

Clap Light Switch

Yejun M. Kim

February 20, 2025

Contents

1	Introduction	1
2	Implementation	1
3	Appendix	2
3.1	A (Figures)	2
3.2	B (Code)	3

1 Introduction

The following design will allow the user to toggle a standard single pole light switch, found in most households across the United States, by physically toggling the light switch in response to a sound that exceeds a sound intensity threshold. A circuit and mechanical component must be integrated together for this task.

The tools required are a standard SG90 Arduino-compatible servo, a DC-DC voltage converter, LiPo battery, a KY037 sound sensor module, and an ATtiny85 chip. A circuit enclosure was designed to contain these components, provide enough clearance for the servo to rotate to turn the light switch on and off, and cover the light switch wall mount.

2 Implementation

The only component that potentially needed surge protection was the ATtiny, since all other components were rated for the 7V-12V range. A pullup resistor leading into the ATtiny was sufficient for the surge protection. A small capacitor was also used for further protection between the battery source and the ATtiny.

The sound sensor was adjusted via the built-in potentiometer for sensitivity and successfully detected sounds in the desired sound intensity range. The implementation was successful, and a picture of the components are shown in

figure 1. The designed box is shown in figure 2, and the servo arm is shown in figure 3, attached to the sg90 servo.

Since the project is relatively simple, the Arduino code used is provided directly in code 1.

3 Appendix

3.1 A (Figures)

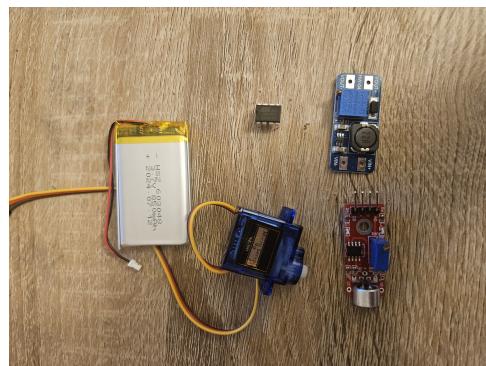


Figure 1: Electrical components required



Figure 2: 3D-printed electrical enclosure



Figure 3: Custom SG90 servo arm

3.2 B (Code)

```
ClapSwitch.ino
1 #include <Servo.h>
2
3 int analin = A0; //assign to pin A2
4 int digin = 8;
5 int servoPin = 9;
6 int val = 0;
7 int lastState = 0;
8 bool rotate = true;
9 Servo myServo;
10
11 void setup()
12 {
13     pinMode(digin, INPUT);
14     Serial.begin(9600);
15     myServo.attach(9);
16 }
17
18 void loop()
19 {
20     val = digitalRead(digin);
21
22     if (val == HIGH && laststate == LOW) {
23         if (rotate) {
24             myServo.write(60);
25             delay(1000);
26         }
27         else {
28             myServo.write(0);
29             delay(1000);
30         }
31         Serial.println("Rotate Servo");
32         rotate = !rotate;
33     }
34     lastState = val;
35 }
```

Figure 4: Arduino code for Clap Switch