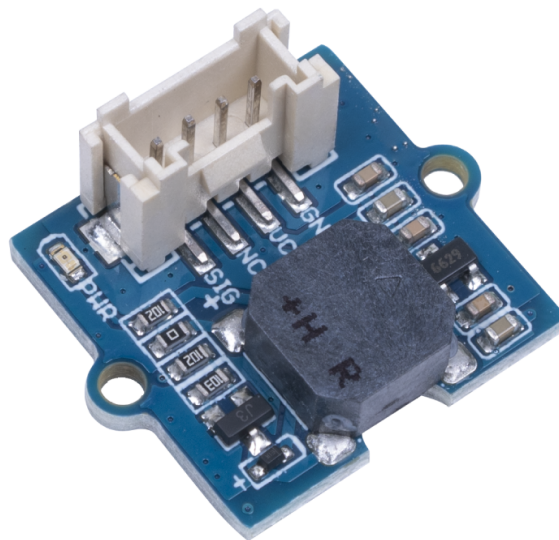


Grove passive Buzzer



This is a 3-5V passive buzzer. You can change the PWM frequency to award different beep sound to get a "buzzer music". Also, the buzzer can be set as an alarm for security applications. So get one and start your own project!

[Get One Now !\[\]\(3dfb8d66e81160ad61421a3452093d1b_img.jpg\)](https://www.seeedstudio.com/Grove-Passive-Buzzer-p-4525.html)

[<https://www.seeedstudio.com/Grove-Passive-Buzzer-p-4525.html>]

Features

- Passive: Tunable passive buzzer
- Interface: Grove

Specification

Item	Value
Voltage range	3V-5V
Resonant Frequency	2700 Hz
sound output	> 80dB
Working temperature	-20-70 °C
Dimensions	20mm * 20mm * 10mm
Weight	3g
Battery	Exclude

Getting Started

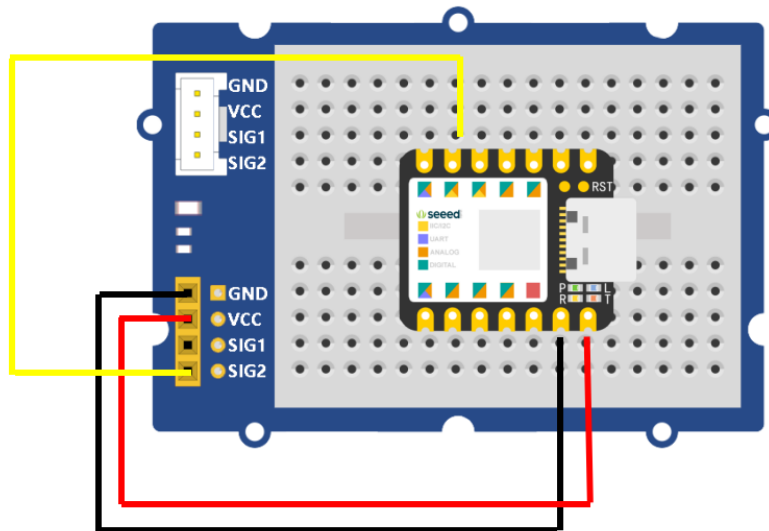
Materials Required

- [Seeeduino XIAO](https://www.seeedstudio.com/Seeeduino-XIAO-Arduino-Microcontroller-SAMD21-Cortex-M0+-p-4426.html) [https://www.seeedstudio.com/Seeeduino-XIAO-Arduino-Microcontroller-SAMD21-Cortex-M0+-p-4426.html]
- [Grove Passive Buzzer](https://www.seeedstudio.com/Grove-Passive-Buzzer-p-4525.html) [https://www.seeedstudio.com/Grove-Passive-Buzzer-p-4525.html]
- [Grove Breadboard](https://www.seeedstudio.com/Grove-Breadboard-p-4034.html) [https://www.seeedstudio.com/Grove-Breadboard-p-4034.html]
- [Grove cable](https://www.seeedstudio.com/catalogsearch/result/?q=grove+cable) [https://www.seeedstudio.com/catalogsearch/result/?q=grove+cable]

Platform Select



Hardware connection



The Grove interface on the breadboard and on the Grove Passive Buzzer are connected by the Grove cable.

Software

- **Step1** Copy the code below to the Arduino IDE and upload. If you do not know how to update the code, please check [How to upload code](https://wiki.seeedstudio.com/Upload_Code/) [https://wiki.seeedstudio.com/Upload_Code/].

Code example1 - simply get the beep sound

```

1  int buzzer = 5; // Buzzer connect with Pin 5
2  int frequency = 2700; //reach the Resonant Frequency
3  int cycle = 1000000/frequency;
4  void setup()
5  {
6  Serial.begin(9600); // set the baud rate
7  pinMode(buzzer,OUTPUT); // set buzzer as output
8  }
9
10 void loop()
11 {

```

```
12  digitalWrite(buzzer,HIGH);
13  delayMicroseconds(cycle/2);
14  digitalWrite(buzzer,LOW);
15  delayMicroseconds(cycle/2); // run the PWM cycle
16  }
```

- **Step2** After uploading the code to the board, you will hear the buzzer beep.

Code example2 - Use buzzer to play music

```
1  //set the corresponding notes with frequency
2  #define NOTE_D0 0
3  #define NOTE_D1 294
4  #define NOTE_D2 330
5  #define NOTE_D3 350
6  #define NOTE_D4 393
7  #define NOTE_D5 441
8  #define NOTE_D6 495
9  #define NOTE_D7 556
10
11 #define NOTE_DL1 147
12 #define NOTE_DL2 165
13 #define NOTE_DL3 175
14 #define NOTE_DL4 196
15 #define NOTE_DL5 221
16 #define NOTE_DL6 248
17 #define NOTE_DL7 278
18
19 #define NOTE_DH1 589
20 #define NOTE_DH2 661
21 #define NOTE_DH3 700
22 #define NOTE_DH4 786
23 #define NOTE_DH5 882
24 #define NOTE_DH6 990
25 #define NOTE_DH7 112
26
27 #define WHOLE 1
```



```
28 #define HALF 0.5
29 #define QUARTER 0.25
30 #define EIGHTH 0.25
31 #define SIXTEENTH 0.625
32
33 //the note part of the whole song
34 int tune[] =
35 {
36     NOTE_DH1, NOTE_D6, NOTE_D5, NOTE_D6, NOTE_D0,
37     NOTE_DH1, NOTE_D6, NOTE_D5, NOTE_DH1, NOTE_D6, NOTE_
38     NOTE_D6, NOTE_D6, NOTE_D5, NOTE_D6, NOTE_D0, NOTE_D6
39     NOTE_DH1, NOTE_D6, NOTE_D5, NOTE_DH1, NOTE_D6, NOTE_
40
41     NOTE_D1, NOTE_D1, NOTE_D3,
42     NOTE_D1, NOTE_D1, NOTE_D3, NOTE_D0,
43     NOTE_D6, NOTE_D6, NOTE_D6, NOTE_D5, NOTE_D6,
44     NOTE_D5, NOTE_D1, NOTE_D3, NOTE_D0,
45     NOTE_DH1, NOTE_D6, NOTE_D6, NOTE_D5, NOTE_D6,
46     NOTE_D5, NOTE_D1, NOTE_D2, NOTE_D0,
47     NOTE_D7, NOTE_D7, NOTE_D5, NOTE_D3,
48     NOTE_D5,
49     NOTE_DH1, NOTE_D0, NOTE_D6, NOTE_D6, NOTE_D5, NOTE_D
50     NOTE_D0, NOTE_D5, NOTE_D1, NOTE_D3, NOTE_D0,
51     NOTE_DH1, NOTE_D0, NOTE_D6, NOTE_D6, NOTE_D5, NOTE_D
52     NOTE_D0, NOTE_D5, NOTE_D1, NOTE_D2, NOTE_D0,
53     NOTE_D3, NOTE_D3, NOTE_D1, NOTE_DL6,
54     NOTE_D1,
55     NOTE_D3, NOTE_D5, NOTE_D6, NOTE_D6,
56     NOTE_D3, NOTE_D5, NOTE_D6, NOTE_D6,
57     NOTE_DH1, NOTE_D0, NOTE_D7, NOTE_D5,
58     NOTE_D6,
59 };
60
61 //the duration time of each note
62 float duration[] =
63 {
64     1, 1, 0.5, 0.5, 1,
65     0.5, 0.5, 0.5, 0.5, 1, 0.5, 0.5,
66     0.5, 1, 0.5, 1, 0.5, 0.5,
67     0.5, 0.5, 0.5, 0.5, 1, 1,
68
```

```

69     1, 1, 1 + 1,
70     0.5, 1, 1 + 0.5, 1,
71     1, 1, 0.5, 0.5, 1,
72     0.5, 1, 1 + 0.5, 1,
73     0.5, 0.5, 0.5, 0.5, 1 + 1,
74     0.5, 1, 1 + 0.5, 1,
75     1 + 1, 0.5, 0.5, 1,
76     1 + 1 + 1 + 1,
77     0.5, 0.5, 0.5 + 0.25, 0.25, 0.5 + 0.25, 0.25, 0.5 + (
78     0.5, 1, 0.5, 1, 1,
79     0.5, 0.5, 0.5 + 0.25, 0.25, 0.5 + 0.25, 0.25, 0.5 + (
80     0.5, 1, 0.5, 1, 1,
81     1 + 1, 0.5, 0.5, 1,
82     1 + 1 + 1 + 1,
83     0.5, 1, 0.5, 1 + 1,
84     0.5, 1, 0.5, 1 + 1,
85     1 + 1, 0.5, 0.5, 1,
86     1 + 1 + 1 + 1
87 };
88
89 int length;//define the number of notes
90 int buzzer = 5; //set the buzzer Pin
91 void setup()
92 {
93     pinMode(buzzer, OUTPUT); // set the buzzer as output
94     length = sizeof(tune) / sizeof(tune[0]); //count the
95 }
96
97 void loop()
98 {
99     for (int x = 0; x < length; x++) //"sing" the note (
100     {
101         tone(buzzer, tune[x]); //output the "x" note
102         delay(400 * duration[x]); //rythem of the music,it
103         noTone(buzzer);//stop the current note and go to t
104     }
105     delay(5000);//after playing the whole song, delay f
106 }

```

- **Step3** After uploading the code to the board, you can hear a music from the buzzer.

Schematic Online Viewer



Resource

- **[PDF]** [MLT_8530_DATASHEET](https://files.seeedstudio.com/products/107020109/document/MLT_8530_datasheet.pdf)
[https://files.seeedstudio.com/products/107020109/document/MLT_8530_datasheet.pdf]

- **[PDF]** [Hardware schematic](#)

[https://files.seeedstudio.com/products/107020109/document/Grove_Passive_Buzzer_SCH_190925.pdf]