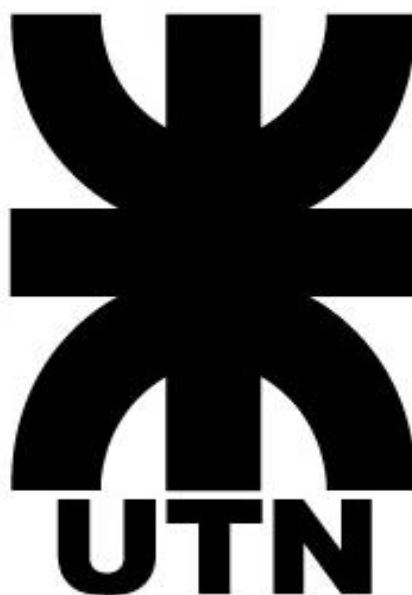


29-11-2023

PROYECTO ARDUINO

Teoría de Control 2023



Profesor:

- Ing. José Luis Catalano.
- JTP Ing. Nicolas Oliva

Integrantes:

- Leonardi Axel
- Mairone Nicolas
- Mattalia Enzo
- Namuncurá Manuel
- Simonin Eloy
- Zegatti Tomás

Contenido

Resumen	2
Breve descripción del proyecto.	2
Objetivos principales.	2
Resultados destacados.	2
Introducción	2
Contexto y justificación del proyecto	2
Breve descripción de la problemática a resolver	2
Objetivo	2
Objetivo general del proyecto	2
Objetivos específicos	2
Marco Teórico	3
Fundamentos Teóricos Relacionados con el proyecto	3
Tecnología o Conceptos Clave Utilizados	3
Diseño y Metodología	3
Descripción detallada del diseño del sistema	3
Metodología utilizada en el desarrollo.....	3
Esquema de Conexionado	4
Diagrama de conexión de los componentes de Arduino	4
Descripción de la conexión de cada componente.....	5
Desarrollo del código	5
Extractos clave del código utilizado.....	5
Explicaciones sobre el funcionamiento del código	7
Resultados	8
Presentación de los resultados obtenidos	8
Datos cuantitativos o cualitativos relevantes	8
Conclusiones	8

Resumen

Breve descripción del proyecto.

El proyecto consiste en la creación de un dispositivo utilizando Arduino que emplea un sensor de inclinación para detectar cambios en la posición física de un objeto. Cuando el sensor detecta que el objeto está invertido o inclinado en relación a su posición inicial, activa un relay que enciende un LED como señal visual de esta condición

Objetivos principales.

El objetivo principal que deseamos lograr, es que cuando pase determinado tiempo que el objeto mantenga una inclinación de más del 50%, esto aumente la intensidad del LED en modo de señal de que algo está funcionando mal. Mientras mas tiempo pase invertido, más intenso será el LED.

Resultados destacados.

Se logro el objetivo con éxitos de encender el LED, en el momento que el dispositivo superase cierto grado de inversión, además se aplico un porcentaje en relación a cuál es su grado de inversión.

Se logro aumentar la intensidad del LED a medida que pasa determinada cantidad de segundos en una posición establecida.

Introducción.

Contexto y justificación del proyecto

El proyecto se enmarca como un trabajo final para la asignatura "Teoría de Control", buscando aplicar los conceptos teóricos aprendidos y se justifica por el hecho de ser una ejercitación para poner en práctica, al menos de manera básica, la teoría de controles explicada en las clases y fundamentos de control al detectar y responder a cambios específicos en la posición de un objeto físico.

Breve descripción de la problemática a resolver

La problemática a resolver consiste en que debemos implementar en el lenguaje de programación de Arduino, un sistema que encienda un LED y regule su intensidad en relación a cuanto se mantiene en determinado grado de inclinación respecto a su posición inicial.

Objetivo

Objetivo general del proyecto

El objetivo general del proyecto es detectar y reconocer el grado de inclinación de un dispositivo, activando una alerta visual mediante un LED, con el fin de notificar al usuario sobre posibles situaciones incorrectas en el funcionamiento del producto.

Objetivos específicos

- Lograr que el LED encienda superado cierto grado de inclinación
- Lograr manipular la intensidad del LED, en base a la inclinación del dispositivo físico
- Lograr manipularlo mediante una conexión ethernet

Marco Teórico

Fundamentos Teóricos Relacionados con el proyecto

El proyecto involucra conceptos fundamentales de electrónica, programación y teoría de control.

1. Electrónica básica: Entender principios de circuitos, resistencias, voltaje, corriente eléctrica y la conexión de componentes como LED, sensores y relays
2. Sensor de inclinación: Conocer su funcionamiento, cómo detectan cambios en la orientación y traducen estos cambios en señales eléctricas interpretables por Arduino.
3. Programación Arduino: Conocer la estructura de programación básica, configuración de pines, lectura de señales de sensores y control de dispositivos de salida como LEDs.
4. Teoría de control básica: Comprender conceptos de sistemas de control, retroalimentación, detección de errores y acciones correctivas para activar el LED en respuesta a la detección de inclinación anormal.
5. Interfaz hardware-software: Comprender la vinculación entre la programación en Arduino y los componentes físicos para lograr el funcionamiento del dispositivo.

Tecnología o Conceptos Clave Utilizados

El desarrollo del proyecto se llevo a cabo con aportes de contenido encontrado en:

- Videos de YouTube
- ChatGPT
- Infografías halladas en internet
- Aportes de diferentes páginas web
- Los conocimientos del futuro ingeniero Tomas Zegatti
- Aportes de la página PIJA Education

Diseño y Metodología

Descripción detallada del diseño del sistema

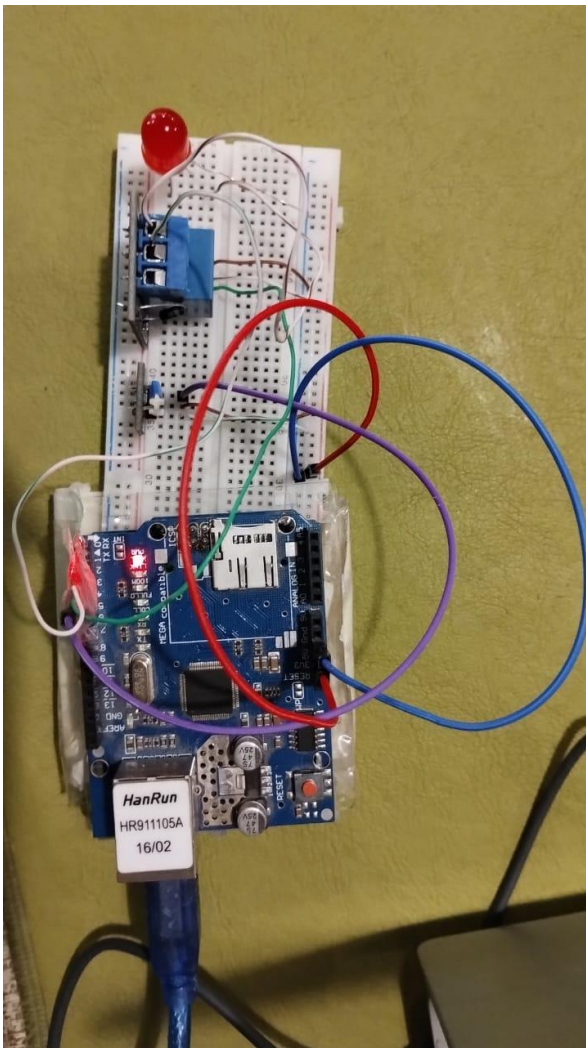
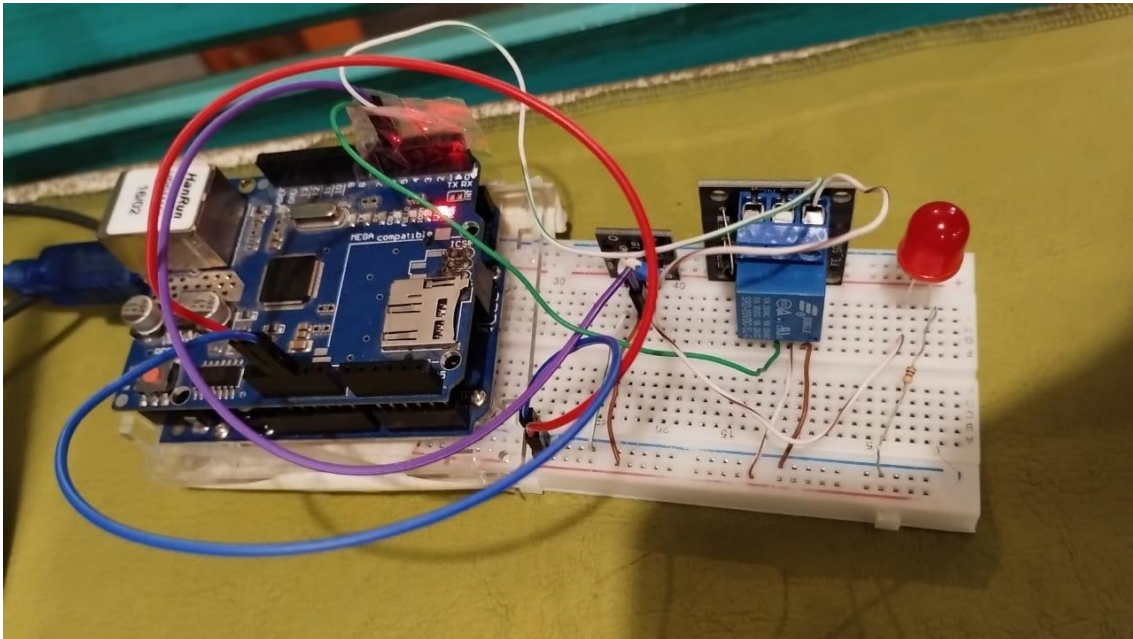
El sistema consta de un sensor de inclinación conectado a una placa Arduino. El sensor detecta la inclinación y envía señales a la placa. La metodología incluye la lectura de estas señales por el Arduino y el uso de una programación específica para activar un LED en respuesta a diferentes grados de inclinación. Este diseño permite alertar visualmente sobre cambios de orientación del dispositivo mediante la iluminación del LED.

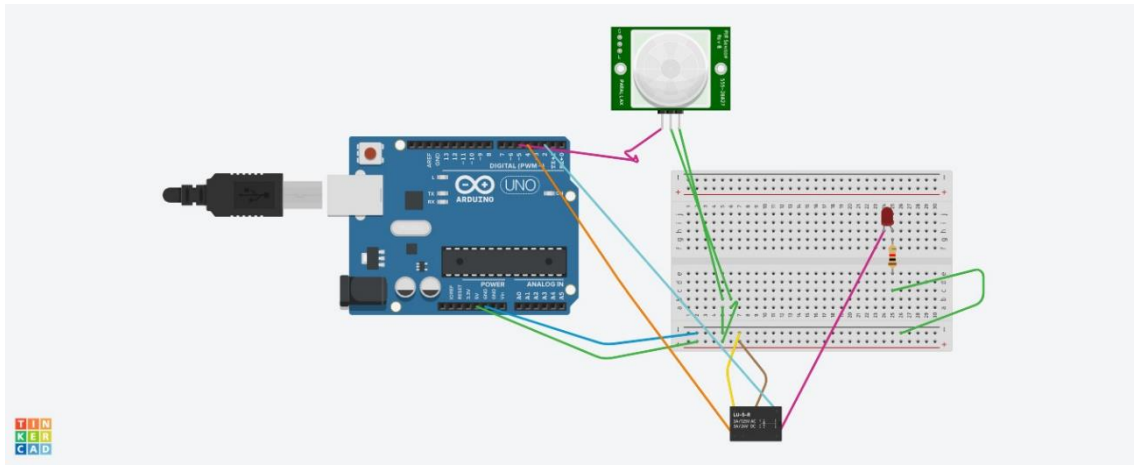
Metodología utilizada en el desarrollo

Nos regimos bajo la metodología del trabajo en conjunto, planificamos reunión en el transcurso de la semana para poder llevar a cabo el desarrollo del sistema, además, de realizar un aprovechamiento optimo del tiempo de clases para despejar dudas.

Esquema de Conexionado

Diagrama de conexión de los componentes de Arduino





Descripción de la conexión de cada componente

Componentes:

1. Sensor de inclinación: Consta de tres pines, usualmente VCC (alimentación), GND y la salida de señal. Conecta VCC a un pin de 5V en Arduino, GND a un pin GND en Arduino y la salida de señal a un pin digital de entrada en Arduino (por ejemplo, pin 2).
2. Relé: El relé también tiene pines de VCC, GND y uno o más pines de control (generalmente IN). Conecta VCC a una fuente de alimentación adecuada y GND a un pin GND en Arduino. Conecta el pin de control del relé a un pin digital de salida en Arduino (por ejemplo, pin 7).
3. LED: Conecta el ánodo (lado largo) del LED a través de una resistencia limitadora a uno de los contactos comunes del relé. El cátodo (lado corto) del LED se conecta a una fuente de alimentación GND.

Desarrollo del código

Extractos clave del código utilizado

// Definición de pines

const int sensorPin = 5; // Pin conectado al sensor de inclinación

const int relayPin = 4; // Pin conectado al relé

const int relayOutput = 2; // Pin conectado al pin de salida del relé (NO o NC)

int estadoAlerta = 0;

long tiempo = 0;

long tiempoCero = 0;

unsigned long millisPrevio = 0;

void setup() {

pinMode(sensorPin, INPUT); // Configurar el pin del sensor como entrada

pinMode(relayPin, OUTPUT); // Configurar el pin de control del relé como salida

pinMode(relayOutput, OUTPUT); // Configurar el pin de salida del relé como salida

tiempoCero = millis();

Serial.println(tiempoCero);

Serial.begin(9600);

```

}

void loop() {
  delay(100);
  int sensorValue = digitalRead(sensorPin); // Leer el estado del sensor
  Serial.println(sensorValue);
  Serial.println(tiempoCero);
  Serial.println("tiempo vale");
  Serial.println(tiempo);
  // Si el sensor de inclinación está activado (cambia según tu tipo de sensor)

  // el sensor nuestro "arranca" en HIGH

  if (sensorValue == LOW) {
    //Serial.print("el coso se dio vuelta");
    digitalWrite(relayPin, HIGH); // Activar el relé
    digitalWrite(relayOutput, HIGH); // Activar el pin de salida del relé (NO o NC)
    //se arranca a contar el tiempo, restandole el tiempo previo para resetear el contador
    tiempo = millis() - millisPrevio;

    if ((tiempo > (tiempoCero + 10000)) && (tiempo < (tiempoCero + 15000))){
      Serial.println("alerta 10 segundos");
      parpadeo(1000);
    } else {
      if ((tiempo > tiempoCero + 15000) && (tiempo < tiempoCero + 20000)){
        Serial.println("alerta 15 segundos");
        parpadeo(500);
      } else {
        if (tiempo > tiempoCero + 20000){
          Serial.println("alerta 20 segundos");
          parpadeo(250);
        }
      }
    }

    // cuando se da vuelta, envia la señal LOW
  } else {
    //Serial.print("se dio vuelta");
    millisPrevio = millis();

    digitalWrite(relayPin, LOW); // Desactivar el relé
    digitalWrite(relayOutput, LOW); // Desactivar el pin de salida del relé (NO o NC)

  }
}

```

Explicaciones sobre el funcionamiento del código

1. **Configuración de pines:** Se definen los pines que se utilizarán para el sensor de inclinación, el control del relé y su salida.
2. **Inicialización:** Se establecen los pines como entradas o salidas, y se inicia la comunicación serial para mostrar mensajes en el monitor serial.
3. **Bucle principal (loop):**
 - a. **Lectura del sensor de inclinación:** Lee el estado del sensor de inclinación (sensorPin) y almacena su valor en la variable sensorValue.
 - b. **Manejo de la inclinación:**
 - Si sensorValue es LOW, significa que se ha detectado una inclinación (dependiendo del sensor, la lógica puede variar).
 - Se activa el relé (relayPin) y su salida (relayOutput) para activar un dispositivo (puede ser una alarma).
 - Se inicia un temporizador (tiempo) que comienza a contar desde el momento en que se detecta la inclinación.
 - c. **Control de tiempo:**
 - Dependiendo del tiempo transcurrido desde que se detectó la inclinación (tiempo), se activan diferentes niveles de alerta, indicados por mensajes en el monitor serial y parpadeos (mediante la función parpadeo()).
 - Los distintos niveles de alerta se controlan por el valor de tiempo, aumentando la frecuencia del parpadeo a medida que pasa más tiempo.
 - d. **Fin de la inclinación:**
 - Cuando el sensor vuelve a su estado original (posición normal), se restablece el temporizador y se desactiva el relé y su salida.

Notas importantes:

- El código se basa en la lógica de un sensor que cambia a LOW cuando detecta inclinación y vuelve a HIGH cuando regresa a su posición normal. Esto puede variar dependiendo del tipo de sensor de inclinación utilizado en tu proyecto.
- Los tiempos de alerta (10, 15 y 20 segundos) se controlan mediante la variable tiempoCero. La función millis() se utiliza para llevar un seguimiento del tiempo transcurrido.
- La función parpadeo() controla el parpadeo de algún dispositivo (como un LED) según el nivel de alerta, aunque no se proporciona en el código mostrado. Presumiblemente, esta función controla la activación y desactivación del dispositivo de manera intermitente.

Recuerda ajustar el código según las especificaciones de tu sensor de inclinación, el hardware utilizado y las acciones específicas que deseas realizar al detectar la inclinación.

Resultados

Presentación de los resultados obtenidos

Los resultados del proyecto se demostrarán el día Miércoles 29/11, a través de una presentación en clases.

Datos cuantitativos o cualitativos relevantes

Conclusiones

Podemos concluir que el proyecto desarrollado fue una forma entretenida de llevar a cabo un proyecto de la vida real, denotado en un proyecto simple.

Descubrimos la importancia que conllevan los proyectos desarrollados en Arduino para el progreso y avance de la sociedad hacia practicas mas controladas, optimizadas y que tienen relación con el cuidado de los proyectos y las personas, como son en los casos de esos proyectos de medición de temperatura.