80dB Passive Buzzer Module

DEVLAB FACTOR

SKU: UE0088

Product Reference Manual (V1.0)

Description

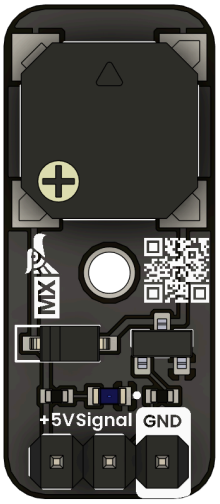
The DevLab passive buzzer module is a compact and easy-to-integrate passive buzzer designed for embedded systems and prototyping. It features a standard 3-pin interface consisting of VCC (5V), Signal, and GND, allowing direct connection to popular microcontrollers such as Arduino, ESP32, CH552, and STM32.

This module is intended for generating audible alerts in systems that require alarms, notifications, timing events, or user interaction feedback. It includes a pre-mounted passive buzzer element and supporting circuitry, enabling it to be driven directly from a digital output or PWMcapable GPIO pin. When a PWM signal is applied, the frequency determines the pitch of the tone produced by the buzzer. This approach provides flexibility to generate different sounds based on system state, user input, or event priority.

DevLab format compatibility.

Simplicity and compatibility are the core principles of the **DevLab form factor**. This standard defines a **compact and communication-optimized board layout**, ensuring straightforward connection and interoperability among DevLab modules.

By adhering to this format, the UNIT Capacitive Touch Sensor guarantees **efficient prototyping**, **ease of integration**, and **unified accessibility** across a wide ecosystem of devices and development platforms.



Key Features

- Requires external PWM signal (passive buzzer type)
- Compact module with 3-pin interface
- Wide voltage range (3.0 V to 5.5 V)
- Clear tone generation based on input frequency
- Breadboard-friendly layout - Low power consumption

Hardware Features

- Operating Voltage: 3.0 V – 5.5 V (5 V recommended for optimal sound pressure)
- Current Consumption: Typically between 5 mA and 30 mA depending on frequency and supply voltage
- Input Signal Type: PWM or square wave, externally generated
- Frequency Range: 500 Hz – 5 kHz
- Logic Compatibility: Accepts 3.3 V and 5 V logic levels (TTL compatible)
- Input Impedance: High, acts as capacitive load to the signal pin

Software Support

PlatformIO / VS Code
Provides a professional development environment with multi-board build automation, advanced debugging, and integrated version control for streamlined workflows.

MicroPython
General support for scripting and rapid prototyping on selected boards (mainly RP2040, STM32, and ESP32 families).

Development Frameworks by Platform

- **PY32 (Puya / Cortex-M0+)**
  Uses **HAL-based libraries** (PY32 HAL). Offers STM32-like API for GPIO, I²C, SPI, ADC, and timers.

- **STM32 (STMicroelectronics)**
  Compatible with **STM32 HAL** and **Arduino Core**. Supported by STM32CubeMX, PlatformIO, and Arduino IDE.

- **nRF Series (Nordic Semiconductor)**
  Works with **nRF SDK** and **Arduino Core (nRF5/nRF52)**. Enables BLE, Thread, and low-power IoT applications.

- **RP2040 (Raspberry Pi)**
  Supports **Arduino Core**, **MicroPython**, and **Pico-SDK** (official C/C++ SDK).

- **Espressif (ESP32 / ESP32-C6 / ESP32-H2)**
  Compatible with **ESP-IDF**, **Arduino Core**, and **MicroPython**. Includes Wi-Fi, BLE, and 802.15.4 Thread support.

- **MAX II (Intel CPLD)**
  Uses **Quartus Prime** and standard **JTAG programming tools** for HDL development and automation.

Note: MicroPython is officially supported for **RP2040**, **STM32**, and **ESP32** families. Other platforms may require community ports or custom builds.

Applications

- Audio indication for buttons or events
- Timers and countdown alerts
- Warning and alarm systems
- Feedback for embedded user interfaces
- Educational or DIY electronics kits

CONTENTS

1 The Board

1.1 Accessories

- 1×6-pin 2.54 mm male header

2 Ratings

2.1 Recommended Operating Conditions

Symbol	Description	Min	Typ	Max	Unit
V_{DD}	Operating voltage	2.0	-	5.5	V
I_{DD}	Operating Current	-	-	-	mA

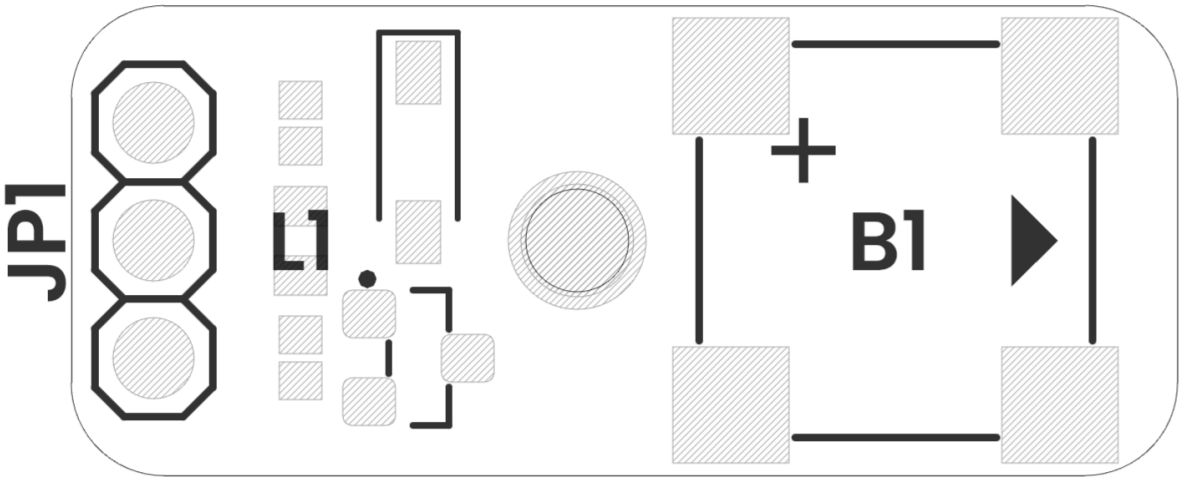
3 Functional Overview

The UNIT Buzzer Module integrates a passive piezoelectric transducer, which produces sound by converting electrical oscillations into mechanical vibrations. Unlike active buzzers, this module lacks an internal oscillator and requires an external signal source to operate.

Sound is generated when a square wave or PWM signal is applied to the Signal input pin. The frequency of this signal determines the pitch of the sound, while the duty cycle can influence the perceived volume and clarity. For best results, the input frequency should fall within the typical audible range of 500 Hz to 5 kHz.

The module's compact PCB includes clearly labeled VCC, Signal, and GND pins for easy connection to a microcontroller or signal generator. Due to its passive nature, the buzzer remains silent unless driven by a toggling signal.

3.2 Board Topology



Top View of Board Topology

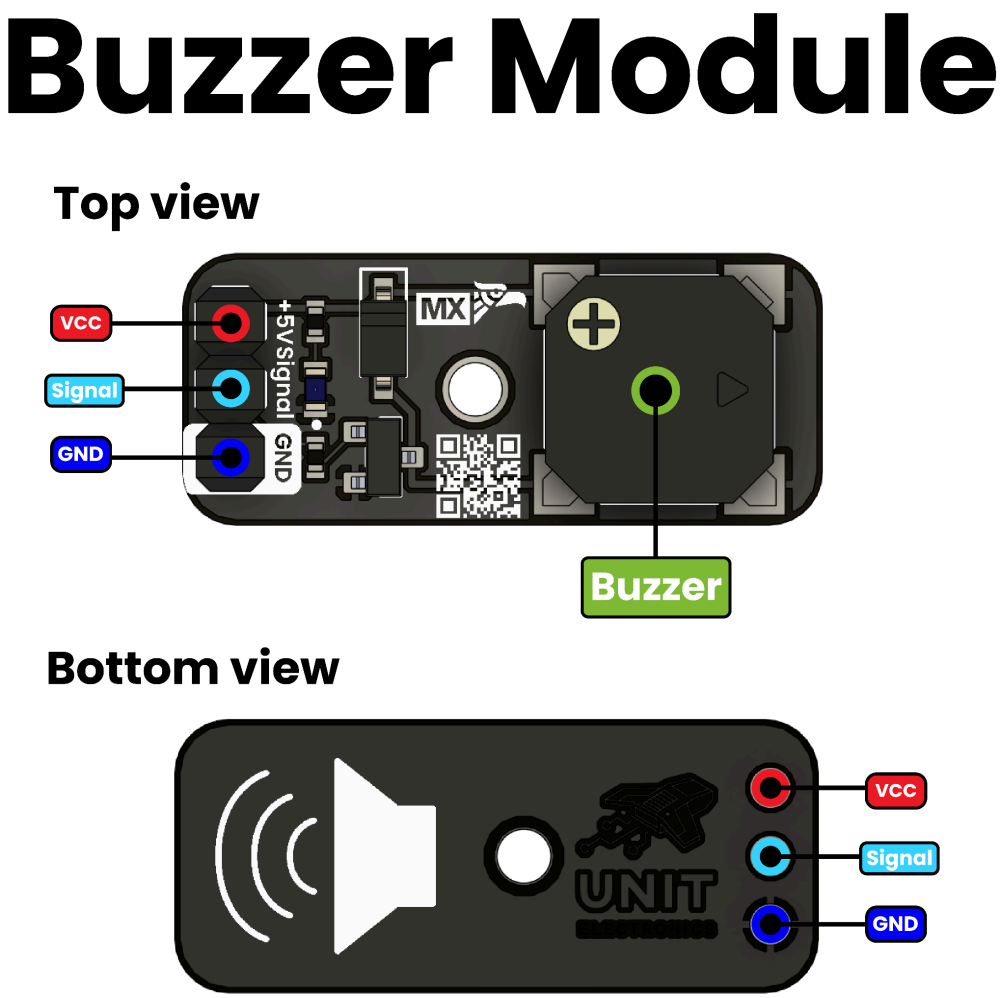
Views of Topology

Table 3.2.1 - Components Overview

Ref.	Description
L1	Built-In LED
JP1	Connector (JST 2.54 mm pitch)
B1	Buzzer

4 Connectors & Pinouts

4.1 General Pinout



General Pinout

4.2 Pinout General Description

Pin Label	Description
VCC	Power supply (3.3V or 5V)
GND	Ground
PIN	INPUT Pin

5 Board Operation

5.1 Getting Started with Micropython

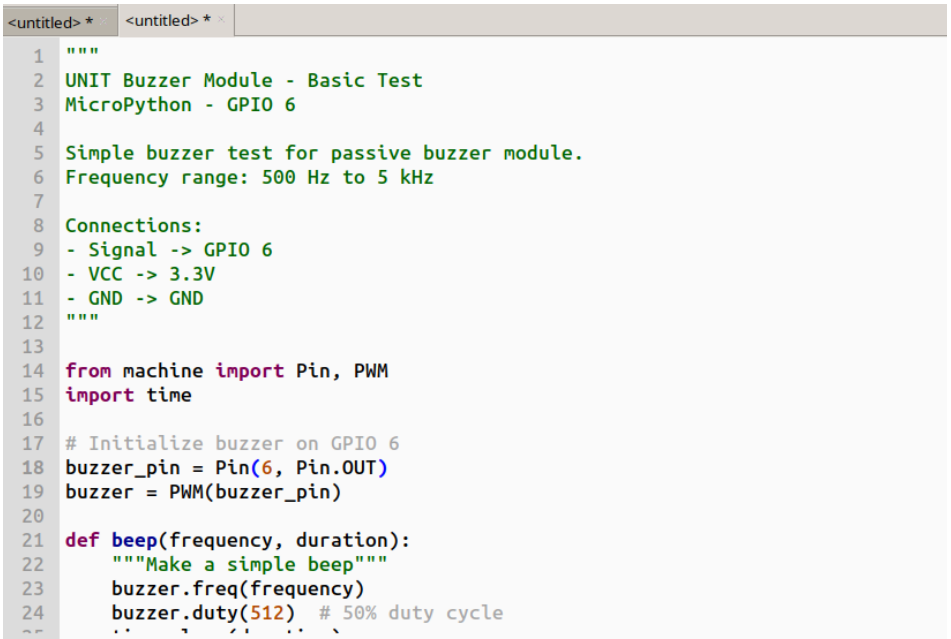
The DevLab 80dB Passive Buzzer Module generates sound when a square wave or PWM signal is applied to the Signal input pin. The frequency of this signal determines the pitch of the sound, while the duty cycle can influence the perceived volume and clarity. Powered by a **passive buzzer element,** it continuously monitors the input signal and reports audio output through a clean **PWM logic output**. You can easily customize its behavior by configuring GPIO pins to select between different **frequency ranges** and **duty cycle** modes, or control the output level.

What You'll Need

- DevLab **80dB Passive Buzzer Module**
- Microcontroller board (e.g., ESP32 or RP2040)
- Jumper wires
- USB cable
- MicroPython firmware and **Thonny IDE**

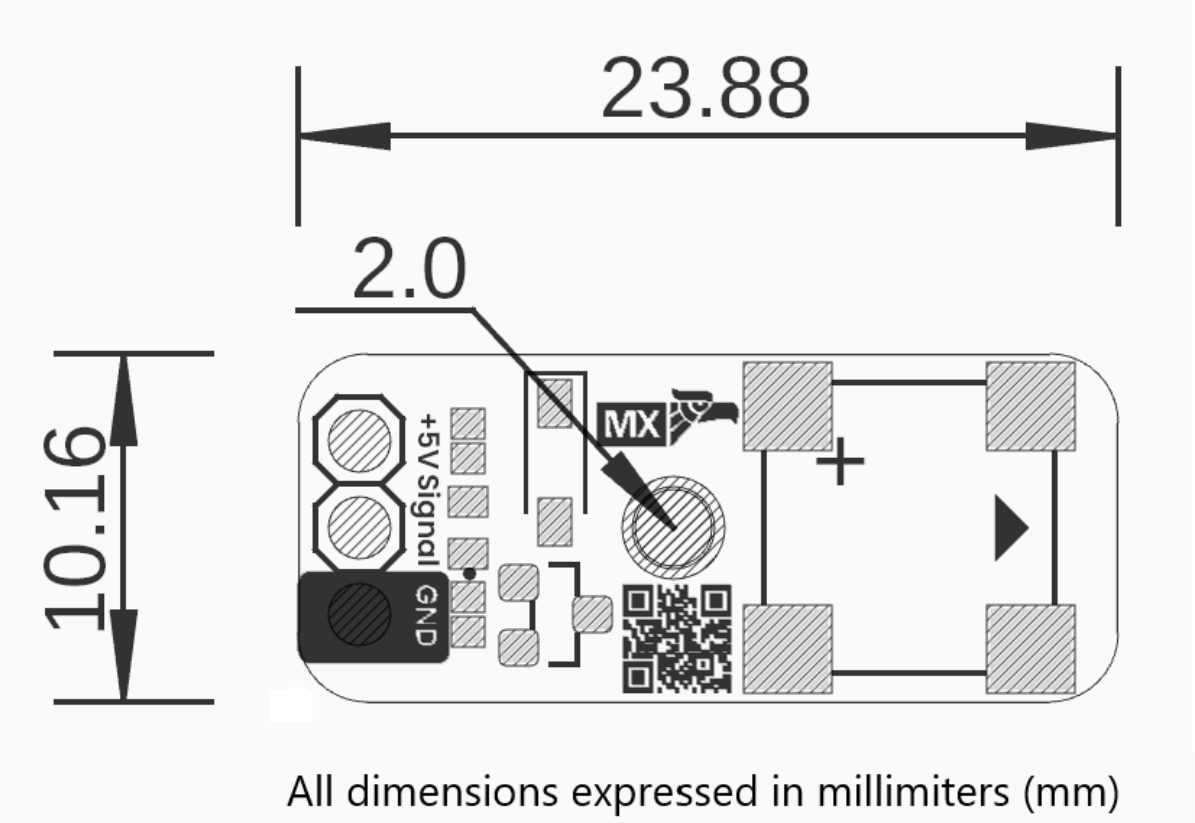
MicroPython Example

The official example is available in the DevLab repository:
https://github.com/UNIT-Electronics-MX/unit_devlab_80db_passive_buzzer_module




```
"""
UNIT Buzzer Module - Basic Test
MicroPython - GPIO 6

Simple buzzer test for passive buzzer module.
Frequency range: 500 Hz to 5 kHz

Connections:
- Signal -> GPIO 6
- VCC -> 3.3V
- GND -> GND
"""

from machine import Pin, PWM
import time

# Initialize buzzer on GPIO 6
buzzer_pin = Pin(6, Pin.OUT)
buzzer = PWM(buzzer_pin)

def beep(frequency, duration):
    """Make a simple beep"""
    buzzer.freq(frequency)
    buzzer.duty(512)  # 50% duty cycle
```

6 Mechanical Information



Mechanical dimensions in millimeters

7 Company Information

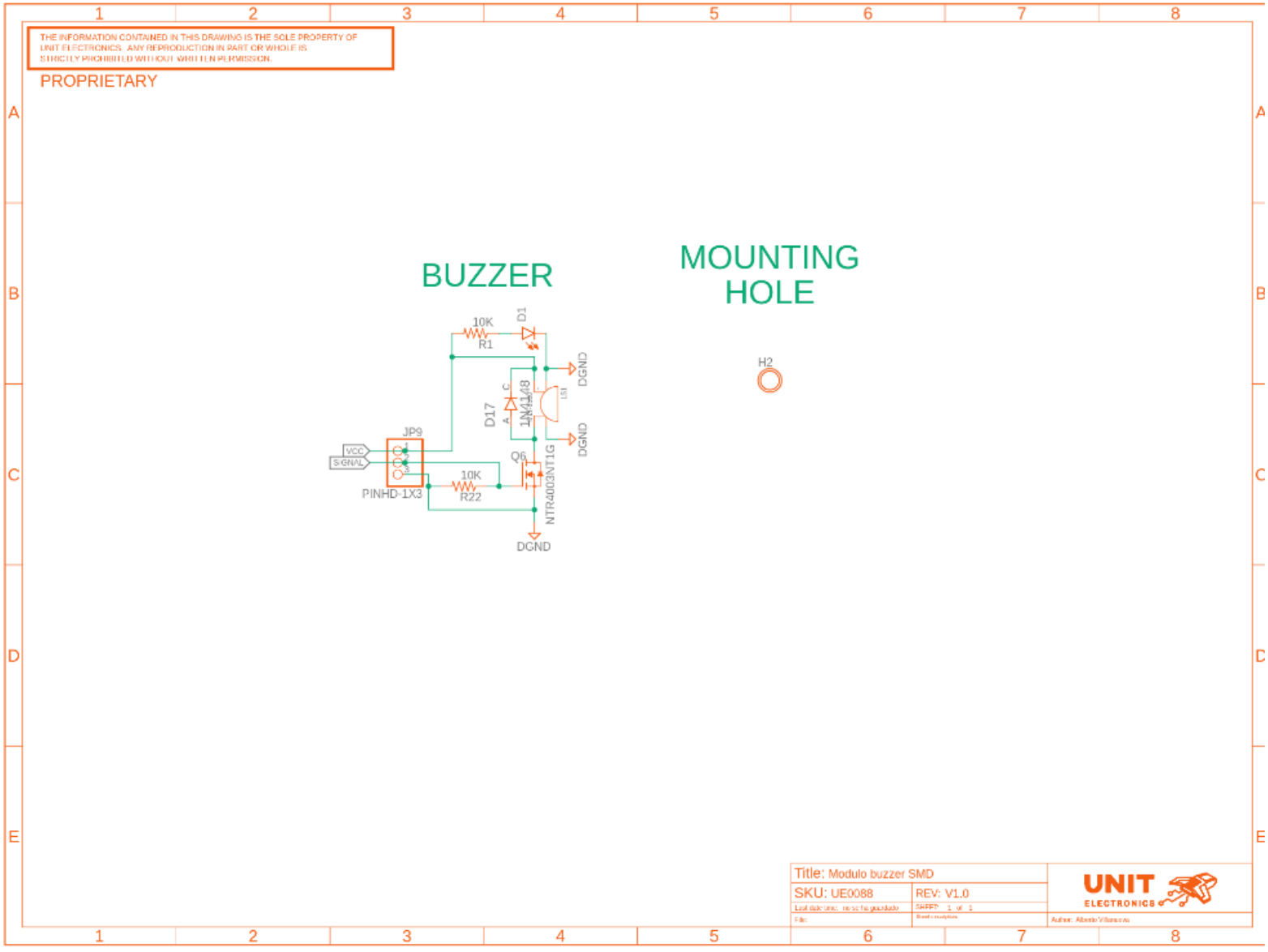
Company name	UNIT Electronics
Company website	https://uelectronics.com/
Company Address	Salvador 19, Cuauhtémoc, 06000 Mexico City, CDMX

8 Reference Documentation

Ref	Link
Documentation	https://github.com/UNIT-Electronics-MX/unit_devlab_80db_passive_buzzer_module
Thonny IDE	https://thonny.org/
Arduino IDE	https://www.arduino.cc/en/software
Visual Studio Code	https://code.visualstudio.com/download

9 Appendix

9.1 Schematic (*https://github.com/UNIT-Electronics-MX/unit_devlab_80db_passive_buzzer_module/blob/main/hardware/unit_sch_v_1_1_0_ue0088_modulo_buzzer.pdf*)


THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UNIT ELECTRONICS. ANY REPRODUCTION IN PART OR WHOLE IS STRICTLY PROHIBITED WITHOUT WRITTEN PERMISSION.
PROPRIETARY
BUZZER
MOUNTING HOLE
10K
R1
D1
DGND
D17
1N4148
JP9
VCC
SIGNAL
PINHD-1X3
10K
R22
Q6
NTR4003NT1G
DGND
H2
Title: Modulo buzzer SMD
SKU: UE0088
REV: V1.0
UNIT ELECTRONICS