

## Missão 4.0: Arduino PID Ballance Ball

Rogério Moreira Almeida

### Introdução:

A Missão 4.0 teve como objetivo a construção de um projeto de bancada intitulado Ballance Ball, utilizando Arduino, servo motor e sensor ultrassônico, além da implementação de um sistema de controle em malha fechada com controle PID. Este relatório documenta o processo de construção do projeto, os desafios encontrados e os resultados obtidos.

### Materiais Utilizados:

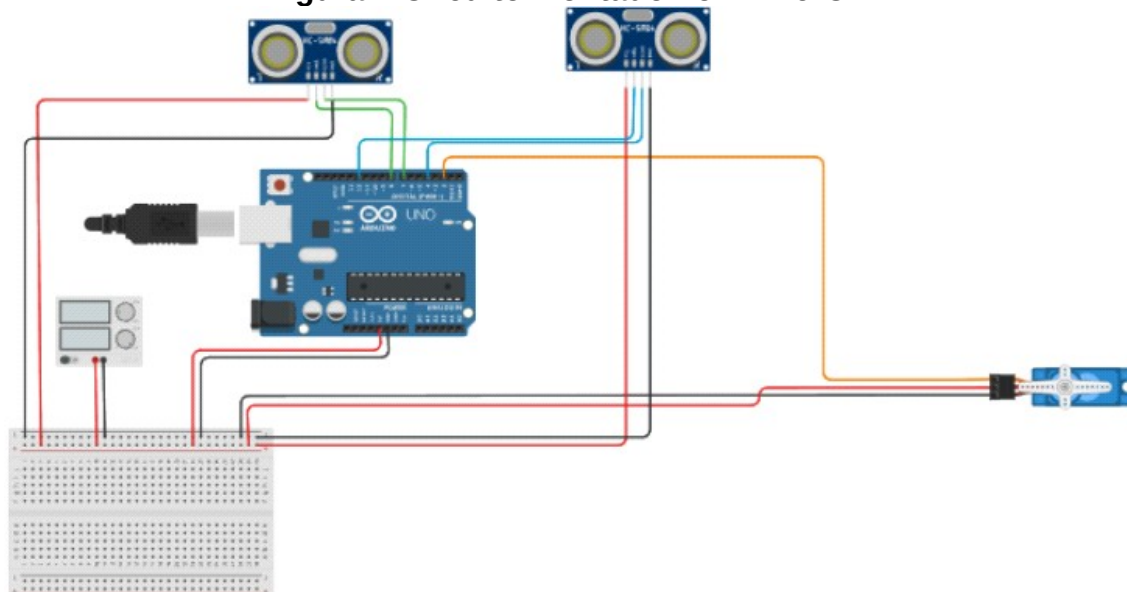
Computador  
AutoCAD  
TinkerCAD  
Arduino  
Servo Motor  
Sensor Ultrassônico

### Desenvolvimento:

#### Replicação do Circuito no TinkerCAD:

Utilizamos o TinkerCAD para replicar o circuito da bancada, incluindo o Arduino, o sensor ultrassônico e o servo motor. Isso nos permitiu simular o funcionamento do projeto antes da construção física.

**Figura1: Circuito montado no TinKerCAD**



Fonte: Autoria própria

### Programação do Arduino:

Programamos o Arduino para ler dados do sensor ultrassônico e calcular a distância dos objetos. Além disso, implementamos o controle da posição do servo motor de acordo com a entrada dos sensores.

// C++ code

```
#include<Servo.h> //biblioteca do servo motor
#define TRIG 8 //pino do sensor 1
#define ECHO 7 //pino do sensor 1
#define TRIGb 12 //pino do sensor 2
#define ECHOb 4 //pino do sensor 2
//motor
Servo SINAL;
unsigned long tempo;
int distancia;
unsigned long tempob;
int distanciab;
int n=90; //iniciamos o servo com 90° porque o servo só percorre angulos possitivos 0°
ao 360°
void setup()
{
  pinMode (TRIG, OUTPUT);
  pinMode (ECHO, INPUT);
  pinMode (TRIGb, OUTPUT);
  pinMode (ECHOb, INPUT);
  SINAL.attach(2); //motor
  Serial.begin(9600);
}
void loop()
{
  //medindo o tempo entre o som bater no objeto e voltar até o sensor 1
  digitalWrite(TRIG,HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG,LOW);
    tempo = pulseIn (ECHO, HIGH, 23529);
  //medindo o tempo entre o som bater no objeto e voltar até o sensor 2
  digitalWrite(TRIGb,HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIGb,LOW);
    tempob = pulseIn (ECHOb, HIGH, 23529);
  distancia = tempo/58;
  distanciab =(tempob)/58;
  delay(100);

  //Comparativos entre a ditancia do objeto e o sensor 1 e entre o carrinho e o sensor 2
```

```

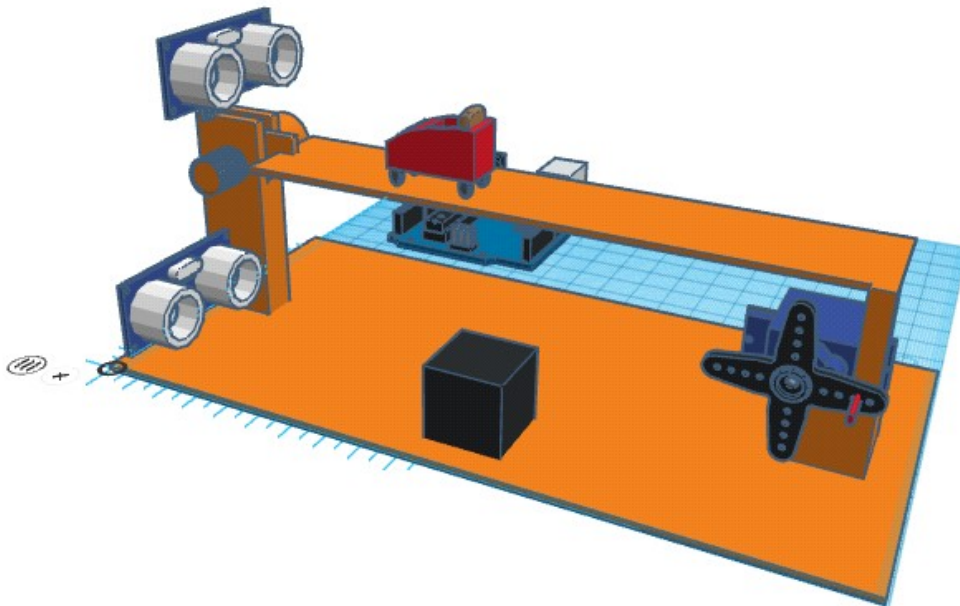
if (distancia > distanciab){
  Serial.println("levanta plataforma");
  n++; //aumentando 1° para elevarmos a rampa
}else if(distancia < distanciab){
  Serial.println("baixa plataforma");
  n--; //subtraímos 1° para abaixamos a rampa
}else{
  Serial.println("mantem plataforma");
}
SINAL.write(n);
                        delayMicroseconds(2000); // tempo para voltamos a reazar o loop
}

```

### Modelagem 3D da Plataforma:

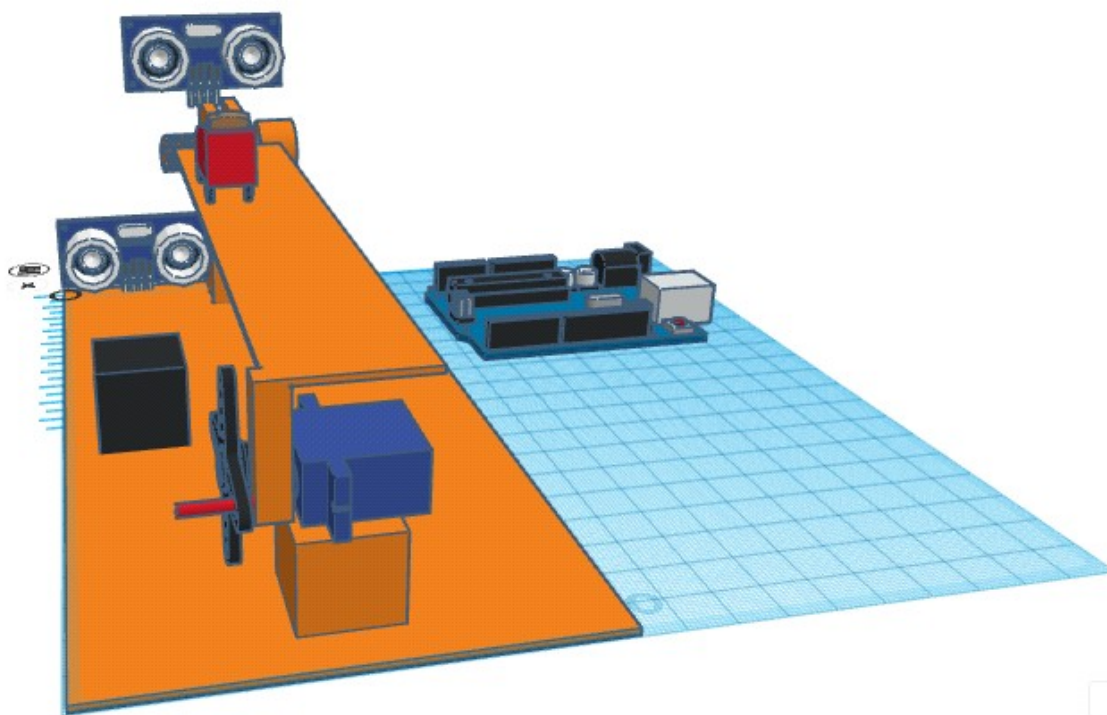
Utilizamos o TinkerCAD para desenhar o modelo da plataforma onde a bola seria colocada. Esse modelo serviu como base para a construção física da bancada.

**Figura2: Plataforma modelada no TinKerCAD**



Fonte: Autoria própria

**Figura3: Plataforma modelada no TinKerCAD**

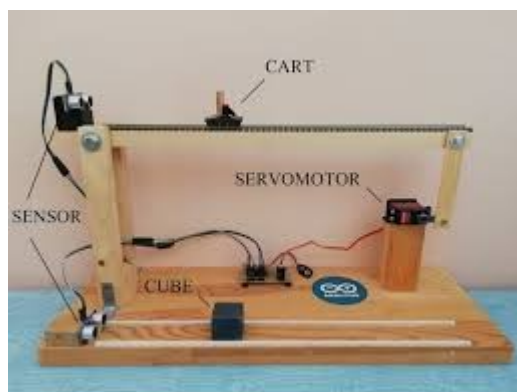


Fonte: Autoria própria

### **Construção do Protótipo:**

Infelizmente só foi contruido a parte eletrica do prototipo, onde foi observado os resultados simulados .

**Figura4: Modelo do Sistema real a ser construido**



Fonte: RAS UFCG

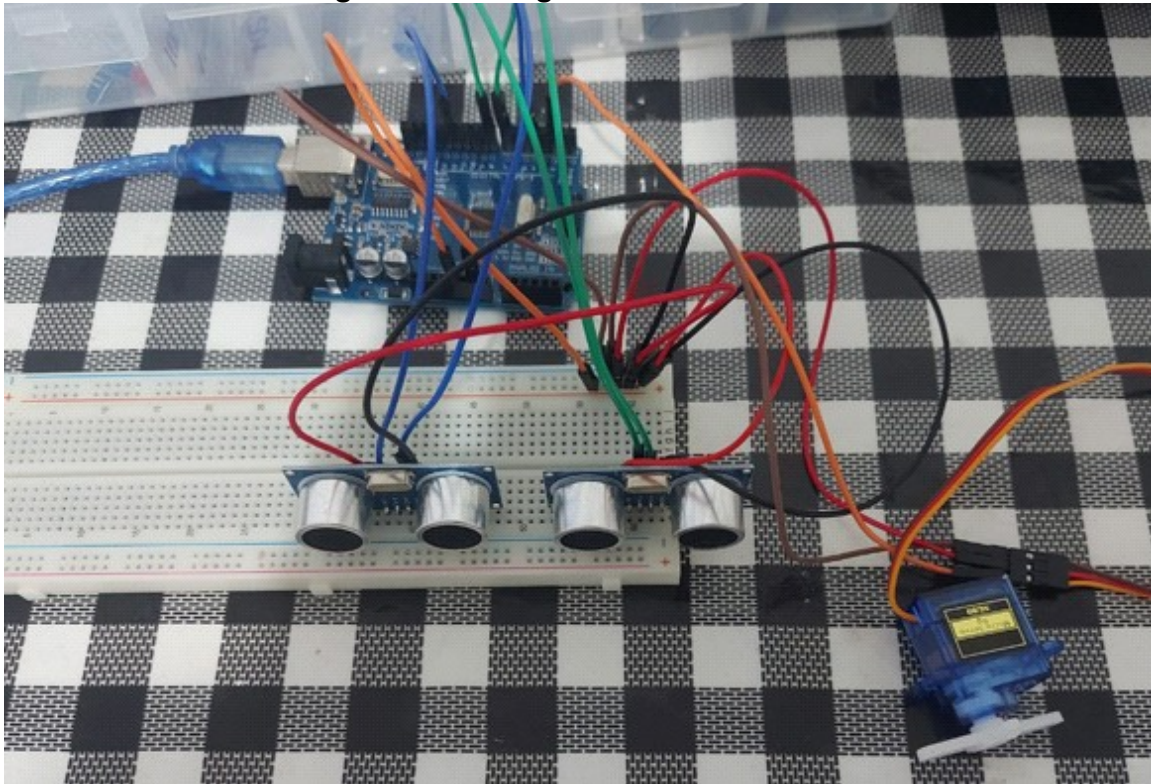
**Estudo de Conceitos Teóricos:**

Dedicamos um tempo para estudar os conceitos de realimentação, malha fechada e controle PID, fundamentais para a implementação do sistema de controle.

**Montagem do Circuito Real:**

Montamos o circuito físico da bancada, utilizando os componentes reais conforme o projeto no TinkerCAD.

**Figura5: Montagem do Circuito Real**



Fonte: Autoria própria

**Implementação do Controle PID:**

Programamos o Arduino para implementar um sistema de controle em malha fechada com controle PID. Isso nos permitiu manter a bola balanceada na plataforma, ajustando a posição do servo motor de acordo com as leituras do sensor ultrassônico.

**Resultados:**

O projeto foi concluído com sucesso, conseguindo manter nas simulações o carrinho balanceado na plataforma de acordo com as leituras do sensor ultrassônico. O sistema de controle PID mostrou-se eficaz para corrigir desvios e manter a bola na posição desejada.

**Conclusão:**

A Missão 4.0 foi uma experiência enriquecedora, que nos permitiu aplicar conhecimentos teóricos de controle de sistemas na prática. A construção do projeto envolveu habilidades de programação, eletrônica e modelagem 3D, culminando na implementação de um sistema funcional e eficiente. Este projeto serviu como uma excelente oportunidade de aprendizado e desenvolvimento de habilidades.

**Anexo:**

Link para a simulação do circuito no TinkerCAD:

<https://www.tinkercad.com/things/kra2KLmZoho-smooth-allis-habbi/editel?returnTo=%2Fdashboard&sharecode=tRsCW8X81mE6tQQ-Kdhuw6-RKRnN8ewPz5-CN5WYUBA>

Link para a modelagem física do circuito no TinkerCAD:

<https://www.tinkercad.com/things/8yUe3APgItx-incredible-blad/edit?returnTo=%2Fdashboard&sharecode=RhRAGYO7kvXgPa7PD3tNBZstoQQvuLJ6y7UadsH-2i4>