幻影箱

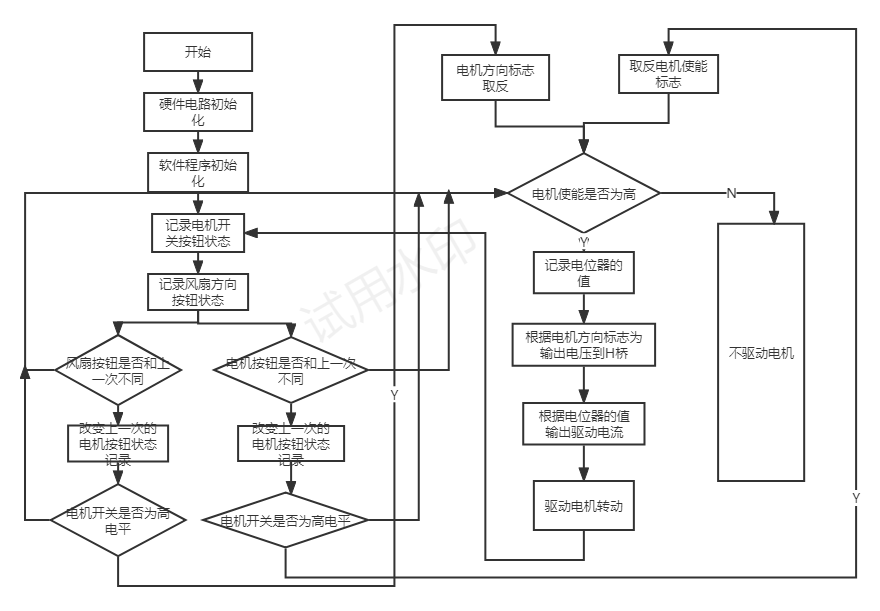
1. 实验原理

H桥是一种常用的电子器件，用于控制直流电机的转向和速度。它由四个开关（MOSFET或BJT）和一个直流电源组成，其中两个开关位于上部，两个开关位于下部，形成一个H形结构，因此称为H桥。

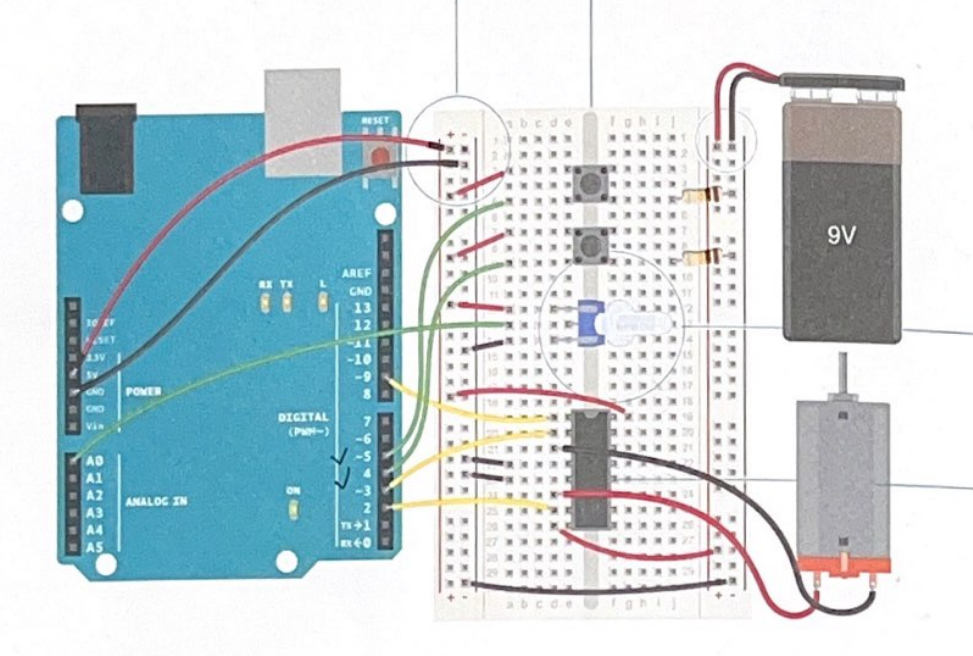
H桥有两个输入端口，一个用于控制电机的正向运转，另一个用于控制电机的反向运转。当正向输入端口接收到高电平信号时，上部两个开关S1和S4闭合，下部两个开关S2和S3断开，电机就会正向运转；而当反向输入端口接收到高电平信号时，上部两个开关S1和S4断开，下部两个开关S2和S3闭合，电机就会反向运转。如果输入端口的信号是低电平，则H桥中的所有开关都断开，电机停止运转。

H桥不仅可以控制电机的运转方向，还可以控制电机的速度。当需要控制电机的速度时，可以通过调节上部和下部开关的占空比来改变电机的电压和电流。具体来说，当上部开关S1和S4闭合时，下部开关S2和S3断开，电机会受到电源的正向电压，电机转速较快；当上部开关S1和S4断开时，下部开关S2和S3闭合，电机会受到电源的反向电压，电机转速较慢。通过调节上部和下部开关的占空比，可以改变电机受到的电压和电流，从而控制电机的速度。

1. 程序框图
   * 1. 软件框图



* + 1. 硬件框图



1. 程序代码

const int controlPin1 = 2; //设置控制引脚和使能引脚的数字引脚编号

const int controlPin2 = 3;

const int enablePin = 9;

//设置方向开关引脚、开关状态引脚和电位器引脚的数字引脚编号

const int directionSwitchPin = 4;

const int onOffSwitchStateSwitchPin = 5;

const int potPin = A0;

//定义变量来存储开关和方向开关的状态、电机状态、电机速度和电机方向

int onOffSwitchState = 0;

int previousOnOffSwitchState = 0;

int directionSwitchState = 0;

int previousDirectionSwitchState = 0;

int motoeEnabled = 0;

int motorSpeed = 0;

int motorDirection = 1;

void setup() {

// 将方向开关引脚、开关状态引脚和控制引脚的模式设置为输入或输出

pinMode(directionSwitchPin, INPUT);

pinMode(onOffSwitchStateSwitchPin, INPUT);

pinMode(controlPin1, OUTPUT);

pinMode(controlPin2, OUTPUT);

pinMode(enablePin, OUTPUT);

// 将使能引脚设置为低电平

digitalWrite(enablePin, LOW);

}

void loop() {

// 读取开关状态引脚的状态并存储在变量中

onOffSwitchState = digitalRead(onOffSwitchStateSwitchPin);

// 稍作延迟

delay(1);

// 读取方向开关引脚的状态并存储在变量中

directionSwitchState = digitalRead(directionSwitchPin);

// 读取电位器引脚的模拟值并将其除以4，存储在变量中作为电机速度

motorSpeed = analogRead(potPin) / 4;

// 如果开关状态发生变化

if (onOffSwitchState != previousOnOffSwitchState) {

// 如果开关状态为高电平

if (onOffSwitchState == HIGH) {

// 切换电机状态

motoeEnabled = !motoeEnabled;

}

// 将开关状态存储在变量中

previousOnOffSwitchState = onOffSwitchState;

}

// 如果方向开关状态发生变化

if (directionSwitchState != previousDirectionSwitchState) {

// 如果方向开关状态为高电平

if (directionSwitchState == HIGH) {

// 切换电机方向

motorDirection = !motorDirection;

}

// 将方向开关状态存储在变量中

previousDirectionSwitchState = directionSwitchState;

}

// 如果电机已启用

if (motoeEnabled == 1) {

// 将电机速度作为PWM信号写入使能引脚

analogWrite(enablePin, motorSpeed);

// 根据电机方向设置控制引脚的状态

if (motorDirection == 1) {

digitalWrite(controlPin1, HIGH);

digitalWrite(controlPin2, LOW);

} else {

digitalWrite(controlPin1, LOW);

digitalWrite(controlPin2, HIGH);

}

} else {

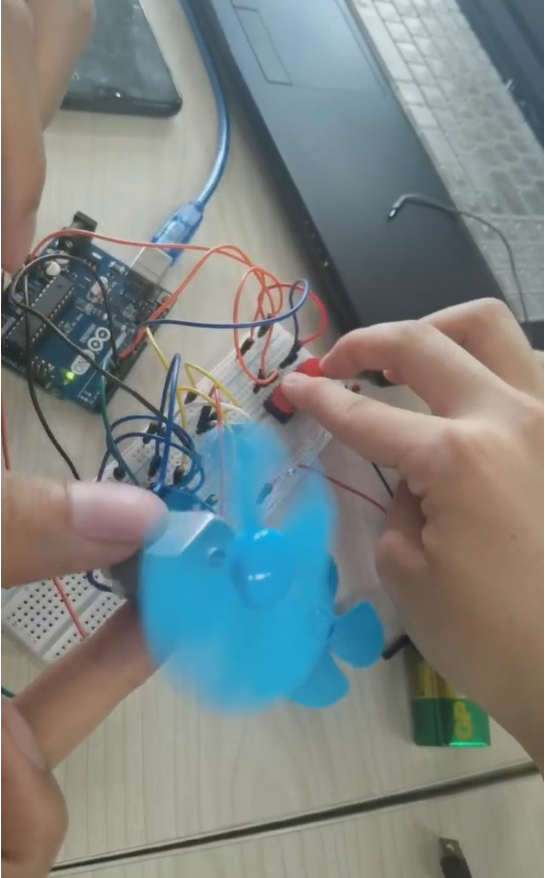
// 如果电机未启用，则将使能引脚设置为低电平

analogWrite(enablePin, 0);

}

}

1. 实验结果



1. 实验总结
2. 了解到了H桥的作用，其可通过引脚电压改变去放大并且改变输出的电压方向。
3. 此实验流程较为复杂，能够独立设计出实验代码并应用。
4. 在进行具有优先级并且对之前状态有关联的程序设计时，最好画好流程图进行代码编写，方便debug程序。