

控工實驗 FINAL 結報

循跡輪型機器人

Group 2

林珮玉 E24084096

林玠志 E24083074

林哲緯 E24086129

薛博文 F14071148

蔡孟宗 F44071055

Control Engineering Laboratory – Lab Final Report

Group 2 - 林珮玉 林哲緯 林玠志 薛博文 蔡孟宗

I. Objectives

循跡輪型機器人

II. Exploration

將事先組好的輪型機器人，搭配顏色 sensor，實現機器人能沿著給定道路前進。

道路等比例圖：

unit:cm

實現 Code

```
#pragma config(Sensor, S4, US, sensorEV3_Ultrasonic)
#pragma config(Motor, motorB, MotorB, tmotorEV3_Large,
PIDControl, encoder)
/*!!Code automatically generated by 'ROBOTC' configuration
wizard !!*/
void TurnLeft(int angle);
void TurnRight(int angle);
void set2MotorSpeed(int speed);
int CheckRight(int angle);
int CheckLeft(int angle);
#define Near 20 // cm
#define Motor1 motorB
#define Motor2 motorC
int white = 30;
task main() {
    while (true) {
        if (getTouchValue(S1) == 1)
            break;
        delay(50);
    }
    resetGyro(S2);
    int goRight = 0;
    int goLeft = 0;
    int range = 30;
    while (true) {
        if (getColorReflected(S3) < white)
            // normal case: go straight
            set2MotorSpeed(10);
        else {
            // if we are in wrong position, reverse a little
            bit, and then
            // dicide go R or L
            while (getColorReflected(S3) >= white)
                set2MotorSpeed(-10);

            goRight = CheckRight(range);
            TurnLeft(range);
```

```

        goLeft = CheckLeft(range);
        TurnRight(range);

        if (goRight > goLeft) {
            // if we decide to go right, turn right until
            we meet the black
            while (getColorReflected(S3) >= white)
                TurnRight(1);
            TurnRight(7);
        } else {
            while (getColorReflected(S3) >= white)
                TurnLeft(1);
            TurnLeft(7);
        }
    }
}

int CheckRight(int angle) { // Cnt how many black sensor meet
when turn right
    resetGyro(S2);
    int goRight = 0;

    repeatUntil(getGyroDegrees(S2) > angle) {
        displayTextLine(2, "%d", getGyroDegrees(S2));
        if (getColorReflected(S3) < white)
            goRight++;

        setMotorSpeed(Motor2, -10);
        setMotorSpeed(Motor1, 10);
    }
    set2MotorSpeed(0);
    return goRight;
}

int CheckLeft(int angle) {
    resetGyro(S2);
    int goLeft = 0;

```

```

    repeatUntil(getGyroDegrees(S2) < -1 * angle) {
        displayTextLine(2, "%d", getGyroDegrees(S2));
        if (getColorReflected(S3) < white)
            goLeft++;

        setMotorSpeed(Motor2, 10);
        setMotorSpeed(Motor1, -10);
    }
    set2MotorSpeed(0);
    return goLeft;
}

void TurnRight(int angle) {
    resetGyro(S2);
    repeatUntil(getGyroDegrees(S2) > angle) {
        displayTextLine(2, "%d", getGyroDegrees(S2));

        setMotorSpeed(Motor2, -10);
        setMotorSpeed(Motor1, 10);
    }
    set2MotorSpeed(0);
    return;
}

void TurnLeft(int angle) {
    resetGyro(S2);
    repeatUntil(getGyroDegrees(S2) < -1 * angle) {
        displayTextLine(2, "%d", getGyroDegrees(S2));

        setMotorSpeed(Motor2, 10);
        setMotorSpeed(Motor1, -10);
    }
    set2MotorSpeed(0);
    return;
}

void set2MotorSpeed(int speed) {
    setMotorSpeed(Motor1, speed);

```

```
setMotorSpeed(Motor2, speed);  
}
```

程式中先寫好 TurnRight(int angle)、TurnLeft(int angle)、CheckRight(int angle)、CheckLeft(int angle)以及 set2MotorSpeed(int speed)這幾個函式，再回到主函式中，利用 getTouchValue(S1) 控制機器人開始運作。

TurnRight/Left 的功能是使機器人向對應方向轉指定角度，角度為傳入的整數值。

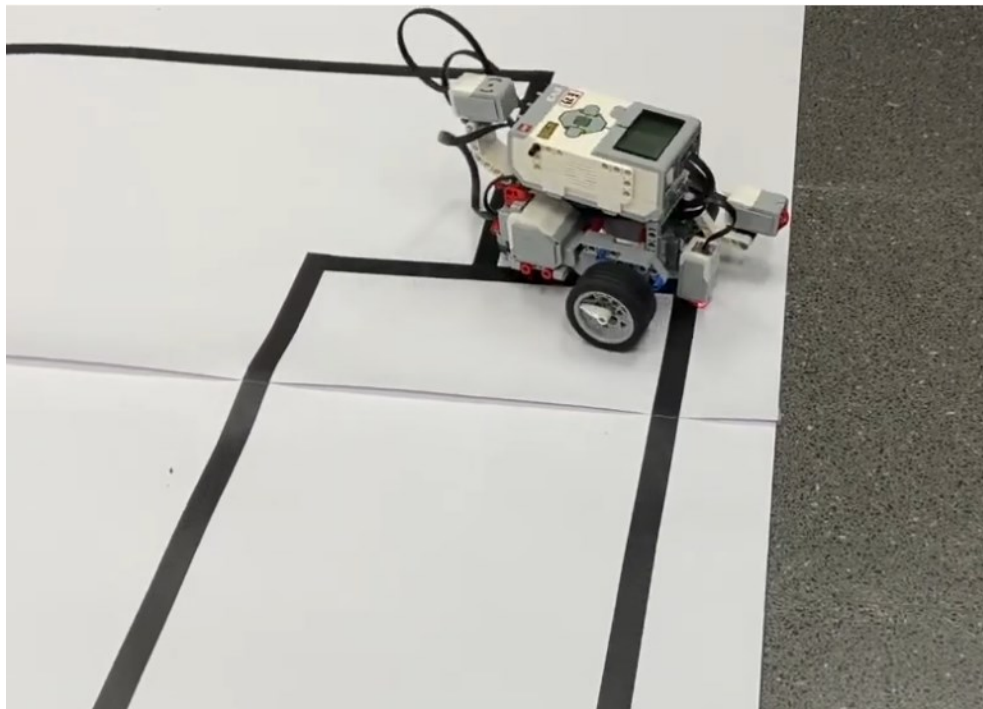
CheckRight/Left 的功能和 TurnRight/Left 相同，只是多了紀錄轉彎時感應到黑線次數的功能。

進入主函式之後，程式會不斷利用 getColorReflected(S3)讀取反射值分為兩個情況執行：

1. 反射值小於一域值，表示目前車子在黑線上，直走
2. 反射值大於一域值，表示目前車子偏離軌道，開始修正路線：
 1. 後退直到車子接收到黑線訊號
 2. 往右轉 30 度，紀錄 sensor 接收到幾次黑線訊號
 3. 往左轉 30 度回正
 4. 再往左轉 30 度，紀錄 sensor 接收到幾次黑線訊號
 5. 往右轉 30 度回正

最後將車子往記錄到比較多次黑線訊號的地方轉，便能順利回到軌道上

Result



循跡影片：[影片連結](#)

III. Conclusion

1. 林珮玉

這次實驗是很棒的體驗，沒想到在大學還可以接觸到小時候玩的 LEGO，特別懷念。這個實驗課程真的學習不少有趣的東西，包含行進方向控制、超音波偵測、顏色辨識等等，最後實戰自動循跡的機器人。透過一堂堂課的練習，確實提升我對於控制組的認識。

而在這次的 Final Project 中，原本以為跟某次課堂練習差不多，沒想到還有好多方面要考慮，包含偵測顏色臥數要開多少、角度設定、每個步伐以及方向判定等等，所以過程有修了幾次 code。不過我們還是很厲害，特別感謝我的組員們，有的在開會之前就帮大家買好材料，有的準備好程式於開會時測試和討論，節省時間，以助於我們的期末實驗早早完成。謝謝我的組員，也謝謝辛苦的助教！

2. 林哲緯

這堂課的 final project 是做循跡車。我們在前幾次 lego 的上課其實就有差不多做出一個版本了，那時候也算是很成功，但是這次要讓車子轉 90 度的彎，又讓問題變的更難了。一開始車子都會卡在第一個彎反覆判讀轉不過去，而後來靠著隊友以 trial and error 的方式，才得以讓車子順利轉彎。我覺得這次的 project 十分有趣，雖然用到不多控制工程的理論，但卻讓我們學會如何解決問題，收穫滿滿。謝謝助教的教導，與隊友整個學期的 carry！

3. 林玠志

這次的 final project 是做循跡車車，而我們在座的過程中遇到許多困難。一開始是由於程式的作法，導致可以右轉卻無法左轉，而經過修正後，卻又因為車子在移動過程中是無法同時使用 sensor，所以必須要移動一小段就停下來，導致行進不連貫，速度很慢。最後，仰賴隊友的幫助，有把這個問題稍微修正，這學期的實驗非常的具有啟發性，盡管以後並沒有要從事這個領域的研究或工作，但收穫還是很多。真的謝謝助教這學期的幫忙跟解惑。

4. 薛博文

因為大三的課大家都很忙，可以一起做車車的時間，大概就只有原本實驗課的時間，在第一次嘗試碰壁之後，時間好像就所剩不多 XD，所以在第二次實驗前就有在腦袋裡面模擬一下車子的運作大概會是怎麼跑，然後先想一下可能會出錯的地方有哪些，有想到一些點像是車子可能會衝出黑線太多，所以設計成讓他在出線之後，會先倒退回黑線的地方，再開始判斷左右。先想好會出錯的點之後，現場跑還是怪怪的，一直在想機器到底是哪裡判斷錯了，結果最後發現判斷都沒錯，錯的是程式碼有個地方漏掉 if 了，這個 bug 改掉之後就跑得蠻順利～很感謝孟宗告訴我陀螺儀的精確度

問題可能會造成誤差，也感謝其他隊友提供意見，一起找出程式碼的 bug。

5. 蔡孟宗

期末做的是循跡轉彎的車子，原本以為直接用之前樂高 lab2 的 code 進行修改很快就能順利完成這個實驗，想不到我還是太天真了，當初的 code 只適用於小幅度的轉彎，遇到直角灣基本上就是沒轍。因此第一次做這個實驗時我們只得一次次的嘗試又一次次的看著車子卡在直角處。但這些失敗並沒有白費，在第一次失敗後，我和組員們討論了並分析了各種可能發生失敗的原因，並且針對那些原因事先寫了應對的 code，因此第二次實驗開始我們做的事基本上就是把之前準備好的各種 code 刷一遍上去，無奈出師不利，很快的 code 都試完之後還是沒能順利通過地圖。正當我們一籌莫展時，組員提議說可以把 code 解釋給其他人聽，讓其他人幫忙想想這樣做是不是會有什麼盲點，結果博文解釋時發現他有個 if block 構想上應該要有但是程式中沒有，把他補齊之後車子就順利通過地圖了。

這學期的課程即將告一個段落了，很感謝有這個機會能和一群志同道合的夥伴們共事，我們團隊中每個人都很熱衷討論並樂意付出，很多實驗看似困難，也在一次次的交流溝通中找到了解決方法，在此感謝每個組員的付出，也感謝助教在課堂上提供的協助！