# Project: Buzzer Beeping

For this project, we will use a **3-pin Active Buzzer** module with our **ESP32**. Since it's an active buzzer, it has an internal oscillator, meaning we only need to send a simple `HIGH` or `LOW` signal to make it beep.

## 1. Connections

The 3-pin buzzer module usually labels the signal pin as **S** (Signal) and the ground pin as - (minus sign). The middle pin is typically **VCC**.

| ESP32 Pin | Buzzer Module Pin | Description |
|---|---|---|
| **GPIO 13** | **S** (Signal) | Digital Control Pin |
| **3.3V** | **Middle Pin** | Power Supply |
| **GND** | **-** (Minus) | Ground |

## 2. Libraries that are needed to be downloaded

No external libraries required

## 3. Code

### *How it works*

The code configures GPIO 13 as an output. It uses a `HIGH` signal to complete the circuit inside the module, sounding the buzzer, and a `LOW` signal to turn it off. The `delay(1000)` pauses the program execution for exactly 1000 milliseconds (1 second).

### *Steps to connect*

1. Connect the pins as shown in the table above.
2. Plug your ESP32 into your computer.
3. Coyp-paste the code below in the IDE and click **Upload** (arrow sign at top left).

### *Code*

```
// a = Aim: Toggle buzzer on and off at 1-second intervals
// p = Primary Key (Not applicable for this hardware-only logic)

const int buzzerPin = 13; // Signal pin

void setup() {
  // Initialize the digital pin as an output
  pinMode(buzzerPin, OUTPUT);
}

void loop() {
  digitalWrite(buzzerPin, HIGH); // Turn buzzer ON
  delay(1000);                   // Wait for 1 second

  digitalWrite(buzzerPin, LOW);  // Turn buzzer OFF
  delay(1000);                   // Wait for 1 second
}
```

## 4. What to expect

- ✓ Once uploaded, the buzzer will emit a continuous beep for 1 second, followed by 1 second of silence.
- ✓ This will loop indefinitely as long as the ESP32 has power.