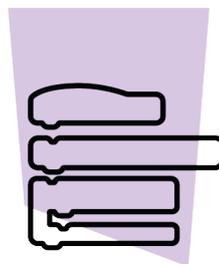
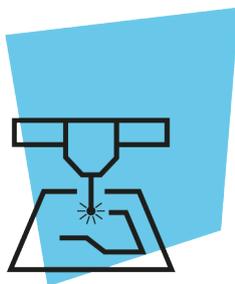
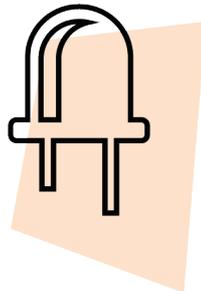
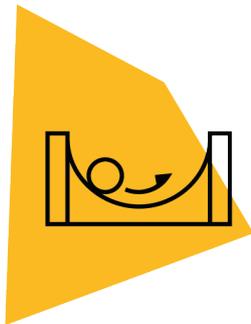
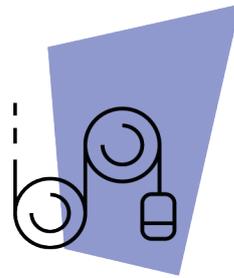
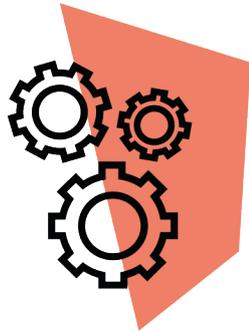


ROBÓTICA

VERSÃO 3

Módulo 2



Braço Robótico

AULA 21

GOVERNADOR DO ESTADO DO PARANÁ

Carlos Massa Ratinho Júnior

SECRETÁRIO DE ESTADO DA EDUCAÇÃO

Renato Feder

DIRETOR DE TECNOLOGIA E INOVAÇÃO

Andre Gustavo Souza Garbosa

COORDENADOR DE TECNOLOGIAS EDUCACIONAIS

Marcelo Gasparin

Produção de Conteúdo

Adilson Carlos Batista

Cleiton Rosa

Validação de Conteúdo

Cleiton Rosa

Revisão Textual

Adilson Carlos Batista

Projeto Gráfico e Diagramação

Edna do Rocio Becker

2021



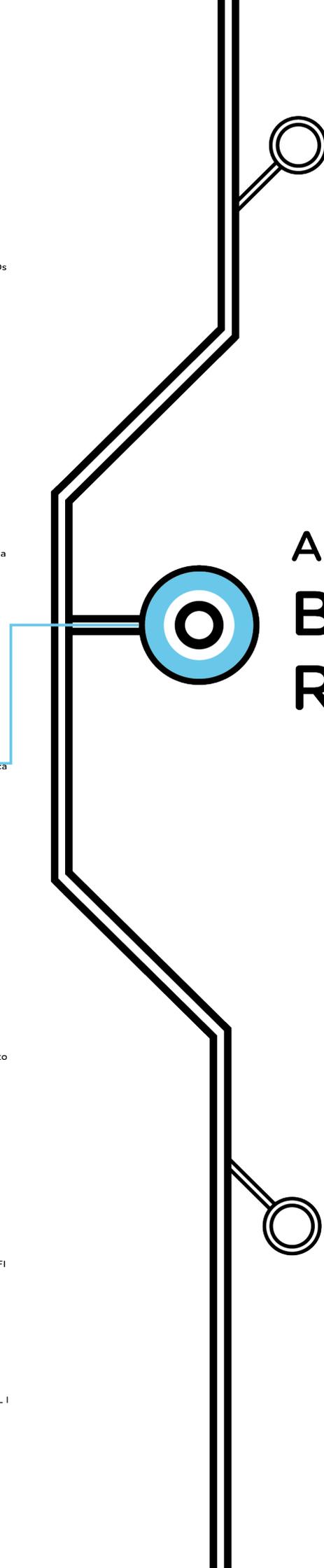
Este trabalho está licenciado com uma Licença Creative Commons
Atribuição NãoComercial - Compartilhável 4.0 Internacional

Aula 01	O que já vimos?
Aula 02	Arduino: Bibliotecas e Funções
Aula 03	Código Morse
Aula 04	Semáforo Inteligente com IR
Aula 05	Semáforo Completo com Display
Aula 06	Matriz de LED 8x8
Aula 07	Desenhando na matriz de LEDs
Aula 08	Painel de Senhas
Aula 09	Escrevendo mensagens
Aula 10	Robô Autônomo
Aula 11	Sensor de Chuva
Aula 12	Sensor de Umidade do Solo
Aula 13	Irrigador Automático
Aula 14	Feedbacks + Inventário I
Aula 15	Teclado Matricial de Membrana
Aula 16	Servos Motores
Aula 17	Fechadura Eletrônica
Aula 18	Controlando Servos Motores
Aula 19	JoyStick Shield
Aula 20	Braço Robótico
Aula 21	Sensor de Movimento Presença
Aula 22	Sensor de Som
Aula 23	Sensor de Umidade e Temperatura
Aula 24	Termômetro Digital
Aula 25	Sensor de Gás e Fumaça
Aula 26	Acelerômetro e Giroscópio
Aula 27	Motor de Passo
Aula 28	Feedbacks + Inventário II
Aula 29	Receptor IR e Controle Remoto
Aula 30	Relé
Aula 31	Módulo RF 433mhz - I
Aula 32	Módulo RF 433mhz - II
Aula 33	Projeto CHAT via RF
Aula 34	Módulo Wireless
Aula 35	Comunicação do Módulo WI-FI em HTML
Aula 36	Módulo WI-FI - IoT com Sensores
Aula 37	Módulo WI-FI - IoT com Atuadores (LED)
Aula 38	Módulo WI-FI - IoT com Atuadores (Relé)
Aula 39	Monitor de Sensores em HTML I
Aula 40	Monitor de Sensores em HTML II
Aula 41	Mostra de Robótica
Aula 42	Feedbacks + Inventário III

Aula 20
Braço Robótico
[Montagem]

Aula 21
Braço Robótico

Aula 22
Sensor de Movimento Presença



Sumário

Introdução	2
Objetivos desta Aula	2
Competências Gerais Previstas na BNCC	3
Habilidades do Século XXI a Serem Desenvolvidas	4
Lista de Materiais	4
Roteiro da aula	5
1. Contextualização	5
2. Montagem e Programação	11
3. Feedback e Finalização	23
Videotutorial	24



Introdução

Você já observou como as cirurgias médicas estão cada vez mais utilizando robôs colaborativos ou braços robóticos para trabalhar com precisão na atualidade? O mesmo tem ocorrido nas grandes indústrias automotivas, químicas, alimentícias, entre outras, tanto para fazer trabalhos repetitivos como para executar ações com maiores precisões, bem como manipular objetos que teriam risco para a vida humana. E você, sabe como esses braços funcionam? Você observou na “Aula 1 Para que Robótica” um exemplo deste braço dançando com uma bailarina na cerimônia de abertura dos Jogos Paraolímpicos no Rio de Janeiro em 2016.

Nesta aula, além de retomarmos uma conversa sobre robôs, iremos montar um braço robótico, utilizando alguns componentes do kit de robótica, e configurar o braço para que ele possa realizar movimentos e executar pequenas ações, ou serviços.



Objetivos desta Aula

- Retomar a história dos robôs;
- Compreender como é um Braço Robótico e quais são suas funcionalidades;
- Programar um braço robótico e suas articulações para colocar em movimento usando um Servo Motor, Placa de Joystick Shield, uma placa Arduino Uno R3 e uma Protoboard.



Competências Gerais Previstas na BNCC

[CG02] - Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.

[CG04] - Utilizar diferentes linguagens – verbal (oral ou visual-motora, como Libras, e escrita), corporal, visual, sonora e digital –, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo.

[CG05] - Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.

[CG09] - Exercitar a empatia, o diálogo, a resolução de conflitos e a cooperação, fazendo-se respeitar e promovendo o respeito ao outro e aos direitos humanos, com acolhimento e valorização da diversidade de indivíduos e de grupos sociais, seus saberes, identidades, culturas e potencialidades, sem preconceitos de qualquer natureza.

[CG10] - Agir pessoal e coletivamente com autonomia, responsabilidade, flexibilidade, resiliência e determinação, tomando decisões com base em princípios éticos, democráticos, inclusivos, sustentáveis e solidários.



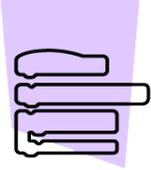
Habilidades do Século XXI a Serem Desenvolvidas

- Pensamento crítico;
- Afinidade digital;
- Resiliência;
- Resolução de problemas;
- Colaboração;
- Comunicação.



Lista de Materiais

- 01 Placa Protoboard;
- 01 Arduino Uno R3;
- 01 Cabo USB;
- 14 Jumpers Macho-Macho;
- 04 Servos;
- 01 Joystick Shield DIY para Arduino;
- 01 Kit Braço MDF;
- 01 Notebook;
- Software Arduino IDE.



Roteiro da Aula

1. Contextualização (15min):

Os robôs passaram a existir em nosso meio no início do século XX, mas suas idealizações começaram bem antes, há relatos de um brinquedo para criança – um cachorro mecânico, datado de 2000 A.C., no Egito antigo. Mas foi mesmo com a Revolução Industrial, ocorrida na Inglaterra no Século XVIII, que o sonho foi impulsionado com o desenvolvimento de grandes máquinas nas indústrias.

A palavra “Robot” surgiu, primeiro, na ficção, em uma peça teatral do dramaturgo tcheco Karel Čapek chamada “R.U.R – Rossum Universal Robot”. A palavra originou da palavra tcheco “Robota” que significa trabalho árduo, duro, e é sinônimo de trabalho forçado. Nesta dramaturgia, um cientista chamado Rossum cria humanos mecanizados para realizar ações repetitivas e pesadas.

O primeiro robô surgiu em 1924, quando um engenheiro elétrico da Westinghouse chamado Roy J. Wensley criou um robô mecânico, ou uma unidade de controle supervisionada. Esse dispositivo, usando o sistema de telefonia, podia ligar, desligar ou regular qualquer coisa que estivesse conectado a ele.

Essas invenções iniciais podem parecer simples para nós que temos contato com Inteligência artificial em casa e na palma da mão, por meio dos smartphones, contando piadas e realizando diversas ações como mudar de canal, ligar e desligar algum equipamento eletrônico, dar a previsão do tempo, entre outras possibilidades, para a época isso foi uma revolução.

Três anos após a criação do robô mecânico, Roy cria um outro chamado Televox, esse apresentava aspecto humano e, de acordo com os comandos de seu operador, realizava movimentos básicos.



Os robôs ganharam popularidade a partir de uma exposição Mundial realizada em Chicago, em 1933, nessa foi apresentado o robô Willie Vocalite, uma espécie de robô controlado por comando de voz. Sua aparência era daquelas dos filmes de ficção antiga, dois metros de altura, feitos de aço, também podia ligar, desligar e regular dispositivos conectados a ele. Além disso, movia os braços, sentava, ficava de pé e conversava com as pessoas repetindo frases gravadas em discos de 78 rotações.

Você poderia pensar que seria igual ao C-3PO, de Star Wars, mas não, esse robô ainda precisaria muito aperfeiçoamento tecnológico para se chegar ao modelo criado pela ficção científica, C-3, além das articulações e movimentos, era diplomata fluente em mais de 6 milhões de línguas e dialetos.

A partir daí os robôs passam a ser sucesso em todo o mundo e, cada vez mais, esses “pequenos grandes” têm se tornando parte da vida dos humanos na execução de serviços e ações.

Depois dos robôs teleoperados, foi criado em 1954, por George Devol, o primeiro robô automático, o Unimate, esse modelo passou a ser utilizado na indústria da General Motors em 1961. Ele executava serviços de pegar pedaços de metal quente e colar nos chassis dos carros, ele suportava até 1800 Kg e seus comandos eram gravados em fitas magnéticas.



BRAÇO ROBÓTICO

Em 1966, veio através do MIT (Massachusetts Institute of Technology) uma inteligência artificial chamada Eliza, desenvolvida por Joseph Weizenbaum (Americano). Três anos depois, 1969, um estudante de Engenharia Mecânica chamado Victor Scheinman, também estadunidense, desenvolveu um braço mecânico e hidráulico “Stanford Arm”.



Fonte: Wikimedia Commons
https://commons.wikimedia.org/wiki/File:The_Stanford_Arm.jpg

Em 1972, foi criado outro modelo, Braço do MIT, aperfeiçoando o modelo anterior, esse era mais compacto e possuía uma estrutura de concha feita de folha de metal, em vez de vigas, que continha toda a fiação. Ele também usou trens de engrenagens, especialmente projetados, em parte, para minimizar a folga e motores elétricos personalizados, em vez de apenas componentes prontos para uso.



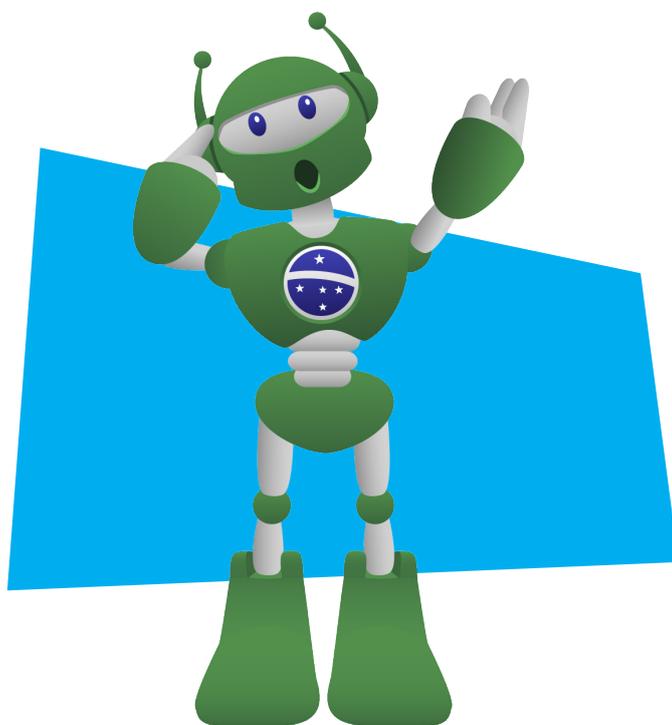
Fonte: Wikimedia Commons
https://commons.wikimedia.org/wiki/File:Scheinman_MIT_Arm.agr.jpg



A corrida de descobertas continuou e, em 1981, o engenheiro Takeo Kanade desenvolveu e montou o primeiro braço mecânico com motor instalado diretamente nas junções do braço. Esta mudança fez com que os movimentos se tornassem mais rápidos e precisos.

Depois disso, os modelos criados foram cada vez mais aperfeiçoados, misturando inteligência artificial e mecânica. Em 1986, a Honda criou um robô chamado Asimo, totalmente humanoide que caminhava lentamente em linha reta, colocando um pé diante do outro. Essa criação foi uma revolução e a empresa passou a aperfeiçoá-lo cada vez mais, de 1991 a 1997 eles conseguiram dar mais mobilidade ao robô, articulando coluna e membros.

No ano de 2000, esse robô ficou conhecido mundialmente, com um metro e trinta de altura, quarenta e oito kilos, reconhecendo rostos e sons, cumprimentar com aperto de mão, caminhar de mãos dadas com humanos, correr até 9 km por hora, caminhar em pisos irregulares, subir e descer escadas, foi um sucesso.



A tecnologia utilizada no Asimo foi utilizada em outros projetos da empresa, inclusive desenvolver um braço robótico para auxiliar na recuperação do reator nuclear da usina de Fukushima.

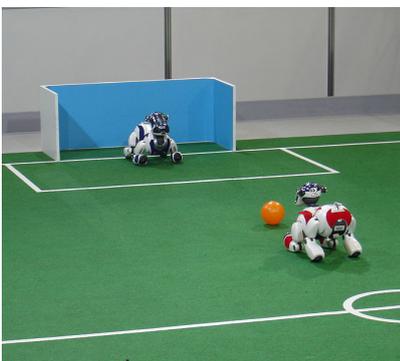
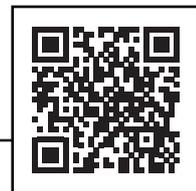
Fonte: Wikimedia Commons
https://commons.wikimedia.org/wiki/File:ASIMO_4.28.11.jpg



Sugestão de Vídeo:

Veja a história do Asimo e confira todos os movimentos que ele realiza:

<https://youtu.be/eKvwgmHFwhc>
(2min03)



Fonte: Wikimedia Commons
Adaptado de https://upload.wikimedia.org/wikipedia/commons/3/31/Robocup_2005_Aibos.jpg

A partir de 2007, as indústrias de Braços Robóticos chegam a sua maturidade, os braços possuem velocidade, precisão, facilidade de manipulação de diferentes produtos, controle por vídeo etc. Ainda precisa melhorar mais a flexibilidade das mãos, mas esses robôs são de grande importância para a humanidade, cada vez mais são utilizados nas indústrias automotivas, químicas, alimentícias, entre outras.

Em Curitiba, o Hospital Nossa Senhora das Graças tem o robô Da Vinci XI, um dos melhores robôs do mundo para realizar cirurgias de alta complexidades, o robô cirúrgico permite realizar atos precisos, além daqueles que se conseguiria com as mãos humanas.



Robô Da Vinci XI

Fonte: <https://www.hnsg.org.br/curitiba-ganha-ultima-versao-do-robot-da-vinci/>

Imagem: <https://www.hnsg.org.br/wp-content/uploads/2019/12/davinci-xi-surgical-system.png>

E você, quer aprender mais sobre robótica e programação? Então, vamos em frente! Agora é com você! Vamos colocar o **Braço Robótico para funcionar!**



Para Saber Mais...

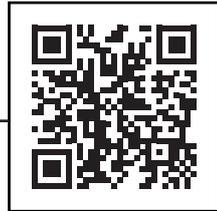
Sobre Joseph Weizenbaum



https://pt.wikipedia.org/wiki/Joseph_Weizenbaum

Sobre a inteligência Eliza

<https://pt.wikipedia.org/wiki/ELIZA>



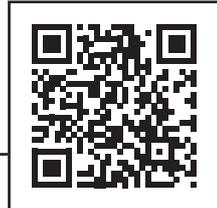
Sobre Victor Scheinman



https://en.wikipedia.org/wiki/Victor_Scheinman

Sobre o robô Asimo

<https://pt.wikipedia.org/wiki/ASIMO>



Sobre o robô “paranaense” Da Vinci

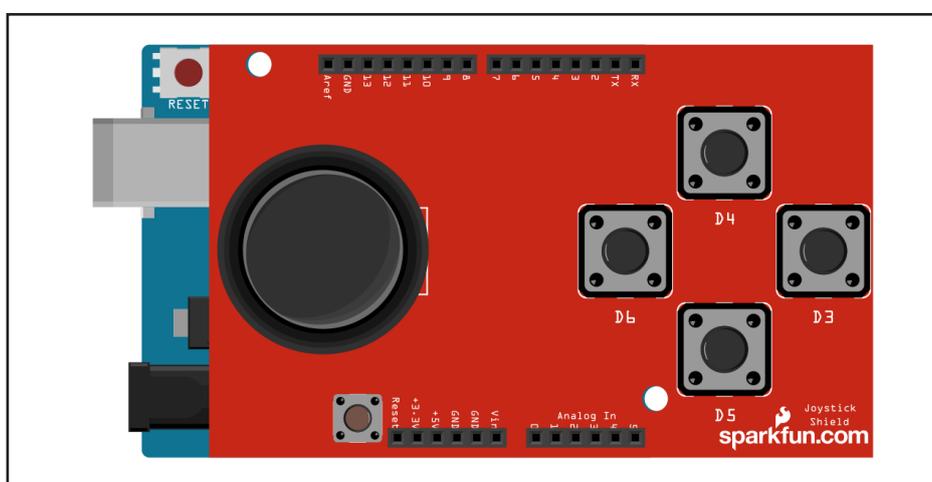


<https://www.hnsg.org.br/curitiba-ganha-ultima-versao-do-robo-da-vinci/>

2. Montagem e Programação (60min):

Primeiramente, vamos iniciar pelos componentes eletrônicos. Pegue a Joystick Shield e acople sobre a Placa Arduino para que possamos controlar os Servos que serão inseridos, figura 1.

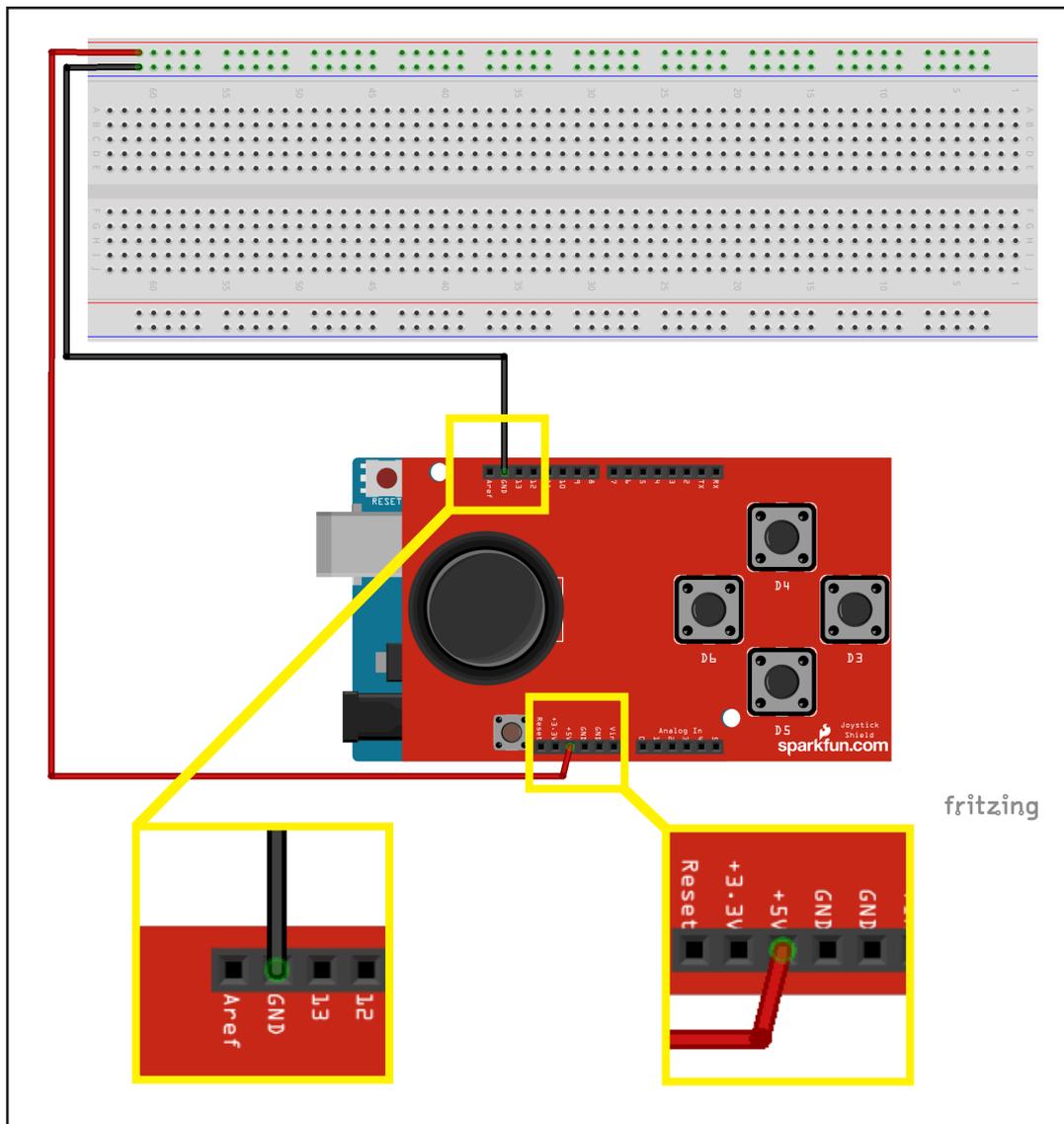
Figura 1 - Acoplagem do Joystick Shield ao Arduino.



Depois, fazemos as conexões entre a Placa de Protoboard e a Placa Joystick Shield, que está acoplada ao Arduino Uno R3, por meio de Jumpers. Para isso, conecte 1 Jumper Macho-Macho entre a porta GND do Joystick Shield e a linha azul da Protoboard. Na sequência, conecte outro Jumper Macho-Macho entre a porta 5V do Joystick e a linha vermelha da Protoboard, como mostra a figura 2.



Figura 2 - Ligação de Jumpers Macho-Macho à placa de Protoboard e Joystick Shield

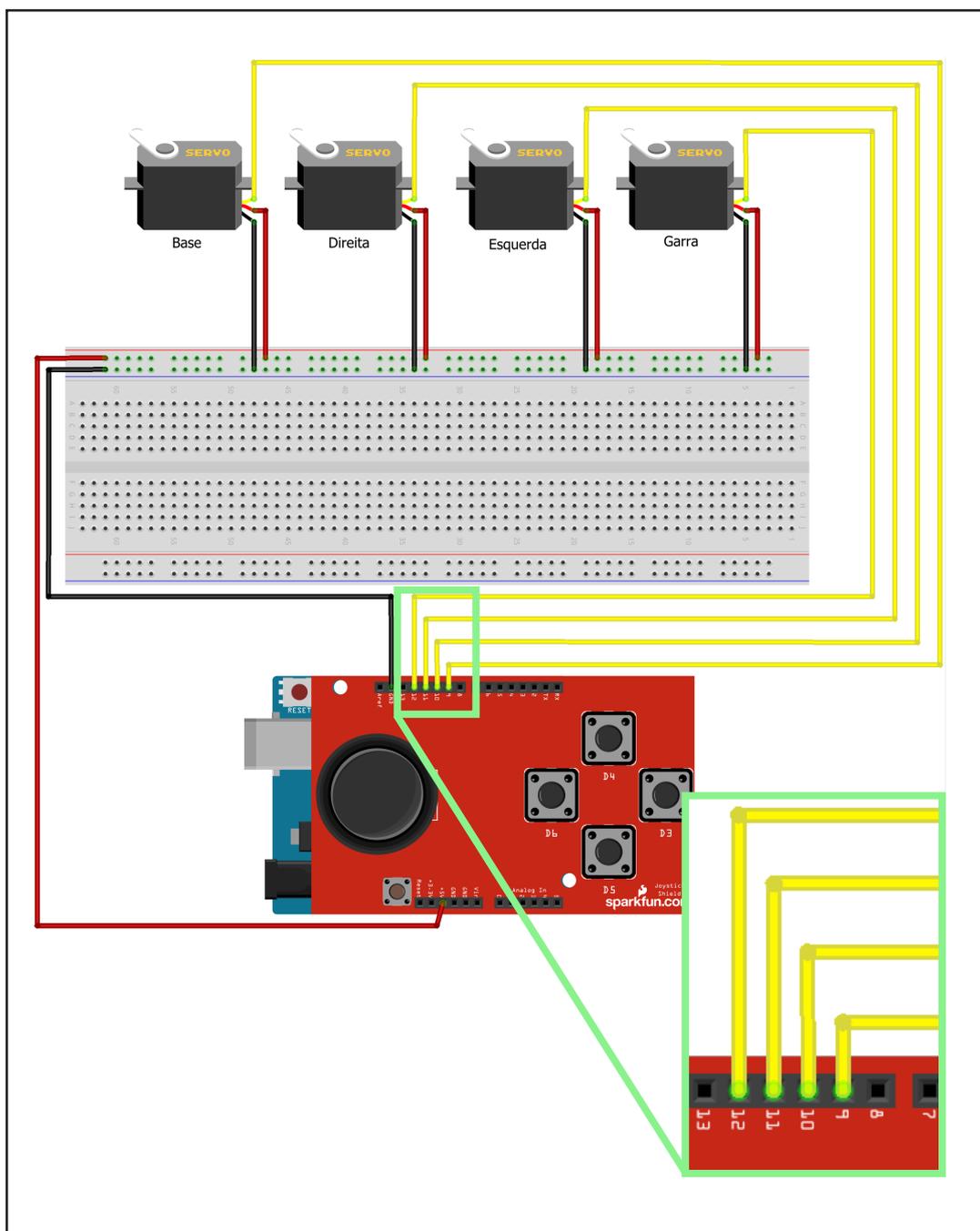


Por último, conectamos os Servos por meio de 12 Jumpers, às placas Arduino e Protoboard. Para tanto, ligamos 3 Jumpers aos terminais de cada Servo.

Os Jumpers conectados aos terminais marrons são ligados às linhas azuis da Protoboard, enquanto os Jumpers inseridos nos terminais vermelhos dos Servos são conectados às linhas vermelhas da

Protoboard. O terceiro Jumper conectado em cada Servo é interligado, respectivamente, às portas digitais 9, 10, 11 e 12 do Arduino, conforme demonstra a figura 2.

Figura 3 - Conexão dos Servos às Placas Protoboard e Arduino





Agora, vamos programar!

Com os componentes eletrônicos montados, vamos programar, por codificação, o funcionamento do Braço Robótico.

Linguagem de programação por código

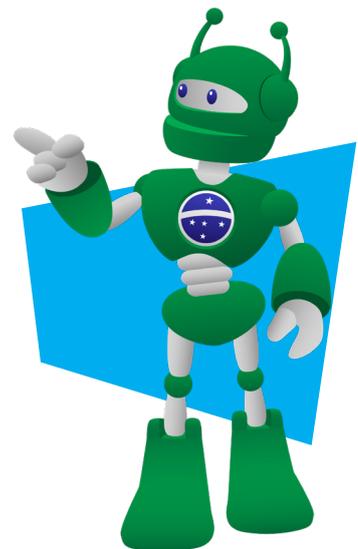
Para iniciar a programação, conecte a placa Arduino ao computador, através de um cabo USB, para que ocorra a comunicação entre a placa microcontroladora e o software Arduino IDE.

Nesta programação, utilizaremos a biblioteca “Servo”. Essa biblioteca não necessita ser instalada, pois vem instalada na IDE Arduino e auxiliará no controle dos Servos Motores.

No software IDE, crie um sketch e lembre-se de selecionar a porta que o computador atribuiu ao Arduino; então, escreva ou copie e cole o código-fonte de programação, conforme apresentado no quadro 1.

Atenção!

Ao copiar o código diretamente do pdf, evite quebra da página (e conseqüentemente erros na compilação), copiando o código por partes.



Quadro 1 - Código-fonte da programação na linguagem do Arduino (Wiring)

```
/* **** */
/* Aula 21 - Braço Robótico */
/* Abaixo um exemplo de código controlará o */
/* braço robótico utilizando a placa joystick*/
/* shield. A seleção do servomotor a ser */
/* controlado é feita através dos botões A, */
/* B, C e D. O movimento da base e da garra */
/* do braço é feito através do movimento do */
/* joystick na direção do eixo x e dos */
/* servomotores das laterais do braço é feito*/
/* movendo o joystick no eixo y. */
/* **** */
/* Inclui a biblioteca Servo. */
#include <Servo.h>
/* Definição das portas dos botões. */
#define btA 2
#define btB 3
#define btC 4
#define btD 5

/* Variáveis auxiliares. */
int potA0, potA1, incremento = 1;
char escolha;

/* Definindo o ângulo inicial dos servos. */
int angBase = 90;
int angDir = 90;
int angEsq = 90;
int angGarra = 90;

/* Criando objetos de controle dos servos. */
Servo servo_9;
Servo servo_10;
Servo servo_11;
Servo servo_12;
```

```
void setup() {
  /* Configurando o modo de operação das          */
  /* portas dos botões ativando o resistor         */
  /* interno do Arduino.                          */
  pinMode(btA, INPUT_PULLUP);
  pinMode(btB, INPUT_PULLUP);
  pinMode(btC, INPUT_PULLUP);
  pinMode(btD, INPUT_PULLUP);
  /* Definindo as portas dos servos.             */
  servo_9.attach(9);
  servo_10.attach(10);
  servo_11.attach(11);
  servo_12.attach(12);
  /* Posiciona os servos no angulo inicial.      */
  servo_9.write(angBase);
  servo_10.write(angDir);
  servo_11.write(angEsq);
  servo_12.write(angGuarra);
}
void loop() {
  /* Guarda os valores analógicos dos            */
  /* potenciômetros do joystick.                 */
  potA0 = analogRead(A0);
  potA1 = analogRead(A1);
  /* Se for pressionado o botão A.              */
  if (digitalRead(btA) == 0) {
    /* Defina a escolha como A.                 */
    escolha = 'A';
  }
  /* Se for pressionado o botão B.              */
  if (digitalRead(btB) == 0) {
    /* Defina a escolha como B.                 */
    escolha = 'B';
  }
  /* Se for pressionado o botão C.              */
  if (digitalRead(btC) == 0) {
    /* Defina a escolha como C.                 */
    escolha = 'C';
  }
}
```

```
/* Se for pressionado o botão D. */
if (digitalRead(btD) == 0) {
  /* Defina a escolha como D. */
  escolha = 'D';
}
/* Após um botão ser pressionado, entrará */
/* em um dos quatro modos de operação (A, */
/* B, C e D). */
switch (escolha) {
  /* Caso a escolha for A. */
  case 'A':
    /* Se o joystick for deslocado para a */
    /* esquerda e o ângulo do servo da */
    /* garra for menor que 100°, abrirá a */
    /* garra. */
    if (potA0 > 700 && angGarra < 100) {
      /* Posiciona o servo no ângulo atual */
      servo_12.write(angGarra);
      /* Incrementa o valor do ângulo. */
      angGarra = angGarra + incremento;
      /* Aguarda 50 milissegundos para */
      /* atualizar o ângulo. */
      delay(50);
    }
    /* Se o joystick for deslocado para a */
    /* direita e o ângulo do servo da */
    /* garra for maior que 60°, fechará a */
    /* garra. */
    if (potA0 < 300 && angGarra > 60) {
      /* Posiciona o servo no ângulo atual.*/
      servo_12.write(angGarra);
      /* Decrementa o valor do ângulo. */
      angGarra = angGarra - incremento;
      /* Aguarda 50 milissegundos para */
      /* atualizar o ângulo. */
      delay(50);
    }
    /* Final desse caso. */
    break;
  }
}
```

```
/* Caso a escolha for B. */
case 'B':
  /* Se o joystick for deslocado para */
  /* frente e o ângulo do servo da */
  /* direita for menor que 180°, */
  /* abaixará o braço. */
  if (potA1 > 700 && angDir < 180) {
    /* Posiciona o servo no ângulo atual.*/
    servo_10.write(angDir);
    /* Incrementa o valor do ângulo. */
    angDir = angDir + incremento;
    /* Aguarda 50 milissegundos para */
    /* atualizar o ângulo. */
    delay(50);
  }
  /* Se o joystick for deslocado para */
  /* trás e o ângulo do servo da direita */
  /* for maior que 50°, levantará o braço.*/
  if (potA1 < 300 && angDir > 50) {
    /* Posiciona o servo no ângulo atual.*/
    servo_10.write(angDir);
    /* Decrementa o valor do ângulo. */
    angDir = angDir - incremento;
    /* Aguarda 50 milissegundos para */
    /* atualizar o ângulo. */
    delay(50);
  }
  /* Final desse caso. */
  break;
/* Caso a escolha for C. */
case 'C':
  /* Se o joystick for deslocado para a */
  /* direita e o ângulo do servo da base */
  /* for menor que 180°, o braço girará */
  /* no sentido horário. */
```

```
if (potA0 < 300 && angBase < 180) {
  /* Posiciona o servo no ângulo atual.*/
  servo_9.write(angBase);
  /* Incrementa o valor do ângulo.      */
  angBase = angBase + incremento;
  /* Aguarda 50 milissegundos para     */
  /* atualizar o ângulo.                 */
  delay(50);
}
/* Se o joystick for deslocado para a  */
/* esquerda e o ângulo do servo da     */
/* base for maior que 0°, o braço      */
/* girará no sentido anti-horário.    */
if (potA0 > 700 && angBase > 0) {
  /* Posiciona o servo no ângulo atual.*/
  servo_9.write(angBase);
  /* Decrementa o valor do ângulo.    */
  angBase = angBase - incremento;
  /* Aguarda 50 milissegundos para     */
  /* atualizar o ângulo.                 */
  delay(50);
}
/* Final desse caso.                   */
break;
/* Caso a escolha for D.                */
case 'D':
  /* Se o joystick for deslocado para  */
  /* frente e o ângulo do servo da     */
  /* esquerda for menor que 180°,      */
  /* levantará o antebraço.           */
  if (potA1 > 700 && angEsq < 180) {
    /* Posiciona o servo no ângulo atual.*/
    servo_11.write(angEsq);
    /* Incrementa o valor do ângulo.    */
    angEsq = angEsq + incremento;
    /* Aguarda 50 milissegundos para     */
    /* atualizar o ângulo.                 */
    delay(50);
  }
}
```

```
/* Se o joystick for deslocado para */
/* trás e o ângulo do servo da */
/* esquerda for maior que 90°, */
/* abaixará o antebraço. */
if (potA1 < 300 && angEsq > 90) {
  /* Posiciona o servo no ângulo atual.*/
  servo_11.write(angEsq);
  /* Decrementa o valor do ângulo. */
  angEsq = angEsq - incremento;
  /* Aguarda 50 milissegundos para */
  /* atualizar o ângulo. */
  delay(50);
}
/* Final desse caso. */
break;
}
}
```

Com o código-fonte inserido no Arduino IDE, compile o programa pressionando o botão **Verificar** para examinar se não há erros de sintaxe.

Estando o código correto, pressione o botão **Carregar** para realizar a transferência do programa para o Arduino.

Após a transferência do programa para o Arduino, este ativa o joystick, informando ao Servo qual ângulo deve ocupar, formar ou ir. Cada Servo é responsável por um movimento: para cima e para baixo; da esquerda para direita; abertura e fechamento da garra; e rotação do pulso. A execução de cada movimento é determinada pelo acionamento de cada chave botão de pressão (formato de cruz) da placa de joystick Shield.



Desafio:

Que tal fazer este braço, que está sendo controlado pelo Joystick Shield, ficar autônomo? Faça uma programação para que o braço pegue um objeto de um lado da estrutura e o transporte para o outro lado, em um movimento de 180 graus da base, e fazendo movimentos suaves para que o braço não derrube o objeto que está sendo transportado, nem no momento de pegá-lo e nem no momento de soltá-lo.



E se...?

O projeto não funcionar, se atente a alguns dos possíveis erros:

- a.** Verifique se os Jumpers estão nos pinos certos, se estão na mesma coluna dos terminais dos componentes, fazendo assim as conexões;
- b.** Verifique o correto acoplamento da Joystick Shield ao Arduino;
- c.** Verifique se a programação está adequada a cada porta digital.

3. Feedback e Finalização (15min):

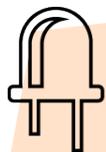
- a. Confira, compartilhando seu projeto com os demais colegas, se o objetivo foi alcançado.
- b. Analise seu projeto desenvolvido, de modo a atender aos requisitos para funcionamento de um braço robótico.
- c. Reflita se as seguintes situações ocorreram:
 - i. Colaboração e Cooperação: você e os membros de sua equipe interagiram entre si, compartilhando ideias que promoveram a aprendizagem e o desenvolvimento deste projeto?
 - ii. Pensamento Crítico e Resolução de Problemas: você conseguiu identificar os problemas, analisar informações e tomar decisões de modo a contribuir para o projeto desenvolvido?
- d. Reúna todos os componentes utilizados nesta aula e os organize novamente, junto aos demais, no kit de robótica.



AULA

21

BRAÇO ROBÓTICO



Videotutorial

Com o intuito de auxiliar na montagem e na programação desta aula, apresentamos um videotutorial, disponível em:



<https://rebrand.ly/a21robotica2>

Acesse, também, pelo QRCode:



