

2025 仰望盃全國科學 HomeRun 實作大賽

決賽作品說明書

隊伍名稱： 有病要吃【藥】

作品名稱： 中藥分裝器

科學概念 1： 正壓電效應

當裝置受到外力（壓力或機械應力）作用時，會在其內部產生電荷分離，材料表面產生電壓或電場的現象。本裝置運作過程中，藥粉掉落時，感測器受到壓力作用，內部電偶極矩發生重新排列，導致其內部極化方向發生變化。使感測器數值發生變化。

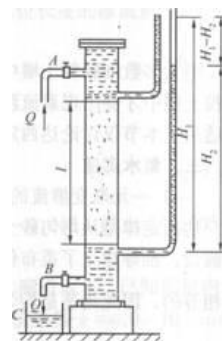
$$Q = Cd \times V = F \times d33$$

科學概念 2： 達西定律

適用於低速且層流的流體滲透流，表明流速與壓力差成正比，而與流體黏度成反比，在運作過程中為了使藥粉順利通過漏斗且不卡粉，因此我們設計了旋轉攪拌棒以及震動馬達解決此問題。

達西定律的基本形式： $Q = -k, A, \Delta h/L$

單位流量的形式： $q = -k, \Delta h/L$



決賽作品說明書內文

1. 發想動機：之前我因為季節交替引發了過敏性鼻炎，透過朋友推薦，我決定試試中藥調理。某天早上，我早早地來到了中醫診所，希望趁人少快速看診並拿藥。診所的掛號很順利，中醫師也很快給出了處方，並告訴我需要到藥房領取中藥。我心想應該不會等太久吧。

然而，當我拿著處方走到藥房時，卻發現長長的隊伍幾乎擠到了診所外。窗口前的兩位藥師正忙著，桌上堆滿了未完成的藥包。我注意到，藥師們不僅需要核對處方，還得親手抓藥、稱重、分包，甚至要回答患者的問題。想起醫師說的：「中藥需要講究準確的劑量，每一步都不能馬虎。」想到藥師們的專業和責任，我心裡的煩躁減輕了一些。

最終，在藥房裡等了近兩個小時才拿到藥包。藥師一邊將中藥交給我，一邊對我說：「不好意思，讓您久等了。」雖然過程漫長，但藥師的專業讓我感到敬佩。回家的路上，我心想：這麼專業的工作，卻需要藥師們如此忙碌，或許未來可以透過自動化分藥，減輕藥師負擔也減少患者的等待時間，讓領藥這件事更加順暢。

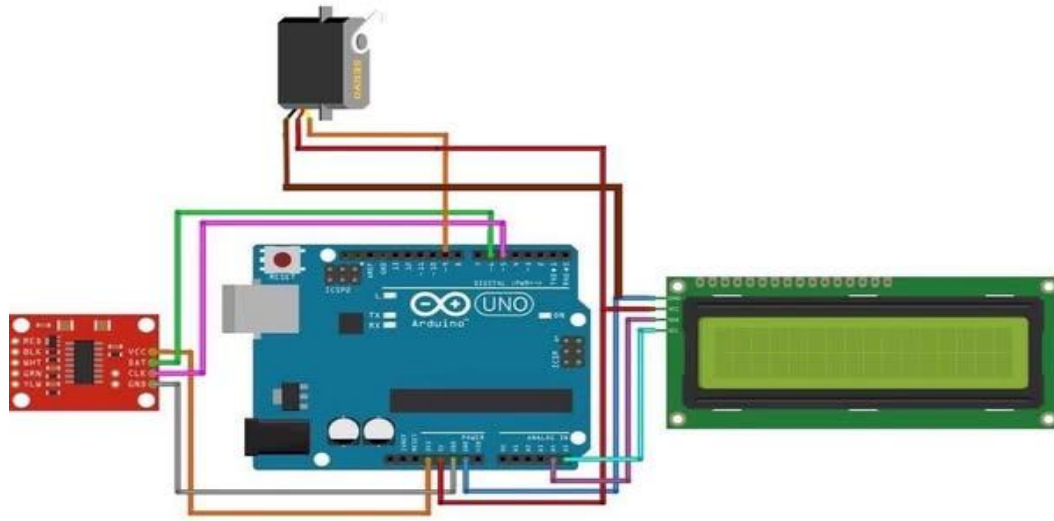
2. 作品創意性：

- A. 更加精準地控制藥粉劑量，減少人為誤差
- B. 減少人力成本還有時間，提高整體效率，無需手動操作，減少更多的麻煩，提高藥品便利性
- C. 減少藥品在空氣中接觸的時間，防止藥品的變質
- D. 減少浪費，搭配自動計算適量藥粉，避免多餘的浪費，也可減少藥品的成本
- E. 自動化操作，比手動更加來的便利，適合更大規模的醫院和藥廠使用
- F. 可儲存多種藥粉，根據不同的處方需求，自動混合成適合的藥粉劑量

3. 硬體及電路架構圖：

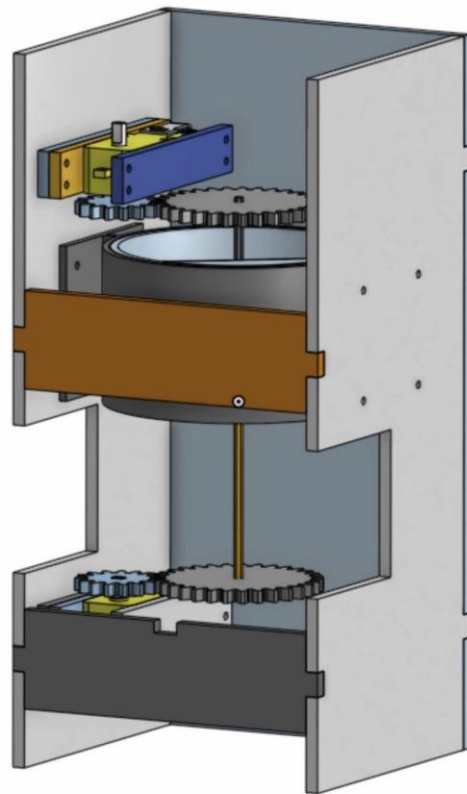
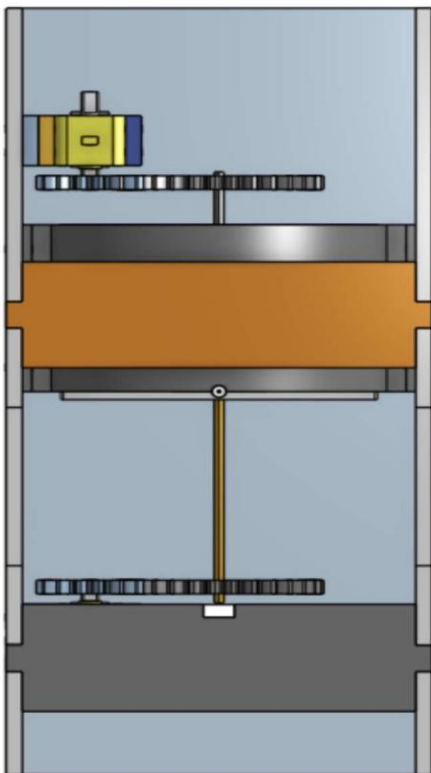
本作品使用：

Arduino Uno 主板、HX711 壓力傳感器模組、SG90 伺服馬達、TT 馬達

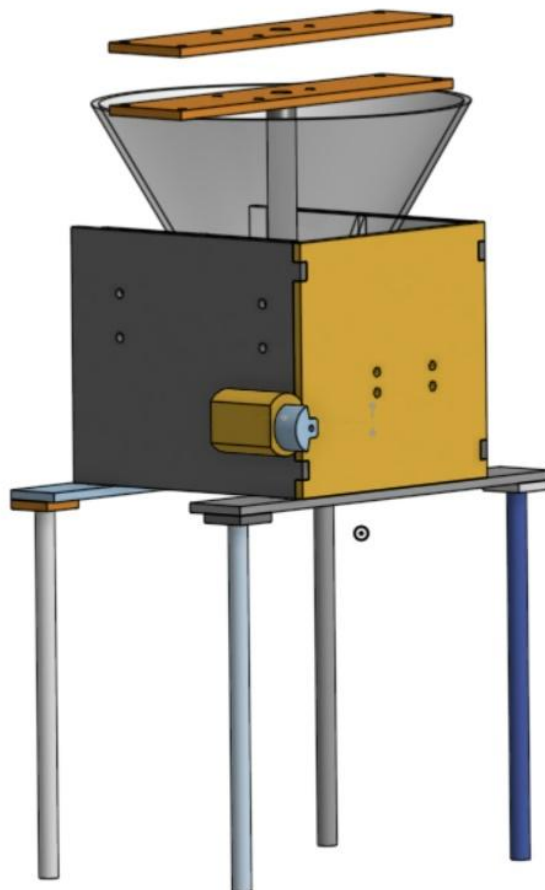
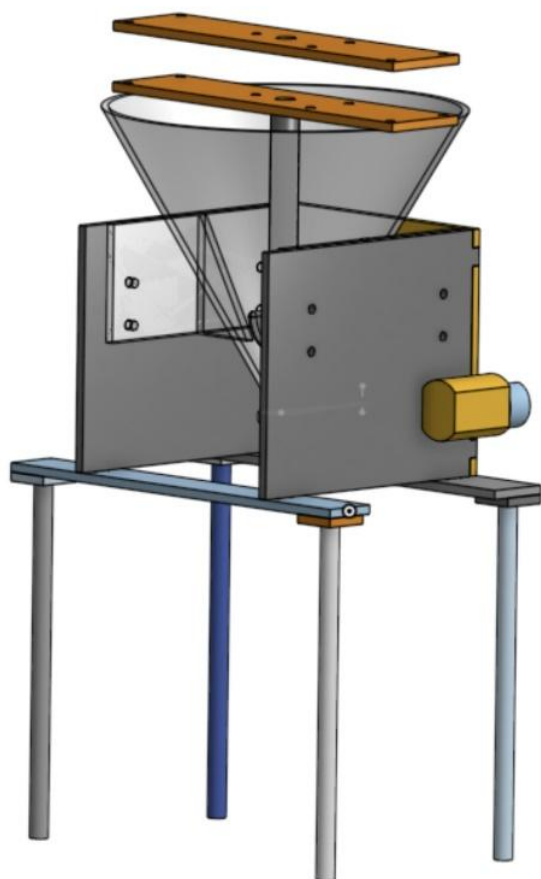
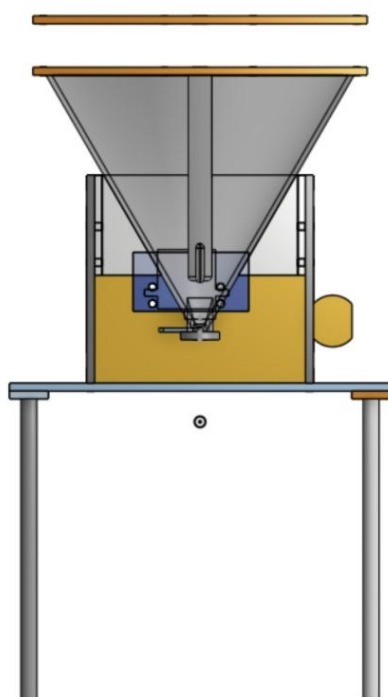


電路接線圖

第一版 - 「定量體積」量測版本



第二版 - 「重量感測器」量測版本



Arduino IDE 程式碼

```
sketch_mar19a.ino
1  #include "HX711.h"
2
3  // 接線設定
4  const int DT_PIN = 6;
5  const int SCK_PIN = 5;
6
7  const int scale_factor = -120; //比例參數，從校正式中取得
8
9  HX711 scale;
10
11 void setup() {
12     Serial.begin(9600);
13     Serial.println("Initialzing the scale");
14
15     scale.begin(DT_PIN, SCK_PIN);
16
17     Serial.println("Before setting up the scale:");
18
19     Serial.println(scale.get_units(5), 0); //未設定比例參數前的數值
20
21     scale.set_scale(scale_factor); // 設定比例參數
22     scale.tare(); // 歸零
23
24     Serial.println("After setting up the scale:");
25
26     Serial.println(scale.get_units(5), 0); //設定比例參數後的數值
27
28     Serial.println("請取重量為："); //在這個訊息之前都不要放東西在電子稱上
29 }
30
31 void loop() {
32
33     Serial.println(scale.get_units(10), 0);
34
35     scale.power_down(); // 進入睡眠模式
36     delay(1000);
37     scale.power_up(); // 結束睡眠模式
38 }
39
```

```
sketch_mar19a.ino
57     Serial.println("Initialzing the scale");
58
59     scale.begin(DT_PIN, SCK_PIN);
60     lcd.begin(16, 2); // 初始化LCD
61     lcd.setBacklight(255);
62     lcd.clear();
63
64     Serial.println("Before setting up the scale:");
65
66     Serial.println(scale.get_units(5), 0); //未設定比例參數前的數值
67
68     scale.set_scale(scale_factor); // 設定比例參數
69     scale.tare(); // 歸零
70
71     Serial.println("After setting up the scale:");
72
73     Serial.println(scale.get_units(5), 0); //設定比例參數後的數值
74
75     Serial.println("Readings:"); //在這個訊息之前都不要放東西在電子稱上
76 }
77
78 void loop() {
79
80     Serial.println(scale.get_units(10), 0);
81     lcd.clear();
82     lcd.setCursor(0, 0); //設定游標位置 (字,行)
83     lcd.print("Weight: ");
84     lcd.setCursor(9, 0);
85     float weight = scale.get_units(10);
86     //避免出現負數
87     if(weight<=0){
88         weight = 0;
89     }
90     lcd.print(weight,0);
91     lcd.setCursor(13, 0);
92     lcd.print("g");
93     scale.power_down(); // 進入睡眠模式
94     delay(500);
95     scale.power_up(); // 結束睡眠模式

```

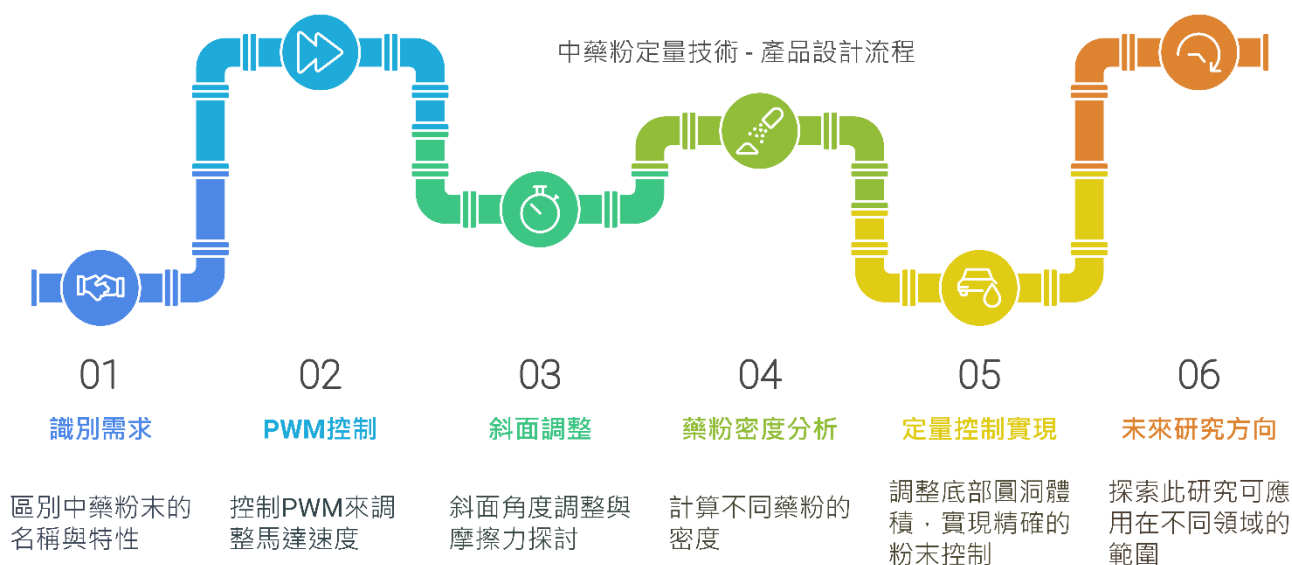
4. 作品成果說明：

第一版的實作產品設計概念主要分為三個部分(內容器、外容器、下擋板)，內容器內含中藥粉末、斜面與四分之一簍空的底，主要負責刮平藥粉讓下一層外容器挖空的洞可以完整填滿藥粉，一開始擋板會擋住外容器的洞讓藥粉不會掉落，等藥粉添滿後，下擋板轉動藥粉掉落完成。

一開始我們希望藉由固定體積的藥粉掉落，與不同藥粉密度分析，就可以回推調整不同的外容器圓洞大小，藉此達到掉落的藥粉皆為相同重量的目的。

但是我們很快發現隨著內容器放入藥粉的多寡，重量的壓縮會輕微影響圓洞內藥粉的含量多寡，因此我們經由工程設計流程步驟地回推，再重新設計了第二版作品解決第一版問題的不足。

第一版作品 - 「定量體積」 | 量測版本 - 產品設計流程



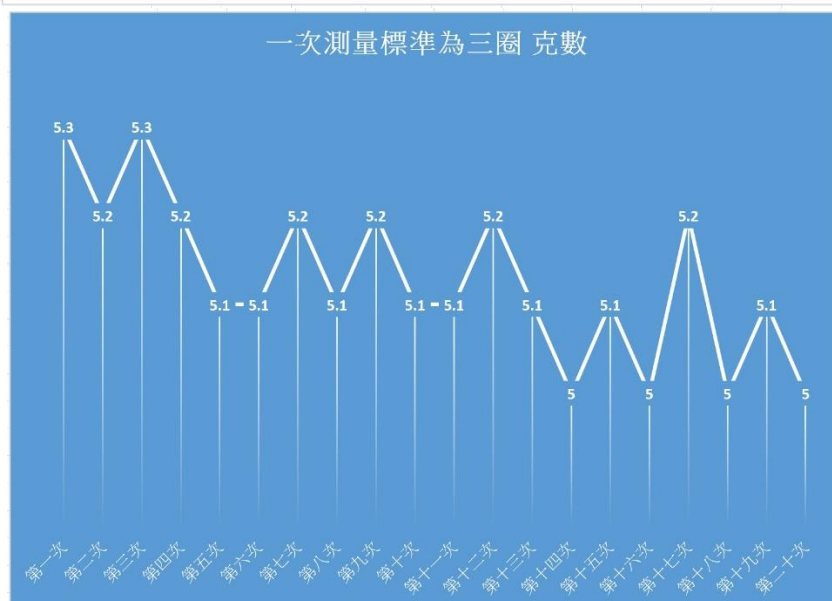
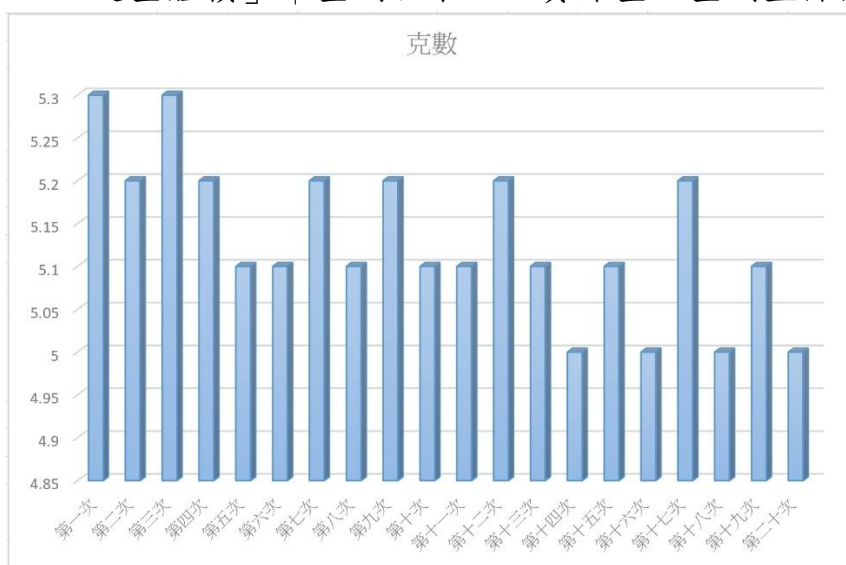
第一版作品 - 設計與實作時遭遇的困難與解決方法

	實作時遭遇的困境	實際解決的辦法
1	較粗的粉末倒入第一版機器轉動時會卡住導致轉動不順。	* 修改第一版結構與機構 * 重新設計第二版機器
2	第一版機器以體積為量測基準，隨著上方粉末重量減少，容易產生重量不準確的問題。	* 重新設計第二版以重量為量測基準的機器

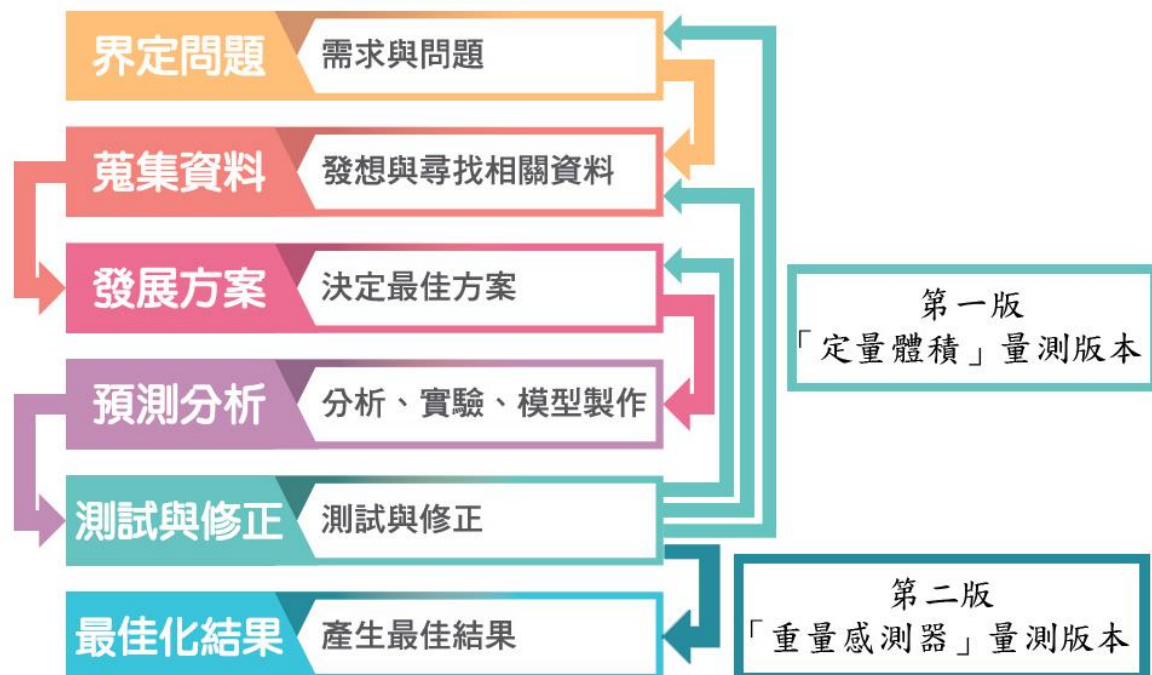
第一版作品 - 「定量體積」 | 量測版本 - 實際量化量測表格

一次測量標準為三圈	
次數	克數
第一次	5.3
第二次	5.2
第三次	5.3
第四次	5.2
第五次	5.1
第六次	5.1
第七次	5.2
第八次	5.1
第九次	5.2
第十次	5.1
第十一次	5.1
第十二次	5.2
第十三次	5.1
第十四次	5
第十五次	5.1
第十六次	5
第十七次	5.2
第十八次	5
第十九次	5.1
第二十次	5

第一版作品 - 「定量體積」 | 量測版本 - 實際量化量測直條圖、折線圖



第一版與第二版 - 工程設計流程步驟圖



產出的結果若有問題時，即會開始執行反覆測試與修正的動作，甚至可能重新執行前面的步驟

藉由以上工程設計流程步驟，我們設計出了第二版作品，原理是使用重量感測器量測，就可以避免重量偏差問題

第二版作品透過 **Arduino** 微控制器進行藥粉的自動定量分裝，確保每次配藥的劑量準確無誤，減少人為誤差，提升醫療效率。

當醫生開立藥物時，系統首先接收醫生所輸入的指令，包括藥粉的種類與所需劑量。這些資訊經過數據處理後，透過網路或藍牙傳輸至 **Arduino** 晶片，作為控制系統運作的核心指令。接收到指令後，**Arduino** 會啟動程式並開始執行分裝過程。首先，馬達開始轉動，驅動藥粉輸送裝置，使藥粉緩緩掉落至秤重容器中。

在藥粉掉落的同時，感測器開始即時測量質量，並將數據回傳給 **Arduino** 進行判斷。一旦感測器偵測到藥粉重量達到設定的劑量，**Arduino** 便會立即發出停止信號，指示伺服馬達停止運轉，終止藥粉輸送。此時，藥粉分裝過程完成，確保每次配藥的劑量符合醫療標準。

運作過程



第二版作品 - 設計與實作時遭遇的困難與解決方法

	實作時遭遇的困境	實際解決的辦法
1	第二版機器掉落粉末不順暢	* 把洞口改大 * 根據達西定律 - 3D 設計攪拌棒 * 根據達西定律 - 3D 設計震動馬達
2	第二版機器無法擋住粉末掉落	* 調整擋板位置 * 在擋板位置新增零件
3	第二版機器結構不穩度容易晃動	* 不影響傳感器的情況下強化結構補強
4	重量傳感器感應不準確的問題	* 找出最準確的位置標註記號

透過這套自動化系統，不僅能有效提高藥粉分裝的精準度與穩定性，還能減少人工作業帶來的誤差，確保病患獲得最適當的藥物治療。未來，該系統還可進一步整合 AI 技術與遠端監控，實現更智能化的藥物管理模式。

未來展望：



A. 材料科學領域

- 奈米材料研究（如納米碳管、石墨烯分裝）
- 金屬與陶瓷粉體混合（用於高性能材料製造）
- 複合材料製造（確保不同粉體成分均勻混合）
- 半導體與電子材料製造（矽粉、氧化鋁等高精度分裝）

B. 烹飪領域

- 調味料分裝（鹽、糖、胡椒粉、香料等）
- 烘焙粉體控制（發酵粉、可可粉、奶粉）
- 工業食品生產（如即溶咖啡、蛋白粉、奶茶粉）

C. 化學領域

- 實驗室試劑分裝（氧化鋁粉、催化劑粉末）
- 塗料與顏料生產（確保顏色與化學成分穩定）
- 塑膠與聚合物材料（均勻混合塑膠添加劑）
- 電池材料製造（如鋰電池正極、負極粉體精準填充）

D. 3D 列印領域

- 選擇性雷射燒結（SLS）（高精度聚合物粉末供應）
- 選擇性雷射熔化（SLM）（金屬粉末，如鈦、鋁、不鏽鋼）
- 陶瓷 3D 列印（如氧化鋁、二氧化矽粉末供應）

5. 參考文獻：

壓電效應. <https://zh.wikipedia.org/zh-tw/壓電效應>

達西定律. <https://zh.wikipedia.org/zh-tw/達西定律>

高中物理網. <http://gaozhongwuli.com/zt/niudun/470098.html>

斜面原理. <https://zetria.tw/physics/04d3b626cf>