

AULA 29

Primeiros Passos Módulo 3

ROBÓTICA



Alarme Musical



Diretoria de Tecnologia e Inovação

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

Andrea da Silva Castagini Padilha

Validação de Conteúdo

Cleiton Rosa

Darice Alessandra Deckmann Zanardini

Revisão Textual

Kellen Pricila dos Santos Cochinski

Projeto Gráfico e Diagramação

Edna do Rocio Becker

Apoio Técnico

Equipe UFMS

2024

SUMÁRIO

Introdução	2
Objetivos desta aula	2
Roteiro da aula	3
1. Contextualização	3
2. Montagem	9
3. Feedback e finalização	33
Referências bibliográficas	33

Alarme musical

Introdução

No módulo 3, você teve oportunidade de programar notas musicais na **Aula 10 – Teclado musical**. Agora, retomaremos a programação do buzzer para criar um protótipo bem útil na vida de um estudante.

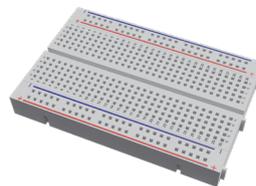
Nesta aula, vocês serão desafiados a criar um alarme sonoro com componentes do kit de Robótica.

Objetivos desta aula

- Construir um protótipo de alarme musical com buzzer (piezoelétrico passivo) e Arduino;
- Compreender a lógica de programação de sons e melodias;
- Programar notas musicais no mBlock 5.0.

Lista de materiais

- 01 Protoboard;
- 01 Buzzer passivo;
- 01 Placa Arduino;
- 01 LED;
- 01 Resistor 220 Ω ;
- 04 Jumpers macho-macho.





Roteiro da aula

1. Contextualização

O alarme sonoro, como conhecemos hoje, é um dispositivo relativamente recente. No entanto, a ideia de um sistema que desperte as pessoas de maneira sonora tem uma longa história. Alguns dos precursores do alarme musical incluem:

Relógios de água egípcios (1500 a.C.): esses relógios usavam água pingando para marcar o tempo e podiam ser adaptados para emitir sons ao encherem.

Mecanismos de alarme mecânicos medievais (século XIV): relógios mecânicos complexos podiam soar sinos em determinados horários.

Despertadores de velas (século XVIII): velas com marcações de tempo eram acesas à noite e, ao derreter, acendiam fios presos a sinos, fazendo-os soar.

A invenção do despertador mecânico moderno é creditada a Levi Hutchins em 1789. No entanto, esses primeiros despertadores eram barulhentos e desagradáveis. Com o advento da eletricidade no final do século XIX, surgiram os primeiros despertadores elétricos. Isso pavimentou o caminho para o desenvolvimento de alarmes musicais, que utilizavam campainhas ou buzzers para tocar melodias suaves.

Atualmente, os alarmes sonoros estão integrados em aplicativos de celulares, que a pessoa pode personalizar com músicas, sons de natureza ou outro efeito sonoro de sua preferência para acordar e começar bem o dia. Hoje em dia, smartphones têm a funcionalidade de alarme sonoro, bem como vários eletrodomésticos das casas "inteligentes". É possível programar determinados eletrodomésticos (como os robôs aspiradores, por exemplo) para funcionarem por determinado tempo, ou serem acionados em um horário específico. A aula de hoje ajudará a entender como programar funções de timer e alarme para disparar um aviso sonoro.

Figura 1 – Despertador de vela



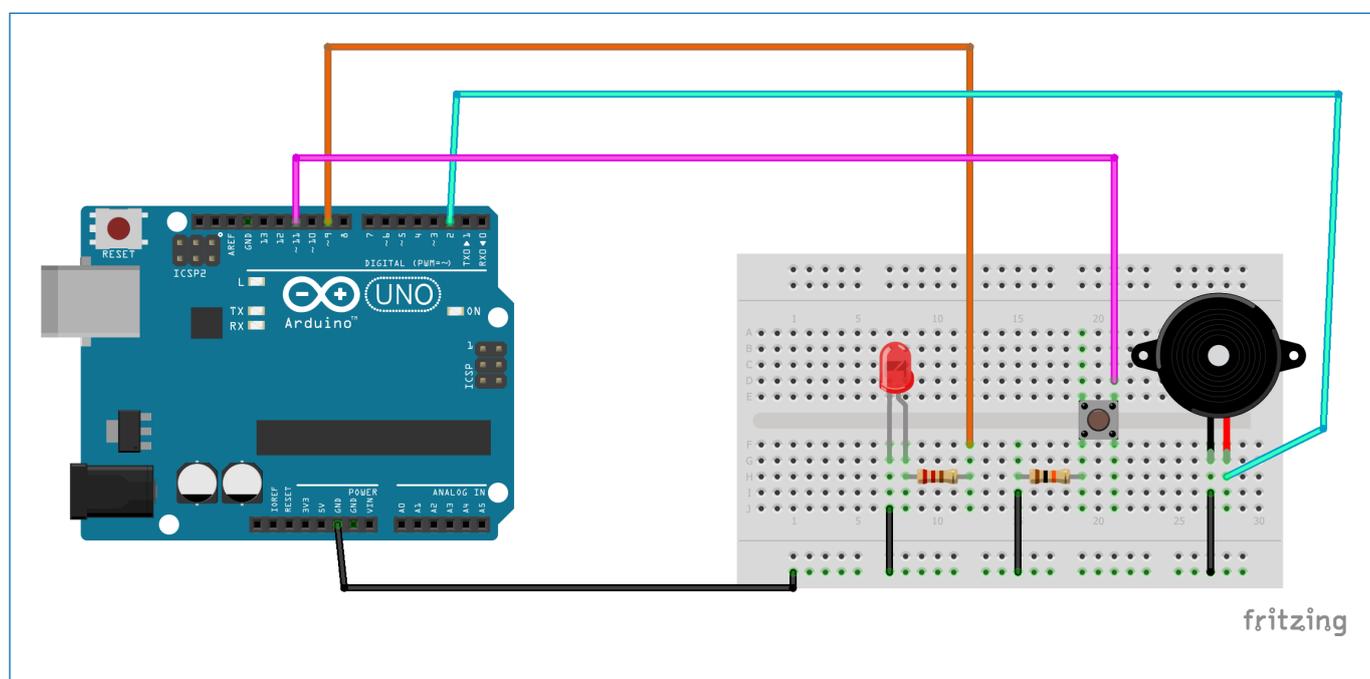
Fonte: Wikipedia.

2. Montagem e programação

A montagem deste protótipo é bastante simples. Conecte o buzzer na protoboard, com atenção para o terminal negativo que pode ser conectado à lateral negativa da protoboard e o terminal positivo que será conectado à porta digital 02 do Arduino.

Insira o push button na divisão da protoboard (veja a Figura 2), e na coluna de um dos pinos do push button insira um jumper cujo outro terminal deve ser conectado na porta digital 11 do Arduino. Insira um resistor de $220\ \Omega$ na coluna em que está o outro terminal do push button e esse conectado à lateral negativa da protoboard (poderá usar um jumper como mostra a Figura 2, caso necessário). Por último, conecte o LED que emitirá o aviso visual do alarme, observando qual o terminal positivo. Na coluna em que esse terminal estiver inserido, posicione um resistor de $220\ \Omega$ e desse um jumper que ligará o LED e resistor à porta digital 09 do Arduino. No terminal negativo do LED, insira um jumper, conectando-o à lateral negativa da protoboard. Utilize um jumper para ligar a lateral negativa da protoboard à porta GND do Arduino. Veja na Figura 02 a ligação esquemática do protótipo.

Figura 2 - Modelo esquemático da montagem

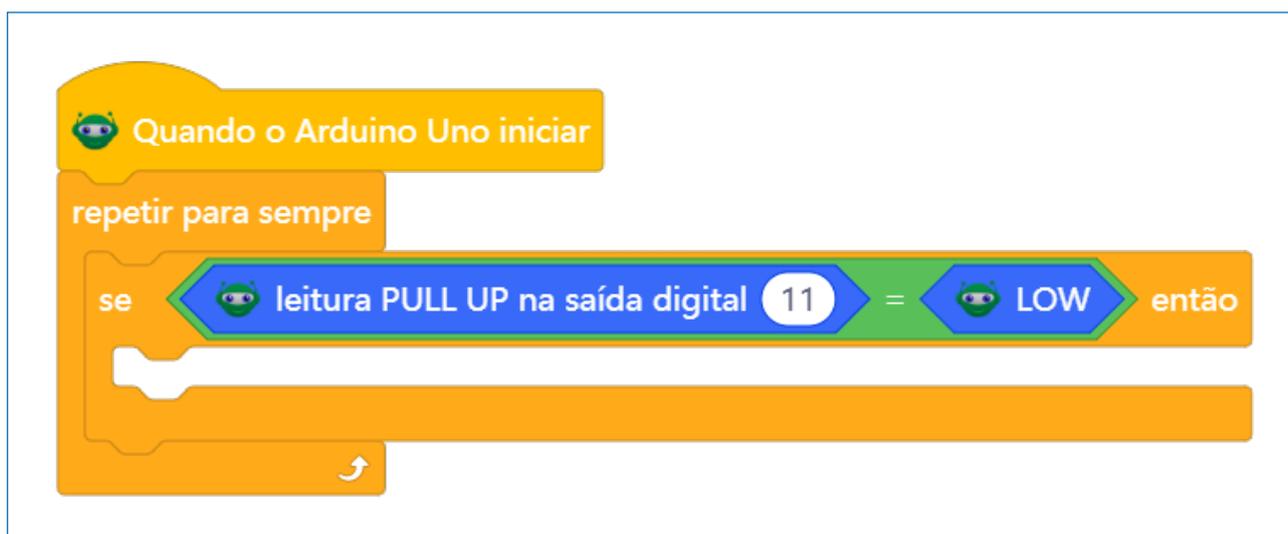


Alarme musical

Agora, vamos programar!

Inicie como de costume, trazendo para o palco do mBlock o bloco <Quando o Arduino Uno iniciar> e a seguir o bloco <repetir para sempre>, encontrado na categoria **Controlo**. Agora insira dentro desse bloco, a condicional que irá determinar o que acontece quando o push button for acionado. Para tanto, conecte o bloco <se ... então...>. No espaço após o "se", insira o bloco operador <igual>. Na parte anterior ao sinal de igual, insira o bloco <leitura PULL UP na saída digital 11> e no espaço após o sinal de igual o bloco <LOW>, encontrado na categoria portas. Esse bloco indica que a leitura PULL UP do push button conectado na porta 11, quando pressionado (Low) desencadeará ações e, se não estiver pressionado, não fará nada. Veja os blocos na Figura 3.

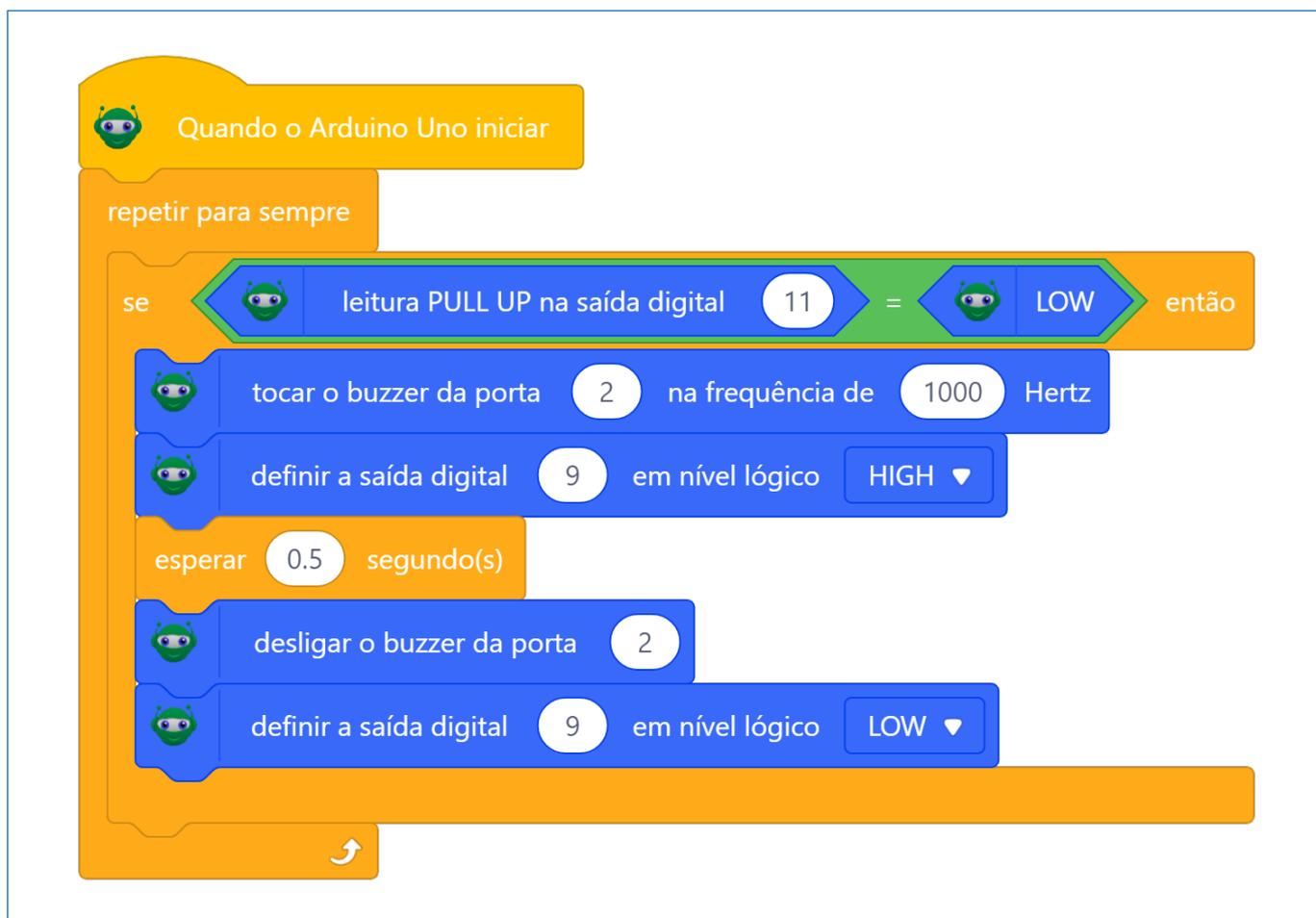
Figura 3 - Blocos de programação iniciais



Agora programe o buzzer para tocar, indicando que a partir desse toque será computado o tempo para o acionamento do alarme sonoro. Para tanto, na categoria portas, selecione o bloco <tocar o buzzer na porta 2 na frequência 1000 hertz> e em seguida o acionamento do LED, com o bloco <definir a saída digital 9 em nível lógico HIGH>. Agora, insira o bloco <esperar 0,5 segundos>, <desligar o buzzer da porta 2> e <definir a saída digital 9 em nível lógico LOW>. Esses blocos irão apagar o LED e desligar o buzzer.

Alarme musical

Figura 4 - Acionamento inicial do buzzer e LED

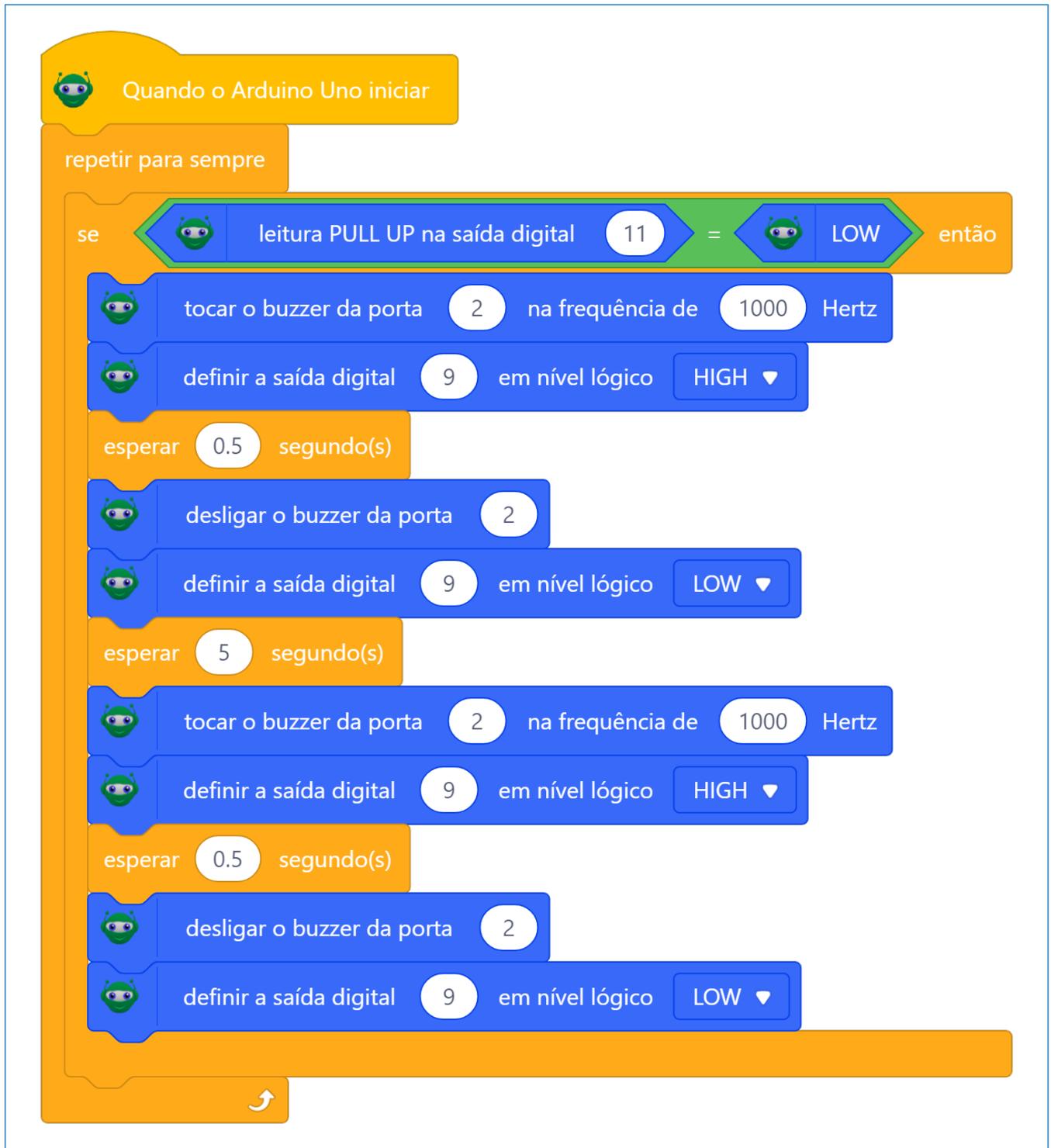


Para programar o tempo que o alarme tocará, selecione o bloco <esperar **xx segundos**>. Esse será o bloco que você e seus colegas poderão personalizar, indicando quanto tempo será programado para determinada tarefa, por exemplo, 5 minutos... Lembre-se de converter o tempo para segundos. A seguir, insira os blocos <tocar o buzzer na porta **2 na frequência 1000 hertz**> e em seguida o acionamento do LED, com o bloco <definir a saída digital **9 em nível lógico HIGH**>. Agora, insira o bloco <esperar **0,5 segundos**>, <desligar o buzzer da porta **2**> e <definir a saída digital **9 em nível lógico LOW**>, para tocar o buzzer indicando que o tempo programado acabou!

Alarme musical

Veja a montagem completa na Figura 5.

Figura 5 - Programação completa



Alarme musical

Desafios:

Que tal incluir a matriz de LED 8x8 para indicar um emoticon indicando que o tempo acabou?

Como você e sua equipe podem alterar o código com indicação do LED piscando conforme o tempo estiver acabando?

Outro desafio é personalizar o efeito sonoro do alarme com uma música, alternando as notas musicais. Há repositórios de músicas para Arduino (em programação de código, mas não em programação em blocos), que você e sua equipe poderão se basear para programar. Na aula 08 você encontrará indicações dos blocos para criar sua melodia.

E se...

O projeto não funcionar?

- Verifique a ligação dos jumpers;
- Observe se os terminais positivo e negativo do buzzer estão corretamente conectados e energizados;
- Verifique a programação.

3. Feedback e finalização

- a. Confira, com seus colegas, se o objetivo de criar um alarme sonoro foi atingido.
- b. Quais as aplicabilidades deste protótipo no mundo real?
- 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. Houve alguma dificuldade na construção deste protótipo? Como você e seus colegas identificaram os problemas, analisaram informações e superaram os obstáculos para desenvolver o projeto?
- d. Reúna todos os componentes utilizados nesta aula e os organize novamente, junto aos demais, no kit de Robótica.

REFERÊNCIAS

ARDUINO. **Documentação de Referência da Linguagem Arduino**. Disponível em: <https://www.arduino.cc/reference/pt/>. Acesso em: 27 mai. 2024.



UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL (UFMS)
FACULDADE DE COMPUTAÇÃO (FACOM)

PROFESSORES

- Amaury Antônio de Castro Junior
- Anderson Corrêa de Lima
- Glauder Guimarães Ghinozzi
- Graziela Santos de Araújo
- Said Sadique Adi

ESTUDANTES (elaboração prévia)

- Filipe de Andrade Machado - Ciência da Computação
- Gabriel Alves Massuda Duarte - Engenharia de Computação
- José Augusto Lajo Vieira Vital - Ciência da Computação
- Lorena Valente Cavalheiro - Engenharia de Computação
- Matheus Kazumi Silva Miyashiro - Engenharia de Computação
- Nathalia dos Santos Melo - Engenharia de Software
- Yan Arruda Cunha - Engenharia de Computação
- Thiago Ferronato - Ciência da Computação
- Vitor Hugo dos Santos Duarte - Engenharia de Computação
- Wilker Sebastian Afonso Pereira - Ciência da Computação

ESTUDANTES (revisão)

- Arthur Henrique Andrade Farias - Ciência da Computação
- Bruno Pereira Wesner da Silva - Engenharia de Computação
- Fernanda das Neves Merqueades Santos - Ciência da Computação
- Gabriel Pereira Falcão - Ciência da Computação
- Jenniffer Oliveira Checchia - Ciência da Computação
- Leonardo Vargas de Paula - Sistemas de Informação
- Marcos Gabriel da Silva Rocha - Engenharia de Computação
- Maria Paula do Nascimento Santos - Engenharia de Computação
- Nathanael Martins Wink - Ciência da Computação
- Victor Luiz Marques Saldanha Rodrigues - Ciência da Computação

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

EQUIPE ROBÓTICA PARANÁ

- Adilson Carlos Batista
- Ailton Lopes
- Andrea da Silva Castagini Padilha
- Cleiton Rosa
- Darice Alessandra Deckmann Zanardini
- Edna do Rocio Becker
- Kellen Pricila dos Santos Cochinski
- Marcelo Gasparin
- Michele Serpe Fernandes
- Michelle dos Santos
- Roberto Carlos Rodrigues
- Sandra Aguera Alcova Silva
- Viviane Dziubate Pittner

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/)



DTI - DIRETORIA DE TECNOLOGIA E INOVAÇÃO