

Oficina de Robótica Educacional

Planejando a Prática Educativa

Apresentação da Proposta

Neste material apresentamos o desenvolvimento de uma oficina e de um tutorial de robótica. A oficina e o tutorial são produtos educacionais frutos de pesquisa realizada no Mestrado Profissional em Educação Profissional e Tecnológica (ProfEPT), com polo no Instituto Federal de Alagoas (IFAL), que resultou na dissertação intitulada "Oficina de robótica educacional como mecanismo de articulação entre teoria e prática no curso Técnico em Eletrônica". A oficina compreende todos os procedimentos e a dinâmica desenvolvidos durante a experiência educativa, enquanto o tutorial de montagem e programação de robôs se constitui um instrumento de suporte educacional, que pode ser utilizado na oficina para contribuir no processo de construção do conhecimento.

Fernanda Cordeiro dos Santos
Geraldo Alves Sobral Júnior



Dados Internacionais de Catalogação na Publicação
Instituto Federal de Alagoas
Campus Avançado Benedito Bentes
Biblioteca

S237o

Santos, Fernanda Cordeiro dos.

Oficina de robótica educacional – planejando a prática educativa / Fernanda Cordeiro dos Santos. – 2020.

119 f. : il.

1 CD-ROM: il.

Produto Educacional da Dissertação – Oficina de robótica educacional como mecanismo de articulação entre teoria e prática no curso técnico em eletrônica (Mestrado em Educação Profissional e Tecnológica) - Instituto Federal de Alagoas, Campus Avançado Benedito Bentes, Maceió, 2020.

1. Robótica Educacional. 2. Ensino de Ciência. 3. Produto Educacional. I. Título.

CDD:371.3

Fernanda Isis Correia da Silva
Bibliotecária - CRB-4/1796

SUMÁRIO

1	DESENVOLVIMENTO DO PRODUTO EDUCACIONAL.....	04
1.1	Planejamento da Prática Educativa da Oficina.....	04
1.1.1	Sequências de Atividades de Ensino Aprendizagem.....	04
1.1.2	Conteúdos de Aprendizagem.....	05
1.1.3	Relação Professor-Aluno.....	06
1.1.4	Organização Social da Aula.....	06
1.1.5	Organização dos Conteúdos de Aprendizagem.....	07
1.1.6	Materiais Curriculares.....	07
1.1.7	Avaliação.....	08
1.2	Elaboração do Tutorial	08
1.3	Aplicação da Oficina.....	09
1.3.1	Primeiro Encontro.....	10
1.3.2	Segundo Encontro	14
1.3.3	Terceiro Encontro	15
1.3.4	Quarto Encontro	17
1.3.5	Quinto Encontro	18
1.3.6	Sexto Encontro	18
	TUTORIAL PARA A OFICINA DE ROBÓTICA EDUCACIONAL	21
	REFERÊNCIAS	60

1.1 Planejamento da Prática Educativa da Oficina

O processo educacional inclui três fases inseparáveis da prática docente: o planejamento, a aplicação e a avaliação. Deste modo, com base em Zabala (1998) utilizamos as seguintes variáveis para o planejamento da prática educativa da oficina de oficina de robótica educacional: as sequências de atividades de ensino aprendizagem, os conteúdos de aprendizagem, o papel dos professores e alunos, a organização social da aula, a organização dos conteúdos de aprendizagem, os materiais curriculares e a avaliação. Detalhes sobre nosso referencial teórico para definição das variáveis metodológicas podem ser encontrados no texto da dissertação.

1.1.1 Sequências de Atividades de Ensino Aprendizagem

Elaboramos etapas sequenciais para o desenvolvimento e aplicação da oficina de robótica. A oficina pode ser desenvolvida numa carga horária de 20 horas. As atividades descritas a seguir correspondem à carga horária total da oficina, sendo sugerido o tempo de execução das mesmas.

1. Inicialmente deve-se apresentar aos participantes o conceito de robótica. Para isso, pode ser levantado um questionamento referente à interpretação pessoal sobre o que é um robô, a fim de que se possa comparar os diferentes pontos de vistas dos participantes e apresentar uma explicação científica para a questão levantada. (Tempo sugerido: 1 h).
2. A partir das discussões da turma e de suas contribuições, deve ser explicado o objetivo da oficina, a fim de que os participantes saibam o que vão fazer e porque irão fazer, de modo que compreendam o processo a ser seguido. (Tempo sugerido: 1 h).
3. Apresentar os conceitos e princípios de funcionamento do robô de modo progressivo e relacionando aos conhecimentos prévios dos participantes. Ex.: Um dos componentes constituintes do robô é o Arduino, dispositivo que tem a função de tomar decisões e controlar as saídas (atuadores) de acordo com as entradas (sensores). Esse conceito pode ser muito simplificadamente relacionado ao corpo humano: quando estamos em uma rua e desejamos atravessá-la, primeiro olhamos para ambos os lados e caso não haja nenhum veículo seguimos em frente, atravessamos. Observe que os nossos olhos (sensor) detectaram o ambiente, em seguida enviaram a informação para o cérebro (Arduino),

- que por sua vez fez uma comparação: se estiver passando veículo aguarde, se a rua estiver livre movimente as suas pernas (atuadores) e atravesse. (Tempo sugerido: 4 h).
4. A cada apresentação de um componente constituinte do robô, deve ser promovido um diálogo com os alunos, o que poderá trazer o levantamento de dúvidas, questões e contribuições sobre os novos conceitos apresentados.
 5. Introduzir o conceito de lógica para a apresentação da estrutura da linguagem de programação e os comandos básicos. Nesta etapa deve ser proposto um problema cotidiano, para que os alunos possam em dupla, dirigidos e ajudados pelo tutor, expor suas respostas intuitivas ou suposições para a situação proposta. Ex.: Solicitar aos alunos para escrever uma sequência lógica (um passo a passo) de ações necessárias para ir à escola. (Tempo sugerido: 4 h).
 6. Depois das propostas de soluções dos alunos deve ser promovido um diálogo, a fim de possibilitar o levantamento de dúvidas, questões e contribuições sobre o novo conhecimento aprendido.
 7. A partir do tutorial elaborado e disponibilizado para os alunos, realizar, em dupla, a montagem dos robôs. (Tempo sugerido: 2 h).
 8. Com o robô montado, iniciar à sua programação. Cada etapa da programação deve ocorrer de forma progressiva: detecção da arena, detecção do oponente e movimentos do robô. Nesta etapa os alunos devem ser dirigidos e ajudados pelo tutor. Os alunos devem ser incentivados a realizar diferentes programações que lhe permitam praticar o novo conhecimento. (Tempo sugerido: 4 h).
 9. Ao fim do tutorial deve ser proposta uma lista de desafios progressivos, sendo que cada desafio alcançado direciona o robô para o cumprimento das regras que regem os campeonatos de robótica. (Tempo sugerido: 4 h).

1.1.2 Conteúdos de Aprendizagem

Os conteúdos de aprendizagem que compõem a oficina devem ser trabalhados de acordo com a tipologia: conceituais, procedimentais e atitudinais. É importante destacar que os diferentes conteúdos não devem ser abordados simultaneamente em cada etapa, mas sim distribuídos ao longo de toda a sequência de atividades, de modo que haja um equilíbrio entre os conteúdos conceituais, procedimentais e atitudinais. Assim tem-se:

- Conteúdos conceituais: este conteúdo responde a pergunta “o que se deve saber?”, presente nas etapas 1, 2, 3 e 5 da sequência de atividades apresentados na subseção

anterior, onde são abordados conceitos e princípios sobre motor, sensor, controlador, lógica e linguagem de programação.

- Conteúdos procedimentais: este conteúdo responde à pergunta “o que se deve saber fazer?”, presente nas etapas 7 e 8 da sequência apresentados na subseção anterior, onde é realizada a montagem e programação do robô.
- Conteúdos atitudinais: este conteúdo responde à pergunta “como se deve ser?”, presente nas etapas 1, 6, 7, 8 e 9 apresentados na subseção anterior, onde ocorre a exposição das ideias e opiniões dos alunos participantes, bem como a realização das tarefas em duplas, de modo que permite refletir sobre o respeito aos outros, a participação nas tarefas, a colaboração e a cooperação.

1.1.3 Relação Professor-Aluno

Durante a realização da oficina o papel que o professor deve desempenhar para favorecer as relações interativas é: a) contar com a contribuição e conhecimentos dos alunos durante as atividades, a fim de criar vínculos entre os conhecimentos prévios e os novos conteúdos, conforme descrito nas etapas 1, 3, 4 e 6 da sequência de atividades, a fim de favorecer a aprendizagem significativa; b) ajudar os alunos a encontrar sentido no que fazem, levando os alunos a conhecer antecipadamente as atividades que serão trabalhadas e porque serão trabalhadas, conforme detalhado na etapa 2 da sequência de atividades; c) estabelecer metas ao alcance dos alunos, estimulando desafios que os levem a compreender e fazer uso dos novos conhecimentos, conforme descrito nas etapas 5 e 9 da sequência de atividades; e d) oferecer ajuda adequada ao aluno para auxiliar na construção e no progresso do conhecimento.

É importante destacar que nesta relação interativa entre professor-aluno se faz necessário que o aluno assuma uma postura ativa, estando aberto às atividades que propõem o diálogo, a observação, a experimentação e a manipulação, a fim de atribuir significado e enriquecer o novo conhecimento.

1.1.4 Organização Social da Aula

Durante a realização da oficina deve haver momentos em que a sala será organizada em grande grupo e em outros momentos organizada em equipes fixas de dois alunos. A organização em grande grupo pode ocorrer nos momentos referentes às etapas 1, 3 e 4 da sequência de atividade, apresentados nas páginas 37 e 38, permitindo ao professor e aos alunos se dirigirem

ao grupo em geral, através de exposições e diálogos. A organização em grupo permite as relações pessoais e a integração dos participantes, estabelecendo relações de amizade, colaboração e aceitação das diferenças.

Entretanto, esta opção organizativa apresenta limitações para o ensino de conteúdos conceituais e procedimentais, que demandam um acompanhamento do processo de aprendizagem de cada aluno.

Desse modo, se torna necessário adotar medidas que possibilitem conhecer o grau de compreensão de cada aluno diante do novo conhecimento. Para isso, os alunos participantes da oficina devem ser organizados em equipes fixas de dois alunos, que podem ser agrupados segundo suas convivências, de modo que eles tenham liberdade para escolher sua dupla. Este tipo de organização irá favorecer a proposição de atividades que representem um desafio pessoal para cada aluno, como ocorre nas etapas de montagem e programação dos robôs, sendo possível prestar a ajuda necessária durante o processo de aprendizagem.

1.1.5 Organização dos Conteúdos de Aprendizagem

Os conteúdos trabalhados na oficina estão organizados de forma interdisciplinar, a fim de favorecer a interação entre diferentes disciplinas. Podemos citar como exemplo de interação entre disciplinas o conceito do componente Arduino, que possui em sua estrutura pinos de entradas e saídas que permitem a interação com os conceitos presentes nas disciplinas de eletrônica analógica e eletrônica digital. Também podemos citar como exemplo de interação entre disciplinas o funcionamento do sensor ultrassônico, que utiliza fenômenos elétricos e acústicos (física), mas, ao mesmo tempo, se inspira em animais que utilizam ecolocalização (biologia), como morcegos, por exemplo.

1.1.6 Materiais Curriculares

Para a realização da oficina devem ser utilizados materiais curriculares e recursos didáticos que servirão de suporte no processo de ensino-aprendizagem. Neste sentido, pode ser disponibilizado para os alunos o tutorial sobre a concepção, montagem e programação dos robôs, que encontra-se no final deste documento.

O professor por sua vez pode contar com slides, vídeo e recursos midiáticos, que servem de suporte para esclarecer e ilustrar os conteúdos da aula. Os slides devem ser desenvolvidos a partir da sequência de atividades, sendo acrescentado tópicos relevantes para facilitar a

compreensão e estabelecer relações sobre os novos conhecimentos, por exemplo, no tutorial desenvolvido para o aluno, não consta questões sobre lógica de programação, operadores lógicos, operadores de comparação, etc., sendo que estes tópicos devem estar presente nos slides e serem utilizados pelo professor para ajudar na construção de conceitos. Quanto aos recursos midiáticos, podem ser utilizados as plataformas "Padlet" e "Kahoot". O "Padlet" permite o compartilhamento de ideias e opiniões de determinado grupo, possibilitando que o professor suscite questões e perceba a partir das respostas o ritmo da aula e a profundidade dos conteúdos previstos. O "Kahoot" pode ser utilizado para o desenvolvimento de um *quiz* sobre os novos conhecimentos trabalhados durante a aula.

1.1.7 A avaliação

A primeira fase a avaliação deve ocorrer a partir das discussões e opiniões sobre as questões levantadas, bem como a partir do diálogo com os alunos, que terão a oportunidade de falar se já participaram de algum evento de robótica, como foi a experiência e qual a motivação para participar da oficina.

O instrumento a ser utilizado para avaliação dos conteúdos conceituais e procedimentais pode ser a aplicação de um quiz, a observação do uso dos conceitos discutidos em aula para as diferentes situações propostas e a construção do robô. Para avaliação dos conteúdos atitudinais deve ser realizada a observação do comportamento de cada aluno no decorrer de todo o processo.

Avaliação final por sua vez, consiste em conhecer e compreender como ocorreu o processo educativo, desse modo, o aluno não é o único centro da avaliação, mas também a equipe que irá intervir no processo educativo. O instrumento a ser utilizado nesta fase para validar a prática educativa pode ser um questionário semiestruturado.

1.2 Elaboração do Tutorial

O tutorial desenvolvido para ser disponibilizado para os alunos, tem como finalidade orientar, guiar e ilustrar sobre os conceitos e procedimentos necessários para a montagem e programação do robô. A organização do conteúdo se dá de forma progressiva, buscando acompanhar a sequência de atividades conforme descrita abaixo:

1. Breve apresentação para justificar o objetivo do tutorial;
2. Questionamento e definição sobre "o que é um robô?";
3. Apresentação das diferentes categorias nos campeonatos de robótica;

4. Componentes constituintes do robô;
5. Breve descrição de cada componente;
6. Detalhamento dos componentes chaves: Arduino , sensores e ponte H;
7. Passo a Passo sobre a montagem do robô, justificando o uso de cada componente;
8. Breve definição sobre programação;
9. Apresentação do ambiente de desenvolvimento integrado;
10. Comando básicos;
11. Programar para detectar a arena;
12. Hora de explorar (testes);
13. Programar para detectar o oponente;
14. Hora de explorar (testes);
15. Programar para movimentar o robô;
16. Hora de explorar (testes);
17. Lista de Desafios.

O tutorial desenvolvido para o aluno não pretende substituir a mediação do professor, embora nas primeiras fases da sequência haja um trabalho de construção de conceitos, este só se consolidará através da mediação ativa do professor, que não deve ficar restrito apenas ao material fornecido, pois este assumiria um papel apenas de exposição e transmissão do conteúdo.

Assim, o objetivo do tutorial é o de ser utilizado como suporte que contribua no processo de aprendizagem, para realizar as atividades de aplicação e exercitação, não descartando o papel do professor para conduzir a construção conjunta do conhecimento a partir de diálogos, opiniões, ideias ou dúvidas. Após a compreensão dos conceitos e realização da montagem do robô, é preciso que o aluno aprenda a fazer o uso destes conceitos em diferentes situações, o que torna necessário a proposição de problemas ou exercícios sequenciados e progressivos, sendo estes promovidos nas etapas finais a partir da lista de desafios.

1.3 Aplicação Da Oficina

A oficina de robótica teve como tema “Montagem e Programação de Robôs”, sendo realizada no laboratório de Sistemas Digitais Programáveis do IFAL-Campus Maceió. A divulgação da oficina ocorreu por meio de cartazes, conforme ilustrado na Figura 10, que foram fixados nos murais dos cursos técnicos em eletrônica, mecânica, eletrotécnica e informática.

Figura 10 – Cartaz para divulgação da oficina



Fonte: Autor, 2019

A oficina foi realizada em seis encontros, distribuída no período de três semanas, totalizando uma carga horária de 20 horas. Sua aplicação, bem como as atividades desenvolvidas, foram norteadas pelo planejamento descrito no subtópico 3.1 – Planejamento da Prática Educativa. O relato a seguir descreve brevemente os acontecimentos durante a realização da oficina.

1.3.1 Primeiro Encontro

Com a acomodação dos alunos no laboratório, teve início a fase de apresentação, tanto do tutor (pesquisadora), quanto dos participantes. Neste momento, os alunos tiveram a oportunidade de apresentar seus nomes, curso pertencente e comentar se já tiveram experiências em torneios de robótica. Dentre as exposições, alguns alunos afirmaram já ter participado de

torneios de robótica externos ao IFAL utilizando a plataforma LEGO ou Modelix para a construção e controle do robô.

Após isso teve início a apresentação da proposta da oficina, bem como o roteiro de atividades a serem trabalhadas em cada encontro, conforme ilustra a Figura 11.

Figura 11 – Eslaide de apresentação da oficina



Fonte: Autor, 2019

Em seguida, os alunos utilizaram a plataforma “*Padlet*”, cujo acesso foi disponibilizado por meio do computador, para expor suas opiniões sobre “o que é um robô?”. A Figura 12 ilustra as respostas dos alunos.

Figura 12 – Opinião dos alunos sobre o questionamento “Mas afinal, o que é um robô?”

The screenshot shows a Padlet board titled "Mas afinal, o que é um robô?". The board has a dark theme with a grid background. It displays four main posts from students:

- Aurélio e Urandy**: "é um dispositivo eletromecânico, programado para executar funções diversas." (A response from Aurélio and Urandy stating that a robot is an electro-mechanical device programmed to perform various functions.)
- clevir e jakson**: "é uma maquina programada semi autônoma que visa e suprime em alguns casos o trabalho humano." (A response from clevir and jakson stating that a robot is a machine programmed semi-autonomously, aiming to replace or supplement human work in some cases.)
- Hentony e Rafael**: "Um robô, é uma máquina programada pra realizar determinadas ações, onde essas ações são especificadas pelo seu" (A response from Hentony and Rafael stating that a robot is a machine programmed to perform certain actions, where these actions are specified by its programming.)
- Diogo e Alan**: "É uma maquina que só executa uma função que ela foi mandada e nada além disso." (A response from Diogo and Alan stating that a robot is a machine that only executes a function it was programmed to do, nothing more.)

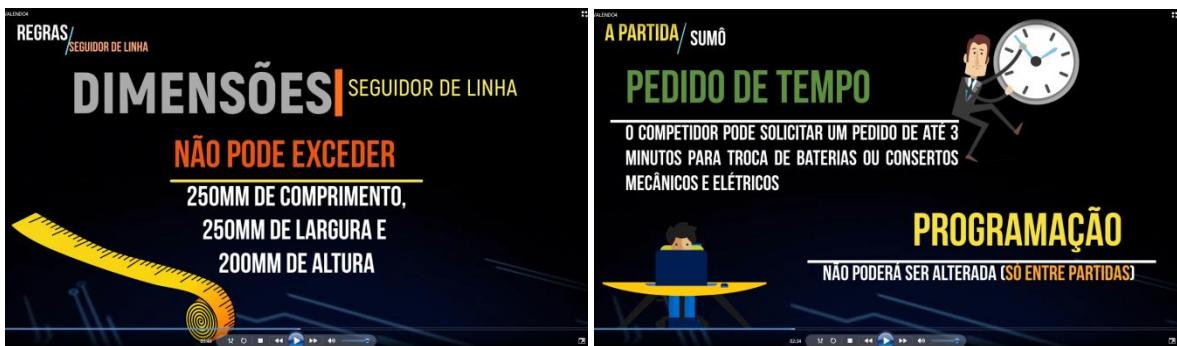
Each post includes a "Adicionar comentário" (Add comment) button. There are also several anonymous comments and a large pink "+" button in the bottom right corner.

Fonte: Autor, 2019

As respostas ficaram disponíveis para visualização e discussão de toda a turma, a fim de compararmos as respostas com a definição científica apresentada. A partir da definição e compreensão do conceito de robôs, bem como seus campos de aplicações, foram mostrados para os alunos por meio de vídeo, alguns torneios de robótica já realizados no âmbito do IFAL, assim como as regras e especificações das diferentes categorias de robôs. O vídeo foi elaborado pelas alunas Evelyn R. A. F. do Nascimento, Jaíne M. S. Santos e Júlia R. S. de Albuquerque

da disciplina de prática profissional do curso técnico em eletrônica, sendo uma contribuição espontânea e bem vinda, que agradecemos e reconhecemos que possibilitou elucidar as regras de forma ilustrativa. A Figura 13 ilustra partes do vídeo exibido para os participantes.

Figura 13 – *Print screen* de partes do vídeo com as regras dos torneios de robótica



Fonte: Autor, 2019

Em seguida, foi exibido para os alunos a estrutura e os componentes que constituem os robôs. Foram apresentados os conceitos e funcionamento de cada componente: chassi, bateria, sensores, Arduino, motores, etc. Os alunos acompanharam a aula por meio do tutorial disponibilizado em formato PDF, conforme mostra o Apêndice A na página 77.

Buscamos abordar cada componente de maneira próxima ao dia-a-dia do aluno, como no caso dos sensores e do Arduino, que foram muito simplificadamente relacionados ao corpo humano, ambos os exemplos descritos na etapa 3 da sequência de atividades proposta no subtópico 3.1 deste trabalho. Outro exemplo de abordagem foi no momento de diferenciar o comportamento da eletrônica analógica e digital, de modo que estas foram comparadas a partir da visão humana, sendo a analógica representada pela diversidade de cores que os nossos olhos percebem, e a digital representada por apenas dois níveis de cor, como ocorre ao visualizar uma silhueta.

Ao tratar do componente sensor infravermelho, percebemos a necessidade de realizar um experimento prático que não havia sido previsto no planejamento inicial. Isto foi necessário pois, no kit fornecido aos alunos este sensor já é soldado em uma placa de circuito impresso com os resistores para limitar a corrente, sendo necessário que os próprios alunos montassem o esquema composto pelo sensor e resistores em uma *protoboard*, para que identificassem o fotodiodo e o fototransistor e verificassem o funcionamento deste componente através do multímetro, a fim de possibilitar melhor assimilação do conceito abordado.

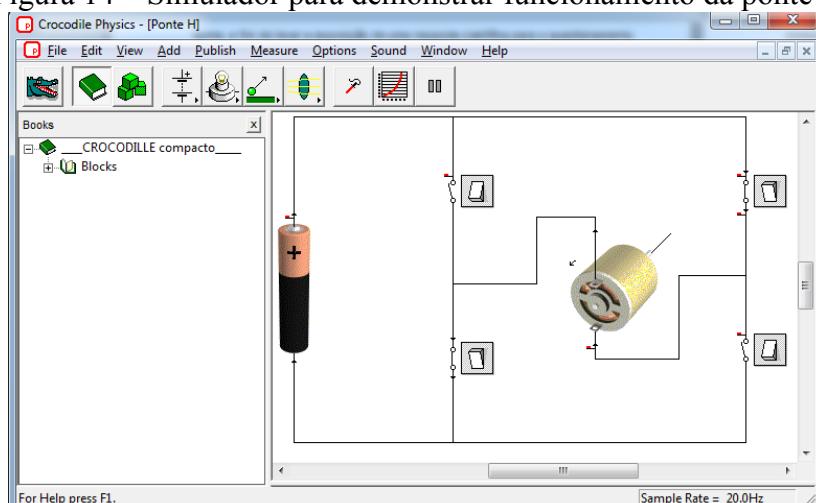
Com o andamento da aula, foi percebido que o tempo para execução do roteiro previsto para o primeiro encontro precisaria ser maior, portanto, foram abordados todos os componentes do robô, com exceção da ponte H, sendo este componente trabalhado no encontro seguinte.

1.3.2 Segundo Encontro

Esta aula teve início com uma breve revisão do que foi visto no primeiro encontro. Em seguida foi trabalhado o conceito e funcionamento da ponte H. Para melhor compreensão deste componente, o tutor utilizou o computador para demonstrar aos alunos através do simulador *Crocodile Physics*, o comportamento de um motor de corrente contínua (DC) ao inverter a polaridade de sua alimentação, conforme ilustrado na Figura 14. É importante destacar que neste momento não foi realizado um experimento prático com a ponte H, pois ainda não havia sido apresentado para os alunos os conceitos básicos de programação.

Também foi demonstrado a partir de uma tabela verdade, tabela que determina a saída em função das entradas, todas as possíveis combinações de movimento dos dois motores, que quando acoplados às rodas, garantem a propulsão do robô, e o controle de direção do mesmo. Na tabela verdade, foram elencadas as condições para obter os movimentos de ir para frente, ir para atrás, girar para a direita e girar para a esquerda.

Figura 14 – Simulador para demonstrar funcionamento da ponte H



Fonte: Print screen do software, 2019

Concluída a apresentação dos componentes do robô, foi abordado o conceito de lógica de programação. Neste momento foi discutido como a lógica faz parte de atividades do dia-a-dia como fazer um bolo, tomar banho, ir à escola, acordar, etc. Para exercitar o pensamento

lógico, foi proposto para os alunos que escrevessem uma sequência lógica das ações necessárias para a ida à escola. A partir das respostas mencionadas pelos alunos, foi esclarecido que a principal diferença entre um algoritmo do dia-a-dia e um algoritmo de sistemas é a linguagem utilizada, onde foi citado a necessidade de uma linguagem de programação.

Em seguida foi apresentado o ambiente de desenvolvimento integrado do Arduino e comentado sobre as funcionalidades dos seguintes itens: barra de ferramentas, menu, ambiente de trabalho, monitor serial e console de feedback. A aula prosseguiu com a explicação sobre a estrutura padrão do programa (*setup* e *loop*), os operadores de comparação, os comandos básicos *pinMode()*, *digitalWrite()*, *digitalRead()*, *delay()* e a estrutura condicional *if()*, conforme descrito no Apêndice A na página 77.

As respostas dos alunos sobre o algoritmo para a ida à escola foi relembrado a fim de introduzir os conceitos sobre bibliotecas e variáveis. Foi citado o exemplo da resposta de uma dupla cujas instruções eram bem gerais: acordar, tomar banho, tomar café, sair de casa, pegar o ônibus, etc., e citado também a resposta de outra dupla que descreveu minuciosamente cada etapa: abrir os olhos, levantar, pegar as roupas, abrir a porta do banheiro, abrir o box, fechar o box, abrir o chuveiro e etc. Através da comparação destas respostas, foi discutido que se o banho for considerado uma ação padrão, é possível deixar esse código já escrito e chamar a função “banho” sempre que necessário. O conjunto de instruções deste código que agora pode ser compartilhado é chamado de biblioteca. Para abordar sobre o conceito e tipos de variáveis, foi explicado que a duração do banho pode variar de uma para outra pessoa, sendo preciso informar esse tempo através de uma variável.

A aula continuou com a realização da programação para testar os sensores infravermelhos e ultrassônico, e a visualização de seus respectivos comportamentos através do monitor serial.

Para a conclusão da aula foi proposto para os alunos participarem do *quiz* na plataforma *Kahoot*. O *quiz* continha questões sobre os componentes constituinte dos robôs, assuntos estes trabalhados no primeiro e segundo encontro.

1.3.3 Terceiro Encontro

Nesta aula os alunos realizaram a montagem dos robôs seguindo o passo a passo elencado no tutorial disponibilizado, conforme ilustra a Figura 15. Também foi fornecido para os alunos um formulário para preenchimento das informações referentes aos robôs, com o intuito de identificar em quais pinos os componentes estavam conectados, a fim de facilitar a

programação dos robôs. A duração desta atividade foi menor do que o previsto, pois os alunos não tiveram dificuldade para montar os robôs, otimizando o tempo e antecipando parte das atividades prevista para a próxima aula.

Figura 15 – Alunos realizando a montagem dos robôs



Fonte: Autor, 2019

Em seguida as duplas iniciaram a programação dos robôs. Como na aula anterior os alunos já tinham realizado a programação para testar os sensores infravermelhos, não houve dificuldade nesta etapa. Deste modo, conforme orientado na sequência do tutorial, os alunos utilizaram uma pequena placa de PVC com uma face de cor branca e uma face de cor preta, com o objetivo de verificar o retorno do sinal do sensor diferenciando a superfície clara da superfície escura. Em seguida, os valores referentes a cada uma das superfícies foram anotados, para posteriormente completar a programação cujo objetivo é detectar a arena. Ainda nesta aula,

os alunos iniciaram a programação para detectar o oponente através do sensor ultrassônico. Neste momento além da programação, as duplas dialogaram e decidiram sobre a distância máxima necessária para “enxergar” o oponente sem que houvesse interferência dos objetos ou pessoas que estivessem próximas. A aula foi finalizada com um *quiz* sobre os conceitos de programação trabalhados no encontro anterior, conforme ilustrado na Figura 16.

Figura 16 – Demonstração do quiz sobre programação

Pelo monitor serial podemos enviar e receber informações na forma de texto. Sobre isso sabemos:

58

0 Answers

kahoot.it Game PIN: 32309

▲ Que a função Serial.begin() permite imprimir uma informação	◆ A função Serial.begin() permite interromper a comunicação
● A função serial.Begin() permite inicializar uma comunicação	■ A função Serial.println() permite imprimir uma informação

Fonte: Autor, 2019

1.3.4 Quarto Encontro

Otimizar o tempo da aula anterior foi de fundamental importância para a conclusão das atividades planejadas, pois neste encontro a duração das atividades foi maior do que o esperado, visto que os alunos apresentaram dúvidas para realizar a programação que permite a movimentação dos robôs. Nesta etapa foi revisado o funcionamento da ponte H e houve acompanhamento particular para cada dupla durante a realização desta atividade. A intervenção do tutor nesta fase também foi necessária, pois parte das duplas não preencheram o formulário entregue para eles durante a etapa de montagem do robô, acarretando em dificuldades para identificação dos pinos do Arduino correspondentes a cada motor. O preenchimento do formulário teve por objetivo evitar o esquecimento por parte dos alunos de quais pinos correspondiam a cada componente, bem como facilitar o tutor a identificar possíveis erros de programação durante a execução das atividades.

Apesar dos imprevistos que ocorreram durante a aula, as duplas conseguiram concluir a programação de movimento dos robôs. Foi orientado aos alunos que escrevessem a

programação com movimento para frente, atrás, direita e esquerda com comentários no próprio código para facilitar as atividades da aula seguinte.

1.3.5 Quinto Encontro

Esta aula teve por objetivo dar prosseguimento à programação dos robôs com foco na categoria de mini sumô. Como nas aulas anteriores os alunos já haviam realizado testes e desenvolvido códigos referentes à identificação da arena, detecção do oponente e movimento dos robôs, nesta etapa, foi proposta para os alunos a resolução da lista de desafios presente no tutorial.

A medida que os alunos buscavam resolver os desafios, estes realizaram diversos testes para melhorar o desempenho dos robôs. Alguns alunos apresentaram dúvidas durante a realização das atividades, porém com o acompanhamento do tutor e de outros alunos que já haviam concluído a programação de seus robôs, todas duplas conseguiram concluir a atividade.

Para concluir as atividades desta aula, decidimos realizar um torneio com a participação de todas as duplas, conforme ilustrado na Figura 17. Esta atividade possibilitou aos alunos perceberem as melhorias que poderiam ser feitas e debaterem sobre as possíveis mudanças de estratégias que poderiam ser adotadas na programação.

Figura 17 – Torneio na categoria mini sumô com os participantes da oficina



Fonte: Autor, 2019

1.3.6 Sexto Encontro

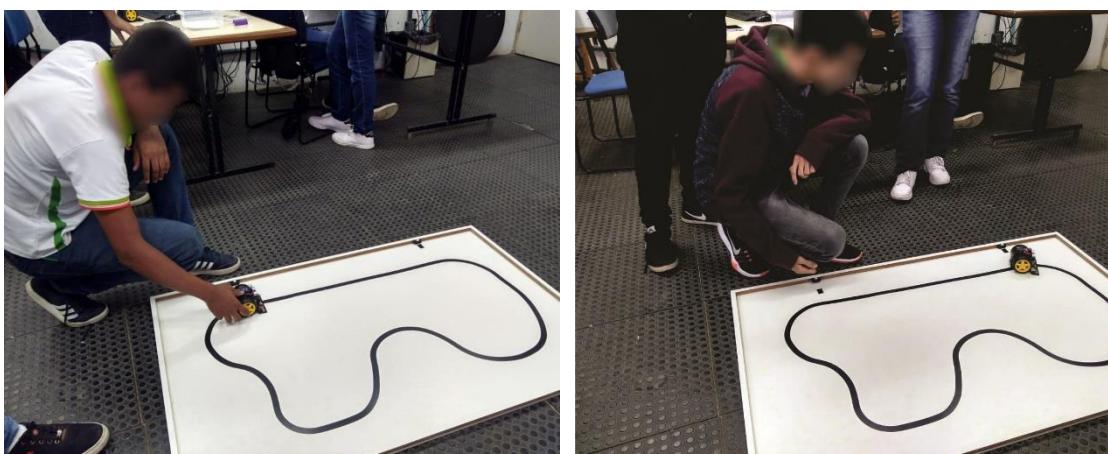
Esta última aula teve por objetivo desenvolver a programação dos robôs com foco na categoria seguidor de linha, a partir da lista de desafios do tutorial. Como os alunos já haviam realizado no encontro anterior a programação dos robôs na categoria de mini sumô, foi

pressuposto que a realização das atividades da aula fluiria mais facilmente, entretanto, algumas dificuldades foram percebidas no decorrer das atividades, como por exemplo, a ausência de uma superfície em cada bancada para simular o caminho a ser percorrido pelo robô, exigia que os alunos levantassem constantemente para realizar os testes de programação, provocando atrasos por ter apenas uma área de teste.

Outra dificuldade identificada durante as atividades foi que robô precisava de um peso maior na parte frontal, onde estavam localizados os sensores infravermelhos, pois ao realizar os movimentos necessários para continuar seguindo a linha da área, a parte frontal levantava, interferindo na leitura correta da superfície pelos sensores, provocando um comportamento não esperado dos robôs. Para resolver esse problema foi fornecido aos alunos pequenos pesos de chumbo para serem fixados na parte frontal do chassi.

Apesar dos imprevistos indicarem a necessidade de melhoria nos aspectos mencionados, conseguimos concluir a programação dos robôs conforme planejado. A Figura 18 ilustra a participação de alguns alunos nesta etapa.

Figura 18 – Alunos testando a programação dos robôs para a categoria seguidor de linha

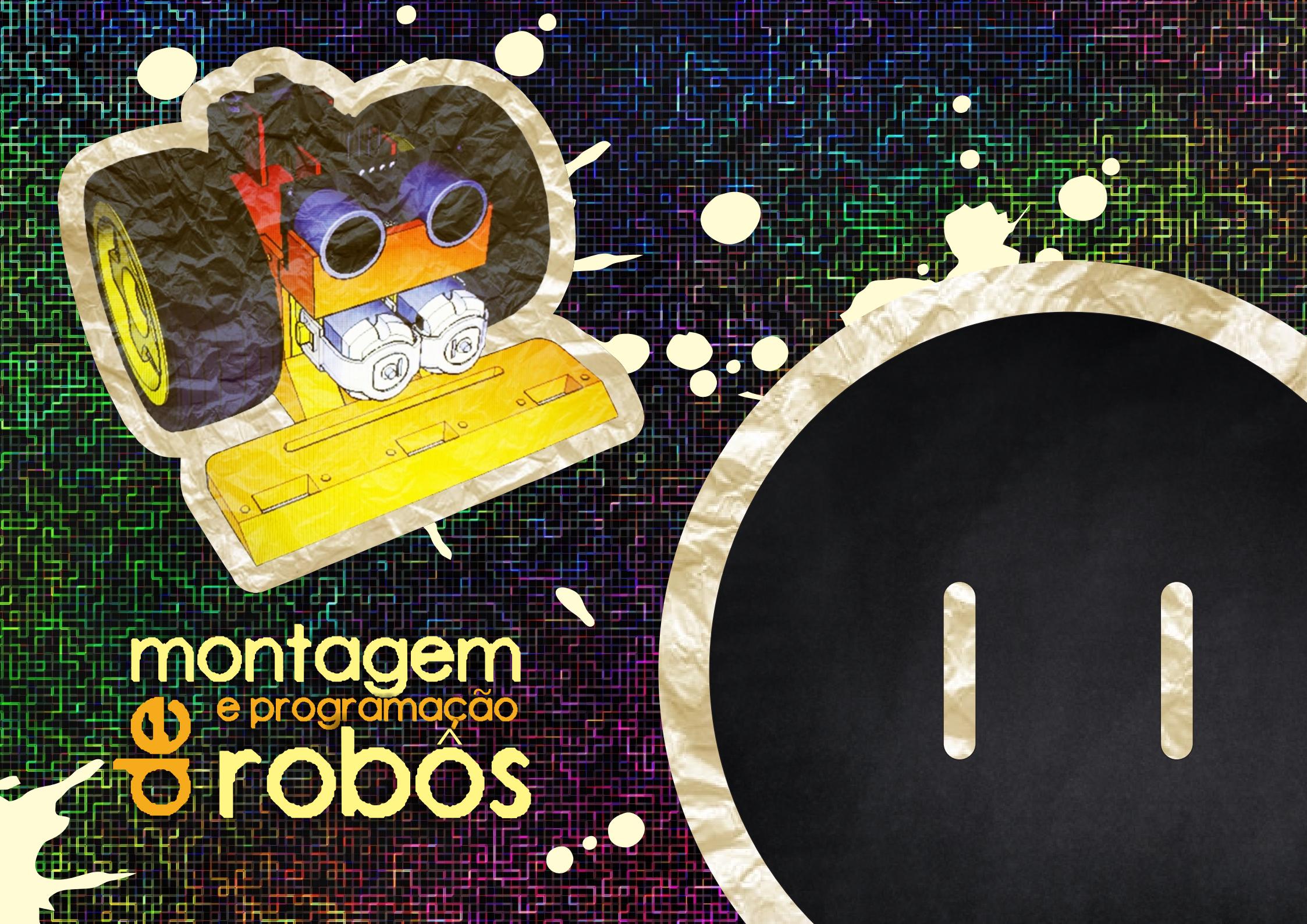


Fonte: Autor, 2019

Após o desenvolvimento das atividades propostas para este dia, foi dialogado com os alunos sobre as possíveis falhas que os robôs podem apresentar durante os torneios e quais as possíveis medidas a serem adotadas para identificar a origem da falha e solucioná-la. Também foi esclarecido para os alunos que o modelo de robô proposto na oficina apresenta limitações para participar de uma competição na categoria mini sumô, mas que uma vez entendido a sua concepção, é possível desenvolver um robô que apresente melhor desempenho durante a competição.



Em seguida os alunos avaliaram o processo de realização da oficina através do questionário semi-estruturado. Para encerrar a aula, a pesquisadora agradeceu aos alunos pela participação e colaboração de todos os presentes.



montagem e programação de robôs

Sumário

Apresentação	3
O que é um robô?	4
Torneios de robótica	5
Componentes do robô	7
Arduino	10
Sensor Ultrassônico	11
Sensor IR	12
Ponte H	13
Atividades Práticas	14
Mãos à obra!	15
Agora, bora programar!.....	26
Conhecendo o IDE do Arduino	27
Comandos Básicos	29
Programando...detectando a arena	31
Programando...detectando o oponente	33
Programando...movimentando o robô	35
Lista de Desafios	39

Apresentação

Este é um tutorial para orientar a montagem e programação de robôs, que irá prepará-lo para participar em campeonatos de robótica.

Ele foi pensado especificamente para as categorias de sumô de robôs e seguidor de linha, realizadas no Instituto Federal de Alagoas (IFAL), entretanto, os fundamentos aqui abordados podem ser utilizados também em outras atividades e torneios.

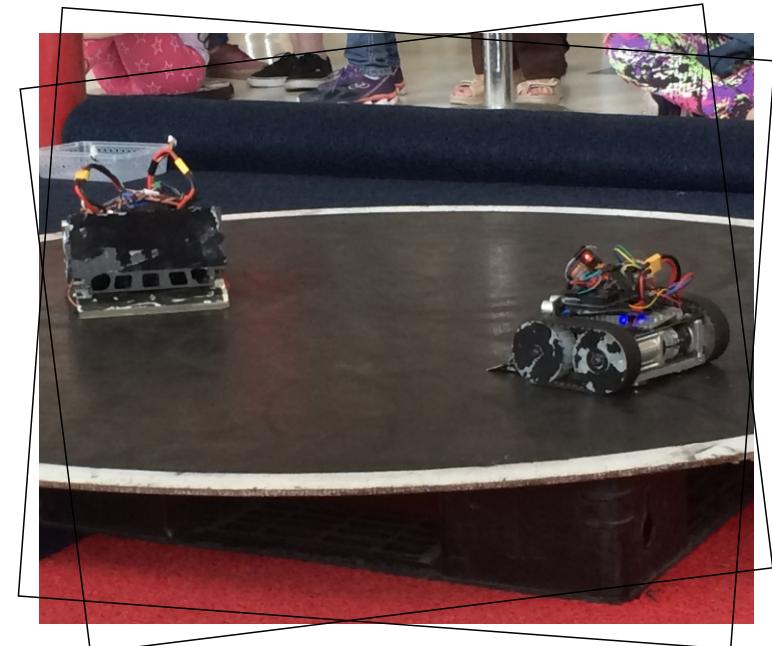
O que é um robô?

Um robô é um dispositivo capaz de realizar tarefas de maneira autônoma ou pré-programada. Em nosso caso, faremos a montagem e programação dos robôs de acordo com as regras dos torneios de robótica do IFAL. Nossa robô será montado a partir de um kit que contém os itens necessários para a sua construção.

Torneios de robótica

Nas competições de robótica existem diferentes categorias que competem entre si:

Sumô: nesta modalidade os robôs competem em uma arena (dojo) e tem por objetivos empurrar o oponente para fora da área de disputa. No caso do mini sumô, os robôs devem ter as dimensões de 10x10 (largura x comprimento) e peso de 500g.

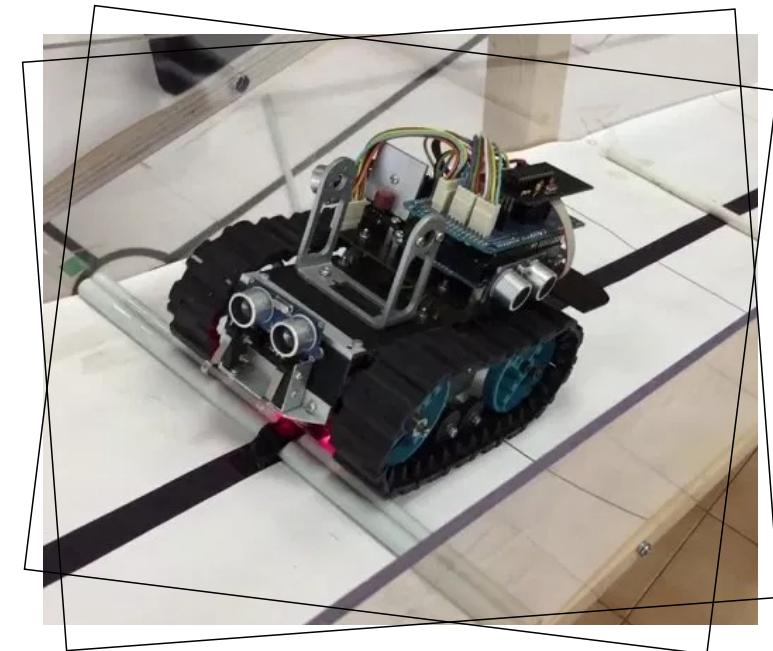


Resgate: nesta categoria é simulado um ambiente de desastre, e o robô precisa seguir um determinado percurso, superar certos obstáculos e realocar as vítimas em um local seguro.

Nesse tutorial montaremos um robô que possibilita a participação na categoria de mini sumô e seguidor de linha..

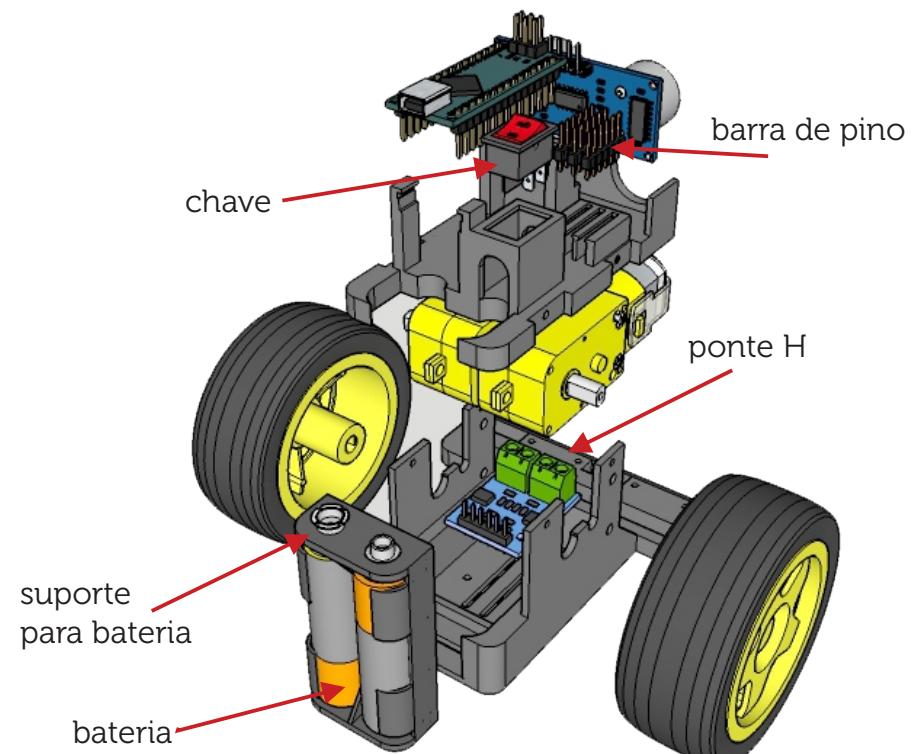
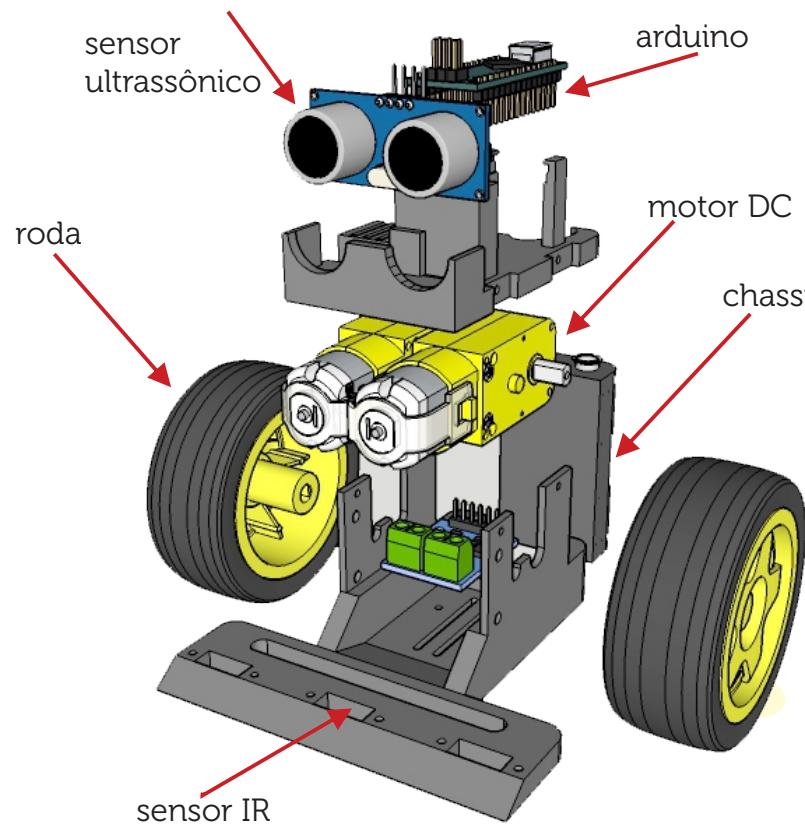


Seguidor de linha: são robôs que por meio de sensores detectam e seguem um caminho em uma superfície. A vitória é dada para o robô que fizer o percurso no menor tempo.



Componentes do robô

Antes de colocarmos a “mão na massa”, vamos entender quais componentes fazem parte do robô e como eles funcionam.



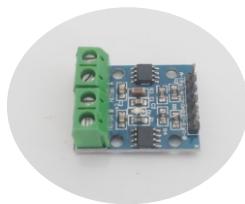
Chassi: é a estrutura do robô que servirá de suporte para os demais componentes.



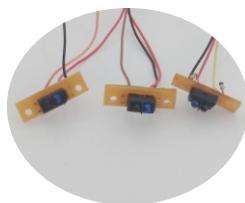
Chave: utilizada para ligar/desligar um determinado circuito elétrico/eletrônico.



Ponte H: este é um componente que permite controlar o sentido de rotação e a velocidade de até dois motores DC. Sua tensão de alimentação varia de 2,5V a 12V. Na página 11, veremos seu princípio de funcionamento.



Sensor IR: este sensor permite identificar obstáculos a sua frente, ou detectar se uma superfície é clara ou escura. Sua detecção varia de 1mm a 8mm. Na página 10 traremos detalhes de seu funcionamento.



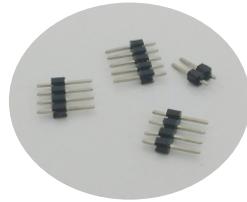
Motor DC: possui dois pólos de alimentação que ao serem alimentados em polarização contrária, permite o giro do motor no sentido contrário. Utilizaremos um motor com operação de 3 a 6v.



Roda: a roda quando acoplada ao eixo do motor permite a locomoção do robô. Em nosso kit temos uma roda de plástico com 68mm de diâmetro.



Barra de pinos: componente que permite a conexão para outros dispositivos, podendo ser facilmente destacada, dando mais flexibilidade e autonomia na sua utilização.



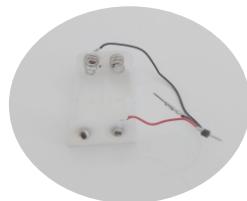
Arduino: é uma plataforma de prototipagem, de hardware e software livre, composta por um microcontrolador, circuitos de entradas/saídas, que pode ser conectado à um computador e programada utilizando linguagem própria. Detalhes na página 8.



Sensor ultrassônico: dispositivo utilizado para medir distâncias de 2cm a 4m. Entenderemos seu funcionamento na página 9.



Suporte para bateria: o suporte contido neste kit conecta duas baterias de 3,7V em paralelo, facilitando a fixação das baterias.

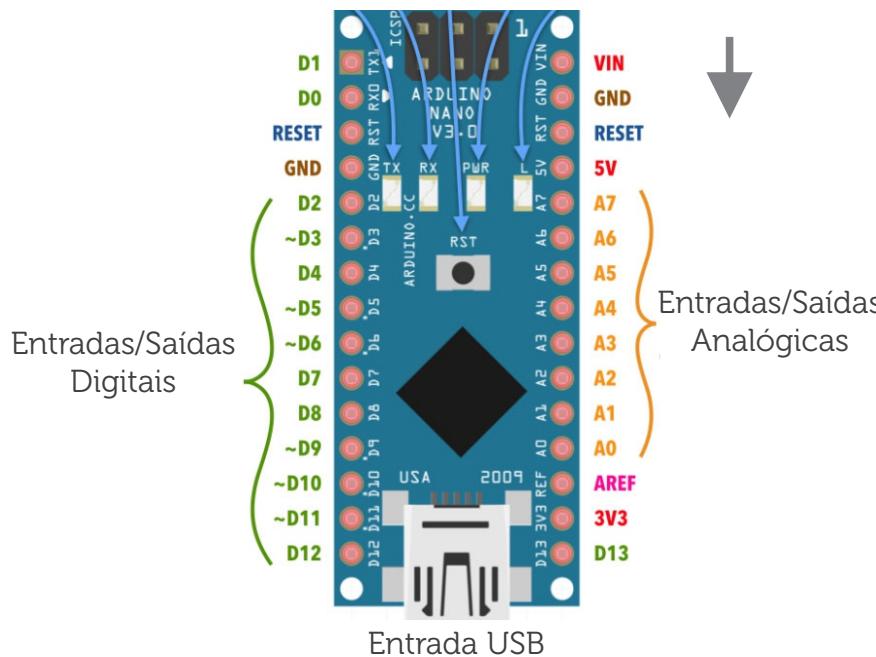


Bateria: este é o dispositivo que permitirá o funcionamento do nosso robô, através do fornecimento de energia para todo o circuito.



Nas próximas páginas trataremos com mais detalhes sobre o funcionamento do seguintes componentes: Arduino, ponte H, sensor IR e sensor ultrassônico.

Arduino



Entradas/Saídas
Digitais

Entradas/Saídas
Analógicas

Entrada USB

Falando em termos práticos, o arduino possui o funcionamento semelhante ao de um pequeno computador, no qual, é possível programar a maneira como suas portas de entradas e saídas de dados devem se comportar em meio aos componentes externos (sensores, motores) que podem ser conectados as mesmas.

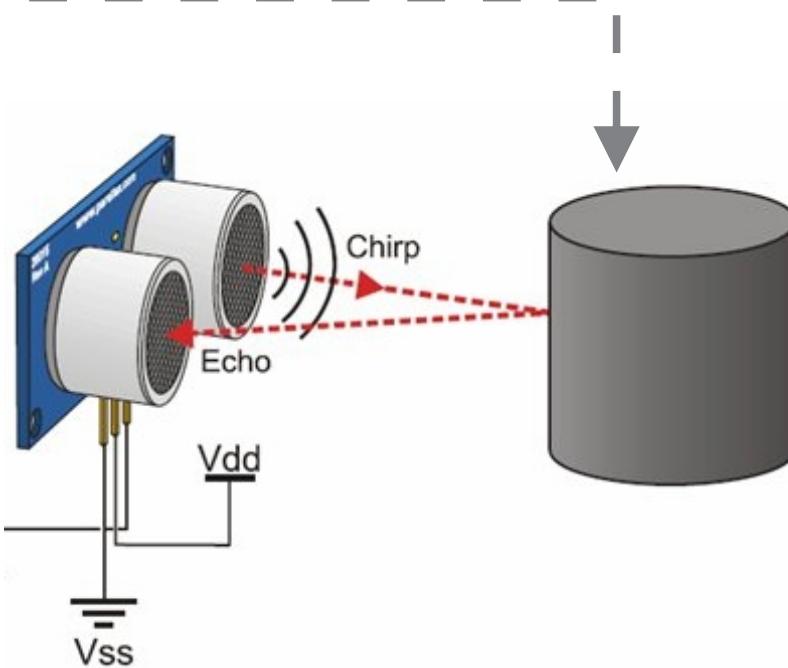
Sendo assim, o arduino interpreta os dados e controla as saídas afim de criar sistemas automáticos. Entretanto, para isso, é necessário programá-lo.

A programação nada mais é do que "falar" ao controlador quais decisões devem ser tomadas em cada circunstâncias.

Utilizamos a sua IDE (ambiente integrado de desenvolvimento), no qual escrevemos um código em uma linguagem baseada em C/C++, que após ser compilado será traduzido em um código comprehensível pela nossa placa.

A partir da página 24 traremos mais detalhes sobre a sua programação.

Sensor Ultrassônico



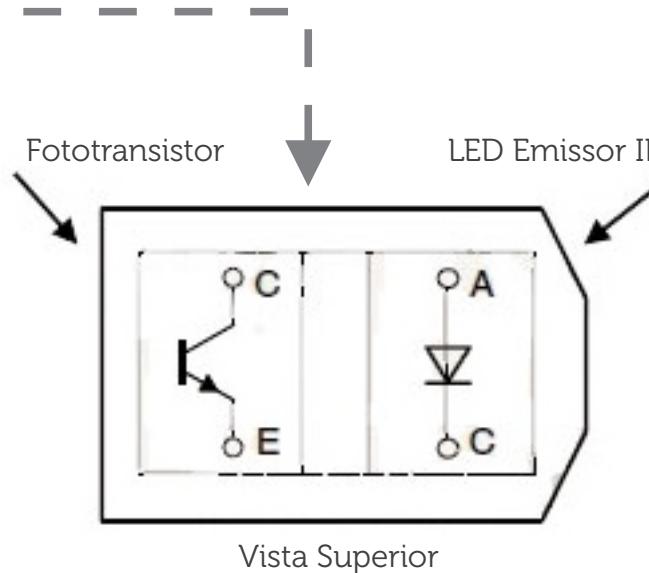
Este sensor permite que você faça leituras de distâncias entre 2 cm e 4 metros, com precisão de 3 mm, sendo utilizado como detector de objetos.

Seu funcionamento consiste basicamente em enviar um sinal que, ao atingir um objeto, volta para o sensor e com base nesse tempo entre o envio e recebimento, é calculada a distância entre o sensor e o objeto. O processo de medição ocorre em três etapas:

- 1) É enviado um sinal com duração de 10 µs (microsegundos) ao pino trigger, indicando que a medição terá início;
- 2) Automaticamente, o módulo envia 8 pulsos de 40 KHz e aguarda o retorno do sinal pelo receptor;
- 3) Caso haja um retorno de sinal (em nível HIGH), determinamos a distância entre o sensor e o obstáculo utilizando a seguinte equação: Distancia = (pulso em nível alto x velocidade do som (340m/s)) / 2.

A divisão por 2 é necessária já que estamos contando o tempo de ida e volta do sinal. O programa para esse sensor utiliza a biblioteca Ultrasonic, como veremos na página 29.

Sensor IR



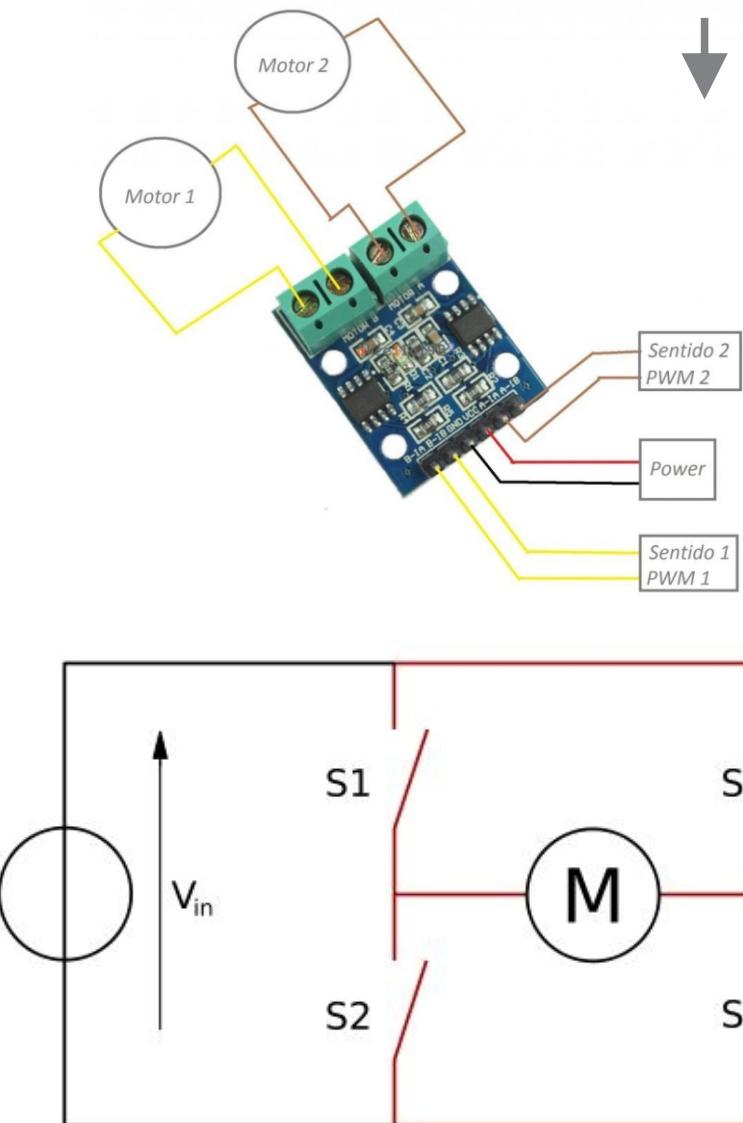
Este sensor contém dois componentes: um LED que emite um feixe de luz infravermelha e um fototransistor, que tem a função de capturar o feixe de luz emitido pelo LED.

Assim, quando o LED emite um feixe de luz, a mesma é refletida pelo objeto que estiver à frente do sensor e que é por sua vez detectada pelo fototransistor.

Vale lembrar que a cor e o material do objeto ou da superfície, interferem na intensidade da luz refletida, permitindo a distinção entre superfícies claras e escuras.

É importante lembrarmos que o LED e o fototransistor precisam de resistores para limitar a passagem de corrente no circuito.

Ponte H



A ponte H é um circuito que permite controlar a direção de rotação de um motor de corrente contínua (DC), através da inversão da direção da corrente que flui entre seus terminais.

Para isso, o circuito precisa de um caminho que carregue a corrente ao motor em uma direção, e outro caminho que leve a corrente no sentido oposto.

É possível compreender melhor observando a imagem acima, que possui quatro chaves, que podem ser controladas de forma independente, assim, dependendo do estado das chaves (aberta ou fechada) e da forma de combinação, é possível controlar a rotação de um motor em um sentido e no sentido inverso.



Entendido esses
componentes,
agora sim...

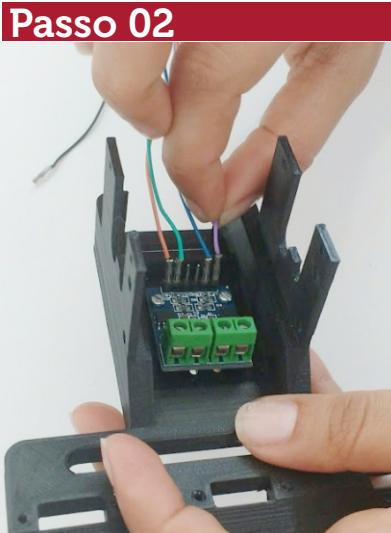
Atividades Práticas

Agora você irá desenvolver suas habilidades práticas montando um robô com os componentes apresentados anteriormente, que estará pronto para receber a programação e realizar determinadas atividades como andar, parar e identificar obstáculos.

Mãos à obra!



Passo 02



Conecte quatro jumpers nas entradas de controle da ponte H. Esses fios serão conectados posteriormente nos pinos digitais do arduino, para o controle do sentido de rotação.

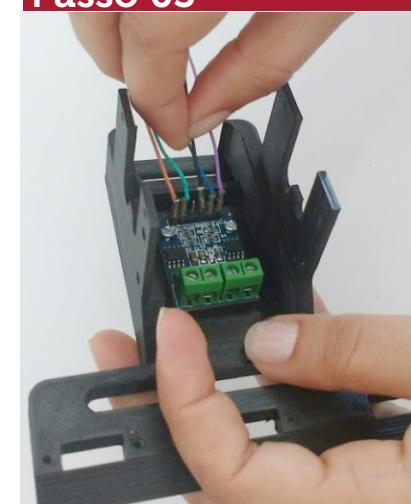


Fixe a ponte H no chassi do robô, através de parafusos. Coloque a ponte H na posição como mostra a foto ao lado, para facilitar a conexão dos fios dos motores.

Passo 01

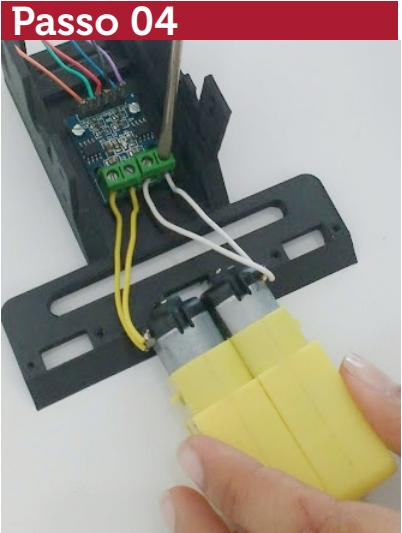


Passo 03



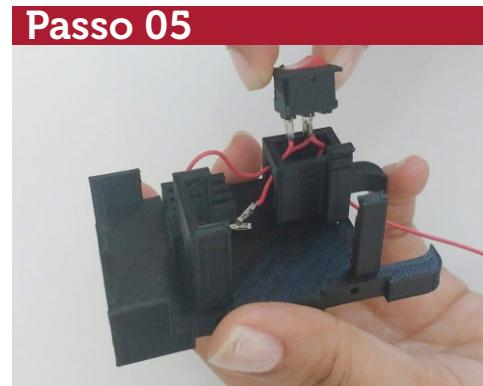
Conecte o jumper preto no GND da ponte H. Esse fio será interligado no barramento negativo como será visto nos passos 32 e 33.

Passo 04



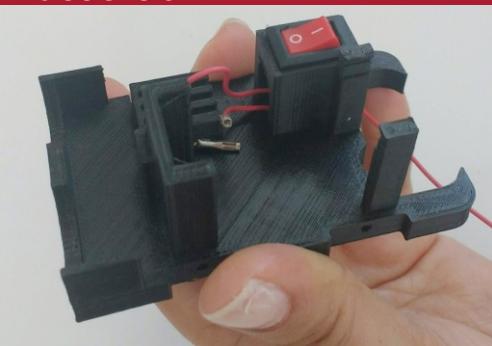
Conekte os fios dos motores aos bornes da ponte H. Observe a ligação para que seja conectado os fios do motor ao seu borne correspondente.

Passo 05



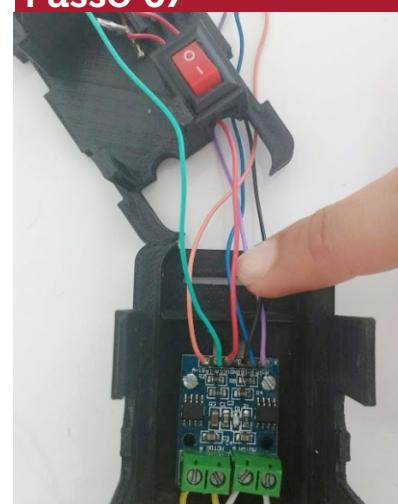
Agora, precisamos encaixar a chave na peça menor que faz parte do chassi. Observe que nos terminais da chave um dos lados tem apenas um fio (que será conectado à bateria) e o outro lado possui dois fios, que será

Passo 06



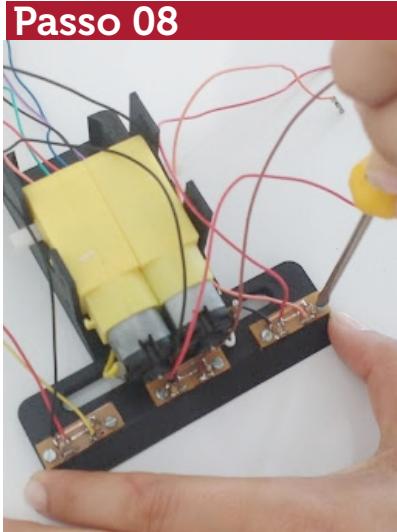
conectado à ponte H e o outro irá para a barra de pinos, para fazer a distribuição do sinal positivo, como vamos ver posteriormente.

Passo 07



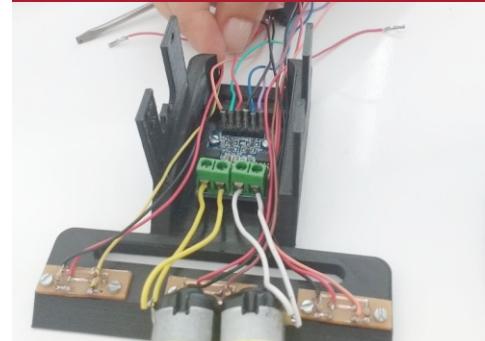
Conekte o fio de maior comprimento proveniente da chave, no pino de alimentação (Vcc) da ponte H. A chave permite ou interrompe o positivo da bateria, permitindo ligar ou desligar a alimentação dos componentes.

Passo 08



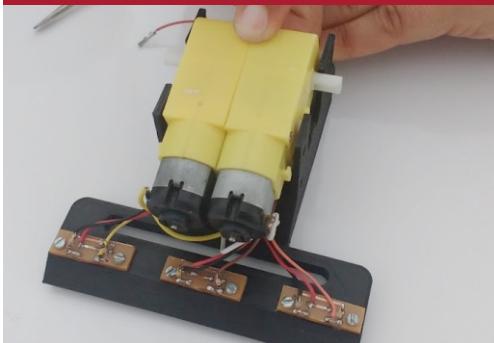
Fixe os três sensores IR na base do chassi através de parafusos, como mostra a imagem ao lado.

Passo 09



Passe todos os fios dos sensores IR (sensor de piso) pelo meio do chassi, ao lado ponte H, conforme mostrado na imagem.

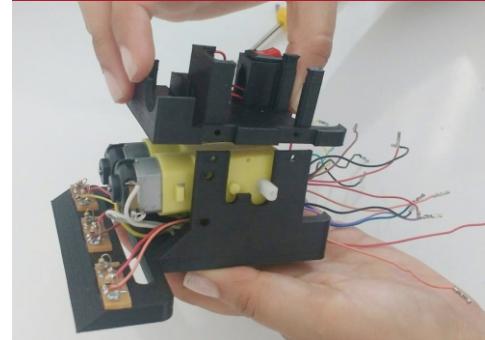
Passo 10



Os fios devem ficar embaixo do motor. Agora, encaixe o motor no chassi, como mostra a imagem.

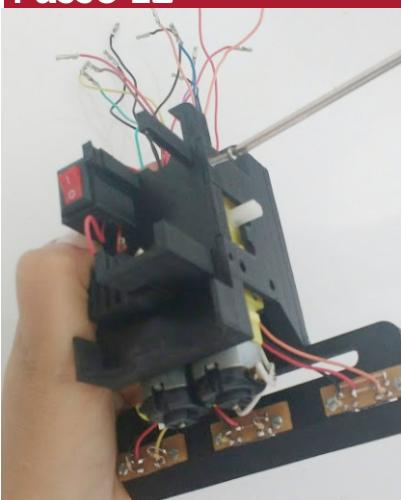


Passo 11



Encaixe a peça menor que faz parte do chassi. Observe a posição de encaixe, pois a parte onde será colocado o sensor ultrassônico deve estar do lado frontal.

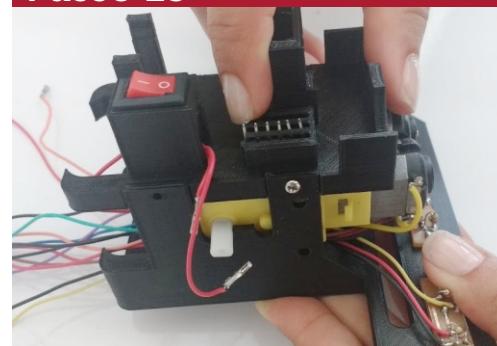
Passo 12



Fixe o chassi a parte menor do chassi com parafusos.



Passo 13



Encaixe as barras de pinos nos “cortes” contidos na parte superior da peça menor do chassi. Os “cortes” devem estar totalmente preenchidos como mostra o passo 14.

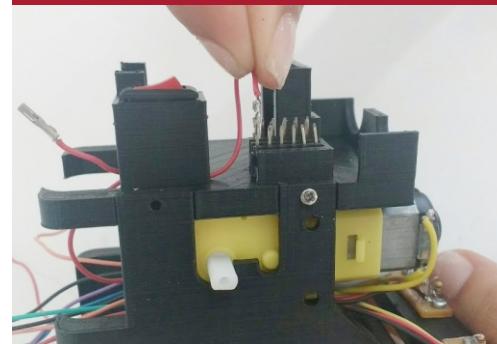
Passo 14



Essas barras de pinos serão utilizadas como barramento, a fim de fazer a distribuição do sinal negativo e positivo, provenientes da bateria e da chave para os demais componentes.



Passo 15



Conekte o terminal proveniente do contato da chave que possui dois fios, à barra de pinos . A partir de agora, todo esse “corte” tem potencial positivo.

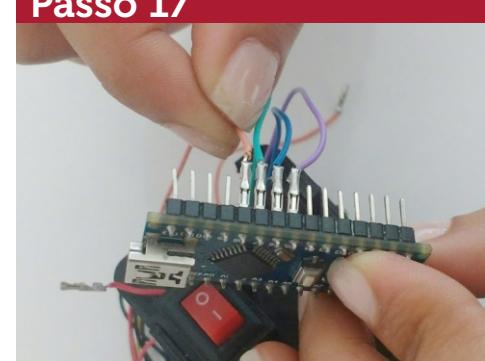
Passo 16



Conecte os quatro terminais provenientes das entradas de controle da ponte H, nas entradas digitais do Arduino.

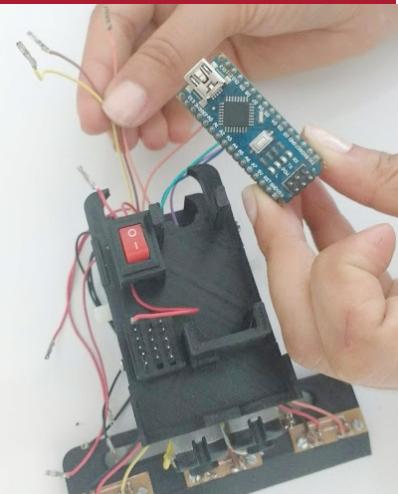


Passo 17



Observe quais pinos digitais estão sendo conectados no Arduino e a qual entrada da ponte H ele corresponde. Precisaremos dessas informações mais adiante, quando faremos sua programação.

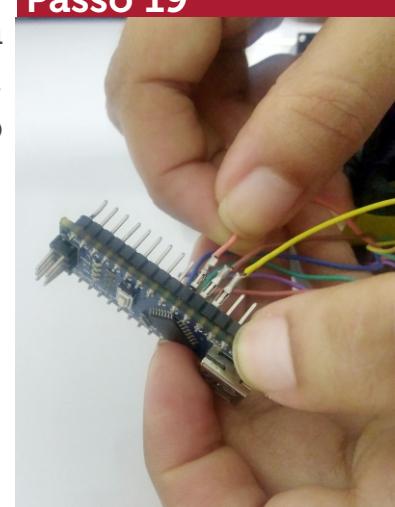
Passo 18



Conecte os terminais de sinal dos sensores IR (os fios são os coloridos, pois o vermelho e o preto correspondem à alimentação dos sensores), às entradas analógicas do Arduino.



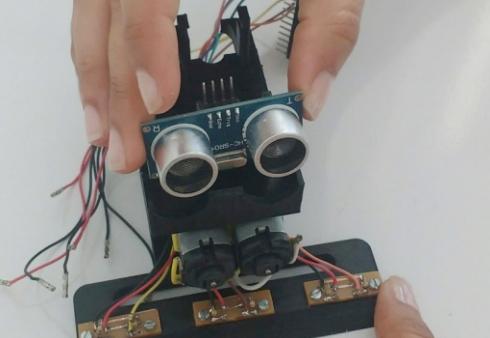
Passo 19



Observe quais pinos analógicos estão sendo conectados no Arduino e a qual sensor IR ele corresponde. Precisaremos dessas informações mais adiante, quando faremos sua programação.



Passo 20



Encaixe o sensor ultrassônico no suporte da parte frontal do chassi.



Passo 21

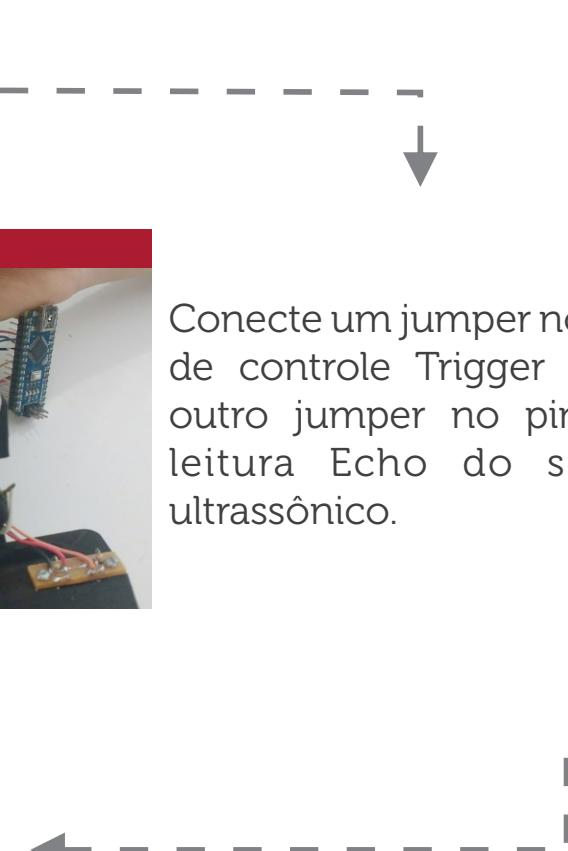


Conekte um jumper no pino de controle Trigger e um outro jumper no pino de leitura Echo do sensor ultrassônico.

Passo 22



Conekte as extremidades desses jumpers nas entradas digitais do arduino, como mostrado na imagem. Observe quais pinos estão sendo utilizados, pois precisaremos dessas informações durante a programação.

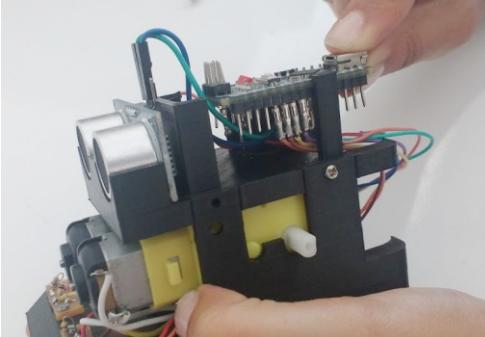


Passo 23



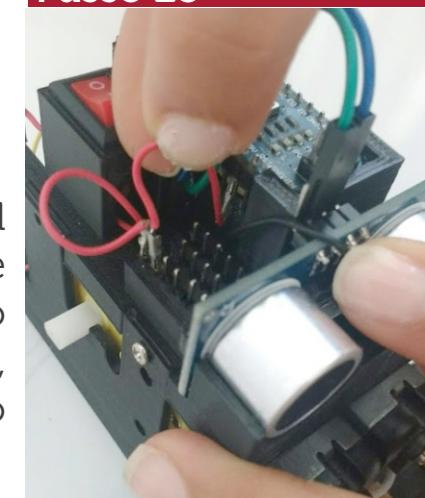
Conekte o jumper vermelho no 5V e o jumper preto no GND do Arduino.

Passo 24



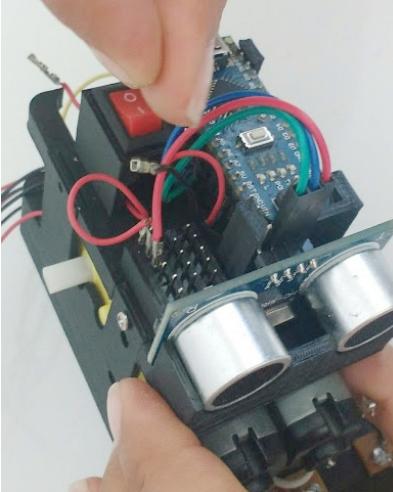
Encaixe o Arduino no suporte estilo “gaveta” do chassi, como mostrado na imagem ao lado.

Passo 25



Conekte o terminal vermelho proveniente do 5V do Arduino, no barramento positivo, conforme explicado no passo 15.

Passo 26



Conekte o jumper vermelho no pino Vcc do sensor ultrassônico, e em seguida a outra extremidade do jumper ao barramento positivo.

Passo 27



Conekte o jumper preto no GND do sensor ultrassônico, e em seguida a outra extremidade do jumper ao barramento negativo. Conekte o terminal preto proveniente do GND Arduino a esse barramento.

Passo 28



Precisaremos de mais um barramento negativo, para isso, utilize um pequeno jumper para fazer a ligação entre os dois "cortes" de barras de pinos.



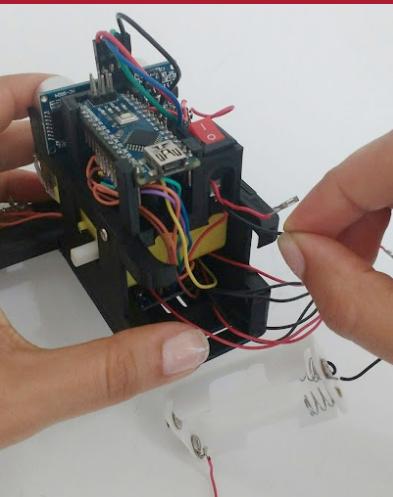
Passo 29



Assim, teremos o primeiro "corte" como barramento positivo, e o segundo e terceiro "cortes" como barramento negativo.



Passo 30



Agora, precisamos conectar o negativo da bateria no barramento correspondente. Para isso, passe o terminal negativo do suporte da bateria por dentro do suporte da chave.



Passo 31



De modo que sua extremidade seja conectado em um dos pinos do barramento negativo, como mostrado nas imagens ao lado.

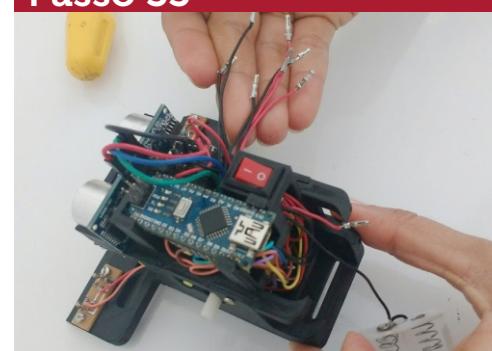
Passo 32



Ainda está faltando alimentar os sensores de piso (IR) e conectar o GND da ponte H. Para que os fios não fiquem expostos, passe cada um deles por dentro do suporte da chave ou por baixo do Arduino.



Passo 33



Depois de passar todos os fios, faremos a ligação dos fios vermelhos no barramento positivo e os fios pretos no barramento negativo.



Passo 34



Como mostrado na imagem ao lado, todos os componentes estão devidamente alimentados.

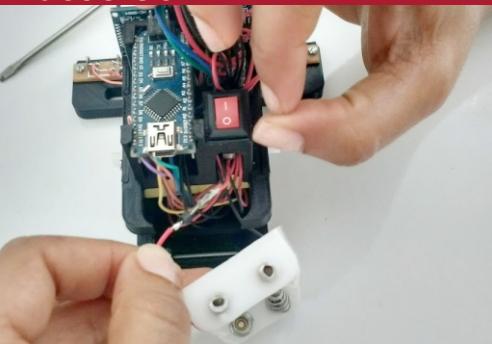


Passo 35



Conecte o terminal positivo do suporte da bateria, no terminal da chave.

Passo 36



Neste passo, concluímos todas as conexões necessários para o funcionamento do nosso robô.

Passo 37



Encaixe o suporte da bateria no chassi, de modo que ele fique fixo, conforme demonstrado.

Passo 38



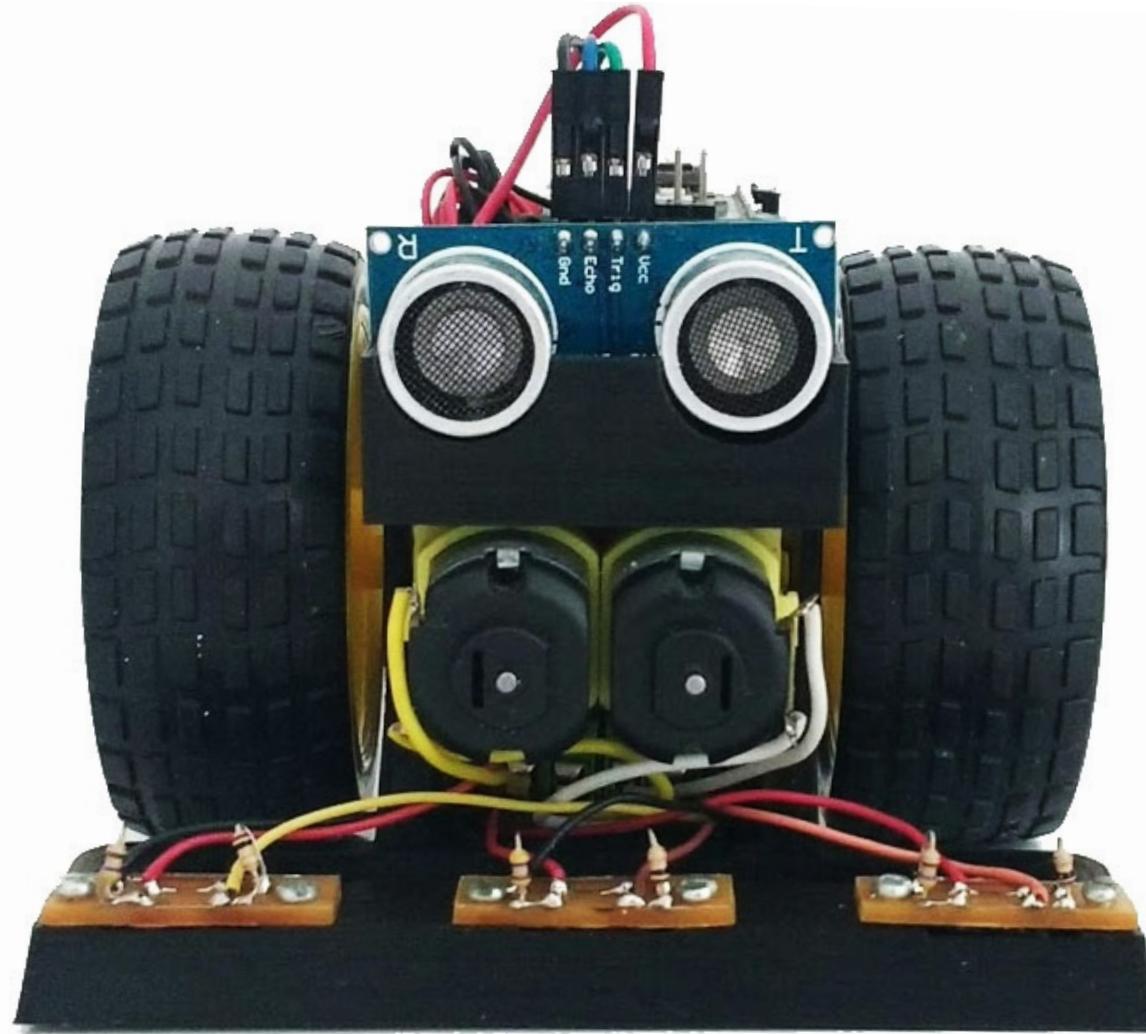
Encaixe as rodas da direita e esquerda no eixo dos motores.

Passo 39



Coloque as baterias que irão fornecer a alimentação do robô.

Prontinho!
Seu robô
está montado...



Agora, bora programar!

Com nosso robô montado, nós vamos dizer a ele como se comportar e como interpretar as informações fornecidas pelos sensores, isto é feito através da programação da placa arduino. Cada tomada de decisão vai depender das condições que nós estabelecermos na programação, por exemplo, se quisermos que o robô ande para frente enquanto estiver sobre a superfície preta, devemos receber do sensor de piso a informação de que superfície preta foi detecta, e em seguida acionar os motores, que permitirão o movimento do robô.

Nas páginas seguintes, vamos conhecer o ambiente utilizado para desenvolver a programação e entender seus principais comandos.

Conhecendo o *IDE do Arduino

Barra de Ferramentas

Composta de botões para: compilar, fazer upload, criar novo arquivo e salvar

Menu

Possui funções para edição
do código, inclusão de bibliotecas x
configuração da placa e porta a ser utilizada

Monitor Serial

Permite enviar e receber informações na forma de texto.

Ambiente de Trabalho

Local onde será
escrito toda a
nossa programação

Console de Feedback

Informa o status do upload ou da compilação. Caso o programa tenha algum erro, o mesmo será informado nessa área



Quando você abrir o programa do Arduino, aparecerão na tela as seções obrigatórias Setup e Loop, assim como, também estará disponível o Monitor Serial. Vamos descrever mais de perto estes itens:

Setup: qualquer código que estiver dentro do `setup()`, é executado uma única vez no início do seu programa. Esta função é utilizada para inicializar variáveis, configurar o modo dos pinos(INPUT ou OUTPUT), inicializar bibliotecas, etc.

Loop: esta função faz precisamente o que o seu nome sugere, e se repete consecutivamente enquanto a placa estiver ligada, permitindo o seu programa mudar e responder a essas mudanças. A maior parte do seu código será executado dentro desta seção.

Monitor Serial: o monitor serial é o elo entre o computador e o Arduino. Com ele você pode enviar e receber informações na forma de texto, útil para depuração e também para controlar o Arduino pelo teclado do computador. Para utilizá-lo, é preciso iniciar a comunicação serial por meio da função `Serial.begin()`, bem como informar a taxa de transmissão. Outra função necessária é a `Serial.println()`, que imprime o valor de uma variável e depois adiciona uma nova linha.

Comandos básicos

pinMode(): define se a porta será uma entrada (INPUT) ou uma saída (OUTPUT). Dois parâmetros devem ser informados à função: o pino a ser usado, e se esse pino será entrada ou saída. Ex.: `pinMode(2, OUTPUT)`.

digitalWrite(): liga ou desliga dispositivos digitais enviando 1 (5V) ou 0 (0V) para a saída. Dois parâmetros devem ser informados: o número do pino e o estado lógico (HIGH/LOW). Ex.: `digitalWrite(2, HIGH)` ou `digitalWrite(2, 1)`.

digitalRead(): lê o valor de um dispositivo digital conectado em um pino. Ex.: `digitalRead(3)`.

analogWrite(): o arduino pode gerar tensões analógicas em 6 de seus 14 pinos digitais com essa função. Dois parâmetros são requeridos: o primeiro indica em qual pino será gerada a tensão; o segundo determina a amplitude dessa tensão, devendo ter os valor entre 0 (para 0V) e 255 (para 5V). Ex.: `analogWrite(5, 180)`.

analogRead(): lê o valor de um dispositivo analógico conectado em um pino.

Ex.: `analogRead(A0)`.

if: o if é uma das estruturas mais básicas de programação em geral. If significa "se" em inglês, e é exatamente isso que ele faz: ele verifica uma expressão e, apenas se ela for verdadeira, executa um conjunto de comandos. Em linguagem natural, ele executa uma lógica do tipo: "se isso for verdadeiro, então faça aquilo"

Ex.: `if (condição) {`

...

}

if - else: o if-else pode ser visto como uma extensão do comando if. Else em inglês significa "caso contrário", e ele faz exatamente o que o nome diz: "se isso for verdadeiro, então faça aquilo, caso contrário, faça outra coisa".

Ex.: `if (condição) {`

...

`} else {`

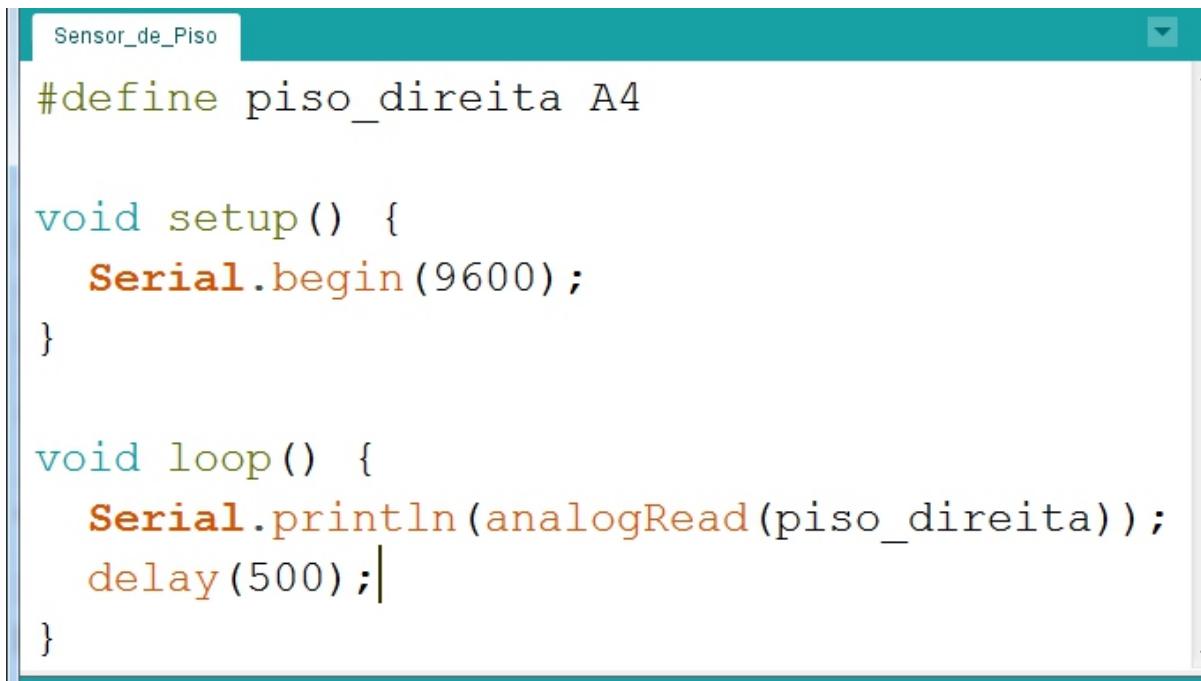
...

}

delay(): pausa o programa por um período em milissegundos indicado pelo parâmetro entre parênteses. Ex.: `delay(1000)`.

Programando... detectando a arena

Nesta etapa devemos fazer a leitura análogica dos sensores de piso da esquerda, central e da direita, e visualizar a alteração dos valores que correspondem as superfícies preta e branca, através do monitor serial. Vejam abaixo o exemplo de leitura do sensor da direita conectado ao pino A4 do arduino.



```
Sensor_de_Piso

#define piso_direita A4

void setup() {
    Serial.begin(9600);
}

void loop() {
    Serial.println(analogRead(piso_direita));
    delay(500);
}
```

Lembre de anotar o valor de referência para a superfície preta e branca, pois vamos precisar desses valores para a programação do robô.



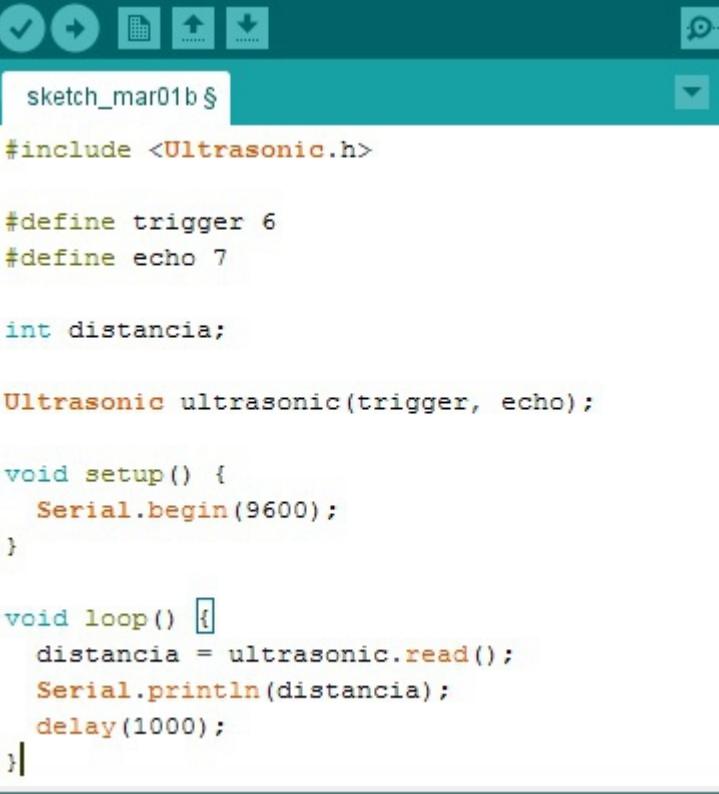
Hora de explorar!

**Teste seu código e
verifique se seu robô está
funcionando adequadamente.**

**Caso precise, peça
orientação ao tutor.**

Programando... detectando o oponente

Agora vamos continuar explorando as capacidades dos sensores instalados em nosso robô de modo que ele possa identificar um obstáculo.



Insere a biblioteca do sensor ultrassônico.

```
#include <Ultrasonic.h>
```

Define os pinos que serão usados para receber os sinais.

```
#define trigger 6  
#define echo 7
```

Variável que armazena o valor do sensor.

```
int distancia;
```

Inicializa o sensor nos pinos definidos.

```
Ultrasonic ultrasonic(trigger, echo);
```

Faz a leitura do sensor e permite visualizar pelo monitor o serial.

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    distancia = ultrasonic.read();  
    Serial.println(distancia);  
    delay(1000);  
}
```

Biblioteca: Uma biblioteca nada mais é do que um conjunto de instruções desenvolvidas para executar tarefas específicas relacionadas a um determinado dispositivo, tornando possível o compartilhamento de códigos.

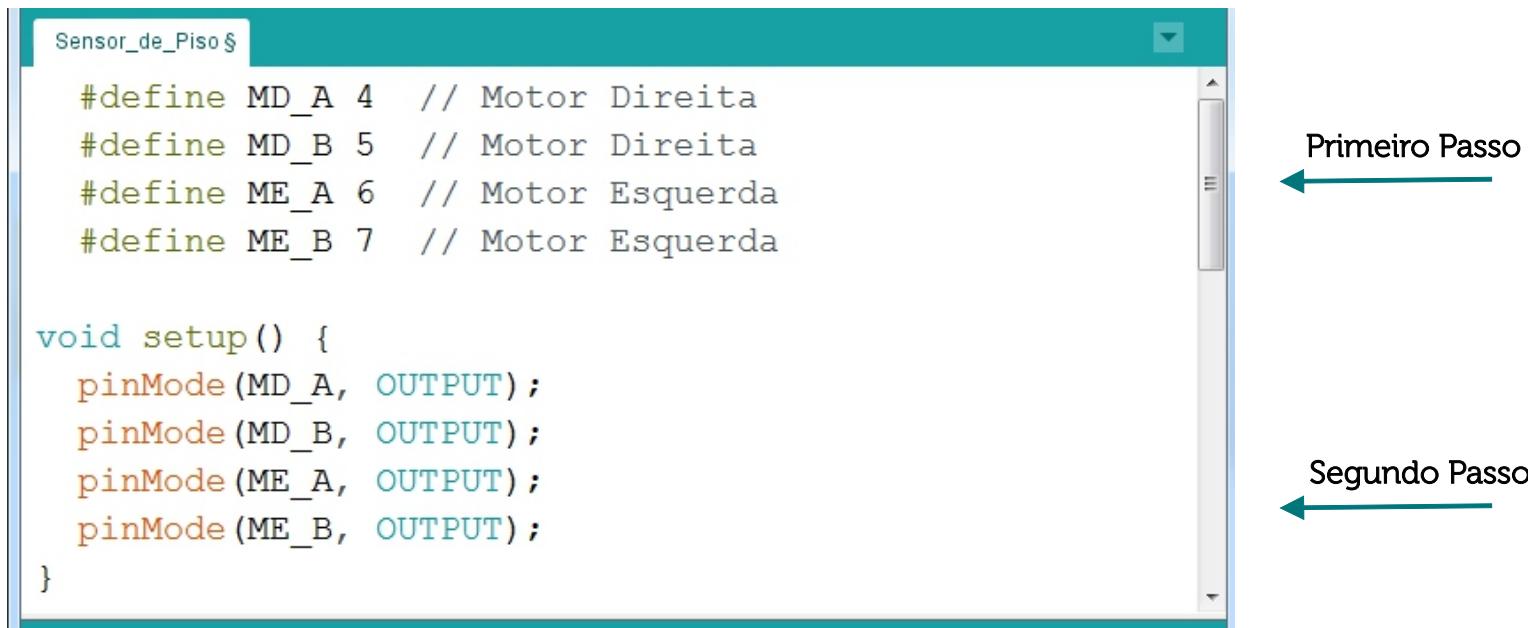
Hora de explorar!

**Teste seu código e
verifique se seu robô está
funcionando adequadamente.**

**Caso precise, peça
orientação ao tutor.**

Programando... movimentando o robô

Agora que nosso robô é capaz de "sentir" o mundo exterior, ele pode usar essa informação para se movimentar pelo ambiente. Para isso vamos seguir alguns passos:



```
Sensor_de_Piso $
```

```
#define MD_A 4 // Motor Direita
#define MD_B 5 // Motor Direita
#define ME_A 6 // Motor Esquerda
#define ME_B 7 // Motor Esquerda

void setup() {
  pinMode(MD_A, OUTPUT);
  pinMode(MD_B, OUTPUT);
  pinMode(ME_A, OUTPUT);
  pinMode(ME_B, OUTPUT);
}
```

Primeiro Passo

Segundo Passo

O primeiro passo nesta etapa é declarar cada motor ao seu respectivo pino no arduino, pois isso facilitará a interpretação do código. No segundo passo, devemos configurar esses pinos como pinos de saída. E o terceiro passo consiste basicamente em fazer diferentes combinações e verificar seus respectivos sentidos de rotação, pois isso determina se o robô vai se movimentar para frente, para trás ou girar.

```
Sensor_de_Piso §

void loop() {
    // andar para frente
    digitalWrite(MD_A, 1);
    digitalWrite(MD_B, 0);
    digitalWrite(ME_A, 1);
    digitalWrite(ME_B, 0);
    delay(1000);

    // parar
    digitalWrite(MD_A, 0);
    digitalWrite(MD_B, 0);
    digitalWrite(ME_A, 0);
    digitalWrite(ME_B, 0);
    delay(50);

    // andar para trás
    digitalWrite(MD_A, 0);
    digitalWrite(MD_B, 1);
    digitalWrite(ME_A, 0);
    digitalWrite(ME_B, 1);
    delay(1000);
}
```

Terceiro Passo

Sabendo que o robô realiza o movimento de girar quando os motores estão em sentidos opostos, você já pode acrescentar à programação o código que permita girar para a direita e girar para a esquerda.

Dica: Sempre que necessário utilize os comentários, pois facilita a interpretação diante do código.

Hora de explorar!

**Teste seu código e
verifique se seu robô está
funcionando adequadamente.**

**Caso precise, peça
orientação ao tutor.**

**Agora que você aprendeu
como programar seu robô
para que ele reaja às
características do ambiente,
iremos propor uma série
de desafios, que lhe ajudarão a
expandir suas habilidades e
criatividade para a
resolução de problemas.**

Lista de desafios!

- 1) O movimento do robô deve ocorrer 5 segundos depois que chave for acionada.
- 2) O robô deve andar para frente enquanto estiver sobre a superfície preta, caso alcance a superfície branca, o mesmo deve parar.
- 3) Caso a superfície branca seja detectada pelo sensor da esquerda, o robô deve andar para trás por 1 segundo, girar para a esquerda e seguir em frente.
- 4) Caso a superfície branca seja detectada pelo sensor da direita, o robô deve andar para trás por 1 segundo, girar para a direita e seguir em frente.
- 5) Se o robô detectar um obstáculo a 30cm de distância, o mesmo deve seguir em direção ao obstáculo, a fim de empurrá-lo para fora da arena.

REFERÊNCIAS

ZABALA, A. **A Prática Educativa:** como ensinar. Porto Alegre: Artmed, 1998.