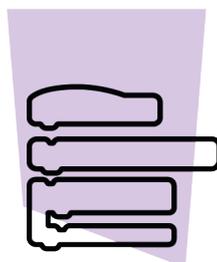
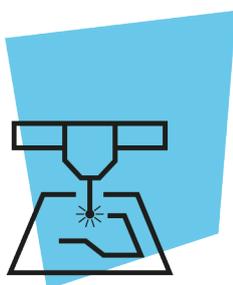
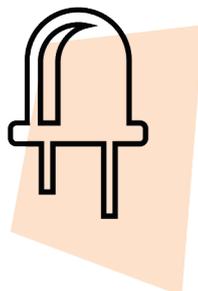
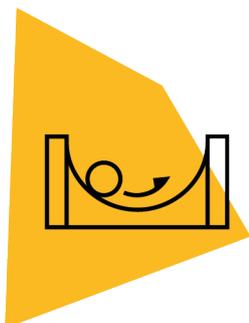
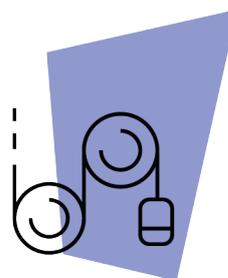
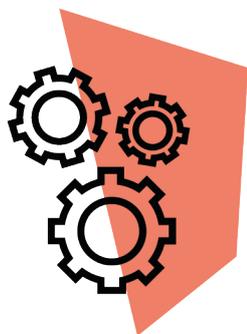


ROBÓTICA

Módulo 2



Projeto Chat via RF

AULA 34

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

Michelle dos Santos

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 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 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 RF
433MHz - II

Aula 34
Projeto CHAT
via RF

Aula 35
Módulo
Wireless

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	9
3. Feedback e finalização	16
Videotutorial	17



Introdução

O uso de aplicativos de mensagens para conversas tem se tornado cada vez mais comum, principalmente pelos celulares. Isso se deve à facilidade e à rapidez que estes dispositivos proporcionam à comunicação. Há várias maneiras possíveis de se ter esta funcionalidade, dentre elas temos a radiofrequência (RF).

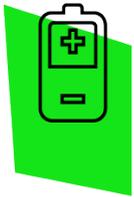
Montar uma rede de comunicação sem fio utilizando a radiofrequência (RF), é simples, barata e realiza a comunicação direta de dados entre dois diferentes componentes, isto é, o RF transmissor e RF receptor.

Nesta aula aprenderemos a programar um protótipo de chat, utilizando a o **módulo RF 433MHz** para se comunicar com outra equipe.



Objetivos desta Aula

- Retomar a utilização do módulo RF 433MHz;
- Criar o protótipo de um chat via RF;
- 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.



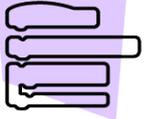
Habilidades do Século XXI a Serem Desenvolvidas

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



Lista de Materiais

- 1 placa arduino Uno;
- 1 cabo USB;
- 1 placa protoboard;
- 1 módulo RF 433MHz – transmissor;
- 1 módulo RF 433MHz – receptor;
- 8 jumpers macho-macho;
- Notebook;
- Software Arduino IDE.



Roteiro da Aula

1. Contextualização (15min)

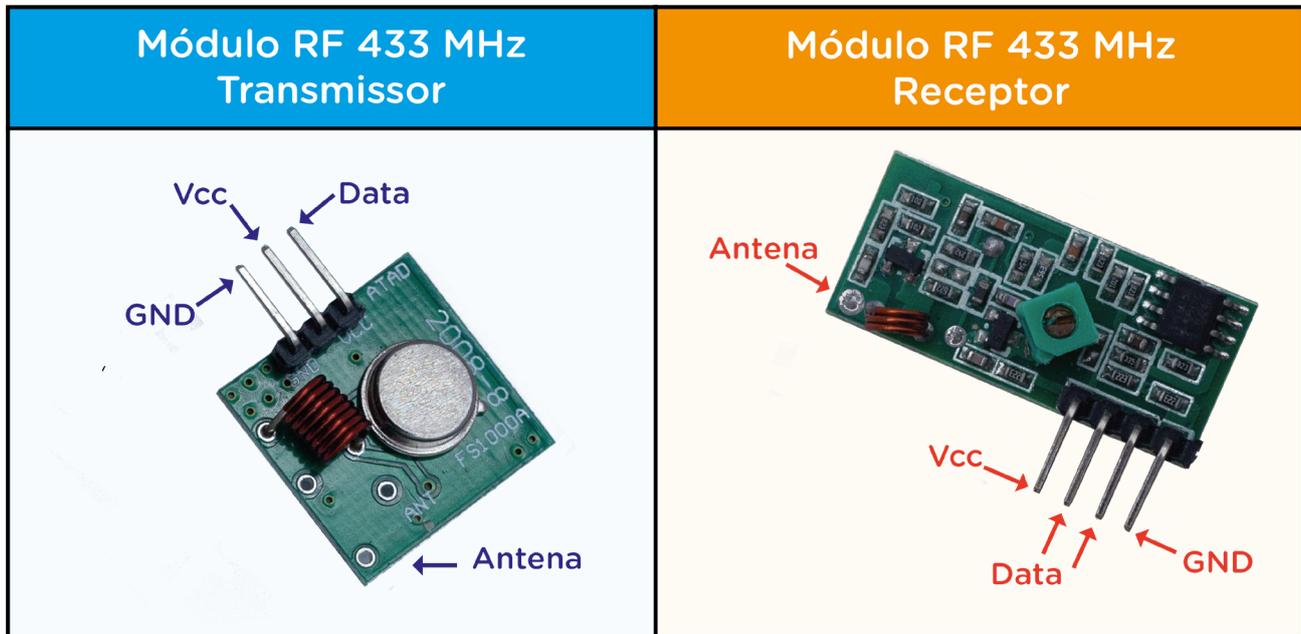
Você utiliza muito aplicativo de conversa para trocar mensagens com amigos, familiares entre outras pessoas? Tem se tornado tão comum que até para fazer compras e pedido de comida estão na lista de atividades corriqueiras do cotidiano. Já imaginou aprender a desenvolver um chat?

Na atualidade com a velocidade de troca de informação se faz necessário ter ferramentas de comunicação rápida e instantânea, a qual vem crescendo com o uso de dispositivos que propiciam envio direto de mensagem.

Como já estudamos na **Aula 31 - Módulo RF 433MHz**, uma alternativa eficiente, simples e de baixo custo para transmissão de dados sem fio é o uso da radiofrequência, que é amplamente utilizado em sistemas eletrônicos.

Esta comunicação é classificada como unidirecional, ocorre entre dois pontos distintos, ou seja, os dados são enviados pelo transmissor e recebidos pelo receptor. Este módulo utiliza frequência de 433MHz e pode transmitir dados numa velocidade de 9200 bits por segundo, e conforme maior for a distância entre o RF transmissor e o RF receptor (figura 1), menor deve ser a velocidade na transmissão dos dados para diminuir a probabilidade de perdas de dados.

Figura 1 - Kit módulo RF transmissor e receptor

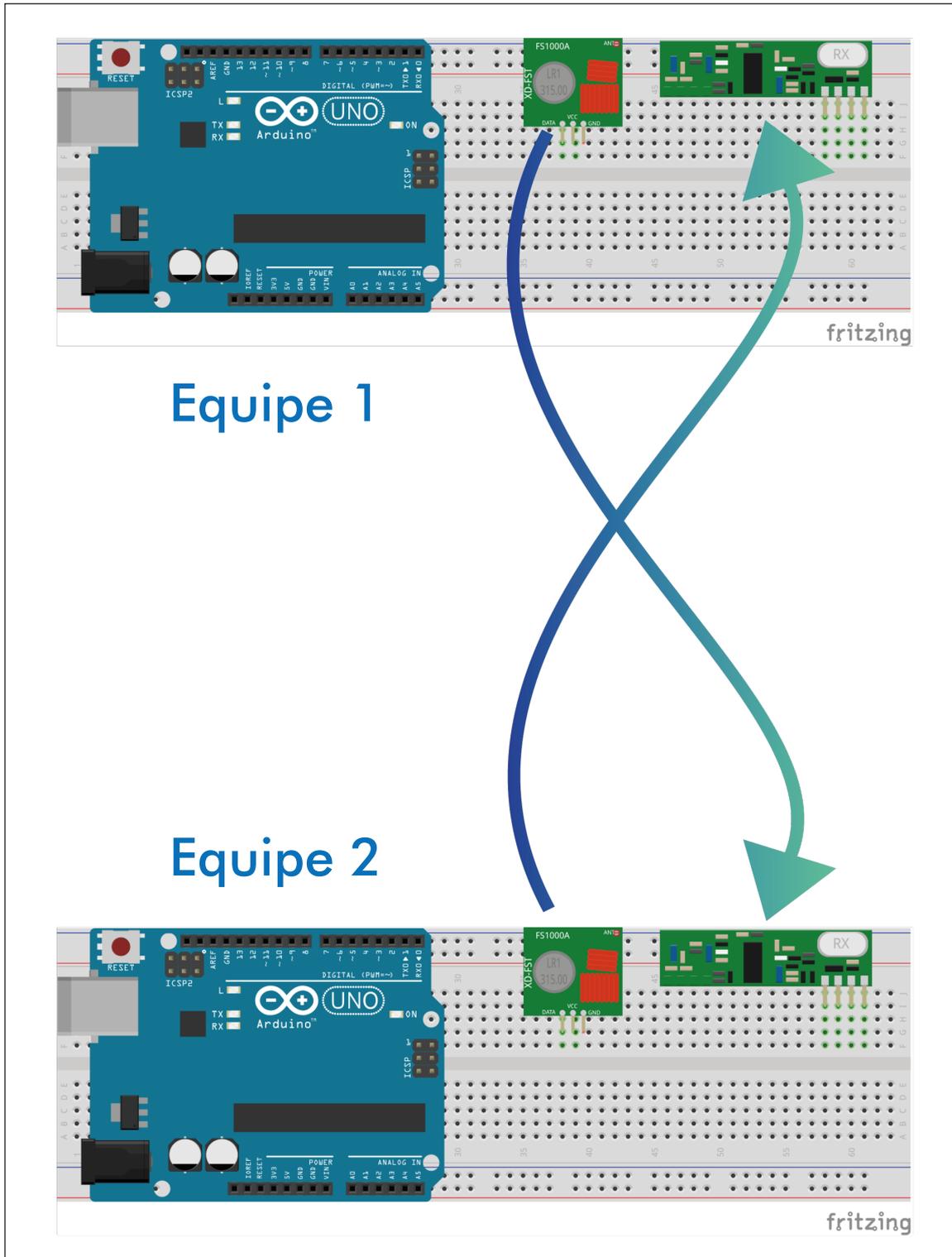


Nesta aula você aprenderá a estabelecer comunicação via chat com outra equipe. Utilizaremos o módulo RF transmissor do protótipo de uma equipe para envio de sinais, que conectado junto ao Arduino processará e encaminhará a informação em comandos de mensagens ao módulo receptor RF do protótipo da outra equipe, e vice-versa, estabelecendo assim a transmissão de dados e a comunicação (figura 2).

ATENÇÃO

Uma das placas dos módulos transmissor/receptor deverá ser trocada com a mesma placa da equipe que irá estabelecer a comunicação, para que o transmissor envie as mensagens para o seu par receptor que estará no outro protótipo.

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!

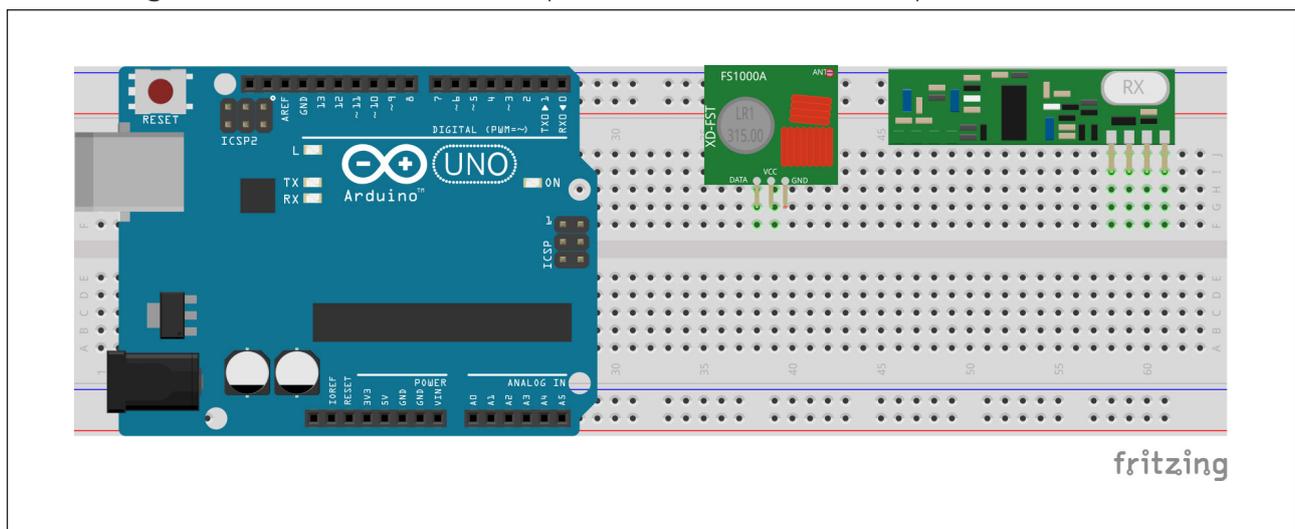
<http://www.eeh2012.anpuh-rs.org.br/resources/pe/anais/encontro5/14-os-desafios/Artigo%20de%20Jos%E9%20Arimat%E9ia%20Santos.pdf>



2. Montagem e programação (60min):

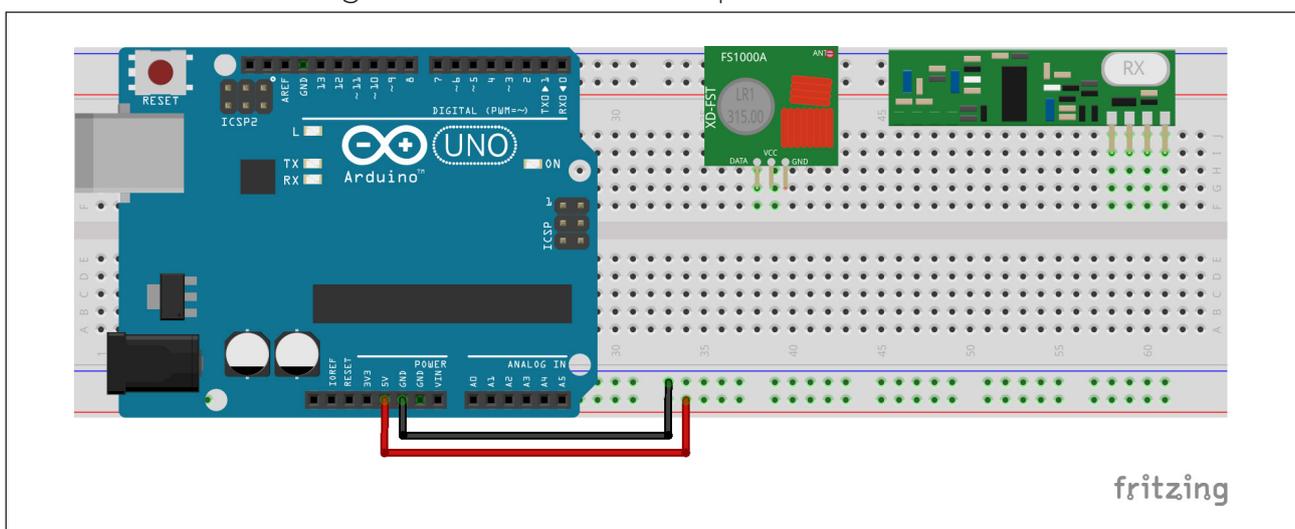
Iniciaremos a montagem do protótipo encaixando os módulos receptor e transmissor na placa protoboard (figura 3).

Figura 3 - Encaixe dos componentes eletrônicos à placa Protoboard



Utilizando dois jumpers, interligue as portas 5V e GND do Arduino com os furos das linhas laterais **Vermelha** e **Azul**, respectivamente, da protoboard (figura 4).

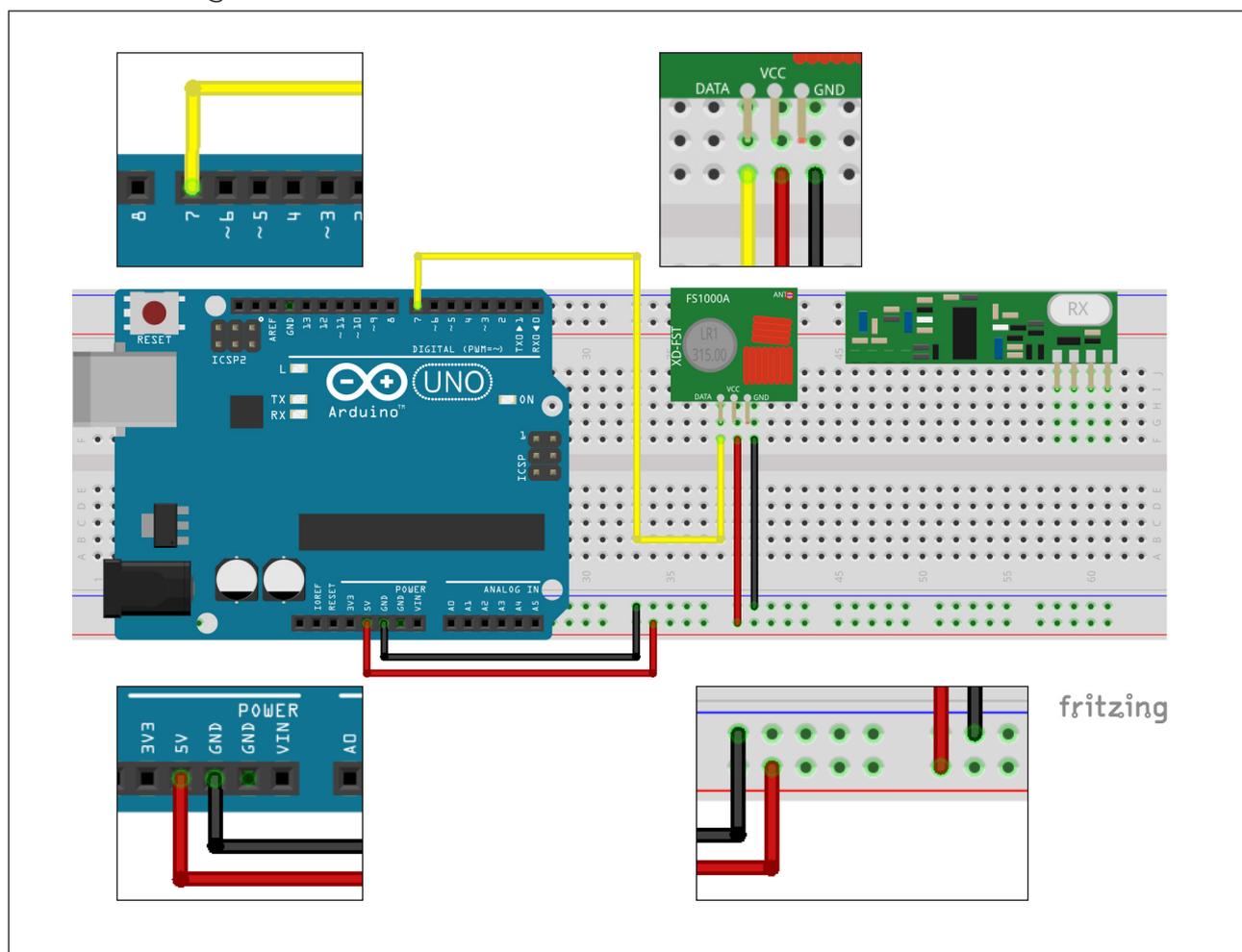
Figura 4 - Alimentando a placa Protoboard



Agora, faremos as conexões do módulo transmissor RF 433MHz com a placa Arduino, para isso, utilize três jumpers e interligue as linhas laterais **Azul** e **Vermelha** da protoboard e a porta digital 7 do Arduino, com os pinos GND, VCC e Data do módulo transmissor, respectivamente (figura 5).

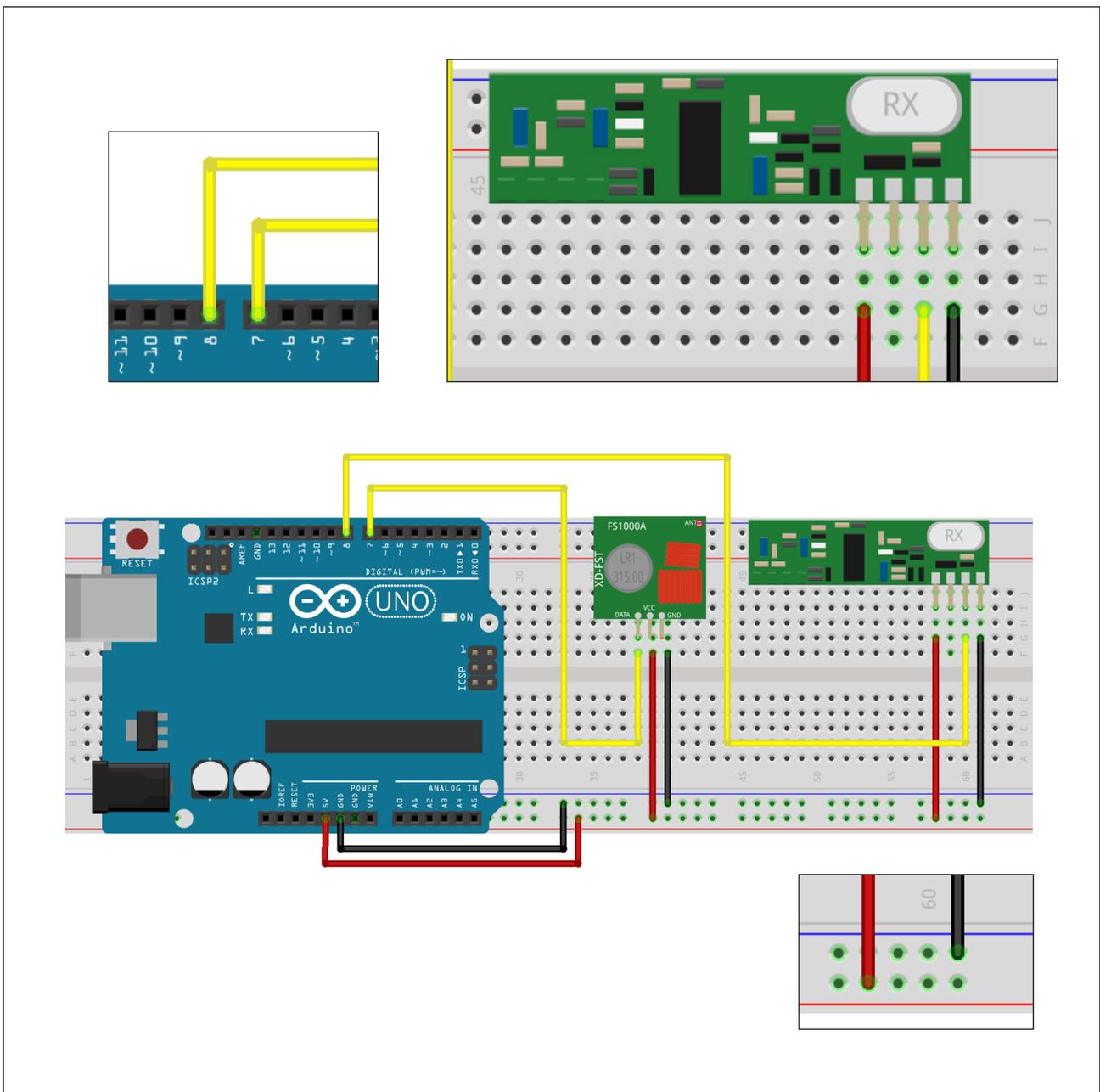
Observação: Para aumentar o alcance da transmissão, você poderá conectar o pino VCC do módulo transmissor na porta VIN do Arduino e utilizar a fonte chaveada de 9V para alimentar o Arduino com o uso do conector plug P4.

Figura 5 - Conexões do Módulo Transmissor com o Arduino



Para finalizar a montagem, faremos as conexões do módulo receptor com o Arduino. Utilize três jumpers e interligue as duas linhas laterais **Azul** e **Vermelha** da protoboard e a porta digital 8 do Arduino com os pinos GND, VCC e Data do módulo receptor, respectivamente, (figura 6).

Figura 6 - Conexões do Módulo Receptor com o Arduino



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

Nessa programação, utilizaremos a biblioteca “RadioHead”, desenvolvida por Mike McCauley, que auxiliará no controle do módulo RF433MHz. Ela 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 < RH_ASK.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 na linha destacada em amarelo no código (quadro 1), a mesma velocidade de comunicação (2000, 2100, 2200 etc.).

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

Atenção!

Ao copiar o código diretamente do pdf, evite quebra da página (e conseqüentemente erros na compilação), copiando o código por partes.



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

```

/*****
/* Aula 34 - Projeto Chat via RF
/* Nessa programação criaremos um chat via RF.
/* IMPORTANTE! Para que não ocorra interferência entre
/* os protótipos de outras duplas, você e seu amigo deverão
/* informar na linha 15 deste código, a mesma velocidade de
/* comunicação: 2000, 2100, 2200, 2300, etc...
/* Links para obtenção da biblioteca by Mike McCauley.
/* http://librarymanager/All#RadioHead#Provides
/* http://www.airspayce.com/mikem/arduino/RadioHead/
*****
/* Inclui a biblioteca de controle do módulo rádio.
#include <RH_ASK.h>
/* Definições para o controle do módulo rádio.
#define velocidade 2000
#define pinoRX 8
#define pinoTX 7
/* Cria o objeto de controle "radio".
RH_ASK radio(velocidade, pinoRX, pinoTX);

void setup() {
  /* Inicia o monitor serial com velocidade 9600.
  Serial.begin(9600);
  /* Inicia o objeto rádio.
  radio.init();
}

void loop() {
  *****/
  /*          Programação para o envio de mensagens
  *****/
  /* Se receber algo pela serial, faça...
  if (Serial.available() > 0) {
    /* Variável que irá guardar a mensagem a ser enviada.
    String msg_a_enviar = "";
    /* Variável que armazenará os caracteres da mensagem.
    char caractere;

```

```
/* Enquanto receber algo pela serial, faça... */
while (Serial.available() > 0) {
  /* Guarda cada caractere capturado da serial. */
  caractere = Serial.read();
  /* Se o caractere NÃO for uma quebra de linha, faça... */
  if (caractere != '\n') {
    /* Concatena cada caractere na variável. */
    msg_a_enviar += caractere;
  }
  /* Pequena pausa antes de ler o próximo caractere. */
  delay(10);
}
Serial.println("Você: " + msg_a_enviar);
/* Converte a String para bytes e envia através do rádio.*/
radio.send((uint8_t *)msg_a_enviar.c_str(), msg_a_enviar.
length());
/* Aguarda o tempo necessário para o envio. */
radio.waitPacketSent();
}
/*****
/*****
/*      Programação para o recebimento de mensagens      */
/*****
/* Recebe a mensagem com o máximo de 58 caracteres.      */
uint8_t buf[RH_ASK_MAX_MESSAGE_LEN];
/* Determina o comprimento da mensagem recebida.        */
uint8_t buflen = sizeof(buf);
/* Se recebeu corretamente a mensagem, faça...          */
if (radio.recv(buf, &buflen)) {
  /* Define a variável para receber a mensagem completa. */
  String msg_recebida = "";
  /* Loop para montar a palavra letra por letra.         */
  for (int i = 0; i < buflen; i++) {
    msg_recebida += (char)buf[i];
  }
  /* Imprime no monitor serial a mensagem recebida.     */
  Serial.println("Amigo(a): " + String(msg_recebida));
}
/*****
}
```

Com o código-fonte inserido no Arduino IDE, compile o programa pressionando o botão **Verificar** para verificar 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 sua mensagem e enviá-la.



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...

O projeto não funcionar? Atente-se para alguns dos possíveis erros:

Teste a comunicação com os protótipos próximos um do outro e vá afastando para identificar o máximo de alcance.

Verifique se a biblioteca está corretamente instalada no software Arduino IDE;

Verifique se os jumpers estão nos pinos certos, se estão na mesma coluna dos terminais dos componentes, fazendo assim a conexão;

Verifique se os jumpers estão ligados nos pinos corretos no Arduino;

Verifique se a programação está adequada às portas digitais.

3. Feedback e finalização (15min)

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 Chat via RF.

c. Reflita se as seguintes situações ocorreram:

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

e. Pensamento Crítico e Resolução de Problemas: você conseguiu identificar os problemas, analisar informações e tomar decisões de modo a contribuir para o projeto desenvolvido?

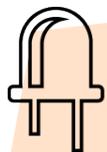
f. Reúna todos os componentes utilizados nesta aula e os organize novamente, junto aos demais, no kit de robótica.



AULA

34

PROJETO CHAT VIA RF



Videotutorial

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



<https://rebrand.ly/a34robotica2>

Acesse, também, pelo QRCode:



Robótica módulo 2

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/)

