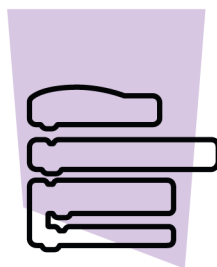
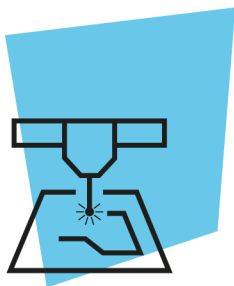
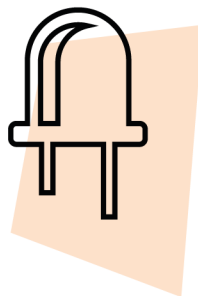
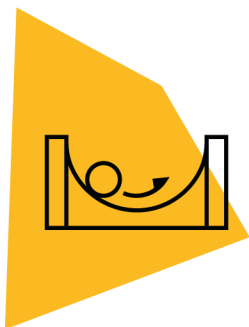
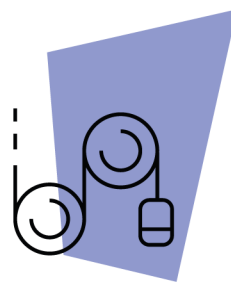


# ROBÓTICA

Módulo 2



## Projeto Chat via nRF24L01

AULA **34**

\*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

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 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 Wireless
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 33  
Módulo  
nRF24L01 - II

Aula 34  
Projeto CHAT  
via nRF24L01

Aula 35  
Módulo  
Wireless

# Sumário

<b>Introdução</b>	<b>2</b>
<b>Objetivos desta aula</b>	<b>2</b>
<b>Competências gerais previstas na BNCC</b>	<b>3</b>
<b>Habilidades do século XXI a serem desenvolvidas</b>	<b>4</b>
<b>Lista de materiais</b>	<b>4</b>
<b>Roteiro da aula</b>	<b>5</b>
1. Contextualização	5
2. Montagem e programação	8
3. Feedback e finalização	15
<b>Videotutorial</b>	<b>16</b>



## Introdução

O uso de aplicativos de mensagens para conversas tem se tornado cada vez mais comum, principalmente pelo uso de celulares. Isto se deve não apenas à facilidade, mas também à rapidez que estes dispositivos têm proporcionado na comunicação. Há várias maneiras possíveis de se ter esta funcionalidade, dentre elas temos a radiofrequência (RF).

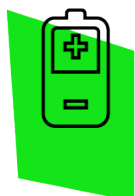
Montar uma rede de comunicação sem fio utilizando a radiofrequência (RF) é simples, barata e realiza a comunicação direta de dados entre dois diferentes componentes, isto é, o RF transmissor e RF receptor.

Nesta aula aprenderemos a programar um protótipo de chat utilizando o módulo nRF24L01 para se comunicar com outra equipe.



## Objetivos desta aula

- Retomar a utilização do módulo nRF24L01;
- Criar o protótipo de um chat via RF;
- Identificar potencialidades de uso desse 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.



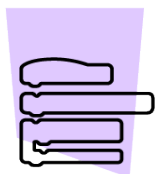
## Habilidades do século XXI a serem desenvolvidas

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



## Lista de materiais

- 1 placa arduino Uno R3;
- 1 cabo USB;
- 1 placa protoboard;
- 1 módulo nRF24L01;
- 7 Jumpers Macho-Macho;
- Notebook;
- Software Arduino IDE.



## Roteiro da aula

### 1. Contextualização

Você utiliza muito aplicativo de conversa para trocar mensagens com amigos, família e outras fontes? Tem se tornado tão comum que até para fazer compras e pedido de comida estão na lista de atividades corriqueiras do dia a dia. Já imaginou aprender a desenvolver um chat?



Na atualidade com a velocidade de troca de informação, fazem-se necessárias ferramentas de comunicação rápidas e instantâneas. E essa condição ainda vem crescendo com o uso de dispositivos que propiciam envio direto de mensagem.

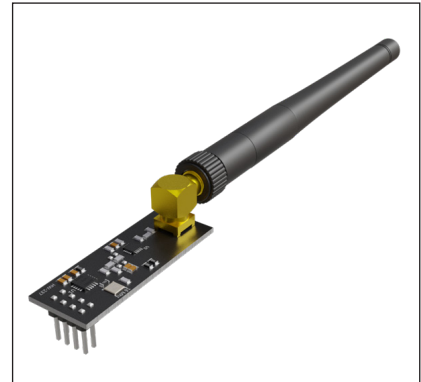
Como já estudamos na **Aula 32 - Módulo Rádio**, uma alternativa eficiente, simples e de baixo custo para transmissão de dados sem fio é o uso da radiofrequência, que é amplamente utilizada em sistemas eletrônicos.



Essa comunicação é classificada como bidirecional, ocorrendo entre dois pontos distintos, mas cujos dados podem ser enviados e recebidos por segundo), e, em tanto pelo transmissor quanto pelo receptor. O módulo nRF24L01 (figura 1) utiliza uma

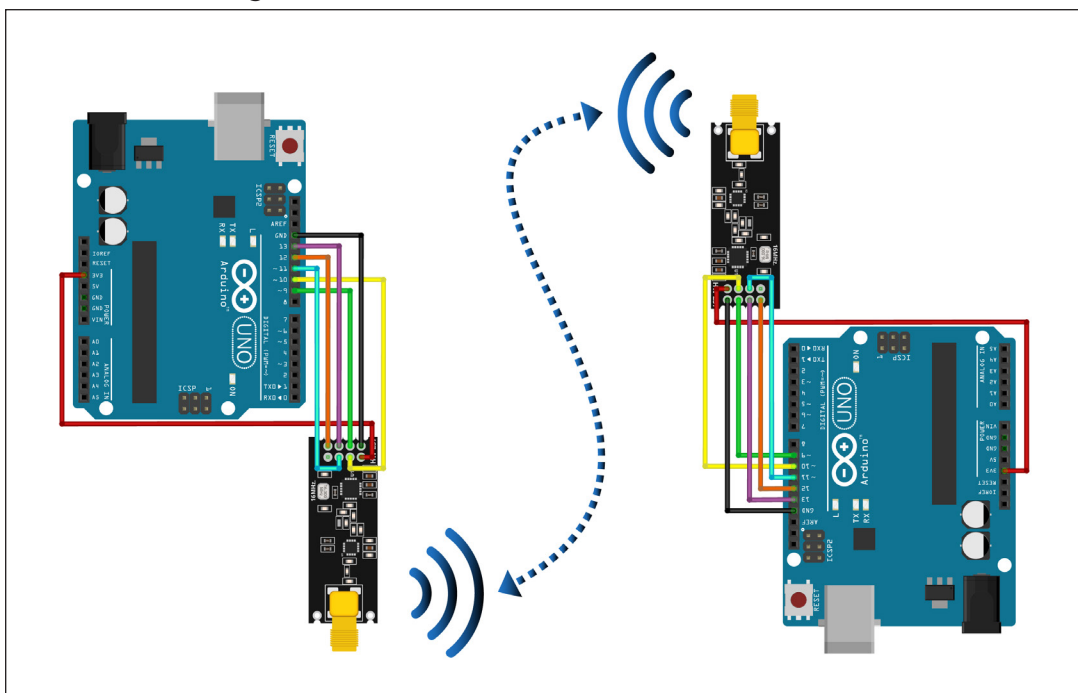
frequência de transmissão de 2400MHz ~ 2524MHz (2.4GHz), pode transmitir dados numa velocidade de até 2Mbps (megabit por segundo), e, em condições ideais, consegue ter um alcance de um pouco mais de mil metros (1,1 Km).

Figura 1 Módulo nRF24L01



Nesta aula você aprenderá a estabelecer comunicação via radio-frequência com outra equipe. Utilizaremos o módulo nRF24L01 nos dois protótipos, que, conectados junto ao Arduino e programados, poderão processar e encaminhar a informação em comandos de mensagens ao protótipo da outra equipe, e vice e versa, estabelecendo assim a transmissão de dados e a comunicação entre elas (figura 2).

Figura 2 - Funcionamento do chat na aula





## Para saber mais...

A reportagem “Conheça a história dos chats antes da popularização do MSN”, comenta sobre os programas que deram início às salas de bate-papo. Confira!



<https://memoria.ebc.com.br/tecnologia/2013/04/90s-os-chats-antes-do-msn-windows-live-messenger>

O artigo “O gênero textual ‘Chat’ e sua aplicabilidade no Ensino de História”, explica as características do chat educacional e as aplicabilidades no ensino. Confira!

<http://www.eeh2012.anpuh-rs.org.br/resources/pe/anais/encontro5/14-os-desafios/Artigo%20de%20Jos%E9%20Arimat%E9ia%20Santos.pdf>



## 2. Montagem e programação

Iniciaremos a montagem do protótipo conectando o módulo nRF24L01 diretamente à placa Arduino Uno R3, utilizando 2 jumpers macho-macho para alimentação do módulo (figura 3), interligue as portas 3,3V e GND do Arduino nos pinos VCC e GND do módulo conforme orientação (tabela 1).

**Lembre-se:** O módulo só pode ser operado com tensão 3,3V (3,6v).

Figura 3 - Alimentação do módulo nRF24L01 pela placa Arduino

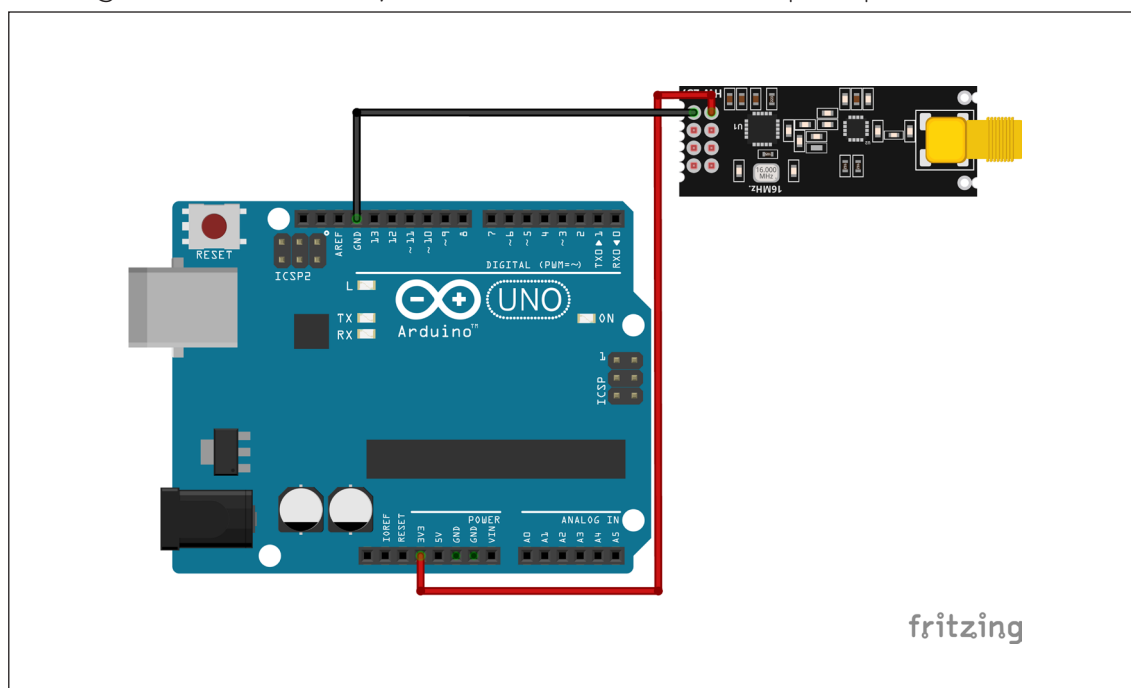


Tabela 1 - Relação entre os pinos do módulo e portas de alimentação do Arduino.

Pino do módulo	Porta
VCC	3,3V
GND	GND

Utilizando cinco jumpers, interligue as portas 9, 10, 11, 12 e 13 do Arduino com os pinos da barra do módulo (figura 4), respeitando a orientação (tabela 2).

Figura 4 - Ligação dos pinos do módulo nRF24L01 ao Arduino

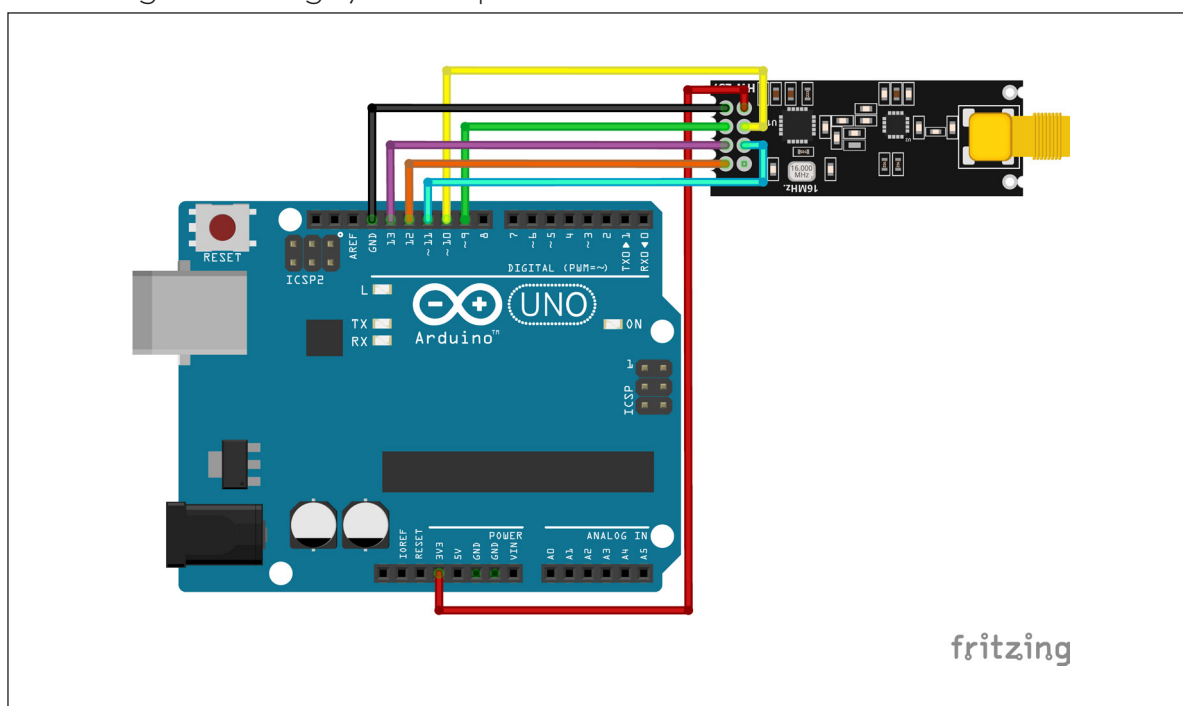


Tabela 2 - Relação entre os pinos do módulo e portas do Arduino.

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

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 o uso de 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 software Arduino IDE. (Para recordar como realizar a instalação de uma biblioteca, consulte a **Aula 05 - Softwares Arduino IDE e mBlock**, presente no módulo 1).

Caso você esteja utilizando o chromebook, a biblioteca é on-line, basta informar o seu uso no preâmbulo do sketch **#include <RF24.h>**.

Também nesse projeto, você deverá escolher outra equipe com a qual deseja realizar o chat para estabelecerem a comunicação entre os seus protótipos. Para evitar a interferência com os protótipos das outras equipes, é preciso informar nas linhas destacadas em amarelo no código (quadro 1), o mesmo endereço de comunicação (até 6 caracteres em cada canal).

Verifique também a sincronização da velocidade do Serial.begin (9600).

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

**Atenção 1:** as equipes devem utilizar identificador diferentes em cada um dos protótipos (IDENTIFICADOR 1 e IDENTIFICADOR 2), destacado em verde.

**Atenção 2:** O endereço de transmissão escolhido em um protótipo e descrito entre aspas (de até cinco caracteres) deve ser o mesmo para o endereço de recepção na programação do outro protótipo.

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

```

/*****
/* Aula 34 - Projeto Chat via RF */
/* Programação do Módulo nRF24L01. */
/* Nessa programação criaremos um chat via RF. */
/* IMPORTANTE: Para que não ocorra interferência entre */
/* os protótipos das outras duplas, vocês deverão informar */
/* nas linhas 35 e 36 deste código, o mesmo endereço de */
/* comunicação (Até 5 caracteres em cada canal). */
/* Link para obtenção da biblioteca. */
/* https://github.com/nRF24/RF24/ */
*****/
/* As equipes deverão utilizar identificadores diferentes */
/* em seus protótipos: 1 ou 2. */
#define IDENTIFICADOR 1
/*****
/* Inclui a biblioteca de controle do módulo NRF24L01. */
#include <RF24.h>
/* Define os pinos de comunicação do módulo. */
#define PINO_CE 9
#define PINO_CSN 10
/* Cria o objeto de controle do módulo para a biblioteca. */
RF24 radio(PINO_CE, PINO_CSN);
/* Variáveis auxiliares. */
String seunome = "";
String MSG_digitada;
char MSG_para_ENVIAR[32];
char MSG_RECEBIDA[32];
/* Bloco condicional para inverter os endereços de */
/* comunicação entre os protótipos de acordo com o */
/* identificador acima. */
#if IDENTIFICADOR == 1
/* ATENÇÃO: o endereço de transmissão deve ser igual nas */
/* linhas 35 e 39 - e o mesmo deve acontecer para endereço */
/* de recepção nas linhas 36 e 38. */
const byte enderecoTransmissao[6] = "00001";
const byte enderecoRecepcao[6] = "00002";
#else
const byte enderecoTransmissao[6] = "00002";
const byte enderecoRecepcao[6] = "00001";
#endif

```



```
void setup() {
  /* Inicia a comunicação serial com velocidade de 9600. */
  Serial.begin(9600);
  /* Inicia o objeto de controle do módulo rádio. */
  radio.begin();
  /* Abre um canal para o envio de mensagens com endereço */
  /* definido no início do sketch. */
  radio.openWritingPipe(enderecoTransmissao);
  /* Configura o canal 1 de recepção (0 a 5). Restringe que */
  /* o canal só receberá dados deste endereço. */
  radio.openReadingPipe(1, enderecoRecepcao);
  /* Configura o nível de potência do módulo rádio. */
  /* RF24_PA_MIN: Nível mínimo de potência. */
  /* RF24_PA_LOW: Nível de potência baixo. */
  /* RF24_PA_HIGH: Nível de potência alto. */
  /* RF24_PA_MAX: Nível máximo de potência. */
  radio.setPALevel(RF24_PA_MIN);
  /* Coloca o módulo no modo de escuta, ou seja, no */
  /* modo de recepção de dados. */
  radio.startListening();

  /* Inicia o chat solicitando o nome do usuário. */
  Serial.println("Informe seu nome...");
  /* Aguarda o usuário informar o seu nome. */
  while (seunome == "") {
    /* Se um nome for enviado no monitor serial, faça... */
    if ( Serial.available() ) {
      /* Armazene na variável o nome recebido. */
      seunome = Serial.readStringUntil('\n');
      /* Imprime a saudação personalizada com o nome do */
      /* usuário. */
      Serial.println("Bem-vindo, " + String(seunome) + "!");
    }
  }
}
```

```
void loop() {
  /* Coloca o módulo rádio na escuta. (recebendo dados). */
  radio.startListening();
  /* Se o módulo receber dados, faça... */
  if ( radio.available() ) {
    /* Leia os dados e armazene na variável MSG_RECEBIDA. */
    radio.read( &MSG_RECEBIDA, sizeof(MSG_RECEBIDA) );
    /* Imprime na tela a mensagem recebida. */
    Serial.println(MSG_RECEBIDA);
  }
  /* Se enviar algum dado no monitor serial, faça... */
  if ( Serial.available() ) {
    /* Interrompa a escuta do rádio. */
    radio.stopListening();
    /* Armazena na variável a mensagem a ser enviada */
    /* personalizada com o nome do remetente. */
    MSG_digitada = "[" + seunome + "]" + Serial.readStringUntil('\n');
    /* Imprima na tela a mensagem a ser enviada. */
    Serial.println(MSG_digitada);
    /* Converte a mensagem a ser enviada para o formato */
    /* CharArray para que possa ser enviada pelo rádio. */
    MSG_digitada.toCharArray(MSG_para_ENVIAR, 32);
    /* Envia a mensagem através do rádio. */
    radio.write( &MSG_para_ENVIAR, sizeof(MSG_para_ENVIAR) );
  }
}
```

Com o código-fonte inserido no Arduino IDE, compile o programa pressionando o botão **Verificar** para certificar se não há erros de sintaxe. Estando o código correto, o próximo passo é realizar a transferência do programa para o Arduino. Pressione o botão **Carregar** para realizar *upload* do programa para o Arduino.

Após a transferência do programa para o Arduino, o protótipo estará pronto para a comunicação. Abra o monitor serial **Ferramentas > Monitor serial** e utilize a caixa de texto para escrever, enviar e receber as mensagens.





## Desafio:

Que tal incrementar seu chat acrescentando um indicador sonoro e visual ao receber ou enviar uma nova mensagem? Agora é a sua vez! Insira um ou mais LEDs e um buzzer em seu protótipo e programe-os para aperfeiçoar o seu chat.



## E se...?

Caso o projeto não funcione:

- a. Verifique se a bibliotecas está corretamente instalada no software Arduino IDE;
- b. Verifique se os jumpers estão nos pinos certos do módulo;
- c. Verifique se os jumpers estão ligados nas portas corretas do Arduino;
- d. Verifique se a programação está adequada às portas digitais.



### 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 os requisitos para funcionamento do chat via RF.

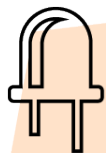
**c.** Reflita se as seguintes situações ocorreram:

**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.





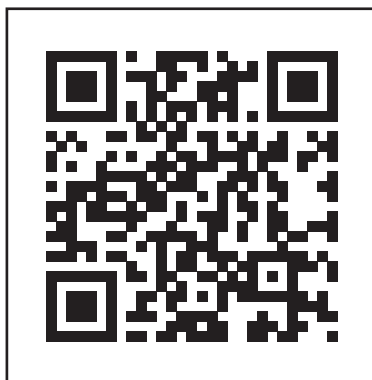
## Videotutorial

Com o intuito de auxiliar na montagem e na programação desta aula, apresentamos um videotutorial, disponível em:



<https://rebrand.ly/ChatnRF24L01>

Acesse, também, pelo QRCode:



DIRETORIA DE TECNOLOGIAS E INOVAÇÃO (DTI)  
KIT 2023  
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/)

