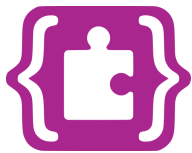


# HACER UN THEREMÍN

CON EL SENSOR DE DISTANCIA

#R1AS07



**Disponible en**



**Requisitos previos**

- R1AS04 - Sensor de luz básico
- R1AS06 - Código Morse

**Material**

- 1 placa de programación "**STM32 IoT Node Board**"
- Cable USB Micro-B
- 1 zumbador piezoeléctrico o un altavoz
- 1 placa de pruebas
- Cables de puente

**¿Qué es?**

Un theremín es un instrumento musical electrónico que puede tocarse sin necesidad de tocar propiamente el instrumento. El concepto original se basa en el uso de dos antenas para detectar la posición de las manos. Una antena se utiliza para el volumen, y la otra para el tono.

**Duración**

30 minutos

**Nivel de dificultad**

Avanzado

## OBJETIVOS DE APRENDIZAJE

- Utilizar un sensor de distancia y comprender su funcionamiento
- Hacer música con un instrumento realmente extraño
- Utiliza la función map para transformar un número de un rango a otro



# HACER UN THEREMÍN CON EL SENSOR DE DISTANCIA



El theremín es un instrumento musical electrónico controlado sin contacto físico por el/la thereminista (intérprete del theremín). Lleva el nombre de su inventor, Leon Theremin, que patentó el dispositivo en 1928. La sección de control del instrumento suele consistir en dos antenas metálicas que detectan la posición relativa de las manos del thereminista y controlan los osciladores de frecuencia con una mano y de amplitud (volumen) con la otra. Las señales eléctricas del theremín se amplifican y se envían a un altavoz.

Nuestra versión será más sencilla, sólo controlaremos el tono del sonido, con el sensor de distancia, el volumen estará predeterminado. **¡Vamos a hacer música!**

Fuente: <https://en.wikipedia.org/wiki/Theremin>, <https://youtu.be/x0NVb25p1oU>



## HAZLO



### Cableado del zumbador/altavoz

En teoría, un zumbador o un altavoz no está polarizado (significa que no hay "+" ni "-"), pero a menudo tiene un par de cables negro/rojo o signos ("+" y/o "-") en el dispositivo. Si se encuentra en esta configuración, conecta el cable del lado "+" del zumbador a **A0** y el otro a **GND**.

Si no hay color o indicación, basta con conectar un cable en **A0** y el otro en **GND**.

### Conecta la placa al ordenador

Con tu cable USB, conecta la placa a tu ordenador utilizando el conector **micro-USB ST-LINK** (en la esquina derecha de la placa). Si todo va bien deberías ver una nueva unidad en tu ordenador llamada **DIS\_L4IOT**. Esta unidad se utiliza para programar la placa simplemente copiando un archivo binario.

### Abre MakeCode

Ve al editor de **Let's STEAM MakeCode**. En la página de inicio, crea un nuevo proyecto haciendo clic en el botón "Nuevo proyecto". Dale un nombre a tu proyecto más expresivo que "Sin título" e inicia tu editor.

Fuente: [makecode.lets-steam.eu](https://makecode.lets-steam.eu)

### Programa tu placa

Dentro del Editor de Javascript de MakeCode, copia/pega el código disponible en la sección "**Prográmalo**" de abajo. Si no lo has hecho ya, da un nombre a tu proyecto y haz clic en el botón "**Descargar**". Copia el archivo binario en la unidad **DIS\_L4IOT**, espera a que la placa termine de parpadear y su programa estará listo.

### Ejecuta, modifica, juega

Tu programa se ejecutará automáticamente cada vez que lo guardes o reinicies tu placa (pulsas el botón etiquetado como RESET). Intenta entender el ejemplo y empieza a modificarlo.

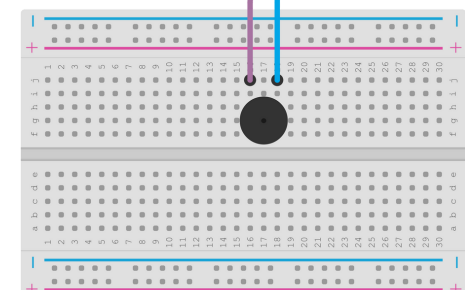
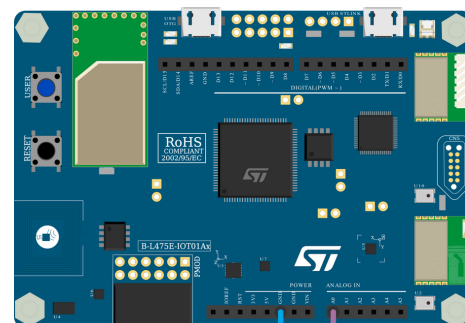
1

2

3

4

5



Cableado del zumbador/altavoz



## PROGRÁMALO

```
let distance = 0
forever(function () {
  // Obtener la distancia
  distance = input.distance(DistanceUnit.Millimeter)

  if (distance > 500) {
    // Convertir la distancia en frecuencia
    let note = Math.map(distance, 0, 500, 440, 830)
    music.ringTone(note)
  } else {
    music.stopAllSounds()
  }
})
```

### Variables

En este programa, hay 2 variables. La primera, la distancia - **distance** - se utiliza para mantener la misma distancia a través de la condición y para el tono de jugar. A continuación, se encuentra la nota - **note** - que no es técnicamente necesaria/obligatoria, pero ayuda a introducir una mayor comprensión de cada paso del programa. Contiene la transformación de la distancia en frecuencia del tono.

### Mide distancia

Utilizar una variable para mantener la distancia es genial, pero saber cómo conseguir la distancia es mejor. Una vez más, no hay ninguna dificultad. Tenemos que llamar a la función **input.distance(DistanceUnit.Millimeter)**. El parámetro **DistanceUnit.Millimeter** especifica a la función que queremos el resultado en milímetros (1 metro = 1.000 milímetros).

### Condición

La condición, **if (distance > 500) {**, da la información de que sólo reproducimos un sonido si la distancia medida es inferior o igual a 500 milímetros.

### Convertir la distancia en frecuencia

La parte más importante es la **conversión**. Para hacerla, utilizamos una función matemática llamada **map**. Esta función reasigna un valor de un rango a otro. En este caso, el valor se reasigna del rango de **distancia** al **rango de frecuencia**. Como puedes ver en el código anterior, esta función toma cinco parámetros, a saber: **map(valor, in\_min, in\_max, out\_min, out\_max)**. Veamos con más detalle cada uno de ellos:

- **valor**: el valor a reasignar
- **in\_min**: el valor mínimo del rango de entrada (distancia)
- **in\_max**: el valor máximo del rango de entrada (distancia)
- **out\_min**: el valor mínimo del rango de salida (frecuencia)
- **out\_max**: el valor máximo del rango de salida (frecuencia)

Por lo tanto, podemos entender lo que hace esta línea, es decir, reasignar la distancia (con un rango de 0 mm a 500 mm) a la frecuencia (con un rango de 440 Hz a 830 Hz).



**Las frecuencias elegidas no son aleatorias, el rango de frecuencia de 440Hz a 830Hz representa una octava. Esto significa que puedes encontrar todas las notas: LA, SI, DO, RE, MI, FA, SOL**

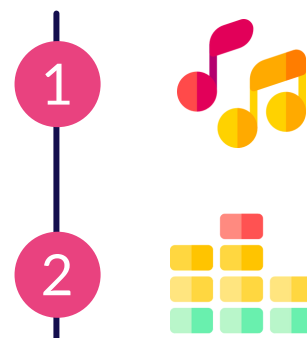
Ahora tenemos una frecuencia. Es el momento de reproducirla, simplemente utilizando el **music.ringTone(note)**.



## MEJÓRALO



Cambia el valor del **mapa** para añadir octavas y/o distancia para mejorar tu canción.

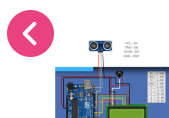


Intenta añadir un **potenciómetro** para controlar el volumen.

## ¿QUIERES IR MÁS ALLÁ?



- **Theremín** - Aprende más sobre la historia, los principios de funcionamiento y los usos del theremín.  
<https://en.wikipedia.org/wiki/Theremin>
- **Sensor de distancia de anillo LED** - Descubre un proyecto divertido, que terminará en un sensor de aparcamiento alternativo.  
<https://www.instructables.com/LED-Ring-Distance-Sensor/>
- **Detector de nivel de agua** - Descubre los sensores ultrasónicos que convierten la energía eléctrica en ondas acústicas.  
<https://www.instructables.com/Water-Level-Detector-2/>
- **Comedero para gatos** - Utiliza un sensor ultrasónico para construir un comedero automático para gatos.  
<https://www.instructables.com/Cat-Feeder/>



### Fichas de actividades enlazadas

**R1AS05 -  
Potenciómetro**

