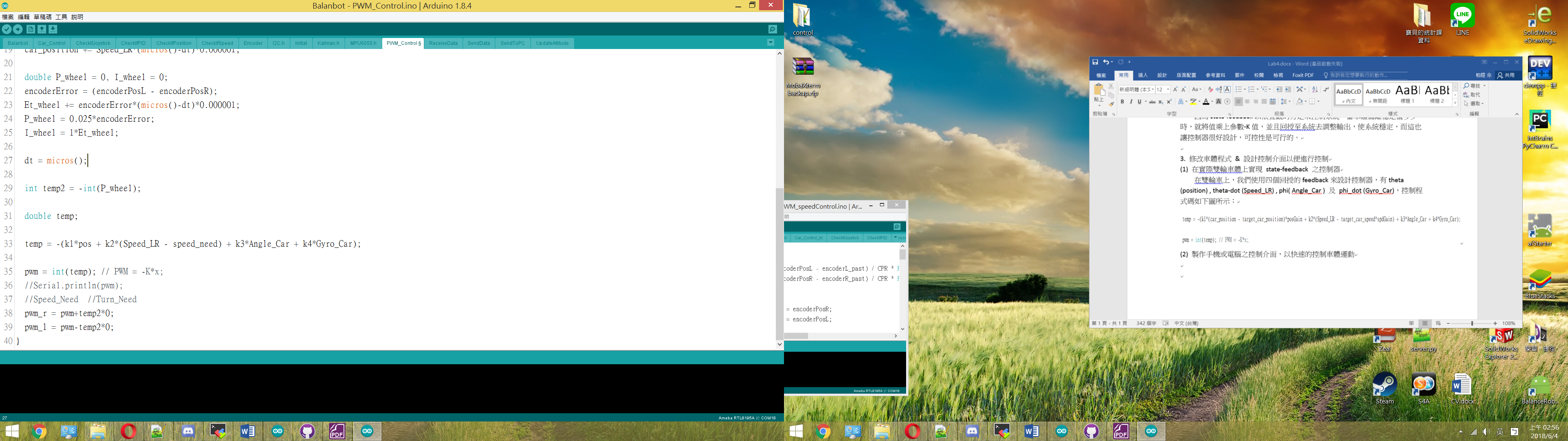
附檔有將數據輸入visual c#程式後的模擬動畫圖可供參考

測試結果後發現在Desire theta上升時一樣會有phi太大而達到1.57邊界的問題，因此在實際code上一樣要避免Desire theta太大的問題。

**3. 修改車體程式 & 設計控制介面以便進行控制**

(1) 在實際雙輪車體上實現 state-feedback 之控制器

在雙輪車上，我們使用四個回授的feedback來設計控制器，有theta (position) , theta-dot (Speed\_LR) , phi( Angle\_Car ) 及 phi\_dot (Gyro\_Car)，控制程式碼如下圖所示：



我們一開始使用模擬時的K1 = 20, K2 = 3, K3 = 10, K4 = 0.3去測試，但發現似乎是K1太大而使車子無法平衡，最終我們try and error得到最穩定的state feedback參數為K1 = 0.38, K2 = 3.45, K3 = 9, K4 = 0.35，但整體的平衡表現來看，跟PID控制器比較來看略為遜色，所以我們打算在期末專題以PID控制來呈現。

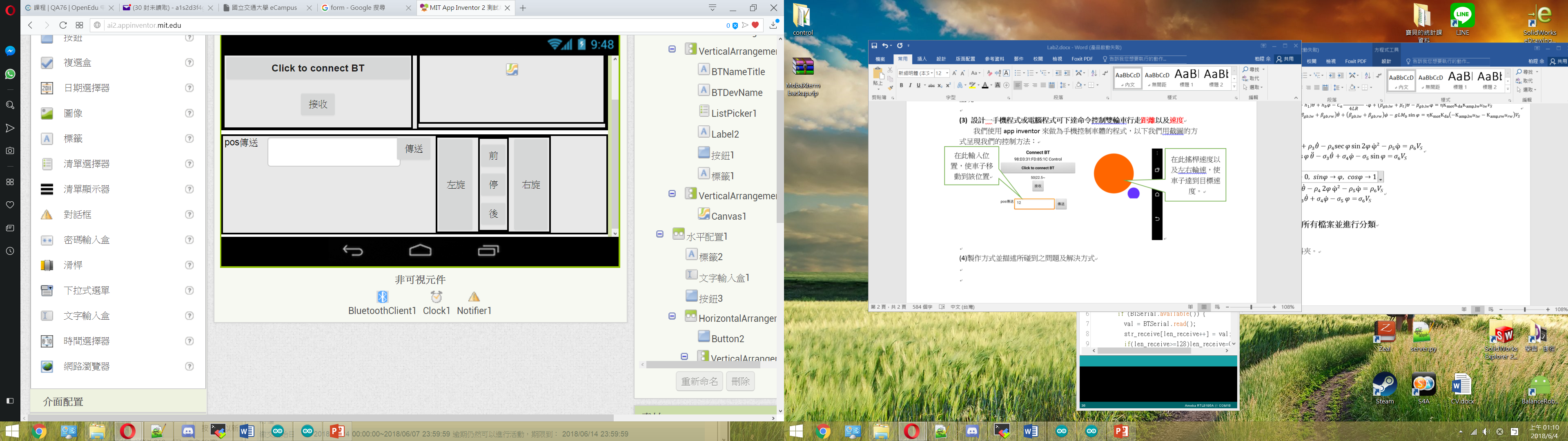
(2) 製作手機或電腦之控制介面，以快速的控制車體運動

我們使用app inventor來做為手機控制車體的程式，以下我們用截圖的方式呈現我們的控制方法：



在此搖桿速度以及左右輪速，使車子達到目標速度。

在此輸入位置，使車子移動到該位置



此為使用按鍵方式控制車體的各種移動模式

最終的期末demo我們打算再加上幾個按鈕 (像是後旋、自轉、或是配合創意發想的物體追蹤來設計特別按鈕…等等) 來增強我們的控制模式。