



## Sessão 04

### Pedra, papel, tesoura no arduino

Professor Thiago Goveia

### Objetivos da sessão:

- Compreender o funcionamento básico da plataforma Arduino e seu ambiente de programação.
- Aprender a controlar um atuador (servomotor) e a ler dados de um sensor (ultrassônico).
- Extrapolar o pensamento lógico do jogo "Pedra, Papel e Tesoura" para um contexto de computação física.
- Implementar a lógica principal do jogo na função `loop()` do Arduino.

### Materiais:

- [Repositório](#) do projeto criado na Sessão 01 no GitHub.
- Kit Arduino (Placa UNO, Protoboard, Fios Jumper).
- 1 Servomotor (ex: SG90).
- 1 Sensor Ultrassônico (HC-SR04).
- Computador com a IDE do Arduino instalada.

### Sumário:

<a href="#">1. Introdução: Do Virtual para o Físico</a>	<a href="#">2</a>
<a href="#">2. A Caixa de Ferramentas do Arduino</a>	<a href="#">2</a>
<a href="#">2.1 O que é o Arduino?</a>	<a href="#">2</a>
<a href="#">2.2 O Atuador: Servomotor</a>	<a href="#">2</a>
<a href="#">2.3 O Sensor: Sensor Ultrassônico</a>	<a href="#">3</a>
<a href="#">3. Mão na Massa! Construindo nosso Oponente Robótico</a>	<a href="#">3</a>
<a href="#">3.1 Simulação no Tinkercad</a>	<a href="#">3</a>
<a href="#">3.2 O Desafio: Implementando a função <code>loop()</code></a>	<a href="#">4</a>
<a href="#">3.3 Montagem do Circuito</a>	<a href="#">4</a>
<a href="#">3.3 Upload do código para o arduino</a>	<a href="#">5</a>
<a href="#">4. Indo além!</a>	<a href="#">5</a>



## Conteúdo da sessão:

### 1. Introdução: Do Virtual para o Físico

Até agora, nosso jogo "Pedra, Papel e Tesoura" viveu inteiramente dentro do computador. Nós digitamos nossa jogada e recebemos uma resposta em texto. Hoje, vamos quebrar essa barreira! O objetivo é usar a mesma lógica que já desenvolvemos para controlar um dispositivo físico.

Para um estudante de Automação Industrial, este é um passo fundamental. Sistemas automatizados funcionam exatamente assim: eles usam **sensores** para perceber o mundo (entrada de dados, como se fosse um `scanf`) e **atuadores** para agir sobre ele (saída de dados, como um `printf`). Nosso projeto de hoje é um sistema de automação em miniatura: um oponente robótico que "vê" sua mão e reage com um movimento físico.

### 2. A Caixa de Ferramentas do Arduino

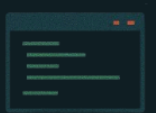
#### 2.1 O que é o Arduino?

O Arduino é um microcontrolador: um pequeno cérebro programável. Ele é a ponte entre nosso código e o mundo físico. Sua programação, chamada de "sketch", é baseada em C/C++ e possui uma estrutura fundamental:

- `void setup()`: Roda **uma única vez** quando o Arduino é ligado. É aqui que configuramos os pinos e inicializamos os componentes.
- `void loop()`: Roda **continuamente**, em um ciclo infinito, após o `setup()` terminar. É aqui que a lógica principal do nosso jogo vai viver.

#### 2.2 O Atuador: Servomotor

O servomotor será a "mão" do nosso oponente. Diferente de um motor comum, ele não gira continuamente; ele se move para um **ângulo específico** (geralmente entre 0 e 180 graus) e permanece lá. Para controlá-lo, usamos a biblioteca `Servo.h` e o comando `servo.write(angulo);`.



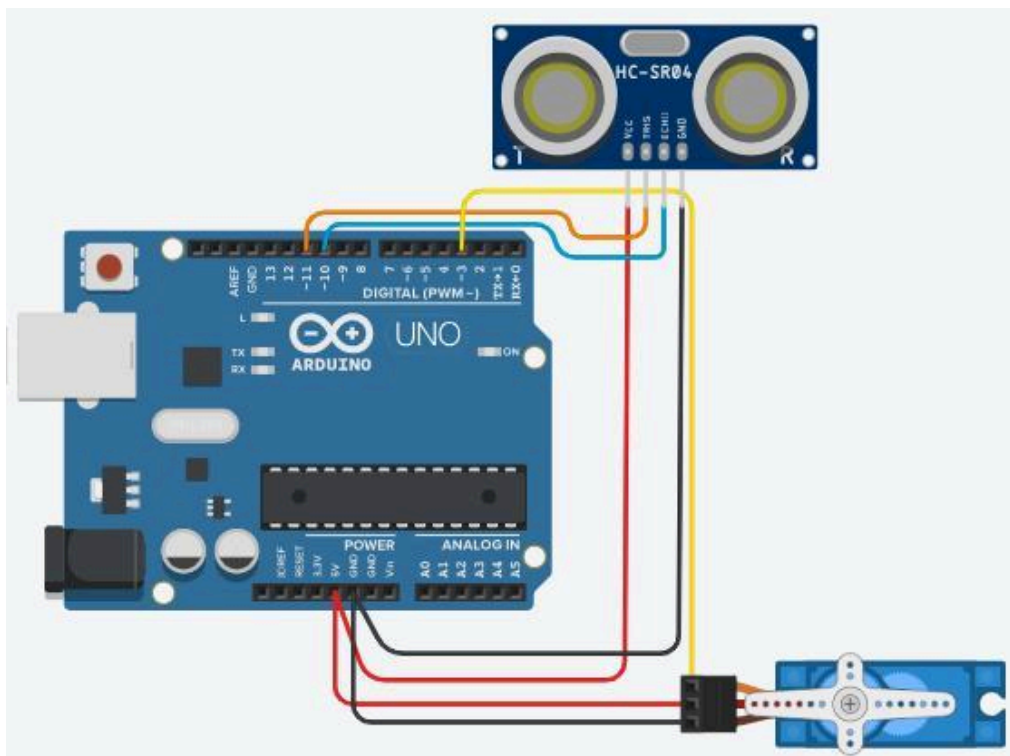
## 2.3 O Sensor: Sensor Ultrassônico

Este sensor será os "olhos" do nosso oponente. Ele funciona como um sonar de morcego: emite um pulso de som (pelo pino **Trig**) e mede o tempo que o eco leva para retornar (pelo pino **Echo**). Com base nesse tempo, ele calcula a distância de um objeto à sua frente. Para simplificar, usaremos uma função pronta que já nos devolve a distância em centímetros.

## 3. Mão na Massa! Construindo nosso Oponente Robótico

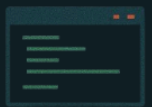
### 3.1 Simulação no Tinkercad

Acesse o [modelo de simulação](#) no Tinkercad.



Vamos analisar o código base disponibilizado. Ele já contém as configurações iniciais que você precisa. Sua missão será completar a função `loop()`.

Acompanhem a simulação (resultado esperado). Este é um bom momento para tirar as dúvidas sobre o funcionamento do programa!



**Observem:** Ao contrário da programação para computadores, em que as variáveis globais são desaconselhadas, no arduino elas são mais que bem-vindas: são fundamentais!

## 3.2 O Desafio: Implementando a função `loop()`

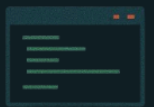
Agora é com vocês! Usem a lógica que já conhecem de C para preencher a função `loop()`. Pensem no fluxo do jogo:

- **Esperar o jogador:** O jogo só começa quando o jogador posiciona a mão. Como podemos verificar isso?
  - **Dica:** Use a função `verifica_distancia()` dentro de um `if`. O jogo pode começar se a distância for menor que um certo valor (ex: 20 cm).
- **O computador joga:** Uma vez que a mão do jogador é detectada, o computador precisa fazer sua jogada.
  - **Dica:** Use a função `random(0, 3)` para sortear um número entre 0 e 2, representando Pedra, Papel e Tesoura.
- **Mostrar a jogada:** O computador precisa mover o servo para a posição correta.
  - **Dica:** Use `if/else` para verificar o número sorteado e chamar `servo.write()` com o ângulo correspondente (0, 90 ou 180).
- **Pausa e Reset:** Após mostrar a jogada, o sistema precisa esperar um pouco antes de estar pronto para a próxima rodada.
  - **Dica:** Use a função `delay(3000);` para pausar por 3 segundos. Depois, mova o servo de volta para uma posição neutra.

## 3.3 Montagem do Circuito

Após a implementação da lógica e simulação no Tinkercad, chegou a hora de dar “vida” ao nosso robô! Vamos montar o circuito com cuidado, seguindo o esquema abaixo. É uma boa prática montar tudo com o Arduino **desligado** do computador.

- **Passo 1 - Conectar o Servomotor:**
  - Fio Marrom (GND) -> Pino GND do Arduino.
  - Fio Vermelho (VCC) -> Pino 5V do Arduino.



- Fio Laranja (Sinal) -> Pino Digital 3 do Arduino.
- **Passo 2 - Conectar o Sensor Ultrassônico:**
  - Pino GND -> Pino GND do Arduino.
  - Pino VCC -> Pino 5V do Arduino.
  - Pino Trig -> Pino Digital 11 do Arduino.
  - Pino Echo -> Pino Digital 10 do Arduino.

### 3.3 Upload do código para o arduino

Conecte o Arduino no computador e abra o programa “Arduino IDE”.

Copie o código (funcional e testado) escrito no Tinkercad e cole na IDE. Selecione a placa “Arduino Uno” e clique no botão de Upload (botão com uma seta).

Os leds do arduino irão piscar algumas vezes e pronto! Já podemos jogar com o nosso robô!

## 4. Indo além!

Nosso oponente robótico funciona, mas como poderíamos melhorá-lo? Quais modificações poderiam ser feitas para marcarmos pontos ou construirmos um placar?

**SPOILER  
ALERT**

**Próximas sessões: Jogo da forca (Roda a roda Jequiti)**

