Part 2

lesson 5

Active Buzzer

Overview

In this lesson, you will learn how to generate a sound with an active buzzer.

Component Required:

(1) x Elegoo ESP32

(1) x Active buzzer

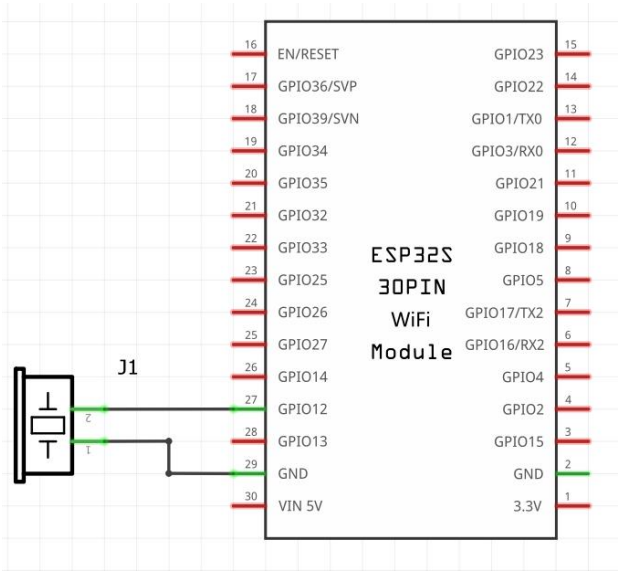
(2) x F-M wires (Female to Male DuPont wires)

Component Introduction

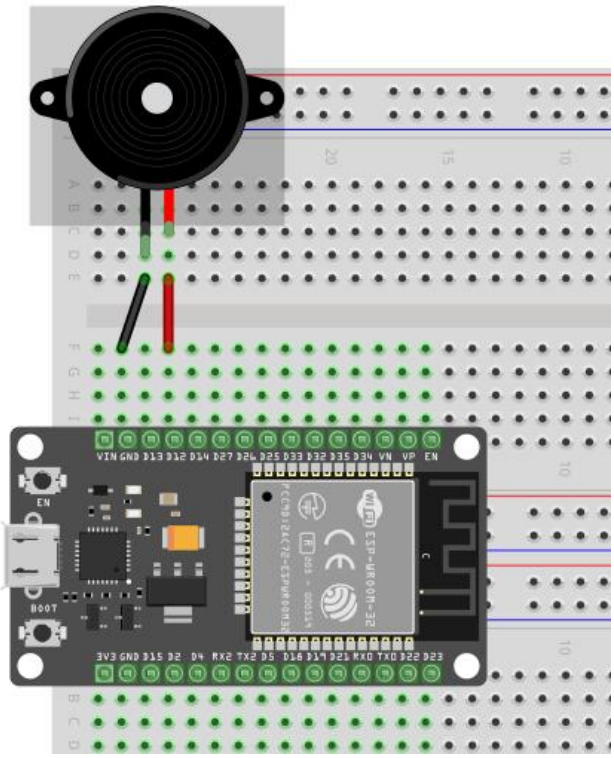
BUZZER:

Electronic buzzers are DC-powered and equipped with an integrated circuit. They are widely used in computers, printers, photocopiers, alarms, electronic toys, automotive electronic devices, telephones, timers and other electronic products for voice devices. Buzzers can be categorized as active and passive ones. Turn the pins of two buzzers face up. 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 the two is that an active buzzer has a built-in oscillating source, so it will generate a sound when electrified. A passive buzzer does not have such a source so it will not tweet 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.



Connection Schematic



Wiring diagram

Code

Please open the program:

while(condition)
[Control Structure]

```
while(1)
{
  ........
}
```

Description

A while loop will loop continuously, and infinitely, until the expression inside the parenthesis, () becomes false. Something must change the tested variable, or the while loop will never exit. This could be in your code, such as an incremented variable, or an external condition, such as testing a sensor.

Parameters

condition: a boolean expression that evaluates to true or false.

Syntax

```
while (condition) {
  // statement(s)
}
```

```
unsigned char i;
```

unsigned char
[Data Types]
Range:0~255
Description

An unsigned data type that occupies 1 byte of memory. Same as the byte datatype.

The unsigned char datatype encodes numbers from 0 to 255.

For consistency of Arduino programming style, the byte data type is to be preferred.

Syntax

```
unsigned char var = val;
```

Parameters

var: variable name.

val: the value to assign to that variable.

Example Code

```
unsigned char myChar = 240;
```