Mapúa University

School of Electrical, Electronics and Computer Engineering

Real-Time Embedded System

COE187P/ E01

Actuators

Lab: #06

Submitted By:

Sardina, Kent Johnric M.

Submitted To:

Engr. Jocelyn Villaverde

I. Introduction

There are many examples of actuators and the most used actuator is Electric Motors which is used in many applications, an actuator is a component of a machine that is capable or responsible for moving a system. A relay is a switch that is controlled by electrical power when you want to control a component that needs a high voltage but your microcontroller only outputs lower voltage; there are many applications for the relay used in automation and controlled high powered devices.

II. Objectives

1. Familiarize the basic connection and functions of Arduino
2. Familiarize with how to code Arduino
3. Familiarize with using different functions of Arduino
4. Familiarize with using different modules for the Arduino
5. Use a Motor
6. Use a Relay module

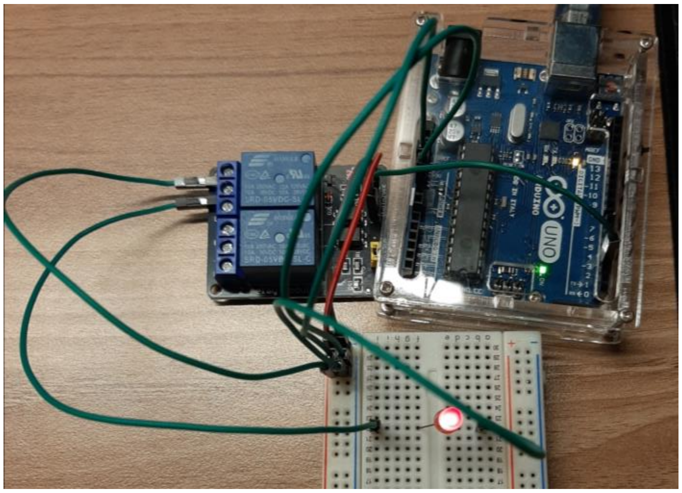
III. Materials and Components

- Arduino UNO and peripherals
- Jumper Wires
- Breadboard
- Arduino IDE / Visual Studio with Arduino IDE plugin
- 2 Relay Module for Arduino
- Motor
- LED

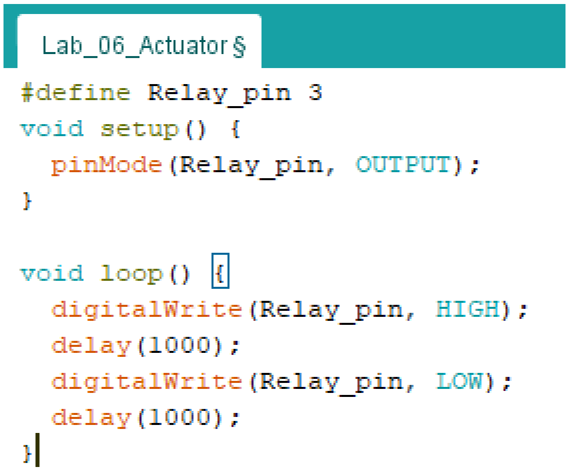
IV. PROCEDURE

Part 01: Relay

1. Follow the Connection to the Arduino



2. Connect the VCC to 5 volts and GND to the Ground of the Arduino
3. Connect the IN1 or signal pin of the Relay to the pin 3 of the Arduino
4. Connect 5 volts to 1 pin of the (always open pin) of the Relay Module
5. Connect the other pin of the (always open pin) of the Relay Module to the LED
6. Open Arduino IDE or Visual Studio (with Arduino IDE plugin)
7. Enter the code below



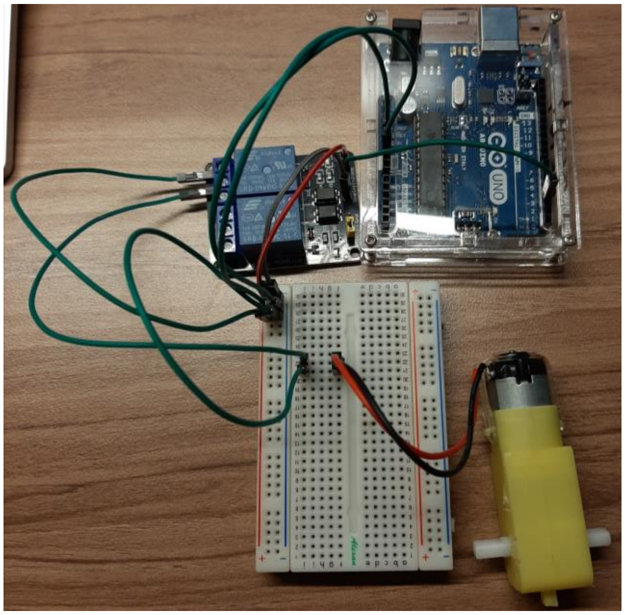
```
#define Relay_pin 3
void setup() {
  pinMode(Relay_pin, OUTPUT);
}

void loop() {
  digitalWrite(Relay_pin, HIGH);
  delay(1000);
  digitalWrite(Relay_pin, LOW);
  delay(1000);
}
```

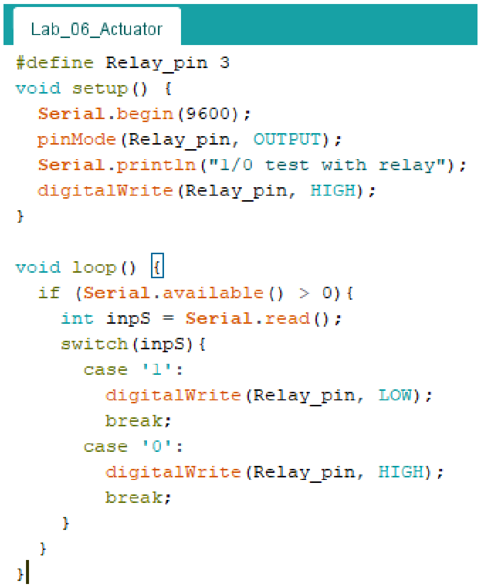
8. Select the correct COM port of the Arduino
9. Verify, Save, and Upload the code to the Arduino UNO
10. Test and observe the results

Part 02: Relay and Motor

1. Follow the Connection to the Arduino



2. The connection is the same from the first part just replace the LED to Motor
3. Open Arduino IDE or Visual Studio (with Arduino IDE plugin)
4. Enter the code below



Lab_06_Actuator

```
#define Relay_pin 3
void setup() {
  Serial.begin(9600);
  pinMode(Relay_pin, OUTPUT);
  Serial.println("1/0 test with relay");
  digitalWrite(Relay_pin, HIGH);
}

void loop() {
  if (Serial.available() > 0){
    int inpS = Serial.read();
    switch(inpS){
      case '1':
        digitalWrite(Relay_pin, LOW);
        break;
      case '0':
        digitalWrite(Relay_pin, HIGH);
        break;
    }
  }
}
```

5. Select the correct COM port of the Arduino
6. Verify, Save, and Upload the code to the Arduino UNO
7. Open the Serial monitor of the Arduino IDE and enter 1 to turn ON the motor and 0 to turn off the motor
8. Test and observe the results

V. Results and Discussion

In this experiment, I used a 2 relay module for the Arduino and this experiment doesn’t need any library just pure code in the Arduino since the relay is not a sensor; I also used a motor as the actuator for the second part of the experiment. In the first part of the experiment, I used a LED to indicate the relay is working and used a delay function in the loop to turn on and off the relay. In the second part of the experiment I replaced the LED with a motor and in the code I utilized the Serial Monitor of the Arduino to turn on and off the relay and motor, I used a Serial.available(), Serial.read() functions, and switch method for the Arduino to read the input from the serial monitor.

VI. Conclusion

After the parts and activities were completed, all the objectives in this experiment were achieved. By using different functions of the Arduino and the knowledge I studied in these past experiments I was able to create, run, and test the functionality of the relay module. I just simply used the basic functions of the Arduino in the first part of the experiment and in the second part I used intermediate functions of the Arduino, by experimenting with different functions of the Arduino you can gain a lot of knowledge of programming the microcontroller and to see how easy it is to code the microcontroller.