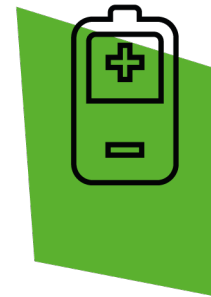
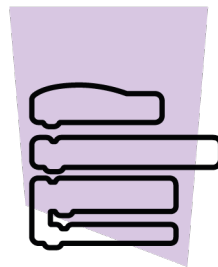
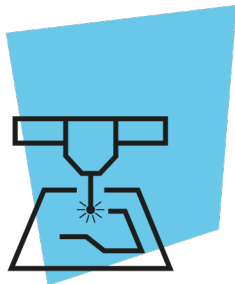
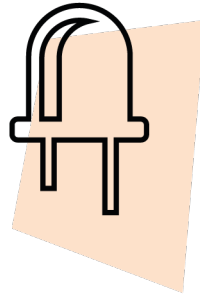
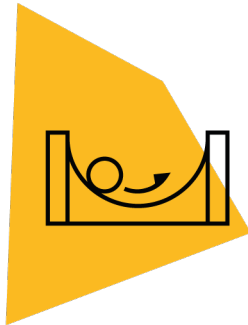
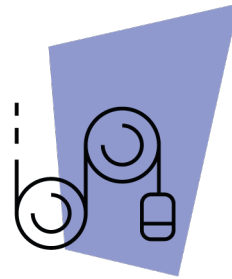
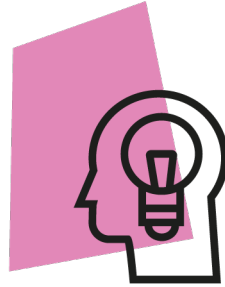


ROBÓTICA

Módulo 2



Módulo
nRF24L01 - II

AULA 33

*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

Orlando de Macedo Junior

Validação de Conteúdo

Cleiton Rosa

Revisão Textual

Orlando de Macedo Junior

Projeto Gráfico e Diagramação

Edna do Rocio Becker

Ilustração das Imagens

Edna do Rocio Becker

Cleiton Rosa

Roberto Carlos Rodrigues

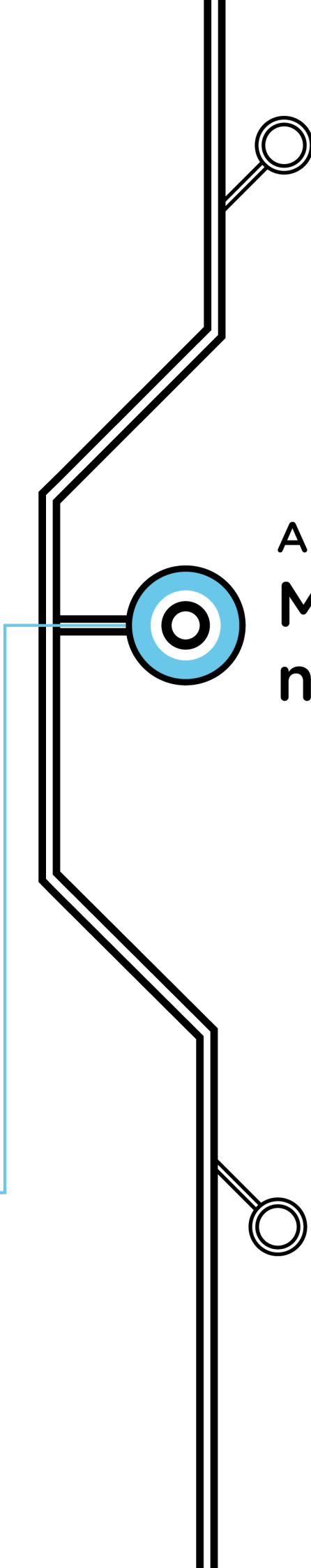
2023

- Aula 01 O que já vimos?
- Aula 02 Arduino: Bibliotecas e Funções
- Aula 03 Código Morse
- Aula 04 Semáforo Inteligente com IR
- Aula 05 Semáforo Completo com Display
- Aula 06 Matriz de LED 8X8
- Aula 07 Desenhando na matriz de LEDs
- Aula 08 Painel de Senhas
- Aula 09 Escrevendo mensagens
- Aula 10 Robô Autônomo
- Aula 11 Sensor de Chuva
- Aula 12 Sensor de Umidade do Solo
- Aula 13 Irrigador Automático
- Aula 14 Feedbacks + Inventário I
- Aula 15 Teclado Matricial de Membrana
- Aula 16 Servos Motores
- Aula 17 Fechadura Eletrônica
- Aula 18 Controlando Servos Motores
- Aula 19 JoyStick Shield
- Aula 20 Braço Robótico
- Aula 21 Sensor de Movimento Presença
- Aula 22 Sensor de Som
- Aula 23 Sensor de Umidade e Temperatura
- Aula 24 Termômetro Digital
- Aula 25 Sensor de Gás e Fumaça
- Aula 26 Acelerômetro e Giroscópio
- Aula 27 Motor de Passo
- Aula 28 Feedbacks + Inventário II
- Aula 29 Receptor IR e Controle Remoto
- Aula 30 Relé
- Aula 31 Módulo RF 433mhz - I
- Aula 32 Módulo RF 433mhz - II
- Aula 33 Projeto CHAT via RF
- Aula 34 Módulo Wirelless
- Aula 35 Comunicação do Módulo WI-FI em HTML
- Aula 36 Módulo WI-FI - IoT com Sensores
- Aula 37 Módulo WI-FI - IoT com Atuadores (LED)
- Aula 38 Módulo WI-FI - IoT com Atuadores (Relé)
- Aula 39 Monitor de Sensores em HTML I
- Aula 40 Monitor de Sensores em HTML II
- Aula 41 Mostra de Robótica
- Aula 42 Feedbacks + Inventário III

Aula 32
Módulo RF
nRF24L01 - I

Aula 33
Módulo
nRF24L01 - II

Aula 34
Projeto CHAT via RF



Sumário

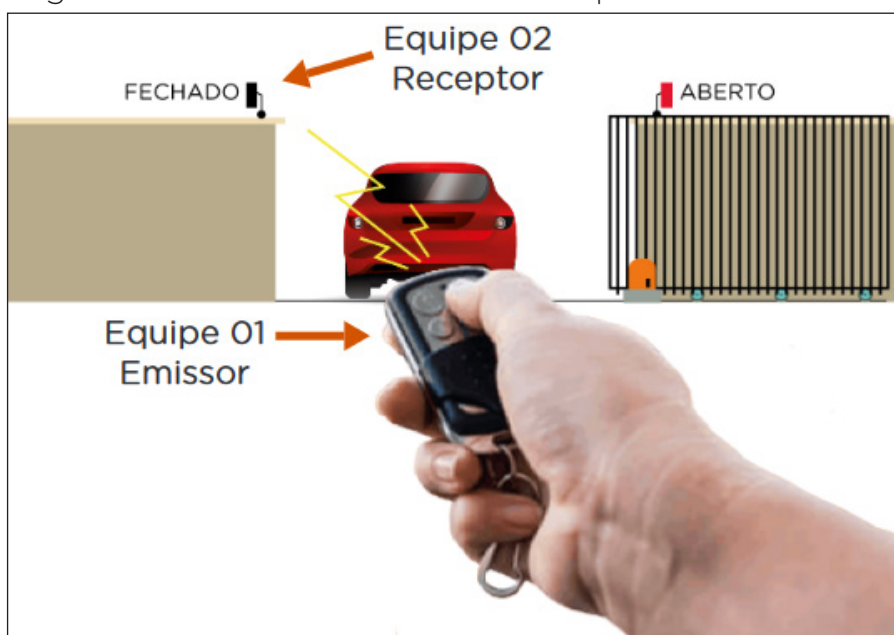
Introdução	2
Objetivos desta aula	3
Competências gerais previstas na BNCC	4
Habilidades do século XXI a serem desenvolvidas	5
Lista de materiais	6
Montagem e programação do Transmissor - Equipe 01	7
Montagem e Programação do Receptor - Equipe 02	12
Feedback e finalização	19



Introdução

As ondas de rádio, conforme você viu na aula anterior, é um tipo de radiação eletromagnética que viaja na velocidade da luz no vácuo. Muitos equipamentos fazem o uso destas ondas porque as informações são rapidamente projetadas e enviadas. Você viu também que elas são utilizadas em vários equipamentos na atualidade e teve contato com o módulo nRF24L01 do kit de robótica. Outro exemplo comum de uso das ondas de rádio são os controles de portões eletrônicos, presentes em quase todas as casas da atualidade.

Figura 1 - Uso de ondas de rádio em portões eletrônicos



Nesta aula, continuaremos o trabalho com esse módulo e vocês farão a montagem e a programação de um protótipo para acionar de um LED usando o módulo nRF24L01 e o Arduino.

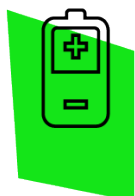
Para a montagem do protótipo e a execução da comunicação entre o transmissor e o receptor serão necessárias duas equipes.



Objetivos desta aula

- Entender o funcionamento de radiocomunicadores;
- Conhecer o módulo nRF24L01 presente no kit;
- Realizar a conexão com o Arduino;
- Realizar a instalação de biblioteca RF24;
- Retomar a utilização de LED;
- Criar o protótipo de um transmissor/receptor para controlar um LED;
- Identificar potencialidades para este módulo na robótica.





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;
- Comunicação.





Lista de Materiais

Equipe 1

- 1 placa Arduino Uno R3;
- 1 cabos USB;
- 1 placas protoboard;
- 1 módulo nRF24L01;
- 1 *push button*;
- 2 jumpers macho-macho;
- 7 jumpers macho-fêmea;
- 1 notebook com software Arduino IDE instalado.

Equipe 2

- 1 placa Arduino Uno R3;
- 1 cabos USB;
- 1 placas protoboard;
- 1 módulo nRF24L01;
- 1 LED 5mm;
- 1 resistor 220 Ω ;
- 2 jumpers macho-macho;
- 7 jumpers macho-fêmea;
- 1 notebooks com software Arduino IDE instalado.

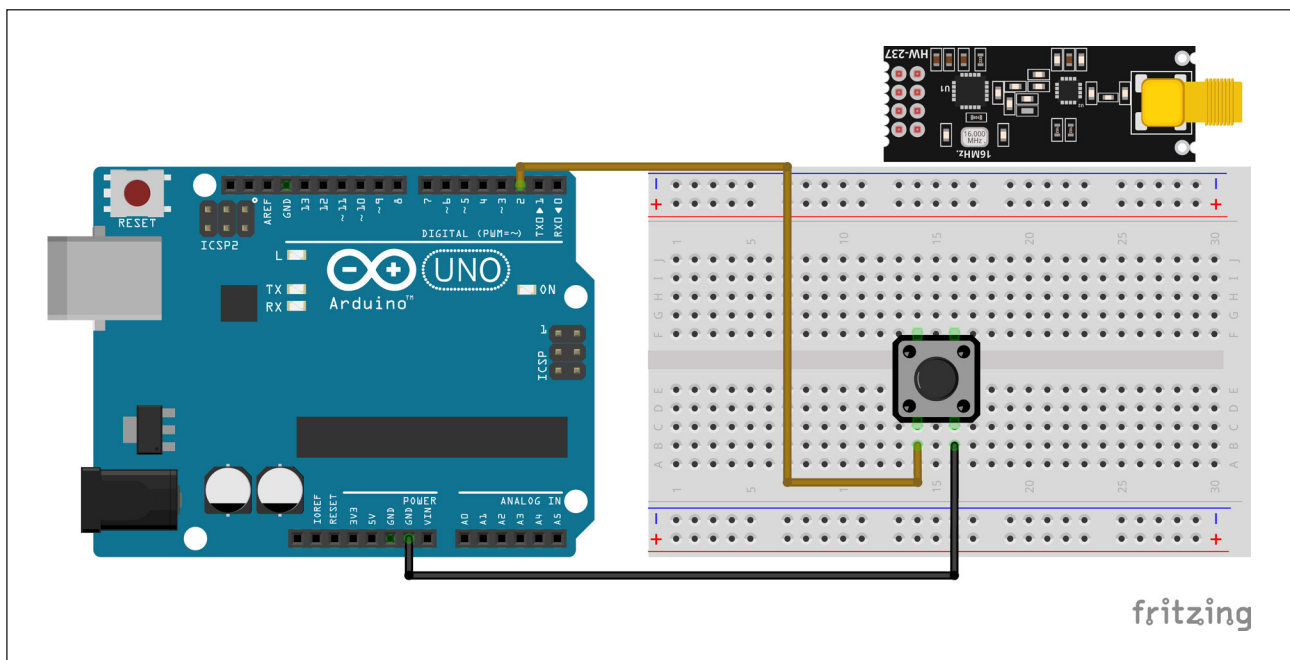


1. Montagem e programação (transmissor)

Equipe 1

Para iniciarmos a montagem do protótipo do transmissor, encaixe o *push button* na placa protoboard. Utilizando 2 jumpers macho-macho, interligue as portas GND e digital 2 do Arduino aos dois terminais do *push button* (figura 2).

Figura 2 - Encaixe do push botton e jumpers na protoboard



Fonte: Fritzing

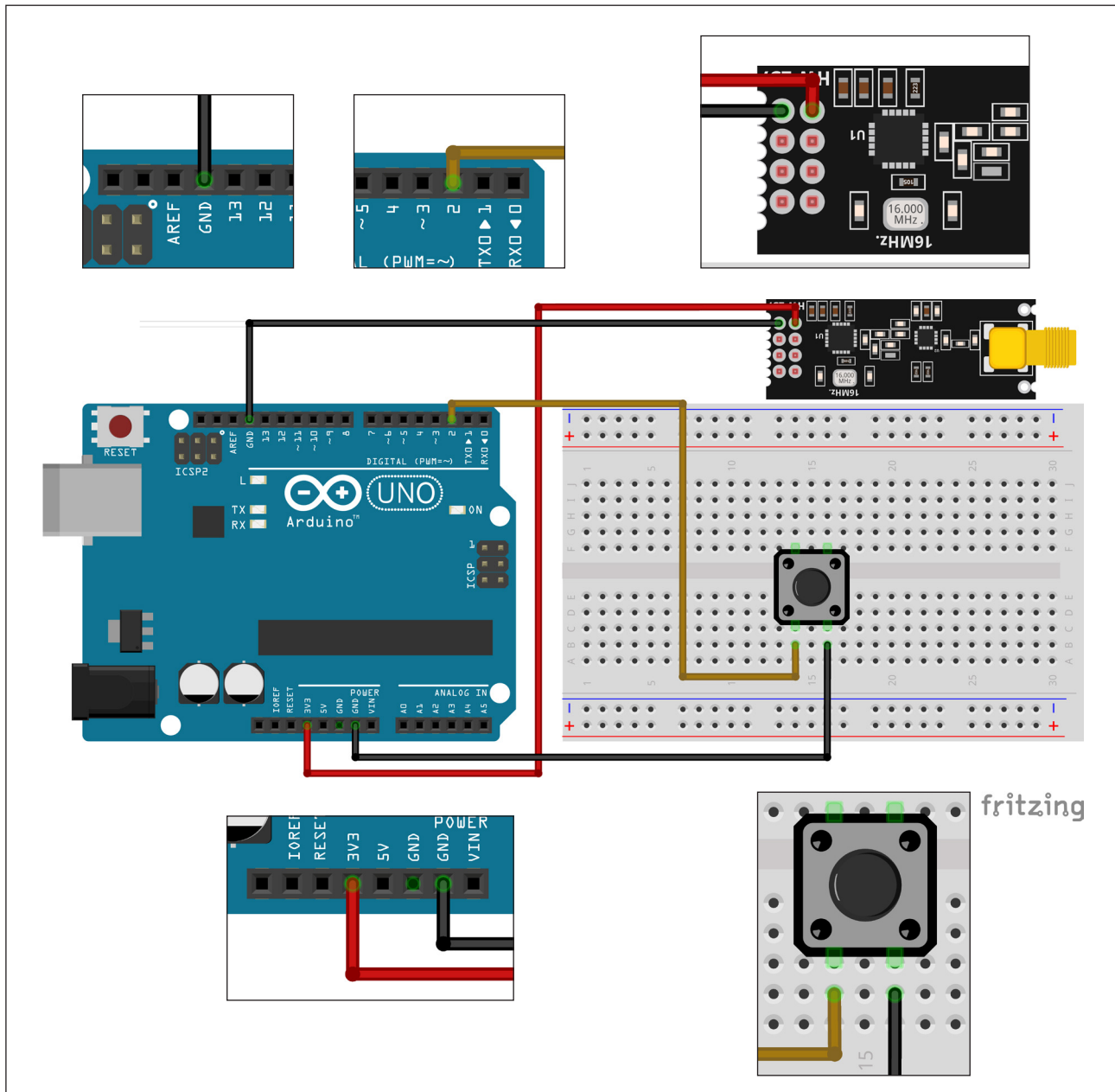
Em seguida, é a vez de ligar o módulo nRF 24L01. Com 2 jumpers macho-fêmea, interligue as portas GND e 3.3V do Arduino aos pinos de alimentação no módulo nRF24L01 (figura 3).

ATENÇÃO:

O módulo só pode ser operado com tensão 3.3V (3.6V) no pino VCC.



Figura 3: Alimentação do módulo nRF 24L01



Fonte: Fritzing

Para finalizar, utilize 5 jumpers macho-fêmea para interligar os pinos do módulo nRF 24L01 nas portas 9, 10, 11, 12 e 13 (tabela 1). Verificar a identificação dos pinos do módulo. (figura 4) para concluir a montagem (figura 5).

Figura 4 - Identificação dos pinos

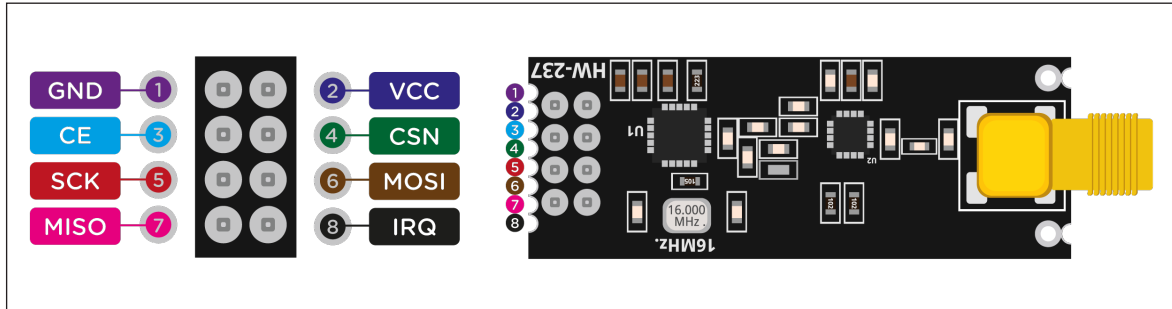
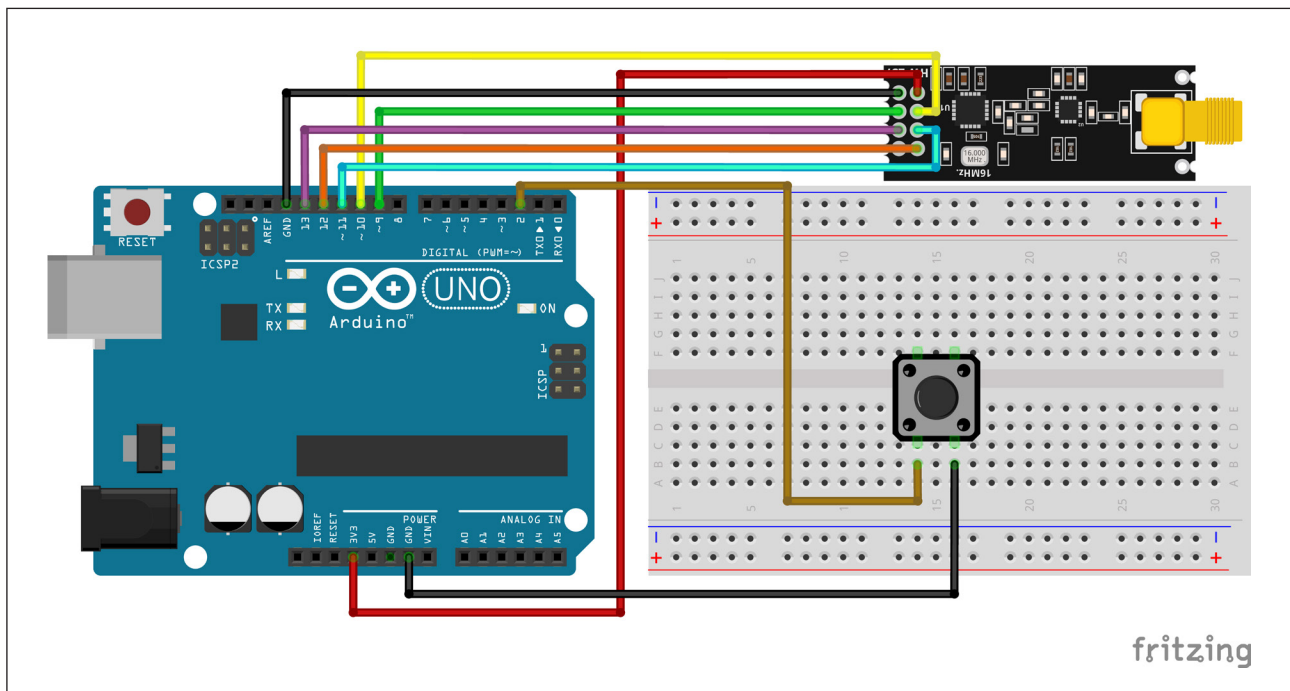


Tabela 1 - Relação entre os pinos do módulo e portas do Arduino

Pino do módulo	Porta
CE	9
CSN	10
SCK	13
MOSI	11
MISO	12

Figura 5 - Conclusão da montagem do transmissor



Fonte: Fritzing

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

Linguagem de programação por código

Para iniciar a programação, conecte a placa Arduino ao computador, com um cabo USB, para que ocorra a comunicação entre a placa microcontroladora e o software Arduino IDE.

Nessa programação, utilizaremos a biblioteca “**RF24**”, desenvolvida por TMRh20 Avamander, que auxiliará no controle do módulo nRF24L01. Essa biblioteca pode ser instalada a partir do repositório do Arduino IDE. (Para recordar como instalar uma biblioteca, consulte a **Aula 05 - Softwares Arduino IDE e mBlock** do módulo 1).

Neste projeto, você deverá entrar em contato com uma outra equipe que estará trabalhando com o protótipo de receptor para definir a palavra entre aspas (`#char mensagem [] = “pressionado”`) que servirá de comando para acionar o LED e os quatro números (`#define endereco 5995`) na programação que podem ser alterados conforme entendimento das equipes para que outros protótipos não interfiram na comunicação.

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 33 - Módulo nRF24L01 - TRANSMISSOR */  
/* Programação do Módulo nRF24L01 - Transmissor. O */  
/* código permite enviar uma informação (palavra: */  
/* pressionado) por meio do módulo nRF24L01 ao */  
/* pressionar o botão. */  
/* ****/  
/* Declara a variável mensagem do tipo char para */  
/* armazenar a palavra que será utilizada como comando* /  
/* para acionar o LED. */  
char mensagem[] = "pressionado";
```

```
/* Inclui a biblioteca de controle do módulo nRF24L01.*/
#include <RF24.h>
/* Define o pino de leitura do botão. */
#define pinoBotao 2
/* Define os pinos de comunicação do módulo nRF24L01. */
#define pinoCE 9
#define pinoCSN 10
/* Define o endereço de comunicação entre transmissor */
/* e receptor. */
#define endereco 5995
/* Cria o objeto de controle do módulo rádio. */
RF24 radio(pinoCE, pinoCSN);

void setup() {
/* Configura o pino do botão como entrada ativando o */
/* resistor interno. */
pinMode(pinoBotao, INPUT_PULLUP);
/* Inicializa o módulo rádio. */
radio.begin();
/* Abre canal de envio de dados. */
radio.openWritingPipe(endereco);
/* Desativa o recebimento de dados, pois somente irá */
/* enviar. */
radio.stopListening();
}

void loop() {
/* Se o botão for pressionado, faz... */
if (digitalRead(pinoBotao) == LOW) {
/* Envia a palavra para o receptor. */
radio.write(&mensagem, sizeof(mensagem));
/* Intervalo para nova leitura do botão. */
delay(400);
}
}
```

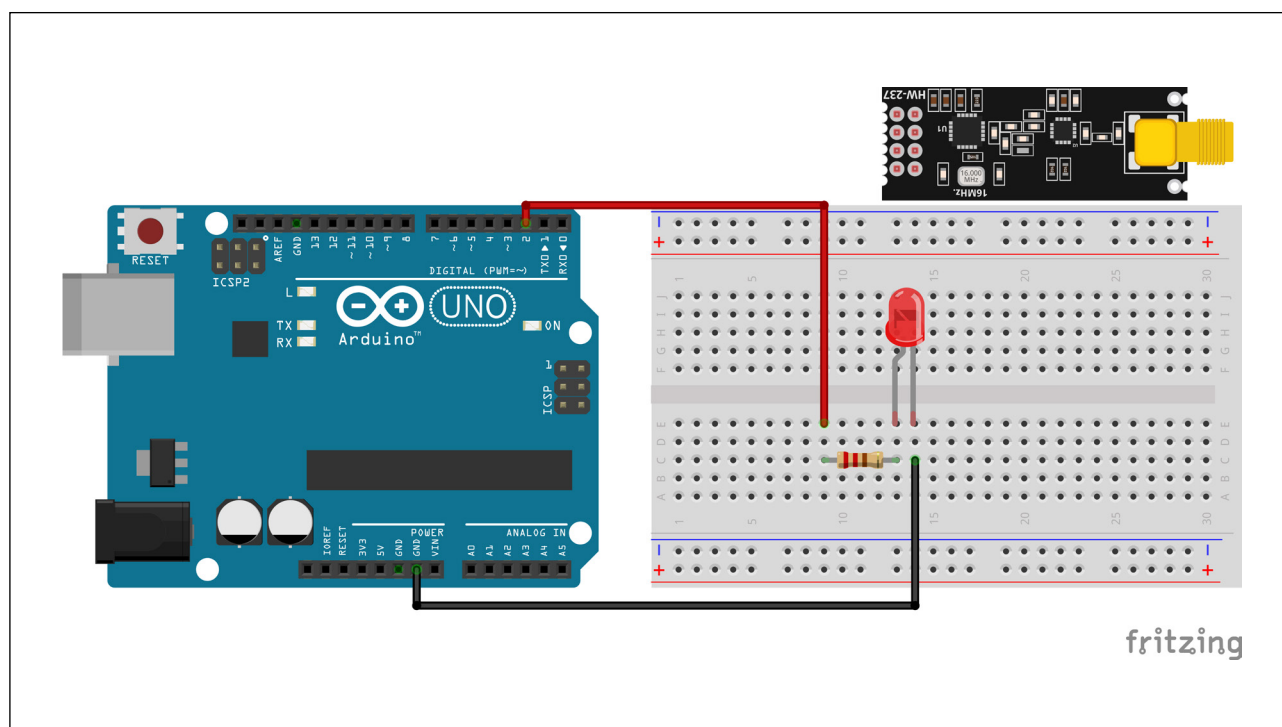
Com o código-fonte inserido no Arduino IDE, compile o programa pressionando o botão **Verificar** para averiguar se não há erros de sintaxe. Estando o código correto, pressione o botão **Carregar** para realizar a transferência do programa para o Arduino. Após a transferência do programa, você controlará o LED pelo protótipo transmissor, ligando ou desligando toda vez que o botão no transmissor for pressionado.

2. Montagem e programação (receptor)

Equipe 2

Para iniciarmos a montagem do protótipo do receptor, encaixe o LED e o resistor na placa protoboard e na sequência, utilizando 2 jumpers macho-macho, interligue as portas GND e digital 2 do Arduino ao terminal negativo do LED (terminal mais curto) e ao terminal livre do resistor, respectivamente (figura 6).

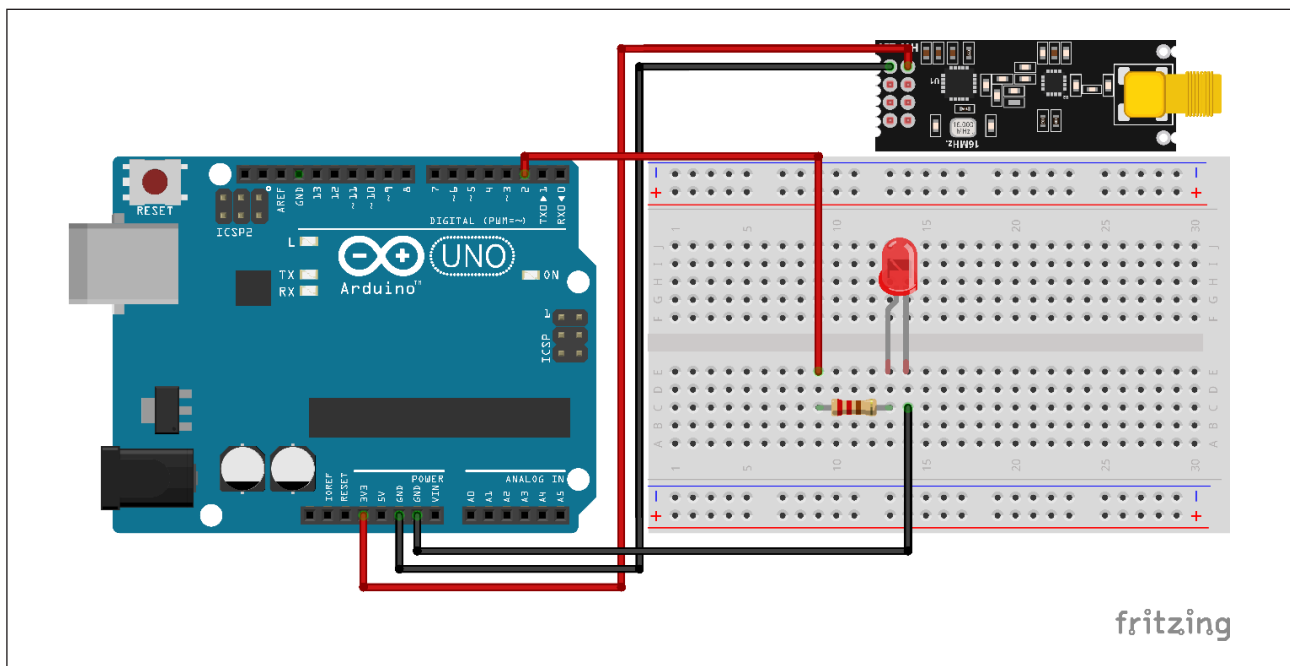
Figura 6 – Encaixe do LED e do resistor na protoboard



Fonte: Fritzing

Utilizando mais 2 jumpers macho-macho, interligue as portas 3V3 e GND do Arduino aos pinos de alimentação do módulo nRF24L01, conforme indicado na figura 7.

Figura 7 - Alimentação do módulo nRF24L01



Fonte: Fritzing

Para finalizar, utilize 5 jumpers macho-fêmea para interligar os pinos do módulo nRF24L01 nas portas 9, 10, 11, 12 e 13 (tabela 2). Verificar a identificação dos pinos do módulo (figura 8) para concluir a montagem (figura 9).

Figura 8 - Identificação dos pinos

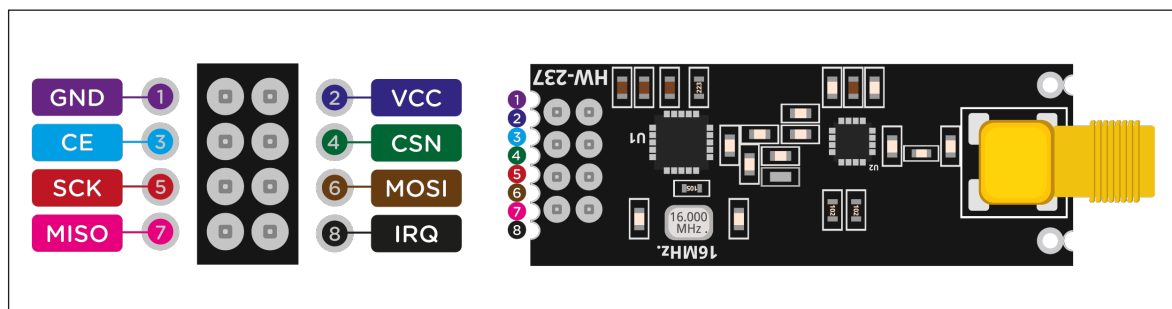
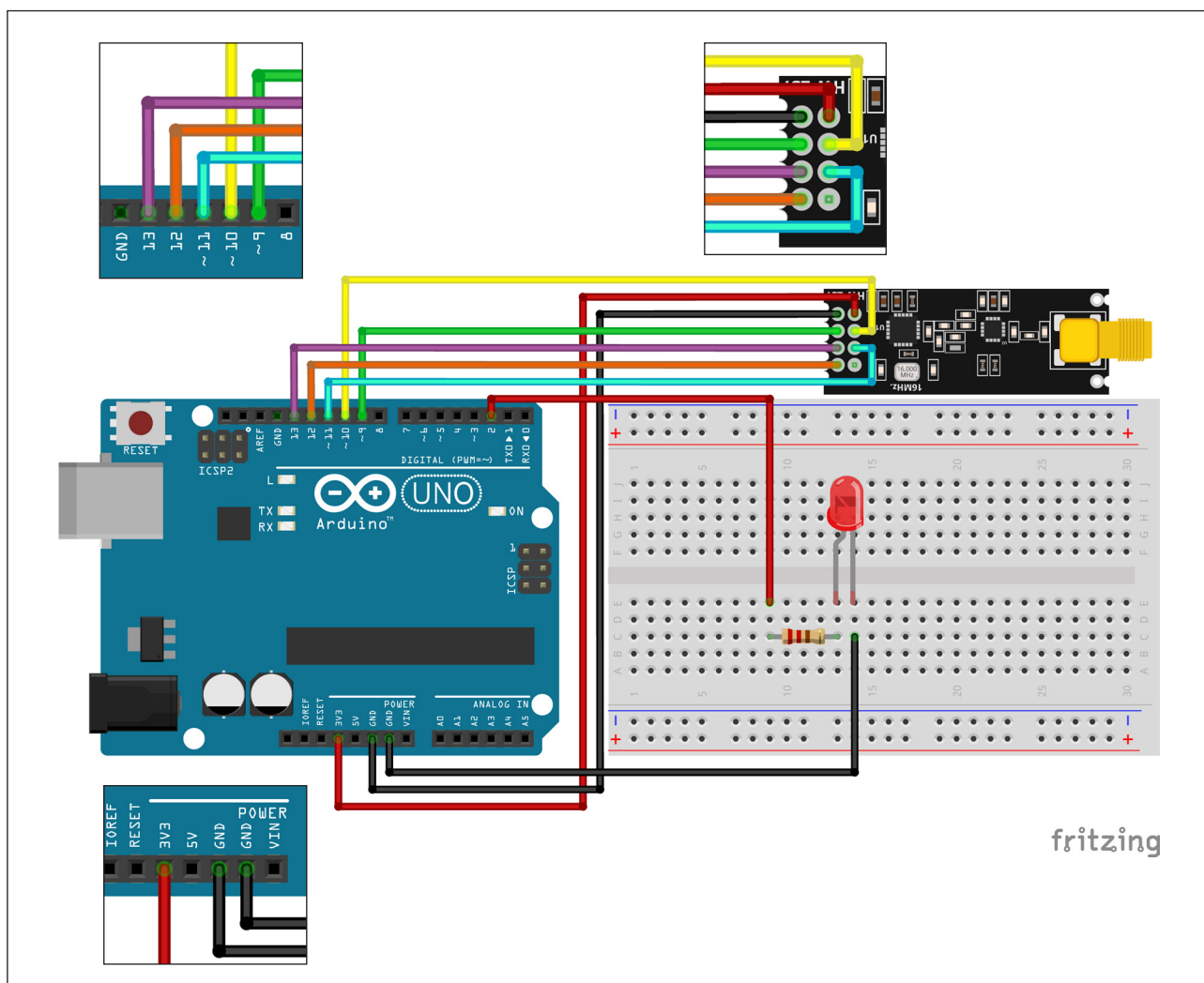


Tabela 2 - Relação entre os pinos do módulo e portas do Arduino

Pino do módulo	Porta
CE	9
CSN	10
SCK	13
MOSI	11
MISO	12

Figura 9 - Conclusão da montagem do receptor



Fonte: Fritzing

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

Linguagem de programação por código

Para iniciar a programação, conecte a placa Arduino ao computador, com um cabo USB, para que ocorra a comunicação entre a placa microcontroladora e o software Arduino IDE.

Nessa programação, utilizaremos a biblioteca “**RF24**” desenvolvida por TMRh20 Avamander, que auxiliará no controle do módulo nRF24L01. Esta biblioteca pode ser instalada a partir do repositório do Arduino IDE. (Para recordar como instalar uma biblioteca, consulte a **Aula 05 – Softwares Arduino IDE e mBlock do módulo 1**).

Neste projeto, você deverá entrar em contato com uma outra equipe que estará trabalhando com o protótipo do transmissor para definir a palavra entre aspas (String mensagem_esperada = “**pressionado**”) que servirá de comando para acionar o LED e os quatro números (#define endereço **5995**) na programação que podem ser alterados conforme entendimento das equipes para que outros protótipos não interfiram na comunicação.

No software IDE, escreva ou copie e cole o código-fonte de programação (quadro 2).

Quadro 2 - Código-fonte da programação na linguagem do Arduino (Wiring)

```
/* *****  
/* Aula 33 - Módulo nRF24L01 - RECEPTOR */  
/* Programação do Módulo nRF24L01 - Receptor. O */  
/* código permite que o receptor receba uma informa_ */  
/* ção (palavra: pressionado) para controlar o */  
/* acionamento do LED. */  
/* ***** *****/  
/* Define a palavra esperada do transmissor como */  
/* comando para acionar o LED. */  
String mensagem_esperada = "pressionado";  
/* Inclui a biblioteca de controle do módulo nRF24L01.*/  
#include <RF24.h>
```

```
/* Define o pino de controle do LED. */
#define pinoLED 2
/* Define os pinos de comunicação do módulo nRF24L01. */
#define pinoCE 9
#define pinoCSN 10
/* Define o endereço de comunicação entre transmissor */
/* e receptor. */
#define endereco 5995
/* Cria o objeto de controle do módulo rádio. */
RF24 radio(pinoCE, pinoCSN);
/* Variável que receberá a mensagem. */
char msg_recebida[32] = "";

void setup() {
  /* Inicia o monitor serial. */
  Serial.begin(9600);
  /* Configura o pino do LED como saída. */
  pinMode(pinoLED, OUTPUT);
  /* Inicializa o módulo rádio. */
  radio.begin();
  /* Abre canal de recebimento de dados no canal 0 (0-6).*/
  radio.openReadingPipe(0, endereco);
  /* Ativa o recebimento de dados. */
  radio.startListening();
}

void loop() {
  /* Se receber dados do transmissor, faça... */
  if (radio.available()) {
    /* Lê os dados recebidos e armazena na variável */
    /* msg_recebida. */
    radio.read(&msg_recebida, sizeof(msg_recebida));
    /* Imprime no monitor serial a mensagem. */
    Serial.println(msg_recebida);
    /* Se a mensagem recebida for igual a palavra */
    /* pressionado, faça... */
  }
}
```

```
if (String(msg_recebida) == mensagem_esperada) {  
  /* Altere o estado lógico do LED (Ligado/Desligado).*/  
  digitalWrite(pinoLED, !digitalRead(pinoLED));  
}  
}  
}
```

Com o código-fonte inserido no Arduino IDE, compile o programa pressionando o botão **Verificar** para averiguar se não há erros de sintaxe. Estando o código correto, pressione o botão **Carregar** para realizar a transferência do programa para o Arduino.

Após a transferência do programa, o seu LED será controlado pelo protótipo transmissor, ligando ou desligando, toda vez que o botão no transmissor for pressionado.





Desafio:

E se ampliarmos os horizontes, que tal ampliar o número de botões, talvez um potenciômetro? Assim, seria possível controlar, via rádio frequência, outros dispositivos eletrônicos: mais LEDs e um buzzer? Use a imaginação.



E se...?

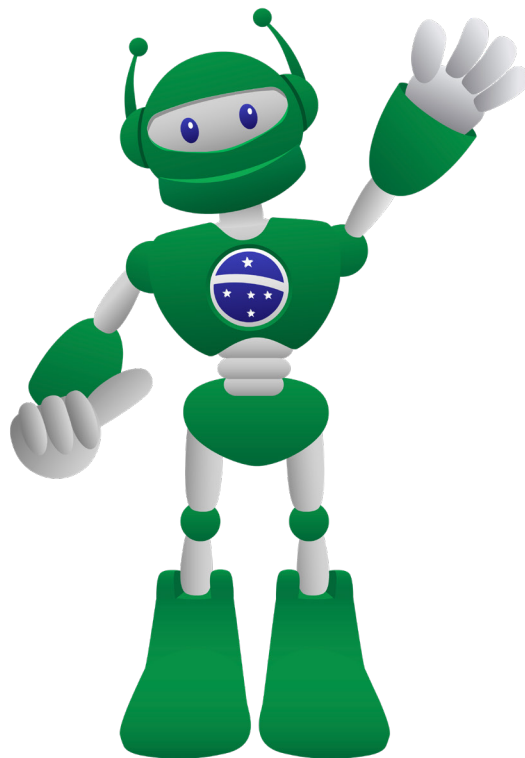
O projeto não funcionar, atente-se para possíveis erros:

- a. Verifique se a biblioteca está corretamente instalada no software Arduino IDE;
- b. Verifique se os jumpers estão conectados nos pinos certos do módulo;
- c. Verifique se os jumpers estão ligados corretamente nas portas do Arduino;
- d. Verifique se a programação está adequada a cada porta digital.



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 funcionamento do módulo nRF24L01.
- 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.



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/)

