

INTRODUÇÃO

A Robótica educacional é uma área do conhecimento que utiliza a programação, mecânica e eletrônica como ferramenta tecnológica e educacional. Seu ambiente de aprendizagem está em devida consonância com a pedagogia construtivista uma vez que se baseia na aprendizagem por descobertas e propicia a experimentação e a reflexão dos alunos. Dessa forma, esse instrumento modifica, significativamente, a transmissão e aprendizagem dos conteúdos.

O QUE É ROBOTICA?

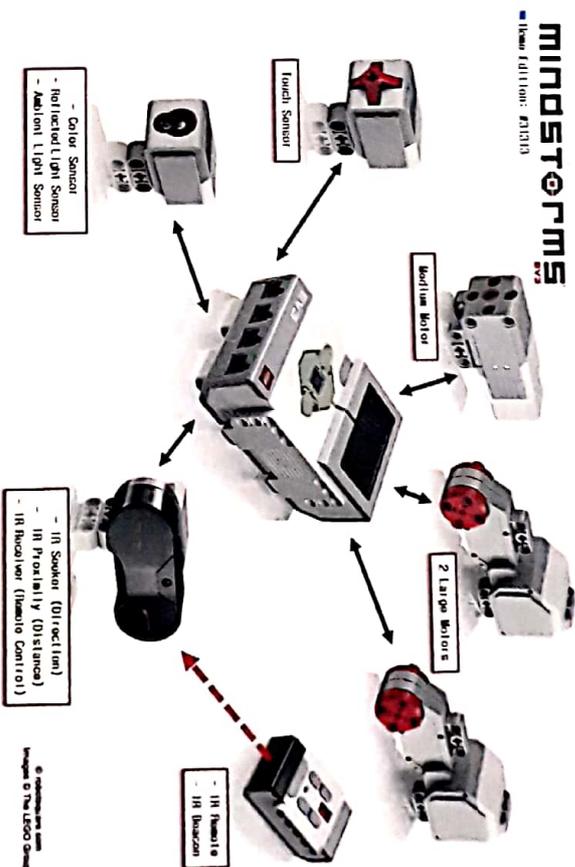
É um conjunto de elementos elétricos e mecânicos que, através de sensores e atuadores, são capazes de interagir com o meio, executando tarefas pré-programadas.

O QUE É PROGRAMAÇÃO?

É um conjunto de códigos que permite ao robô executar uma tarefa pré-definida.

LEGO MINDSTORMS EV3

LEGO MINDSTORMS EV3 será um dos kits de robótica utilizados durante as aulas. Suas principais partes são sensores, motores e bloco de comandos.

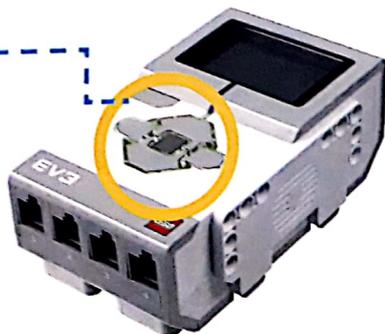


BLOCO DE COMANDOS

O bloco é a parte responsável pelo processamento das informações do robô, de acordo com a programação instalada. Na parte inferior, encontram-se quatro portas: 1, 2, 3 e 4 para conectar os sensores (entrada de dados); na superior, 4 portas de saída: A, B, C e D para os motores (saída de dados). Além da tela LCD, alto-falante e botões de configuração.



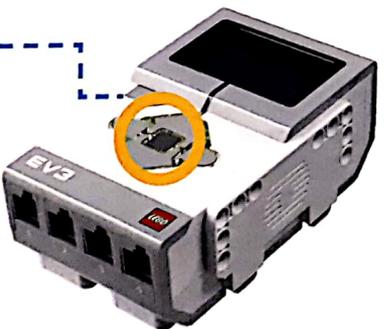
Os quatro botões de tom mais claro consistem no direcionamento da tela de LCD do robô -para cima, para baixo, para a direita e para a esquerda. Observar na figura:



O botão mais isolado é o botão retorno. Este consiste em retornar para a atividade anterior. Observar na figura:



O botão central é o selecionar. Esse botão é de grande importância para o robô, pois ele consiste em selecionar qualquer atividade que o robô venha a executar e também possibilita acioná-lo. Para ligar, basta pressioná-lo e esperar até que a luz seja acesa, depois é só usar os botões de navegação para o manuseio adequado.



MANUSEANDO O BLOCO DE COMANDOS

Inicialmente, quando ligar o bloco, a seção play surgirá na tela de LCD, onde estão os programas recentemente inseridos e prontos para serem executados. Porém, muitas vezes, não se sabe localizar o próprio programa, então basta ir à seção ao lado: seção de pastas e procurar qual a pasta da sua programação, localizada com o nome que você inserir no software de programação. Basta selecionar a pasta e o comando play para rodar sua programação.

Nessa mesma seção (seção de pastas) você encontra a pasta SD Card, onde constará sua programação, caso opte por esse recurso.

MOTORES

Os motores são totalmente responsáveis pelos movimentos do robô, exercem funções de extrema importância e são fundamentais para atividades mecânicas.

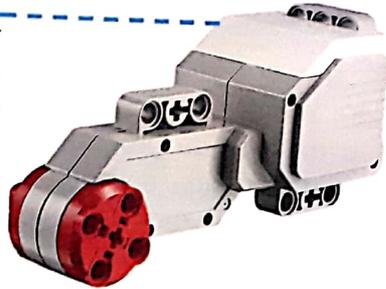
SERVO MOTOR

O servo motor é um atuador bastante utilizado por sua alta precisão no controle e por sua posição angular. Possui sensor de rotação e controle de velocidade embutidos.

Seus comandos para programação estão na aba 'Ação' (Parte verde).

Servo Motor Grande

É um motor de maior potência. Utilizado para transportar cargas mais pesadas, porém possui menor velocidade e maiores dimensões em relação ao motor médio.



MÉDIO MOTOR

O médio motor é um motor de menor porte em relação ao motor grande, que consiste em adicionar movimento em locais menores, possibilitando, assim, um projeto sem ocupar muito espaço. Por esse motivo, ele é bem usado em locais e projetos específicos, adquirindo um resultado satisfatório e mais compacto para quem o utiliza.

Relacionados à programação, existe um bloco bem específico, "Médio Motor", localizado na ala 'ação' (parte verde).



SENSORES

Os sensores são dispositivos eletrônicos cujo objetivo é captar/detectar informações do ambiente no qual eles estão inseridos. Essas informações são utilizadas na programação que comanda as ações exercidas pelo robô.

COLOR SENSOR

O color sensor, por meio da reflexão da luz, tem a capacidade de distinguir sete cores diferentes (azul, vermelha, verde, amarela, marrom, branca e preta), bem como indicar a ausência de superfície, ele também funciona como sensor de luz, medindo sua intensidade da luz refletida ou do ambiente.



TOUCH SENSOR

O sensor de toque é capaz de detectar quando o robô for pressionado ou liberado. É utilizado em diversas funções, como controle do robô e detecção de obstáculos.



INFRARED SENSOR

Sensor Infravermelho e Baliza Remota Infravermelha.

O sensor de infravermelho possui duas funções principais: proximidade e remoto. No modo "proximidade" ele é capaz de detectar a presença de objetos e sua distância, que varia de 0 a 100, essa distância não é uma escala específica como centímetros ou polegadas.



SENSOR ULTRASSÔNICO

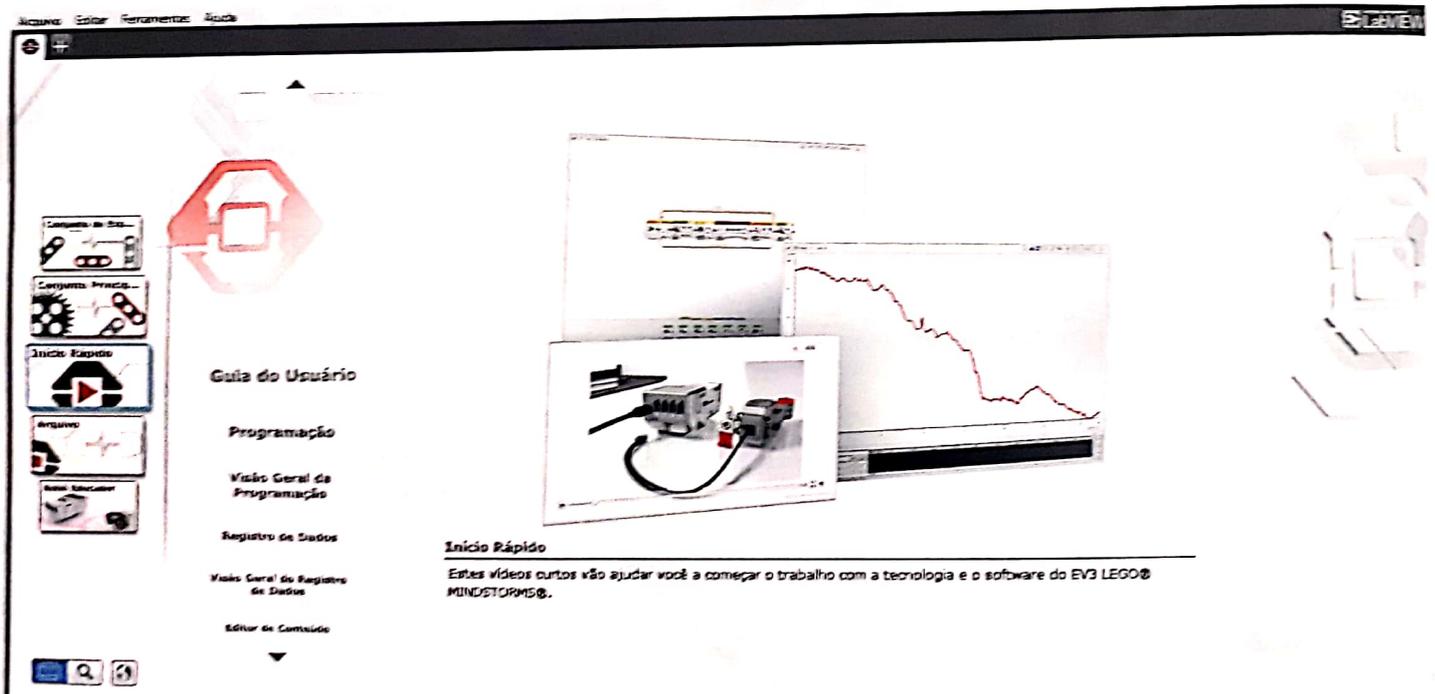
O sensor ultrassônico também serve para sentir a presença e medir a distância dos objetos, com a capacidade de fornecer as medidas em duas unidades específicas: centímetros ou polegadas. Seu funcionamento se resume à emissão e à captação de ondas ultrassônicas. Um exemplo bem claro desse dispositivo na natureza é usado pelos morcegos - a Ecolocalização.

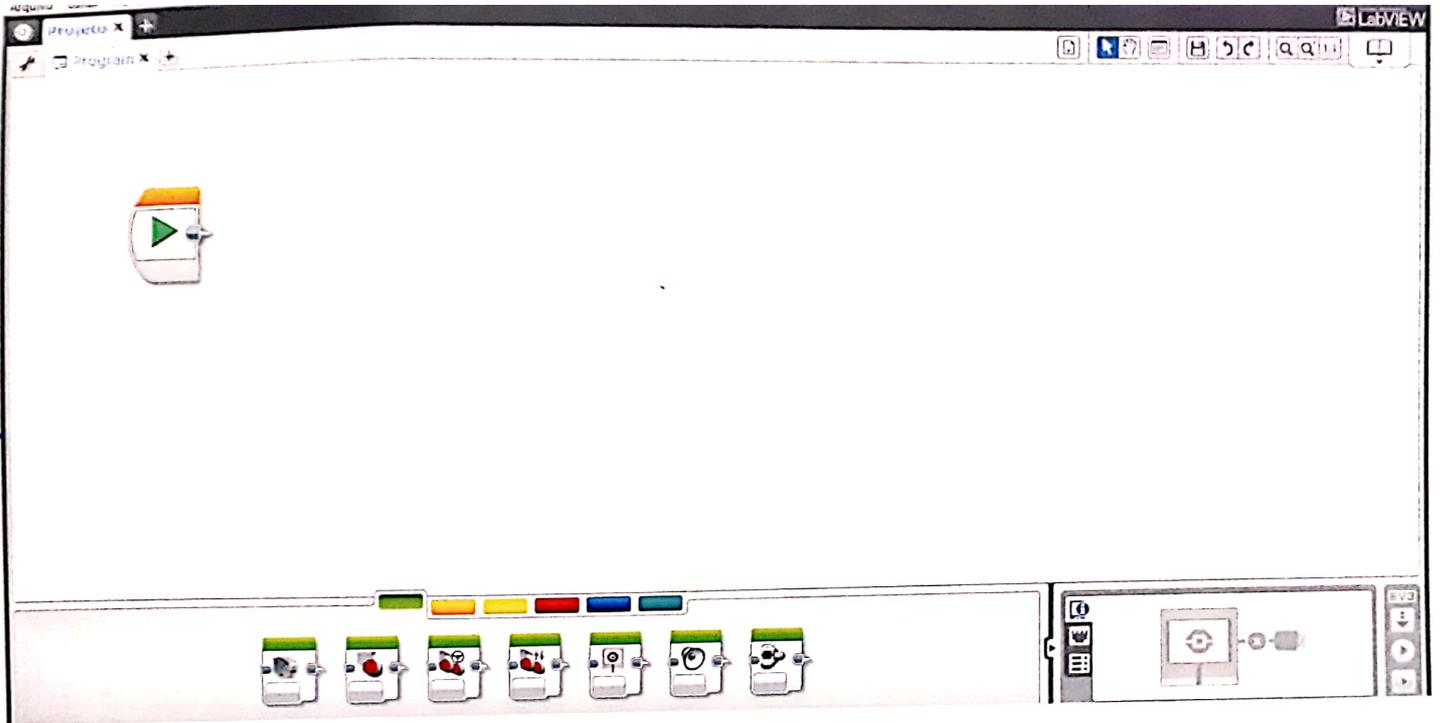


SOFTWARE LEGO MINDSTORMS EV3

O software Lego Mindstorms EV3 foi produzido para ser um ambiente de programação intuitivo, seu formato de programação em blocos permite a criação rápida de códigos até para quem está iniciando no estudo de linguagem de programação. A imagem a seguir é uma representação da tela inicial do software de programação.

Para dar início ao ambiente de programação, basta clicar no local (Adicionar projeto) indicado na figura. A imagem mostra o ambiente de programação.





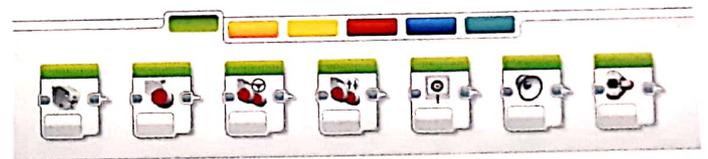
PALETA DE COMANDOS

Na parte de baixo está localizada a paleta de comandos com os blocos utilizados na programação.

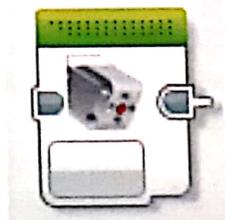
BLOCOS DE PROGRAMAÇÃO

AÇÃO

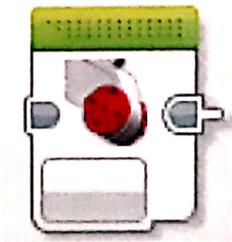
Nessa aba estão localizadas as ações físicas que o robô pode executar, que são: mover os motores, tocar sons, mostrar informações no display e acender as luzes do bloco de comandos.



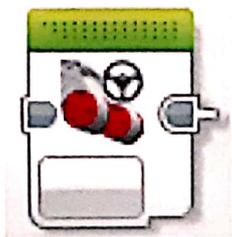
Em ordem, os blocos possíveis encontrados nesta ala são:



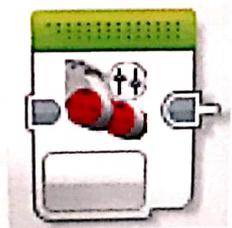
A) Médio motor: Como o nome do próprio bloco já diz, baseia-se em configurar exclusivamente o médio motor.



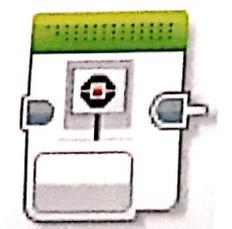
B) Motor grande: este bloco consiste em configurar apenas um único motor.



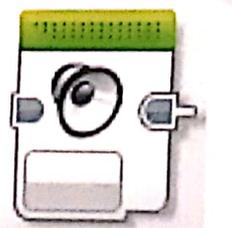
C) Mover direção: este bloco possibilita a configuração de dois motores ao mesmo tempo, além de auxiliar para fazer curvas com a função direção.



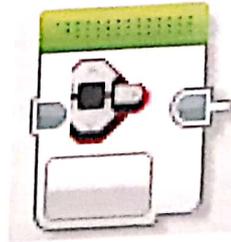
D) Move tanque: este bloco também possibilita configurar dois motores ao mesmo tempo, com a diferença de que para fazer curvas é necessário controlar a velocidade de cada motor.



E) Monitor: este bloco possibilita a configuração da tela do robô em uma determinada atividade.



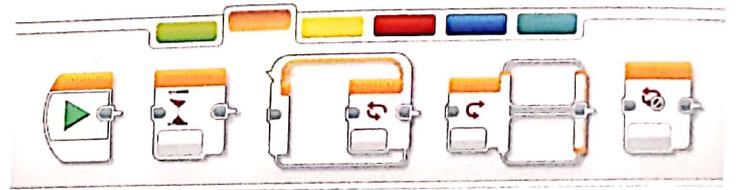
F) Som: este bloco possibilita a reprodução de sons pelo robô



G) Luzes de estado do bloco: este bloco equivale à configuração das luzes presentes em volta dos botões do bloco, podendo modificar a cor preferida entre verde, vermelha e amarela.

CONTROLE DE FLUXO

Estes blocos consistem em estrutura de controle, indicam em quais momentos serão realizadas as ações. São os caminhos pelos quais a programação deve seguir.



Por ordem, os blocos possíveis de serem encontrados nesse conjunto são:

A) Iniciar: este é o bloco inicial e deve estar presente em toda programação. Ele dá início à programação. Para que todo o comando seja válido, deverá estar ligado a este bloco.

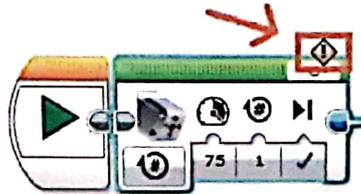


B) Esperar: este bloco consiste em espera. Ele pode ser programado para que o robô possa esperar até que algum dispositivo escolhido seja acionado ou um tempo seja aguardado.

de dados. É necessário criar um nome diferente para cada variável.

Dica de programação

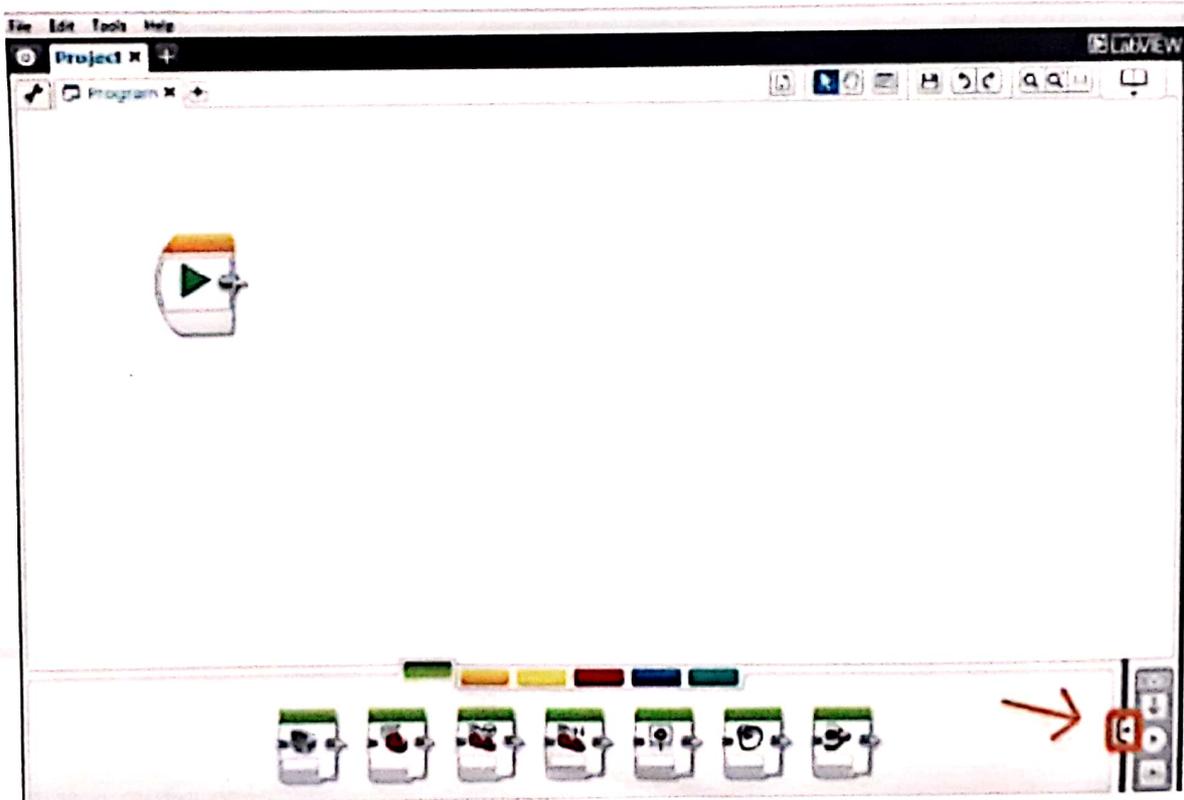
Caso surja uma placa de aviso na porta de algum motor ou sensor, ocorreu algum disparate e será necessário o ajuste. Você pode ter colocado algum cabo errado, como ter inserido o motor médio na porta do large grande.



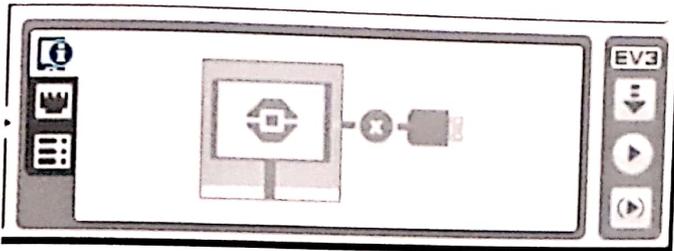
DISPLAY EV3

No canto inferior da página de programação, encontra-se o display do bloco de comandos EV3. Nele é possível fazer alguns ajustes básicos como: verificar a bateria restante, observar a memória ocupada, ajustar os sensores, bem como fazer o download da programação para o bloco.

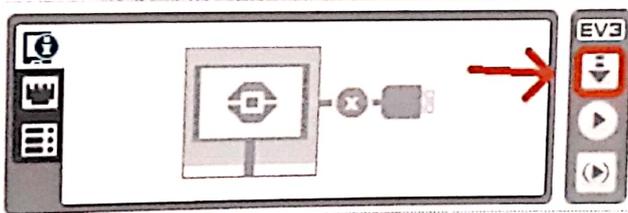
Se o display não abrir automaticamente, basta clicar na seta Expand para abri-lo. Como mostra na imagem abaixo:



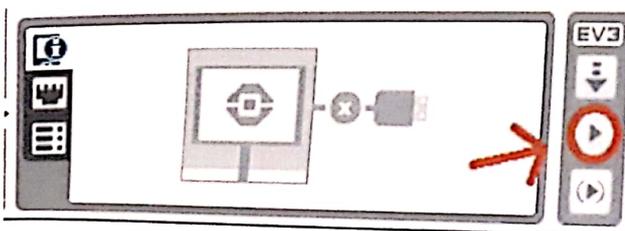
E depois de aberto, a imagem do display é esta:



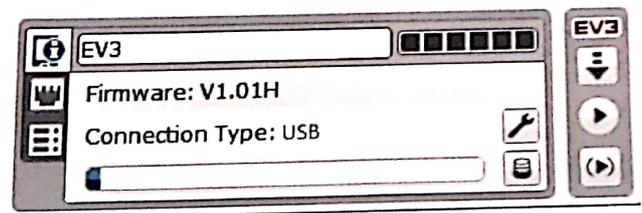
Inicialmente o display permanece desativado, pois o robô não está conectado ao computador. Nesses botões, será encontrado o Download. Esse botão servirá para baixar sua programação no robô:



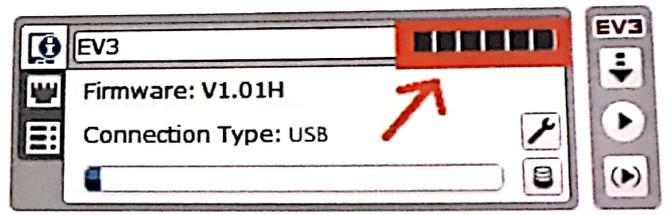
O botão Download and return consiste em iniciar a programação e fazê-la rodar automaticamente.



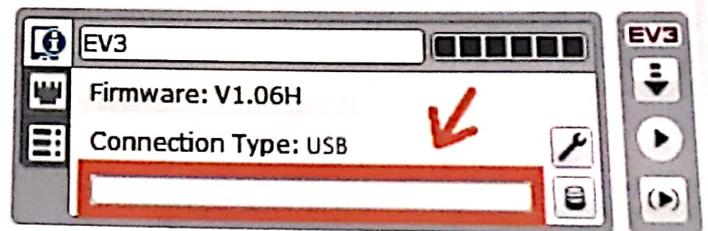
O botão Run selected baixará a programação, mas só irá executar o que for selecionado. Finalmente, quando conectar o robô ao computador, o acesso às informações será possível e poderá manusear seu robô pelo PC. Observe na figura:



O nível de bateria fica localizado na parte superior direita do display:



A memória do blocos se localiza na parte inferior:



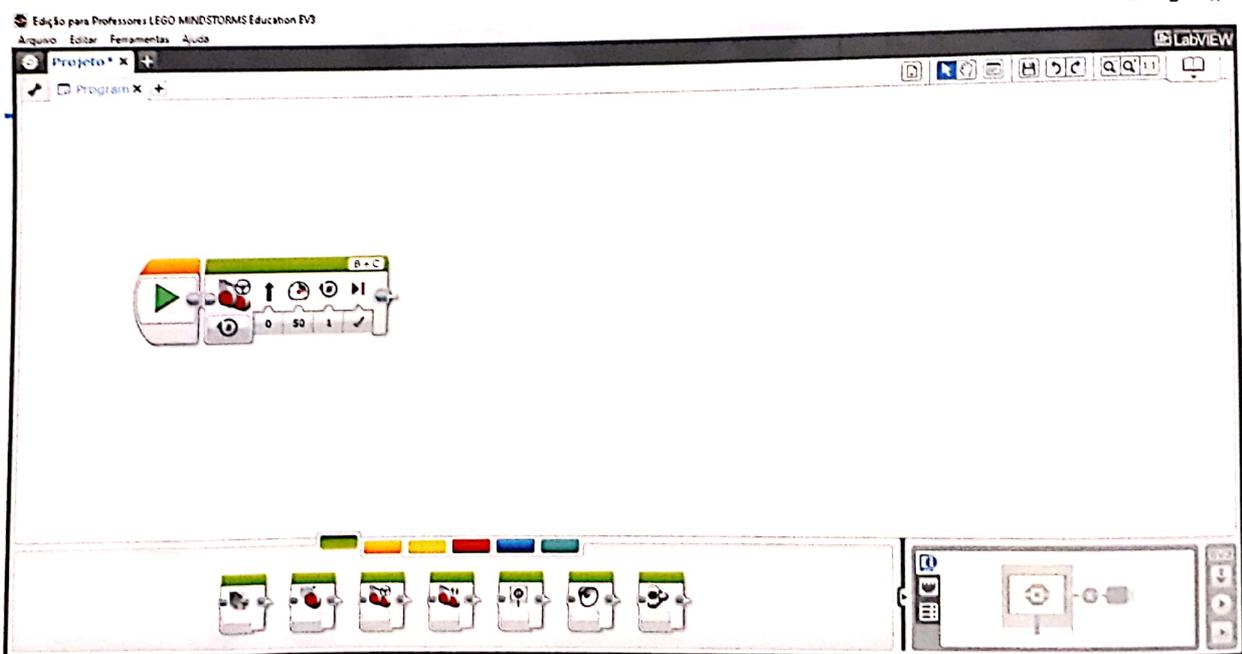
Agora, com o carro esteira pronto, precisamos criar um código de programação para que o nosso robô execute tarefas.

CONTROLANDO OS MOTORES

O primeiro comando que iremos passar para o robô será para fazer com que ele se movimente. Para isso utilizaremos o comando "mover tanque".

Nesse comando existem cinco itens a serem configurados: portas dos motores, controle de giro, direção, potência, quantidade de rotações, graus ou segundos e travamento do motor ao final do comando.

Combinados vários destes comandos, faremos o robô movimentar-se em uma trajetória quadrada, com o seguinte código:



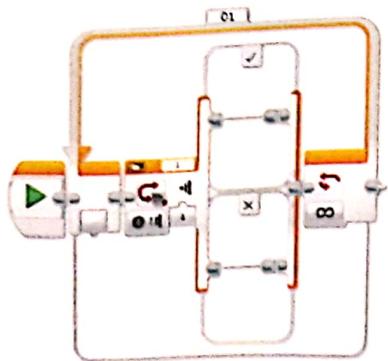
IMPLEMENTANDO O CICLO

Existem várias formas e códigos para que um robô execute uma mesma tarefa. Esse mesmo código pode ser resumido utilizando o "ciclo" e configurando-o para o modo contagem.

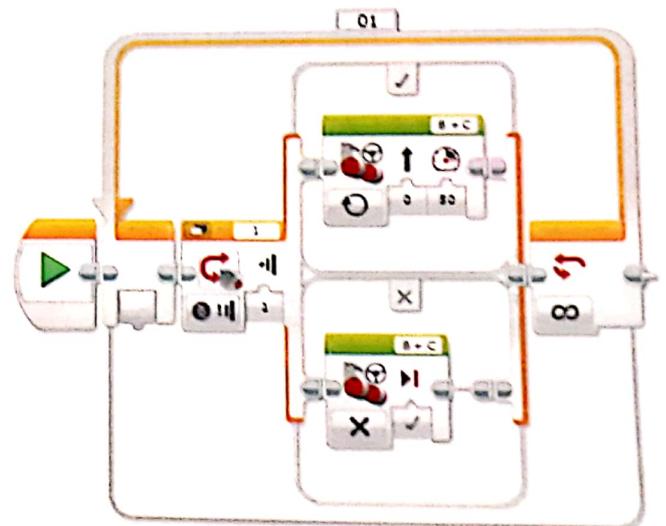


TOMANDO DECISÕES POR MEIO DO SENSOR DE TOQUE

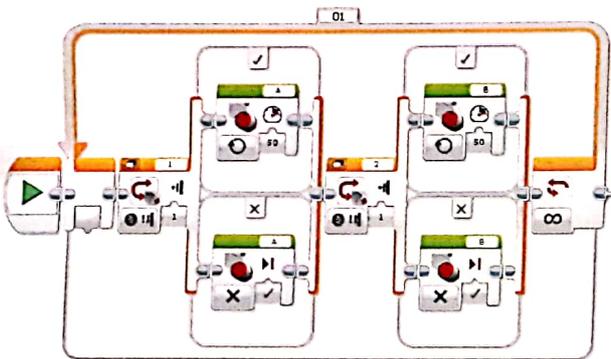
Agora que já aprendemos sobre o ciclo, poderemos utilizar sensores em nosso projeto e deixá-lo mais elaborado. Dessa forma é necessário utilizar o comando "comutação". O sensor de toque é o que possui o funcionamento mais simples e também o mais fácil de programar, então será o primeiro que iremos usar:



A comutação fará o robô tomar uma decisão por meio de um teste, com base no que foi configurado e executará os comandos dependendo do resultado desse teste. Se o resultado for verdadeiro, serão executados os comandos colocados na parte de cima; já se for falso, serão executados os comandos colocados na parte de baixo. Para o sensor de toque, só é preciso configurar a comutação para testar se ele está pressionado, solto ou pulsado. A programação a seguir fará com que o robô movimente-se para frente sempre que pressionarmos o sensor e parar quando não pressionarmos.



Para aperfeiçoar esse código de forma que o carro esteira também faça curvas, adicionaremos mais um sensor de toque. E o código ficará assim:



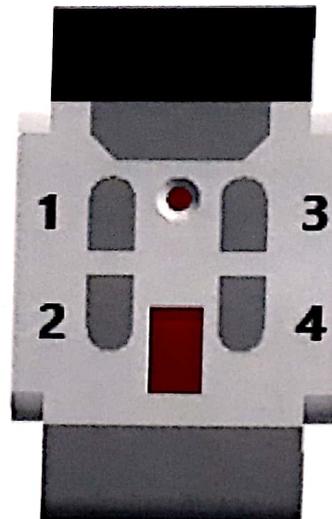
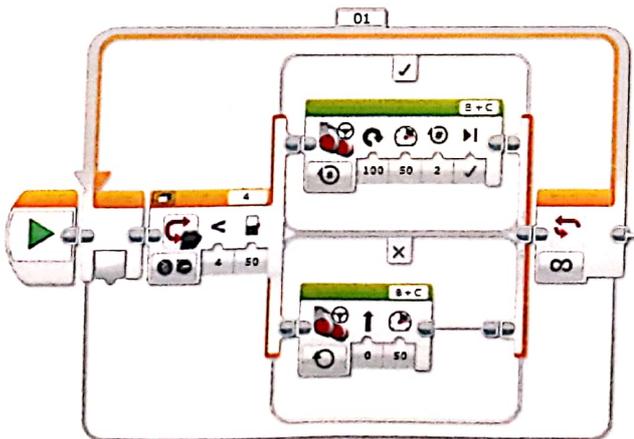
Nessa programação, o robô irá se movimentar em linha reta, enquanto a distância for maior que 50 e fará uma curva quando a distância for menor que 50.

CONTROLANDO O ROBÔ REMOTAMENTE

No modo remoto, o sensor irá captar sinais de luz infravermelha do controle remoto. Dessa forma, precisaremos configurar, na comutação, o canal selecionado no controle remoto e o botão que será realizado o teste, se está pressionado ou não. No código são testados se os botões do controle estão pressionados, com um comando para cada resultado verdadeiro.

DETECTANDO OBSTÁCULOS COM O SENSOR INFRAVERMELHO

Sua programação é bem parecida com a do sensor anterior, mudando apenas o item a ser configurado, que, nesse caso, é a distância a ser testada.



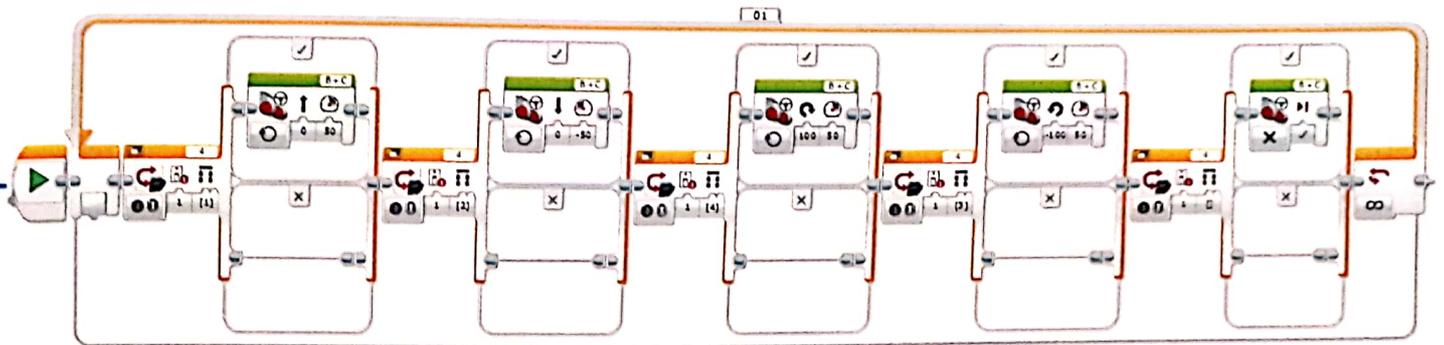
Botão 1 pressionado:
andar para frente e em linha reta

Botão 2 pressionado:
andar para trás e em linha reta

Botão 3 pressionado:
curva para direita.

Botão 4 pressionado:
curva para esquerda.

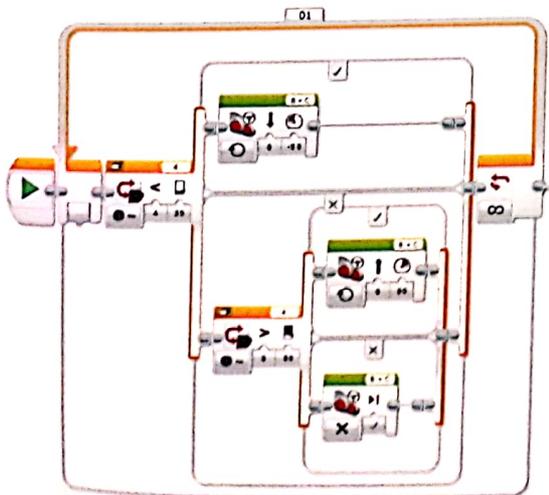
Nenhum botão pressionado:
parado.



SUBSTITUINDO PELO SENSOR DE ULTRASSOM

Sua programação é praticamente igual a do modo proximidade do sensor infravermelho, bastando configurar a distância a ser testada e sua unidade de medida.

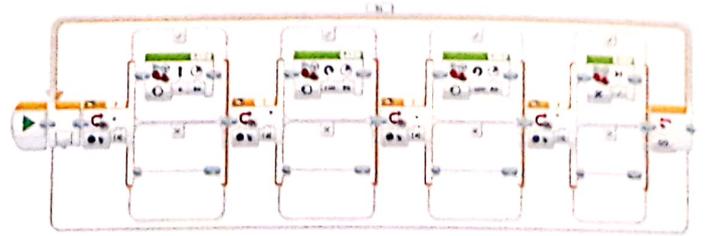
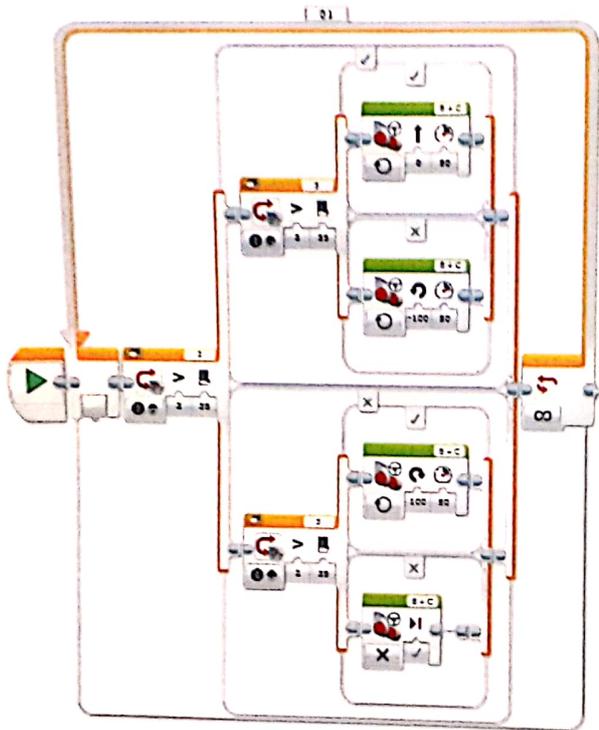
Nesse código, são realizados dois testes de distâncias: o primeiro testa se a ela é menor que 30 cm, se o resultado for verdadeiro o carro irá para trás, caso seja falso, o segundo teste será realizado. Nesse caso, se o segundo teste for verdadeiro, ou seja, a distância seja maior que 80 cm, o carro irá para frente. Mas se ele também for falso, quer dizer que a distância está entre 30 cm e 80 cm, logo o carro ficará parado.



CONTROLANDO O ROBÔ POR MEIO DA LUZ

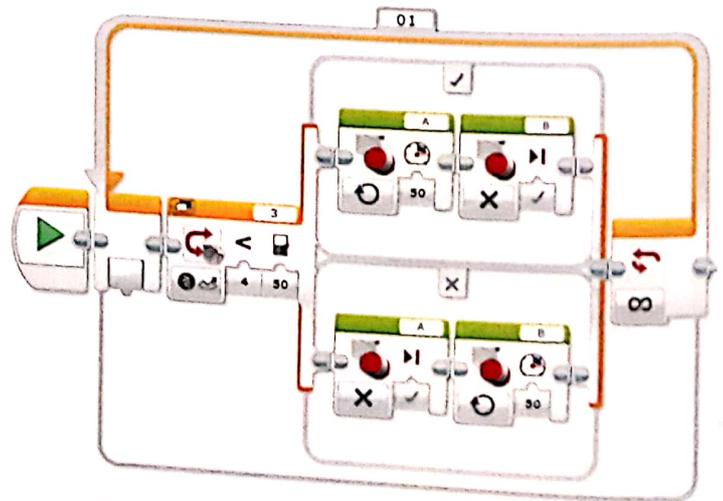
O sensor de luz pode trabalhar em 3 modos que são: modo cor, modo intensidade de luz refletida e modo intensidade de luz ambiente. No modo intensidade de luz refletida, ele próprio emite uma luz vermelha e mede a taxa de luz refletida pela superfície, que varia de 0 a 100. O modo cor funciona de forma semelhante, mas nesse caso ele irá detectar a cor da superfície. Já no modo intensidade de luz ambiente, o sensor irá detectar

as luzes que já estão presentes no ambiente, como a luz do sol ou de uma vela, por exemplo.



Verde: frente
Azul: direita
Amarela: esquerda
Ausência de superfície: parado

O modo intensidade de luz refletida é o mais utilizado desse sensor, tendo as mais variadas aplicações, desde alarmes até um robô seguidor de linha, sendo este essencial em competições de robótica como a OBR (Olimpíada Brasileira de Robótica). No código a seguir, faremos o robô seguir uma linha preta. O sensor deve estar posicionado, apontando para o chão, o mais próximo possível, mas sem tocá-lo.



Com esse código e o nosso robô equipado com dois sensores de luz apontando para cima, poderemos controlá-lo com uma lanterna. Se colocarmos a luz da lanterna sobre os dois sensores, o carro moverá para frente; já se colocarmos apenas sobre o sensor 2 ele irá para direita, para girar para esquerda é só colocarmos a luz apenas sobre o sensor 3 e para fazer o carro parar, basta apagar a lanterna.

Alterando a configuração da comutação para o modo cor, controlamos o carro por meio das cores.

Esse código, utilizando apenas um sensor, funciona bem em alguns formatos de pistas, mas se existirem curvas muito fechadas, com ângulos inferiores a 90 graus ou GAP's (falhas na linha) o robô irá perder a linha.

Uma solução já encontrada para esse problema seria adicionar outro sensor de luz.

