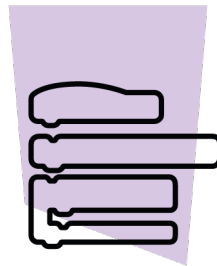
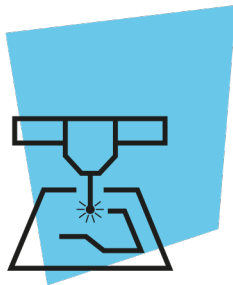
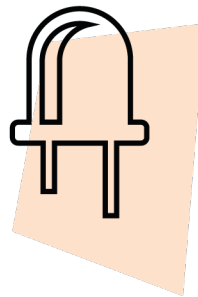
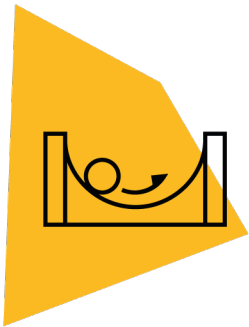
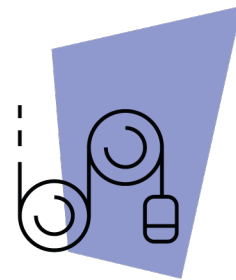


Diretoria de Tecnologia e Inovação

# ROBÓTICA



Módulo 1



Trena Digital  
Display Gráfico OLED\*

AULA 37

\*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

Michelle Santos

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**

# 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	6
3. Feedback e finalização	18
Referências	18



### Introdução

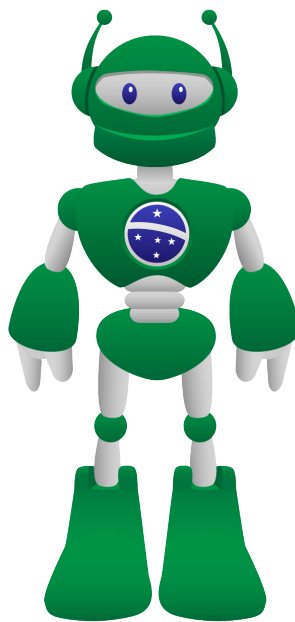
Desde a Antiguidade, a espécie humana vem elaborando instrumentos de medição para fins comerciais, de construção civil, entre outros. As primeiras unidades de medidas eram baseadas no corpo humano, como, por exemplo, polegar, pés, palmos. Com o desenvolvimento das tecnologias, novos instrumentos para se medir com maior precisão foram e estão sendo criados.

Nesta aula, conheceremos, por meio de protótipo, o funcionamento de uma trena digital.



### Objetivos desta aula

- Prototipar uma trena digital a partir de componentes eletrônicos presentes no kit de robótica;
- Rever os funcionamentos do sensor de distância ultrassônico e display OLED.





## 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 protoboard
- 1 placa Arduino Uno R3
- 1 cabo USB
- 9 jumpers macho-macho
- 1 sensor ultrassônico
- 1 display gráfico OLED
- 1 notebook
- Software mBlock ou Arduino IDE



## Roteiro da aula

### 1. Contextualização (15min)

Vamos iniciar esta aula comentando sobre o conceito de medição. O termo medição define um processo, no qual são estabelecidos um valor e uma unidade para determinada grandeza a ser medida. Nesse processo são utilizados dispositivos chamados de instrumentos de medição, que permitem padronizar objetos, como peças e produtos. Baseando-se neste conceito, quais instrumentos de medição você conhece? Você tem algum destes instrumentos em sua casa?

Podemos citar, como instrumentos de medição que presentes no nosso cotidiano: a régua, o termômetro, a trena e/ou fita métrica, a balança, entre outros. Dentre os instrumentos citados, você saberia dizer qual a função da trena?

A trena é um dispositivo de medição que possui vários modelos, como exemplo podemos citar a trena formada por uma fita metálica alojada no interior de uma caixa, dita corpo da trena, e com mecanismo que permite esticá-la e recolhê-la automaticamente. É utilizada para medir distâncias e conferir e comparar medidas de peças e áreas. Com o avanço das tecnologias, a trena convencional ganhou variações, como, por exemplo, a trena digital laser, a qual utiliza sensor ultrassônico para determinar a distância que se encontra o objeto, permitindo assim, uma leitura com alta precisão e em locais de difícil acesso (figura 1).

Figura 1 - Modelo de Trena Digital Laser

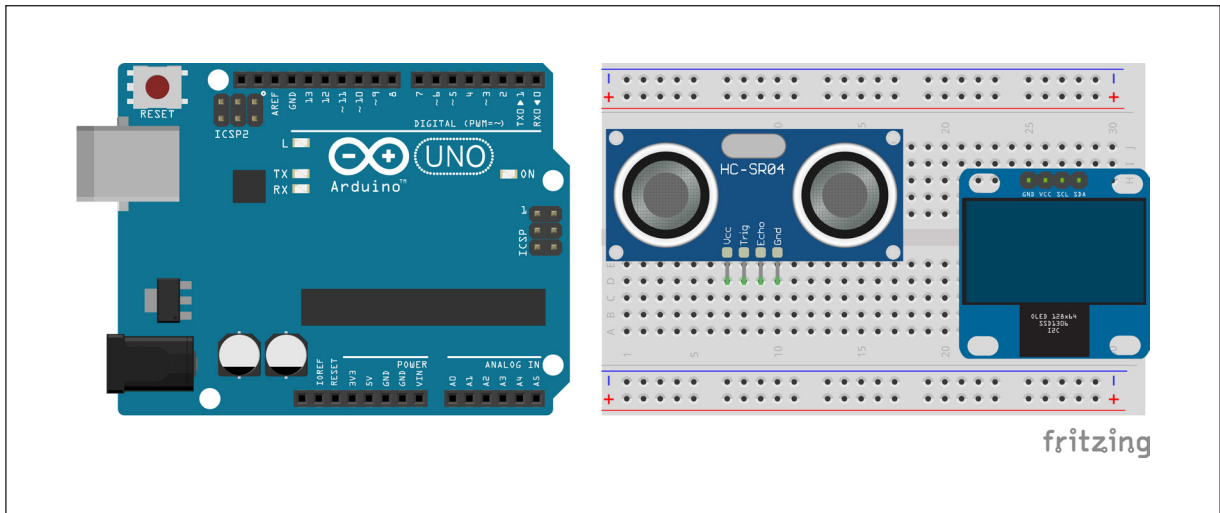


Nesta aula, vamos simular, com o uso de componentes eletrônicos presentes no kit de robótica, o funcionamento de uma trena digital. Para lembrar a função dos componentes utilizados, retorne às aulas: **Aula 34 - Sensor de Distância** e **Aula 36 - Display OLED 128x64**.

## 2. Montagem e programação (60 min)

Vamos começar com a montagem dos componentes eletrônicos. Encaixe na placa protoboard, o sensor ultrassônico, utilizando uma coluna de furos para cada pino do sensor e o display OLED 128x64 (figura 2).

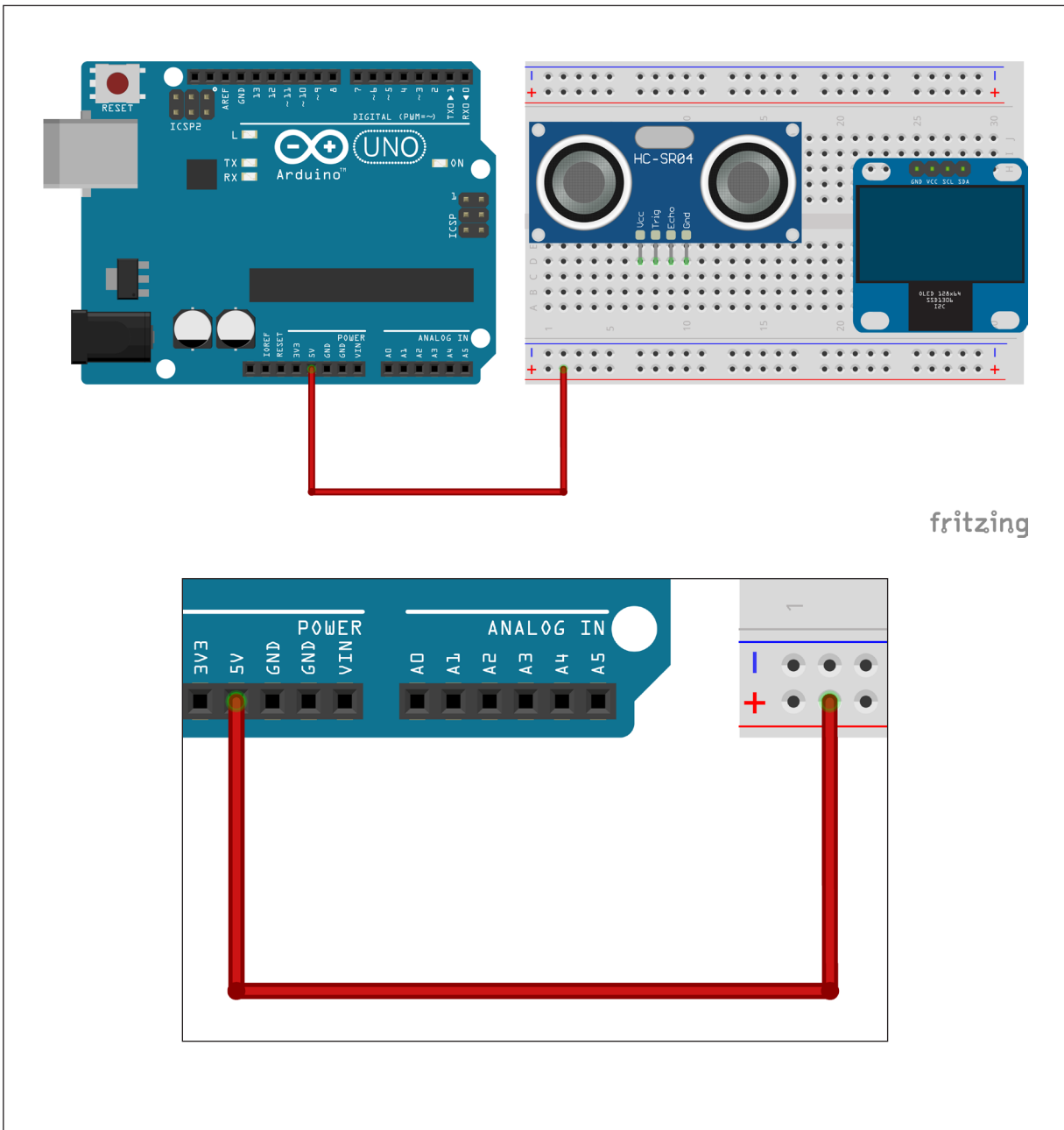
Figura 2 - Inserção dos componentes eletrônicos na protoboard





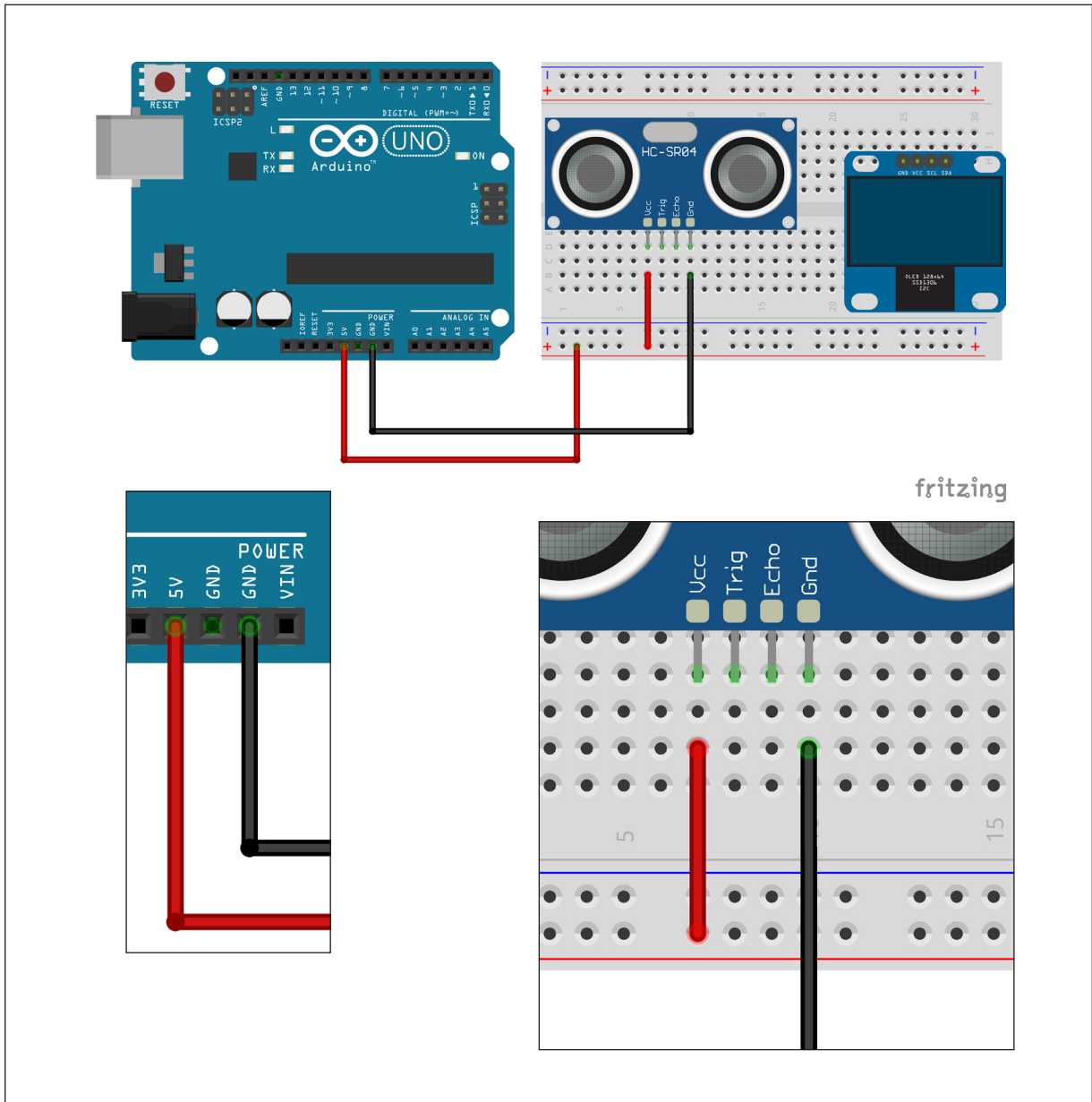
Conecte, com auxílio de um jumper macho-macho, o pino 5V da placa Arduino à linha lateral **VERMELHA** da protoboard (figura 3).

Figura 3 - Conexão do pino 5V do Arduino à protoboard.



Utilize dois jumpers para conectar os pinos VCC e GND do sensor ultrassônico à linha lateral **VERMELHA** da protoboard e um pino GND do Arduino, respectivamente (figura 4).

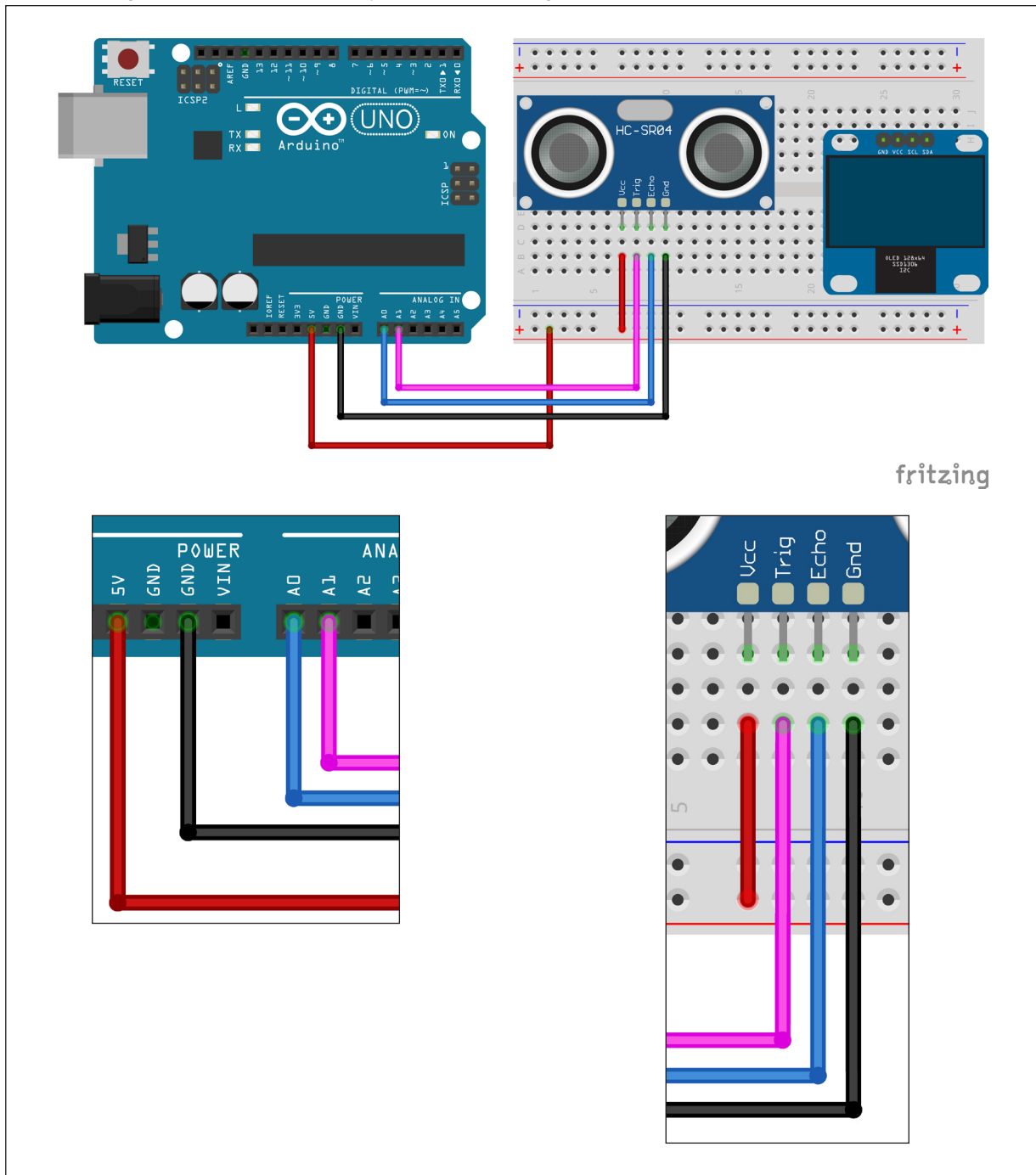
Figura 4 - Alimentando o sensor ultrassônico.



Utilize mais dois jumpers para conectar os pinos echo e trig, do sensor ultrassônico aos pinos A0 e A1 do Arduino, respectivamente (figura 5).

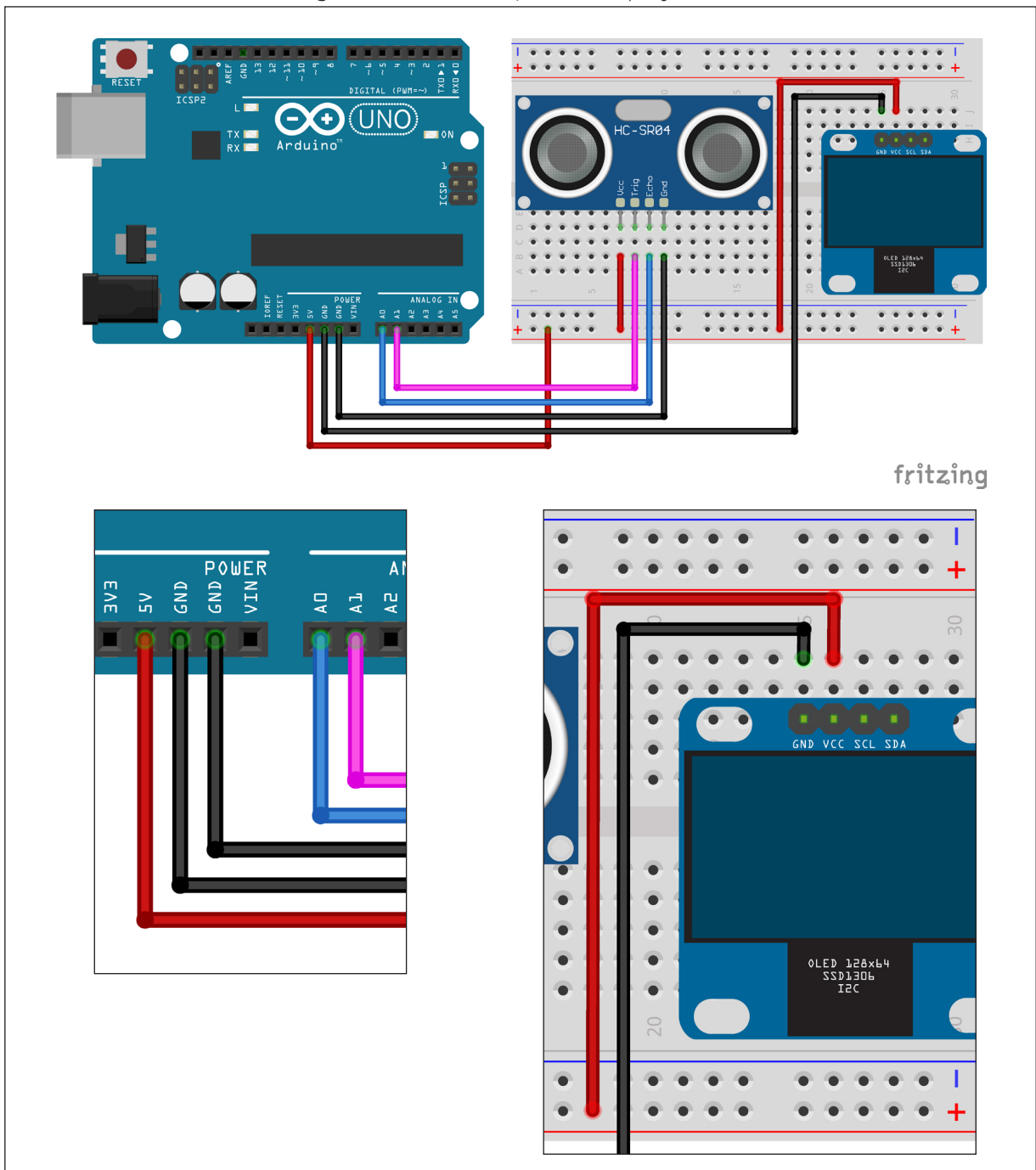
**OBS.:** De forma opcional podemos utilizar as portas analógicas do Arduino como portas digitais, sendo assim, necessitamos apenas informar quais portas do Arduino estamos utilizando para ambos os pinos do sensor (echo e trig).

Figura 5 - Conexão dos pinos echo e trig do sensor ultrassônico ao Arduino



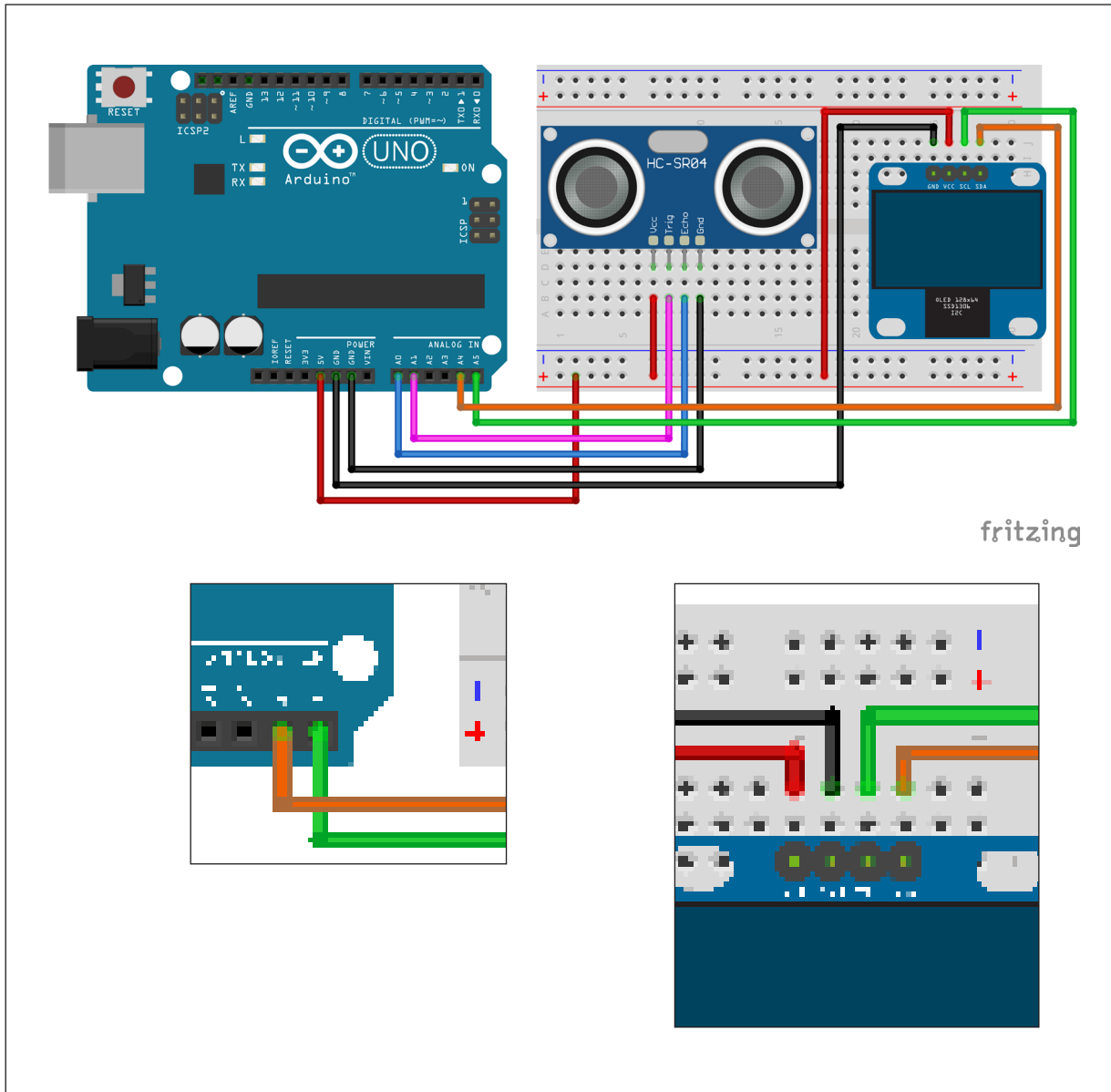
Conecte, com auxílio de dois jumpers, os pinos VCC e GND do display OLED à linha lateral **VERMELHA** da protoboard e um pino GND do Arduino, respectivamente (figura 6).

Figura 6 - Alimentação do display OLED



Finalize a montagem interligando os pinos SDA e SCL do display OLED aos pinos A4 e A5, respectivamente, do Arduino (figura 7).

Figura 7 - Conectando os pinos DAS e SCL do Display OLED ao Arduino



O funcionamento da trena digital consiste na emissão de sinais ultrassônicos pelo sensor e na leitura do sinal de retorno (reflexo/eco) desse mesmo sinal. A distância entre o sensor e o objeto que refletiu o sinal é calculada com base no tempo entre o envio e leitura de retorno e, então, envia ao display OLED, que fica encarregado de mostrar os valores das medições.

Com os componentes eletrônicos montados, vamos programar nosso protótipo, por codificação ou por blocos.

### a. Linguagem de programação por código

Para esta programação, utilizaremos as bibliotecas **Adafruit SSD1306** e **Adafruit GFX Library**, desenvolvidas pela Adafruit. Caso o Arduino IDE (versão Windows) indique que são necessárias outras bibliotecas para a utilização destas, clique na opção “instalar todas” (**Install all**), e também a biblioteca **Ultrasonic**, desenvolvida por Érick Simões.

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

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 37 - Trena Digital */  
/* Programação de uma trena digital que utiliza o */  
/* display OLED 128x64 I2C para exibir o valor da */  
/* medida obtida do sensor ultrassônico. */  
/* *****  
/* Baixe as bibliotecas através dos links abaixo: */  
/*  
/* http://librarymanager/All#minimalist#Ultrasonic */  
/* Biblioteca Ultrasonic by Erick Simões */  
/*  
/* http://librarymanager/All#SSD1306#Adafruit */  
/* Biblioteca: Adafruit SSD1306 by Adafruit */  
/*  
/* http://librarymanager/All#GFX#class#that */  
/* Biblioteca: Adafruit GFX Library by Adafruit */  
/*  
/* *****  
/* Inclusão das três bibliotecas necessárias. */  
#include <Adafruit_SSD1306.h>  
#include <Adafruit_GFX.h>  
#include <Ultrasonic.h>
```

```

/* Cria o objeto de controle para o display OLED. */
Adafruit_SSD1306 display(128, 64, &Wire, -1);

/* Define os pinos do sensor ultrassônico. */
#define pino_trig A1
#define pino_echo A0

/* Cria o objeto de controle para o untrassônico. */
Ultrasonic ultrasonic(pino_trig, pino_echo);
/* Cria a variável que armazenará a medida. */
float medida;

/* Configurações. */
void setup(void) {
  /* Inicia o display OLED, através de seu */
  /* endereço físico 0X3C. */
  display.begin(SSD1306_SWITCHCAPVCC, 0X3C);
}/* Fim das configurações. */

/* Looping infinito. */
void loop(void) {
  /* Faz a leitura da distância e armazena na */
  /* variável medida. */
  medida = ultrasonic.read();
  /* Limpa o display ao iniciar. */
  display.clearDisplay();
  /* Posiciona o cursor na coordenada informada. */
  display.setCursor(8, 0);
  /* Define o tamanho da fonte: 1. */
  display.setTextSize(1);
  /* Define a cor da fonte: Claro (1) - Escuro (0).*/
  display.setTextColor(1);
  /* Imprime a palavra -- TRENA DIGITAL --. */
  display.print("-- TRENA DIGITAL --");
  /* Desenha um retângulo sem preenchimento, com */
  /* origem na coordenada (0, 12), com dimensões */
  /* 128x40 pixels, raios dos cantos arredondados */
  /* 10 pixels na cor clara. */
  display.drawRoundRect(0, 12, 128, 40, 10, WHITE);
  /* Posiciona o cursor na coordenada informada. */
  display.setCursor(20, 55);
  /* Imprime a palavra Rob. */
  display.print("Rob");
}

```

```
/* Imprime a letra ó. */
display.write(162);
/* Imprime a palavra tica. */
display.print("tica Paran");
/* Imprime a letra á. */
display.write(160);
/* Define o tamanho da fonte: 2. */
display.setTextSize(2);
/* Se a medida for menor que 10, faça... */
if (medida < 10) {
  /* Posiciona o cursor na coordenada informada. */
  display.setCursor(25, 25);
  /* Imprime a medida. */
  display.print(medida);
  /* Posiciona o cursor na coordenada informada. */
  display.setCursor(85, 25);
  /* Imprime a unidade de medida. */
  display.print("cm");
}
/* Se a medida for maior ou igual a 10 e menor */
/* que 100, faça... */
if (medida >= 10 && medida < 100) {
  /* Posiciona o cursor na coordenada informada. */
  display.setCursor(13, 25);
  /* Imprime a medida. */
  display.print(medida);
  /* Posiciona o cursor na coordenada informada. */
  display.setCursor(85, 25);
  /* Imprime a unidade de medida. */
  display.print("cm");
}
```



```

/* Se a medida for maior ou igual a 100, faça... */
if (medida >= 100) {
  /* Posiciona o cursor na coordenada informada. */
  display.setCursor(25, 25);
  /* Imprime a medida. */
  display.print(medida / 100);
  /* Posiciona o cursor na coordenada informada. */
  display.setCursor(85, 25);
  /* Imprime a unidade de medida. */
  display.print("m");
}
/* Atualiza as informações do display. */
display.display();
/* Pequeno atraso para a leitura da medida. */
delay(100);
}/* Fim do looping infinito. */

```

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 protótipo da trena digital realizará a medição, conforme determinado na programação.

### b. Linguagem de programação por blocos

Outra forma de programar a leitura da trena digital 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 por blocos.

Nesta programação, utilizaremos variáveis que auxiliarão na estrutura do nosso programa e a extensão **RP - Display OLED**, (para recordar como criar uma variável e adicionar uma extensão, consulte a **Aula 05 - Softwares Arduino IDE e mBlock**).

Monte os blocos, arrastando e soltando, de acordo com a programação do projeto (figura 8).

Figura 8 - Programação em blocos para leitura da trena digital

```
quando o Arduino Uno começar
  Iniciar o display OLED 128x64 I2C com endereço 0X3C
  repetir para sempre
    definir medida para ler pin trigonométrico do sensor de ultrassom 15 pin de eco 14
    Limpa o display
    Posicionar o cursor em ( 9 , 1 )
    Tamanho da fonte 1
    Define a cor da fonte: Claro
    Imprime no display -- TRENA DIGITAL -- com quebra de linha automática: Não
    Retângulo vazio com cantos arredondados: Início em ( 1 , 13 ), dimensões ( 128 , 41 ) e raio das bordas: 10
    Posicionar o cursor em ( 21 , 56 )
    Imprime no display Rob com quebra de linha automática: Não
    Caracter ASCII ( 0 a 255 ): 162
    Imprime no display tica Paran com quebra de linha automática: Não
    Caracter ASCII ( 0 a 255 ): 160
    Tamanho da fonte 2
    se medida < 10 então
      Posicionar o cursor em ( 25 , 25 )
      Imprime no display medida com quebra de linha automática: Não
      Posicionar o cursor em ( 86 , 26 )
      Imprime no display cm com quebra de linha automática: Não
    se medida > 10 ou medida = 10 e medida < 100 então
      Posicionar o cursor em ( 14 , 26 )
      Imprime no display medida com quebra de linha automática: Não
      Posicionar o cursor em ( 86 , 26 )
      Imprime no display cm com quebra de linha automática: Não
    se medida > 100 ou medida = 100 então
      Posicionar o cursor em ( 26 , 26 )
      Imprime no display medida / 100 com quebra de linha automática: Não
      Posicionar o cursor em ( 86 , 26 )
      Imprime no display m com quebra de linha automática: Não
    Atualiza o display
  esperar 0.1 segundo(s)
```

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 irá verificar 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 protótipo da trena digital realizará a medição, de acordo com o determinado na programação por blocos.



### Desafios:

Acrescente ao circuito botões *Push Button* (estudado na **Aula 13 - Push Button**), um para a ativação da leitura do sensor, e outro encarregado de zerar todas as medidas realizadas, caso o usuário não esteja satisfeito com os valores encontrados.

Há possibilidade, ainda, de ter a opção de realizar cálculos como área e perímetro. Para tal, basta adicionar uma chave com a função de mudar a operação da trena, seja para cálculo de área ou para cálculo de perímetro.



### E se... ?

O projeto não funcionar?

1. Verifique se os jumpers estão nos pinos certos, se estão na mesma coluna dos terminais dos componentes, fazendo assim a conexão correta;
2. Verifique se a programação está adequada a cada porta digital.

### 3. Feedback e finalização (15 min)

- a. Confira, compartilhando seu projeto com os demais colegas, se o objetivo foi alcançado.
- b. Analise seu projeto desenvolvido, de modo a atender os requisitos para o funcionamento de uma trena digital.
- 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 (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 - Compartilhalgual 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/)