## **Aula 02 – Bootcamp de Robótica / Arduino Básico**

Bom.. na aula passada vimos um pouco sobre os módulos bluetooth, sobre buzzer e como gerar sons a partir de comandos do arduino, como exibir informações no display lcd 16 por 2 e por fim foi proposto 2 exercícios opcionais pra vocês fazerem no tinkercad.

### **Objetivos da Aula:**

"Olá a todos! Na aula de hoje, vamos explorar como controlar diferentes tipos de atuadores usando Arduino. Atuadores são componentes que transformam energia elétrica em movimento ou outra forma de energia, permitindo que nossos projetos interajam com o mundo físico. Especificamente, vamos abordar motores DC, servomotores, como usar pontes H para controlar motores, e finalmente, relês para controlar dispositivos de maior potência."

* **Motores DC**

"Vamos começar pelos motores DC, que são os mais simples e comuns.

### O que são motores DC?

Um motor DC converte energia elétrica em movimento rotacional contínuo. São encontrados em brinquedos, ventiladores e muitos outros dispositivos.

* + Características básicas
    - Funcionam com corrente contínua (DC)

A alimentação pode vir de baterias, fontes de alimentação DC ou retificadores que convertem AC em DC.

* + - Têm polaridade (positivo e negativo)

Esta polaridade determina tanto o sentido de rotação quanto o correto funcionamento do motor. Conectar com polaridade invertida fará o motor girar no sentido oposto.

* + - A velocidade depende da tensão aplicada

numa relação diretamente proporcional quase linear, ou seja quanto maior a tensão aplicada no motor, é claro… dentro do range da tensão nominal do motor, maior a sua velocidade

* + Demonstração simples

Agora vou fazer uma demonstração simples do funcionamento do motor DC. Vamos conectar nosso motor DC a uma bateria com tensão de 9 volts. Lembrando que o arduino não consegue fornecer corrente suficiente para um motor DC pelas suas portas digitais. Por isso utilizamos baterias. Nesse circuito também temos um potenciômetro para controlar a corrente no circuito e consequentemente a velocidade do motor. Notem que o motor DC básico tem apenas dois terminais. Quando conectamos diretamente o motor à bateria, ele gira em uma direção. Veja que o valor de rotações por minuto é positivo. Ao inverter a polaridade, ele gira na direção oposta. Note que o valor de rotação é negativo nesse exemplo. No próximo exemplo vamos utilizar o potenciômetro para controlar a velocidade do motor. Quanto maior o valor de resistência do potenciômetro, e por isso menor a corrente. Temos uma velocidade menor do motor. Girando em um rpm menor.

* **Servomotores**

"Agora, vamos avançar para os servomotores, que são um pouco mais sofisticados.

O que são servomotores?

Servomotores são motores que permitem controle preciso de posição angular. Ao invés de girar continuamente como o motor dc, eles se movem para um ângulo específico e mantêm essa posição.

* + Características
    - Geralmente possuem três fios: alimentação, terra e sinal

“Geralmente possuem três fios. O de alimentação que geralmente é 5 volts, o terra ou gnd e sinal, que sai de um dos pinos do nosso arduino

* + - Têm um ângulo de operação limitado (normalmente 0-180 graus)

“Têm um ângulo de operação limitado (normalmente 0-180 graus), mas há servos de rotação contínua que permitem o giro completo e outros servos especiais com ranges de 270 ou 360 graus”

* + - Possuem circuito de controle interno

O servo motor é um sistema completo que inclui um Motor DC pequeno Para gerar o movimento. Sistema de engrenagens. Para redução e aumento do torque. Um Potenciômetro: Para detectar a posição atual do eixo. Um Circuito de controle (PCB): Compara a posição desejada com a atual e um Driver de motor que Controla a direção e velocidade para atingir a posição.

* + - São controlados por pulsos PWM (Modulação por Largura de Pulso)

“São controlados por pulsos PWM. Dependendo da largura do pulso da nossa saída PWM temos um respectivo ângulo de rotação no nosso servo”

* + Vantagens:
    - Controle preciso de posição
    - Torque relativamente alto
    - Fácil de controlar com Arduino
  + Conexão básica:

Vamos conectar um servomotor ao Arduino:

* + - Fio vermelho (ou marrom): 5V do Arduino
    - Fio preto (ou laranja): GND do Arduino
    - Fio amarelo (ou branco): Pino digital do Arduino (usaremos o pino 3)
  + Programação:
  + O Arduino tem uma biblioteca específica para controlar servomotores. Vamos ver como é simples:"
  + #include <Servo.h>
  + Servo meuServo; // Cria um objeto servo
  + void setup() {
  + meuServo.attach(9); // Conecta o servo ao pino 9
  + }
  + void loop() {
  + meuServo.write(0); // Move para posição 0 graus
  + delay(3000); // Espera 1 segundo
  + meuServo.write(90); // Move para posição 90 graus
  + delay(3000);
  + meuServo.write(180); // Move para posição 180 graus
  + delay(3000);
  + }
  + "Vamos fazer a demonstração e observar como o servo se move precisamente para as posições que especificamos no código."
* Controle com Ponte H

"Agora que entendemos os motores básicos, vamos aprender como controlar a direção de um motor DC usando uma ponte H.

* + O que é uma ponte H?

Uma ponte H é um circuito eletrônico que permite aplicar tensão em qualquer direção a uma carga. Para motores, isso significa que podemos controlar tanto a velocidade quanto a direção de rotação. A configuração dos transistores ou chaves no circuito forma um 'H', com o motor no meio da barra horizontal.

* + Funcionamento básico:

Quando fechamos certas chaves na ponte H, a corrente flui em uma direção através do motor. Fechando outras chaves, a corrente flui na direção oposta, e o motor gira ao contrário.

* + Usando a ponte H L293DN:
  + Vamos utilizar o módulo L293D, que é a que temos disponível para demonstração no tinkercad
  + Conexões:
    - OUT1 e OUT2: Conectados aos terminais do primeiro motor
    - OUT3 e OUT4: Conectados aos terminais do segundo motor (se necessário)
    - +12V e GND: Alimentação externa para os motores
    - IN1, IN2, IN3, IN4: Pinos de controle conectados ao Arduino
    - ENA e ENB: Pinos para controle de velocidade (PWM)
  + **Programação**:
    - Vamos ver como controlar um motor em ambas as direções:"
  + // Definição dos pinos
  + const int IN1 = 7;
  + const int IN2 = 8;
  + const int ENA = 9; // Pino PWM para controle de velocidade
  + void setup() {
  + pinMode(IN1, OUTPUT);
  + pinMode(IN2, OUTPUT);
  + pinMode(ENA, OUTPUT);
  + }
  + void loop() {
  + // Motor para frente em velocidade máxima
  + digitalWrite(IN1, HIGH);
  + digitalWrite(IN2, LOW);
  + analogWrite(ENA, 255);
  + delay(2000);
  + // Motor para trás em velocidade média
  + digitalWrite(IN1, LOW);
  + digitalWrite(IN2, HIGH);
  + analogWrite(ENA, 150);
  + delay(2000);
  + // Motor parado
  + digitalWrite(IN1, LOW);
  + digitalWrite(IN2, LOW);
  + delay(1000);
  + }
  + 
* Relés

"Finalmente, vamos aprender sobre relês, que nos permitem controlar dispositivos de alta potência com o Arduino.

* + O que é
  + Um relê é um interruptor eletromagnético.

Quando uma corrente pequena passa pela bobina, ela cria um campo magnético que fecha (ou abre) os contatos de um interruptor, permitindo que uma corrente muito maior flua por um circuito separado.

* + Por que usar relês?
    - Isolamento: O circuito de controle (Arduino) fica isolado do circuito de potência
    - Capacidade de corrente: Podem controlar dispositivos de alta corrente ou tensão
    - Versatilidade: Podem controlar dispositivos AC ou DC
  + Usando O rele LU-5-R
  + Conexões:
    - Comum : 12V da Fonte
    - B1 : Terminal da bobina
    - B2 : Terminal da bobina
    - NO (normalmente aberto)
    - NC (normalmente fechado)

### **Programação:**

Vamos ver como controlar um relê para ligar e desligar um dispositivo:"

const int rele = 7; // Pino do relê

void setup() {

pinMode(rele, OUTPUT);

digitalWrite(rele, HIGH); // Muitos módulos são ativados com LOW

}

void loop() {

digitalWrite(rele, LOW); // Ativa o relê

delay(2000); // Espera 2 segundos

digitalWrite(rele, HIGH); // Desativa o relê

delay(2000); // Espera 2 segundos

}

