

Lesson 3 Buzzer

Introduction

In this lesson, we will learn how to drive an active buzzer to beep with a Raspberry Pi and Python code.

Components

- 1 Raspberry Pi
- 1 Breadboard
- 1 Buzzer (Active)
- Jumper wires

Principle

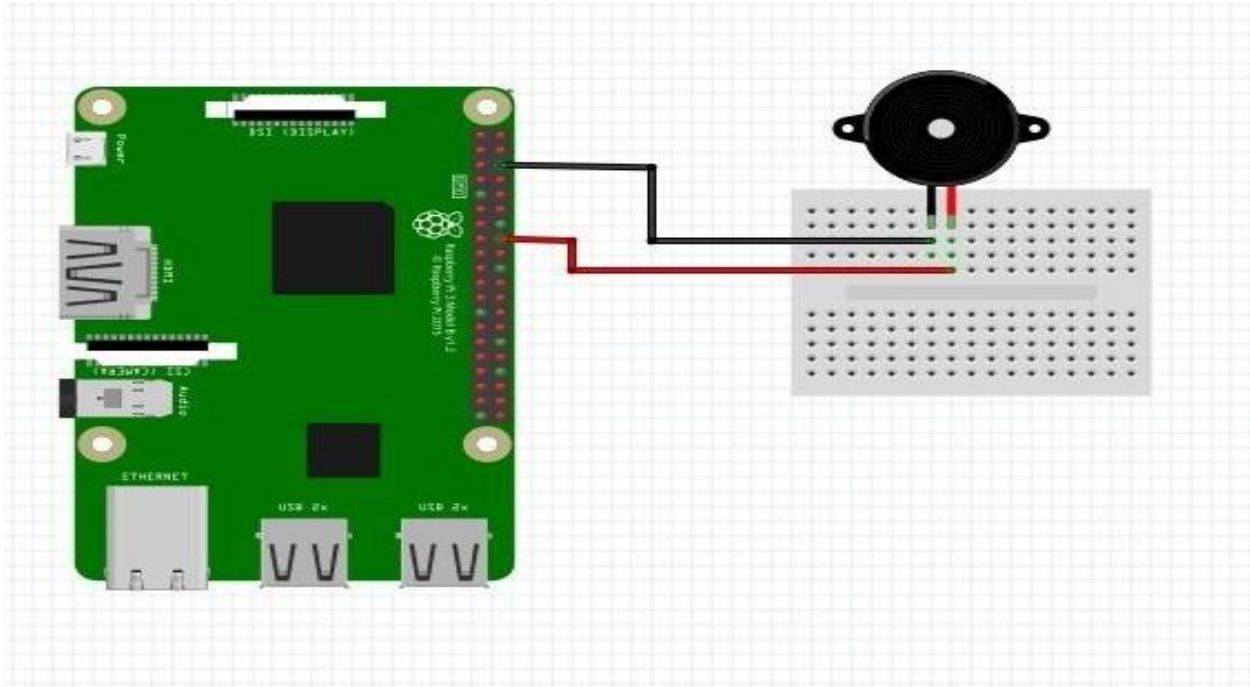
As a type of electronic buzzer with integrated structure, buzzers, which are supplied by DC power, are widely used in computers, printers, photocopiers, alarms and other electronic products for voice devices. Buzzers can be categorized as active and passive ones (see the following picture). Turn the pins of two buzzers face up, and the one with a green circuit board is a passive buzzer, while the other enclosed with a black tape is an active one.



The difference between an active buzzer and a passive buzzer is: An active buzzer has a built-in oscillating source, so it will make sounds when electrified. But a passive buzzer does not have such source, so it will not beep if DC signals are used; instead, you need to use square waves whose frequency is between 2K and 5K to drive it. The active buzzer is often more expensive than the passive one because of multiple built-in oscillating circuits.

The following is the electrical symbol of a buzzer. It has two pins with positive and negative poles. With a + in the surface represents the anode and the other is the cathode.

Hardware Setup



Python Coding

```
import RPi.GPIO as GPIO
import time

buzzer_pin = 23

GPIO.setmode(GPIO.BCM)
GPIO.setup(buzzer_pin, GPIO.OUT)

def buzz(pitch, duration):
    period = 1.0 / pitch
    delay = period / 2
    cycles = int(duration * pitch)
    for i in range(cycles):
        GPIO.output(buzzer_pin, True)
        time.sleep(delay)
```

```
        GPIO.output(buzzer_pin, False)

        time.sleep(delay)

while True:

    pitch_s = raw_input("Enter Pitch (200 to 2000): ")

    pitch = float(pitch_s)

    duration_s = raw_input("Enter Duration (seconds): ")

    duration = float(duration_s)

    buzz(pitch, duration)
```

Output

When you run the program, it will first prompt you for the pitch in Hz and then the duration of the buzz in seconds:

If you want to log into the Raspberry Pi remotely, type in the command:

Run the code:

```
$ sudo python buzzer.py

Enter Pitch (2000 to 10000): 2000

Enter Duration (seconds): 20
```

Application

- Fire Alarm System
- Horn
- Fault Alarm
- Blind Stick Project