

國立龍潭高級中學學生專題製作競賽

製作專題名稱：室內盆栽自動澆水系統

科別：電機科

班級：三年級

學生姓名：徐志有、李東祐、

指導老師：劉哲雄老師

一、現況調查

由於父親時常以種植盆栽當作休閒，但由於生活忙碌以至於盆栽時常缺水壞死，看到父親如此難過，又加上上班壓力，於心不忍，想為父親減少壓力與失落。透過三年級學的 arduino 這樣的微控系統，有了此構想:室內盆栽自動澆水。

設計動機與目的

1. 三年級上過以 arduino 微微電腦制實習，所以有了一些 c/c++ 的程式設基礎，也發覺 arduino 微控器系統功能強大，硬體價格非常便宜，軟體免費、網路資源資料相當多，特別適合高中職學生在不用花費太多金錢下進行開發各種控制系統與專題製作，希望以此創作改善父親的困擾。
2. 經由此一專題製作強化自己對 arduino 微控器應用能力與學習與組員之間分工合作、集體創作。

二、創意發想歷程



構想流程圖

1. **發現盆栽疏於澆水問題:**經由父親生活忙碌維持家計，疏於為盆栽澆水，導致盆栽枯萎。
2. **構思問題解決辦法:**以 arduino 微控器設計製作一套具有自動感測濕度、進行交水的動作，程式也有防止澆過多水的設計，防止水分過多，根部壞死。
3. **蒐集資料與購買材料:**由網際網路與圖書館蒐集相關硬體與軟體技術資料，確立硬體設計電路圖經老師審視修正後提出申請，購買材料及模型製作。
4. **程式設計與硬體製作:**取得材料後分組設計程式與硬體組裝。
5. **系統測試與美工:**進行軟硬整合系統測試與修正，確認無誤完成程式設計與硬體電路後進行外觀美化。
6. 完成。

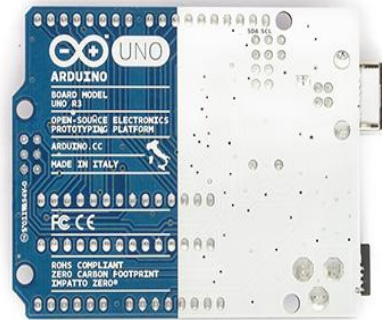
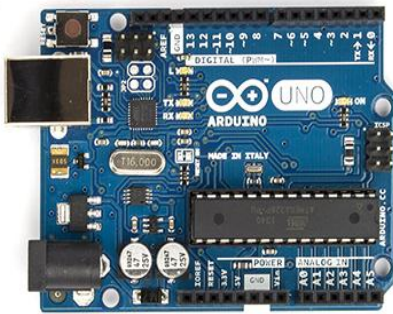
三、應用潛能分析

利用 arduino 微控器為主微電腦控制實習及電子學實習中課程所學及老師從旁指導協助加上不斷測試與修正，就可以逐漸打造出理想的成品模型。本專題開發系統 arduino 硬體與軟體設備屬於開放系統，相信能為長途旅遊而無法為盆栽澆花和太忙而疏於照顧的人們帶來方便。

四、成本分析：arduino 微控器開發版、L298N 電機驅動模塊、HS422 標準舵機(伺服馬達)、舵機支架 Arduino 51 arm U 型、濕度感測器、DC 幫浦。

(一)、主要零件圖片外觀:

1.arduino 微控器開發版



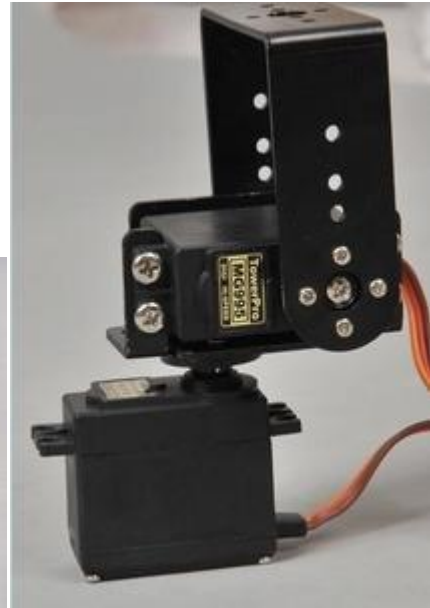
2.L298N 電機驅動模塊



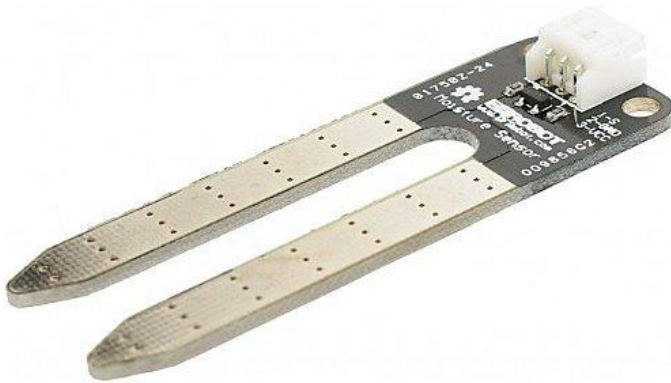
3. HS422 標準舵機(伺服馬達)



4. 舵機支架短 U 型



5. 濕度感測器



(二)、成本分析

總金額：2245 元。

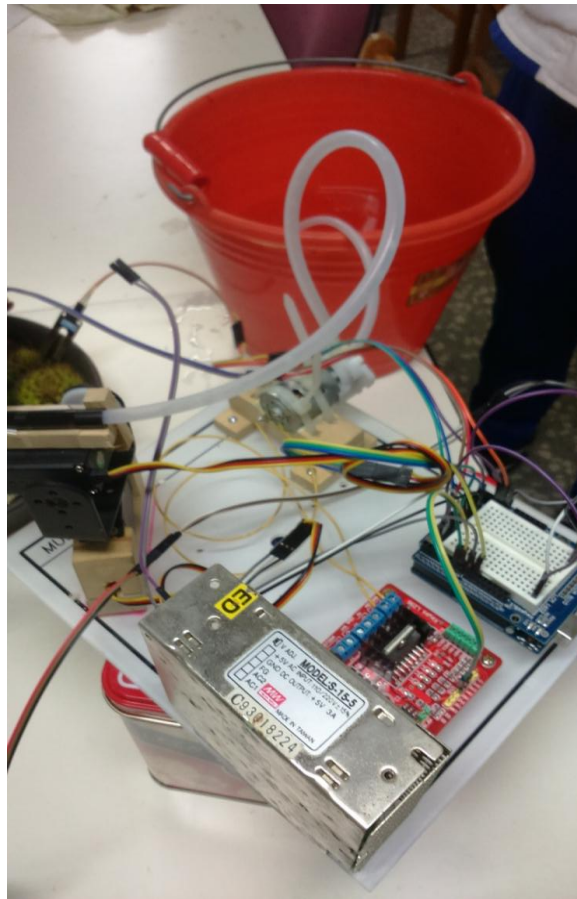
	數量	金額	合計
1.arduino 微控器開發版	1	200	200
2.DC 幫浦	1	600	600

3. L298N 電機驅動模塊	1	200	200
4. HS422 標準舵機(伺服馬達)	2	450	900
5. 舵機支架 Arduino 51 arm U 型	1	150	150
6.濕度感測器	3	65	195

五、作品說明圖說明:

1. 模型結構圖及控制電路：

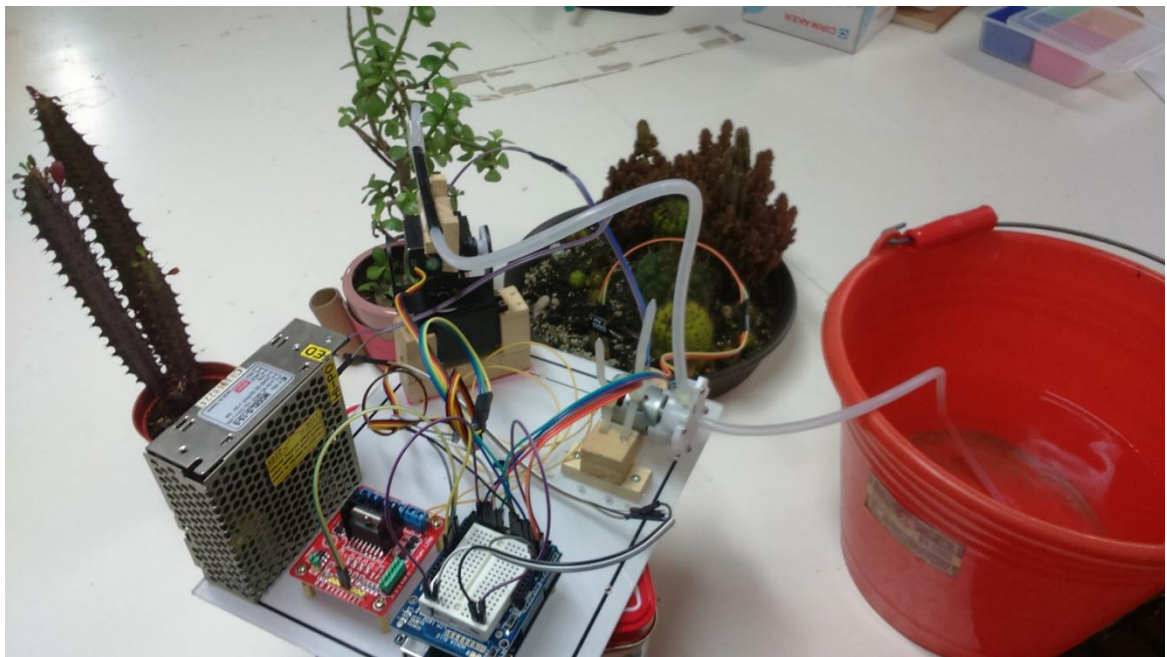
(1) 模型結構圖



上視圖



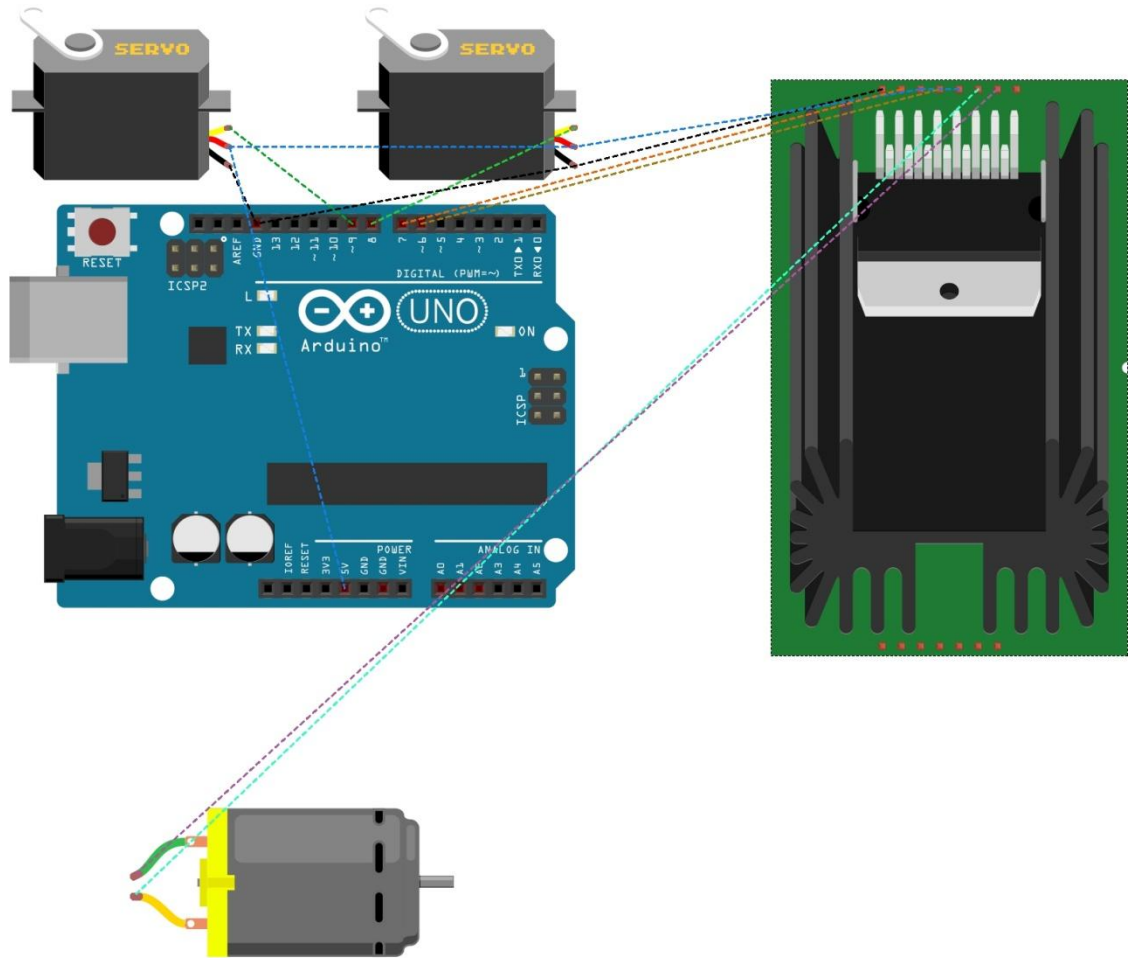
側視圖(長)



側視圖(寬)

(2) 控制電路:

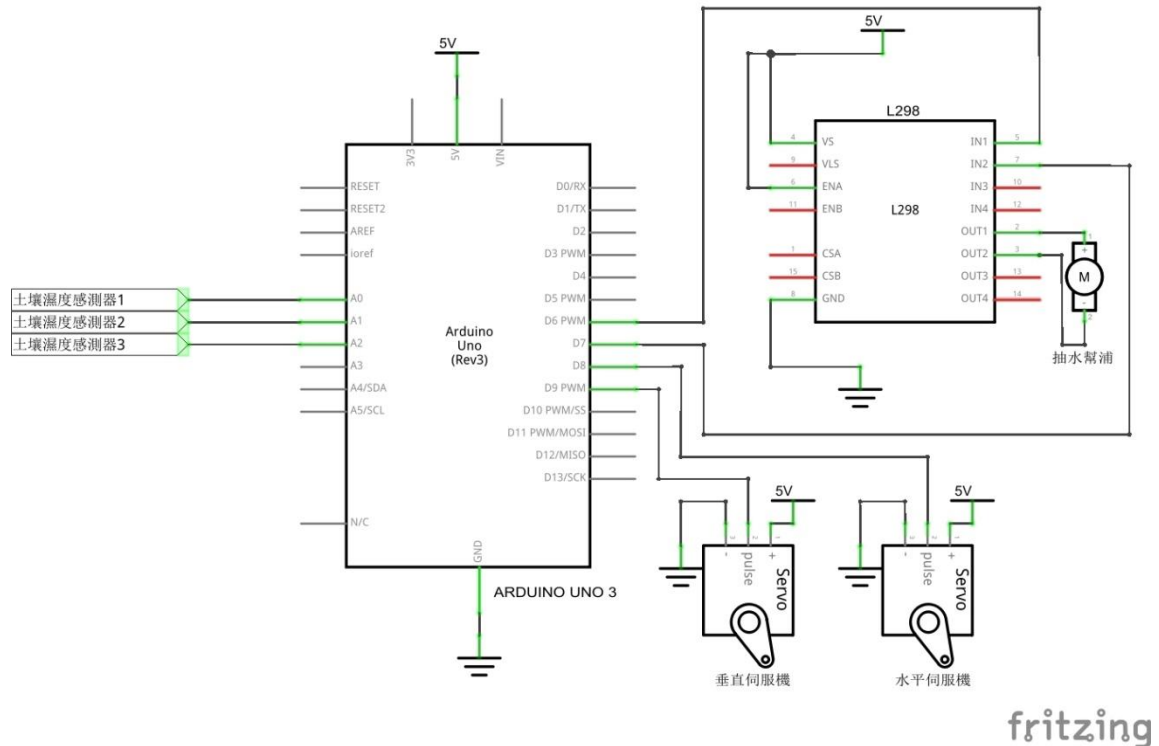
① 元件架構圖:



② 電路圖設計

國立龍潭高中電機科

國立龍潭高中電機三專題製作成果發表			
指導老師:劉哲雄 徐志有 李東祐 製作			
Project	室內盆栽自動澆水系統		
Filename	auto to water 3.fzz	Rev	1.0版
Date	20 三月 2015 16:47:01	Sheet	1/1



2. 程式碼註解說明:

/*盆栽自動澆水系統 2015 年龍潭高中電機三專題製作 徐志有 李東祐 顏臣廷 */

```
#include <Servo.h>
```

```
#define triger 250
```

```
//定義第一個盆栽參數設定
```

```
#define servoPosition0 20 //設定第 1 個盆栽位置水平伺服機(10~160 範圍內)
```

```
#define triggerValue0 135 //設定第 1 個盆栽觸發濕度值,最小值 0(最乾),  
1024(最濕).
```

```
#define wateringTime0 35 //設定第 1 個盆栽澆水時間秒數(ms) 為每 35ms 約  
70 cc 澆水量
```

```
//定義第一個盆栽參數設定
```

```
#define servoPosition1 60 //設定第 2 個盆栽位置水平伺服機(10~160 範圍內)
```

```
#define triggerValue1 180 //設定第 2 個盆栽觸發濕度值,最小值 0(最乾),  
1024(最濕)
```

```
#define wateringTime1 20 //設定第 2 個盆栽澆水時間秒數(ms) 澆水量
```



```

//Define the parameters of the flower3
#define servoPosition2  105  //設定第 3 個盆栽位置水平伺服機(10~160 範圍內)
#define triggerValue2    200  //設定第 3 個盆栽觸發濕度值,最小值 0(最乾),1024(最濕)
#define wateringTime2  15  //設定第 3 個盆栽澆水時間秒數(ms) 澆水量

Servo horizontalServo; //建立伺服機物件:水平伺服機
Servo verticalServo;   //建立伺服機物件:垂直伺服機

//定義濕度感測器接腳與 arduino 的連接
const int analogInPin0 = A0; //第 1 個盆栽
const int analogInPin1 = A1; //第 2 個盆栽
const int analogInPin2 = A2; //第 3 個盆栽

//初始濕度感測器接收值(最乾)
int moistureValue0 = 0;
int moistureValue1 = 0;
int moistureValue2 = 0;

//取樣 30 次濕度感測器接收值得總和再求平均值
long int moistureSum0 = 0; //第 1 個盆栽
long int moistureSum1 = 0; //第 2 個盆栽
long int moistureSum2 = 0; //第 3 個盆栽

//定義抽水直流馬達正負接腳與 arduino 接腳
const int pumpAnodePin = 6; //D6 腳連接馬達正接腳
const int pumpCathodePin = 7; //D7 腳連接馬達負接腳

void setup()
{
    Serial.begin(9600); //開啟監督視窗

```

```

horizontalServo.attach(8); // D8 腳連接至水平伺服機信號腳
verticalServo.attach(9); // D9 腳連接至垂直服機信號腳
initPosition(); //初始化伺服機位置
delay(500);
pinMode(pumpAnodePin, OUTPUT); //設定 D6,D7 馬達連接腳為輸出
pinMode(pumpCathodePin, OUTPUT);
}

void initPosition() //初始化伺服機位置副程式
{
    verticalServo.write(15); //垂直仰角 15 度
    horizontalServo.write(25); //水平轉角 25 度
    delay(1000); //延遲 1 秒讓伺服機完成轉至初始化位置
}

void loop()
{

    delay(1000);
    moistureSampling(); //呼叫濕度取樣副程式

    if(moistureValue0 < triggerValue0) //判斷第 1 個盆栽觸發濕度值是否低於
設定濕度值
    {
        proWatering(servoPosition0, wateringTime0); //低於設定濕度值時,開始澆
水
        return;
    }
    if(moistureValue1 < triggerValue1) //判斷第 2 個盆栽觸發濕度值是否低於
設定濕度值

    {
        proWatering(servoPosition1, wateringTime1); //低於設定濕度值時,開始澆

```

水

```
    return;
  }
  if(moistureValue2 < triggerValue2) //判斷第 3 個盆栽觸發濕度值是否低於
  設定濕度值
  {
    proWatering(servoPosition2, wateringTime2); //低於設定濕度值時,開始澆
    水
    return;
  }
}
```

```
void moistureSampling() // 讀取土壤濕度感測器取樣值副程式
{
  for(int i = 0; i < 30; i++) //30ms 內讀取取樣 30 次累加取樣值
  {
    moistureSum0 = moistureSum0 + analogRead(analogInPin0);
    moistureSum1 = moistureSum1 + analogRead(analogInPin1);
    moistureSum2 = moistureSum2 + analogRead(analogInPin2);
    delay(100);
  }
  moistureValue0 = moistureSum0 / 30; //計算取樣各盆累加值除以 30,取得平
  均值
  moistureValue1 = moistureSum1 / 30;
  moistureValue2 = moistureSum2 / 30;

  // 顯示各盆栽濕度值於監控視窗
  Serial.print("Moisture0 = ");
  Serial.print(moistureValue0);
  Serial.print("\t Moisture1 = ");
  Serial.print(moistureValue1);
  Serial.print("\t Moisture2 = ");
  Serial.println(moistureValue2);
}
```

```

Serial.println();

moistureSum0 = 0; //清除各盆濕度值為零,準備重新讀取
moistureSum1 = 0;
moistureSum2 = 0;
delay(4000);    //延遲 4 秒
}

void proWatering(int servoPosition, int wateringTime) //專業掃描澆水副程式
{
    horizontalServo.write(servoPosition - 10); //設定水平伺服機至盆栽位置
    delay(500); //等待水平伺服機轉至正確位
    digitalWrite(pumpAnodePin, HIGH); //啟動馬達開始澆水
    digitalWrite(pumpCathodePin, LOW);

    for(int pos1 = 15; pos1 < 33; pos1 +=6) //垂直伺服機仰角變動
    {
        verticalServo.write(pos1);
        for(int pos = servoPosition - 10; pos <= servoPosition + 10; pos += 1)
            // goes from 0 degrees to 180 degrees          //25,160
            {
                // 第一階段的第一步
                horizontalServo.write(pos);                //水平伺服機轉至變數 'pos'指定
                //角度
                delay(wateringTime);                        //等待水平伺服機轉至變數 'pos'
                //指定角度
            }

        verticalServo.write(pos1+3); //垂直伺服機仰角增加 3 度
        for(int pos = servoPosition + 10; pos>= servoPosition - 10; pos-=1)
            // goes from 180 degrees to 0 degrees
            {
                horizontalServo.write(pos);                //水平伺服機轉至變數 'pos'指定角
                //度
            }
    }
}

```

```

        delay(wateringTime);          //等待水平伺服機轉至變數 'pos'指
定角度
    }
}

for(int pos1 = 33; pos1 > 15; pos1 -= 6)//
{
    verticalServo.write(pos1);
    for(int pos = servoPosition - 10; pos <= servoPosition + 10; pos += 1)
// goes from 0 degrees to 180 degrees    //25,160
    {
        // in steps of 1 degree
        horizontalServo.write(pos);      //水平伺服機轉至變數 'pos'指定角度
        delay(wateringTime);            //等待水平伺服機轉至變數 'pos'指定
角度
    }

    verticalServo.write(pos1-3);    //垂直伺服機仰角減 3 度
    for(int pos = servoPosition + 10; pos >= servoPosition -10; pos-=1)
// goes from 180 degrees to 0 degrees
    {
        horizontalServo.write(pos);      //水平伺服機轉至變數 'pos'指定角度
        delay(wateringTime);            //等待水平伺服機轉至變數 'pos'指定
角度
    }
}

digitalWrite(pumpAnodePin, LOW); //馬達停止運轉
digitalWrite(pumpCathodePin, LOW);
delay(500);
initPosition();
}

void watering(int servoPosition, int wateringTime) //一般澆水副程式
{

```

```

    Serial.println(servoPosition);
    horizontalServo.write(servoPosition);
    delay(1000);
    digitalWrite(pumpAnodePin, HIGH);
    delay(10000);
    digitalWrite(pumpAnodePin, LOW);
    delay(3000);
}

//單一定點澆水
/*
void basicWatering(int servoPosition, int wateringTime)
{
    Serial.println(servoPosition);
    horizontalServo.write(servoPosition);
    delay(800);
    digitalWrite(pumpAnodePin, HIGH);
    delay(wateringTime);
    digitalWrite(pumpAnodePin, LOW);
    delay(500);
    initPosition();
} */

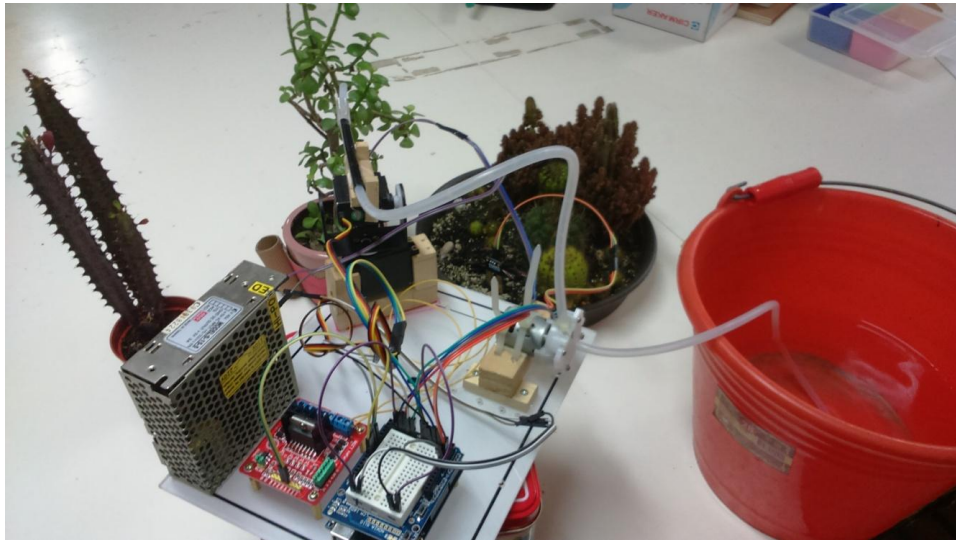
```

3.製作及完成圖：





電路測試及組裝圖



作品完成測試圖

六、參考資料

- 1.arduino 官方網站 <http://arduino.cc/>
- 2.最簡單的互動設計 Arduino 一試就上手 碁峰 孫駿榮、吳明展、盧聰勇
- 3.超圖解 Arduino 互動設計入門(第二版) 旗標 趙英傑
4. Arduino 互動設計專題與實戰 碁峰 柯博文
5. Arduino 最佳入門與應用 碁峰 楊明豐
6. Arduino 自造指南 碁峰 原文作者：John Boxall 譯者：曾繁勛,陳麒元,趙涵捷