



Universidade do Estado de Minas Gerais
Fundação Educacional de Ituiutaba
Engenharia de Computação



CURSO BÁSICO ARDUINO Parte 2

João Ludovico Maximiano Barbosa

Rafael Caetano da Silva

2016

UTILIZANDO UM BOTÃO

Botões ou interruptores são dispositivos que ligam dois pontos em um circuito quando você os pressiona ou solta. Neste exemplo, ao se pressionar o botão será enviado uma mensagem no serial monitor.

Para montar o circuito você terá que ligar três fios ao Arduino. Os dois primeiros, vermelho e preto (Figura 1), serão utilizados para ligar as duas linhas verticais do lado longo do protoboard para proporcionar o acesso à alimentação de 5 volts e ao terra (Ground). O terceiro fio será conectado ao pino digital 2 e a uma perna do botão. Essa mesma perna do botão também será ligado a um resistor de pull-down de 10 KOhms) que estará conectado ao terra. A outra perna do botão liga-se a alimentação 5 V conforme mostra a Figura 1.

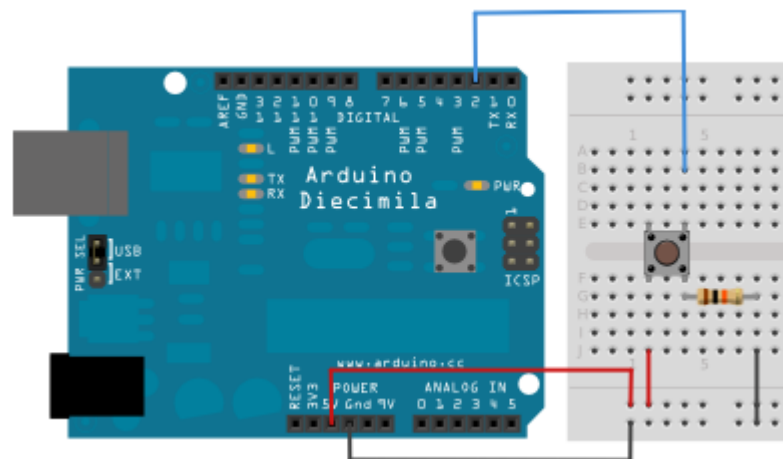


Figura 1 – Circuito de montagem do botão com o arduino

Quando o botão **não** está pressionado, ele funciona como uma chave aberta, assim, não há qualquer ligação entre as duas pernas do botão, de modo que o pino digital 2 está ligado ao terra (através da resistência de pull-down) e podemos ler LOW. Quando o botão é pressionado ele funciona como uma chave fechada, fazendo uma ligação entre as suas duas pernas, ligando o pino digital 2 a 5 volts, de modo que lemos HIGH.

Você também pode ligar este circuito de forma oposta, com um resistor de pull-up mantendo a entrada HIGH quando o botão **não** é pressionado, e LOW quando o botão for pressionado. Se assim for, o comportamento do nosso exemplo será invertida, com a mensagem sendo enviada ao serial monitor quando o botão não estiver sendo pressionado.

Na Figura 2 podemos visualizar o código deste exemplo:

```
sketch_aug20a$
const int botaoPin = 2; // numero do pino em que o botão será conectado
int botaoStatus = 0;    // variavel para a leitura do status do botao

void setup() {

  // Inicializando o serial:
  Serial.begin(9600);
  // Inicializando o PINO do botão como entrada
  pinMode(botaoPin, INPUT);

  // Fazendo a leitura do status do botão
  botaoStatus = digitalRead(botaoPin);

  // Verificando se o botão foi pressionado.
  // Se o botão estiver em nível HIGH (alto) imprime uma mensagem
  //no serial monitor:
  if (botaoStatus == HIGH) {
    Serial.println("Botão Pressionado");
  }
}

void loop(){
}
```

Figura 2 – Código utilizando um botão como entrada

Atividades:

- Passe o trecho de código que realiza a leitura do status do botão para BLOCO LOOP;
- Coloque um delay de 5 segundos após a imprimir a mensagem no serial monitor. Tente pressionar o botão varias vezes neste intervalo e observe o que acontece, depois disso remova o delay e pressione o botão varias vezes novamente;
- Adicione um LED ao circuito, da mesma maneira que fizemos nas atividades anteriores, e altere o código para ligar o LED toda vez que o botão for pressionado e desliga-lo quando o botão estiver sendo pressionado.

Utilizando Buzzer

Buzzer é um componente eletrônico que é composto por 2 camadas de Metal e uma terceira camada interna de cristal Piezoelétrico, este componente recebe uma fonte de energia e através dela emite uma frequência sonora. Ele está presente em diversos tipos de aparelhos, como despertador, carros e até em computadores, que quando ligado emite um Beep informando que a memória foi reconhecida (Correa, 2015).

Para montar o circuito você terá que ligar dois fios ao Arduino. Um dos fios será conectado ao pino digital 5 do arduino e a outra ponta dela ao terminal positivo do Buzzer, já o outro fio será conectado ao GND (Ground/Terra) do arduino e ao terminal negativo do Buzzer, conforme pode ser observado na Figura 3.

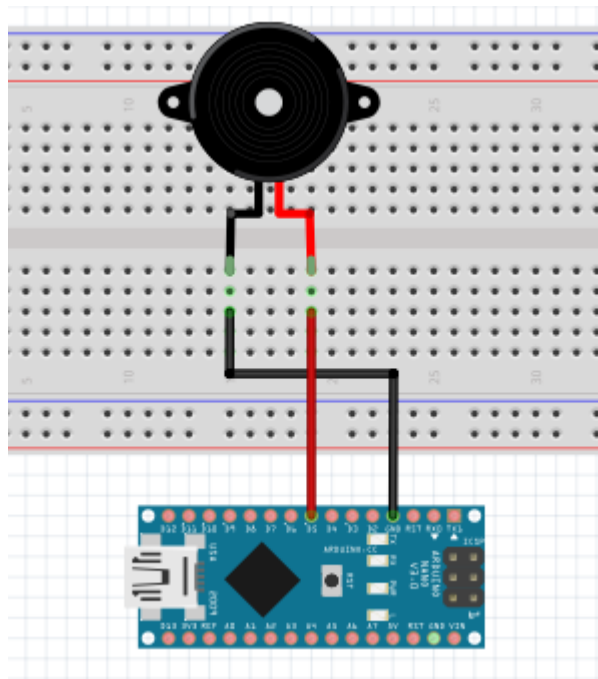


Figura 3 - Circuito de montagem do Buzzer com o arduino

O código que iremos utilizar é um dos exemplos nativos da IDE do Arduino, para isso clique na abas “Arquivo” depois em “Exemplos”, escolha “Digital” e selecione “toneMelody”, conforme ilustra a Figura 4.

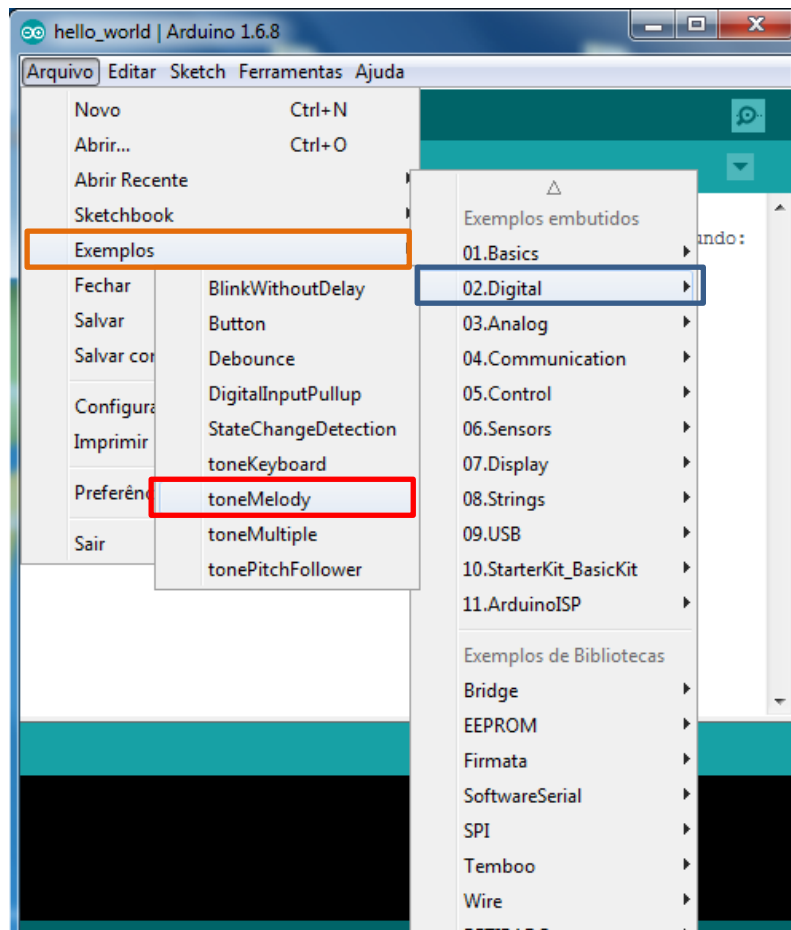


Figura 4 - Selecionando o código exemplo toneMelody

Como o pino positivo do buzzer no código de exemplo está definida como sendo o pino 8, devemos alterar este valor para o pino 5, nas seguintes linhas:

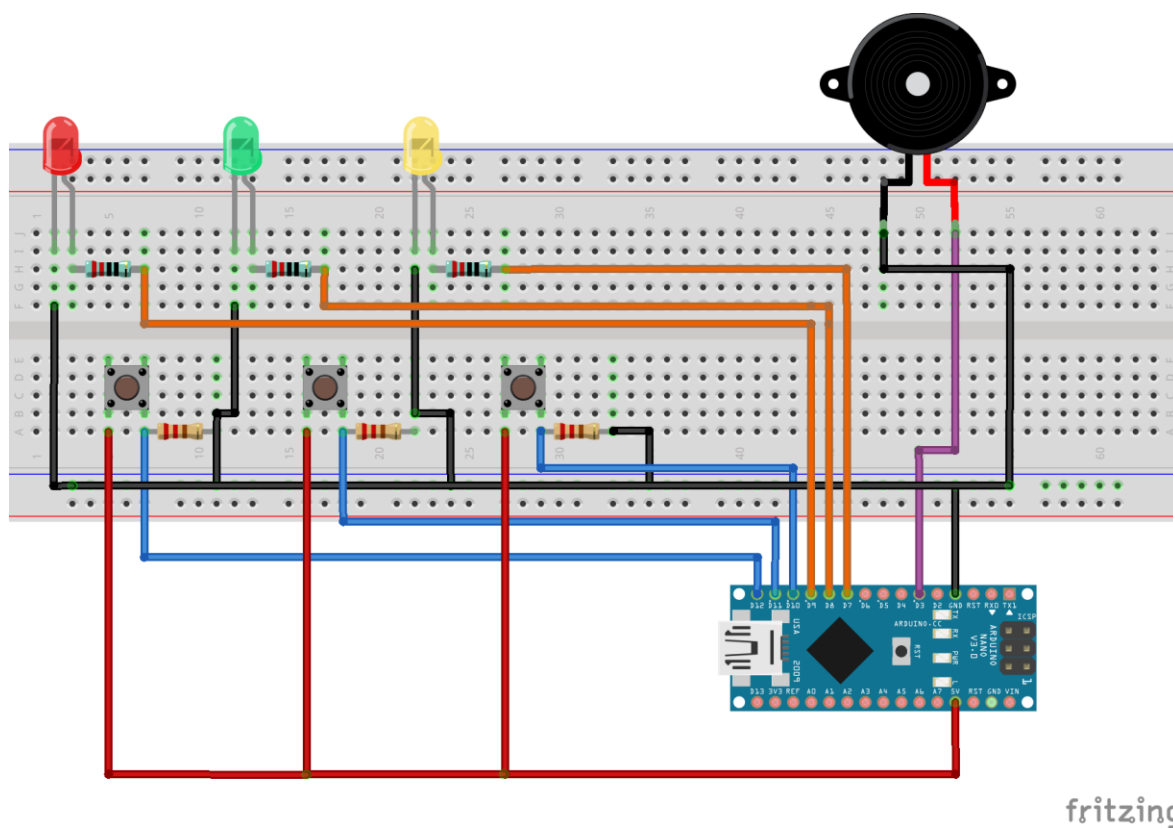
- `tone(8, melody[thisNote], noteDuration); //linha 38`
- `noTone(8); //linha 45`

Feito esta alteração basta carregarmos o código no arduino para testarmos o buzzer.

Game Simon

O Simon é um jogo que busca estimular a memorização de cores e sons. Possui botões coloridos que emitem sons harmônicos e se iluminam em sequência. Cabendo aos jogadores repetir o processo sem errar.

Iremos montar o nosso simon com 3 (botões), ou seja, iremos utilizar 3 botões e 3 led's de preferência de cores diferentes para simularmos ao máximo o jogo simon, e um buzzer para emitir os sons. Na Figura 5 podemos visualizar a montagem do circuito:



fritzing

Figura 5 - Circuito eletrônico jogo simon