Lab Report: Project 10 – Zoetrope

Lucas Porter

Spring 2025 TESY 3300

Trevor P. Robinson, PhD

Lab 10 – Zoetrope

03/09/2025

Abstract

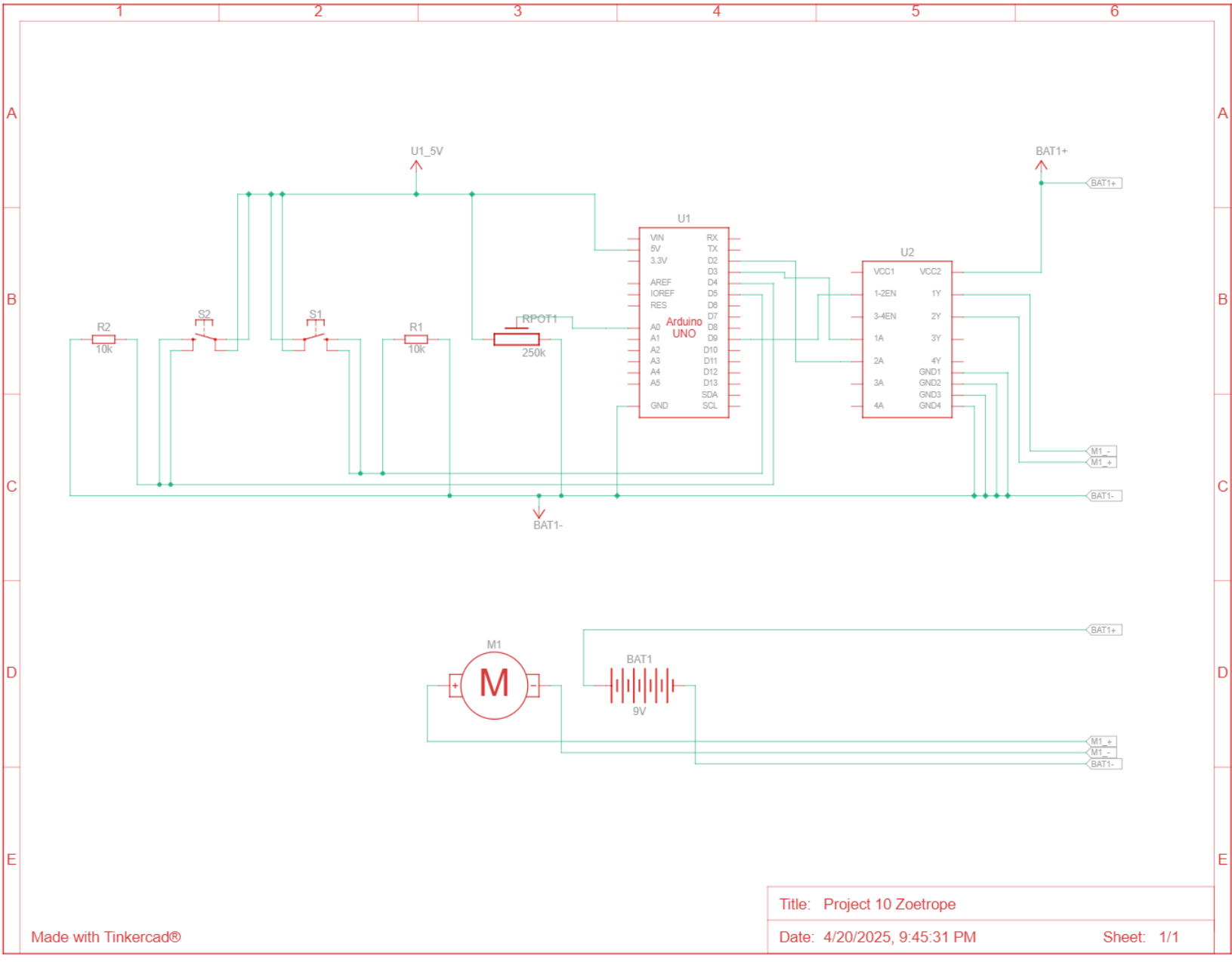
The purpose of this lab was to build a motorized zoetrope using an Arduino Uno, a DC motor, and an H-bridge IC. The project demonstrated how to use an H-bridge to control motor direction and speed electronically. The user was able to start, stop, and reverse the motor using pushbuttons. This lab introduced motor control circuits, external power considerations, and reinforced the importance of matching motor performance with the intended mechanical setup.

Materials

- Arduino Uno Board
- Breadboard
- DC Motor
- L293D H-Bridge IC
- 9V Battery
- Battery Clip
- 2 Pushbuttons
- 10kΩ Resistors
- Jumper Wires
- USB Cable
- Paper Disc (for zoetrope animation)
- Computer with Arduino IDE

Procedure

Circuit Diagram



Steps

1. Connected the Arduino's 5V and GND pins to the power and ground rails of the breadboard.
2. Inserted the L293D H-Bridge IC into the breadboard.
3. Connected the DC motor terminals to the motor output pins of the H-Bridge.
4. Connected a 9V battery to the motor power input of the H-Bridge.
5. Connected Arduino digital pins to the H-Bridge control pins to control motor direction and enable the motor.
6. Connected pushbuttons for start/stop and reverse functions with pull-down resistors.
7. Uploaded the Arduino sketch and tested the circuit.

Code

```
const int controlPin1 = 2;
const int controlPin2 = 3;
const int enablePin = 9;
const int directionSwitchPin = 4;
const int onOffSwitchStateSwitchPin = 5;
const int potPin = A0;

int onOffSwitchState = 0;
int previousOnOffSwitchState = 0;
int directionSwitchState = 0;
int previousDirectionSwitchState = 0;

int motorEnabled = 0;
int motorSpeed = 0;
int motorDirection = 1;
void setup() {

  pinMode(directionSwitchPin, INPUT);
  pinMode(onOffSwitchStateSwitchPin, INPUT);
  pinMode(controlPin1, OUTPUT);
  pinMode(controlPin2, OUTPUT);
  pinMode(enablePin, OUTPUT);

  digitalWrite(enablePin, LOW);
}

void loop() {

  onOffSwitchState = digitalRead(onOffSwitchStateSwitchPin);
  delay(1);

  directionSwitchState = digitalRead(directionSwitchPin);

  motorSpeed = analogRead(potPin) / 4;

  if (onOffSwitchState != previousOnOffSwitchState) {

    if (onOffSwitchState == HIGH) {
      motorEnabled = !motorEnabled;
    }
  }

  if (directionSwitchState != previousDirectionSwitchState) {

    if (directionSwitchState == HIGH) {
      motorDirection = !motorDirection;
    }
  }

  if (motorDirection == 1) {
    digitalWrite(controlPin1, HIGH);
    digitalWrite(controlPin2, LOW);
  } else {
    digitalWrite(controlPin1, LOW);
    digitalWrite(controlPin2, HIGH);
  }

  if (motorEnabled == 1) {

    analogWrite(enablePin, motorSpeed);
  } else {

    analogWrite(enablePin, 0);
  }

  previousDirectionSwitchState = directionSwitchState;

  previousOnOffSwitchState = onOffSwitchState;
}

```

Discussion

1. **Why does the motor need a separate 9 volt battery to power it?**
   The motor requires more current than the Arduino can safely provide. The separate 9V battery supplies the necessary current to the motor without overloading the Arduino.

2. **What does IC stand for in electronics, and what IC was used in this lab?**
   IC stands for Integrated Circuit. The IC used in this lab was the L293D H-Bridge motor driver.

3. **What was the purpose of the H-bridge in this circuit?**
   The H-bridge allowed the motor to spin in either direction based on the control signals from the Arduino, and it safely handled the higher voltage and current required by the motor.

4. **How is the direction of the motor reversed in this lab setup?**
   By toggling the control pins connected to the H-bridge, one motor terminal is energized positively and the other negatively. Reversing these signals changes the motor's rotation direction.

Troubleshooting

1. **Issue:** The motor spun too fast to clearly view the zoetrope animation, making the images appear blurred.
   Solution: Adjusted the RPM of the motor using a potentiometer but it did not really get low enough. A lower-RPM motor would be better suited for this project.

Conclusion

The zoetrope behaved as expected, but the motor's high RPM made the animation difficult to view clearly. Although the basic concept worked, a lower RPM motor would have produced a much better animation effect. The reverse and stop/start buttons were very useful for controlling the motor, and worked exactly as intended. At high RPMs, significant vibrations occurred due to the unbalanced disc, while at lower voltages the motor struggled to maintain smooth rotation. No code adjustments were needed. The project successfully demonstrated motor control with an H-bridge and external power.