



UNIVERSIDAD AUTÓNOMA DE SINALOA
FACULTAD DE INGENIERÍA LOS MOCHIS
Ingeniería de software



MATERIA: Computación Ubicua

Sistema de alarma

PROFESOR(A): Mirsa Paolo Inzunza Martinez

EQUIPO:

- Reyna Yanira Quiñonez Villegas
- Jesus Martin Reyes Hernandez
- Jesus Antonio Bojorquez Espinoza

GRUPO: 4-02

FECHA: 01/Junio/2023

Introducción

En la actualidad, la experimentación con dispositivos electrónicos ha sido simplificada en comparación con años anteriores. Gracias a la evolución de los componentes y a la reducción de sus costos, ahora podemos presentar este proyecto.

Todos compartimos la necesidad de protegernos contra intrusos, y qué mejor manera de hacerlo que con una alarma que se active en su presencia y que no pueda ser desactivada por un ladrón. Es imprescindible que todos tengamos una alarma antirrobo en nuestros hogares, y lo bueno es que el circuito es sencillo y accesible para todos.

En este proyecto, se presentará el diseño de un circuito que cumple con esta necesidad. En primer lugar, se realizará un análisis detallado de los componentes utilizados en su desarrollo y construcción, estudiando su funcionamiento para lograr una aplicación óptima.

De esta manera, en poco tiempo estará listo para instalar una alarma en su propia casa, sin complicaciones.

Contenido

Introducción	2
Antecedentes.....	4
Delimitación del problema.....	5
Justificación	5
Objetivos.....	6
Objetivo general	6
Objetivos específicos	6
Herramientas	7
ESP32.....	7
Protoboard	8
Sensores.....	8
Sensor de ruptura	8
Sensor de movimiento	9
Procedimiento.....	10
Código	12
Gestión del proyecto.....	14
Gestión de costos	14
Tiempo	14
Costos	14
Gestión de riesgo	15
Conclusión.....	16
Referencias	17

Antecedentes

A lo largo de la historia, la seguridad ha sido de suma importancia en la sociedad debido a la necesidad de proteger información y preservar intereses; por esta razón, las empresas de seguridad ofrecen diversas soluciones diseñadas para proporcionar una sensación de seguridad.

Los sistemas de seguridad en el hogar se han implementado desde hace mucho tiempo para proteger la integridad física, la información y los objetos de valor. Se han creado una variedad de sistemas con características especiales que cubren las necesidades de los usuarios y se adaptan a las nuevas tecnologías. Estos sistemas han incorporado diferentes dispositivos, como sensores electrónicos (de humedad, temperatura, iluminación, entre otros), y han evolucionado en la señal de alerta, incluyendo señales sonoras y activación de luces.

A lo largo del tiempo, los sistemas de alarmas para la seguridad han experimentado cambios significativos debido al avance de la tecnología. Los primeros sistemas electrónicos solo emitían una alerta audible, pero luego se incorporaron más sensores y controles remotos utilizando infrarrojos o Bluetooth, con una cobertura limitada de 10 metros. Posteriormente, se agregaron sensores de humedad, temperatura y notificaciones mediante llamadas telefónicas a números fijos, aunque sin control remoto. A medida que los sistemas evolucionaron, los usuarios comenzaron a seleccionar los sensores según sus necesidades, y se introdujo la comunicación por radiofrecuencia con una cobertura de hasta 500 metros. Por último, los sistemas más modernos notifican a los teléfonos móviles mediante Wi-Fi o GSM, pero solo permiten la notificación de la alerta, sin posibilidad de control remoto.

Delimitación del problema

La inseguridad en los hogares es un problema persistente y ha sido uno de los principales desencadenantes de la delincuencia, resultando en hurtos y robos. Los sistemas de alarmas tradicionales suelen ser costosos y requieren de instalaciones complicadas. En este contexto, se plantea el problema de desarrollar un sistema de alarma que sea económico, fácil de implementar y personalizable.

Justificación

Mediante la implementación de este proyecto, se pretende contribuir a la protección de nuestros bienes, ya sean viviendas, vehículos, oficinas o simplemente utilizarlo en nuestro propio beneficio.

Gracias a su sencillo ensamblaje y funcionamiento, cualquier individuo puede realizar este dispositivo, y puede ser utilizado de diversas maneras en ventanas, puertas e incluso instalando sensores en cerraduras y debajo de alfombras. De esta manera, cualquier movimiento, sin importar su brusquedad, activará la alarma de manera inmediata y no podrá ser desactivada por un intruso, asegurándonos de que la alarma permanezca activa.

Objetivos

Objetivo general

Diseñar e implementar un sistema de seguridad utilizando un microcontrolador ESP32 y sensores para el monitoreo de movimiento y vibración.

Objetivos específicos

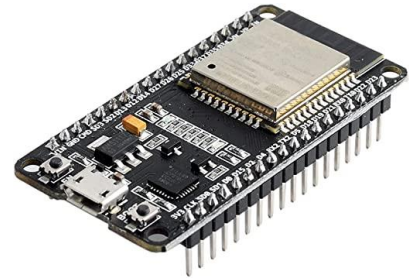
- Seleccionar y conectar sensores adecuados, como sensores de movimiento, sensores de puerta/ventana, sensores de humo, etc., al ESP32 para detectar eventos o situaciones específicas.
- Establecer una comunicación efectiva entre la alarma y otros dispositivos, como teléfonos móviles o servidores en la nube, para enviar notificaciones en caso de eventos de alarma.
- Permitir la configuración y el control remoto de la alarma a través de una interfaz en línea o una aplicación móvil, lo que brinda flexibilidad y comodidad al usuario.
- Diseñar y desarrollar una interfaz de usuario amigable y fácil de usar que permita a los usuarios interactuar con la alarma, configurar ajustes, visualizar registros de eventos, etc.

Herramientas

En este capítulo se abordan las tecnologías empleadas en cada componente del proyecto.

ESP32

El ESP32 es un microcontrolador de bajo costo y bajo consumo de energía que combina un procesador de aplicaciones de doble núcleo y conectividad Wi-Fi y Bluetooth integrada. Es ampliamente utilizado en el desarrollo de proyectos de Internet de las cosas (IoT), debido a su versatilidad y capacidad de conectividad.

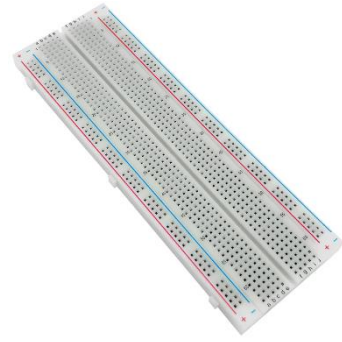


Características

- **Conectividad Wi-Fi y Bluetooth:** Permite la comunicación inalámbrica con otros dispositivos y la conexión a redes Wi-Fi para acceder a Internet.
- **Procesamiento de alto rendimiento:** Es capaz de manejar tareas complejas y exigentes en términos de procesamiento.
- **Amplia gama de puertos y periféricos:** Permite la conexión y control de diversos dispositivos externos, como sensores, pantallas y actuadores.
- **Capacidad de almacenamiento:** Permite el almacenamiento de programas, configuraciones y otros datos relevantes para el funcionamiento del dispositivo.
- **Bajo consumo de energía:** Adecuado para aplicaciones con restricciones de energía, como dispositivos portátiles y sistemas de bajo consumo.

Protoboard

Un protoboard, también conocido como placa de pruebas o breadboard en inglés, es una herramienta utilizada en electrónica y prototipado para realizar conexiones temporales entre componentes electrónicos. Consiste en una placa rectangular con una matriz de orificios y contactos metálicos interconectados debajo de la superficie. Facilita la creación rápida y flexible de circuitos experimentales o prototipos y es ampliamente utilizado por estudiantes, aficionados y profesionales en el campo de la electrónica y el desarrollo de proyectos.



Sensores

Sensor de ruptura

Un sensor de ruptura, también conocido como sensor de vibración, es un dispositivo utilizado para detectar y alertar sobre la ruptura o manipulación de objetos, superficies o estructuras. Estos sensores están diseñados para detectar vibraciones o cambios bruscos en la posición de un objeto, lo cual indica que ha ocurrido una ruptura o interferencia no deseada.



Funcionamiento

El funcionamiento básico de un sensor de ruptura puede variar dependiendo del tipo y la tecnología utilizada.

- Detección de vibraciones generadas por golpes o impactos en un objeto o superficie.
- Interruptores magnéticos o láseres que detectan cambios en el campo magnético o la interrupción de un haz de luz, respectivamente, cuando se produce una ruptura.

Estos sensores desempeñan un papel importante en la seguridad y protección de propiedades, así como en la monitorización de equipos industriales.

Sensor de movimiento

Un sensor de movimiento, también conocido como sensor de presencia o sensor PIR (infrarrojo pasivo), es un dispositivo electrónico diseñado para detectar la presencia de movimiento en su entorno. Estos sensores son capaces de detectar cambios en el nivel de radiación infrarroja emitida por los objetos en su campo de visión.

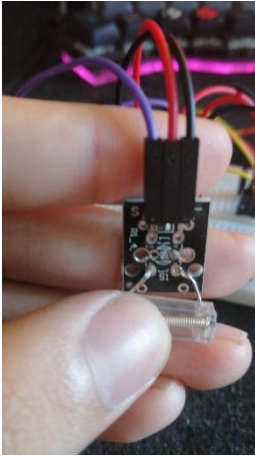


Funcionamiento

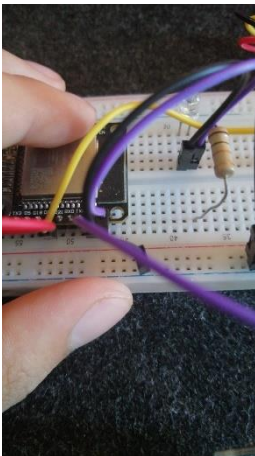
El funcionamiento básico de un sensor de movimiento se basa en la detección de cambios en el patrón de radiación infrarroja. Cuando una persona u objeto se mueve dentro del rango de detección del sensor, este capta la variación en la radiación infrarroja y genera una señal eléctrica. Esta señal se utiliza para activar un sistema de alarma, encender luces u otros dispositivos, o desencadenar una acción específica, dependiendo de la aplicación del sensor.

Procedimiento

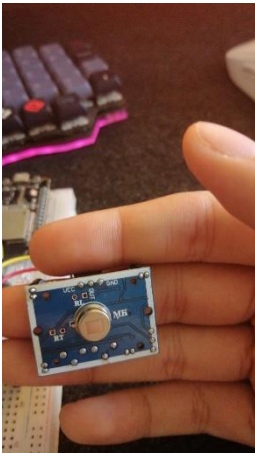
Conectamos el sensor de impacto, el pin de marcado con s es el de datos, el de la derecha es el negativo y el que sobra es de corriente.



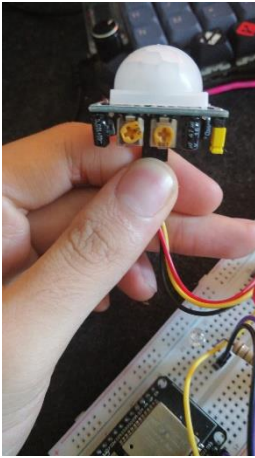
En este caso lo conecte al pin 22.



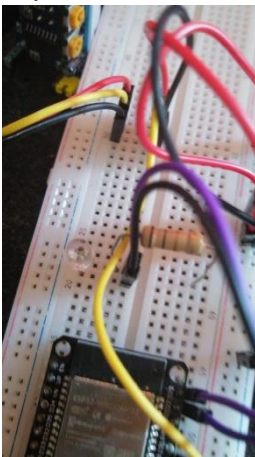
Para conectar el sensor de movimiento hay que quitar la capsula por que varía el cómo están organizados los pines.



Después puedes calibrarlo, la tuerquita de la izquierda es para el ancho y el de la derecha es el largo.



Y para simular una alarma puse un foquito con una resistencia.



Código

```
#include <Arduino.h>
#include <WiFi.h>
#include <HTTPClient.h>
#include <ESPAsyncWebServer.h>
#include <UniversalTelegramBot.h>
#include <CTBot.h>

const char* ssid = "Totalplay_2.4Gnormal";
const char* pssw = "x2BV4K2910";

const char* chat_id = "1994205600";
const int chat = 1994205600;
const char* token = "5624223237:AAGZ4pVhsqI-14F_taxVJhhJDLXCIn2B_Y0";

CTBot mybot;

AsyncWebServer server(8080);

#define LED 23
#define SENSOR 19

void encenderAlaramaMovimiento();
void enviarMensaje(char* mensaje);

void encenderAlaramaMovimiento(){
  char* alarmaMovimientoMsj = "Movimiento detectado";
  digitalWrite(LED, HIGH);
  enviarMensaje(alarmaMovimientoMsj);
}

void enviarMensaje(char* mensaje){
  mybot.sendMessage(chat,mensaje);
}

void setup() {
  Serial.begin(9600);
  pinMode(SENSOR,INPUT);
  pinMode(LED,OUTPUT);
  WiFi.begin(ssid,pssw);
  while(WiFi.status()!= WL_CONNECTED){
```

```

    delay(1000);
    Serial.print("Conectando ....");
}
Serial.print("Conexion Wifi iniciado");

mybot.wifiConnect(ssid,pssw);
mybot.setTelegramToken(token);

server.on("/apagarMovimiento", HTTP_GET,[](AsyncWebServerRequest *request){
    digitalWrite(LED,LOW);
    mybot.sendMessage(chat, "Alarama apagada");
    request->send(200,"text/plain","alarma apagada");
});

server.begin();
Serial.print("Servidor iniciado en:");
IPAddress ip = WiFi.localIP();
Serial.print(ip);
}

void loop() {
    int estado = digitalRead(SENSOR);

    if(estado == HIGH){
        encenderAlaramaMovimiento();
    }
}

```

Gestión del proyecto

En el presente capítulo, se abordarán todos los elementos vinculados a la administración del proyecto. En primer término, se detallarán los tiempos, los costos asociados al proyecto y se expondrá la gestión de los riesgos afrontados durante el desarrollo del proyecto.

Gestión de costos

Tiempo

Tarea	Estimación	Desviación
Programación del código	5 horas	8 horas
Armar el circuito	8 horas	10 horas
Testear y corregir el sistema	8 horas	4 horas

Lo que supone un total de 22 horas, si tenemos en cuenta la hora no prevista inicialmente.

Costos

Elemento	Unidades	Costo
Esp32	1	158
Sensor de movimiento	1	175
Sensor de ruptura	1	125
Protoboard	1	126
Total	4	584

Gestión de riesgo

- **Riesgo de fallos en el hardware:** Los componentes del ESP32, como sensores, actuadores o módulos de comunicación, pueden presentar fallos o mal funcionamiento. Para mitigar este riesgo, se deben utilizar componentes de calidad de proveedores confiables y realizar pruebas exhaustivas antes de implementar la alarma.
- **Riesgo de seguridad de red:** Al conectar la alarma a una red Wi-Fi, existe el riesgo de que terceros no autorizados puedan acceder a la red o interceptar la comunicación. Para gestionar este riesgo, se deben implementar medidas de seguridad, como el uso de contraseñas fuertes, encriptación de datos y configuraciones adecuadas de seguridad en la red Wi-Fi.
- **Riesgo de interferencias electromagnéticas:** La presencia de otros dispositivos electrónicos cercanos puede causar interferencias electromagnéticas y afectar el rendimiento de la alarma. Para abordar este riesgo, se pueden utilizar componentes con buena capacidad de blindaje y realizar pruebas en entornos reales para identificar y solucionar posibles interferencias.
- **Riesgo de falta de actualizaciones y parches de seguridad:** El ESP32 utiliza firmware y software que pueden requerir actualizaciones periódicas para corregir errores y vulnerabilidades de seguridad. Es importante mantenerse al día con las actualizaciones y parches proporcionados por el fabricante para mitigar riesgos de seguridad.
- **Riesgo de errores en la programación:** Al desarrollar el software para la alarma, existe el riesgo de introducir errores o fallas en la programación. Para minimizar este riesgo, se deben seguir buenas prácticas de programación, realizar pruebas exhaustivas y utilizar técnicas de depuración para identificar y corregir posibles errores.

Conclusión

La creación de un sistema de alarma utilizando Arduino es una opción versátil y accesible para la seguridad y protección de diversos entornos. Mediante la combinación de sensores, actuadores y el poder de programación de Arduino, es posible desarrollar un sistema personalizado y adaptado a las necesidades específicas de cada usuario.

Aunque la creación de un sistema de alarma puede requerir ciertos conocimientos técnicos, existen numerosos recursos y tutoriales disponibles en línea para guiar el proceso de desarrollo. Además, en internet existe una comunidad activa que comparte proyectos y soluciones, lo que facilita el aprendizaje y la resolución de problemas.

La creación de un sistema de alarma brinda la oportunidad de diseñar una solución personalizada, económica y eficiente para garantizar la seguridad y protección de hogares, oficinas u otros entornos. Con un enfoque adecuado en el diseño, implementación y pruebas, es posible aprovechar al máximo las capacidades de Arduino y crear un sistema de alarma confiable y efectivo.

Referencias

- <https://homego.es/blog/ventajas-alarma-detector-rotura-cristal/>
- <https://tesisciencia.com/2019/09/23/recomendaciones-para-redactar-los-antecedentes-de-una-investigacion-o-tesis/>
- <https://es.scribd.com/document/353547329/circuito-electronico-de-alarma-de-seguridad>
- <http://repository.unipiloto.edu.co/bitstream/handle/20.500.12277/6471/DISE%C3%91O%20DE%20UN%20SISTEMA%20%20PARA%20CONTROL%20DE%20LAS%20ALARMAS%20%20DE%20SEGURIDAD%20EN%20EL%20HOGAR%20UTILIZANDO%20LA%20TECNOLOG%C3%8DA%20M2M.pdf?sequence=1&isAllowed=y>
- <https://dspace.ups.edu.ec/bitstream/123456789/10401/1/UPS-GT001444.pdf>
- <https://addi.ehu.es/bitstream/handle/10810/13331/PFC.pdf?sequence=2#page=102&zoom=100,109,740>