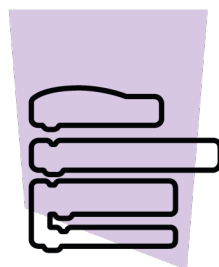
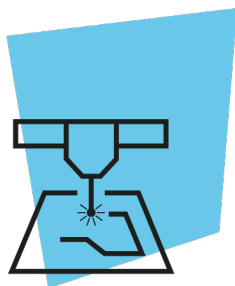
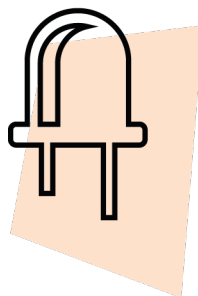
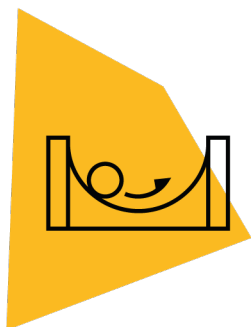
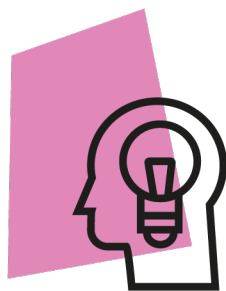


# ROBÓTICA

Módulo 1



## Kit Chassi 2WD Ponte H\* AULA 32

\*Disponível no kit distribuído em 2023

## **GOVERNADOR DO ESTADO DO PARANÁ**

Carlos Massa Ratinho Júnior

## **SECRETÁRIO DE ESTADO DA EDUCAÇÃO**

Roni Miranda Vieira

## **DIRETOR DE TECNOLOGIA E INOVAÇÃO**

Claudio Aparecido de Oliveira

## **COORDENADOR DE TECNOLOGIAS EDUCACIONAIS**

Marcelo Gasparin

### **Produção de Conteúdo**

Cleiton Rosa

Cleres Rozeli Cristófolli Grande

Simone Sinara De Souza

### **Validação de Conteúdo**

Cleiton Rosa

### **Revisão Textual**

Orlando de Macedo Junior

### **Projeto Gráfico e Diagramação**

Edna do Rocio Becker

### **Ilustração**

Jocelin Vianna (ASCOM)

### **Modelagem Blender**

Cleiton Rosa

Roberto Carlos Rodrigues

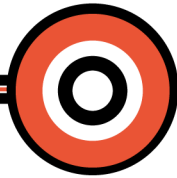
2023

- Aula 01 Por Que Robótica?
- Aula 02 Tensão, Corrente e Resistência
- Aula 03 Kit de Robótica
- Aula 04 Arduino Uno R3
- Aula 05 Softwares Arduino IDE e mBlock
- Aula 06 Portas Digitais
- Aula 07 Circuito Elétrico
- Aula 08 LED e Resistor
- Aula 09 Semáforo [Carros]
- Aula 10 Semáforo [Cruzamento Carros]
- Aula 11 Semáforo [Pedestres]
- Aula 12 Semáforo [Cruzamento Carros + Pedestres]
- Aula 13 Push Button
- Aula 14 Feedbacks + Inventário I
- Aula 15 Semáforo [Carros + Pedestres com Botão]
- Aula 16 Display 7 Segmentos
- Aula 17 Fonte DC + Plug P4
- Aula 18 Portas PWM
- Aula 19 LED Fade-In
- Aula 20 LED Fade-Out
- Aula 21 Super Máquina 80's
- Aula 22 Super Máquina 2008
- Aula 23 Potenciômetro
- Aula 24 Buzzer Passivo
- Aula 25 LED RGB
- Aula 26 Arco-Iris
- Aula 27 Sensor LDR
- Aula 28 Feedbacks + Inventário II
- Aula 29 Sensor de Temperatura
- Aula 30 Sensor de Obstáculo IR
- Aula 31 Controle Motor DC
- Aula 32 Kit Chassi 2WD Robô**
- Aula 33 Seguidor de Linha
- Aula 34 Sensor de Distância
- Aula 35 Sensor de Estacionamento
- Aula 36 Display LCD 16x2
- Aula 37 Trena Digital
- Aula 38 Robô Sumô [Estrutura]
- Aula 39 Robô Sumô [Programação + Treinamento I]
- Aula 40 Robô Sumô [Programação + Treinamento II]
- Aula 41 Disputa de Sumôs
- Aula 42 Feedbacks + Inventário III

Aula 31  
Controle Motor DC

Aula 32  
**Kit Chassi  
2WD  
Ponte H**

Aula 33  
Seguidor de Linha



# Sumário

Introdução	2
Objetivos desta aula	2
Competências gerais previstas na BNCC	3
Habilidades do século XXI a serem desenvolvidas	4
Lista de materiais	4
Roteiro da aula	5
1. Contextualização	5
2. Montagem e programação	5
3. Feedback e finalização	16
Referências	16



## Introdução

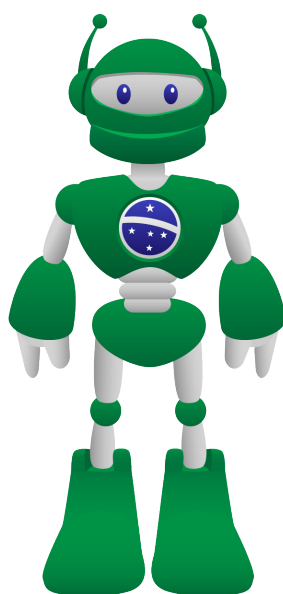
Uma área da robótica que vem apresentando avanços na atualidade é a robótica móvel, um exemplo disso, são os robôs exploradores enviados a outros planetas pela Administração Nacional da Aeronáutica e Espaço (NASA) visando a obter informações sobre a superfície e atmosfera desses astros.

Nesta aula, apresentamos o funcionamento de um robô móvel com trajetória programável, a partir da montagem do kit chassi 2WD, presente no kit de robótica.



## Objetivo desta aula

- Entender o funcionamento de um robô chassi 2WD (2 rodas);
- Programar o robô chassi 2WD para se deslocar em uma trajetória pré-determinada.





## Competências gerais previstas na BNCC

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

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

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

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

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



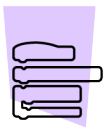
## Habilidades do século XXI a serem desenvolvidas

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



## Lista de materiais

- 1 placa Arduino Uno R30
- 1 cabo USB
- 1 ponte H L298n (Driver)
- 1 fonte de energia (pilhas ou bateria 9V)
- 6 jumpers macho-fêmea
- 2 jumpers macho-macho
- 1 kit chassi 2WD
- 1 notebook
- Software mBlock ou Arduino IDE



## Roteiro da aula

### 1. Contextualização

Quando pensamos em robôs, provavelmente, a primeira imagem que vem à mente é a de um robô humanoide, isto é, semelhante à espécie humana, como os apresentados em diversos filmes de ficção científica. Entretanto, há vários formatos de robôs móveis que estão sendo utilizados cada vez mais em nosso cotidiano, principalmente pelo campo industrial e polícia investigativa. Você saberia dizer em que atividades os robôs são utilizados nesses campos?

Os robôs de vários tamanhos e dotados de sensores e atuadores realizam atividades consideradas difíceis ou perigosas para a espécie humana, como, por exemplo, carga e descarga de materiais, desarmamento de bombas e minas terrestres, pintura em recinto fechado, exploração espacial, entre outras funções.

No ensino de robótica, utilizamos os robôs móveis para medir distâncias, testar a velocidade, identificar obstáculos, participar de competições, entre outras atividades. Nesta aula, usaremos o kit chassi 2WD (2 rodas), presente no kit de robótica, para simular os possíveis movimentos de um robô móvel deste formato.

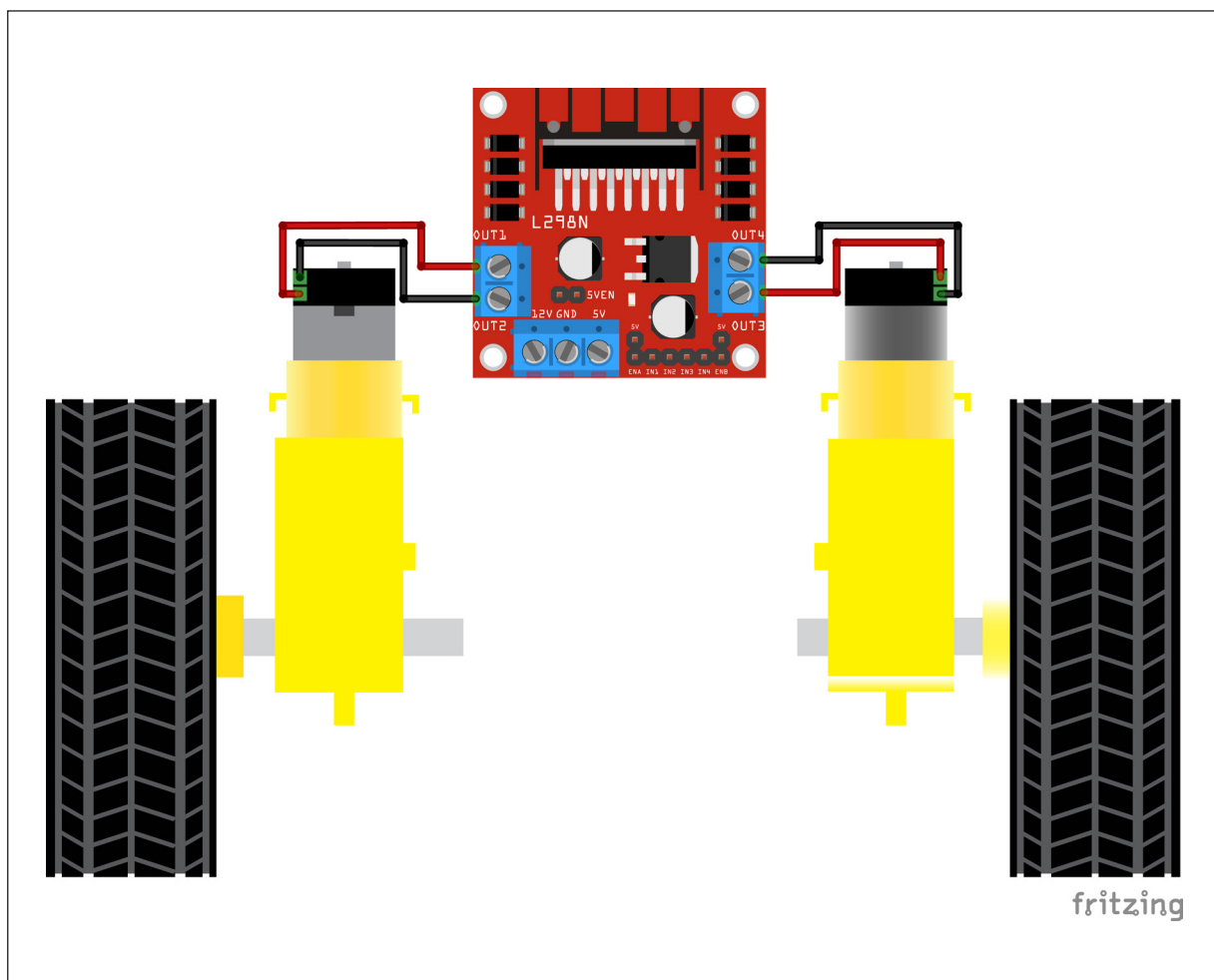
### 2. Montagem e programação

O kit chassi 2WD vem desmontado no kit de Robótica. Caso seja o primeiro a utilizar este kit, você provavelmente terá que montá-lo. Para tanto, utilize o **Manual de Montagem do Kit Chassi 2WD Robô**, disponível no kit de robótica. Mas, atenção! Uma vez montado o kit chassi e, após sua utilização na aula, ele deverá ser guardado montado, com os demais componentes do kit de robótica.

Com o kit chassi 2WD montado, vamos montar os demais componentes eletrônicos que serão utilizados nesta aula. Fixe com o auxílio de fita adesiva ou parafusos a ponte H L298n sobre o chassi 2WD, mantendo os bornes dos motores voltados para as laterais. A seguir, fixe com auxílio de uma pequena chave fenda, os dois motores DC nos bornes da ponte H (figura 1).

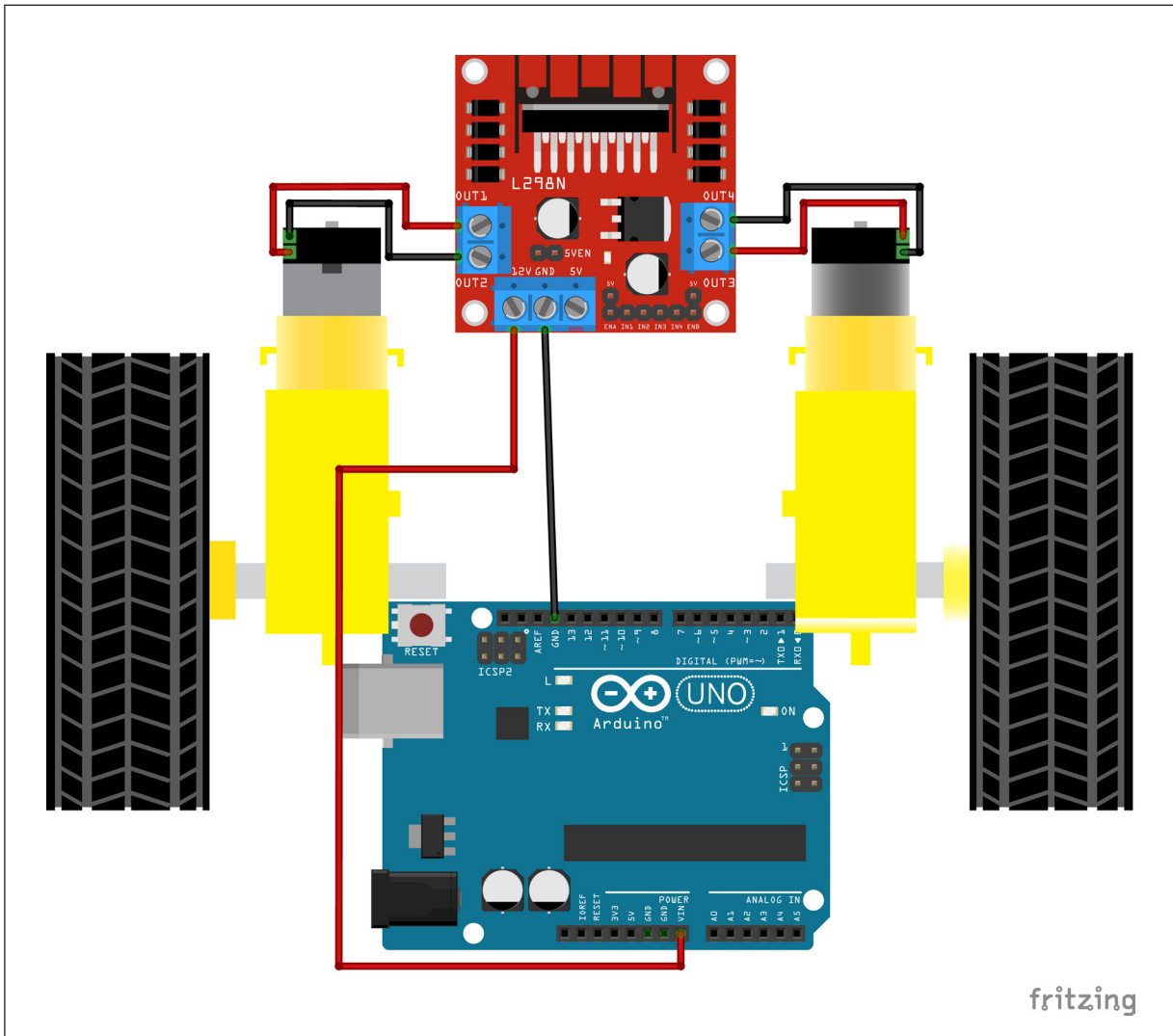


Figura 1 - Fixando a ponte H no chassi 2WD e nos motores DC



Com o auxílio de fita adesiva ou parafusos e da chave fenda, fixe a placa Arduino sobre o chassi 2WD e a seguir, utilizando dois jumpers macho-macho, interligue os pinos GND e VIN do Arduino aos bornes GND (central) e 12V (esquerda), respectivamente, da ponte H (figura 2).

Figura 2 - Alimentação da Ponte H através da placa Arduino



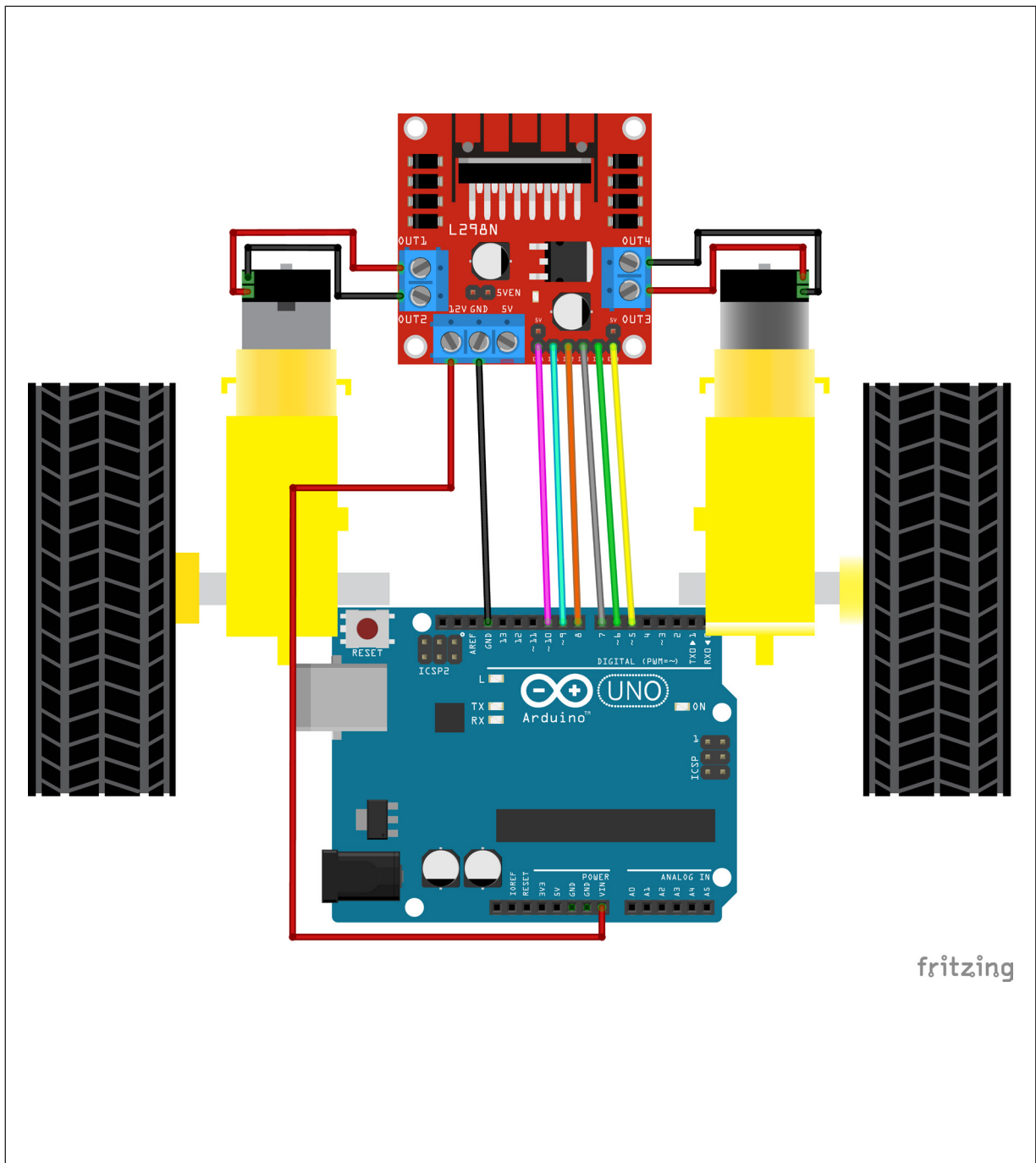
Utilizando 6 jumpers macho-fêmea conecte os pinos da ponte H com as portas digitais do Arduino (tabela 1):

Tabela 1 - Conexão entre a ponte H e o Arduino

Pinos Ponte H	ENA	IN1	IN2	IN3	IN4	ENB
Portas Arduino	~10	9	8	7	6	~5

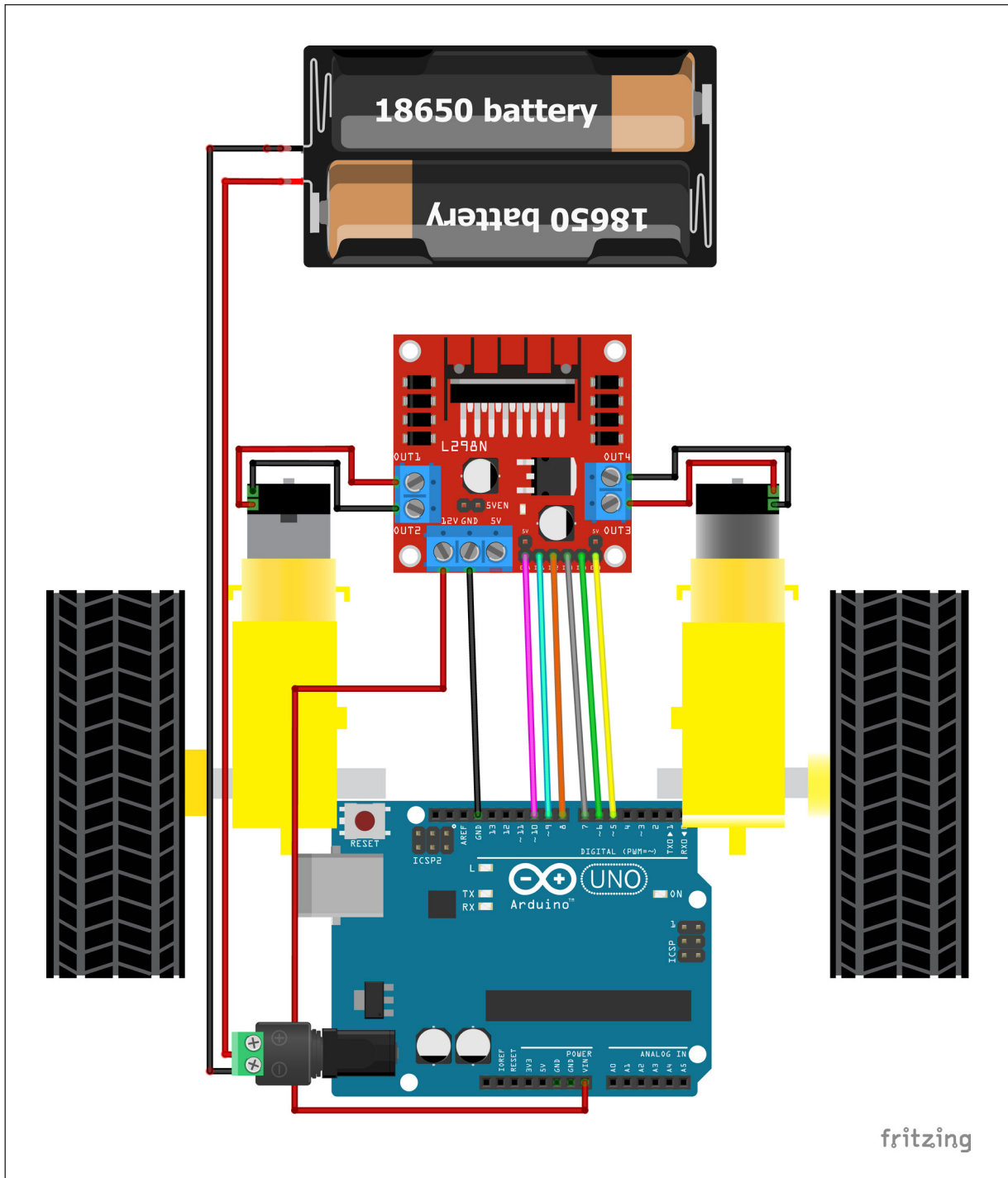
Fique atento para os pinos ENA e ENB da ponte H, eles devem ser conectados às portas digitais com recurso PWM (portas indicadas com o símbolo ~), para que seja possível controlar a potência dos dois motores (figura 3).

Figura 3 - Conexão dos pinos da ponte H ao Arduino



Agora finalize a montagem conectando a fonte de energia (pilhas ou bateria) ao conector Jack P4 da placa Arduino (figura 4).

Figura 4 - Conexão da fonte de energia ao Arduino



Com os componentes eletrônicos montados, vamos programar, por codificação e por blocos, os movimentos do robô.

### b. Linguagem de programação por código

Para essa programação, será necessária a biblioteca **L298N** desenvolvida por Andrea Lombardo, (para recordar como instalar uma biblioteca, consulte a **Aula 05 - Softwares Arduino IDE e mBlock**).

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

No software IDE, escreva ou copie e cole o código-fonte de programação (quadro 1):

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

```
/*
*****
/* Aula 32 - Kit Chassi 2WD com ponte H - Robô */
/* Programação: Nesta programação controlamos dois */
/* motores DC utilizando a placa Motor Driver L298N. */
/* para mover o robô em todas as direções possíveis. */
/* Links para obtenção da biblioteca by Andrea Lombardo */
/* http://librarymanager/All#L298N#EASY */
/* https://github.com/AndreaLombardo/L298N */
*****
/* Inclui a Biblioteca para controle da ponte H. */
#include <L298NX2.h>

#define ENA 10 // ENA precisa estar em uma porta PWM
#define IN1 9
#define IN2 8
#define IN3 7
#define IN4 6
#define ENB 5 // ENB precisa estar em uma porta PWM

/* Criamos um objeto de controle dos motores. */
L298NX2 motores(ENA, IN1, IN2, ENB, IN3, IN4);
```

```

void setup() {
  /* Inicia com os motores parados.          */
  motores.stop();
}

void loop() {
  /* Define a velocidade em 180 (0-255).      */
  motores.setSpeed(180);
  /* Chama a função frente por 1000 milissegundos */
  frente(1000);
  /* Pausa de meio segundo                    */
  delay(500);
  /* Chama a função ré por 1000 milissegundos  */
  re(1000);
  /* Pausa de meio segundo                    */
  delay(500);
  /* Chama a função girar em torno do centro no sentido*/
  /* antihorário por 1000 milissegundos          */
  giro_centro_antihorario(1000);
  /* Pausa de meio segundo                    */
  delay(500);
  /* Chama a função girar em torno do centro no sentido*/
  /* horário por 1000 milissegundos              */
  giro_centro_horario(1000);
  /* Pausa de meio segundo                    */
  delay(500);
  /* Chama a função girar em torno da roda no sentido */
  /* antihorário por 1000 milissegundos          */
  giro_roda_antihorario(1000);
  /* Pausa de meio segundo                    */
  delay(500);
  /* Chama a função girar em torno da roda no sentido */
  /* horário por 1000 milissegundos              */
  giro_roda_horario(1000);
  /* Pausa de cinco segundos                  */
  delay(5000);
}

/* Função que move para frente pelo tempo definido */
void frente(int tempo)

```

```
{
  motores.forward();
  delay(tempo);
  motores.stop();
}
/* Função que move para trás pelo tempo definido */
void re(int tempo)
{
  motores.backward();
  delay(tempo);
  motores.stop();
}
/* Função que gira em torno do centro no sentido */
/* antihorário pelo tempo definido */
void giro_centro_antihorario(int tempo)
{
  motores.backwardA();
  motores.forwardB();
  delay(tempo);
  motores.stop();
}
/* Função que gira em torno do centro no sentido */
/* horário pelo tempo definido */
void giro_centro_horario(int tempo)
{
  motores.backwardB();
  motores.forwardA();
  delay(tempo);
  motores.stop();
}
/* Função que gira em torno da roda no sentido */
/* antihorário pelo tempo definido */
void giro_roda_antihorario(int tempo)
{
  motores.forwardB();
  delay(tempo);
  motores.stop();
}
```

```
/* Função que gira em torno da roda no sentido          */
/* horário pelo tempo definido                          */
void giro_roda_horario(int tempo)
{
  motores.forwardA();
  delay(tempo);
  motores.stop();
}
```

Com o código-fonte inserido no Arduino IDE, compile o programa pressionando o botão **Verify(Verificar)** (botão com sinal de tique) para verificar se não há erros de sintaxe. Estando o código correto, o próximo passo é realizar a transferência do programa para o Arduino. Para tal, pressione o botão **Upload (Carregar)** (botão com uma seta apontando para a direita).

Após a transferência do programa para o Arduino, o robô executará uma série de movimentos programados, dentre eles: mover para frente, mover para trás, girar nos sentidos horário e anti-horário e parar.

#### b. Linguagem de programação por blocos

Outra forma de programar o robô é por meio da linguagem de programação que utiliza blocos de funções prontas, os quais representam comandos de programação. Vamos utilizar o software mBlock.

Para conectar o mBlock ao Arduino, você deve clicar no ícone **Adicionar**, localizado no campo **Dispositivos**, e selecionar o Arduino, na biblioteca de dispositivos do mBlock, clicando, na sequência, no botão **OK**.

Uma vez selecionado, o Arduino Uno é visualizado no campo **Dispositivos** do mBlock e já é possível iniciar a programação em blocos.

Nessa programação, utilizaremos a extensão **RP - Driver L298N**, que auxiliará na estrutura do nosso programa. Para lembrar como instalar uma extensão no mBlock, consulte a **Aula 05 - Softwares Arduino IDE e mBlock**.

Monte os blocos, arrastando e soltando, de acordo com a programação dos movimentos do robô (figura 5).



Figura 5 - Programação em blocos para controle do robô



Assim que os blocos estiverem montados, clique no botão **Conectar** para iniciar a comunicação entre o software mBlock com a placa de Arduino Uno. Ao clicar sobre o botão **Conectar**, aparecerá um *tooltip* solicitando a confirmação da conexão entre os dois dispositivos.

Uma vez realizada a conexão entre os dispositivos, será ativado, na interface do mBlock, o botão **Upload**, o qual ao ser clicado, o software verificará se não há erros na estrutura do programa e então compilará para enviar o programa à placa Arduino.

Com a transferência do código para o dispositivo Arduino Uno, o robô executará movimentos, conforme programação.

## Desafios

Faça as alterações necessárias para seu robô se mover numa trajetória quadrada pelo chão.

Que tal introduzir um sensor em seu robô para que ele pare de se mover ao se deparar com um obstáculo?

## E se... ?

O projeto não funcionar como o esperado?

- a.** Confira se a ponte H foi conectada corretamente com a placa Arduino;
- b.** Caso um ou os dois motores estejam girando no sentido oposto ao programado, inverta a ordem dos fios na ponte H, assim a rotação será invertida e adequará aos comandos programados.



### 3. Feedback e finalização

**a.** Confira, compartilhando seu projeto com os demais colegas, se o objetivo foi alcançado.

**b.** Analise seu projeto desenvolvido, de modo a atender aos requisitos para o funcionamento do robô.

**c.** Reflita se as seguintes situações ocorreram:

i. Colaboração e cooperação: você e os membros de sua equipe interagiram entre si, compartilhando ideias que promoveram a aprendizagem e o desenvolvimento deste projeto?

ii. Pensamento crítico e resolução de problemas: você conseguiu identificar os problemas, analisar informações e tomar decisões de modo a contribuir para o projeto desenvolvido?

**d.** Reúna todos os componentes utilizados nesta aula e os organize novamente, junto aos demais, no kit de robótica.

### Referências

ARDUINO. Site oficial. **Ambiente de Programação do Arduino**. Disponível em: <https://create.arduino.cc/editor>. Acesso em: 15 out. 2021.

ARDUINO. Site oficial. **Downloads**. Disponível em: [www.arduino.cc/en/Main/Software](http://www.arduino.cc/en/Main/Software). Acesso em: 15 out. 2021.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular**. Brasília, 2018. Disponível em: [http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518-versaofinal\\_site.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518-versaofinal_site.pdf). Acesso em: 18 out. 2021.

MAKEBLOCK. mBlock. **Download mBlock**. Disponível em: <https://mblock.makeblock.com/en-us/download/>. Acesso em: 15 out. 2021.

MAKEBLOCK. mBlock. **Programação em blocos**. Disponível em: <https://ide.mblock.cc/>. Acesso em: 15 out. 2021.

**DIRETORIA DE TECNOLOGIAS E INOVAÇÃO (DTI)**  
**COORDENAÇÃO DE TECNOLOGIAS EDUCACIONAIS (CTE)**

**EQUIPE ROBÓTICA PARANÁ**

Andrea da Silva Castagini Padilha  
Cleiton Rosa  
Darice Alessandra Deckmann Zanardini  
Edgar Cavalli Júnior  
Edna do Rocio Becker  
José Feuser Meurer  
Marcelo Gasparin  
Michele Serpe Fernandes  
Michelle dos Santos  
Orlando de Macedo Júnior  
Roberto Carlos Rodrigues

Os materiais, aulas e projetos da “Robótica Paraná” foram produzidos pela Coordenação de Tecnologias Educacionais (CTE), da Diretoria de Tecnologia e Inovação (DTI), da Secretaria de Estado da Educação do Paraná (SEED), com o objetivo de subsidiar as práticas docentes com os estudantes por meio da Robótica. Este material foi produzido para uso didático-pedagógico exclusivo em sala de aula.



Este trabalho está licenciado com uma Licença  
Creative Commons – CC BY-NC-SA  
[Atribuição - NãoComercial - Compartilha Igual 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/)

