Proyecto final Microcontroladores:

Mousetrap

Integrantes:

Lucio Quintana
Ruben Novelli
Fernando Montero

Índice

1 Objetivo

2 Desarrollo del proyecto

- 2.1 Materiales necesarios
- 2.2 Diagrama
- 2.3 Montaje

3 Presupuesto

4 Conclusión

5 Código fuente del programa

1 Objetivo

El objetivo del proyecto es la realización de un dispositivo atrapa roedores. El trabajo esta basado en el proyecto de instructables ***https://www.instructables.com/id/Arduino-Mousetrap/*** al cual se le realizaron mejoras, tales como un led indicador y un botón de inicio/reset del dispositivo.

2 Desarrollo del proyecto

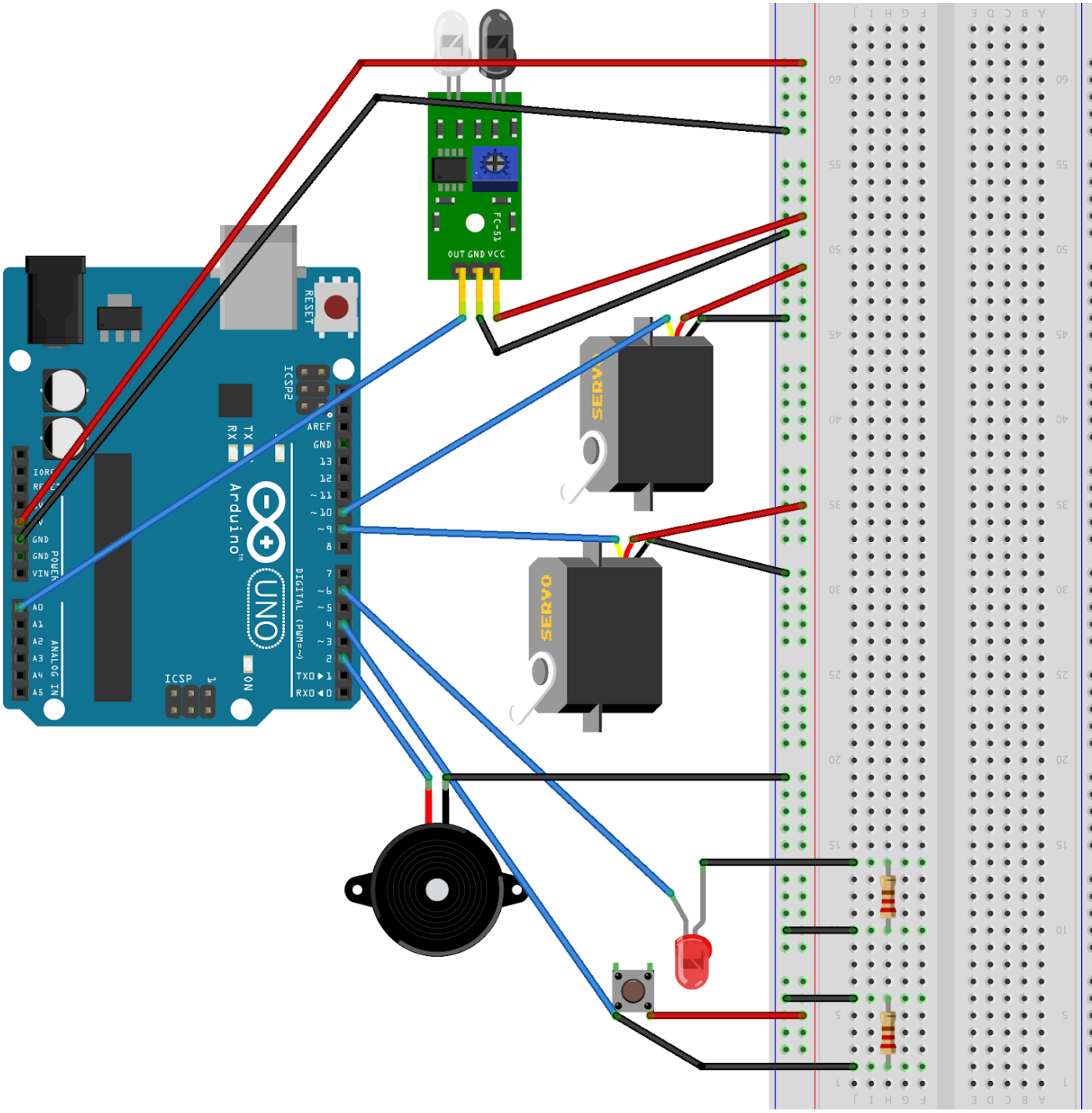
2.1 Materiales necesarios

Los materiales necesarios para la realización del proyecto son:

1. Arduino Uno
2. Servomotor
3. Led
4. Buzzer
5. Sensor de obstáculos
6. Botón
7. Resistencias (10k para el boton, 1K para el LED)
8. Protoboard
9. Cables
10. Caja

2.2 Diagrama

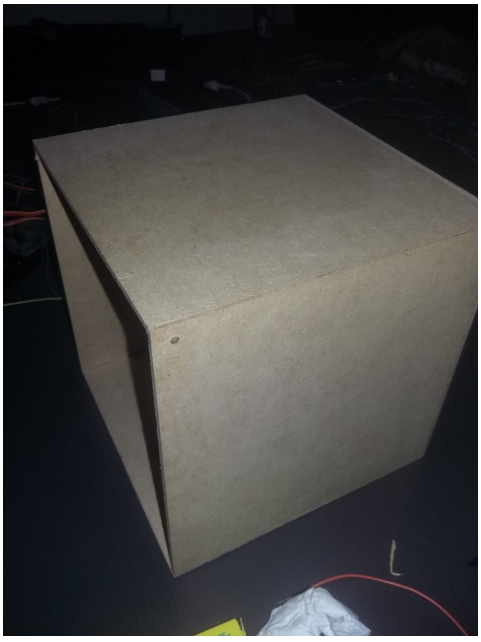
Para realizar el montaje de los componentes se debe utilizar el siguiente diagrama:



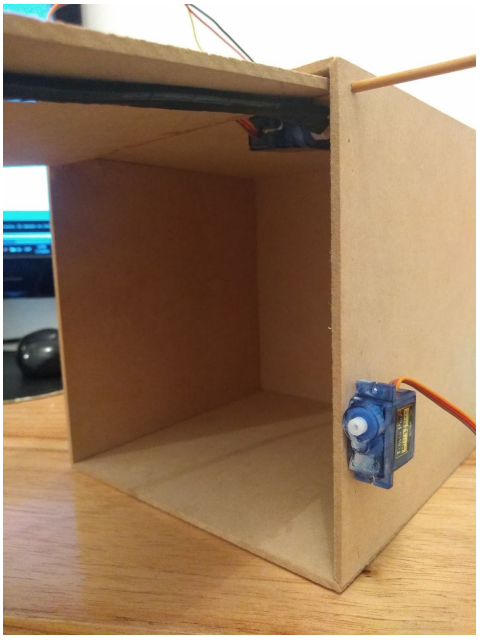
2.3 Montaje

Todo el prototipo se ha montado utilizando una caja de madera. Los diferentes componentes fueron pegados usando ***eccole*** que dio excelentes resultados.

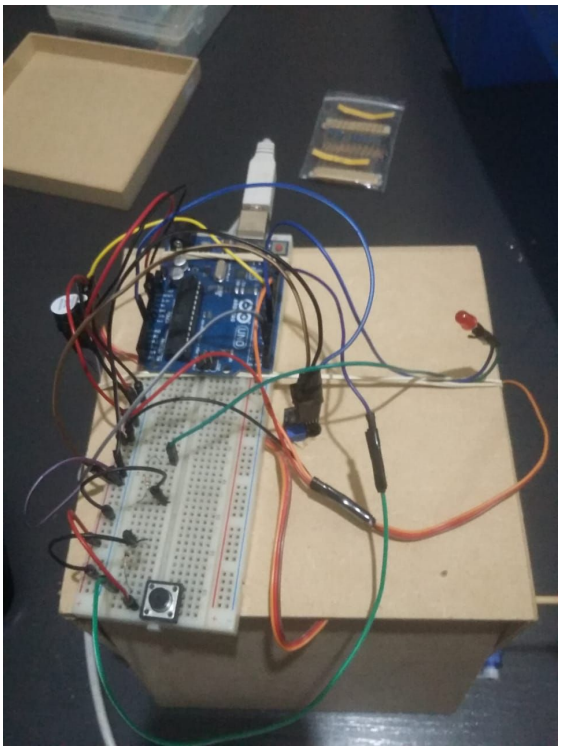
Primero se agujereó la caja para pasar la bisagra de la puerta.



Luego se instaló la bisagra y se pegaron los dos servos. Uno dentro y otro en el lateral.

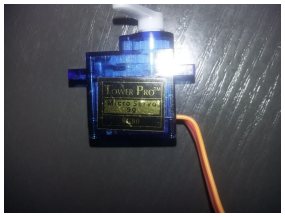
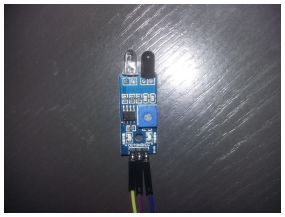
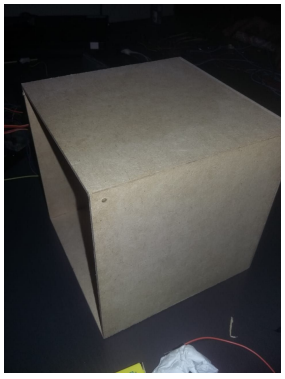


Se agregó el sensor de obstáculos sobre la caja, el LED, el buzzer y el botón.



3 Presupuesto

Para la realización del proyecto se compraron los siguientes componentes:

Producto	Nombre	Unidades	Precio
	Servomotor	2	$400
	Sensor de obstáculos	1	$300
	Buzzer	1	$70
	Caja de madera	1	$80

Los componentes faltantes fueron utilizados los provistos por el curso.

4 Conclusión

Se pudo realizar el proyecto con éxito y también logramos hacerle dos mejoras, que son el agregado de un botón para reiniciar el sistema y un led.
Las dificultades que tuvimos fueron:

El armado de la caja, tuvimos varios problemas para pegar los componentes y las partes de la caja de madera. Lo solucionamos principalmente usando mucho pegamento Eccole.

Problemas por contactos débiles en los cables y el uso de un cable que no funcionaba. Se probó cada cable y se ajustaron las conexiones.

Error en la conexión del botón que provocaba errores en la lectura de los valores. Buscamos en internet y usamos un esquema pull down para evitar un cambio imprevisto al presionar el botón.

Problemas con el ajuste de la sensibilidad del sensor, optamos por dejarlo original.

5 Código fuente del programa

```
#include <Servo.h>

Servo myservo;
Servo myservo2;

const int ledPin = 6;
const int buttonPin = 11;
int buttonState = 0;
int ledState = LOW;
int speakerPin = 2;
bool encendido = false;

void setup() {
  myservo.attach(9);
  myservo2.attach(10);
  pinMode(speakerPin, OUTPUT);
```

```
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
}

void loop() {
  int sensorValue = analogRead(A0);
  buttonState = digitalRead(buttonPin);

  if ((sensorValue < 500) && encendido) {
    myservo.write(110);
    delay(300);
    myservo2.write(10);
    digitalWrite(speakerPin, HIGH);
    delay(250);
    digitalWrite(speakerPin, LOW);

    if (ledState == LOW) {
      ledState = HIGH;
    } else {
      ledState = LOW;
    }

    digitalWrite(ledPin, ledState);
  }

  if (buttonState == HIGH) {
    if (encendido == true){
      encendido = false;
      myservo.write(0);
      myservo2.write(80);
      digitalWrite(speakerPin, LOW);
      digitalWrite(ledPin, LOW);
      } else {
        encendido = true;
      }
  }

}
```