

# 蓝牙遥控及牵引式控制智能旅行箱底座

## 研究背景与思维过程

### 研究背景

随着智能手机的发展，涉及人们生活方方面面的产品都逐渐走向智能化，包括旅行用智能产品。针对旅途中人们使用旅行箱的需要，设计出一款可通过蓝牙手机控制，或使用拉绳结构自动行进的旅行箱，能够有效解决旅途中拖拉旅行箱的不适感，为人们在路途上多添几分乐趣。

### 提出问题与研究解决方案

1. 如何让旅行箱自动行走并且容易控制？

使用麦克纳姆轮底盘实现旅行箱的全向移动，使用方便的同时为旅途增添乐趣

1. 如何实现拉绳结构控制？

通过两块3D打印的活页结构实现，可控制车子向左前方，右前方，前方的平移，

1. 可否具备其它功能拓展？

可把旅行箱当作是遥控车来玩；通过霍尔传感器实现定向定距移动。

### 设计思路

本次课题针对旅行箱移动的局限性，提出一种基于Arduino的可通过蓝牙控制的全向移动旅行箱。本作品通过Arduino程序设计，电机驱动，蓝牙模块，活页结构等进行装置的整体运作。而整体的功能可分为两个，一是蓝牙遥控的运动模块，二是拉绳结构的控制模块。可实现的功能包括遥控和手动拉绳控制。通过L298N电机板控制小车底盘移动。综合分析，本次课题的主要研究内容有以下几个方面：

- (1) 在设计时需要了解麦克纳姆轮实现全向移动的原理
- (2) 电路设计需要了解L289N的运作原理，配合Arduino Nano及扩展板做出合理的接线设计
- (3) 拉绳结构的考虑，需要对力的使用进行数次测试分析，并且要通过Fusion360等工具设计出合理的传动装置

### 原理分析

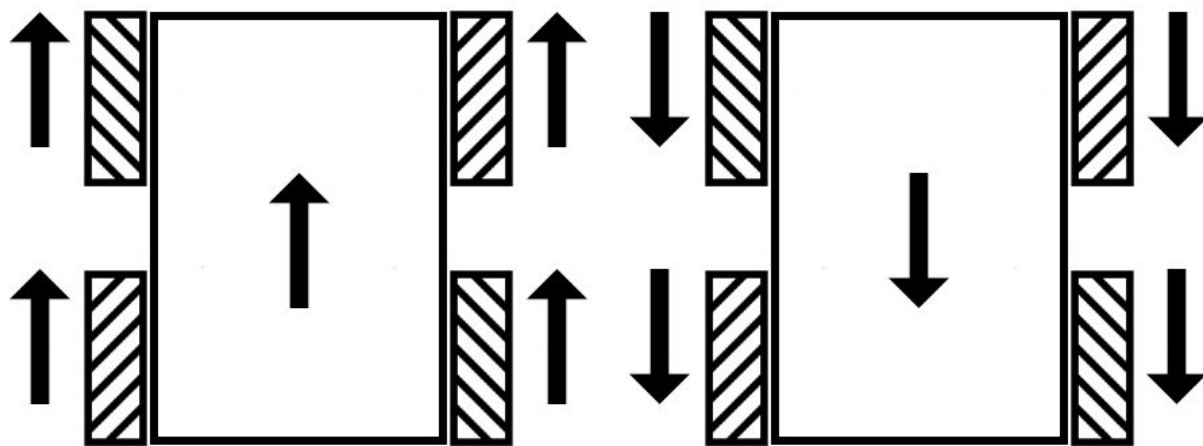
#### 硬件原理介绍

1. Arduino Nano
2. Nano扩展板
3. L298N电机控制板
4. 麦克纳姆轮底盘带550电机

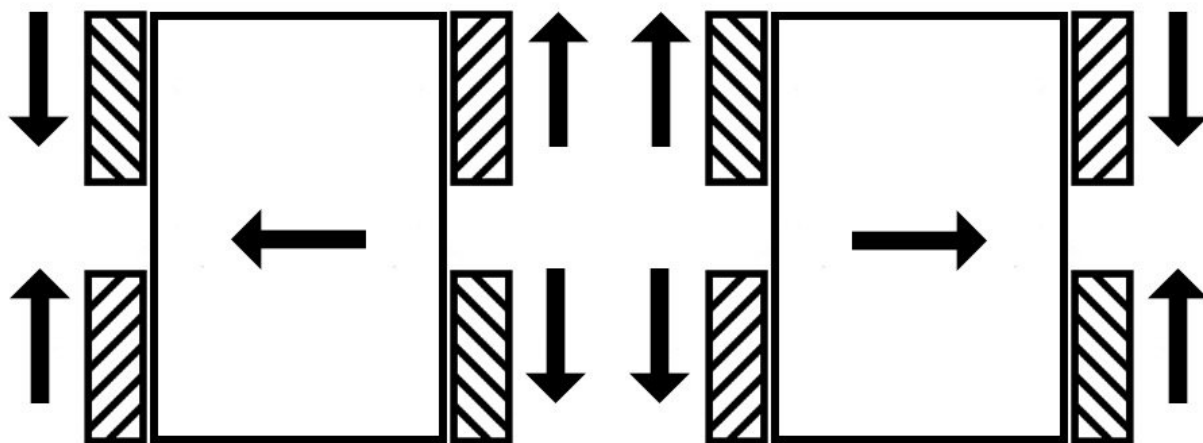


麦克纳姆轮：轮子倾斜的方向跟行进方向是很有关系的，一般轮子的安装方式就是俯视车子会呈现"X"状，轮子转动方向和车子行进方向之间的关系如下图所示：

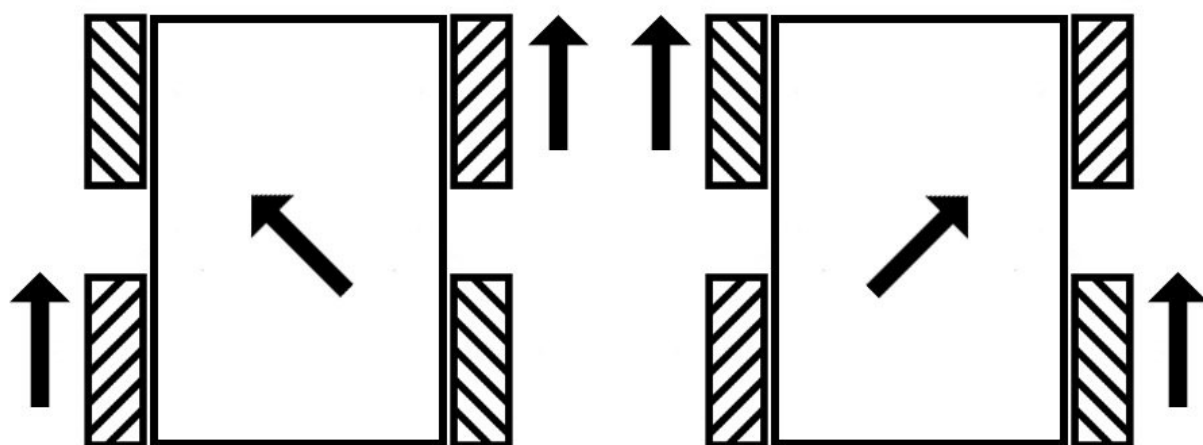
前后移动



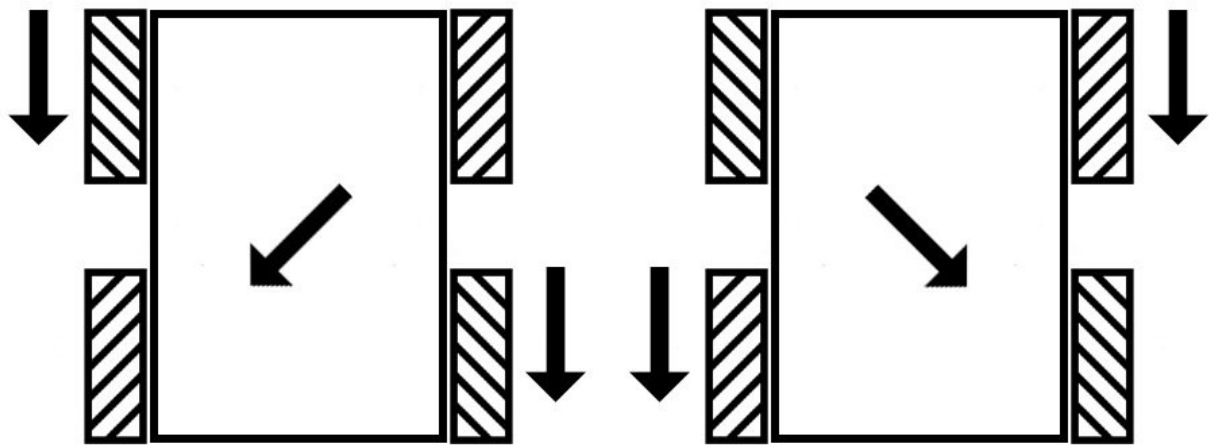
左转右转



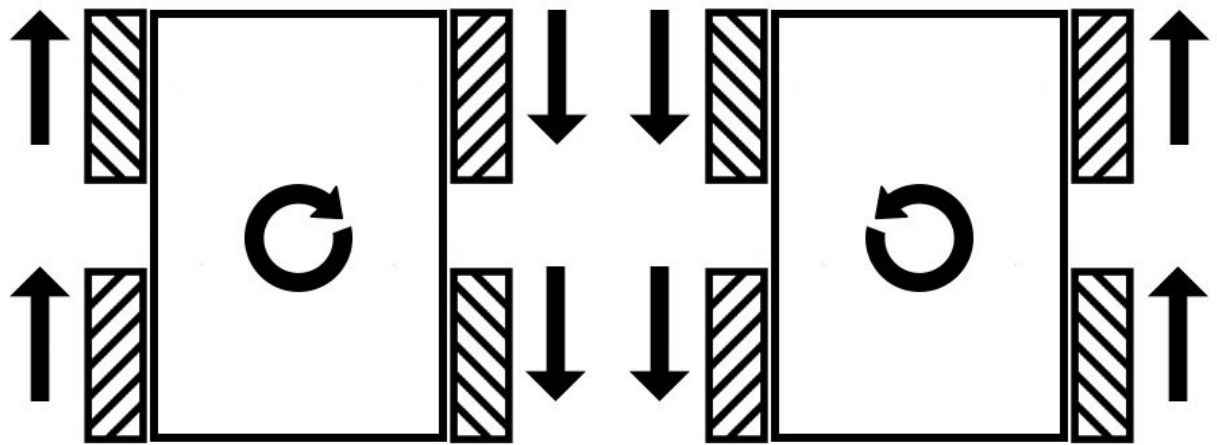
左前右前



左后右后



顺时针转和逆时针转

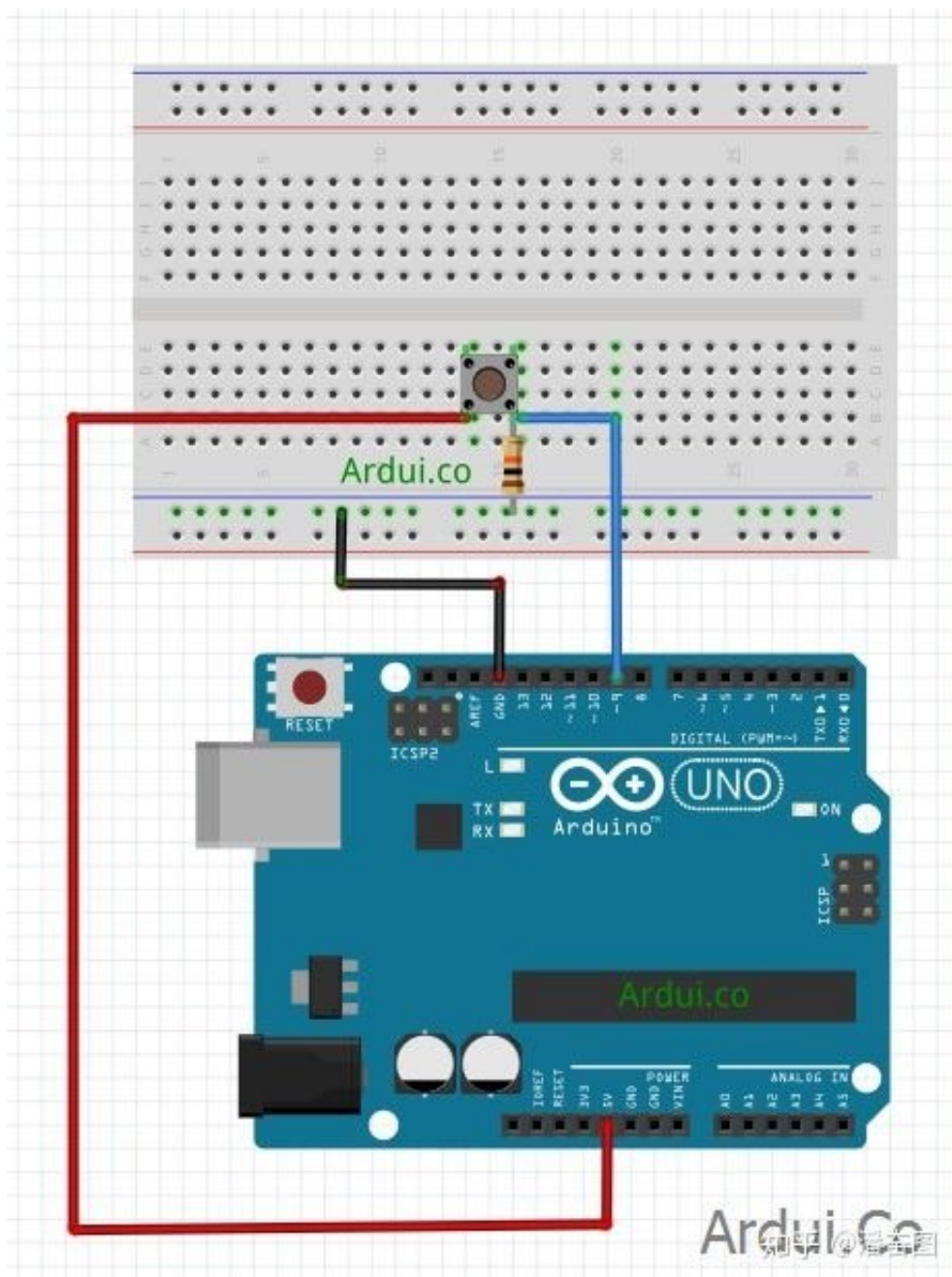


550电机



1. HC-05蓝牙模块
2. 按钮连接件设计：

原理如下图：



图中的按钮测试用于测试Uno的板载LED小灯，我们在使用按钮控制小车时，可以让蓝色的线成为信号。

### 接线表

Nano扩展板	L298N	HC-05	连接内容	备注
2	IN1		右前马达	
4	IN2		右前马达	
3	EN1		右前马达PMW	

7	IN3		左前马达	
6	IN4		左前马达	
5	EN2		左前马达PWM	
9	IN5		右后马达	
8	IN6		右后马达	
10	EN3		右后马达PWM	
12	IN7		左后马达	
13	IN8		左后马达	
11	EN4		左后马达PWM	
	VCC		12V电源正极	
	GND		12V电源负极	
5V	5V			
GND	GND			
	OUT1		右前马达正极	
	OUT2		右前马达负极	
	OUT3		左前马达正极	
	OUT4		左前马达负极	
	OUT5		右后马达正极	
	OUT6		右后马达负极	
	OUT7		左后马达正极	
	OUT8		左后马达负极	
RX		TX		
TX		RX		

5V		5V		
GND		GND		
A4			拉绳结构按钮1	
A5			拉绳结构按钮2	

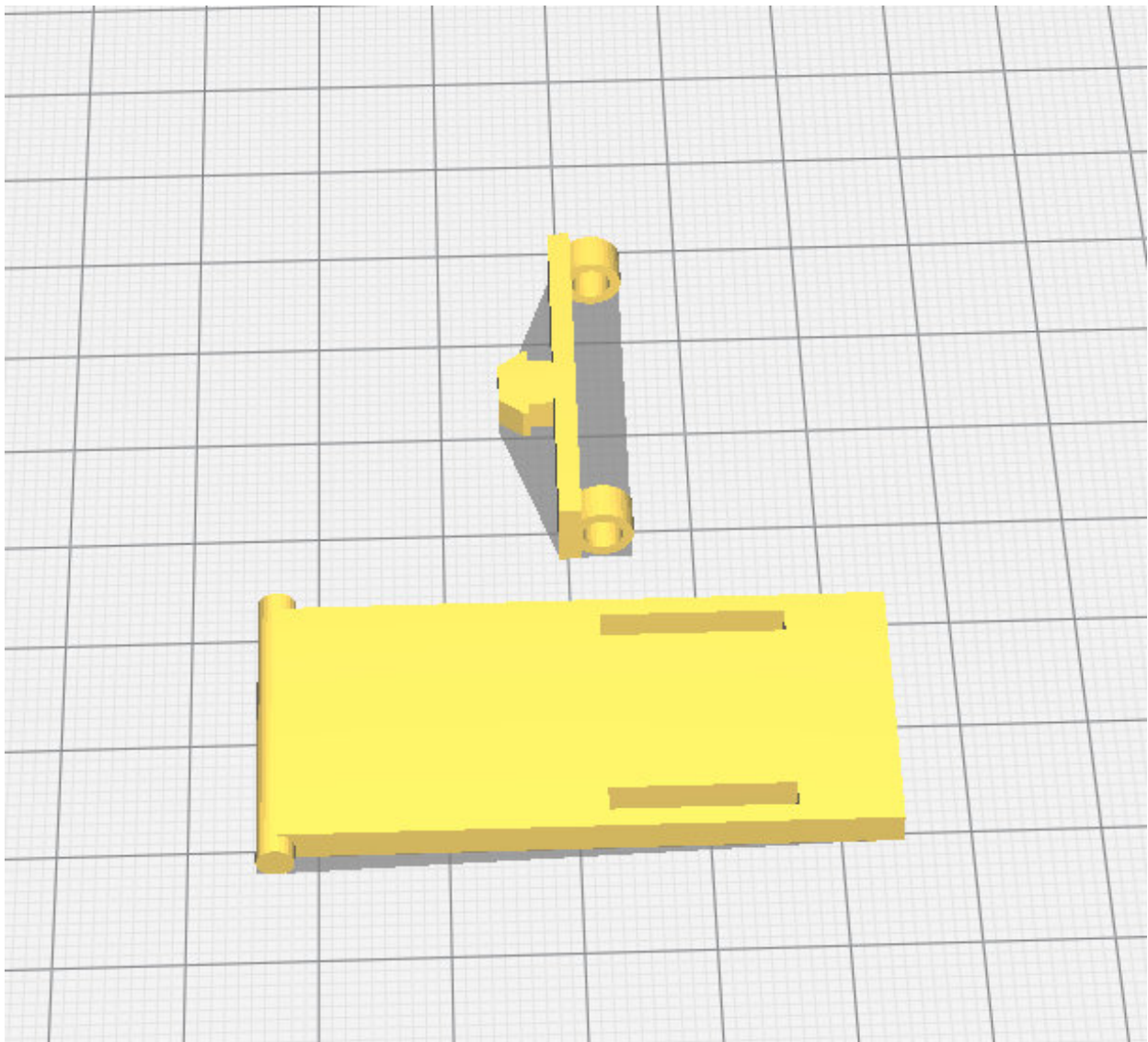
## 制作过程

## 材料

材料	数量	备注	价格
麦克纳姆轮底盘带电机	1		520
Arduino Nano	1	带线	16
Nano扩展板	1		5
L298N	1		29
蓝牙HC-05	1		18
12V电池	1		30
杜邦线	若干	母对母，公对母	2
		合计	620

## 打印外观

图为牵引结构的外观：



## 安装

- 底盘安装需要使用六角铁，钢板没有富余，故安装时要耐心，不用一下子把螺丝上紧而需要慢慢调整
- 马达和轮子的安装要注意使用联轴器，并且四个轮子的方向需要注意，从上方俯视，麦轮应呈"X"状

## 调试

- 虽底盘有涂层可以绝缘，但调试底部时最好使用海绵垫垫住元器件以防万一
- 先进行单个马达的调试，后再调通四个马达的运转方式，结合代码进行理解

## 调主程序



- 单个马达测试代码

```
/*
 * Code developped by @PastorEdu
 */

//单个马达测试代码 For Nano
//Define the Pins

#define pinIN1 2      //nano的D2 接上 L298N的IN1
#define pinIN2 4      //nano的D4 接上 L298N的IN2
#define pinPWM 3      //nano的D3 接上 L298N的EN1
#define pinA A0       //编码器的A相 接上 nano的A0
#define pinB A1       //编码器的B相 接上 nano的A1

const int d_time=100;
int speed = 255;  //统一设置速度
int i = 0;
int valA = 0;     //以下参数用于霍尔传感器测速代码
int valB = 0;
unsigned long duration = 0;
unsigned long times;
unsigned long newtime;

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);

    //Set the PIN Modes
    pinMode(pinIN1, OUTPUT);
    pinMode(pinIN2, OUTPUT);
    pinMode(pinPWM, OUTPUT);
    pinMode(pinA, INPUT);
    pinMode(pinB, INPUT);
}

void loop() {
```

```

// put your main code here, to run repeatedly:

Serial.println("前");

digitalWrite(pinIN1, 1); //设置行进方向
digitalWrite(pinIN2, 0);
analogWrite(pinPWM, speed); //启动马达
//SpeedCheck(); //调用测速函数，启用后可在串口监视器看到速度（霍尔传感器有接线的情况下）
delay(2000);

Serial.println("结束");
analogWrite(pinPWM, 0);
delay(1000);

Serial.println("后");
digitalWrite(pinIN1, 0); //设置行进方向
digitalWrite(pinIN2, 1);
analogWrite(pinPWM, speed); //启动马达
delay(2000);

Serial.println("结束");
analogWrite(pinPWM, 0);
delay(1000);

}

}

```

- 蓝牙控制测试代码

```

/*
 * A9G returning lat & lng
 * Code developed by @PastorEdu
 *
 * 功能：通过手机蓝牙BlueSpp工具控制麦轮小车移动

```

```

*
* 使用说明:
* 1. 请参考代码的pin定义来接线
* 2. 部分行进方向代码需学生自行填充
*/

//Define the Pins
#define pinAIN1 4 //右前马达
#define pinAIN2 2 //右前马达
#define pinPWMA 3 //右前马达PWM
#define pinAIN3 7 //左前马达
#define pinAIN4 6 //左前马达
#define pinPWMB 5 //左前马达PWM
#define pinAIN5 10 //右后马达
#define pinAIN6 8 //右后马达
#define pinPWMC 9 //右后马达PWM
#define pinAIN7 12 //左后马达
#define pinAIN8 13 //左后马达
#define pinPWMD 11 //左后马达PWM

#define speed 255 //定义统一速度

#define pinSwitchLeft A4 //左前开关信号
#define pinSwitchRight A5 //右前开关信号

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
    //Set the PIN Modes
    pinMode(pinAIN1, OUTPUT);
    pinMode(pinAIN2, OUTPUT);
    pinMode(pinPWMA, OUTPUT);
    pinMode(pinAIN3, OUTPUT);
    pinMode(pinAIN4, OUTPUT);
    pinMode(pinPWMB, OUTPUT);
    pinMode(pinAIN5, OUTPUT);
    pinMode(pinAIN6, OUTPUT);

```

```

    pinMode(pinPMMC, OUTPUT);

    pinMode(pinAIN7, OUTPUT);

    pinMode(pinAIN8, OUTPUT);

    pinMode(pinPWMD, OUTPUT);

}

void loop() {

    //蓝牙信号处理, 通过BlueSPP控制车子行进

    String c = readTtl();

    if (c == "FL" )    move_forward_left();
    if (c == "FF" )    move_forward();
    if (c == "FR" )    move_forward_right();
    if (c == "LL" )    move_left();
    if (c == "STOP" ) move_stop();
    if (c == "RR" )    move_right();
    if (c == "BL" )    move_backward_left();
    if (c == "BB" )    move_backward();
    if (c == "BR" )    move_backward_right();

    int switch_left  = digitalRead(pinSwitchLeft);
    int switch_right = digitalRead(pinSwitchRight);

}

String readTtl()
{
    String comdata = "";
    while (Serial.available())
    {
        comdata += char(Serial.read());
        delay(2);
    }
    return comdata;
}

```

```

//小车行进指令封装函数

void move_forward_left(){

    Serial.println("moving forward left");

    digitalWrite(pinAIN1, 1);

    digitalWrite(pinAIN2, 0);

    digitalWrite(pinAIN3, 0);

    digitalWrite(pinAIN4, 0);

    digitalWrite(pinAIN5, 0);

    digitalWrite(pinAIN6, 0);

    digitalWrite(pinAIN7, 0);

    digitalWrite(pinAIN8, 1);

    analogWrite(pinPWMB, speed);

    analogWrite(pinPWMC, speed);

}

void move_forward(){

    Serial.println("moving forward");

    digitalWrite(pinAIN1, 1);

    digitalWrite(pinAIN2, 0);

    digitalWrite(pinAIN3, 0);

    digitalWrite(pinAIN4, 1);

    digitalWrite(pinAIN5, 1);

    digitalWrite(pinAIN6, 0);

    digitalWrite(pinAIN7, 0);

    digitalWrite(pinAIN8, 1);

    analogWrite(pinPWMA, speed);

    analogWrite(pinPWMB, speed);

    analogWrite(pinPWMC, speed);

    analogWrite(pinPWMD, speed);

}

void move_forward_right(){

    Serial.println("moving forward right");

    digitalWrite(pinAIN1, 0);

    digitalWrite(pinAIN2, 0);

    digitalWrite(pinAIN3, 0);

    digitalWrite(pinAIN4, 1);

```

```

    digitalWrite(pinAIN5, 1);
    digitalWrite(pinAIN6, 0);
    digitalWrite(pinAIN7, 0);
    digitalWrite(pinAIN8, 0);
    analogWrite(pinPWMB, speed);
    analogWrite(pinPWMC, speed);
}

void move_left(){
    //todo 补充向左移动的代码

    Serial.println("moving left");
}

void move_right(){
    //todo 补充向右移动的代码

    Serial.println("moving right");
}

void move_backward_left(){
    //todo 补充向右后移动的代码

    Serial.println("moving backward left");
}

void move_backward(){
    //todo 补充向后移动的代码

    Serial.println("moving backward");
}

void move_backward_right(){
    //todo 补充向右后移动的代码

    Serial.println("moving backward");
}

void move_clockwise(){
    //todo 补充顺时针转动的代码
}

```

```
void move_ct_clockwise(){  
    //todo 补充逆时针转动的代码  
}  
  
void move_stop(){  
    //让车子停止  
    Serial.println("stopping");  
    analogWrite(pinPWMA, 0);  
    analogWrite(pinPWMB, 0);  
    analogWrite(pinPWMC, 0);  
    analogWrite(pinPWMD, 0);  
    digitalWrite(pinAIN1, 0);  
    digitalWrite(pinAIN2, 0);  
    digitalWrite(pinAIN3, 0);  
    digitalWrite(pinAIN4, 0);  
    digitalWrite(pinAIN5, 0);  
    digitalWrite(pinAIN6, 0);  
    digitalWrite(pinAIN7, 0);  
    digitalWrite(pinAIN8, 0);  
}
```

## 模型与实物图





## 项目特色

### 新颖性

本设计使用麦克拉姆轮实现全向移动，适合旅行箱移动，同时兼具旅途添趣效果，在智能出行领域有一定市场和发展空间

### 先进性



本作品在现有产品进行功能添加和优化改良结果，符合人们对旅行箱的使用需求和市场需要，具有创新性和先进性。

## 实用性

拉绳结构控制既可实现非常省力的牵引，也可让使用者放心使用。在国内的使用环境下，全自动跟随的旅行箱反而不符合用户场景，而有一根拉绳能让使用者更加放心。

## 作品展望

当前作品仍处于样品模型阶段，但已充分体现项目可行性和巨大的发展潜力。若是能借助更专业的手段将作品进一步与旅行箱融合，相信在不久的将来便能在市面上进行广泛的推广。