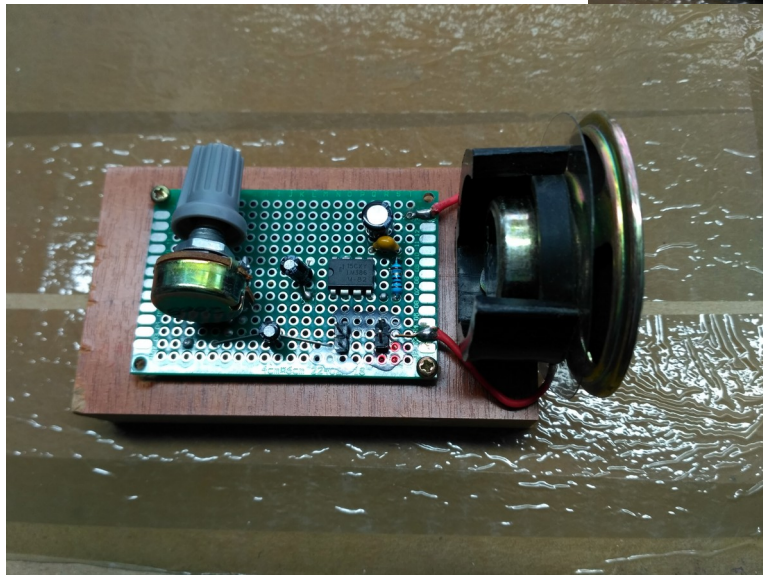
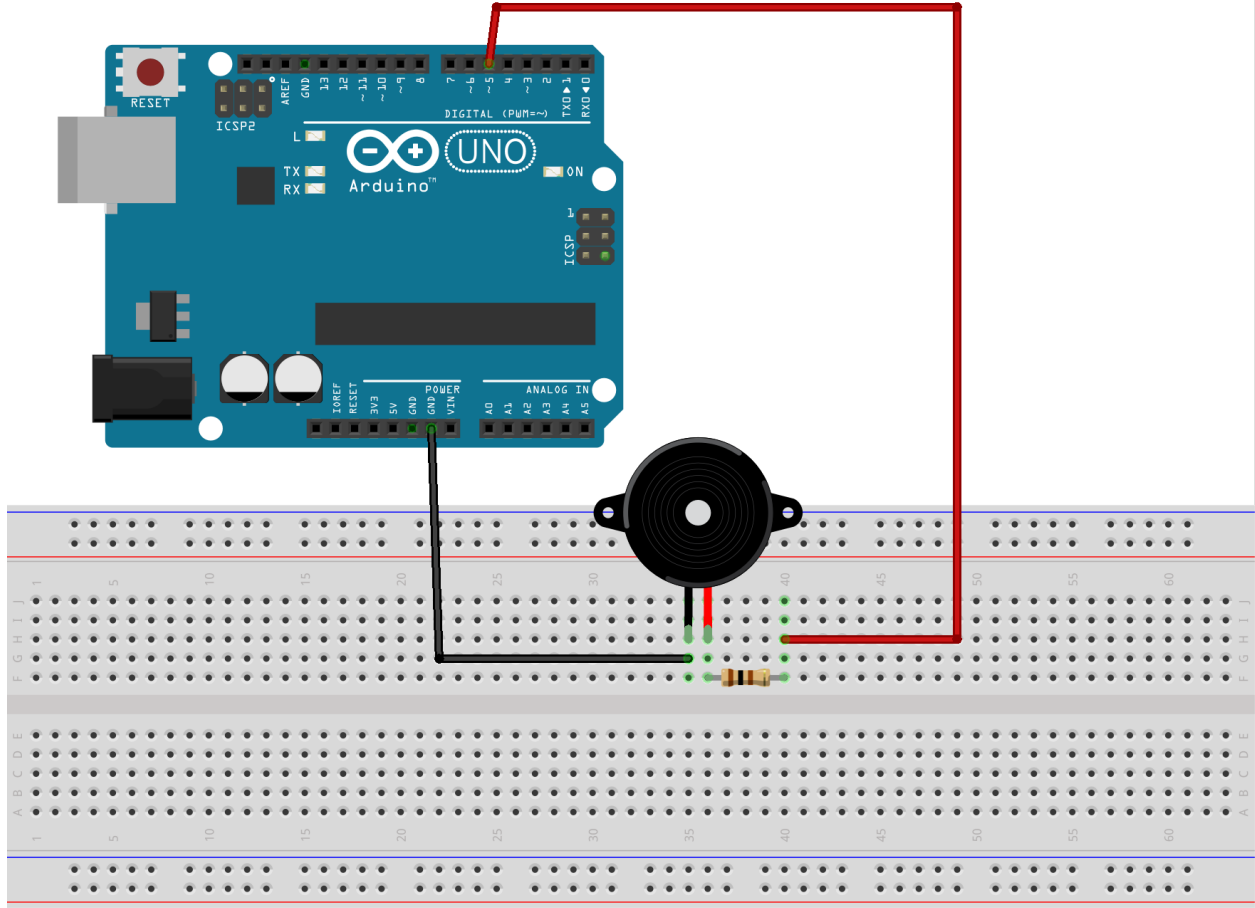


Arduino



Som





fritzing

```
1 #define beeper 5 //beeper ligado no pino 5, por meio de um resistor de 100 ohms
2 #define saida OUTPUT
3
4 void setup() {
5   pinMode(beeper,saida);
6 }
7
8
9 void loop() {
10  /*
11   tone(beeper,1500, 30); //pino, frequencia, duracao
12   delay(500);
13   noTone(beeper);
14   delay(500);
15  */
16  //do projeto do trem
17  tone(beeper, 1350);
18  delay(200);
19  tone(beeper, 1050);
20  delay(200);
21 }
```

Dó, Ré...

Frequência das notas

```
1 #define Do 262
2 #define Re 294
3 #define Mi 330
4 #define Fa 349
5 #define Sol 392
6 #define La 440
7 #define Si 494
8 #define Pausa 0
9
10 char buz = 11; //pino conectado no buzzer com resistor de 100ohms em série
11
```



```

12 void setup(){
13   pinMode(buz, OUTPUT); // seta o pino como saída
14   //Após a identificação do buzzer, o primeiro parâmetro representa a frequência, o segundo, a duração
15   tone(buz, La,500);
16   delay(500);
17   tone(buz, Re,300);
18   delay(500);
19   tone(buz, Fa,250);
20   delay(500);
21   tone(buz, Sol, 250);
22   delay(500);
23   tone(buz, La, 250);
24   delay(500);
25   tone(buz, Re, 300);
26   delay(500);
27   tone(buz, Fa, 200);
28   delay(500);
29   tone(buz, Sol, 200);
30   delay(500);
31   tone(buz, Mi, 700);
32   delay(500);
33   tone(buz, Pausa, 200);
34   delay(500);
35   tone(buz, Sol, 500);
36   delay(500);
37   tone(buz, Do, 500);
38   delay(500);
39   tone(buz, Fa, 200);
40   delay(500);
41   tone(buz, Mi,200);
42   delay(500);
43   tone(buz, Sol,200);
44   delay(500);
45   tone(buz, Do,500);
46   delay(500);
47   tone(buz, Fa,200);
48   delay(500);
49   tone(buz, Mi,200);
50   delay(500);
51   tone(buz, Re,500);
52

```



Natal




```

2 /* Codigo adaptado por Gustavo Silveira
3  * www.musiconerd.com
4  */
5
6 int speakerPin = 11;
7
8 int length = 26;
9 char notes[] = "eeeeeeegcde fffffeeeeddedg";
10 int beats[] = { 1, 1, 2, 1, 1, 2, 1, 1, 1, 1, 4, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2 };
11 int tempo = 300;
12
13 void playTone(int tone, int duration) {
14     for (long i = 0; i < duration * 1000L; i += tone * 2) {
15         digitalWrite(speakerPin, HIGH);
16         delayMicroseconds(tone);
17         digitalWrite(speakerPin, LOW);
18         delayMicroseconds(tone);
19     }
20 }
21
22 void playNote(char note, int duration) {
23     char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C' };
24     int tones[] = { 1915, 1700, 1519, 1432, 1275, 1136, 1014, 956 };
25
26     // play the tone corresponding to the note name
27     for (int i = 0; i < 8; i++) {
28         if (names[i] == note) {
29             playTone(tones[i], duration);
30         }
31     }
32 }
33
34 void setup() {
35     pinMode(speakerPin, OUTPUT);
36 }
37
38 void loop() {
39     for (int i = 0; i < length; i++) {
40         if (notes[i] == ' ') {
41             delay(beats[i] * tempo); // rest
42         } else {
43             playNote(notes[i], beats[i] * tempo);
44         }
45
46         // pause between notes
47         delay(tempo / 2);
48     }
49 }

```



O que você aprendeu:

1. Utilizar `tone()` / `noTone()`
2. Reproduzir som usando Arduino

Parabéns!

