

創成プロジェクト発表

A1班

詳細は右記QRコードを参照



製品としての目標

軽量化

低コスト

安定動作性

必要最低限の装備

マイコン仕様で
ゴール可能

部品の再利用
→SDGs 目標達成

成功率90%以上



車体1(ファン駆動)の概要

- ・車体サイズ： $150 \times 120 \times ?(\text{mm})$
- ・重量： $?(\text{g})$
- ・駆動方法：ファン
- ・主要部品：タイヤ，モータ，サーボモータ，プラスチックの車体
【トリガ機構】
- ・金属ピースがスイッチの役割を果たす
【リリース機構】
- ・マイクロコンピュータ制御により、車体が 180° 回転
- ・ベアリングを取り付け、回転抵抗を減らす。



製作過程

① 車体の設計

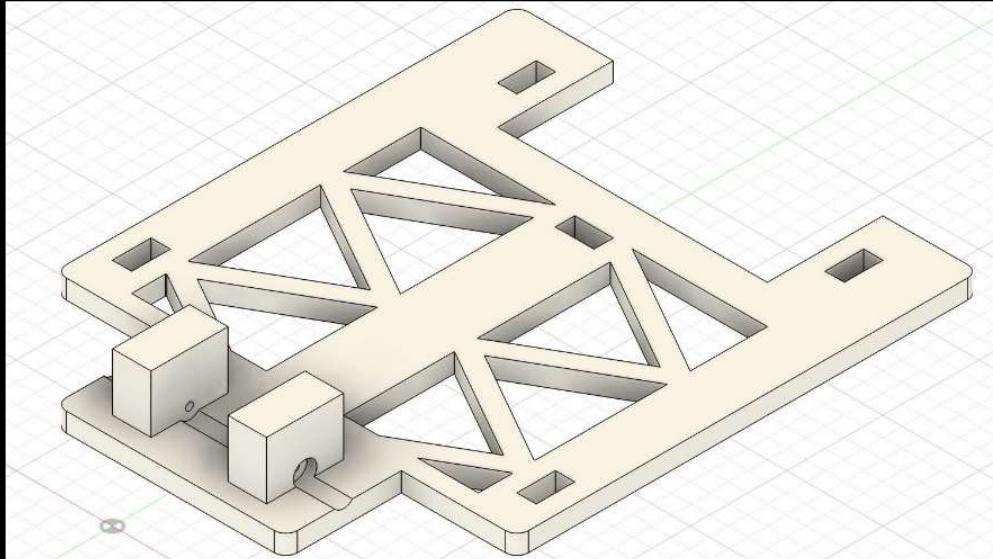


図1：台車部分

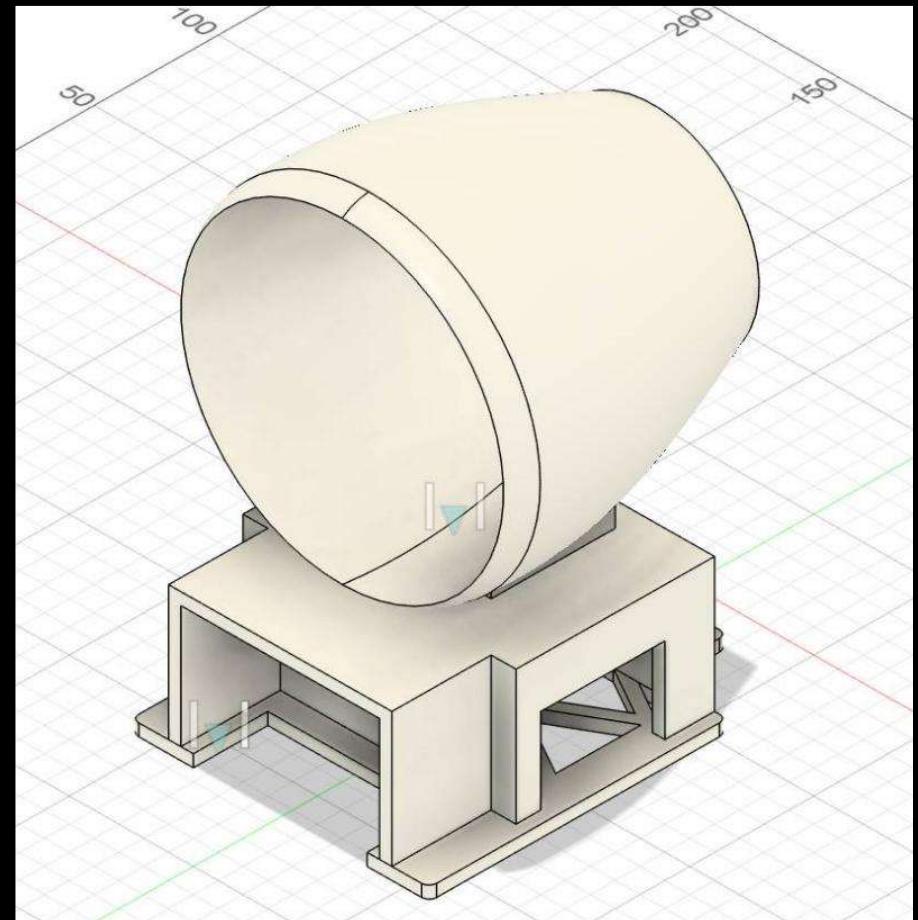


図2：全体図

製作過程

- ① 車体の設計
- ② 部品の強度計算

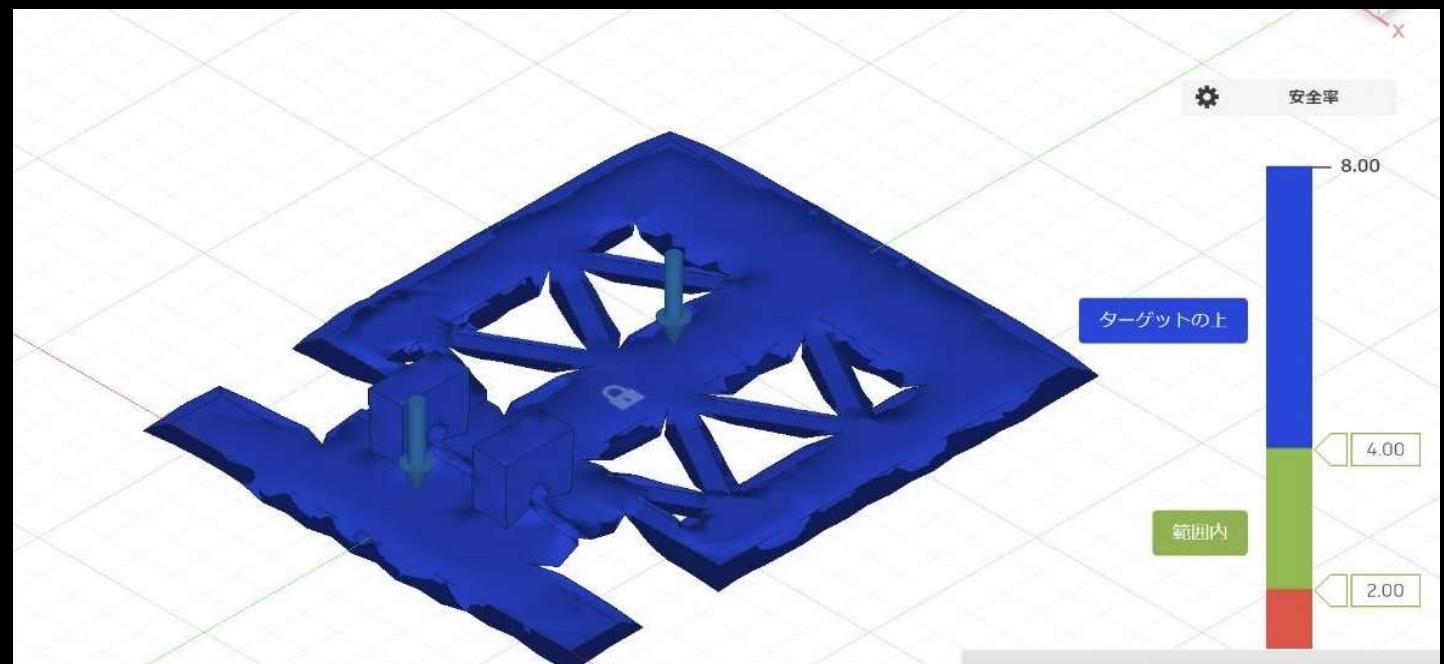


図3：台車部分の強度計算

製作過程

- ① 車体の設計
- ② 部品の強度計算
- ③ ファン部分の確認

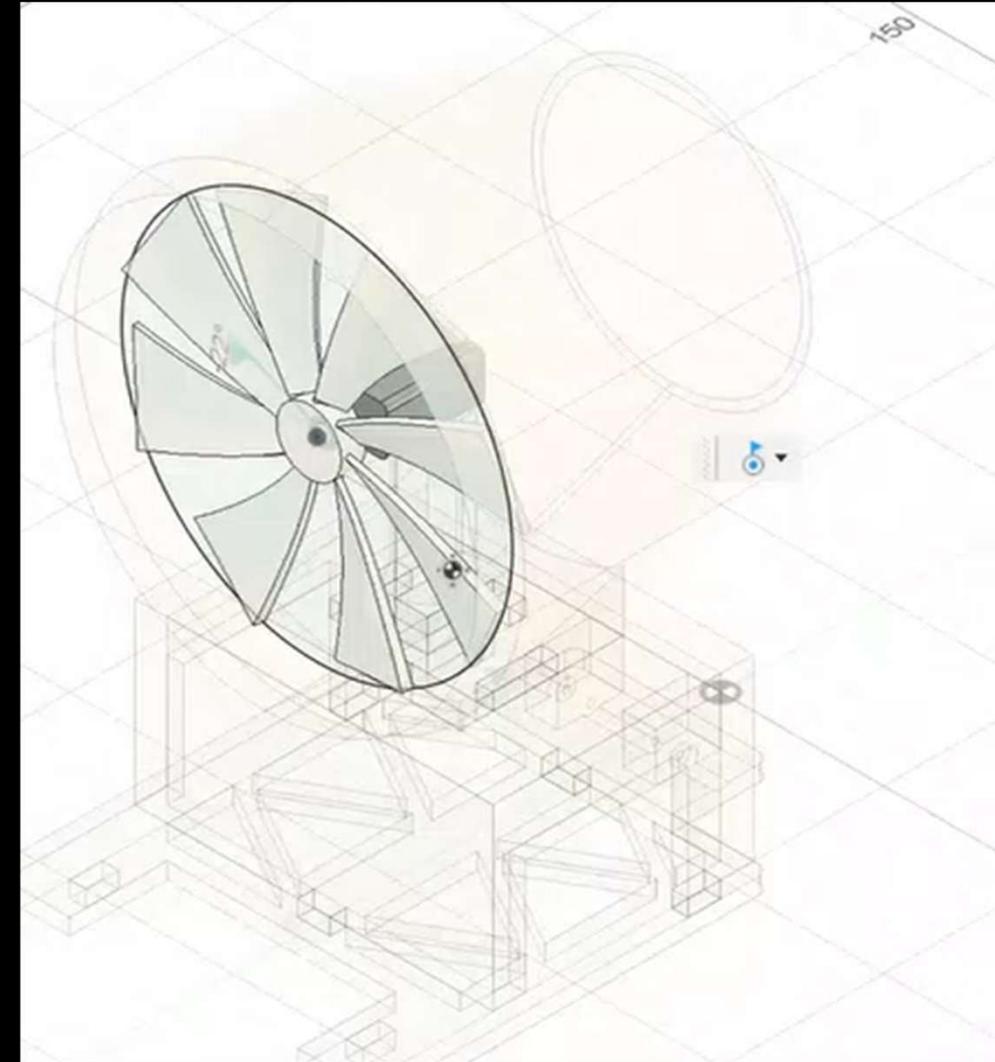


図4：ファン部分の確認

製作過程

車体の設計

部品の強度計算

ファン部分の確認

回路の設計

マイコンでタイヤを制御

問題発生

方針変更

ファン駆動では十分な推力が得ることが難しく、タイヤ駆動で走行する方法で方針変更する

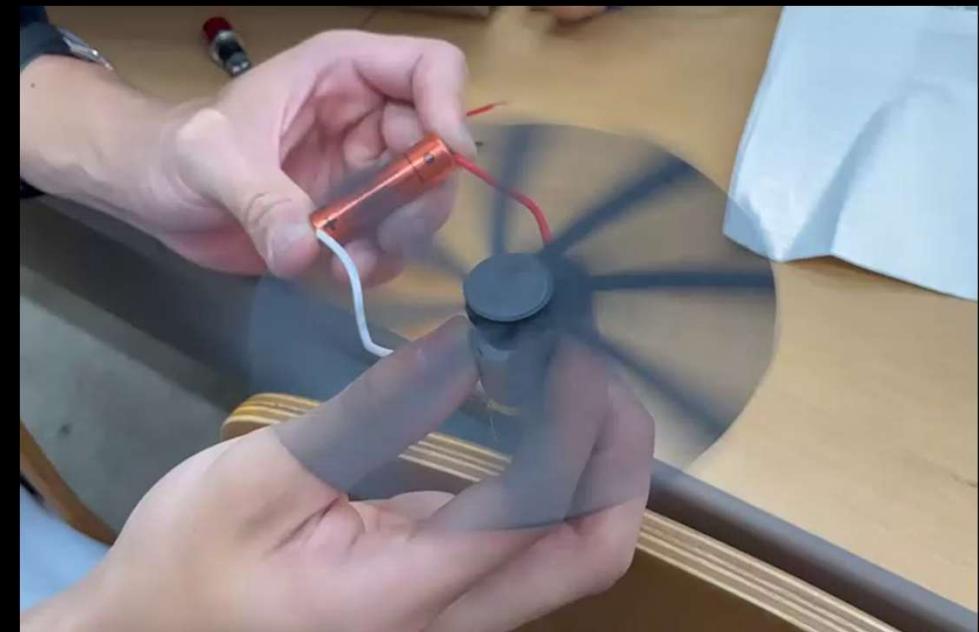


図5：方針変更の原因

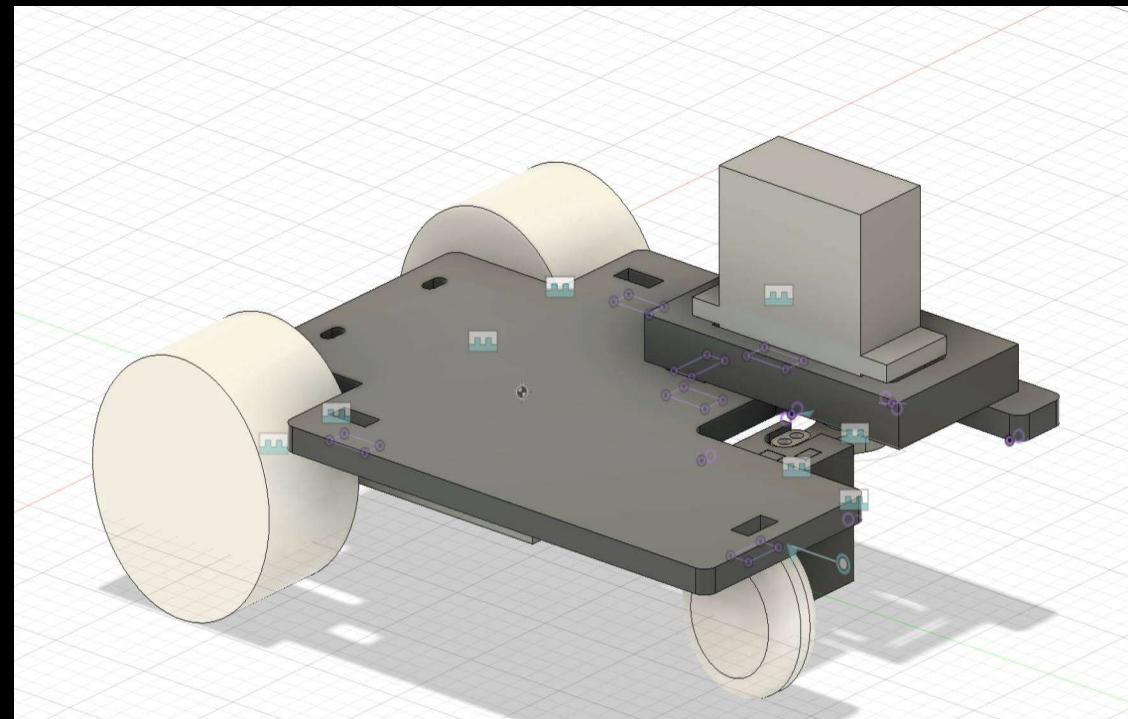


車体2(タイヤ駆動)の概要

- ・車体サイズ：150×120×高さ(mm)
- ・重量：(g)
- ・駆動方法：二輪駆動タイヤ
- ・主要部品：タイヤ、モータ
【トリガ機構】
- ・金属ピースを置くことでスイッチが作動し、折り返し地点で止まり、
金属ピースを置きなおすことで旋回する
【反転機構】
- ・マイクロコンピュータ(C++ / Arduino)制御により、サーボモータを駆動し車体を180°回転
- ・サーボモータを用いて旋回する

製作過程

① 車体の設計



製作過程

- ① 車体の設計
- ② 回路の設計

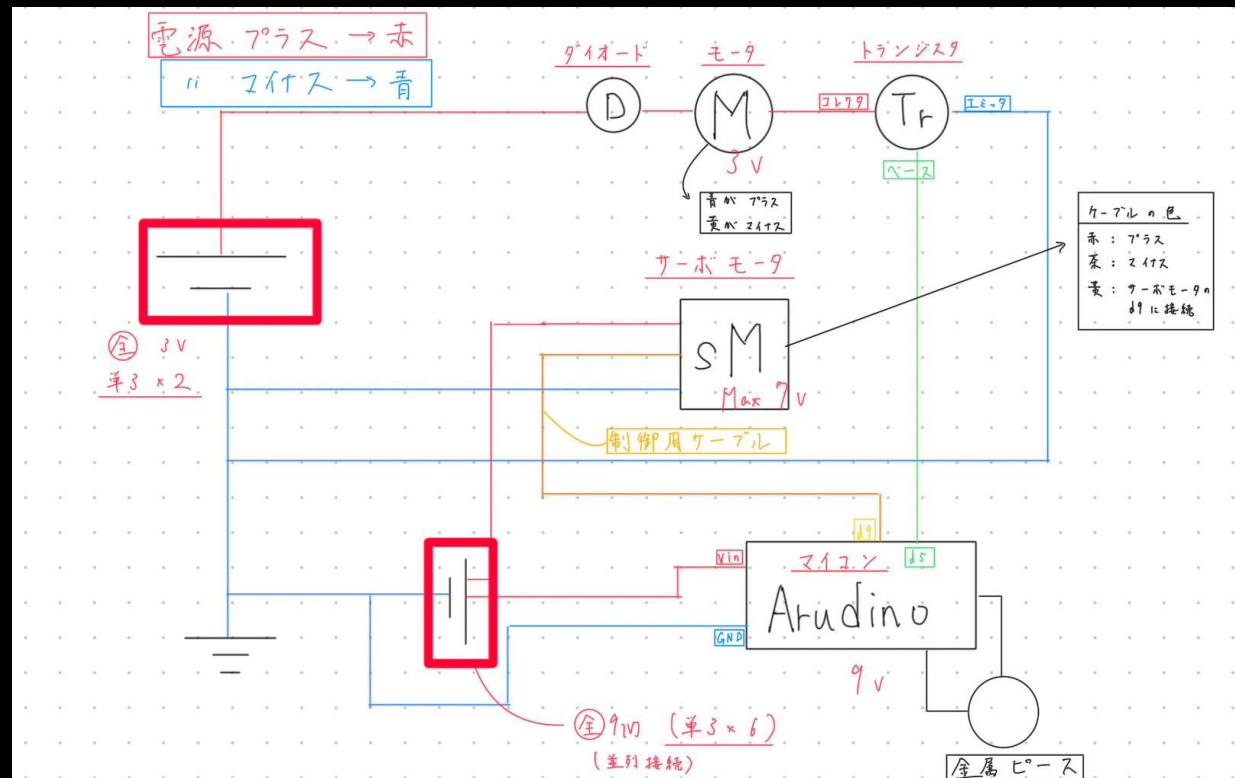


図 6：回路の設計

製作過程

- ① 車体の設計
- ② 回路の設計
- ③ マイコンでサーボモーターを制御

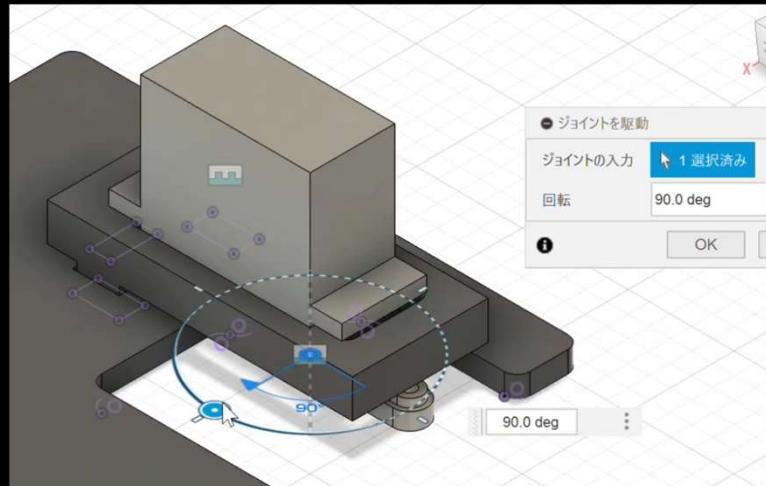


図7：マイコンでサーボモーターを制御

製作過程

- ① 車体の設計
- ② 回路の設計
- ③ マイコンでサーボモーターを制御
- ④ 回路の配線

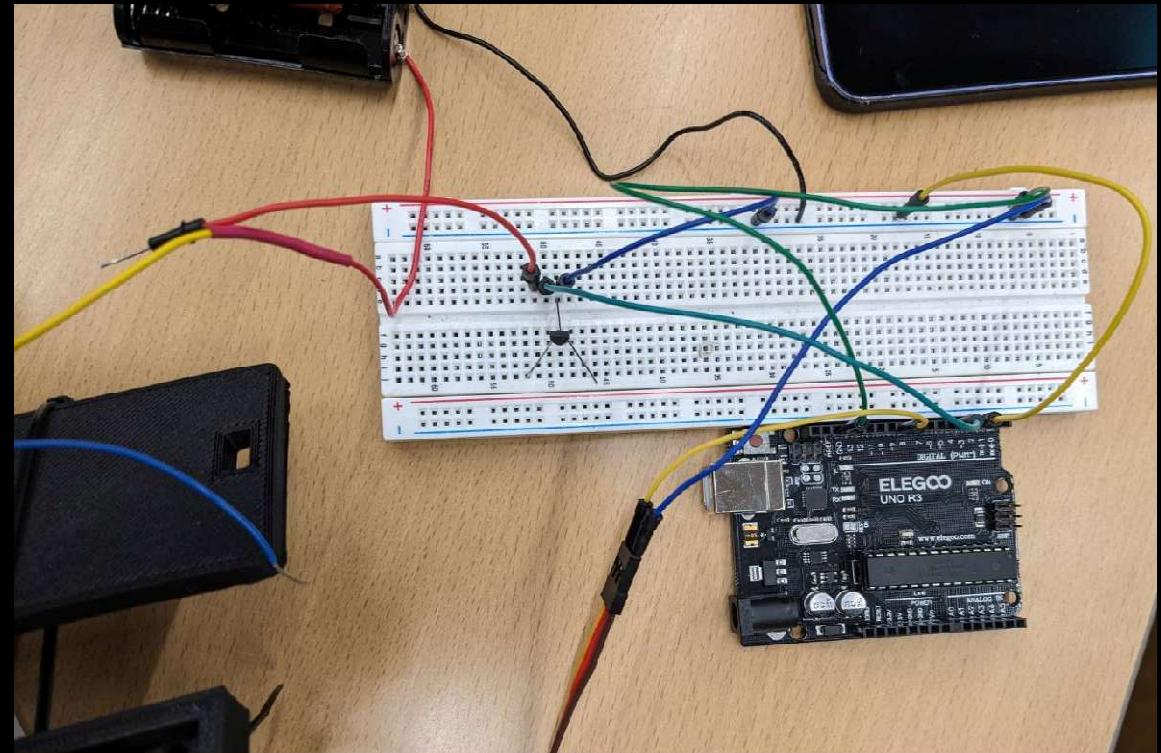


図 7：配線の設計

製作過程

- ① 車体の設計
- ② 回路の設計
- ③ マイコンでサーボモーターを制御
- ④ 回路の配線
- ⑤ プログラムの調整

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

// ① ピン設定
const int PIN_MOTOR_PWM = 5;
const int PIN_MOTOR_DIR = 7;
const int PIN_SERVO = 9;
const int PIN_SENSOR = 2;

// ② 動きの設定
const int SPEED = 255;
const int ANGLE_STRAIGHT = 86;
const int ANGLE_TURN = 45;

// ③ 時間の設定 (Uターンだけは角度制御のために時間を使用)
const int TIME_TURN = 5850;

Servo myServo;
int stage = -1;

void motorOn() { analogWrite(PIN_MOTOR_PWM, SPEED); digitalWrite(PIN_MOTOR_DIR, LOW); }
void motorOff() { analogWrite(PIN_MOTOR_PWM, 0); digitalWrite(PIN_MOTOR_DIR, LOW); }

void setup() {
    pinMode(PIN_MOTOR_PWM, OUTPUT);
    pinMode(PIN_MOTOR_DIR, OUTPUT);
    pinMode(PIN_SENSOR, INPUT_PULLUP);
    myServo.attach(PIN_SERVO);

    motorOff();
    myServo.write(ANGLE_STRAIGHT);
    delay(1000);
}

void loop() {
    // -----
    // ステージ -1 : 出発待ち (金属を載せたらスタート)
    // -----
    if (stage == -1) {
        if (digitalRead(PIN_SENSOR) == LOW) {
            delay(500); // 安定待ち
            stage = 0;
        }
    }
}
```

```
// -----
// ステージ 0 : 行き (金属を外すまで走り続ける)
// -----
else if (stage == 0) {
    myServo.write(ANGLE_STRAIGHT);
    motorOn();

    // ★修正ポイント：金属が載っている間(LOW)はずっとループして待つ
    // 金属を外した(HIGHになった)瞬間にループを抜ける
    while(digitalRead(PIN_SENSOR) == LOW) {
        delay(10);
    }

    motorOff(); // 金属が外れたので即停止
    delay(1000); // 念のための停止時間
    stage = 1;
}

// -----
// ステージ 1 : 帰り (再度金属を載せたらUターンして帰る)
// -----
else if (stage == 1) {
    // 金属が再度載るので待つ
    if (digitalRead(PIN_SENSOR) == LOW) {
        delay(500);
    }

    // 1. Uターン (ここだけは回転角度を決めるために時間が必要)
    myServo.write(ANGLE_TURN);
    motorOn();
    delay(TIME_TURN);

    // 2. 帰りの走行 (直進に戻し、金属を外すまで走り続ける)
    myServo.write(ANGLE_STRAIGHT);
    // motorOnは継続中

    while(digitalRead(PIN_SENSOR) == LOW) {
        delay(10); // 金属が載っている間は走り続ける
    }

    // 3. 停止
    motorOff();
    delay(1000);
    stage = -1; // 最初に戻る
}
```

製作過程

- ① 車体の設計
- ② 回路の設計
- ③ マイコンでサーボモーターを制御
- ④ 回路の配線
- ⑤ プログラムの調整
- ⑥ 試走

発生しうる主な問題点1

<機械系>

- 部品干渉, 摩擦, ガタ, 軸ずれ
- トルク不足, 慣性大, 重心バランス不良
- 材料たわみ, 固定緩み, 取付角度ずれ
- 構造精度不足による動作不安定

<電気・電子系>

- 電圧降下, 電源容量不足, ノイズ干渉
- 配線不良, 接触不良, GND不統一
- モータとサーボの電源干渉
- 電源投入時の突入電流でリセット

発生しうる主な問題点2

<制御・プログラム系>

- 動作タイミングのずれ（反転前進が競合）
- サーボ角度指定・初期化・方向設定ミス
- 時間制御のみで誤差蓄積（距離ずれ）
- フィードバック不足・制御周期不均一

<環境・運用系>

- 床面摩擦・段差・傾きによる軌道誤差
- 温度・電圧低下による性能変化
- 長時間運転での加熱・劣化
- 組立誤差・操作手順ミス・メンテ不足

発生した問題点と改良及び工夫1

- ・ ファン駆動では推力が不足した→タイヤ駆動にした
- ・ 乾電池を固定する方法で工夫が必要→車体自体を作りかえた
- ・ スイッチが電池1個ではサーボモータがうまく動かない→9V角電池を使用することで出力を向上する
- ・ 金属ピースを置いたことがトリガーになるようにしなければいけない→金属ピースで電通させることで作動させる
- ・ トルク不足で前進しない→配布モータのギア比を変え、トルクが強いものに組み替えた
- ・ プログラム上の問題→試走後に解決
- ・ 回路上の問題→arduinoの端子などが誤って電池などに触れないように管理する
- ・ サーボモータとマイコンで使用する電源を分ける。→arduinoで9V角電池と駆動用モータで乾電池をそれぞれ2本ずつ活用する
- ・ 回路がうまく動かない（焦げた）→arduinoの端子などが誤って電池などに触れないように管理する必要がある
- ・ トランジスタがうまく動かないときがある→単純に差し込む端子を間違えていたり、接触不良だったりした。テープで頑強に補強して解決
- ・ 配線に不調が起こりやすい→接触不良だった。テープで頑強に補強して解決
- ・ 配線が切れたりする→はんだ付けのときに固めすぎないようにある程度自由が利くようにつなぐことで寿命を延ばす
- ・ からくりを置く位置によって帰還する位置がわずかに異なる。→?
- ・ 直進せずに右方向にずれて進む。→直進のときにサーボモータをわずかに4°ほど左に切るようにする(直進だがプログラム上は常に左に微旋回する状態にした)

発生した問題点と改良及び工夫2

- ・配線が複雑で間違いやすい→マイコンの配線で色分けを行うことで再設計しやすくした
- ・電力の過不足問題→マイコンとサーボモータを同一の起電力で動かし、モータのみ別の起電力で動かすことにより解決した。
- ・コンテストの台が水平でなく中央が高くなっていてスタート位置によって帰還場所が変わる→スタート位置を決めて金属ピースを折り返しラインを超えた瞬間取り外すどうまくいく

目標の達成

- ・軽量化→達成
- ・低コスト→達成
- ・安定動作性→達成
- ・必要最低限の装備→達成
- ・マイコン仕様でゴール可能→達成
- ・部品の再利用→達成
- ・成功率90%以上→達成