

CIRCUITOS ELECTRICOS

PROYECTO FINAL

Saul Cervantes Candia – 177927

Ana Karen Cuenca Esquivel - 177932

Universidad Politecnica de San Luis Potosi

Contenido

Marco teorico.....	3
Introducción.....	4
problemática	4
OBJETIVO.....	4
Desarrollo.....	5
SOLUCION PROPUESTA	5
Sistema de alimentacion automatizada.....	5
SISTEMA DE RIEGO AUTOMATIZADO.....	5
Diseño del circuito.....	6
ARMADO DEL CIRCUITO.....	7
Desarrollo del software.....	9
RESULTADOS	13
Conclusion.....	16
bibliografia	17



MARCO TEORICO

Los circuitos eléctricos son fundamentales en nuestra vida cotidiana. Desde los sistemas de iluminación y la electrónica de consumo hasta la generación y distribución de energía eléctrica, los circuitos eléctricos están presentes en todas partes. En este proyecto, exploraremos los fundamentos de los circuitos eléctricos, incluyendo la ley de Ohm, la resistencia, la corriente, el voltaje y la potencia y aplicación de las leyes de Kirchhoff. Las leyes de Kirchhoff son dos principios fundamentales de la teoría de circuitos eléctricos: la Ley de Corrientes de Kirchhoff (LCK) y la Ley de Voltajes de Kirchhoff (LCK).

La LCK establece que, en cualquier nodo de un circuito eléctrico, la suma algebraica de las corrientes entrantes y salientes es igual a cero.

Por otro lado, la LCK establece que, en cualquier trayectoria cerrada de un circuito eléctrico, la suma algebraica de las caídas de voltaje es igual a la suma algebraica de las fuentes de voltaje.

Estas leyes son fundamentales para el análisis y diseño de circuitos eléctricos, y su aplicación es esencial para el éxito del proyecto de automatización de una granja, donde se pueden controlar los flujos eléctricos y la distribución de energía para alimentar y controlar los diferentes sistemas y dispositivos de la granja.

INTRODUCCIÓN

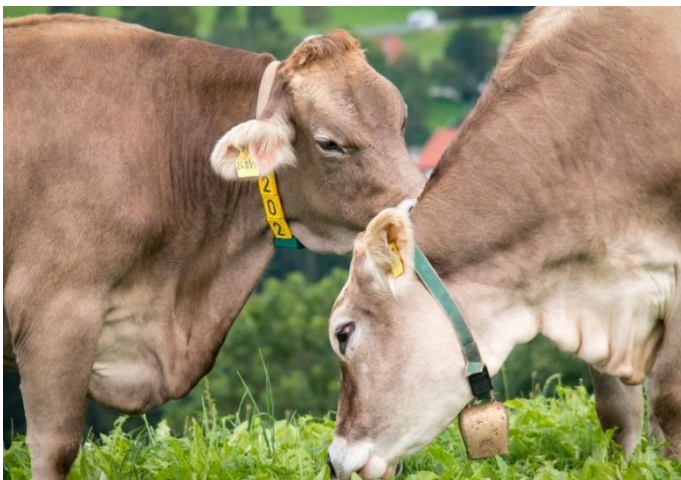
Los circuitos eléctricos son fundamentales en nuestra vida cotidiana. Desde los sistemas de iluminación y la electrónica de consumo hasta la generación y distribución de energía eléctrica, los circuitos eléctricos están presentes en todas partes. En este proyecto, exploraremos los fundamentos de los circuitos eléctricos, incluyendo la ley de Ohm, la resistencia, la corriente, el voltaje y la potencia y aplicación de las leyes de Kirchhoff. También aprenderemos a diseñar y construir circuitos eléctricos simples, utilizando componentes como resistencias, capacitores y diodos. Además, analizaremos circuitos más complejos y discutiremos cómo se pueden utilizar para aplicaciones prácticas, como el control de motores y la electrónica de potencia. Al final del proyecto, los participantes tendrán una comprensión sólida de los circuitos eléctricos y estarán equipados con las habilidades necesarias para diseñar y construir sus propios circuitos.

PROBLEMÁTICA

En una granja se busca tecnificar sus operaciones, se plantea el desafío de optimizar y controlar de manera eficiente sus sistemas automatizados, como el riego y la alimentación, mediante el uso de un sensor de infrarrojo. La problemática implica lograr una calibración precisa del sensor para garantizar una detección confiable, programar y sincronizar adecuadamente los sistemas automatizados, implementar un sistema de monitoreo remoto en tiempo real y establecer un plan de mantenimiento efectivo.

OBJETIVO

Aplicar el conocimiento aprendido sobre los circuitos eléctricos, como ley de Ohm, leyes de Kirchhoff y sus derivadas, esto dentro un proyecto que sea útil para la vida diaria. En este caso, será la automatización de una granja.





Arduino

- Sera el “cerebro” o base del proyecto



Resistencias



Protoboard

Fuente externa arduino



Relay



Sensor ultrasónico



DESARROLLO

SOLUCION PROPUESTA

Para la creación de este proyecto se utilizará prototipos en base a la plataforma de Arduino, con sus sensores correspondientes, IDE y demás componentes, esto ya que es una plataforma de hardware y software de código abierto diseñada para la creación de proyectos electrónicos interactivos. Igualmente, se usará material reciclado para promover los valores sustentables.

Se utilizará el método de mallas ya que es una metodología que ya dominamos y podremos sacar el valor de la corriente para cada resistencia, etc, también se usará la ley de ohm para obtener voltaje, corriente, etc.

SISTEMA DE ALIMENTACION AUTOMATIZADA

Un sistema de alimentación automatizado es una tecnología de automatización que se utiliza en la industria agroalimentaria para alimentar automáticamente a los animales de granja. El sistema automatizado se compone de varios componentes, incluyendo tolvas de alimentación, transportadores de alimentos, controles de alimentación y sensores. Dentro de nuestro proyecto usaremos:

- Arduino UNO: es un microcontrolador que nos ayudara a controlar cada sensor. Soporta 50 mA, ofrece 5v y costo \$140.
- Tubo PVC: servirá para guardar el alimento, la función de la tolva. Tuvo un precio de \$60.
- Servomotor: servirá para la salida del alimento guardado en la tolva. Maneja entre 4 y 8 volts, con un precio de \$70.
- Sensor ultrasónico: servirá para saber la cantidad del alimento guardado, de acuerdo con la distancia que haya entre este sensor y el alimento. \$60
- Sensor Infrarrojo y control IR: servirá para generar la señal de inicio y fin de cada sensor. \$80
- Buzzer: servirá para señalar cuando se acabe el alimento. \$15
- Batería de 9v: nos ayudara con la alimentación del microcontrolador. \$60

SISTEMA DE RIEGO AUTOMATIZADO

Los sistemas de riego automatizados permiten un riego más preciso y eficiente, ya que se pueden programar para regar en momentos específicos del día o la semana, en función de las necesidades del cultivo o de las condiciones climáticas.

Mini bomba de agua

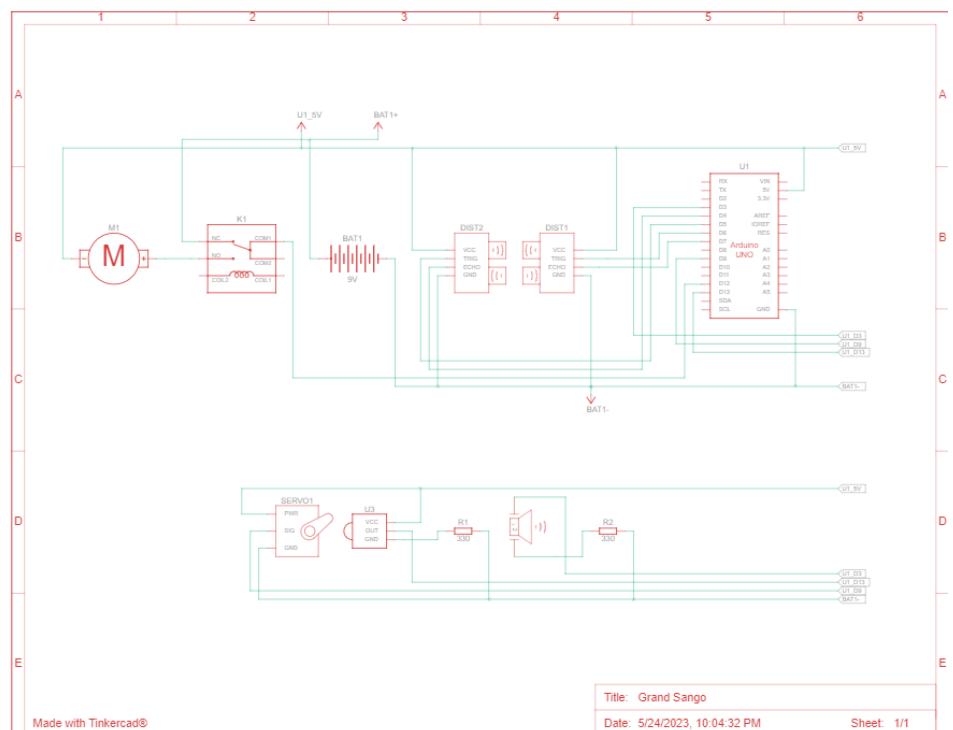
Por lo que será importante para la automatización de la granja, para esto usaremos los siguientes componentes:

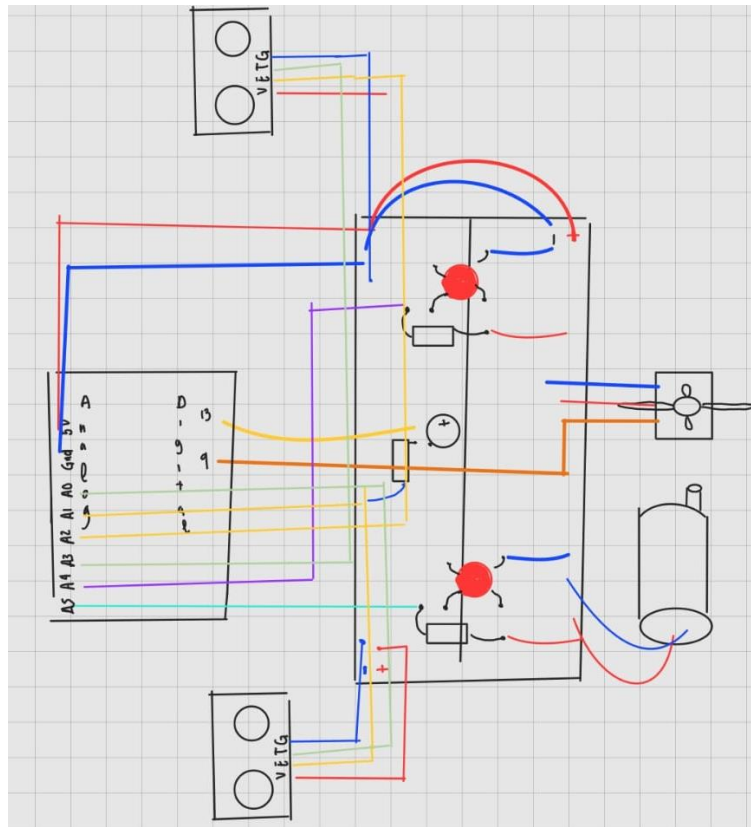
- Tubo PVC: servirá para guardar el líquido, la función de la tolva.
- Mini bomba de agua: este nos ayudara con el movimiento del agua y así proporcionar el agua cuando se requiera. (precio)
- Tubos: nos ayudaran a transportar el agua
- Sensor ultrasónico: servirá para saber la cantidad del agua guardada, de acuerdo con la distancia que haya, y así saber si se acabó el producto. \$60
- Buzzer: servirá para señalar cuando se acabe el alimento. \$15
- Batería de 9v: nos ayudara con la alimentación del microcontrolador. \$60

DISEÑO DEL CIRCUITO

Para el armado del circuito se ocupó dos sensores ultrasónicos, cables para generar la conexión, un buzzer, vasos de plástico, un servomotor, mini bomba de agua, relé, un sensor infrarrojo.

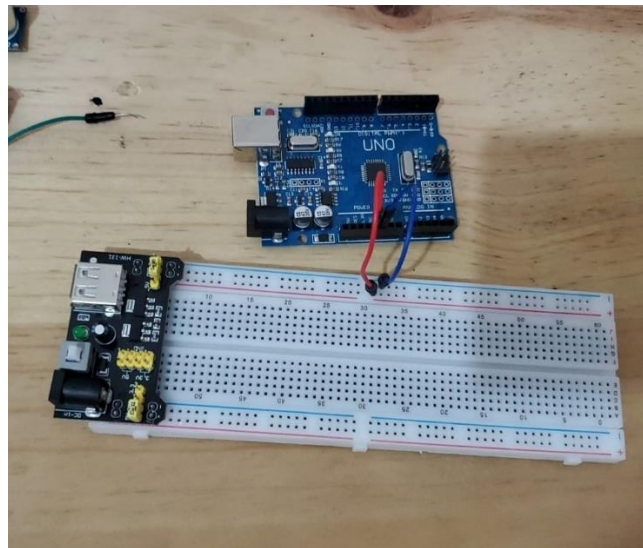
Con el fin de simplificar el circuito, lo manejaremos con una sola protoboard. Elegimos como seria la conexión de la tierra y el voltaje de este, alimentándolo a partir del mismo Arduino y una fuente externa que alimentara un carril para la Mini bomba. Para el diseño, fue imprescindible investigar como es que debían ser conectados cada sensor, por lo que se decidió de esta manera el diseño del circuito:



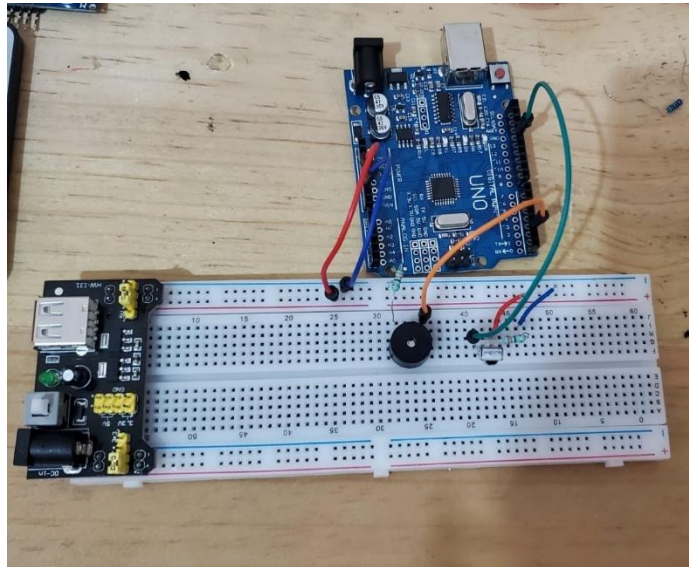


ARMADO DEL CIRCUITO

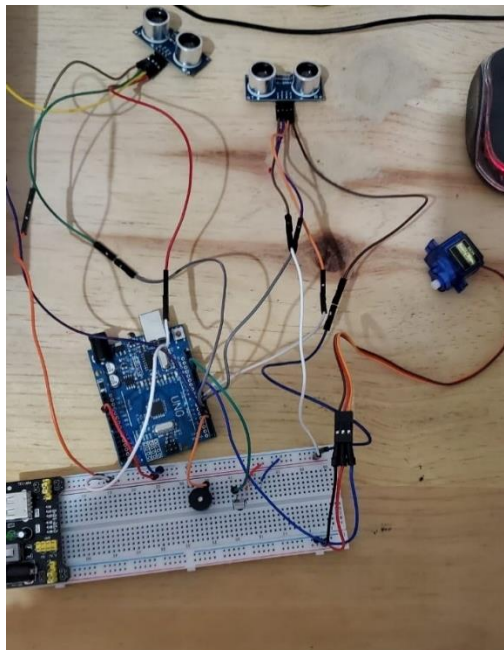
Colocamos la proto de manera que iniciemos con tierra y luego voltaje, para así coincidir con la fuente externa. Igualmente, se hizo la conexión del Arduino a uno de los carriles de la protoboard, de ese lado, la fuente externa estaba en off, mientras que del otro está entregando 5v.



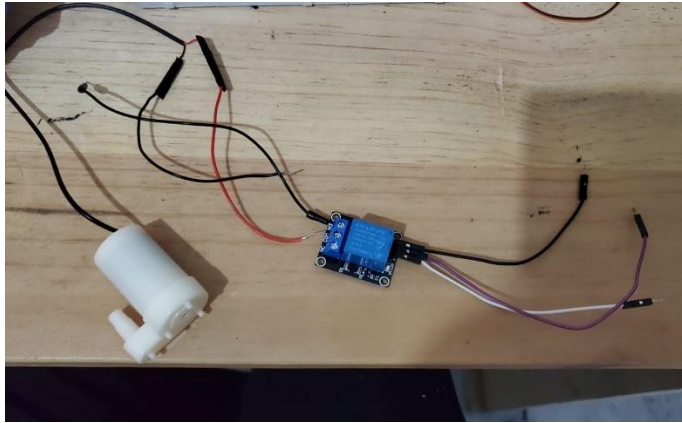
Continuamos con los sensores de infrarrojo y buzzer, cada uno conectado en el negativo con una resistencia y este a su vez a la tierra del circuito, de su lado positivo, se conectó a los 5v y se conectaron a los pines digitales 13 y 3 respectivamente.



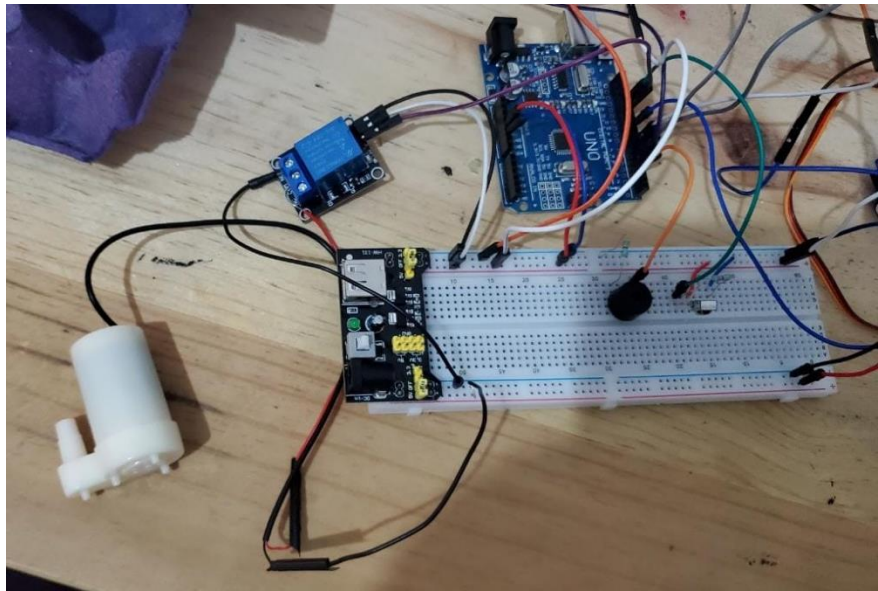
Le colocamos jumper macho-hembra a los sensores ultrasónicos y unos jumpers macho-macho al servomotor. Se les conecta a su voltaje y tierra y a los ultrasónicos se les conecta a los pines 7,6,5 y 4 para sus echo y trig. Al servomotor lo conectamos a el pin 9.



Conectamos la mini bomba de su lado positivo a la entrada común del relé y agregamos un cable a la entrada NO.



Después conecte al relé al positivo y negativo de la proto y la señal al pin 12 del Arduino. Por último, conecte el cable negro de la mini bomba a positivo y el cable de NO a la tierra.



Durante la construcción del circuito, tuvimos un problema con la alimentación de este, puesto que la mini bomba demanda demasiados voltajes a comparación de los demás componentes. Igualmente se tuvo el problema de las conexiones previo a terminar el circuito



WhatsApp Video 2023-05-24 at 8.20.32 PM.mp4

DESARROLLO DEL SOFTWARE

El objetivo del programa es recibir comandos a través del control remoto infrarrojo y controlar el movimiento del servo motor, activar una bomba y emitir señales acústicas utilizando el buzzer.

```
#include <Servo.h>
#include <IRremote.h>

int receptor = 13;
```

```

IRrecv irrecv(receptor);
decode_results codigo;
#define SERVO_PIN 9
#define BOMBA 12
Servo servoMotor;
bool servoMoved = false; // Saber si se movió o no
bool bombaMoved = false; // Saber si se movió o no
const int echo = 6;
const int trigger = 7;
long tiempo;
float distancia;
const int echo2 = 5;
const int trigger2 = 4;
long tiempo2;
float distancia2;
const int buzzerPin = 3;

void setup() {
    // Infrarrojo
    irrecv.enableIRIn();
    // Ultrasonico
    pinMode(echo, INPUT);
    pinMode(trigger, OUTPUT);
    // Ultrasonico 2
    pinMode(echo2, INPUT);
    pinMode(trigger2, OUTPUT);
    // Buzzer
    pinMode(buzzerPin, OUTPUT);
    // Servomotor
    servoMotor.attach(SERVO_PIN);
    // Boton
    pinMode(BOMBA, OUTPUT);

    servoMotor.writeMicroseconds(1500); // Posición inicial del servo (detenido)
    Serial.begin(9600);
}

void medir() {
    digitalWrite(trigger, LOW);
    delayMicroseconds(2);
    digitalWrite(trigger, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigger, LOW);
    tiempo = pulseIn(echo, HIGH, 10000);
    distancia = tiempo / 58;
    delay(10);
}

void medir2() {
    digitalWrite(trigger2, LOW);

```

```

    delayMicroseconds(2);
    digitalWrite(trigger2, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigger2, LOW);
    tiempo2 = pulseIn(echo2, HIGH, 10000);
    distancia2 = tiempo2 / 58;
    delay(10);
}

void loop() {
    medir();
    medir2();
    Serial.println(irrecv.decode(&codigo));
    if (irrecv.decode(&codigo)) {
        switch (codigo.value) {
            case 0xFF6897: // Botón cero// ABRIR SERVOMOTOR
                if (distancia < 8) { // Si la distancia es menor a 8, hay obstrucción
                    digitalWrite(buzzerPin, LOW);
                    // Gira el servo 90 grados en una dirección
                    servoMotor.writeMicroseconds(1400);
                    delay(1000); // Ajusta este tiempo según la velocidad de tu servo
                    // Detiene el servo
                    servoMotor.writeMicroseconds(1500);
                    servoMoved = true;
                }
            else if(distancia>8) {
                digitalWrite(buzzerPin, HIGH);
                Serial.println(distancia);
                delay(500);
            }
            break;
            case 0xFF30CF: // Botón uno// CERRAR SERVOMOTOR
                // Regresa el servo a la posición inicial
                if(servoMoved==true){
                    servoMotor.writeMicroseconds(1600);
                    delay(1000); // Ajusta este tiempo según la velocidad de tu servo
                    servoMotor.writeMicroseconds(1500);
                    servoMoved = false;
                }
            break;
            case 0xFF18E7: // Botón dos
                //apagar buzzer
                digitalWrite(buzzerPin, LOW);
                break;
            case 0xFF7A85: // Botón tres
                // Enciende el relé para activar la bomba
                if (distancia2 < 8) { // Si la distancia es menor a 8, hay obstrucción
                    digitalWrite(BOMBA, HIGH);
                    bombaMoved=true;
                    delay(5000);
                }
            }
        }
    }
}

```

```

        digitalWrite(BOMBA, LOW);
    }else {
        digitalWrite(buzzerPin, HIGH);
        delay(500);
    }
    break;
case 0xFF10EF://boton 4
// Apaga el relé para detener la bomba
    if(bombaMoved==true){
        digitalWrite(BOMBA, LOW);
    }
    break;
}
irrecv.resume();
}
delay(100);
}

```

Arquitectura y Funcionamiento del Software:

El programa está estructurado en torno a la función setup() y la función loop(). La función setup() se ejecuta una vez al iniciar el programa y se utiliza para realizar la configuración inicial de los componentes, mientras que la función loop() se ejecuta continuamente en un ciclo y es responsable de leer los comandos del control remoto y controlar los componentes según corresponda.

Librerías utilizadas:

- Servo.h: Esta librería permite controlar servos motores. Proporciona funciones para adjuntar el servo a un pin, establecer el ángulo del servo y más.
- IRremote.h: Esta librería permite recibir y decodificar señales infrarrojas de un control remoto. Proporciona funciones para habilitar el receptor infrarrojo, recibir y decodificar los códigos de señales infrarrojas.

Funciones implementadas:

medir(): Esta función se utiliza para medir la distancia utilizando un sensor ultrasónico. Genera un pulso en el pin de activación del sensor y mide la duración del pulso de eco en el pin de recepción. Luego calcula la distancia en función del tiempo de eco y la almacena en la variable distancia.

medir2(): Esta función es similar a medir(), pero se utiliza para medir la distancia utilizando un segundo sensor ultrasónico. Almacena la distancia medida en la variable distancia2.

setup(): Esta función se ejecuta una vez al inicio del programa. Configura los pines de los componentes, como el receptor infrarrojo, los sensores ultrasónicos, el buzzer y el servo motor. Además, establece la posición inicial del servo motor y habilita la comunicación serial.

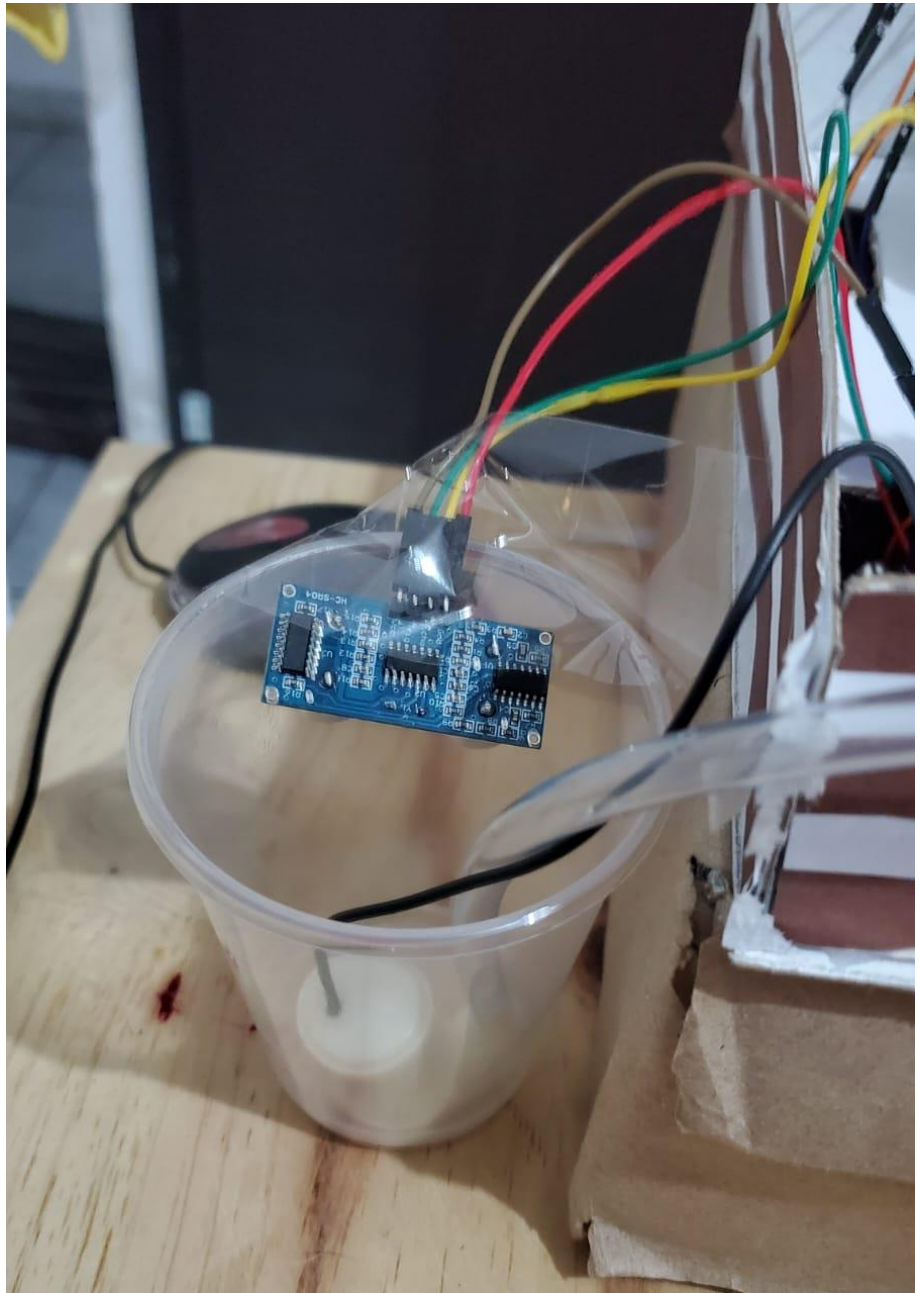
loop()): Esta función se ejecuta continuamente en un ciclo. Primero, llama a las funciones medir() y medir2() para actualizar las mediciones de distancia. Luego, verifica si hay un código de señal infrarroja recibido utilizando irrecv.decode(&codigo). Si se recibe un código, se verifica su valor y se ejecuta la acción correspondiente según el código recibido. Por ejemplo, si se recibe el código correspondiente al botón cero, verifica la distancia medida por el primer sensor ultrasónico y gira el servo motor si la distancia es menor a 8. Si se recibe el código correspondiente al botón uno, devuelve el servo motor a la posición inicial. Si se recibe el código correspondiente al botón tres, activa la bomba si la distancia medida por el segundo sensor ultrasónico es menor a 8. Si se recibe el código correspondiente al botón cuatro, detiene la bomba.

RESULTADOS

El resultado de este proyecto culmina con la unión del circuito con la maqueta, a la minibomba se le unió una manguera y se metió a un bote con agua y para el servomotor se le unió a una botella con alimento. Cada recipiente tiene encima un ultrasónico. Debajo de la caja se encuentra todo el circuito, conectado a la corriente. Las fuentes fueron conectadas a la corriente. No se obtuvieron los resultados deseados ya que hubo problemas con el sensor infrarrojo por la ubicación por lo que el circuito tuvo una respuesta lenta. Algunos posibles problemas del circuito fueron mala codificación del Arduino, un posible falso contacto entre cables con la protoboard, falta de voltaje para algún componente del circuito, ubicación del sensor infrarrojo. Algunas limitaciones presentes fueron el corto alcance del cargador de 9 V para la alimentación del Arduino o la falta de componentes por los precios que pueden estar elevados.

Nuestras soluciones fueron revisar con detalle cada línea de la programación para encontrar nuestros errores y corregirlos para darle una mejor funcionalidad al circuito, revisar las conexiones entre el Arduino, la protoboard y los componentes utilizados para no tener un falso contacto y no obtener la funcionalidad correcta, observar los movimientos a detalle de cada componente en la ejecución del circuito para poder corregir el error en caso de algún fallo.





CONCLUSION

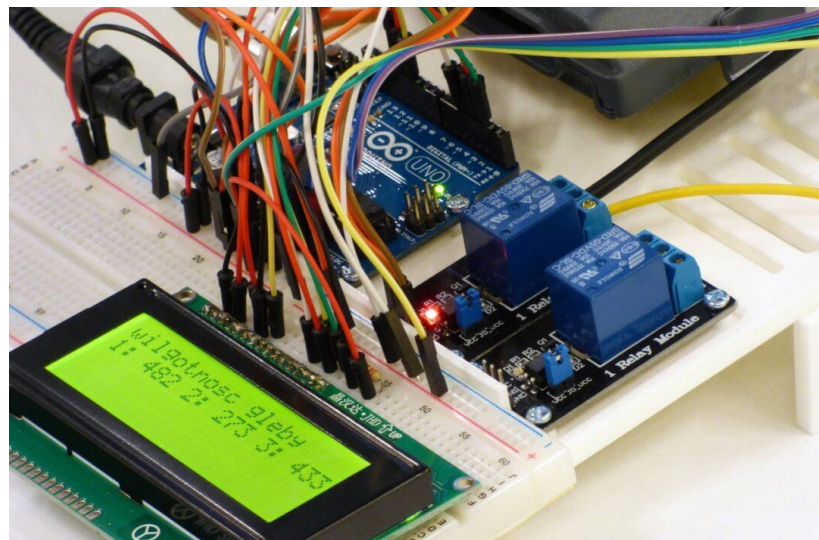
Los circuitos eléctricos son fundamentales en la automatización de una granja, ya que permiten el control y monitoreo de los diferentes sistemas eléctricos y electrónicos involucrados en el proceso de producción y alimentación de los animales.

Para entender los circuitos eléctricos, es importante conocer las bases teóricas de la electricidad, tales como la corriente, el voltaje, la resistencia, la potencia y la energía. Asimismo, las leyes de Kirchhoff son esenciales para el análisis y diseño de circuitos complejos.

El uso del Arduino como microcontrolador permite la automatización y control de los sistemas eléctricos y electrónicos de una granja de forma más sencilla y económica. Además, ofrece una amplia variedad de opciones para la integración de sensores y actuadores, lo que permite un monitoreo en tiempo real y una retroalimentación constante para mejorar la eficiencia y la productividad en la granja.

En resumen, la aplicación de la teoría de circuitos eléctricos y el uso del Arduino en la automatización de una granja pueden contribuir significativamente al aumento de la producción y al cuidado del medio ambiente, al permitir un uso más eficiente de los recursos y una reducción de los costos operativos.

"En la teoría de los circuitos eléctricos, las leyes de la naturaleza son las leyes de la aritmética." (Kirchhoff, 1847)



BIBLIOGRAFIA

- [1] W. H. Hayt y J. E. Kemmerly, "Análisis de circuitos en ingeniería," vol. 3, McGraw-Hill, 1986.
- [2] Arduino, "Arduino," 2021. [En línea]. Disponible en: <https://www.arduino.cc/> [Último acceso: 8 abril, 2023].
- [3] C. Acuña, "Granjas inteligentes: ¿el futuro de la industria agrícola?," Forbes México. [En línea]. Disponible en: <https://www.forbes.com.mx/granjas-inteligentes-el-futuro-de-la-industria-agricola/> [Último acceso: 8 abril, 2023].
- [4] G. R. Kirchhoff, "Ueber den durchgang eines electrischen stromes durch eine ebene scheibe," Annalen der Physik, vol. 138, no. 4, pp. 513-516, 1845.
- [5] G. S. Ohm, "Die galvanische kette, mathematisch bearbeitet," Annalen der Physik und Chemie, vol. 88, no. 6, pp. 1-52, 1827.