

**INSTITUTO FEDERAL SUL-RIO-GRANDENSE**

*CAMPUS* PELOTAS VISCONDE DA GRAÇA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS E TECNOLOGIAS NA EDUCAÇÃO

MESTRADO PROFISSIONAL EM CIÊNCIAS E TECNOLOGIAS NA EDUCAÇÃO

**TUTORIAL PARA A UTILIZAÇÃO DOS KITS DE ROBÓTICA DA  
MARCA LEGO MINDSTORMS NXT 2.0**

**MARCELO SCHILLER DE AZEVEDO**

**PRODUTO EDUCACIONAL DO TRABALHO DE DISSERTAÇÃO: ROBÓTICA EDUCACIONAL DOS  
ANOS FINAIS DO ENSINO FUNDAMENTAL: UM ESTUDO DE CASO**

**ORIENTADOR: PROF. DR. MARCOS ANDRÉ BETEMPS VAZ DA SILVA**

Pelotas - RS

Maio / 2017

# LISTA DE FIGURAS

Figura 1 - Imagem de embalagem do Kit LEGO Mindstorms NXT 2.0.....	9
Figura 2 - NXT Intelligent Brick.....	10
Figura 3 - NXT Intelligent Brick com sensores e atuadores .....	10
Figura 4 - NXT Intelligent Brick.....	11
Figura 5 - Representação externa e interna do servo motor .....	11
Figura 6 - Sensor de luminosidade.....	12
Figura 7 - Sensor de toque.....	12
Figura 8 - Sensor ultrassônico.....	13
Figura 9 - Montagem 01 – Fonte: Software LEGO Mindstorms NXT .....	15
Figura 10 - Montagem 02 – Fonte: Software LEGO Mindstorms NXT .....	15
Figura 11 - Montagem 03 – Fonte: Software LEGO Mindstorms NXT .....	15
Figura 12 - Montagem 04 – Fonte: Software LEGO Mindstorms NXT .....	16
Figura 13 - Montagem 05 – Fonte: Software LEGO Mindstorms NXT .....	16
Figura 14 - Montagem 06 – Fonte: Software LEGO Mindstorms NXT .....	16
Figura 15 - Montagem 07 – Fonte: Software LEGO Mindstorms NXT .....	17
Figura 16 - Montagem 08 – Fonte: Software LEGO Mindstorms NXT .....	17
Figura 17 - Montagem 09 – Fonte: Software LEGO Mindstorms NXT .....	18
Figura 18 - Montagem 10 – Fonte: Software LEGO Mindstorms NXT .....	18
Figura 19 - Montagem 11 – Fonte: Software LEGO Mindstorms NXT .....	18
Figura 20 - Montagem 12 – Fonte: Software LEGO Mindstorms NXT .....	19
Figura 21 - Montagem 13 – Fonte: Software LEGO Mindstorms NXT .....	19
Figura 22 - Montagem 14 – Fonte: Software LEGO Mindstorms NXT .....	19
Figura 23 - Montagem 15 – Fonte: Software LEGO Mindstorms NXT .....	20
Figura 24 - Montagem 16 – Fonte: Software LEGO Mindstorms NXT .....	20
Figura 25 - Montagem 17 – Fonte: Software LEGO Mindstorms NXT .....	20
Figura 26 - Montagem 18 – Fonte: Software LEGO Mindstorms NXT .....	21
Figura 27 - Montagem 19 – Fonte: Software LEGO Mindstorms NXT .....	21
Figura 28 - Tela inicial do Software – Fonte: Software LEGO Mindstorms NXT .....	22

Figura 29 - Tela de início da programação – Fonte: Software LEGO Mindstorms NXT .....	23
Figura 30 - Funções de comunicação do NXT .....	23
Figura 31 - Exemplo de programação .....	24
Figura 32 - Fluxograma da programação .....	24
Figura 33 - Bloco Mover (Motor).....	25
Figura 34 – 1ªTarefa - Programação .....	25
Figura 35 - Configuração do bloco Mover .....	25
Figura 36 - Configuração do bloco Mover para a 1ª tarefa.....	26
Figura 37 – 2ªTarefa - Programação .....	27
Figura 38 - Configuração do segundo bloco Mover para a 2ª tarefa .....	27
Figura 39 – 3ªTarefa - Programação .....	28
Figura 40 - Configuração do primeiro bloco Mover para a 3ª tarefa.....	28
Figura 41 - Configuração do segundo bloco Mover para a 3ª tarefa .....	28
Figura 42 - Bloco Sound (Som) .....	28
Figura 43 - Configuração do bloco Sound .....	28
Figura 44 - Bloco Display (Tela) .....	29
Figura 45 - Configuração do bloco Display .....	29
Figura 46 – 4ªTarefa - Programação .....	29
Figura 47 - Bloco Touch (Toque).....	29
Figura 48 - Configuração do bloco Touch .....	30
Figura 49 – 5ªTarefa - Programação .....	30
Figura 50 - Bloco Loop .....	30
Figura 51 – 6ªTarefa - Programação .....	30
Figura 52 - Bloco Switch (SE) .....	31
Figura 53 – Configuração do bloco Switch .....	31
Figura 54– 7ªTarefa - Programação .....	32
Figura 55 – 8ªTarefa - Programação .....	33
Figura 56 - Pista para robô seguidor de linha.....	34
Figura 57– Montagem do Sensor de cor - 01 – Fonte: Software LEGO Mindstorms NXT .....	34
Figura 58– Montagem do Sensor de cor - 02 – Fonte: Software LEGO Mindstorms NXT .....	35

Figura 59 – Montagem do Sensor de cor - 03 – Fonte: Software LEGO Mindstorms NXT .....	35
Figura 60 - Selecionando sensor de cor no comando Switch.....	35
Figura 61 - Configuração do bloco Switch para sensor de cor .....	36
Figura 62 - 9ªTarefa - Programação.....	36
Figura 63 - Bloco Timer .....	37
Figura 64 - Configuração do bloco Timer .....	37
Figura 65 – 10ªTarefa – Programação .....	37
Figura 66 - Configuração do Sensor de cor S1 .....	38
Figura 67 - Configuração do Sensor de cor S2 .....	38
Figura 68 - Configuração do Motor M1.....	38
Figura 69 - Configuração do Motor M2.....	38
Figura 70 - Configuração do Motor M3.....	39
Figura 71 - Configuração do Motor M4.....	39
Figura 72 – 11ªTarefa – Programação .....	40
Figura 73 – 12ªTarefa – Programação .....	41
Figura 74 – Configuração do sensor ultrassônico .....	41
Figura 75 – 13ªTarefa – Programação .....	42
Figura 76 - Configuração do Sensor Ultrassônico S1 .....	42
Figura 77 - Configuração do Sensor Ultrassônico S2 .....	42
Figura 78 – 14ªTarefa – Programação .....	43
Figura 79 - Pista com a sinalização do obstáculo .....	44
Figura 80 – 15ªTarefa – Programação .....	44
Figura 81 – 16ªTarefa – Programação .....	45
Figura 82 - Pista com os locais para inserir os obstáculos.....	46
Figura 83 – 17ªTarefa – Programação .....	46

# SUMÁRIO

1) APRESENTAÇÃO .....	7
2) INTRODUÇÃO .....	7
3) APRESENTAÇÃO DO KIT LEGO MINDSTORMS NXT 2.0 .....	9
3.1) Bloco NXT .....	10
3.2) Servo Motores .....	11
3.3) Sensores .....	11
3.3.1) Sensor de luminosidade (Light Sensor Block) .....	12
3.3.2) Sensor de toque (Touch Sensor Block) .....	12
3.3.3) Sensor ultrassônico (Ultrasonic Sensor Block) .....	13
4) LAYOUT IDEAL PARA A REALIZAÇÃO DAS OFICINAS DE ROBÓTICA EDUCACIONAL .....	14
5) MONTAGEM DO CARRO .....	15
4.1) Montagem 01 .....	15
4.2) Montagem 02 .....	15
4.3) Montagem 03 .....	15
4.4) Montagem 04 .....	16
4.5) Montagem 05 .....	16
4.6) Montagem 06 .....	16
4.7) Montagem 07 .....	17
4.8) Montagem 08 .....	17
4.9) Montagem 09 .....	18
4.10) Montagem 10 .....	18
4.11) Montagem 11 .....	18
4.12) Montagem 12 .....	19
4.13) Montagem 13 .....	19
4.14) Montagem 14 .....	19
4.15) Montagem 15 .....	20
4.16) Montagem 16 .....	20
4.17) Montagem 17 .....	20
4.18) Montagem 18 .....	21
4.19) Montagem 19 .....	21

6)	INTRODUÇÃO AO SOFTWARE NXT-G .....	22
7)	TAREFAS A SEREM EXECUTADAS PELOS ALUNOS .....	25
7.1	- 1º TAREFA: FAZER O CARRO ANDAR PARA FRENTE.....	25
7.2	- 2º TAREFA: DAR RÉ.....	27
7.3	- 3º TAREFA: DIREITA E ESQUERDA.....	27
7.4	- 4º TAREFA: UTILIZANDO A TELA DO NXT .....	28
7.5	- 5º TAREFA: MOVER O CARRO UTILIZANDO O SENSOR TOUCH .....	29
7.6	- 6º TAREFA: COMANDO LOOP.....	30
7.7	- 7º TAREFA: APRENDENDO O COMANDO SWITCH (SE) .....	31
7.8	- 8º TAREFA: CONTROLAR O CARRO UTILIZANDO DOIS SENSORES DE TOQUE 32	
7.9	- 9º TAREFA: SEGUIR LINHA PRETA .....	34
7.10	- 10º TAREFA: IDENTIFICAÇÃO DE CORES E AÇÕES ESPECÍFICAS. ....	36
7.11	- 11º TAREFA: IDENTIFICAÇÃO DE CORES E AÇÕES ESPECÍFICAS. ....	39
7.12	- 12º TAREFA: DETECTANDO UM OBSTÁCULO .....	40
7.13	- 13º TAREFA: DETECTANDO UM OBSTÁCULO .....	41
7.14	- 14º TAREFA: DETECTANDO UM OBSTÁCULO E DESVIANDO.....	43
7.15	- 15º TAREFA: DETECTANDO UM OBSTÁCULO SOBRE A LINHA PRETA.....	43
7.16	- 16º TAREFA: DETECTANDO UM OBSTÁCULO SOBRE A LINHA PRETA E DESVIANDO .....	45
7.17	- 17º TAREFA: IDENTIFICAR E REMOVER OS OBJETOS DE DENTRO DA PISTA, SEM PODER SAIR DE DENTRO DO CIRCULO.....	45

# 1) APRESENTAÇÃO

Prezado (a) Professor (a)

O presente trabalho constitui o Produto Final obtido da dissertação de mestrado, que se intitula “Robótica Educacional dos Anos Finais do Ensino Fundamental: Um estudo de caso”, a qual foi desenvolvida durante o curso de Mestrado Profissional em Ciências e Tecnologias na educação, ofertado pelo Instituto Federal Sul-rio-grandense, Campus Pelotas Visconde da Graça, sob a orientação do Prof<sup>a</sup>. Dr. Marcos André Betemps Vaz da Silva.

A pesquisa teve seu embasamento teórico nos estudos sobre a robótica educacional, e buscou relacionar os conhecimentos obtidos através de um questionário antes e outro depois da aplicação de uma oficina prática de robótica. Esta metodologia teve o intuito de verificar e mensurar os efeitos de uma intervenção de curto prazo no processo de ensino e aprendizagem a partir desta relação.

Durante a pesquisa foram realizadas sete oficinas práticas de robótica, com duração de 8 horas e com média de 14 alunos por turma. Destas sete turmas, duas foram em escolas municipais, quatro em escolas estaduais e apenas uma escola particular na cidade de Camaquã-RS no ano de 2015. Em todas as turmas participaram somente alunos que estavam frequentando o último ano do ensino fundamental.

Nossa intenção com o presente produto é oferecer uma contribuição aos professores que desejam trabalhar com a robótica educacional em suas escolas. Assim, este material relata a nossa experiência com o uso desta ferramenta de trabalho e apresenta sugestões de atividades que envolvam o ensino dos princípios da robótica no âmbito escolar. A pesquisa mostrou que a abordagem de conteúdos diversos, em forma de oficinas, tornou a aprendizagem mais prazerosa.

Os resultados alcançados foram considerados significativos. Aos interessados em conhecer a pesquisa que originou este produto, a dissertação encontra-se disponível no site ([ppgcited.cavg.ifsul.edu.br](http://ppgcited.cavg.ifsul.edu.br)) do PPGCITED - Programa de Pós-Graduação em Ciências e Tecnologias na Educação.

## 2) INTRODUÇÃO

O avanço da tecnologia nas últimas décadas, propicia ao indivíduo um leque maior de possibilidades de construir e/ou aperfeiçoar seus conhecimentos. Estas ferramentas quando aplicadas de maneira correta, possibilitam ao educador uma gama maior de possibilidades para alcançar seus objetivos. Dentre estes objetivos, está a compreensão do conteúdo proposto naquela aula pelos alunos. Grande parte das escolas do Brasil já possuem um laboratório de informática com acesso à Internet, softwares educacionais e programas básicos.

De acordo com os Parâmetros Curriculares Nacionais (PCN), Brasil (1998, p. 11), é responsabilidade das escolas a ampliação da visão de conteúdo, ir além dos conceitos. Inserir procedimentos, atitudes e valores como algo tão relevantes e de tanta importância quanto os conceitos que já são abordados.

Durante o desenvolvimento deste projeto pretende-se ir além dos editores de texto, programas de edição de imagens e apresentações, planilhas de cálculo, etc. tentamos mostrar a aplicabilidade efetiva do uso da Robótica como aliada ao processo de aprendizagem na educação.

A robótica é considerada hoje a mola mestra de uma nova mutação dos meios de produção, isto devido a sua versatilidade, em oposição à automação fixa ou “hard”, atualmente dominante na indústria. Os robôs, graças ao seu sistema lógico ou informático, podem ser reprogramados e utilizados em uma grande variedade de tarefas. Mas, não é a reprogramação o fator mais importante na versatilidade desejada e sim a adaptação às variações no seu ambiente de trabalho, mediante um sistema adequado de percepção e tratamento de informação. (Ferreira, 1991, p.4)

A nova geração de alunos aprendizes, está inserida em um tempo onde a tecnologia sofre constantes mudanças. A quantidade, bem como a diversidade de informações recebidas, dentro e fora do ambiente escolar tem crescido de maneira assustadora. A escola tem se esforçado ao máximo para acompanhar estas mudanças, no que nem sempre tem sucesso. A utilização de tecnologias aplicadas com o uso do computador, direcionadas a um determinado tipo de conteúdo, quando aliadas a atividades prazerosas, tendem a gerar um impacto no processo ensino aprendizagem.

Assim, com o intuito de proporcionar a todos interessados, a reprodução dos resultados obtidos neste estudo, confeccionamos este tutorial detalhado para realizar as oficinas utilizando os Kits LEGO Mindstorms NXT 2.0. A utilização deste material em oficinas para alunos dos anos finais do ensino fundamental, tem por objetivo trazer atividades práticas que tracem paralelos entre diversas disciplinas, como por exemplo matemática e física, além de trabalhar com informática básica e programação.

Estas oficinas foram idealizadas para serem trabalhadas em 8 horas, divididas em dois turnos de 4 horas. Imagina-se que o ideal para a aplicação desta oficina, seria em duas manhãs ou duas tardes consecutivas, pois assim os alunos teriam um pequeno período para pensar e raciocinar sobre a dinâmica envolvida no processo.

Com o uso dos kits LEGO Mindstorms NXT, os alunos poderão criar experimentos onde será posto em prática a teoria das salas de aulas, aplicada a projetos. Além disso, os alunos terão a chance de se aproximar de certos assuntos, que à primeira vista podem parecer distantes e complicados, mas que com o incentivo certo, se tornam enormes fontes de aprendizado e diversão. Estamos falando de Robótica e Programação.

Nesse tutorial estão contidas informações sobre os componentes do kit LEGO Mindstorms NXT 2.0 e algumas funções, além de um tutorial básico sobre a programação envolvida na criação dos modelos, com tarefas e a resolução das mesmas. A cada nova tarefa a dificuldade aumenta, e exige que o aluno reúna todos os conhecimentos adquiridos ao longo da oficina e aplique através do programa para que o robô execute e complete o desafio.

Cabe ressaltar também que as imagens referentes as montagens, símbolos, configurações e programação, são todas retiradas do próprio Software da LEGO.



### 3) APRESENTAÇÃO DO KIT LEGO MINDSTORMS NXT 2.0

O kit de Robótica Educacional que utilizaremos nas oficinas é o kit 8547 da marca LEGO MINDSTORMS Education NXT 2.0. Diferentemente dos kits de brinquedos facilmente encontrado em lojas de crianças, esse kit é especialmente desenvolvido para fins educacionais, possuindo um microcomputador (considerado o cérebro do robô), servo-motores e sensores diversos, além das tradicionais peças de montagem rápida e fácil. Contabilizando todo conjunto, o kit é composto por 619 peças, assim possibilitando uma gama de combinações, montagens e programações.



Fonte: [http://www.ingeniu.com/ecommerce/control/product/~product\\_id=NXT2CM334;jsessionid=diaqm1l2k726](http://www.ingeniu.com/ecommerce/control/product/~product_id=NXT2CM334;jsessionid=diaqm1l2k726)

Figura 1 - Imagem de embalagem do Kit LEGO Mindstorms NXT 2.0

Com relação a programação dos kits, a empresa fornece um software oficial da linha LEGO MINDSTORMS, que utiliza uma linguagem NXT-G, que é do tipo “programação em blocos”, isto é, possui um ambiente interativo que possibilita escolher um bloco, arrastar, colar e configurar os comandos para fazer a cadeia de programação.

Descrição detalhada do Kit LEGO Mindstorms NXT 2.0:

- 01 Bloco lógico programável LEGO NXT 2.0;
- 01 Sensor de core;
- 02 Sensores de Toque;
- 01 Sensor de Ultrassom;
- 03 Servo-motores com encoder's;
- 07 Cabos conectores (Padrão RJ-12 polarizado à direita);
- Peças LEGO, como: Blocos de tamanhos diversos;
- Elementos estruturais, como vigas (em L, retas e angulares) e pranchas;
- Engrenagens (retas, cônicas e mistas, sistema “coroa-pinhão”);
- Correias, buchas e polias; Rodas, pneus e eixos de diversos tamanhos;
- Conectores, conectores com inversão e conectores em ângulos;
- Bateria recarregável de Íon-Lítio 1400mA;

### 3.1) Bloco NXT



Fonte: <http://hackeducation.com/2015/04/10/mindstorms>

Figura 2 - NXT Intelligent Brick

O NXT é um bloco programável de 32 bits, onde ocorre todo trânsito de informações provenientes da programação realizada pelo usuário, fazendo o processamento dos dados oriundo dos sensores e respondendo com os atuadores (motores). Neste bloco, a comunicação entre o software e o microcontrolador, pode ser realizada sem fio, via Bluetooth, ou utilizando um cabo e uma porta USB do computador. Também possui um display com matriz de ponto programável onde é possível visualizar os programas, calibrar sensores, nível de baterias, entre outras funções. Por essas razões, ele recebe o apelido de cérebro.

Na estrutura do bloco, encontram-se três portas que mandam sinais para os motores (Portas A, B, C) e quatro portas que recebem sinais dos sensores (Portas 1, 2, 3, 4). Além disso, na sua IHM (Interface Homem-Máquina), possui quatro botões e um sistema de alto falante do lado esquerdo, para reproduzir sons. O botão central liga e escolhe as opções no cérebro. O botão logo abaixo do central (cinza escuro), retorna a tela anterior. Todos os botões podem ser usados para controlar as ações do robô, mas é preciso definir seus usos na programação.



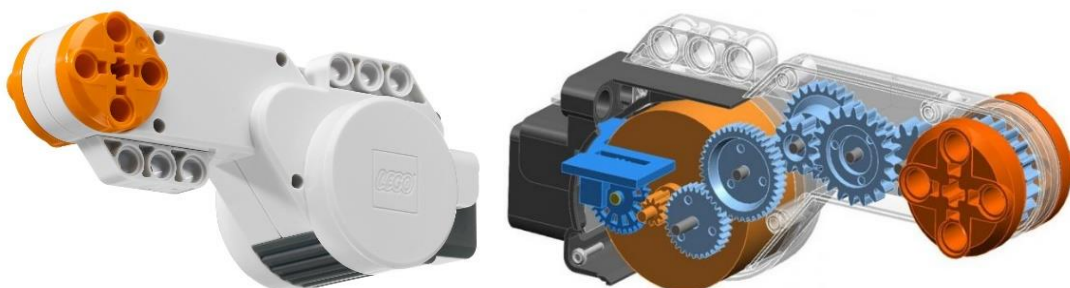
Fonte: [http://www.ingeniu.com/ecommerce/control/product/~product\\_id=NXT2CM334;jsessionid=diaqm1l2k726](http://www.ingeniu.com/ecommerce/control/product/~product_id=NXT2CM334;jsessionid=diaqm1l2k726)

Figura 3 - NXT Intelligent Brick com sensores e atuadores



Figura 4 - NXT Intelligent Brick

### 3.2) Servo Motores



Fonte: <http://www.legoengineering.com/nxt-sensors/>

Figura 5 - Representação externa e interna do servo motor

Os servo-motores são pequenos motores que realizam todas as tarefas mecânicas do seu robô. Como estes motores possuem encoder (sensor que converte um movimento angular em pulsos digitais), isso permite ao motor controlar a precisão de movimento em torno de um grau radiano. Contudo, dentro dos ajustes do motor, é possível configurar a velocidade, duração de ciclo, grau de rotação, número de rotações ou por tempo (medidos em segundos).

### 3.3) Sensores

Podemos definir a palavra sensor como “aquilo que sente”. Na eletrônica, um sensor é conhecido como qualquer componente ou circuito eletrônico que permita a análise de uma determinada condição do ambiente. Estes componentes podem fazer

a leitura de determinadas características, tais como a presença de um obstáculo no caminho de um robô, a velocidade do eixo de um motor ou o fato de uma porta estar fechada ou não, e responder de acordo com elas, ou seja, criar um sistema capaz de interagir com o ambiente (PATSKO 2006).

Os sensores utilizados no Kit (sensor de cor, de toque e ultrassônico) tem como finalidade proporcionar aos educandos muitas de possibilidades, sendo capazes de recolher informações do local onde o robô está. Essas informações são convertidas em dados interpretados pelo bloco NXT, que reage de acordo com a programação descrita no código.

### 3.3.1) Sensor de luminosidade (Light Sensor Block)



Fonte: <http://www.legoengineering.com/nxt-sensors/>

Figura 6 - Sensor de luminosidade

O sensor de luminosidade é capaz de captar diversos níveis de luminosidade, seja de luz ambiente, ou luz refletida em alguma superfície. Ele também pode emitir luz, através de um pequeno LED na ponta do sensor. O Sensor de Cor pode detectar seis cores: Vermelho, azul, verde, amarelo, branco e preto, enquanto a luz emitida pelo LED pode ser vermelha, azul ou verde.

### 3.3.2) Sensor de toque (Touch Sensor Block)



Fonte: <http://www.legoengineering.com/nxt-sensors/>

Figura 7 - Sensor de toque

Os sensores de toque simulam um interruptor ou botão sem trava, isso que dizer que quando é pressionado o botão, ele faz o contato e quando cessa a pressão, o botão retorna a posição inicial, abrindo o contato. Servem para criar a sensação de tato para o seu robô, para que ele possa perceber e reagir ao ambiente a sua volta através do toque. Os botões podem executar comandos quando apertados, pressionados, ou levemente apertados.

### 3.3.3) Sensor ultrassônico (Ultrasonic Sensor Block)



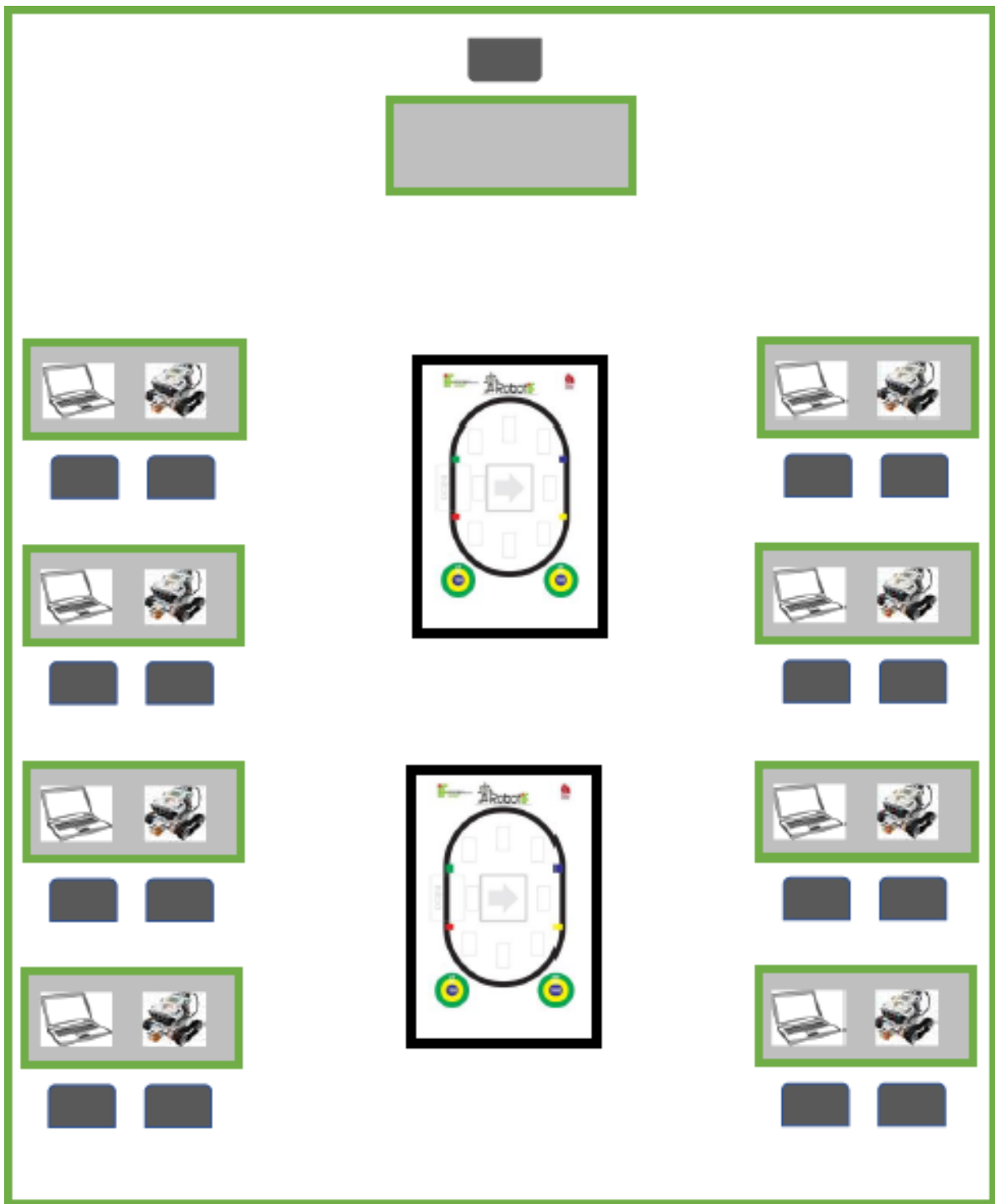
Fonte: <http://www.legoengineering.com/nxt-sensors/>

*Figura 8 - Sensor ultrassônico*

O sensor ultrassônico tem a capacidade de emitir uma onda sonora que, ao encontrar um obstáculo, rebaterá de volta em direção ao módulo sonar, sendo que neste tempo de emissão e recebimento do sinal, o programa consegue calcular a distância do robô até o objeto.

Tendo apresentado a estrutura básica do Kit LEGO, vamos apresentar a seguir a metodologia que será desenvolvida nas atividades com os estudantes dos anos finais da educação Básica da cidade de Camaquã e região.

#### 4) LAYOUT IDEAL PARA A REALIZAÇÃO DAS OFICINAS DE ROBÓTICA EDUCACIONAL





## 5) MONTAGEM DO CARRO

### 4.1) Montagem 01

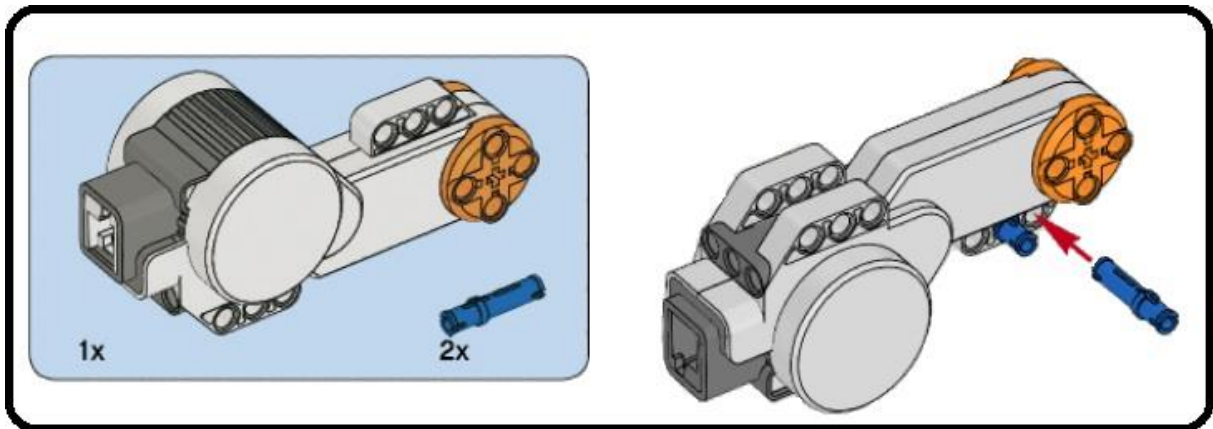


Figura 9 - Montagem 01 – Fonte: Software LEGO Mindstorms NXT

### 4.2) Montagem 02

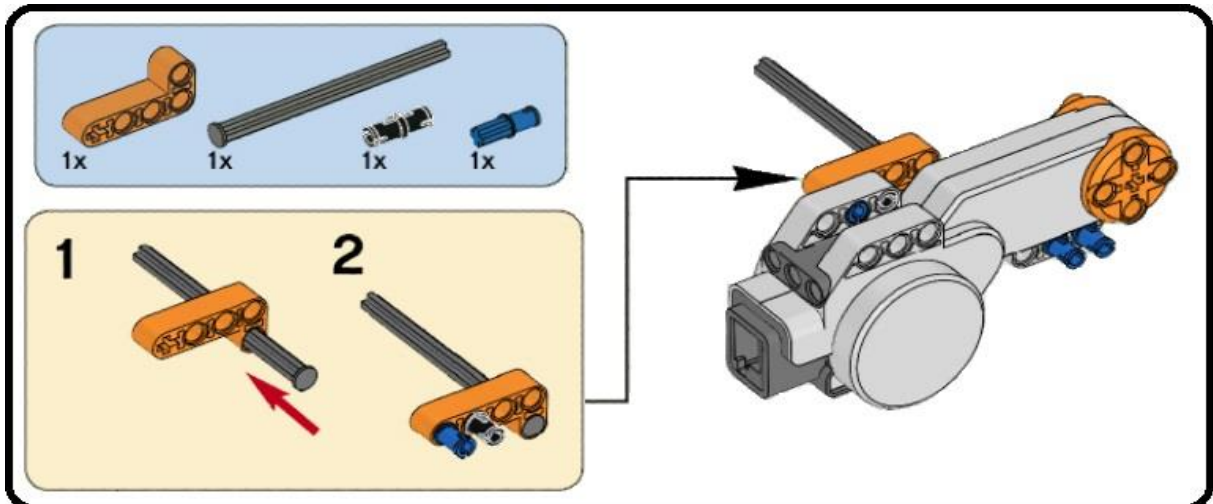


Figura 10 - Montagem 02 – Fonte: Software LEGO Mindstorms NXT

### 4.3) Montagem 03

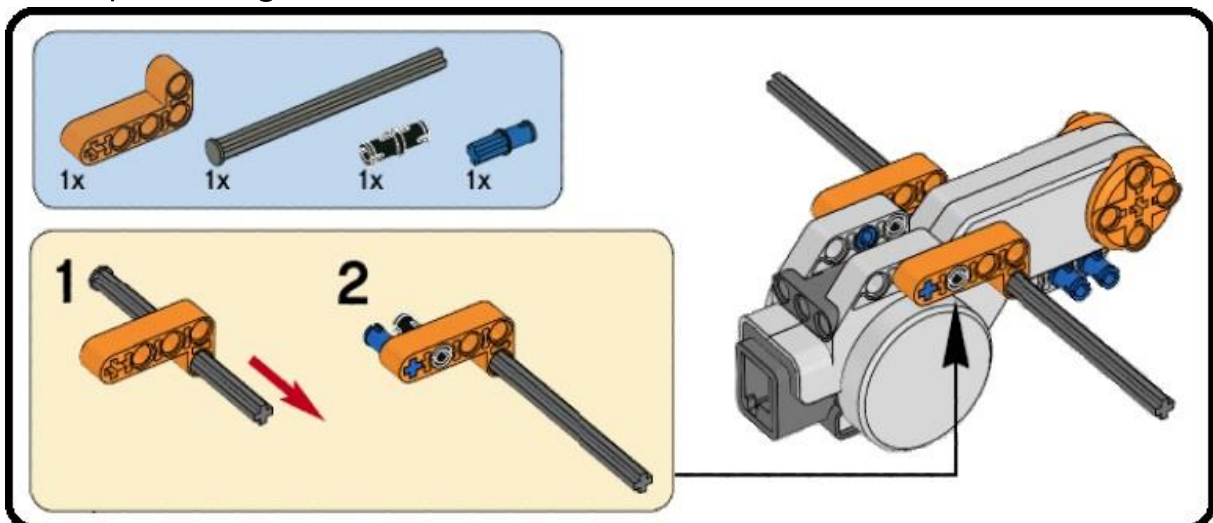


Figura 11 - Montagem 03 – Fonte: Software LEGO Mindstorms NXT

#### 4.4) Montagem 04

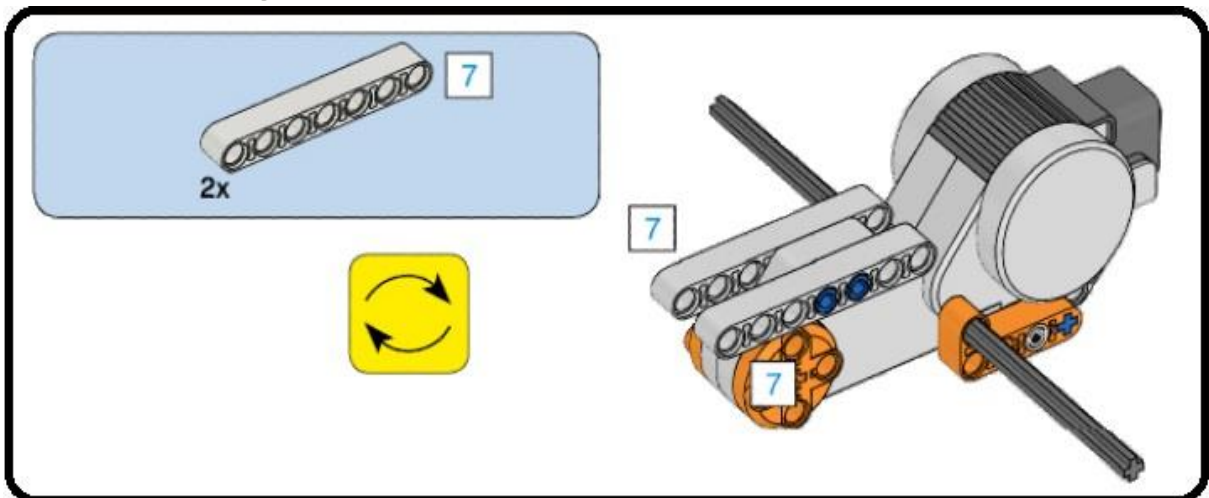


Figura 12 - Montagem 04 – Fonte: Software LEGO Mindstorms NXT

#### 4.5) Montagem 05

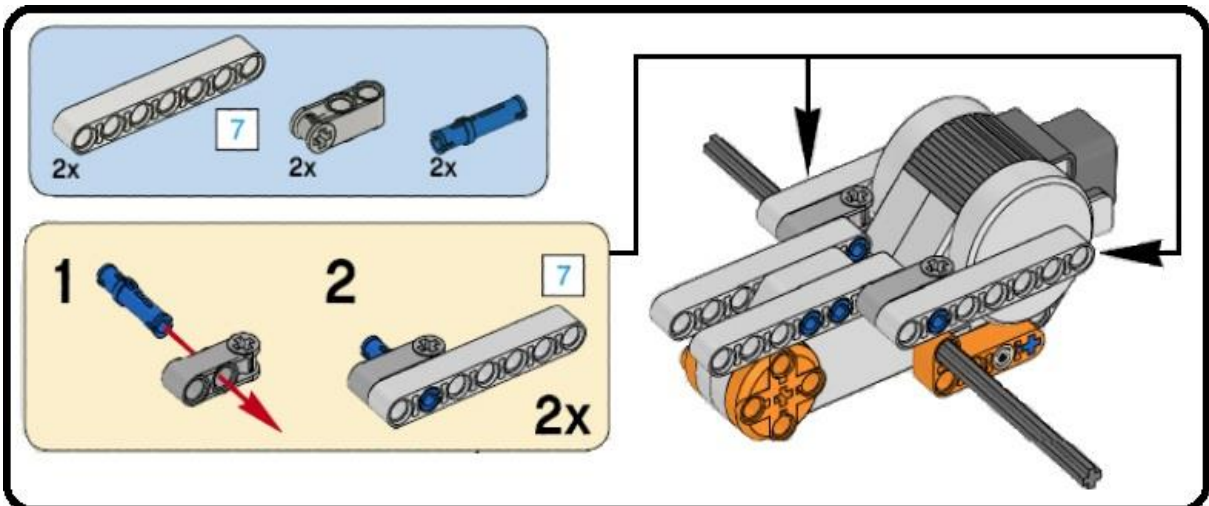


Figura 13 - Montagem 05 – Fonte: Software LEGO Mindstorms NXT

#### 4.6) Montagem 06

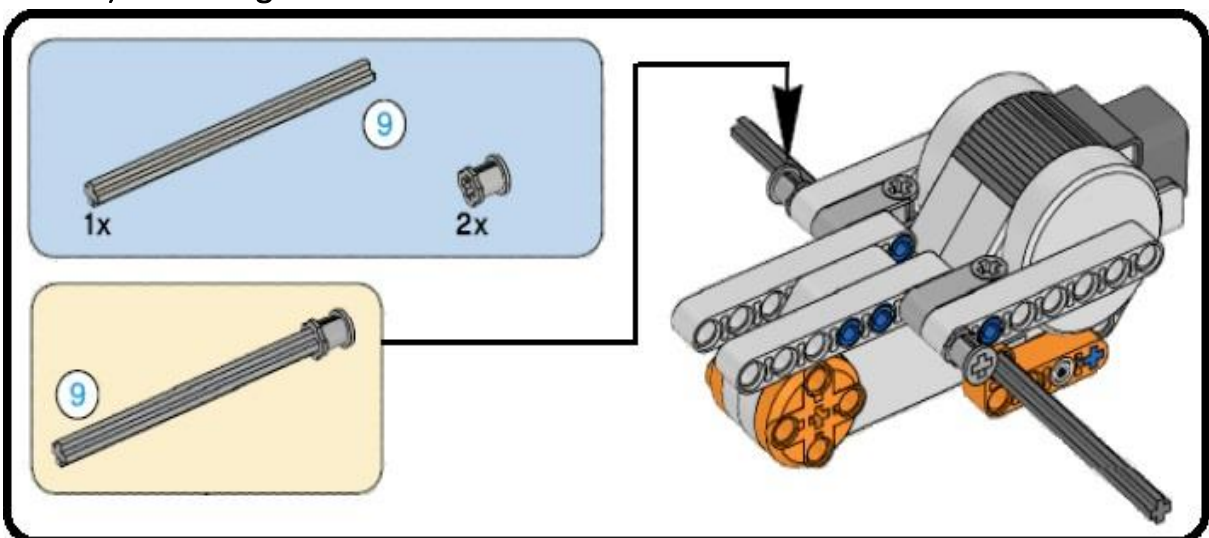


Figura 14 - Montagem 06 – Fonte: Software LEGO Mindstorms NXT



#### 4.7) Montagem 07

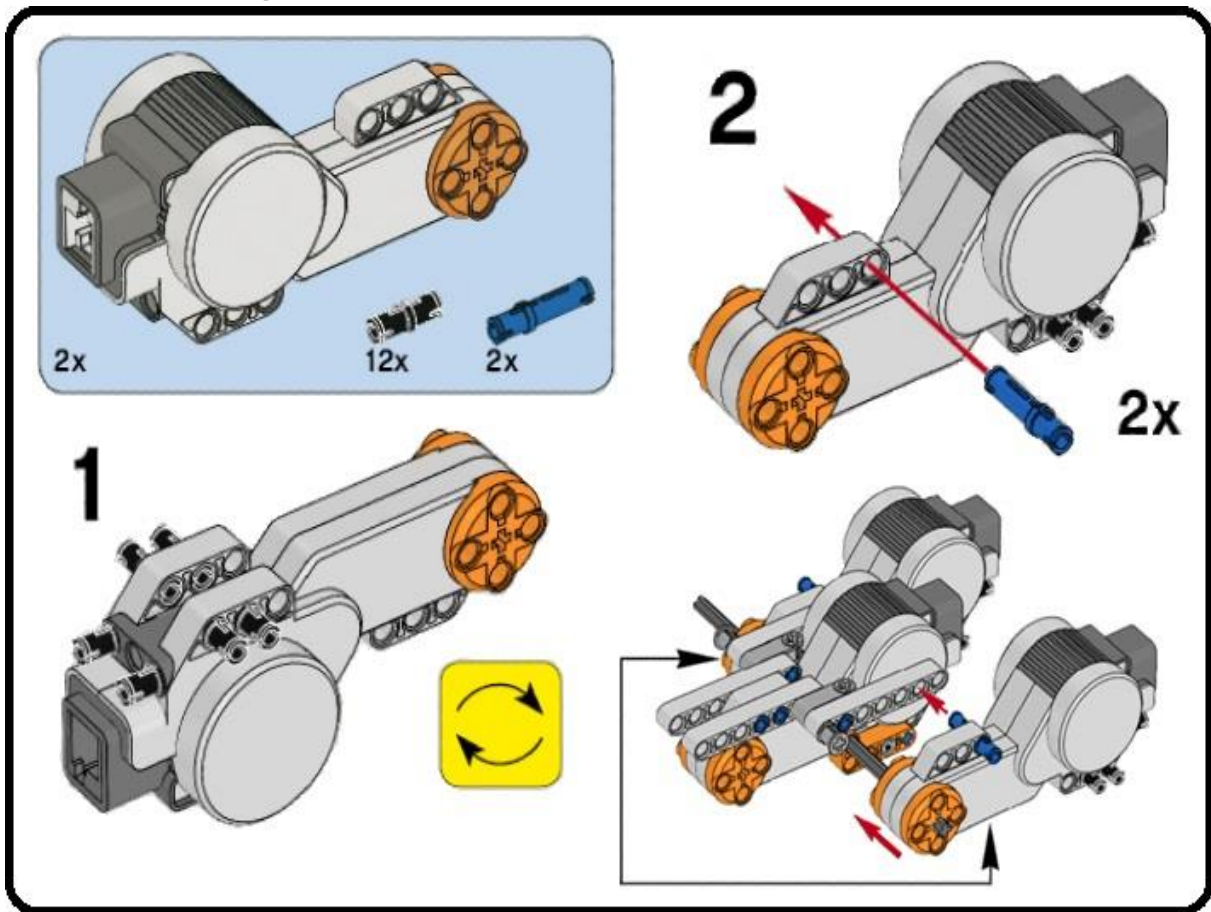


Figura 15 - Montagem 07 – Fonte: Software LEGO Mindstorms NXT

#### 4.8) Montagem 08

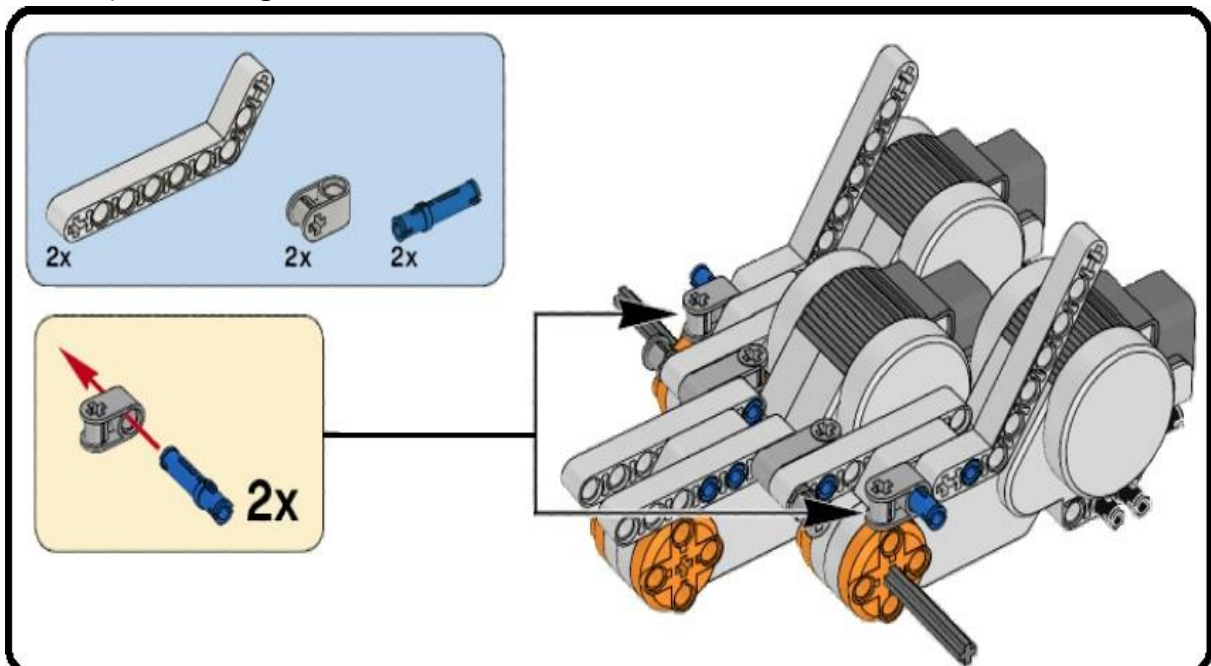


Figura 16 - Montagem 08 – Fonte: Software LEGO Mindstorms NXT

#### 4.9) Montagem 09

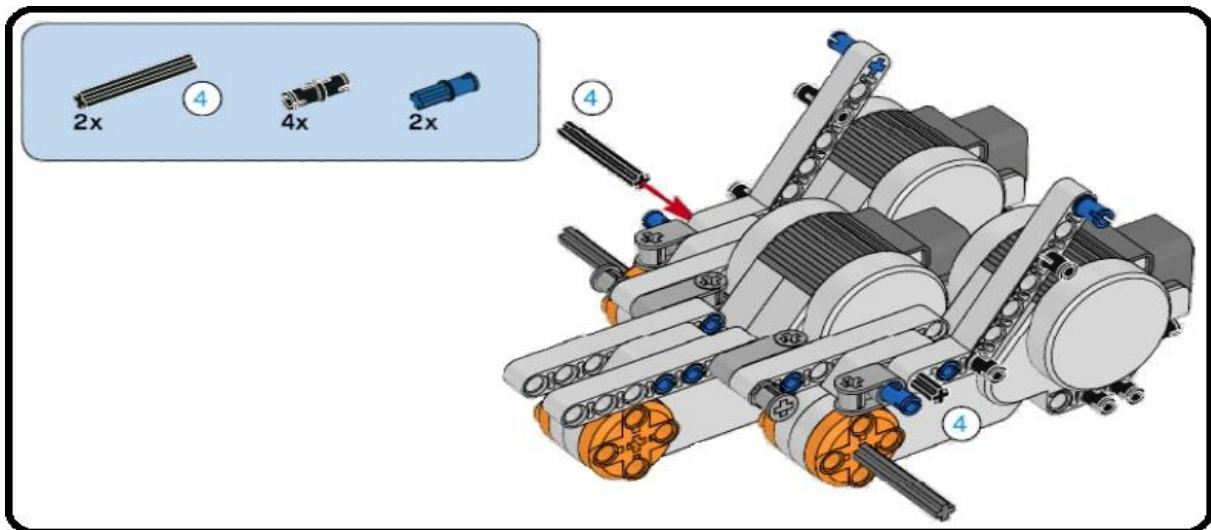


Figura 17 - Montagem 09 – Fonte: Software LEGO Mindstorms NXT

#### 4.10) Montagem 10

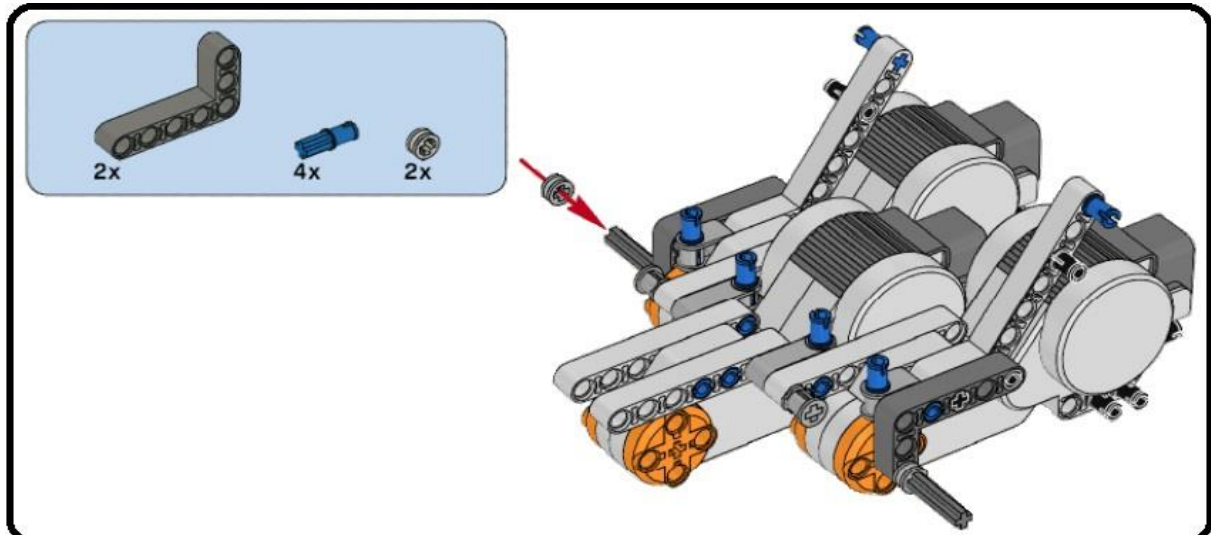


Figura 18 - Montagem 10 – Fonte: Software LEGO Mindstorms NXT

#### 4.11) Montagem 11

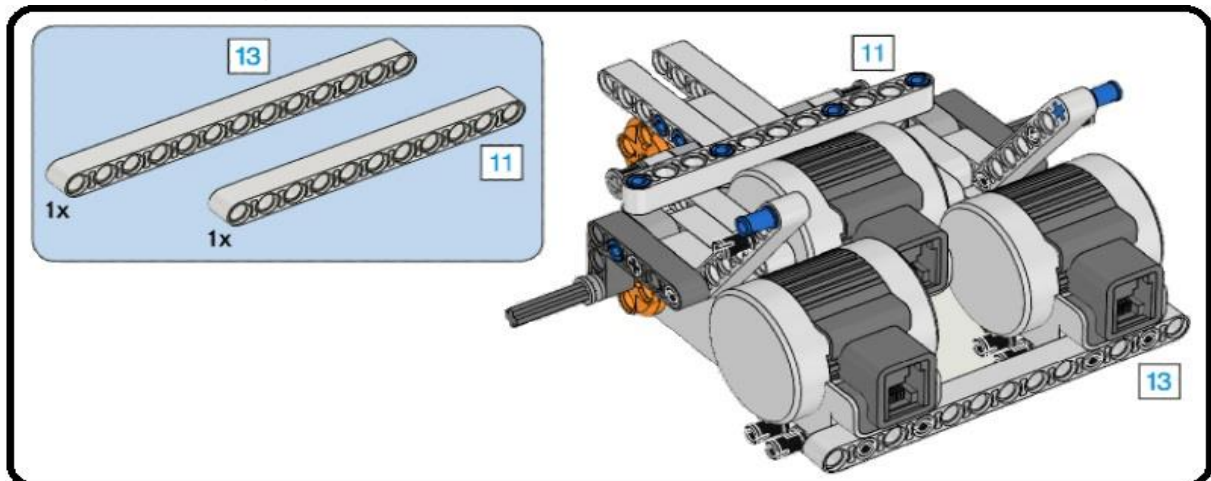


Figura 19 - Montagem 11 – Fonte: Software LEGO Mindstorms NXT

#### 4.12) Montagem 12

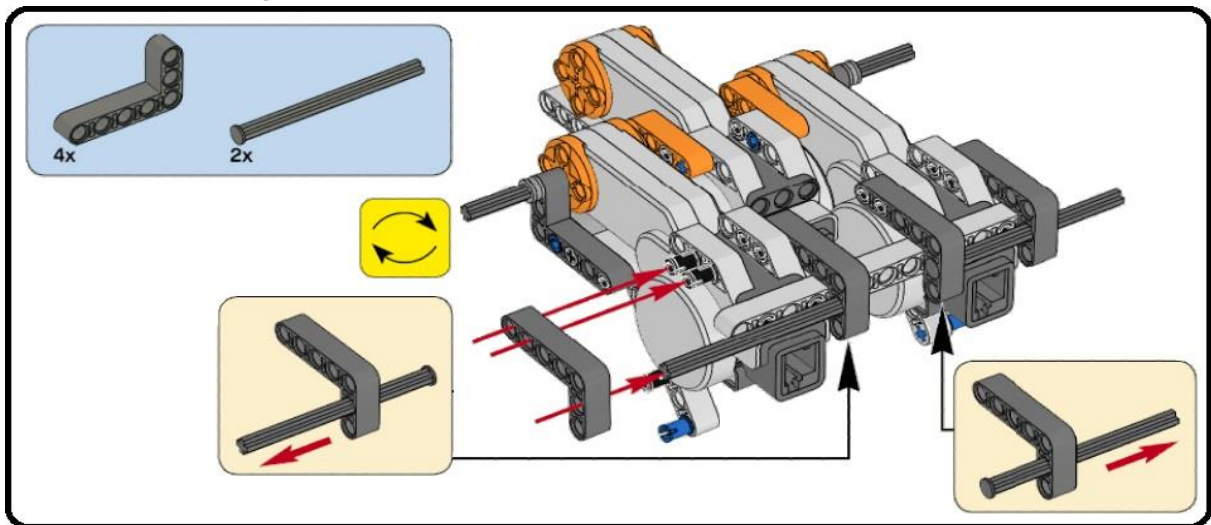


Figura 20 - Montagem 12 – Fonte: Software LEGO Mindstorms NXT

#### 4.13) Montagem 13

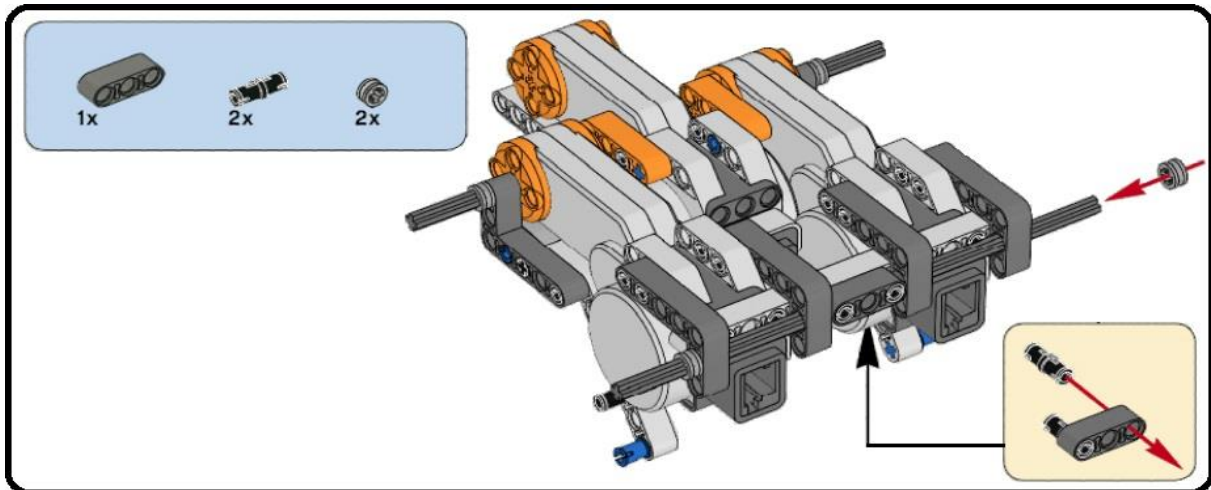


Figura 21 - Montagem 13 – Fonte: Software LEGO Mindstorms NXT

#### 4.14) Montagem 14

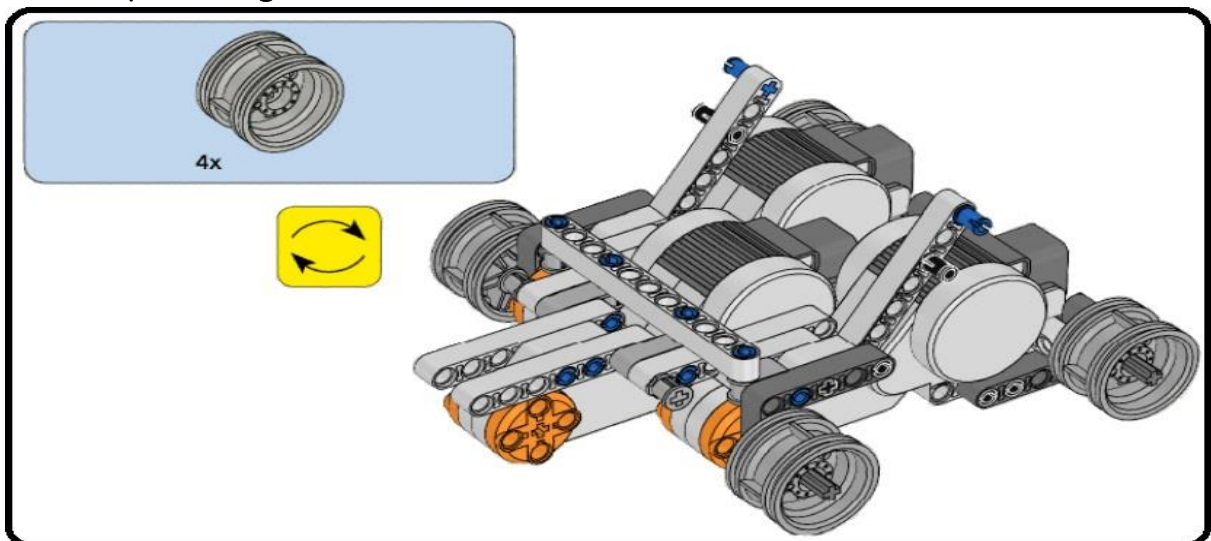


Figura 22 - Montagem 14 – Fonte: Software LEGO Mindstorms NXT



#### 4.15) Montagem 15

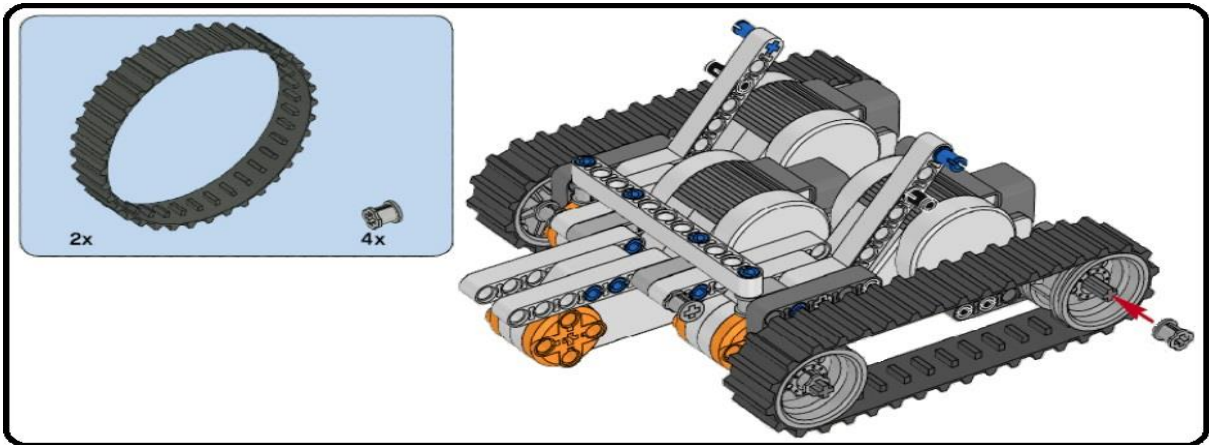


Figura 23 - Montagem 15 – Fonte: Software LEGO Mindstorms NXT

#### 4.16) Montagem 16

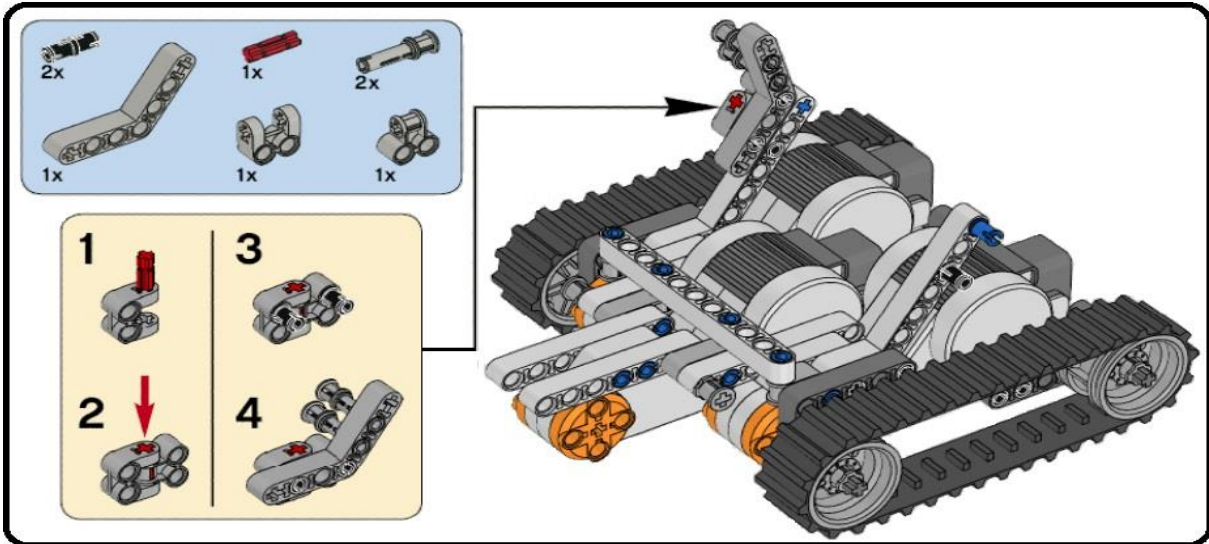


Figura 24 - Montagem 16 – Fonte: Software LEGO Mindstorms NXT

#### 4.17) Montagem 17

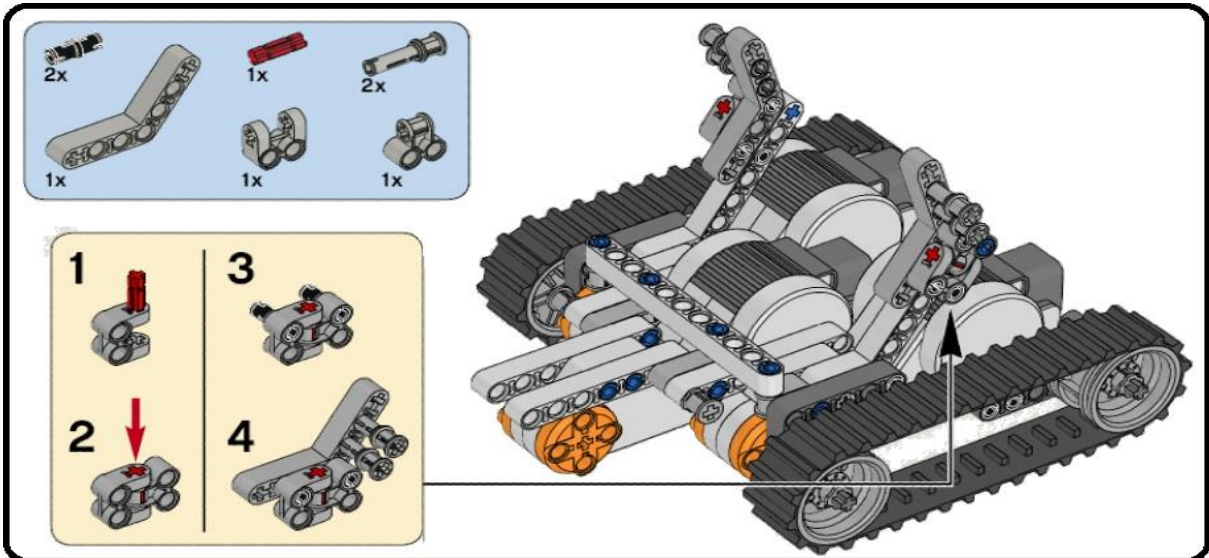


Figura 25 - Montagem 17 – Fonte: Software LEGO Mindstorms NXT

#### 4.18) Montagem 18

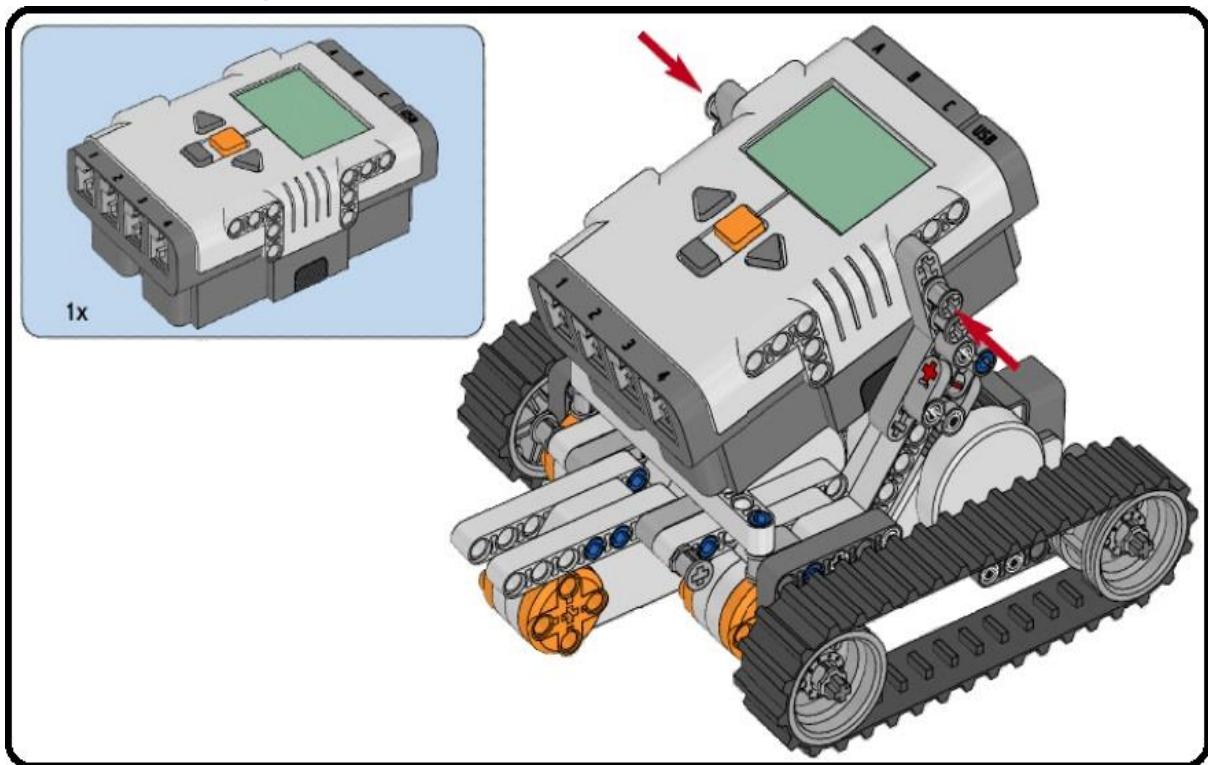


Figura 26 - Montagem 18 – Fonte: Software LEGO Mindstorms NXT

#### 4.19) Montagem 19

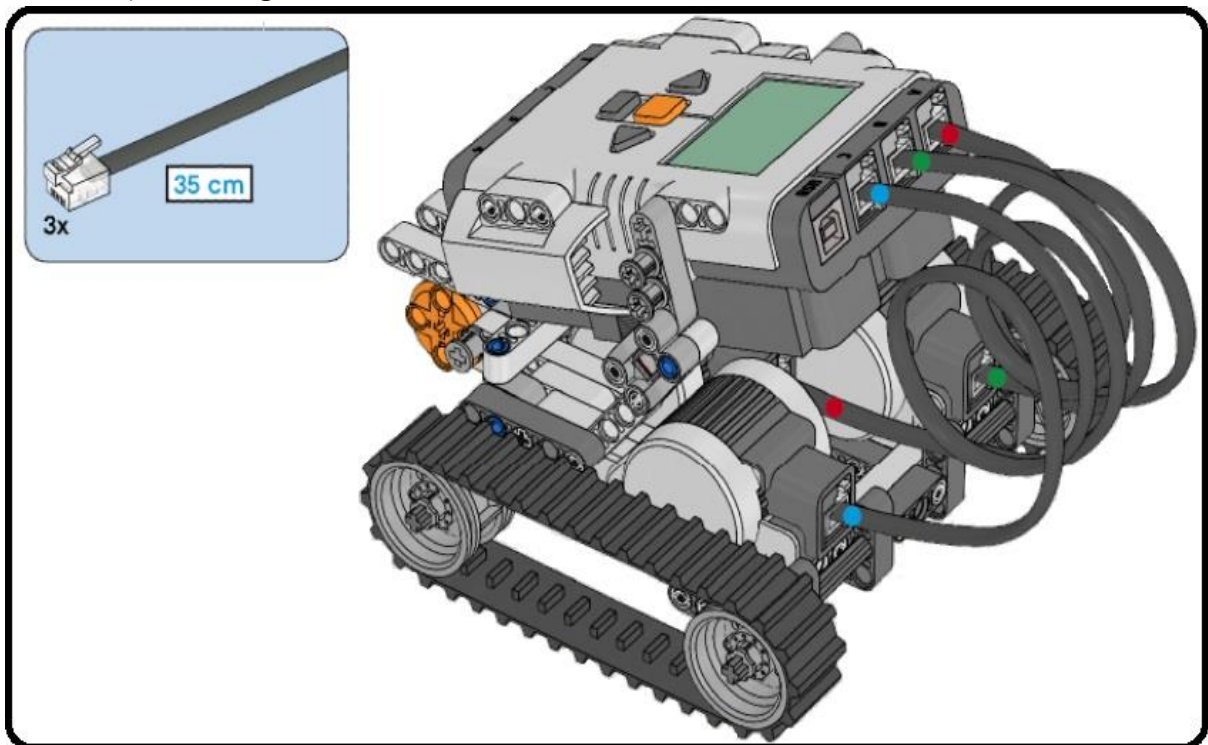


Figura 27 - Montagem 19 – Fonte: Software LEGO Mindstorms NXT

Obs: Na Figura 27 - Montagem 19, o cabo sinalizado na cor vermelha, NÃO precisa ser conectado, pois em todas as atividades, nenhuma utiliza o motor central

## 6) INTRODUÇÃO AO SOFTWARE NXT-G

O LEGO Mindstorms NXT possui um software próprio para a programação dos robôs. A seguir será apresentado como funciona este programa.

Abrindo o software, aparecerá a seguinte tela inicial (Figura 28 - Tela inicial do Software):

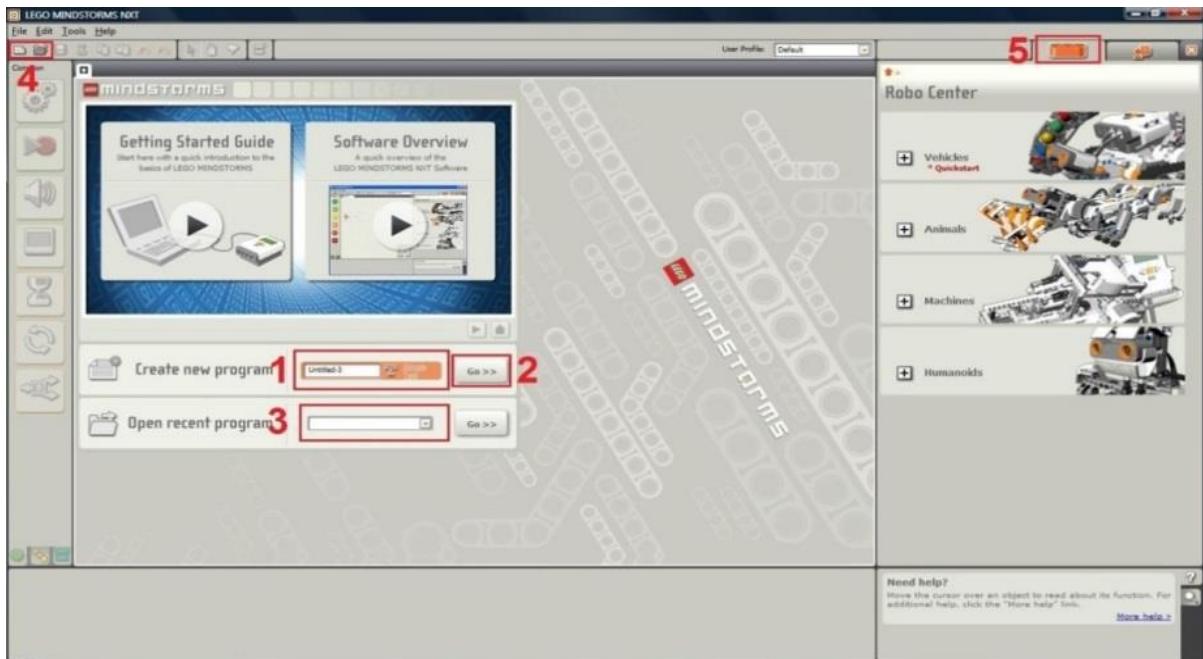


Figura 28 - Tela inicial do Software – Fonte: Software LEGO Mindstorms NXT

O item assinalado em vermelho com o número 1, localizado no centro da tela, possui um espaço onde pode-se escrever o nome do programa que será criado. Após nomear o seu programa, clique na opção “Go >>” (item 2) e será aberto a tela de programação (Figura 29 - Tela de início da programação).

Logo abaixo, no item 3, há a opção que permite escolher e abrir os programas que já foram criados e salvos no computador. Estas opções possuem atalhos localizados no quarto item. A aba localizada no item 5 possui montagens de diversos robôs assim como as instruções para montar o mesmo robô que já foi apresentado neste tutorial.

Após criar um programa com o nome desejado aparecerá a imagem abaixo (Figura 29 - Tela de início da programação):



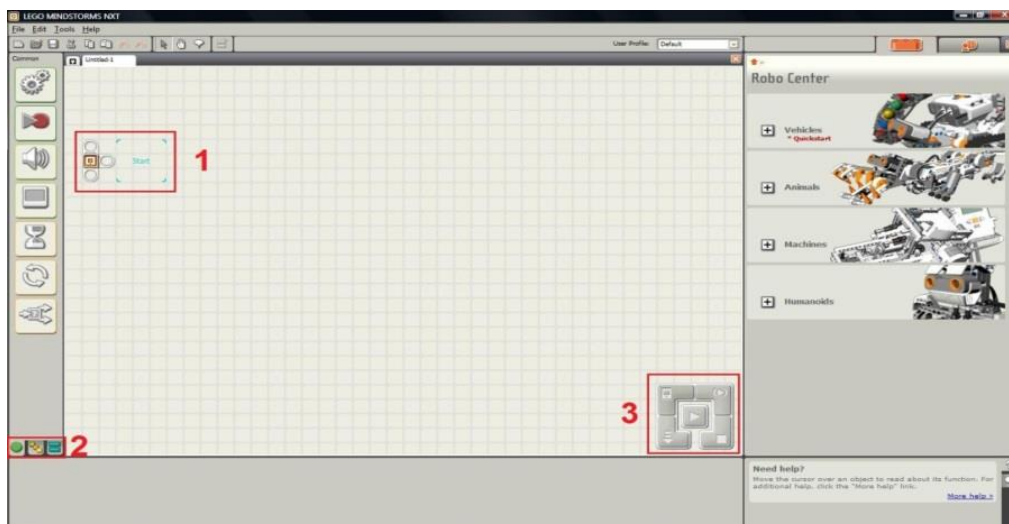


Figura 29 - Tela de início da programação – Fonte: Software LEGO Mindstorms NXT

Nesta tela inicial do software, na parte central, onde aparece uma área quadriculada, é o local onde ficarão dispostos os blocos das funções conforme o objetivo proposto. A programação inicia a partir da região assinalada em vermelho do item 1.

Logo abaixo, no item 2, há a Paleta de Comandos. A partir dela podemos escolher os comandos que utilizaremos para a programação. Na primeira aba, possuímos apenas os comandos mais simples. Já na segunda aba, possuímos tanto os simples como os avançados e na terceira aba, possuímos os comandos customizados que podem ser desenvolvidos. No exemplo acima (Figura 29 - Tela de início da programação) está selecionado a aba contendo os comandos simples.

No item assinalado com o número 3, apresenta 5 botões e suas funções diferenciadas. As sequencia será detalhada cada uma delas.



Figura 30 - Funções de comunicação do NXT

**Função 1 (NXT window):** Configuração de comunicação e acesso à memória interna do cérebro;

**Função 2 (Download):** Transfere o programa para o cérebro;

**Função 3 (Download and run):** Transfere o programa para o cérebro e iniciando-o logo após a transferência;

**Função 4 (Stop):** Interrompe a execução do programa;

**Função 5 (Download and run selected):** Transfere partes do programa para o cérebro.

A programação do LEGO Mindstorms NXT é feita através de blocos. Ela possui este formato com a intenção de ser didática, pois assim torna-se mais fácil a

aprendizagem e a resolução de problemas. Mesmo para os usuários que nunca tiveram contato com algum tipo de programação.

Os comandos são executados um após o outro na programação, tendo seu começo no segmento de blocos do primeiro item da tela inicial (Figura 29 - Tela de início da programação). Veja um exemplo a seguir (Figura 31 - Exemplo de programação).

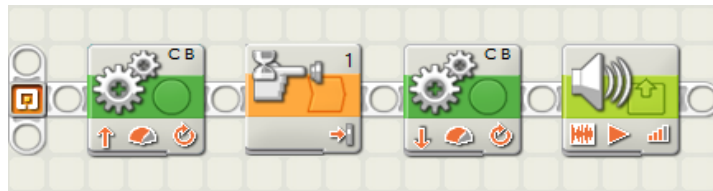


Figura 31 - Exemplo de programação

No exemplo mostrado, a programação está dizendo para que os motores do robô girem para frente um determinado número de vezes e então esperar até o sensor de toque ser acionado. Após o sensor mandar o sinal, os motores devem girar em sentido contrário um determinado número de vezes e após isso, o robô deve emitir um som. Depois de as ações serem executadas, a programação acaba.

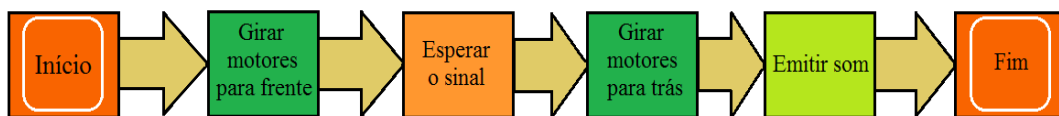


Figura 32 - Fluxograma da programação

Como visto, os blocos de comandos só podem ser executados singularmente e na ordem em que estão colocados nos segmentos de blocos. Os blocos mais importantes serão apresentados e explicados neste tutorial conforme a necessidade de seus usos nas tarefas a seguir.



## 7) TAREFAS A SEREM EXECUTADAS PELOS ALUNOS

### 7.1 - 1ª TAREFA: FAZER O CARRO ANDAR PARA FRENTE

Na primeira tarefa, é preciso fazer o robô andar para frente por 5 segundos. Agora será ensinado como programar o robô para fazer esta tarefa. Para isto, será preciso inserir o bloco chamado “Mover”. Ele possui o ícone mostrado abaixo (Figura 33 - Bloco Mover (Motor)).

Bloco *Mover*



Figura 33 - Bloco Mover (Motor)

O bloco de comando “Mover” permite acionar um ou mais motores ao mesmo tempo, no mesmo sentido, sentidos opostos ou com velocidades diferentes. Como mostrado anteriormente, a programação tem início, meio e fim. O programa deve iniciar dizendo para o robô andar em linha reta por 5 segundos e após isso finalizar a programação.

Primeiramente é preciso arrastar o ícone do comando *Mover* para o segmento de blocos. O programa deve ficar como mostra a imagem a seguir (Figura 34 – 1ªTarefa - Programação).



Figura 34 – 1ªTarefa - Programação

Deste modo o robô irá para frente após a inicialização do programa e, depois de realizado o comando, irá acabar o programa. É preciso ainda configurar o comando, pois o robô precisa andar exatos 5 segundos.

Sempre quando você clicar em um bloco, suas configurações aparecerão em um espaço na parte inferior da tela como o mostrado abaixo (Figura 35 - Configuração do bloco Mover). Você pode mudar a velocidade, direção e duração dos motores nas configurações deste comando.



Figura 35 - Configuração do bloco Mover

Como este bloco é um dos mais importantes para a movimentação dos robôs, vamos explicar detalhadamente todos os parâmetros possíveis para a configuração do bloco mover (Figura 33 - Bloco Mover (Motor)):

**Port:** *Seleciona qual ou quais as portas de saída que estão sendo utilizadas. É possível selecionar uma, duas ou as três portas ao mesmo tempo. No exemplo da Figura 35 - Configuração do bloco Mover, estão sendo selecionadas as portas B e C.*

**Direction:** *Seleciona qual a direção que o(s) motor(es) irá(ão) funcionar. É possível selecionar para frente, para trás e/ou mandá-lo parar.*

**Steering:** *No caso de selecionar dois motores ao mesmo tempo, como no exemplo da Figura 35 - Configuração do bloco Mover, é possível configurar que os mesmos não se comportem da mesma maneira, podendo colocar mais potência em um determinado motor e, conseqüentemente, menos potência no outro.*

**Power:** *Seleciona qual a porcentagem de alimentação que o motor receberá quando for acionado. Na barra de ajuste, é possível pegar e arrastar com o mouse ou apenas digitar no espaço onde aparece a porcentagem.*

**Duration:** *Seleciona a duração do ciclo que o motor irá desempenhar quando for acionado. É possível ajustar o mesmo como ciclo ilimitado, por grau de rotação, por rotação e por tempo (segundos).*

**Next Action:** *Seleciona a ação seguinte ao término do ciclo. É possível ajustar para que ele termine a ação ou continue indefinidamente.*

Depois de ter detalhado todas as funções possíveis, configure o bloco conforme a figura abaixo (Figura 36 - Configuração do bloco Mover para a 1ª tarefa):



Figura 36 - Configuração do bloco Mover para a 1ª tarefa

Note que na opção *Port* ficou selecionado as portas B e C, justamente as portas onde estão conectados os motores das esteiras da direita e da esquerda. Para uma melhor identificação das esteiras e das portas na hora da configuração, definiremos que o motor do lado direito está conectado na porta B e o motor do lado esquerdo está conectado na porta C. Sempre é importante conferir se os motores ou sensores estão conectados nas devidas portas configuradas no programa.

Com isso a programação já está pronta, mas ainda precisa ser enviada para o robô através do cabo USB já detalhada nas funções da Figura 30 - Funções de comunicação do NXT. Nessa parte, aconselhamos utilizar a função 2, pois se utilizar a função 3, quando terminar de carregar o programa no cérebro, automaticamente o carrinho sairia funcionando, mesmo estando conectado ao cabo USB. Assim, caso tenha realizado o carregamento do programa pela função 2, é preciso achar o arquivo no display do cérebro e clicar no botão laranja. Acionando este botão dar-se-á início a execução da programação.

## 7.2 - 2ª TAREFA: DAR RÉ

Agora o robô, além de precisar andar 5 segundos para frente, ele precisa dar ré por 3 rotações. Já que no programa anterior, o comando *Mover* já está dizendo para o robô andar 5 segundos para frente, basta adicionar outro bloco *Mover*.

Neste bloco novo, deve-se mudar sua direção e duração. Para mudar a direção dos motores, há a opção *Direction* que, além de mudar a direção, você também pode parar os motores.

A programação ficará desta maneira:

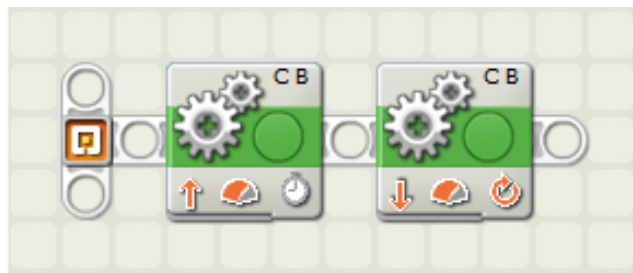


Figura 37 – 2ª Tarefa - Programação



Figura 38 - Configuração do segundo bloco Mover para a 2ª tarefa

Conecte o cabo USB e envie a programação.

## 7.3 - 3ª TAREFA: DIREITA E ESQUERDA

Na terceira tarefa, o robô precisa ir para frente e fazer uma curva para direita por 5 rotações. Depois, seguir indo para frente e fazer uma curva para esquerda por 3 segundos.

Nas configurações do comando *Mover*, há a opção *Steering*. Ela funciona apenas quando há dois motores selecionados, pois ela aumenta a potência de um motor e, conseqüentemente, diminui a potência de outro, no caso deste robô, fazendo uma curva. Basta mover o cursor para escolher o nível da curva.

A programação deve ficar assim:

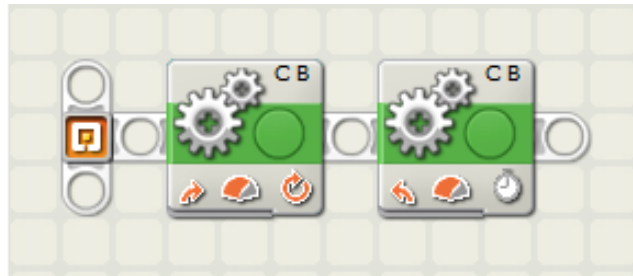


Figura 39 – 3ª Tarefa - Programação



Figura 40 - Configuração do primeiro bloco Mover para a 3ª tarefa



Figura 41 - Configuração do segundo bloco Mover para a 3ª tarefa

Conecte o cabo USB e envie a programação.

## 7.4 - 4ª TAREFA: UTILIZANDO A TELA DO NXT

Na quarta tarefa, o carro deve comportar-se exatamente igual a tarefa Nº3, mas depois que o carro terminar o segundo comando, deve emitir um som e mostrar uma figura na tela do bloco NXT.

Obs: O Som e a Figura ficam a escolha dos alunos.

### Bloco de Sound (Som)



Figura 42 - Bloco Sound (Som)



Figura 43 - Configuração do bloco Sound

### Bloco *Display* (Tela)



Figura 44 - Bloco *Display* (Tela)

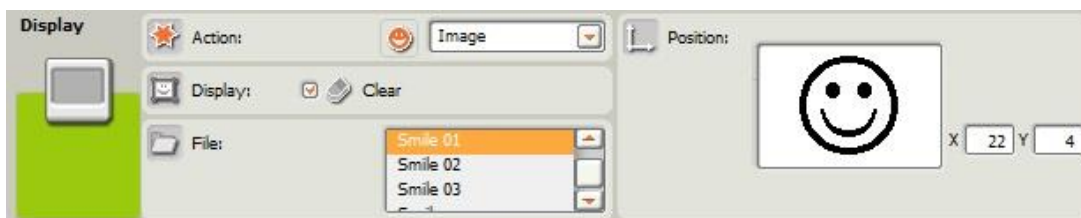


Figura 45 - Configuração do bloco *Display*

A programação deve ficar assim:



Figura 46 – 4ª Tarefa - Programação

Conecte o cabo USB e envie a programação.

## 7.5 - 5ª TAREFA: MOVER O CARRO UTILIZANDO O SENSOR TOUCH

Na quinta tarefa, o carro deve mover-se para frente durante 3 segundos, somente quando for acionado um sensor *touch* (toque). Assim, deve-se pegar o sensor e conectar ao cérebro. Cuide para que a porta de entrada do sensor seja a mesma que você vai configurar no programa.

No exemplo mostrado na Figura 48 - Configuração do bloco *Touch*, colocamos o sensor na porta de número 1.

### Bloco *Touch* (Toque)



Figura 47 - Bloco *Touch* (Toque)



Figura 48 - Configuração do bloco Touch

A programação deve ficar assim:

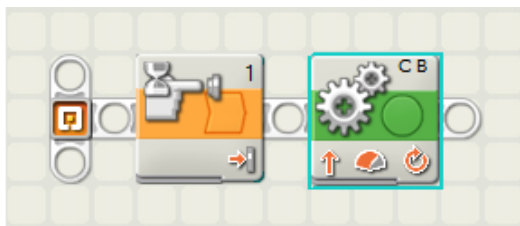


Figura 49 – 5ª Tarefa - Programação

Conecte o cabo USB e envie a programação.

## 7.6 - 6ª TAREFA: COMANDO LOOP

Nota-se que na resolução da tarefa anterior, o programa termina depois de ter executado os blocos, assim, mesmo que o aluno volte a pressionar o sensor de toque, o carro não se mexe. Para corrigir esse problema e deixar o programa rodando sempre, vamos acrescentar o comando *Loop*, que pode ser configurado para retornar ao início do programa infinitamente ou retornar uma quantidade de vezes estipulada pelo programador.

Bloco *Loop*



Figura 50 - Bloco Loop

A programação deve ficar assim:

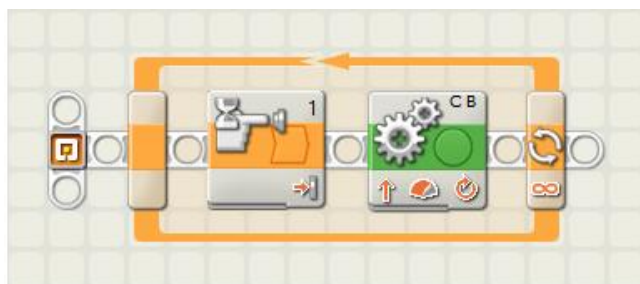


Figura 51 – 6ª Tarefa - Programação

Conecte o cabo USB e envie a programação.

## 7.7 - 7ª TAREFA: APRENDENDO O COMANDO SWITCH (SE)

Nota-se também que, na resolução da tarefa anterior, o programa espera o sinal oriundo do sensor de toque, que depois de acionado, habilita a execução do bloco *Mover* que fará sua rotina até termina sua configuração. Enquanto o programa estiver executando o bloco *mover*, mesmo que o aluno volte a apertar o sensor de toque, não acontece nada.

Para resolver esse problema, vamos acrescentar o comando *switch*, que pode ser configurado de várias formas. Para essa tarefa vamos deixar na configuração original, onde aparece o sensor de toque (Touch Sensor).

Bloco *Switch* (Se)



Figura 52 - Bloco *Switch* (SE)

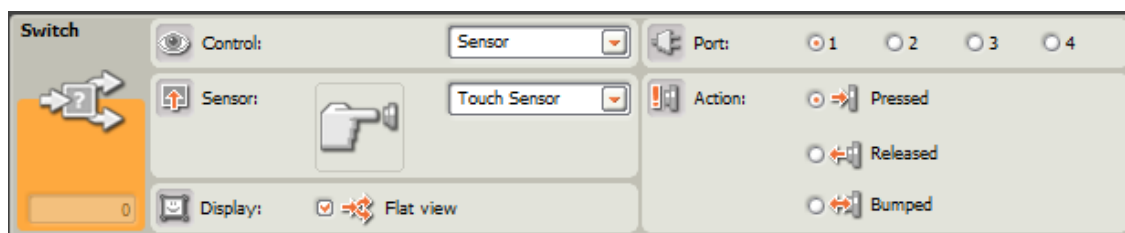


Figura 53 – Configuração do bloco *Switch*

Como este bloco será muito utilizado para a resolução das próximas tarefas, vamos explicar detalhadamente todos os parâmetros possíveis para a configuração do bloco *Switch*.

Ajuste dos parâmetros do comando “Se” (Figura 52 - Bloco *Switch* (SE)):

**Control:** *Seleciona se deseja utilizar com um sensor ou em modo de verdadeiro ou falso;*

**Sensor:** *Seleciona o sensor que deseja utilizar. Neste caso está selecionado o sensor de toque;*

**Display:** *Ao selecionar a opção “Flat view” mostrará os dois ‘caminhos’ possíveis. Se a opção for desmarcada, mostrará apenas um, sendo um modo de economizar espaço no programa;*

**Port:** *Seleciona qual porta está sendo utilizadas.*

**Action:** *Seleciona como deve-se comportar o sensor. Pressed (pressionado), Released (liberado) e Bumped (pressionado e depois liberado).*

Depois de ter esmiuçado as possibilidades de configuração do comando *Switch*, vamos ver como é o funcionamento deste comando.







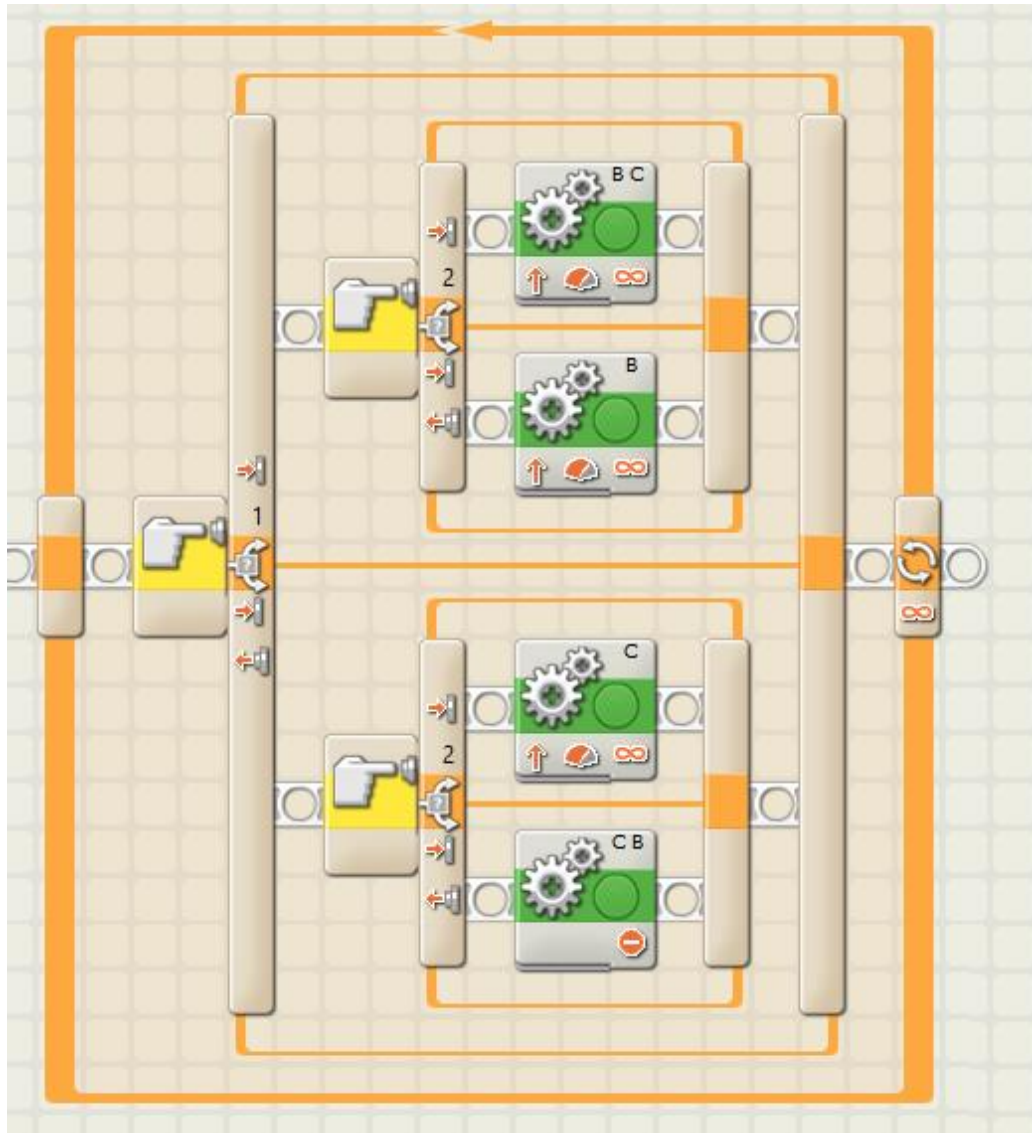


Figura 55 – 8ª Tarefa - Programação

Após a análise do programa, veja que o primeiro sensor de toque está conectado na porta 1 e o outro sensor está na porta 2 do cérebro. Explicando a lógica utilizada, colocamos os sensores em um efeito cascata. Desta forma, para chegar até as funções dos motores, precisa passar pelo teste dos dois sensores. Assim, chegaremos a quatro combinações possíveis, detalhadas na tabela abaixo:

Possibilidades	Sensor porta 1	Sensor porta 2	Ação
1º Possibilidade	Acionado	Acionado	Motores B e C para frente
2º Possibilidade	Acionado	Desacionado	Apenas motor B para frente
3º Possibilidade	Desacionado	Acionado	Apenas motor C para frente
4º Possibilidade	Desacionado	Desacionado	Motores B e C parados

Conecte o cabo USB e envie a programação.

## 7.9 - 9ª TAREFA: SEGUIR LINHA PRETA

Para esta tarefa, será utilizado a pista padrão que vem junto com o Kit da Lego Mindstorms NXT 2.0. No caso do nosso estudo, acabamos redesenhando e confeccionando a nossa própria pista em lona (material de banner) para uma maior durabilidade. Na Figura 56 - Pista para robô seguidor de linha, está o modelo utilizado nas oficinas.

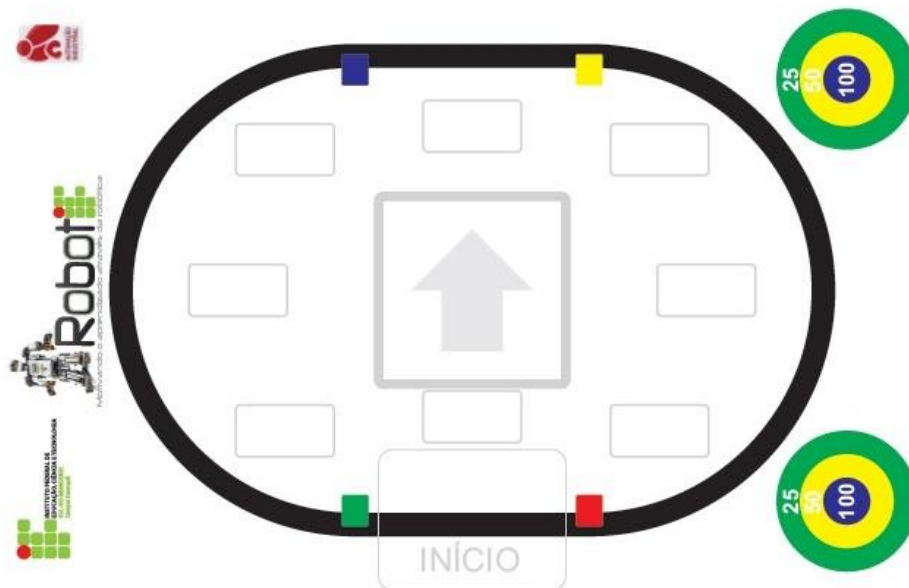


Figura 56 - Pista para robô seguidor de linha

Antes de explicarmos a tarefa e a utilização do sensor de cor, temos que fixar o próprio sensor na frente do carro. Para essa fixação, segue a sequência da montagem.

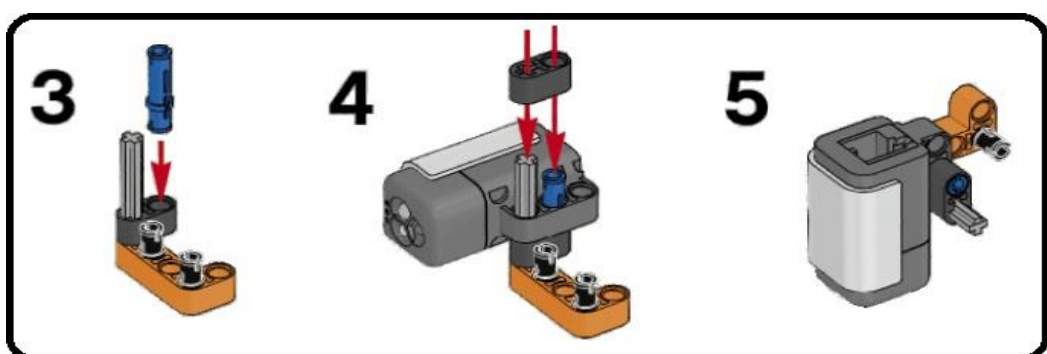


Figura 57– Montagem do Sensor de cor - 01 – Fonte: Software LEGO Mindstorms NXT

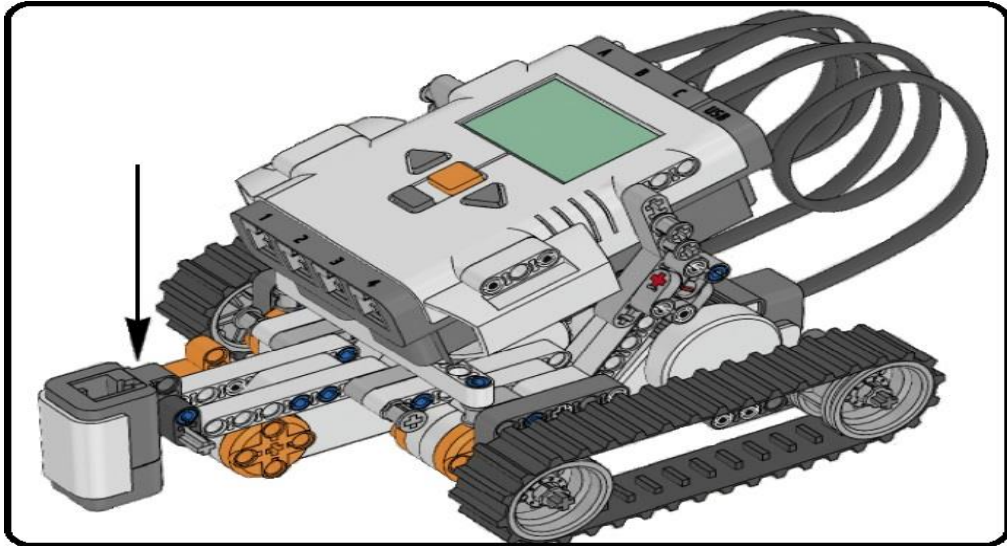


Figura 58– Montagem do Sensor de cor - 02 – Fonte: Software LEGO Mindstorms NXT

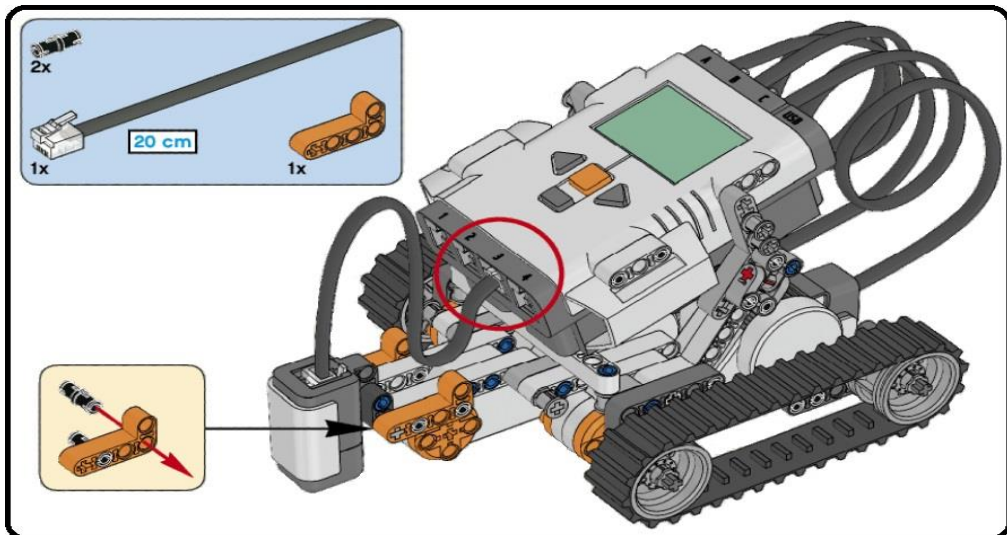


Figura 59 – Montagem do Sensor de cor - 03 – Fonte: Software LEGO Mindstorms NXT

Para seguir a linha preta vamos usar o comando *Switch* novamente, porém ao invés de escolher a opção *Touch Sensor* (Sensor de toque), deve-se escolher a opção *Color Sensor* (Sensor de cor).

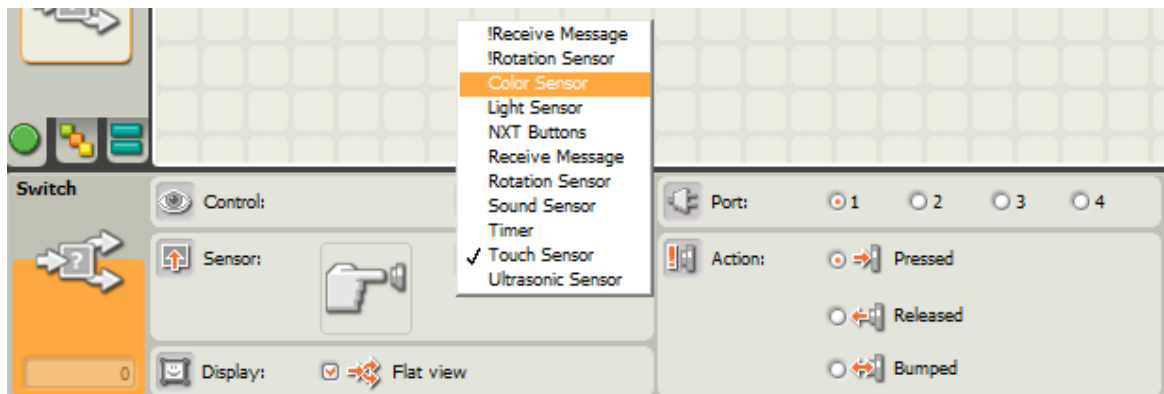


Figura 60 - Selecionando sensor de cor no comando Switch



Figura 61 - Configuração do bloco Switch para sensor de cor

O seu funcionamento é o mesmo, ele executará a linha de cima apenas quando a cor ou as cores forem detectadas pelo sensor.

A lógica usada para seguir a linha será essa: Se o sensor detectar branco (não está sobre a linha), ande para esquerda, se não detectar branco (está sobre a linha), ande para frente.

A programação deve ficar assim:

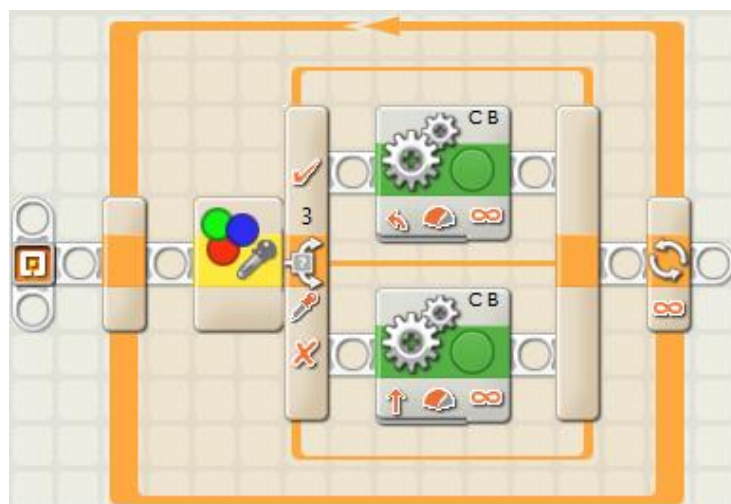


Figura 62 - 9ª Tarefa - Programação

Conecte o cabo USB e envie a programação.

## 7.10 - 10ª TAREFA: IDENTIFICAÇÃO DE CORES E AÇÕES ESPECÍFICAS.

Nesta tarefa o robô deve seguir a linha preta, igualmente como na tarefa anterior, e quando identificar o quadrado na cor azul, deve parar por 3 segundos. Para o robô parar e esperar o tempo determinado, deve ser usado o bloco *Time* e configurar como 3 segundos.

Bloco *Time*



Figura 63 - Bloco Timer

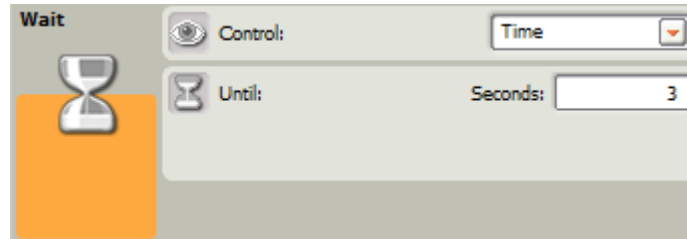


Figura 64 - Configuração do bloco Timer

Interessante é que o carro deve continuar a seguir a linha preta depois de ter esperado os 3 segundos, para isto é necessário fazer com que o sensor de cor saia de cima do quadrado de cor azul, assim os alunos devem inserir um outro bloco *Mover* para realizar essa ação.

A programação pode ficar assim:

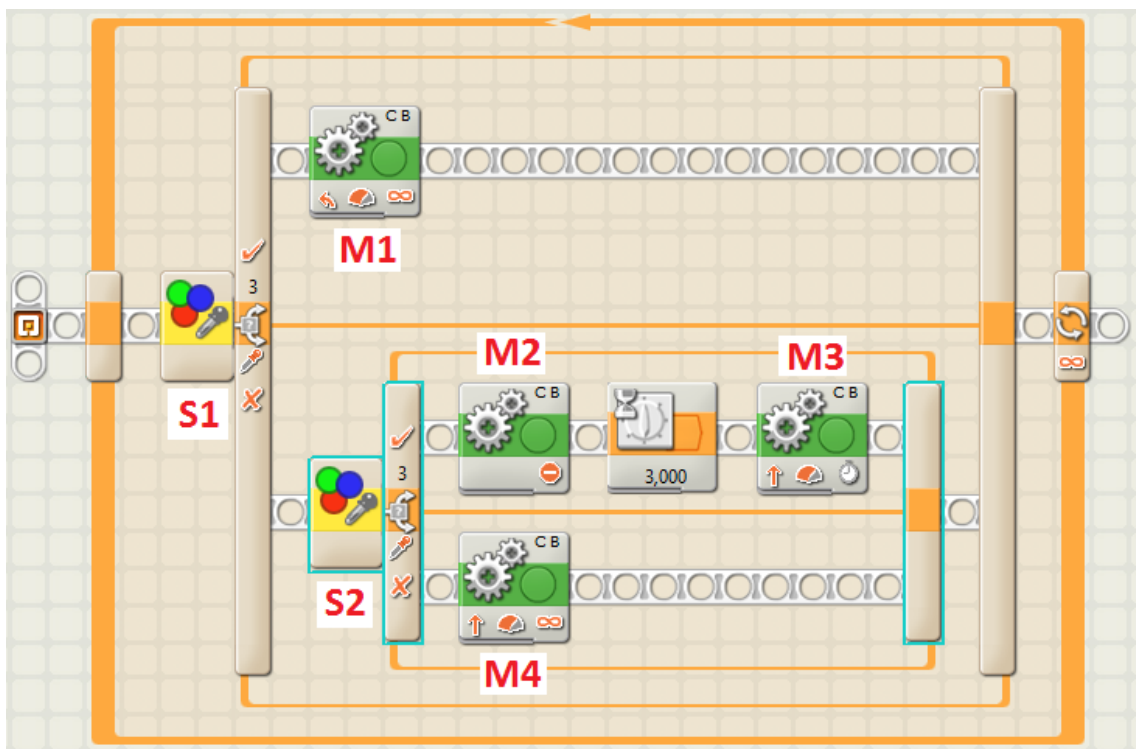


Figura 65 – 10ª Tarefa – Programação

Descrição do programa:

Notamos que o primeiro sensor (S1) a ser testado faz a primeira análise onde ele identifica apenas a cor branca, assim caso essa análise seja positiva, o programa executa a linha de cima e o robô dobra para o lado esquerdo, exatamente como na tarefa anterior.



Caso o primeiro sensor (S1) detecte outra cor sem ser branco, o programa passa a executar a linha de baixo, onde ele vai perguntar para o mesmo sensor (agora sensor S2), qual cor ele está detectando. Caso ele veja a cor azul, o programa executa a linha de cima, onde está programado para fazer o robô parar por 3 segundos e depois tirar o sensor de cima do quadrinho azul. Isto faz-se necessário pois se não fizer isto, o sensor ficará sempre detectando a cor azul e o robô não vai se mexer mais. E caso o segundo sensor (S2) detecte qualquer outra cor, sem ser azul e branco, o robô locomove-se para frente ilimitadamente.

### Configuração do sensor de cor S1:



Figura 66 - Configuração do Sensor de cor S1

### Configuração do sensor de cor S2:



Figura 67 - Configuração do Sensor de cor S2

### Configuração do Motor M1:



Figura 68 - Configuração do Motor M1

### Configuração do Motor M2:

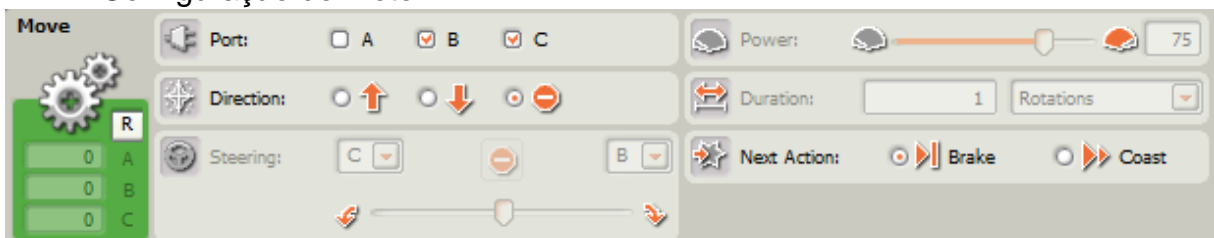


Figura 69 - Configuração do Motor M2

### Configuração do Motor M3:



Figura 70 - Configuração do Motor M3

### Configuração do Motor M4:



Figura 71 - Configuração do Motor M4

Conecte o cabo USB e envie a programação.

## 7.11 - 11ª TAREFA: IDENTIFICAÇÃO DE CORES E AÇÕES ESPECÍFICAS.

Agora o robô precisará, além de realizar as tarefas 9 e 10, mostrar uma figura na tela e parar por 2 segundos quando passar pelo quadrado vermelho. Seguindo na mesma tarefa, também programa-lo para falar a cor verde (em inglês) quando o sensor de cor identificar o quadrado na cor verde. A programação pode seguir a mesma ideia da programação anterior, ou seja, deve-se usar um bloco *Switch* para o vermelho e outro para a cor verde.

Assim, a programação incluiria os sensores de cor sinalizados como S3 e S4. No S3, a configuração ficaria a mesma dos outros, mas ficaria marcando apenas a cor vermelha. No caso do sensor S4, marcaria apenas a cor verde. Nos outros blocos, a configuração já foi explicada nos exercícios anteriores.

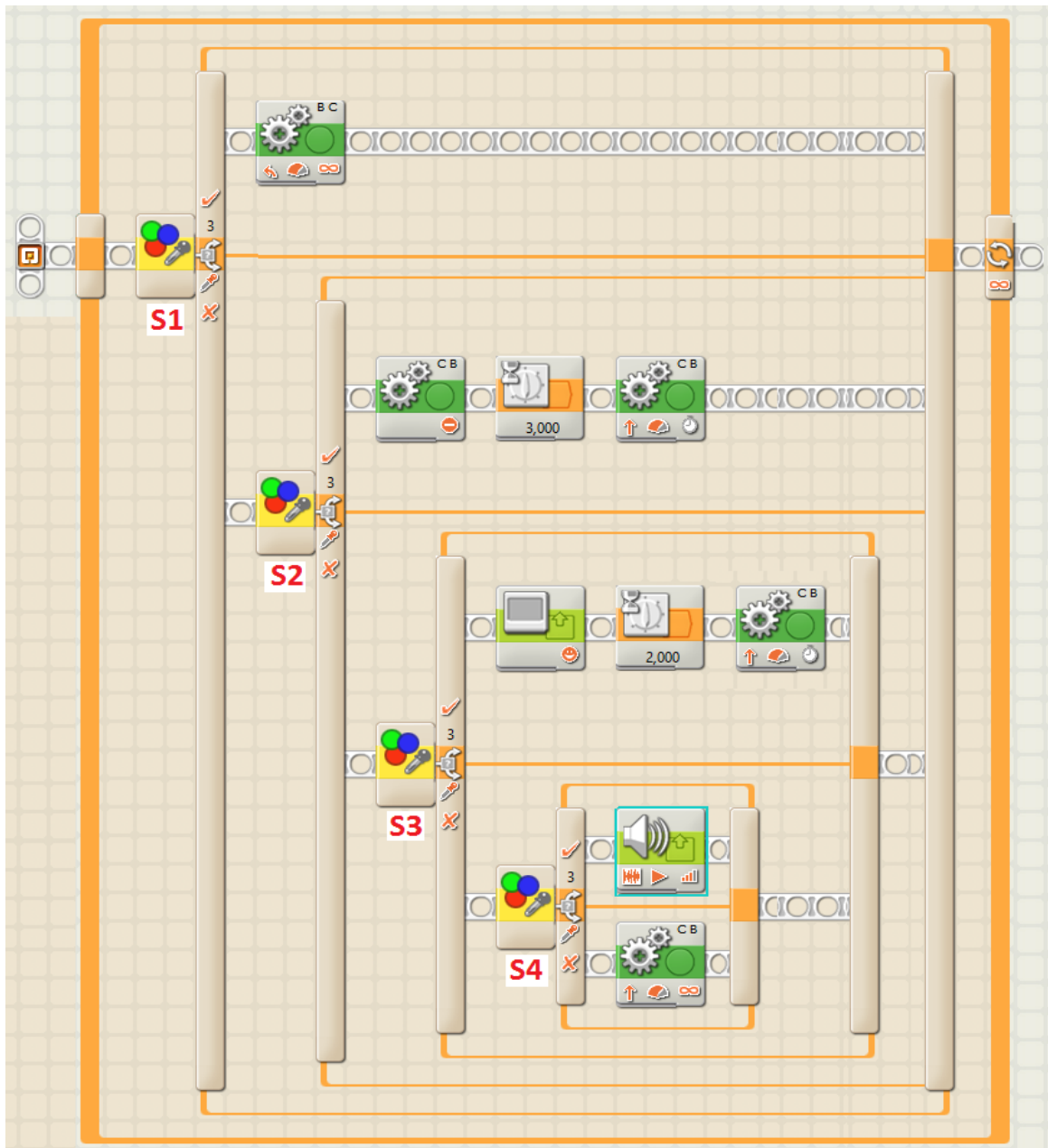


Figura 72 – 11ª Tarefa – Programação

Conecte o cabo USB e envie a programação.

## 7.12 - 12ª TAREFA: DETECTANDO UM OBSTÁCULO

Nesta tarefa utilizaremos o sensor ultrassônico que deverá ser fixado na frente do carro, logo acima do sensor de cor. Como não há instruções e figuras para o acoplamento do sensor no robô, contamos com a criatividade dos alunos para a realização desta fixação. Após a instalação do sensor, o carro deve andar para frente em linha reta, até que detecte um obstáculo a menos que 15 centímetros. Caso isto aconteça, o carro deve parar e seguir parado até que o obstáculo seja retirado, assim o carro voltaria a se deslocar normalmente.



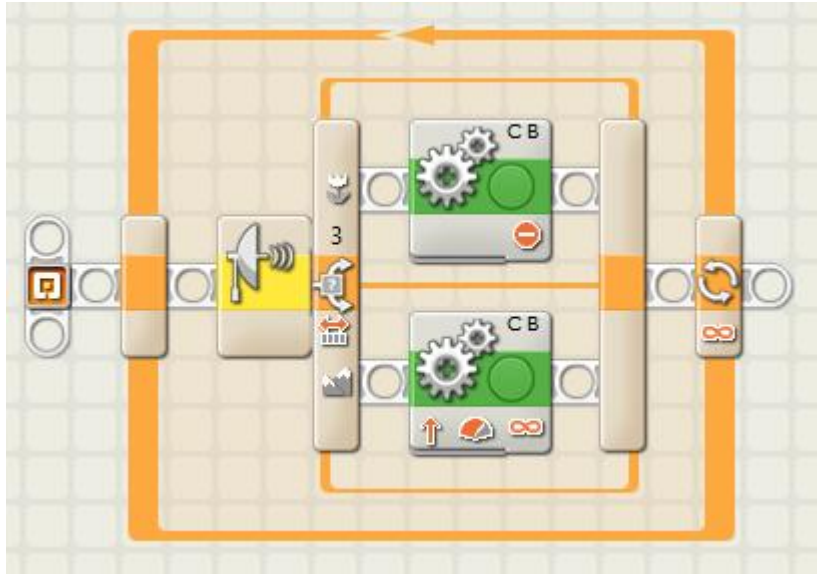


Figura 73 – 12ª Tarefa – Programação

Configuração do sensor ultrassônico:



Figura 74 – Configuração do sensor ultrassônico

Conecte o cabo USB e envie a programação.

### 7.13 - 13ª TAREFA: DETECTANDO UM OBSTÁCULO

Nesta tarefa, o carro deve seguir andando para frente até que detecte um obstáculo a 15 centímetros, fazendo-o parar e seguir parado igualmente a tarefa anterior. A diferença é que quando o sensor ultrassônico detectar um obstáculo a menos que 10 centímetros, o carro deve dar ré.

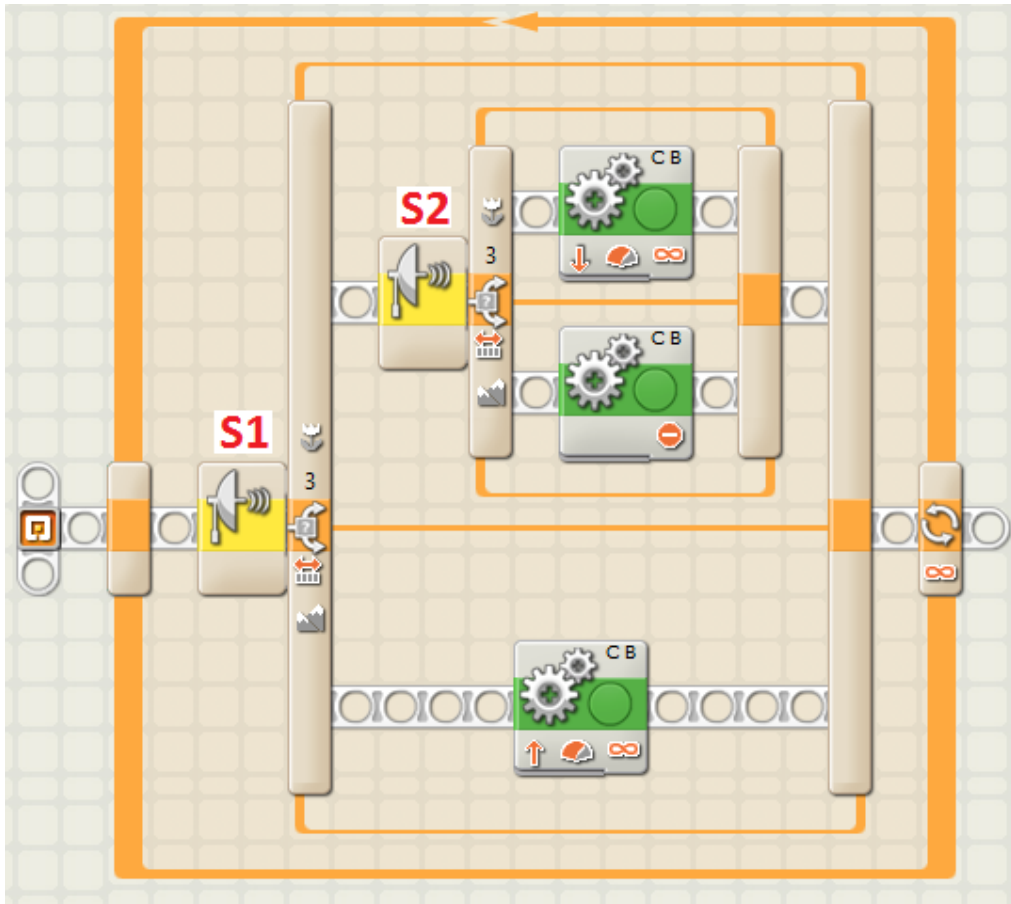


Figura 75 – 13ª Tarefa – Programação

### Configuração do sensor de cor S1:



Figura 76 - Configuração do Sensor Ultrassônico S1

### Configuração do sensor de cor S2:



Figura 77 - Configuração do Sensor Ultrassônico S2

Conecte o cabo USB e envie a programação.

## 7.14 - 14<sup>o</sup> TAREFA: DETECTANDO UM OBSTÁCULO E DESVIANDO

Nesta tarefa, solicitamos aos alunos para programar o carro para identificar e desviar de um obstáculo a 10 centímetros de distância. Quando ocorre a detecção do obstáculo, o carro deverá girar em seu eixo em torno de 90° para um dos lados e seguir na sua trajetória. Lembrando que, caso haja outro obstáculo após realizar o primeiro desvio, o carro tem que estar preparado para refazer a mesma operação infinitas vezes.

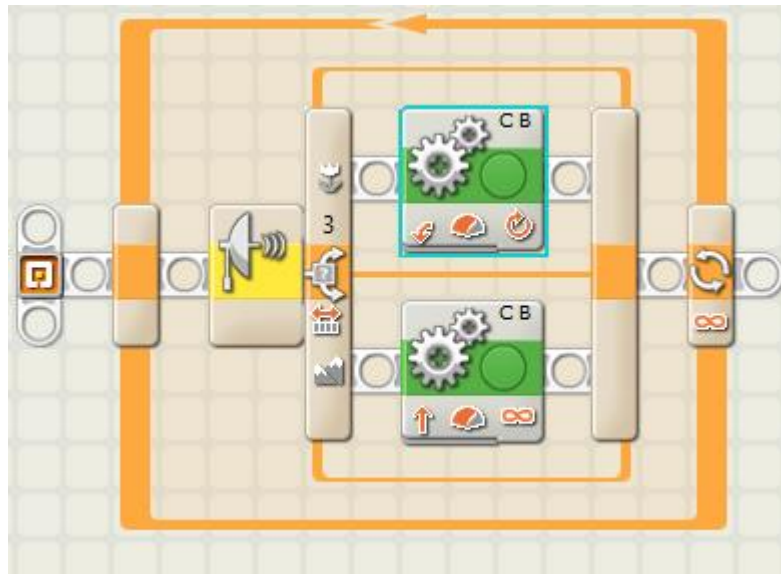


Figura 78 – 14ª Tarefa – Programação

Note que as configurações do bloco *Mover* da linha de cima do programa não estão definidas, pois isto exigirá que o aluno faça por tentativa e erro. Já que o objetivo é que o robô redirecione exatamente 90° para a esquerda, aconselha-se que a configuração do motor seja em rotações ou em segundos.

Conecte o cabo USB e envie a programação.

## 7.15 - 15<sup>o</sup> TAREFA: DETECTANDO UM OBSTÁCULO SOBRE A LINHA PRETA

Nesta tarefa o robô deve seguir a linha preta normalmente e, ao chegar a 5 cm de distância do objeto colocado em alguma parte da pista, ele deverá parar.

Deve-se cuidar onde o sensor foi acoplado, pois o objeto precisa estar em seu “campo de visão” para conseguir identifica-lo.



Figura 79 - Pista com a sinalização do obstáculo

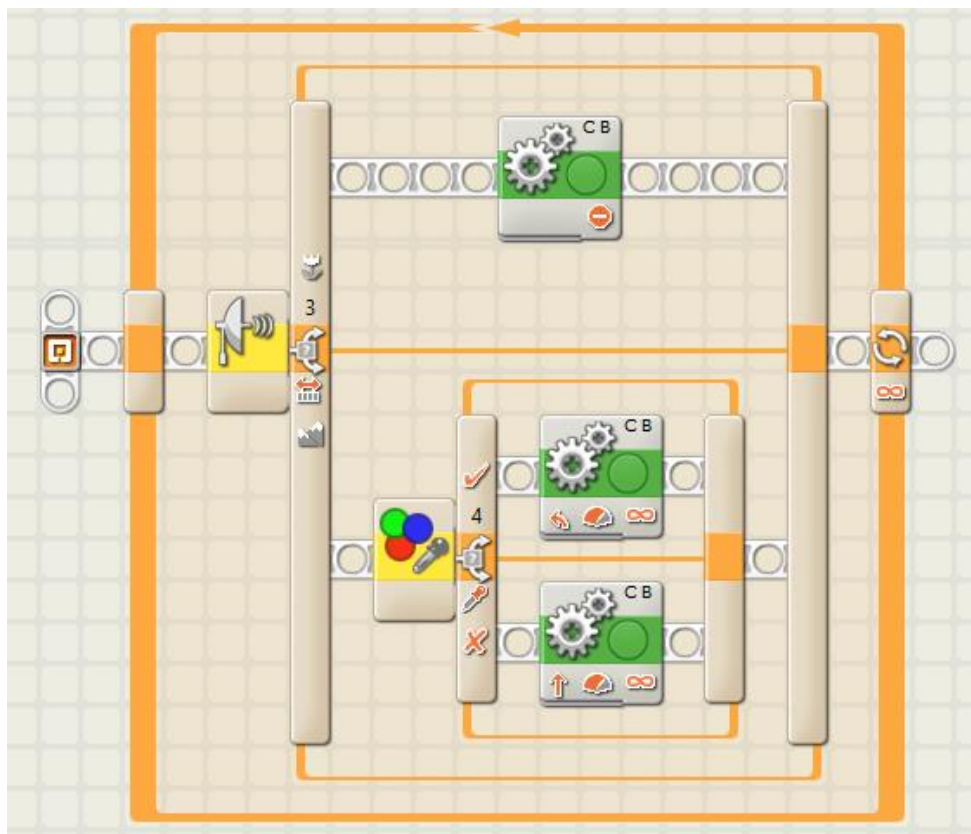


Figura 80 – 15ª Tarefa – Programação

Conecte o cabo USB e envie a programação.

## 7.16 - 16ª TAREFA: DETECTANDO UM OBSTÁCULO SOBRE A LINHA PRETA E DESVIANDO

Nesta tarefa, o robô precisará identificar o objeto como na tarefa anterior, mas ao invés de apenas parar, ele deverá desviar do objeto e continuar o percurso sobre a linha preta. A única alteração que precisa ser feita na programação anterior, é a inclusão de uma rotina de desvio.

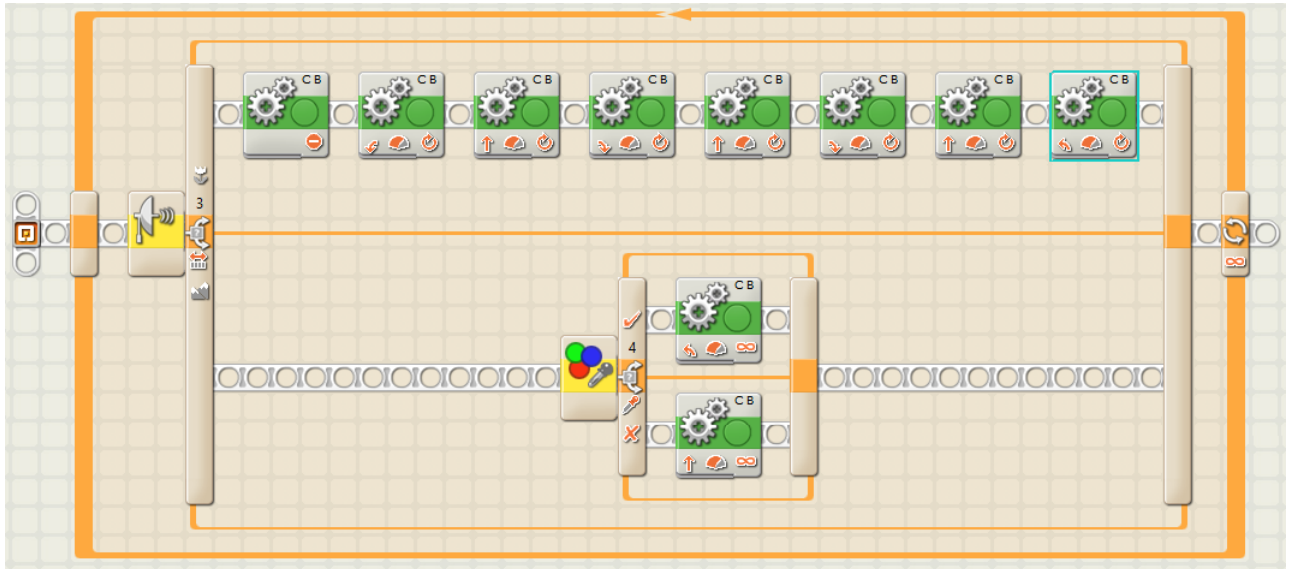


Figura 81 – 16ª Tarefa – Programação

Note que não será detalhado a rotina para que o robô desvie do obstáculo, pois como trata-se de configurações dos blocos *Mover*, ficaria a cargo dos alunos este ajuste mais preciso.

Mas para auxiliar os alunos nesta empreitada, vamos descrever, resumidamente, cada um dos blocos. Começando pelo primeiro bloco da linha de cima, onde pode-se notar que a está configurado para parar o robô. No segundo, tem que dobrar 90° a esquerda. No terceiro, andar para frente por um determinado tempo. No quarto, tem que dobrar 90° a direita. No quinto, andar para frente por um determinado tempo. No sexto, tem que dobrar 90° a direita. No sétimo, andar para frente por um determinado tempo. No último, tem que dobrar 90° a esquerda.

Conecte o cabo USB e envie a programação.

## 7.17 - 17ª TAREFA: IDENTIFICAR E REMOVER OS OBJETOS DE DENTRO DA PISTA, SEM PODER SAIR DE DENTRO DO CIRCULO

Nesta tarefa, serão colocados objetos (caixas de papelão) sobre os números destacados na imagem abaixo (Figura 82 - Pista com os locais para inserir os obstáculos. O carro começará a tarefa no centro da pista (posição indicada pela seta), deverá

identificar e retirar todos os objetos da área, sem sair inteiramente de seu limite, indicado pela linha preta.

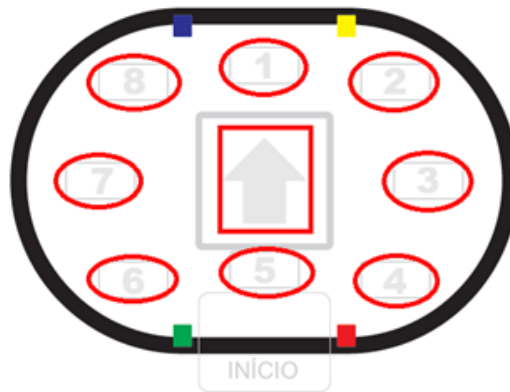


Figura 82 - Pista com os locais para inserir os obstáculos

Nesta tarefa não será preciso adicionar nenhum sensor ou apresentar algum comando novo. Ela pode ser realizada com o mesmo robô e com os mesmos comandos utilizados anteriormente. O aluno poderá programar o carro para girar sobre seu próprio eixo, enquanto o sensor de cor não detectar a cor preta e o sensor ultrassônico não detectar algum objeto. Para fazer o robô girar sobre seu próprio eixo, ou seja, girar sem sair do lugar, o melhor a fazer é usar blocos diferentes para as rodas girarem em sentidos opostos e em velocidades iguais.

Quando ele identifica algo, ele avança e empurra o objeto para fora do círculo. Ele não poderá passar do limite da pista, pois quando o sensor de cor detectar a linha preta, o carro irá parar e voltará para trás, mesmo se ele ainda estiver vendo o objeto.

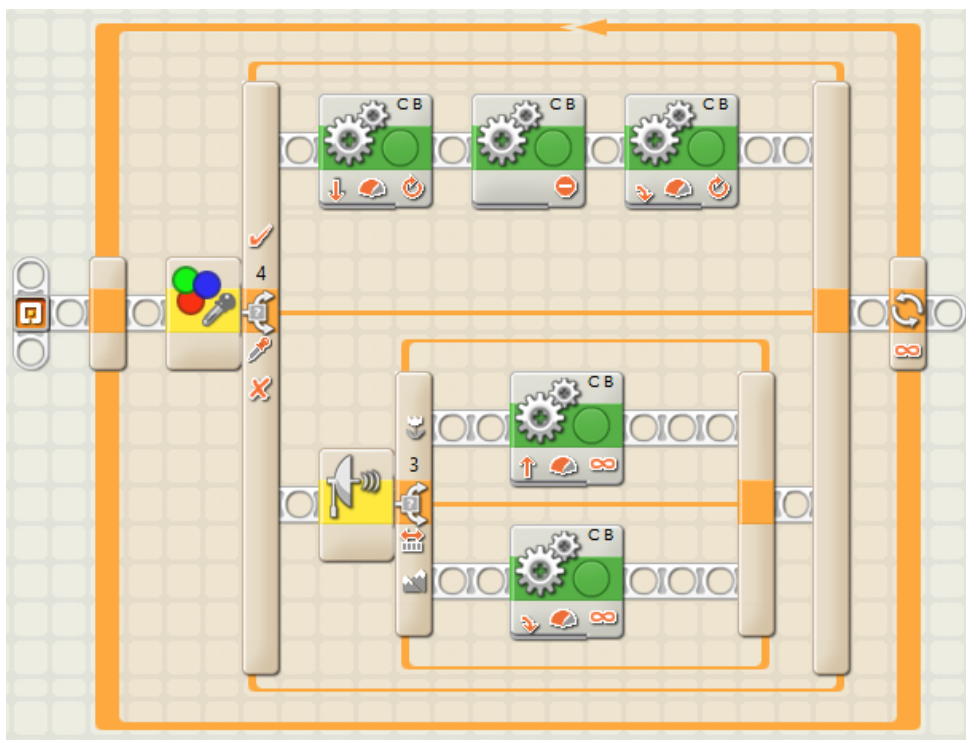


Figura 83 – 17ª Tarefa – Programação

Conecte o cabo USB e envie a programação.