



## Actividad 2 Alarma para Incendios

### Internet de las Cosas

---

Ingeniería en Desarrollo de Software



TUTOR: Marco Alonso Rodríguez

---

ALUMNO: Homero Ramirez Hurtado

---

FECHA: 23 de Junio del 2025

---

Índice.

. Introducción.

. Descripción.

. Justificación.

. Desarrollo.

- Armado del Circuito.
  - Codificación.
- Emulación del Circuito.

. Conclusión.

. Referencias.

## Introducción.

La presente emulación tiene como objetivo el desarrollo de un sistema de alarma de incendios utilizando la plataforma de simulación Tinkercad y un microcontrolador Arduino. Este proyecto surge como una propuesta práctica para entender el funcionamiento de sensores de gas aplicados a la detección temprana de incendios, un componente esencial en la seguridad de espacios residenciales e industriales.

Durante la simulación, se implementará un circuito que incluye un sensor de gas el cual estará conectado a un Arduino. A través del monitoreo constante de estos parámetros, el sistema será capaz de activar una alarma sonora.

La elección de Tinkercad como herramienta de emulación permite no solo validar el diseño sin necesidad de componentes físicos, sino también fomentar la comprensión de los fundamentos de la programación y la electrónica. Este proyecto busca ofrecer una experiencia educativa completa al simular una aplicación concreta de la tecnología Arduino en la prevención de riesgos.

## Descripción.

Este proyecto de emulación en Tinkercad tiene como propósito principal aprender a diseñar e implementar un sistema de detección de gases inflamables utilizando un microcontrolador Arduino y un sensor de gas. Mediante la simulación, se busca comprender el funcionamiento básico de sensores analógicos, enfocándose en cómo estos generan señales proporcionales a la concentración de gas en el ambiente y cómo el Arduino interpreta esas señales para generar una respuesta automatizada.

La experiencia permitirá al estudiante desarrollar habilidades en el diseño de circuitos electrónicos simples, la escritura de código en lenguaje Arduino y la implementación de condiciones lógicas que determinen cuándo activar una alarma sonora. Además, se fomentará la capacidad de verificar el comportamiento del sistema en escenarios simulados de riesgo, fortaleciendo así la comprensión del ciclo de lectura, decisión y acción en un sistema embebido.

En términos prácticos, se pretende que podamos analizar la importancia de estos sistemas en la protección de vidas y bienes, y que reconozca cómo la tecnología puede ofrecer soluciones accesibles y efectivas en entornos reales.

## Justificación.

El uso de Tinkercad para esta práctica representa una herramienta de aprendizaje poderosa, especialmente en el contexto de la educación técnica y tecnológica. Esta plataforma permite simular circuitos eléctricos y programar microcontroladores como Arduino de manera virtual, sin necesidad de contar con los componentes físicos, lo cual resulta ideal para estudiantes que están dando sus primeros pasos en la electrónica.

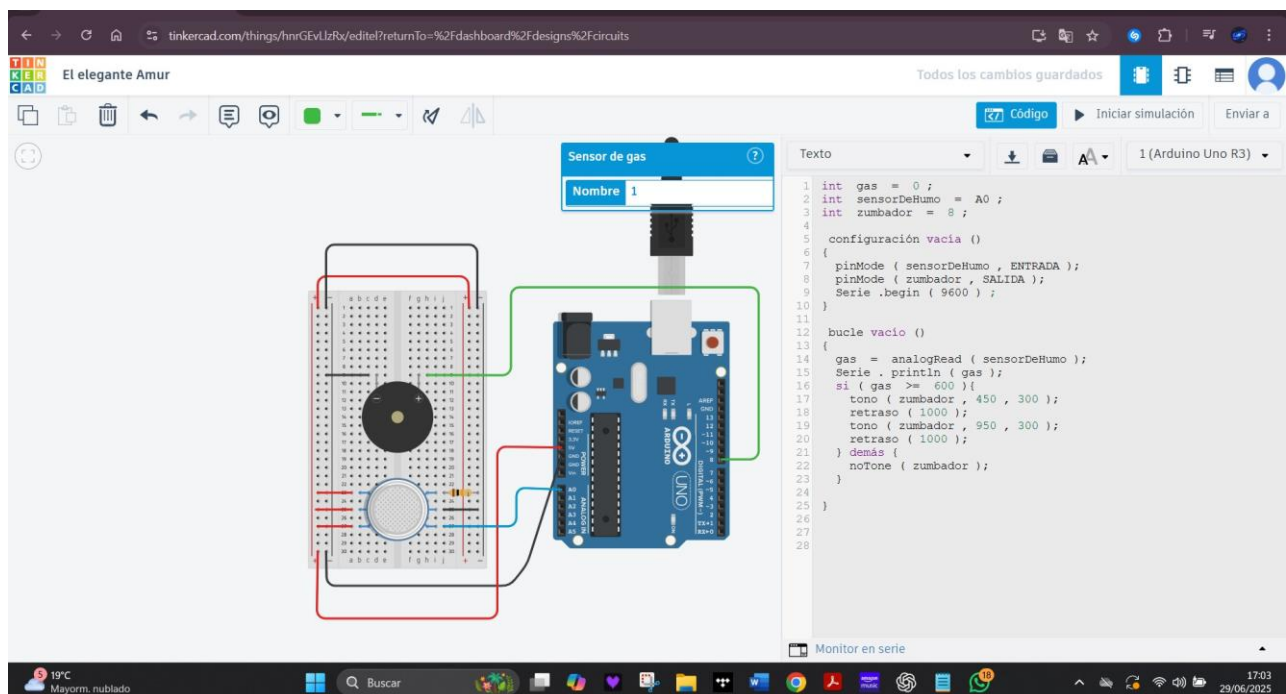
Una de las principales ventajas de Tinkercad es su interfaz intuitiva, que facilita la comprensión de conceptos complejos mediante la manipulación directa de elementos del circuito. Además, brinda retroalimentación inmediata a través de simulaciones en tiempo real, lo que permite identificar errores, ajustar el código y experimentar con diferentes configuraciones sin ningún riesgo físico ni económico.

En el caso específico de este proyecto una alarma de incendios basada en un sensor de gas Tinkercad ofrece un entorno seguro y accesible para ensayar la lectura de datos analógicos, aplicar estructuras condicionales en la lógica del programa y verificar el funcionamiento de una respuesta automática mediante una alarma sonora.

En resumen, Tinkercad no solo refuerza el aprendizaje práctico y la resolución de problemas, sino que también fomenta la experimentación creativa, el pensamiento lógico y la autonomía en el desarrollo de soluciones tecnológicas.

## Desarrollo.

### Armado del Circuito.



Codificacion.

```
int gas = 0;
```

```
int smokeSensor = A0;
```

```
int buzzer = 8;
```

```
void setup()
```

```
{
```

```
  pinMode(smokeSensor, INPUT);
```

```
  pinMode(buzzer, OUTPUT);
```

```
  Serial.begin(9600);
```

```
}
```

```
void loop()
```

```
{
```

```
  gas = analogRead(smokeSensor);
```

```
  Serial.println(gas);
```

```
  if (gas >= 600){
```

```
    tone (buzzer, 450,300);
```

```
    delay (1000);
```

```
    tone (buzzer, 950,300);
```

```
    delay (1000);
```

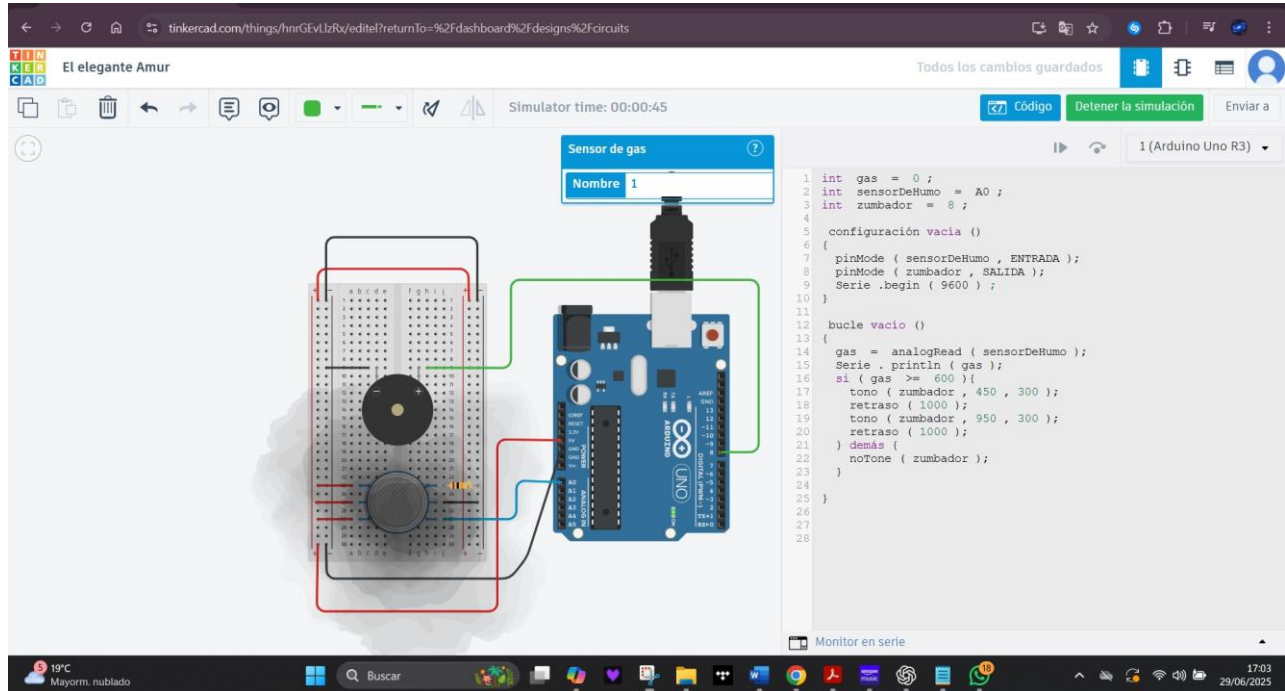
```
  } else {
```

```
    noTone(buzzer);
```

```
  }
```

```
}
```

## Emulación del Circuito.



## Conclusión.

Este proyecto ejemplifica de manera práctica los fundamentos de la programación en Arduino y la integración de sensores para generar una alerta temprana ante niveles críticos de gas. Partiendo de la declaración de variables enteras que representan el pin analógico del sensor de gas y el pin digital del buzzer, aprendimos a configurar cada componente en el `setup()` mediante `pinMode()`, definiendo entradas y salidas de forma clara. En el `loop()`, la función `analogRead()` permitió medir continuamente la concentración de gas, mientras que la estructura condicional `if (valorGas >= 600)` actuó como umbral: al superarse, el buzzer emite un sonido de alerta; de lo contrario, permanece apagado.

Este enfoque modular y basado en variables favorece la legibilidad y facilita ajustes posteriores, como modificar el nivel de sensibilidad o cambiar el pin asignado al buzzer. Además, el uso del monitor serial ofrece retroalimentación en tiempo real, esencial para calibrar el sensor y comprobar su precisión antes de desplegar el sistema en escenarios reales.

## Referencias.

Video Tutoría 2.

Copilot.

ChatGPT.