

Actividad | 2 | Alarma para

Incendios

Internet de las Cosas

Ingeniería en Desarrollo de Software



TUTOR: Marco Alonso Rodríguez Tapia

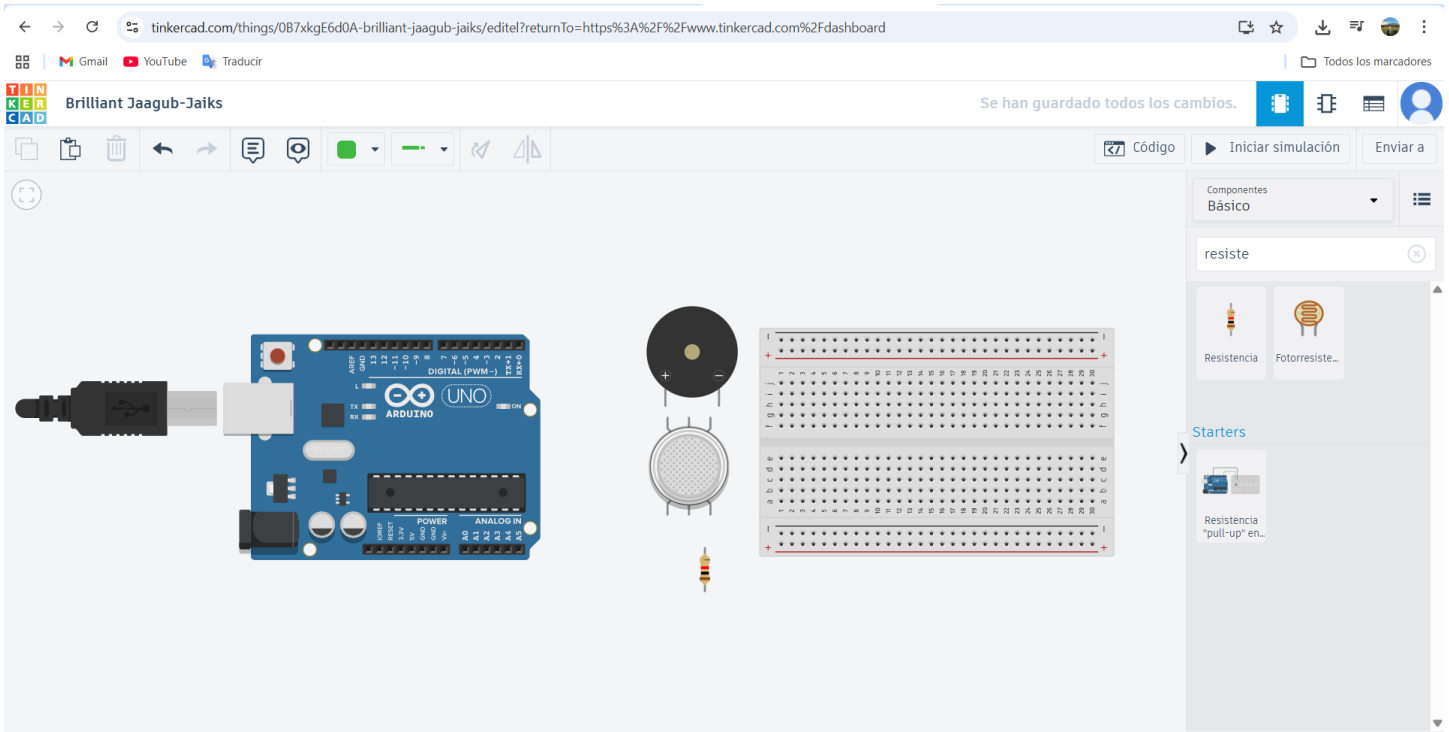
ALUMNO: Eduardo Monroy Hernández

FECHA: 21-10-2025

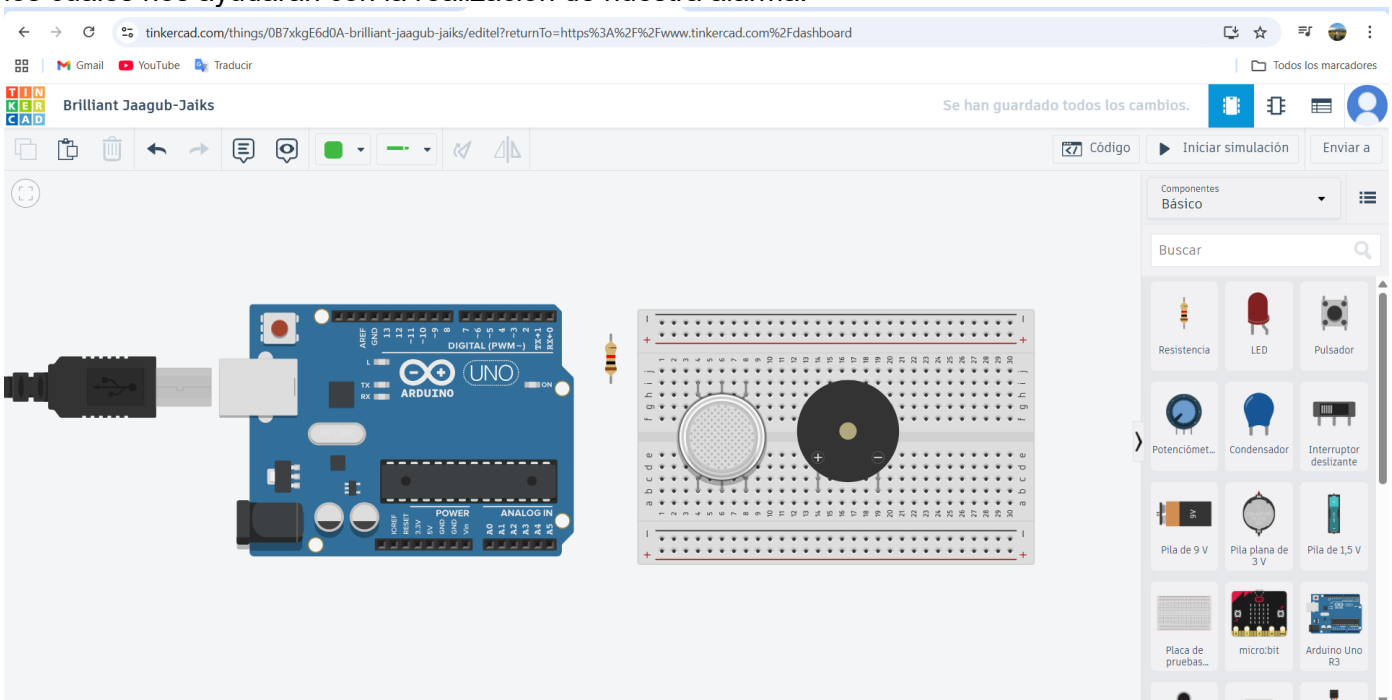
Armado del circuito:

Comenzaremos con nuestro armado de circuito en donde tenemos como actividad generar una alarma para incendios dentro de Tinkercad en donde mostraremos el proceso paso a paso de la elaboración de esta actividad.

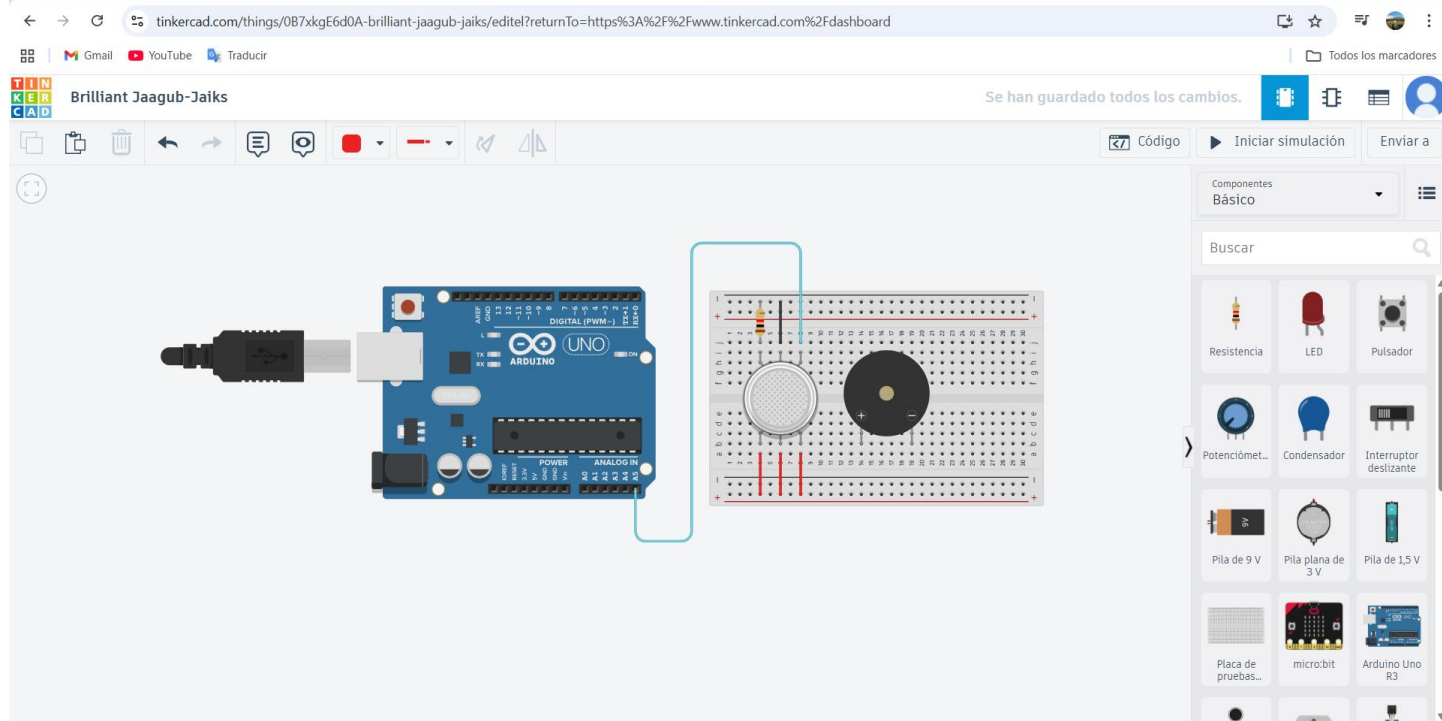
Primeramente, generaremos un nuevo proyecto, donde agregaremos un Arduino, una placa de pruebas, un sensor de gas, un piezo y una resistencia así agregando todos los componentes para nuestra alarma.



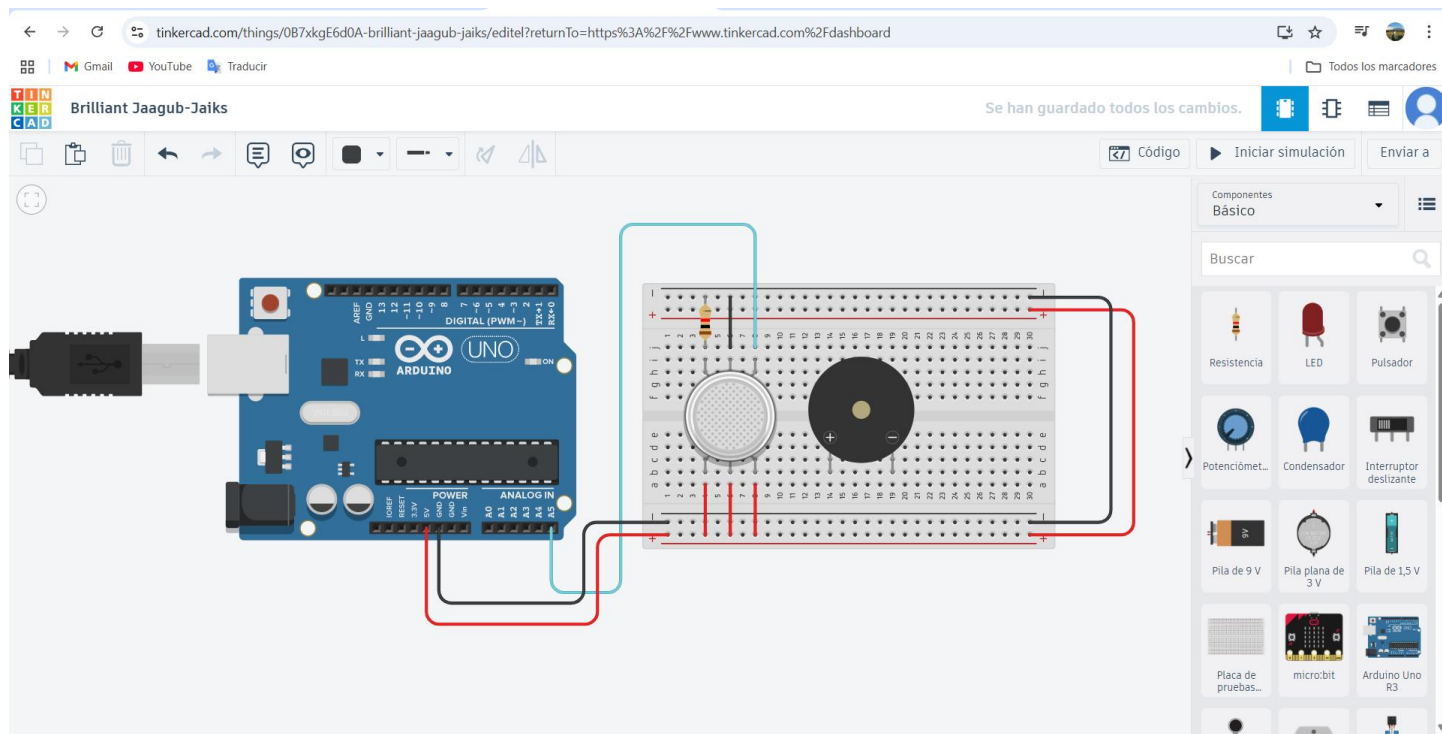
Continuaremos realizando la conexión de nuestros componentes de igual manera dando valores a componentes los cuales nos ayudaran con la realización de nuestra alarma.



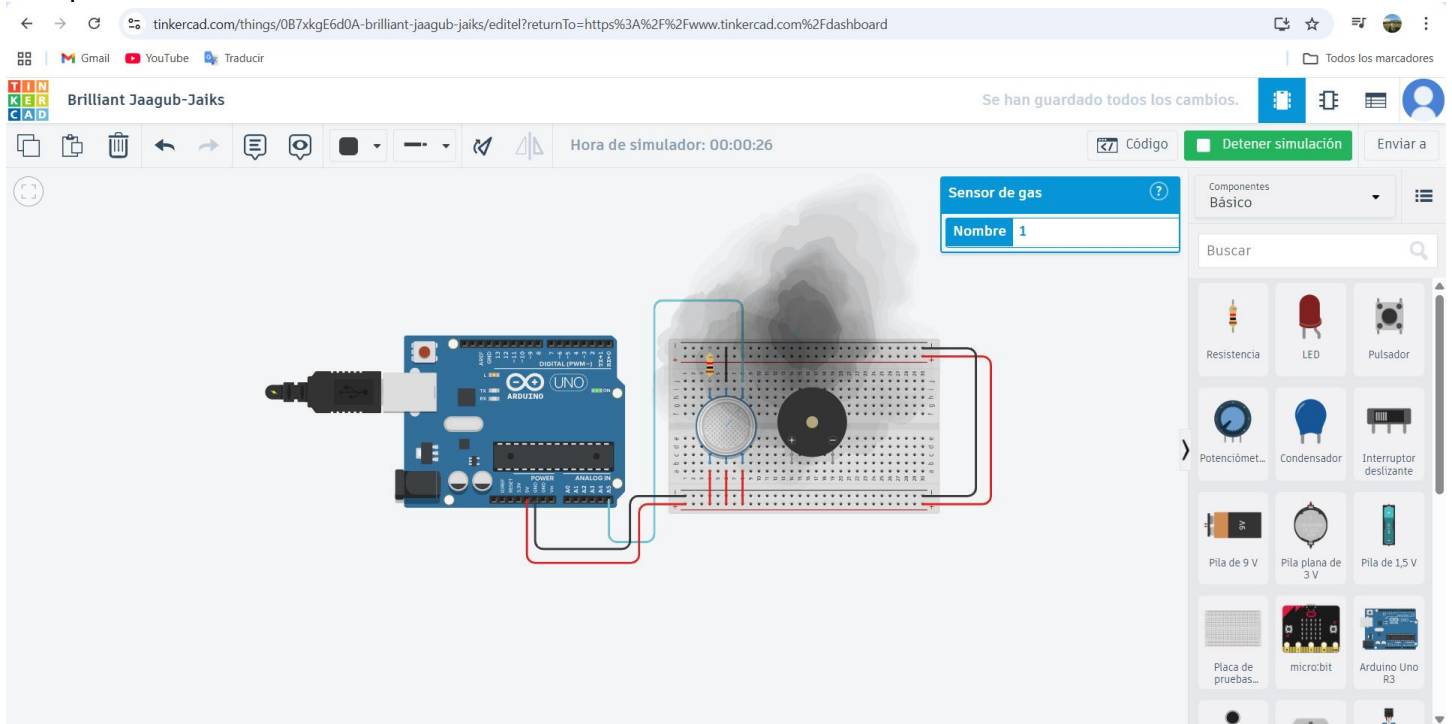
Continuaremos realizando la conexión a tierra donde de igual modo conectaremos la resistencia, realizaremos una conexión a un puerto del Arduino en este caso será el puerto A5 conectando al sensor de gas, este sensor proporcionará información de entrada a nuestra placa Arduino, para después conectar el sensor a voltaje.



Ahora para que nuestra placa tenga comunicación tendremos que limitarla a 5V, de igual manera realizaremos un puenteo del lado superior de la placa, para que esta tenga voltaje de igual manera. Para este caso realizamos de igual manera la conexión de GND (Tierra).

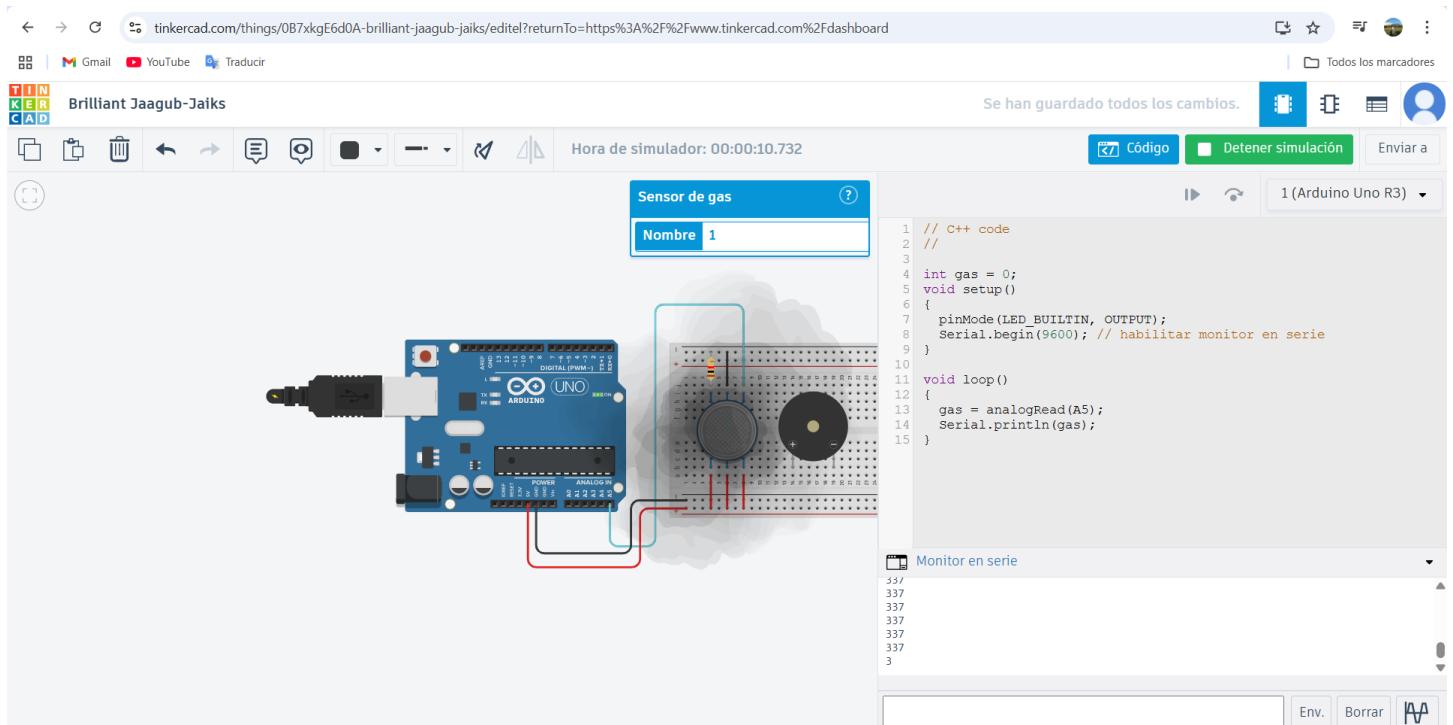


Iniciamos una simulación en donde al ejecutar en nuestro sensor aparece una nube donde igual podemos manipular.



Codificación:

En esta parte de nuestra actividad continuaremos con la codificación, agregaremos una variable de tipo entero, agregando valores a nuestro código como habilitar un monitor en serie.



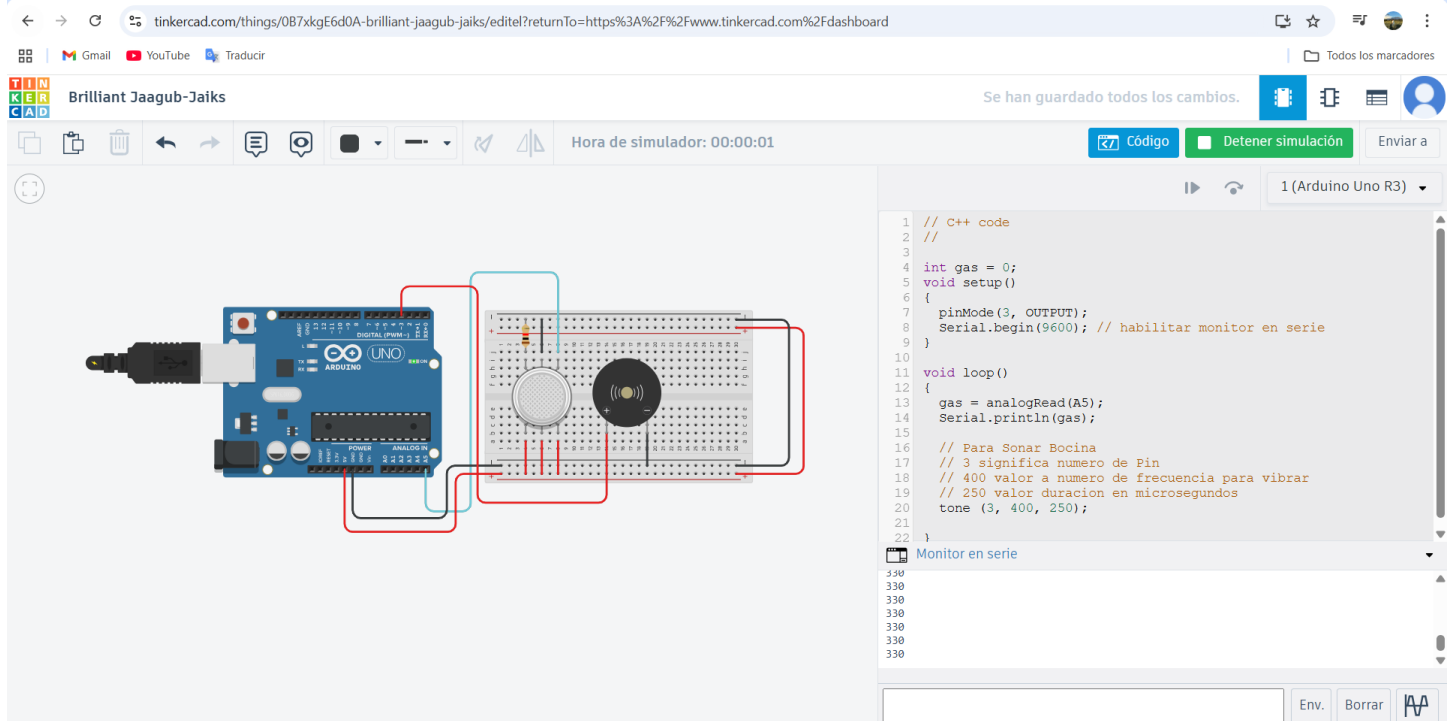
The screenshot shows the Tinkercad web interface. On the left, an Arduino Uno is connected to a breadboard. A gas sensor is placed on the breadboard, with its pins connected to the Arduino. A label "Sensor de gas" with "Nombre 1" is positioned above the sensor. The code editor on the right shows the following code:

```
1 // C++ code
2 //
3
4 int gas = 0;
5 void setup()
6 {
7   pinMode(LED_BUILTIN, OUTPUT);
8   Serial.begin(9600); // habilitar monitor en serie
9 }
10
11 void loop()
12 {
13   gas = analogRead(A5);
14   Serial.println(gas);
15 }
```

Below the code editor, there is a "Monitor en serie" section with a scrollable list of output values: 337, 337, 337, 337, 337, 337, 3.

Continuaremos conectando nuestra bocina en el caso del lado positivo la conectaremos a un puerto en este caso será 3 y del lado negativo conectaremos a GND, una vez realizando la conexión continuaremos con la codificación.

Se ejecuta código y se muestra simulación de bocina sonando.



The screenshot shows the Tinkercad web interface with the same setup as before, but with a buzzer added to the breadboard. The buzzer is connected to pin 3 of the Arduino and GND. The code editor on the right shows the following code:

```
1 // C++ code
2 //
3
4 int gas = 0;
5 void setup()
6 {
7   pinMode(3, OUTPUT);
8   Serial.begin(9600); // habilitar monitor en serie
9 }
10
11 void loop()
12 {
13   gas = analogRead(A5);
14   Serial.println(gas);
15
16   // Para Sonar Bocina
17   // 3 significa numero de Pin
18   // 400 valor a numero de frecuencia para vibrar
19   // 250 valor duracion en microsegundos
20   tone (3, 400, 250);
21 }
22
```

Below the code editor, there is a "Monitor en serie" section with a scrollable list of output values: 330, 330, 330, 330, 330, 330, 330.

Posteriormente se continua con la generación de esta alarma, agregando especificaciones para que esta pueda sonar, se agrega evidencia de alarma sonando bajo la condición que se le especifico.

Alarma sin sonar:

The screenshot shows the Tinkercad web interface. On the left, an Arduino Uno R3 is connected to a breadboard. A gas sensor module is connected to the breadboard, and a speaker is also connected. The gas sensor is labeled 'Sensor de gas' with 'Nombre 1'. The code on the right is in C++ and includes a condition 'if (gas > 350)' to trigger a tone. The simulation time is 00:00:04.

```
1 // C++ code
2 //
3
4 int gas = 0;
5 void setup()
6 {
7   pinMode(3, OUTPUT);
8   Serial.begin(9600); // habilitar monitor en serie
9 }
10 void loop()
11 {
12   gas = analogRead(A5);
13   Serial.println(gas);
14   // Hacer sonar Bocina bajo condiciones
15   if (gas > 350 ) {
16     // Para Sonar Bocina
17     // 3 significa numero de Pin
18     // 400 valor a numero de frecuencia para vibrar
19     // 250 valor duracion en microsegundos
20     tone (3, 400, 250); } else {
21     noTone (3);}
22     // Apagar Led
23 }
```

Alarma sonando:

The screenshot shows the Tinkercad web interface. On the left, the same setup as the previous image is shown, but the speaker icon now has sound waves, indicating it is playing a tone. The code on the right is the same as the previous image. The simulation time is 00:00:15.

```
1 // C++ code
2 //
3
4 int gas = 0;
5 void setup()
6 {
7   pinMode(3, OUTPUT);
8   Serial.begin(9600); // habilitar monitor en serie
9 }
10 void loop()
11 {
12   gas = analogRead(A5);
13   Serial.println(gas);
14   // Hacer sonar Bocina bajo condiciones
15   if (gas > 350 ) {
16     // Para Sonar Bocina
17     // 3 significa numero de Pin
18     // 400 valor a numero de frecuencia para vibrar
19     // 250 valor duracion en microsegundos
20     tone (3, 400, 250); } else {
21     noTone (3);}
22     // Apagar Led
23 }
```

REPOSITORIO:

<https://github.com/monroy2321eduardo-sudo/Internet-De-La-Cosas->

ENLACE TINKERCAD:

<https://www.tinkercad.com/things/0B7xkgE6d0A-brilliant-jaagub-jaiks/editel?returnTo=https%3A%2F%2Fwww.tinkercad.com%2Fdashboard>
<https://www.tinkercad.com/things/0B7xkgE6d0A-brilliant-jaagub-jaiks>

Conclusión:

La actividad Alarma contra incendios ayudó a comprender cómo funciona el Internet de las Cosas aplicándolo en un proyecto real con Arduino y diferentes sensores. Al realizar el circuito en la plataforma Tinkercad, se pudo observar cómo un sensor de gas detecta la presencia de humo o gases peligrosos y activa una bocina o alarma mediante un sensor. Esto permitió entender la importancia de la tecnología en la prevención de accidentes y en la creación de sistemas que protegen la vida y los espacios.

Durante el proceso se aprendió a identificar y conectar correctamente los componentes electrónicos como el sensor de gas, los LED, las resistencias y la bocina. También se reforzaron los conocimientos sobre programación básica, utilizando las funciones principales del lenguaje Arduino, como la creación de variables, para establecer condiciones. Todo esto ayudó a darle funcionalidad al circuito y comprobar cómo las instrucciones se transforman en acciones reales dentro del proyecto.

Además, se comprendió que los sensores juegan un papel muy importante en los sistemas modernos, ya que son capaces de detectar cambios en el entorno y responder de manera automática. Gracias a esto, se pueden crear herramientas útiles para el hogar, la escuela o incluso en industrias donde se necesita prevenir riesgos.

En resumen, esta práctica permitió combinar conocimientos de electrónica y programación de una forma sencilla y práctica. También demostró que con materiales básicos y una buena lógica de programación es posible diseñar soluciones innovadoras y seguras. Finalmente, la experiencia dejó claro que la tecnología, cuando se usa correctamente, puede mejorar la calidad de vida y ayudar a construir entornos más seguros y preparados ante cualquier emergencia.