



Universidade de Brasília

FACULDADE UnB PLANALTINA

CIÊNCIAS NATURAIS

**EDUCAÇÃO MODERNA E DE QUALIDADE PARA
TODOS? ROBÓTICA EDUCACIONAL, PROJETO
COM ARDUINO**

Marina Guimarães Brasileiro

Orientador: Ismael Victor Lucena Costa

Planaltina- DF

Novembro-2020



Universidade de Brasília

FACULDADE UnB PLANALTINA

CIÊNCIAS NATURAIS

**EDUCAÇÃO MODERNA E DE QUALIDADE PARA
TODOS? ROBÓTICA EDUCACIONAL, PROJETO
COM ARDUINO**

Marina Guimarães Brasileiro

Orientador: Ismael Victor Lucena Costa

Planaltina- DF

Novembro-2020

DEDICATÓRIA

Dedico aos meus familiares, amigos, professores e a Deus.
Josué 1.9 “Esforça-te, e tem bom ânimo; não temas, nem te espantes; porque o Senhor teu Deus é contigo, por onde quer que andares.”

AGRADECIMENTOS

Agradeço primeiramente a Deus, pois sem ele nada seria. Obrigada senhor por me abençoar com pais maravilhosos, família presente e amorosa, amigos únicos e dedicados, professores diligentes e virtuosos.

Agradeço aos meus pais por todos esses anos de dedicação e cuidados, sem vocês as minhas vitórias e conquistas não teriam graça. Deus me presenteou vocês, seus cuidados e ensinamentos são valiosos e responsáveis pela pessoa que me tornei. Sou grata por tudo, vocês são as grandes preciosidades da minha vida.

Agradeço aos meus avôs paternos que fui agraciada por conhecer, que sempre foram meu maior exemplo para me tornar uma pessoa com vontade de aprender e de ensinar, que me mostraram que o estudo é uma das coisas mais valiosas a se adquirir no mundo. Sou grata por todos os ensinamentos, cuidados e amor que me deram.

Agradeço aos meus irmãos, Janiquel e Jadson, aos meus tios, tias, primos e prima que sempre estiveram presentes nos momentos bons e ruins, que sempre buscam a união familiar e fizeram a minha vida mais divertida. Agradeço em especial as minhas tias Maura e Magna que fizeram do seu lar meu lar.

Agradeço as minhas amigas, Tayne e Wanessa que trilharam toda a graduação ao meu lado, que nunca me abandonaram e tornaram toda a trajetória mais fácil. Sou grata por todas as caronas, ajudas, viagens, risadas e conversas.

Agradeço ao meu futuro esposo e seus familiares por cuidarem de mim e tornar meus dias melhores. Vocês foram presentes enviados por Deus, sou grata por todo amor e carinho dedicados a mim.

Agradeço aos meus professores que me fizeram chegar até aqui, que tiveram a paciência de me ensinar e me ajudar. Agradeço ao meu orientador por me guiar neste trabalho. Agradeço ao Bessa por acreditar em mim e me orientar no seu programa de educação tutorial (PET). Agradeço a Erina, exemplo de mulher, doutora, que vai além de professora, se tornou uma amiga e o meu maior exemplo de carreira profissional, sou grata por ter acreditado no meu potencial, pela sua amizade e apoio.

EDUCAÇÃO MODERNA E DE QUALIDADE PARA TODOS? ROBÓTICA EDUCACIONAL, PROJETO COM ARDUINO

Marina Guimarães Brasileiro¹

Ismael Victor Lucena Costa, orientador²

RESUMO: A globalização que conectou o mundo é a mesma que influencia o ensino, o mundo evolui e a educação deve evoluir junto, a tecnologia aliada ao ensino pode potencializar a aprendizagem.

A robótica Educacional exercita e instiga a curiosidade, a imaginação e a intuição (GOMES; SILVA; BOTELHO; SOUZA, 2010), permite o desenvolvimento do pensamento computacional, trabalha com a resolução de problemas, leva ao letramento computacional e a interdisciplinaridade. No mercado existe muitas opções para se trabalhar a robótica, o Arduino é uma delas.

O Arduino possui um *hardware* e um *software* livre, que permite a construção dos mais diversos modelos educacionais. Aqui será construído o radar ultrassônico e a partir dele será extraído os conteúdos que podem ser ensinados demonstrando a interdisciplinaridade desses projetos.

Palavras-chave: Robótica educacional, Arduino, radar, ultrassom.

ABSTRACT: The globalization that has connected the world is the same influence in teaching, the world evolves and education must evolve together, technology allied to teaching can enhance learning.

Educational robotics exercises and incite curiosity, imagination and intuition (GOMES; SILVA; BOTELHO; SOUZA, 2010), allows the development of computational thinking, works with problem solving, leads to computer literacy and interdisciplinarity. In the market there are many options to work robotics, Arduino is one of them.

Arduino has a hardware and a free software, which allows the construction of the most diverse educational models. Here the ultrasonic radar will be built and from it will be extracted the contents that can be taught demonstrating the interdisciplinarity of these projects.

Keywords: Educational robotics, Arduino, radar, ultrasound.

¹ Curso de Ciências Naturais - Faculdade UnB de Planaltina

² Universidade de Brasília- UnB

1. INTRODUÇÃO

Educação e robótica combinam? O que se pode esperar de um projeto educacional construído com *software e hardware*?

A educação, direito de todos e dever do Estado e da família, [...] visando ao pleno desenvolvimento da pessoa, seu preparo para o exercício da cidadania e sua qualificação para o trabalho (Atr. 205, Constituição Federal, 1988).

A educação de qualidade é um direito prioritário, fazer dos estudantes cidadãos ativos e preparados para o mundo é um dever do Estado e da família. Para uma educação de qualidade é essencial que ela se modele e modernize, caminhe de acordo com os avanços sociais.

Para a Base Nacional Comum Curricular (BNCC), Ciências da Natureza (CN) é responsável por apresentar aos estudantes o desenvolvimento científico e tecnológico. A modernização do ensino e o surgimento de novas práticas permitem uma exploração maior dos campos científicos e tecnológicos.

A robótica educacional é um caminho que fornece práticas divertidas e atraentes, que explora mais de uma área do conhecimento, impulsiona a aprendizagem e permeia em outras práticas pedagógicas, como colaboração e resolução de problemas. Na construção de um modelo robótico, o processo de colaboração acontece quando os problemas são analisados e resolvidos (Silva, 2009).

Existe diversos kits educacionais de robótica, dentre eles, a plataforma Arduino permite a criação de modelos educacionais, de circuitos simples a robôs, o contato com uma linguagem computacional, além de ser um instrumento de fácil acesso.

O projeto aqui trabalhado é um radar ultrassônico construído com Arduino e *Processing*, com o objetivo de explorar o letramento computacional, a interdisciplinaridade, os conteúdos de ciências e o funcionamento do ultrassom e suas aplicações.

2. MARCO TEÓRICO

2.1. Robótica Educacional (RE)

Nós humanos sofremos influências do ambiente em que vivemos. Na recente conformação mundial, presenciamos uma imersão digital, por isso antigos comportamentos e costumes de produtores e receptores de conteúdo midiático e/ou cultural têm sofrido alterações (Souza; Gobbi, 2014, p.131.).

Não tão distante, a educação sofre com os impactos causados por essa nova era. Os estudantes da geração digital, vivem conectados e levam esse comportamento para o ambiente escolar, por isso, o professor precisa compreender que a tecnologia não é um enfeite dentro da sala de aula, mas sim um sistema digital para despertar nos alunos a vontade do aprender (Cunha; Cunha; Cunha, 2015, p.75.).

O estudante digital exerce diversas atividades simultaneamente, assim o professor se vê diante a um desafio: manter seu aprendiz conectado a sua aula. Para melhorar o ensino-aprendizagem e as novas formas de relação, o educador pode e deve utilizar de recursos tecnológicos.

A robótica educacional (RE) surge no ensino como uma ferramenta, pois ela permite que os estudantes tenham contato com diversos tipos de materiais, físicos ou virtuais, e desenvolvam pensamento lógico proporcionando uma melhor associação entre a realidade e os conteúdos da sala de aula, para Bezerra Neto et al. (2015), essas diferentes interações criam uma atmosfera de interesse nos alunos, viabilizando um ambiente interdisciplinar.

A robótica educacional é voltada a desenvolver projetos educacionais envolvendo a atividade de construção e manipulação de robôs, mas no sentido de proporcionar ao aluno mais um ambiente de aprendizagem, onde possa desenvolver seu raciocínio, sua criatividade, seu conhecimento em diferentes áreas, a conviver em grupos cujo interesse pela tecnologia e a inteligência artificial é comum a todos. (CASTILHO, 2002, p.4.).

A interdisciplinaridade é vista como um campo de integração das disciplinas e de conhecimentos.

A interdisciplinaridade não dilui as disciplinas, ao contrário, mantém sua individualidade. Mas integra as disciplinas a partir da compreensão das múltiplas causas ou fatores que intervêm sobre a realidade e trabalha todas as linguagens necessárias para a constituição de conhecimentos, comunicação e negociação de significados e registro sistemático dos resultados. (Brasil, 1999, p.89).

Segundo Grava Gomes et al. (2010), a robótica é uma área que agrega conhecimentos, permiti a conexão de diversas ciências e possui uma natureza interdisciplinar.

Com a RE busca-se atingir a aprendizagem do estudante de forma que ele desenvolva a lógica e consiga associar os conteúdos da sala de aula com o mundo a sua volta, ou seja, que o

aluno consiga visualizar uma situação cotidiana e perceber que um conteúdo de ciências ou de outras áreas se encaixa ali, tornando assim a realidade e a sala de aula mais próximas.

A robótica opera com o pensamento de lógica de programação, o que permite ao estudante desenvolver pensamento computacional, resolvendo na prática problemas de diversas áreas das ciências, como física, química e biologia.

O discente ao desenvolver esse tipo de pensamento se torna cada vez mais o autor de sua própria aprendizagem, a Robótica Educacional colabora com a prática pedagógica instigadora e motivadora da aprendizagem, desenvolvendo o raciocínio lógico e contribuindo para a união entre conhecimentos teóricos e experimentação prática. (Fornaza; Webber; Villas-Boas, 2015, p.1).

2.2. Arduino

De acordo com os criadores do Arduino (2011), ele surgiu em 2005 na Itália, e é um programa de *hardware* e *software open-source*, ou seja, aberto, que se tornou popular entre pessoas criativas, desenvolvedores de *Internet of Things* (IoT) e educadores que trabalham com uma abordagem transdisciplinar, *Science Technology Engineering Arts Mathematics* (STEAM).

Esse instrumento possui um site onde você encontra informações de como usar, ensinar, onde comprar, além de possuir uma grande comunidade internacional que compartilha informações, projetos e conhecimentos, o que o torna acessível e prático de aprender.

De acordo com Júnior (2014) o *Hardware*, é uma palavra inglesa que se refere a parte física, unidades funcionais, como por exemplo computador, telefone, microcontroladores.

No Arduino, os componentes físicos são, as placas:

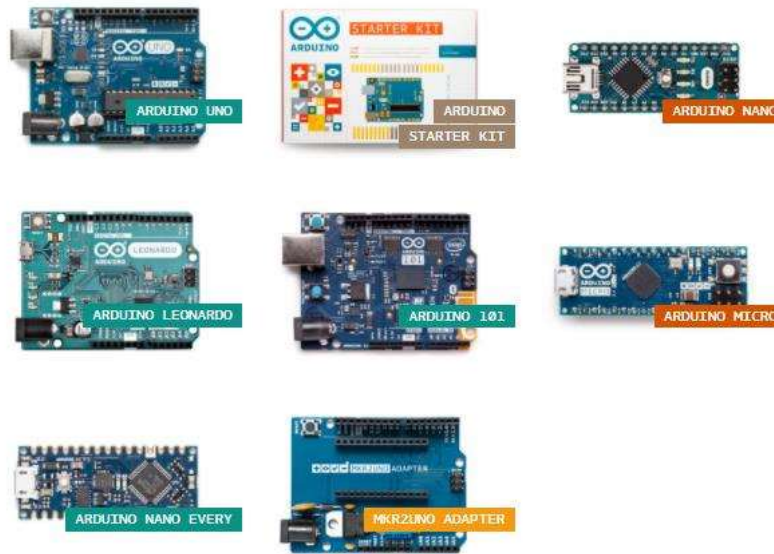


Figura 1- Alguns *Hardware* de Arduino (Fonte: <https://www.arduino.cc/en/Main/Products?from=Main.Hardware>)

As placas são de fácil utilização. Abaixo apresentarei mais detalhadamente o Arduino UNO:

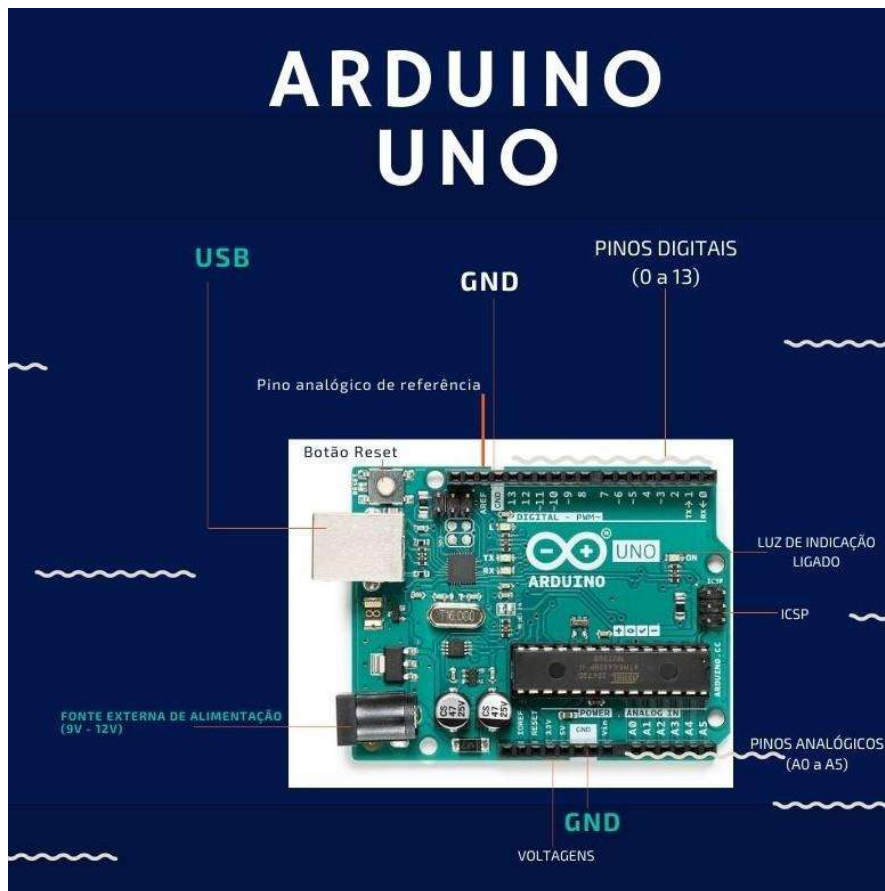


Figura 2- Especificações do *Hardware*, placa Arduino (Fonte: própria)

Para a FBS Eletrônica, a placa do Arduino funciona como um circuito microcontrolador, ou seja, um computador pequeno com potencial menor de processamento, por isso precisa se comunicar com um computador maior na hora da programação. Para essa comunicação usamos a conexão USB.



Figura 3- Cabo USB do Arduino UNO (Fonte: <https://store.arduino.cc/usa/usb-2-0-cable-type-a-b>)

Como dito anteriormente o Arduino possui um *Hardware* e um *Software*. O *software* para o dicionário Aurélio é um conjunto de elementos em um computador compondo um sistema que processa dados.

O *software* do Arduino é Integrated Development Environment (IDE), no *sketch* será utilizado a linguagem de programação, que é modelada a partir da linguagem *processing* (FBS Eletrônica, sem ano).

Para obter o Arduino IDE você precisa acessar o site <https://www.arduino.cc/en/Main/Software#> e realizar o download.

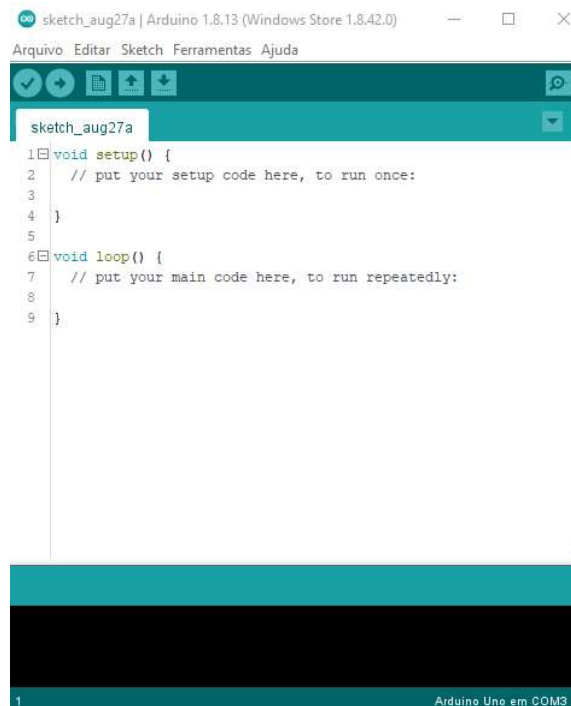


Figura 4- *Sketch* do Arduino IDE. (Fonte: Própria).

A linguagem C/C++, começou com a linguagem BCPL, por Martin Richards, que rodava no computador DEC PDP-7, com sistema operacional Unix (Villas-Boas, 2001). Essa linguagem está sempre passando por mudanças, o avanço tecnológico a alavancou e cobrou mudanças.

Esse tipo de linguagem popularizou e hoje é uma das mais utilizada, quaisquer pessoas podem trabalhar com ela ou modifica-la. Não há um “dono” desta linguagem. Há vários sistemas operacionais e compiladores 100% gratuitos que usam o padrão C/C++ (Villas-Boas, 2001).

O Arduino se promoveu por ser um equipamento de fácil acesso e aprendizagem, com isso se tornou uma ferramenta importante para a robótica. No âmbito educacional a plataforma Arduino UNO é uma das mais utilizadas, permite uma melhor visualização e é uma das mais populares.

O Parâmetro Curricular Nacional prevê, alunos aptos e incorporados nas novas tecnologias, o pensamento computacional passou a ser uma temática significativa. Pensamento computacional envolve a resolução de problemas, projeção de sistemas, e compreensão do comportamento humano (Wing, 2016).

O Arduino é uma fonte de operação do pensamento lógico, ficou mais prático de desenvolver a robótica educacional nas escolas, com propostas de desenvolver projetos ligados a situações do dia-a-dia ou fictícias, permitindo um salto na aprendizagem do aluno.

Esse equipamento pode ser usado como uma técnica de ensino em potencial, podendo ser associada aos métodos de ensino colaborativos. Lupion Torres et al. (2014) relata que o ensino colaborativo é uma metodologia capaz de promover uma aprendizagem mais ativa, estimulando o pensamento crítico, melhorando as interações e resoluções de problemas.

2.3. Processing

O *Processing* é um *software* de código aberto, que permite a construção de artes. *Processing* é um caderno de esboço com *software* flexível e com uma linguagem para aprender a codificar no contexto das artes visuais. (Tradução livre, Processing.org 2001)

2.4. BNCC

A Base Nacional Comum Curricular (BNCC) é um documento elaborado por pesquisadores, educadores e sociedade, que busca proporcionar ao Brasil uma educação de qualidade. A Base deve nortear os currículos dos sistemas e redes de ensino das Unidades Federativas, como também as propostas pedagógicas de todas as escolas públicas e privadas. (BNCC, 2018)

Ciências da Natureza no ambiente escolar é responsável por apresentar o desenvolvimento científico e tecnológico, elaborando debates que tornem os estudantes cidadãos capazes de captar o mundo a sua volta, modificar e produzir soluções.

Para debater e tomar posição sobre alimentos, medicamentos, combustíveis, transportes, comunicações, contracepção, saneamento e manutenção da vida na Terra, entre muitos outros temas, são imprescindíveis tanto conhecimentos éticos, políticos e culturais quanto científicos. Isso por si só já justifica, na educação formal, a presença da área de Ciências da Natureza, e de seu compromisso com a formação integral dos alunos. (BNCC, 2018)

A BNCC entende que ciências da natureza, ou, ciências naturais para o ensino fundamental devem priorizar o letramento científico, tornar o estudante apto ao mundo e consciente ao exercer a cidadania.

No documento, os conteúdos de Ciências Naturais estão divididos em unidades temáticas, que serão trabalhadas durante todo o ensino fundamental.

Matéria e energia é uma temática que contempla o estudo de materiais e suas transformações, fontes e tipos de energia (BNCC, 2018). Para os anos finais essa unidade possibilita desenvolver a compreensão de como os recursos naturais são utilizados e qual seu impacto ambiental e busca reflexões mais sustentáveis.

A unidade temática Vida e evolução propõe o estudo de questões relacionadas aos seres vivos (BNCC, 2018), que vai da percepção das relações até o corpo humano. Ao final do ensino fundamental o discente deve ser conhecedor do funcionamento do seu corpo, autocuidado, impactos emocionais e físicos da adolescência e a importância do papel do Estado.

Terra e universo é a unidade que trabalha o sistema solar, corpos celestes, características da Terra, Sol e Lua (BNCC, 2018). Ao trabalhar essa unidade espera-se que o estudante conheça o local em que mora, Terra, e o que se encontra ao seu redor, espaço.

Para a base nacional comum curricular as temáticas devem ser trabalhadas de forma integrativa, conectadas. Isso faz com que o aprendiz consiga visualizar de forma abrangente o que está sendo trabalhado, desenvolvendo assim um pensamento coletivo.

2.5. Ciências e Arduino. Como trabalhar?

Para trabalhar ciências e Arduino recomenda-se olhar as habilidades disponíveis na BNCC, elaborar, estudar a montagem e execução do projeto.

Ao analisar a BNCC, foi perceptível que vários projetos poderiam ser elaborados, como por exemplo, trabalhar com o 8º ano fontes de energia e montagens de circuitos elétricos, explorar energia solar, previsão do tempo ou temperatura. Com o 7º ano poderia ser explorado juntamente ao sensor MQ135 a composição do ar.

Segundo a BNCC, no ensino de ciências o estudo sobre ultrassom está designado ao 9º ano e faz parte da habilidade EF09CI07.

2.6. Ultrassom

A luz e o som são, ambos, vibrações que se propagam através do espaço como ondas (Hewitt, 2009). O som precisa de um meio para se propagar, diferente da luz que pode se propagar no vácuo.

De acordo com Hewitt, o som sofre processo de reflexão, pois ao encontrar um objeto ele pode retornar ou continuar, o processo de retorno é conhecido como eco.

O ouvido humano pode escutar normalmente sons na faixa de 20 hertz a 20.000 hertz (Hewitt, 2009). As ondas sonoras não audíveis são: ondas infrassônicas se encontram na faixa abaixo de 20Hz e ondas ultrassônicas, que possuem frequências acima de 20.000 hertz e se propagam no ar ou qualquer outra matéria elástica.

Alguns animais como morcegos e golfinhos possuem um sistema natural de ecolocalização, eles emitem o sinal ultrassônico que ao bater em um objeto reflete, esse mecanismo de reflexão permite que o animal saiba a localização do objeto.

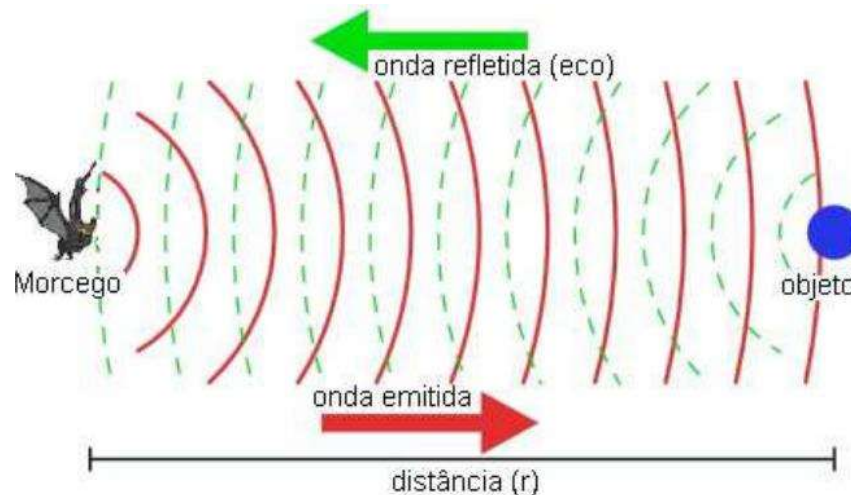


Figura 5- Ecolocalização. (Fonte: Coelho; Júnior; Dias; Marcato, 2012)

A observação dos animais e experimentos levaram a diversas descobertas sobre o ultrassom, em 1880, os irmãos Pierre e Jacques Curie descobriram o efeito piezoeletrico (Santos; Amaral, 2012) que é resultado da pressão em alguns cristais, gerando ondas acima de 20 kHz.

Os transdutores ultrassônicos tem cristais de piezoeletricos que ressonam a uma frequência desejada e convertem energia elétrica em energia acústica e vice-versa. As ondas sonoras são transmitidas e refletidas. (THOMAZINI, 2018).

Segundo Santos e Amaral, os pesquisadores também descobriram por meio da aplicação do ultrassom nos cristais que o som refletido voltava com informações sobre o objeto que encontrou.

O ultrassom está presente em diversas áreas, nos submarinos são usados para detectar objetos, na medicina é usado como tratamento e se expandiu para a produção de imagens, ultrassonografia e a ecografia.

Na medicina o uso do ultrassom evoluiu, os estudos e desenvolvimentos prosseguiram até culminar no ultrassom 3D. Hoje se encontra muito presente na medicina, seu baixo custo e praticidade tornou-o popular.

O exame ultrassonográfico acomoda-se confortavelmente ao paciente, onde o transdutor é o responsável por transformar os ecos refletidos pelo interior do corpo humano em sinais que serão decodificados eletronicamente em uma imagem que, por sua vez, será

interpretada pelo médico que estiver realizando o exame. (Santos, 2012)

Na Segunda Guerra Mundial, o infrassom e o ultrassom se desenvolveram para o uso militar. O infrassom foi usado para localizar armas inimigas: foram construídos dispositivos de rastreabilidade que “escutassem” o infrassom de trens a vários quilômetros de distância (Janzen, 2017), ele também foi cogitado para se tornar uma arma silenciosa, capaz de gerar infrassom a elevados níveis, a fim de destruir construções e até exércitos inimigos, e tudo isso sem qualquer explosão ou barulho (Dietrich, 1968, apud Janzen, 2017).

O sonar foi desenvolvido para fins militares, para Ristow sua função é mapear e monitorar áreas submersas. Ele é uma tecnologia similar ao radar, a diferença entre os dois está no tipo de ondas utilizadas, no sonar a energia é transferida por ondas acústicas que se propagam na água, no radar é utilizado ondas eletromagnéticas (Federation of American Scientists, 1998. Tradução livre).

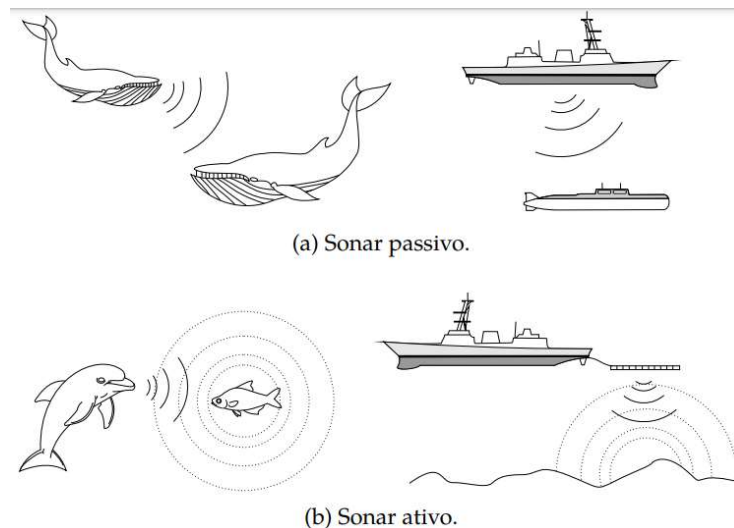


Figura 6- Exemplos de sonar. (Fonte: Ristow, 2015)

O radar é um dispositivo que utiliza as ondas de rádio refletidas em objetos para calcular sua posição (Ristow, 2015).

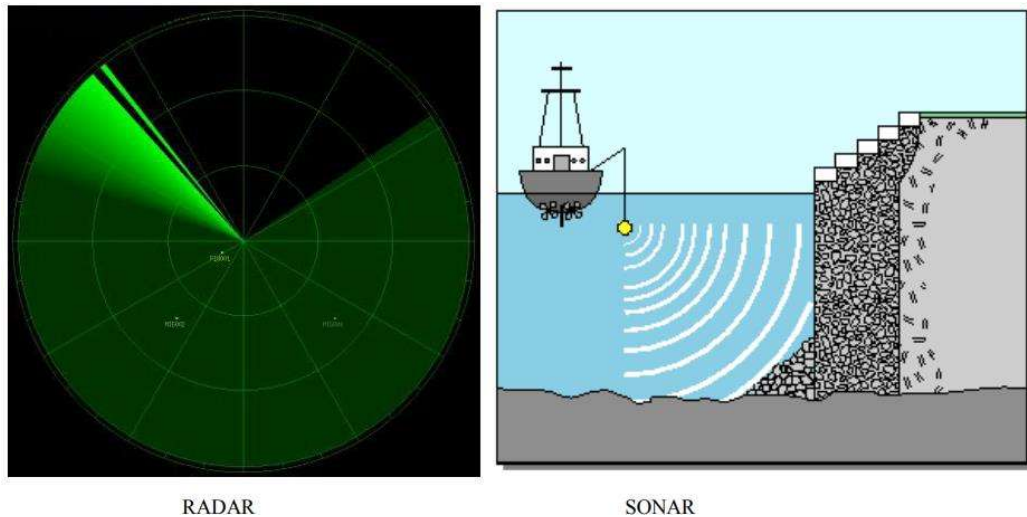


Figura 7- Exemplo de Radar e Sonar. (Fonte: Santos; Amaral, 2012)

3. OBJETIVOS

3.1- Objetivo Geral:

- Construir um projeto educacional utilizando a plataforma Arduino e *Processing*.

3.2- Objetivos Específicos:

- Discutir a utilização do projeto no ensino de ciências e sua face multidisciplinar.
- Apresentar a construção e descrição do funcionamento de um radar ultrassônico.

4. METODOLOGIA

Para que os estudantes utilizem o Arduino, recomenda-se que o professor ministre aulas de conceitos básicos, para depois iniciar o projeto que será apresentado aqui.

4.1. Projeto Radar ultrassônico

Este projeto consiste em montar um radar ultrassônico a fim de simular o funcionamento de um radar e sua aplicação.

4.1.1. Materiais utilizados

- Uma protoboard (Fig. 8-1)
- Um sensor Ultrassônico HC-SR04 (Fig. 8-2)
- *Processing* (Fig. 8-3)
- Arduino IDE (Fig. 8-4)
- Um computador (Fig. 8-5)

- Jumpers (Fig. 8-6)
- Um servo motor 9g (Fig. 8-7)
- Uma placa de Arduino UNO (Fig. 8-8)
- Um cabo USB (Fig. 8-9)

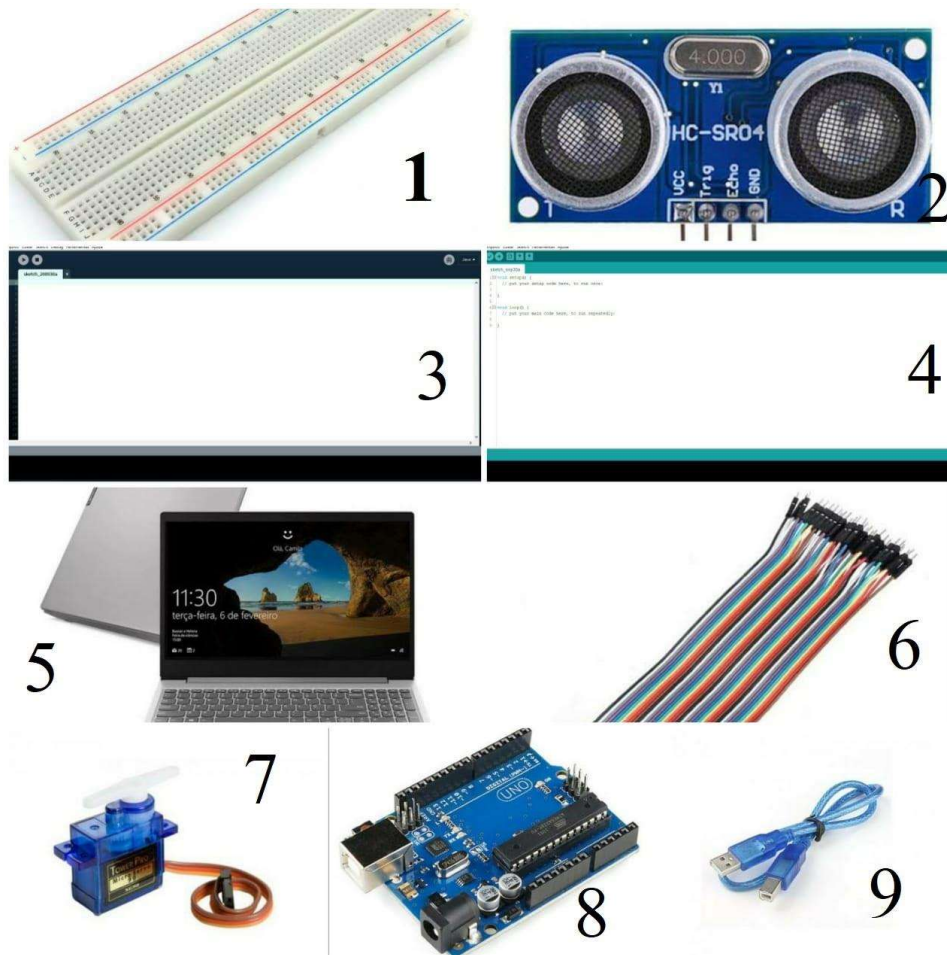


Figura 8- Componentes do projeto. (Fonte: Própria)

4.1.2. Construção

Para demonstrar a montagem foi utilizado o site *tinkercad*. Nele o usuário pode montar todo o projeto e programar o código. O site se torna uma opção para quem quer montar algum projeto e não tem acesso as peças ou apenas deseja esboçar a ideia antes de executar.

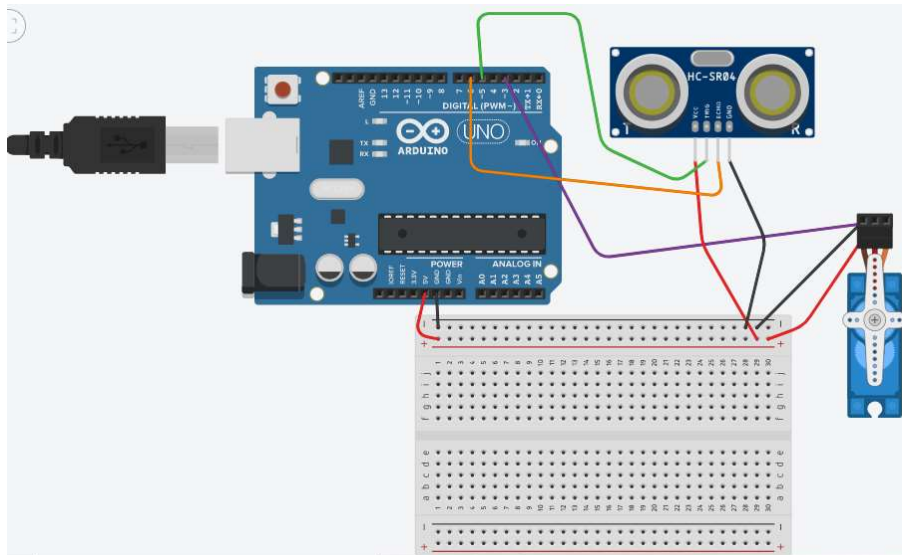


Figura 9- Esquema da montagem do projeto feito no *tinkercad*. (Fonte: Própria)

Montando o projeto passo a passo:

1. Placa Arduino e Protoboard

A *Protoboard* é uma placa com diversos furos que permite realizar conexões sem que realize soldas. A placa possui duas faixas de barramento que estão organizadas de forma vertical, nela conectamos o aterramento (GND) e a voltagem, e na sua parte central se encontra as faixas de terminais que estão organizadas de forma horizontal, nela conectamos os componentes.

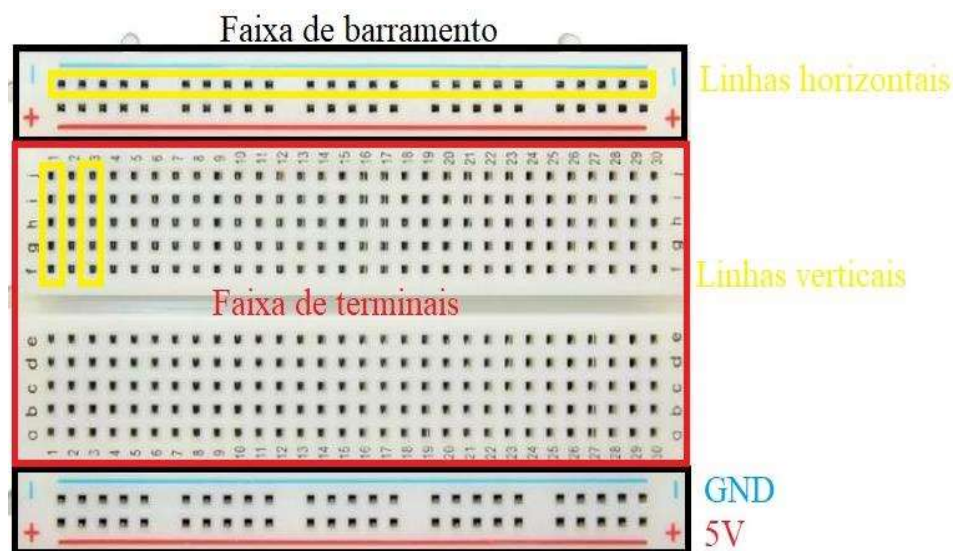


Figura 10- *Protoboard*. (Fonte: Própria)

Para iniciar a montagem do projeto deve-se conectar a placa Arduino UNO a um protoboard. Nesta etapa, ligamos o aterramento, *Ground* (GND), a linha horizontal que corresponde ao negativo e ligamos o 5V à linha horizontal que corresponde ao positivo.

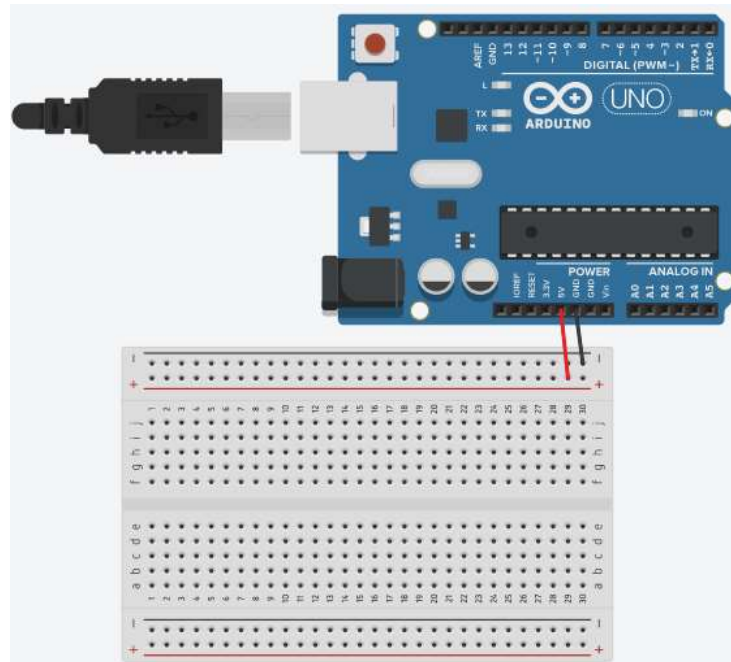


Figura 11- Conexão placa Arduino e *protoboard* feito no *tinkercad*.. (Fonte: própria)

2. Servo Motor

O servo motor é um componente importante para movimentos de 180° e movimentação de robôs.

O servo motor tem que ser conectado a *Protoboard* e a placa Arduino.

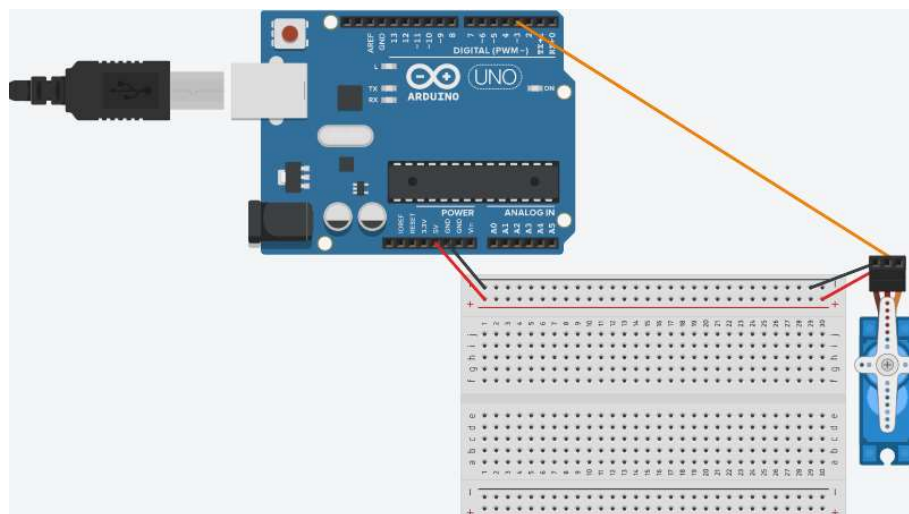


Figura 12- Conexão servo motor, *protoboard* e placa Arduino. (Fonte: própria)

3. Sensor Ultrassônico HC-SR04

O sensor ultrassônico é um componente que emite sinais de ultrassom que ao entrar em contato com um objeto são refletidos e o sensor capta o retorno. Esse mecanismo permite que o sensor seja capaz de medir distâncias.

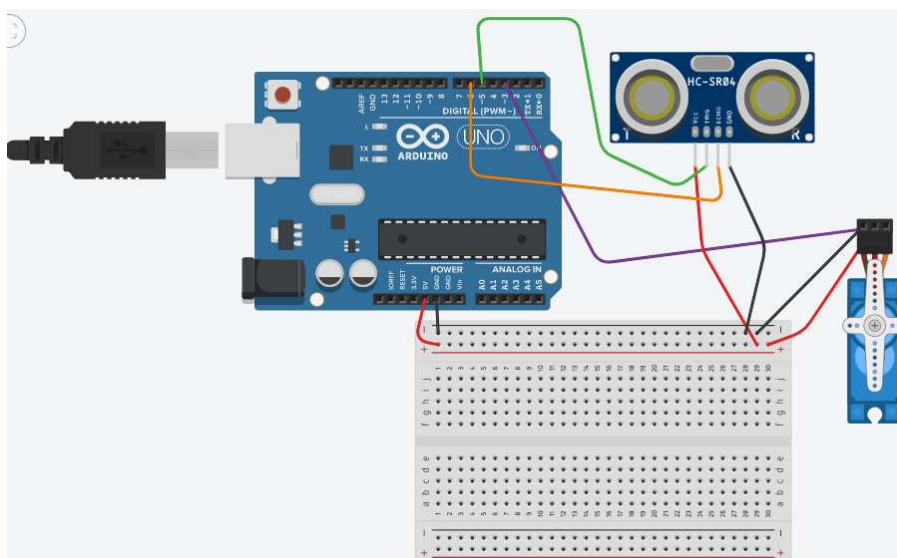


Figura 13- Conexão sensor ultrassônico, *protoboard* e placa Arduino. (Fonte: própria)

Após a montagem no protoboard, a segunda etapa é programar usando os programas Arduino IDE e *Processing*.

Para programar é utilizado a linguagem C/C++. O site tinkercad é uma ferramenta que simula o Arduino e permite realizar programação, para quem não possui os componentes necessários é um excelente instrumento. A programação completa se encontra nos anexos.

Programando passo a passo em *Void setup*:

1. Servo motor

Para iniciar a programação deve-se incluir a biblioteca do servo motor, para isso você deve, na barra de opções do IDE, ir em *Sketch*, clicar no item “incluir biblioteca” e selecionar servo.

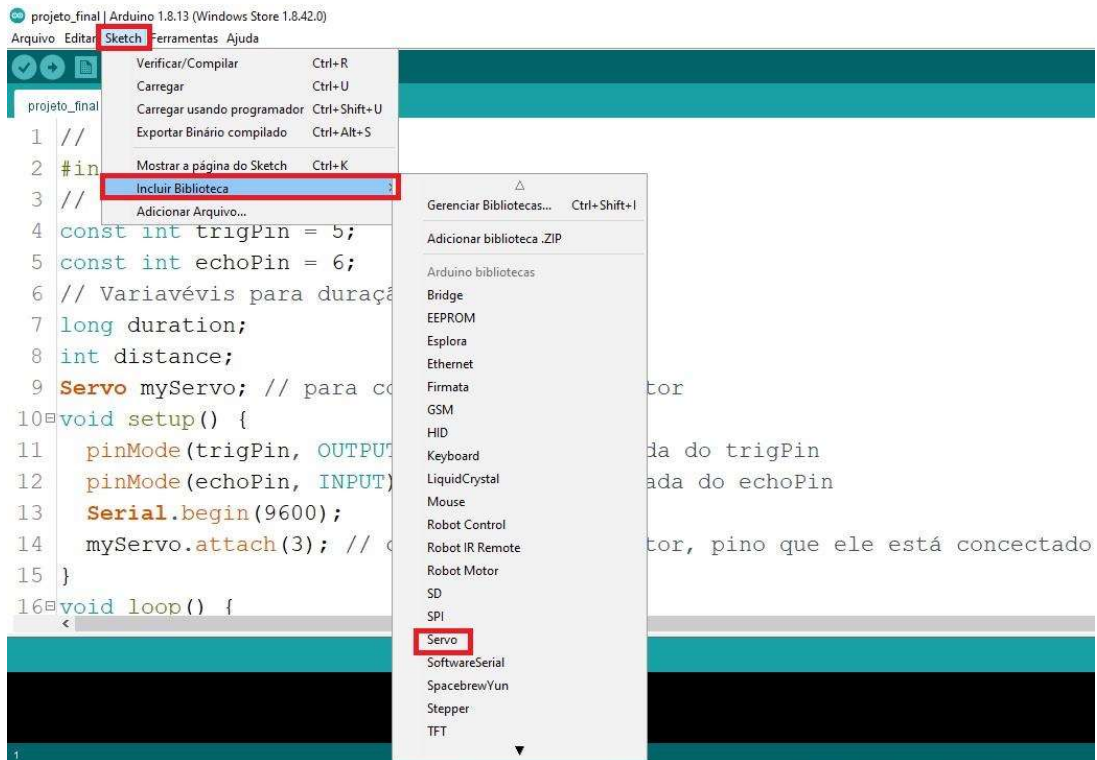


Figura 14- Inclusão da biblioteca servo. (Fonte: Própria).

Após incluir a biblioteca define-se em *void setup* a porta 3 que o servo motor está conectado.

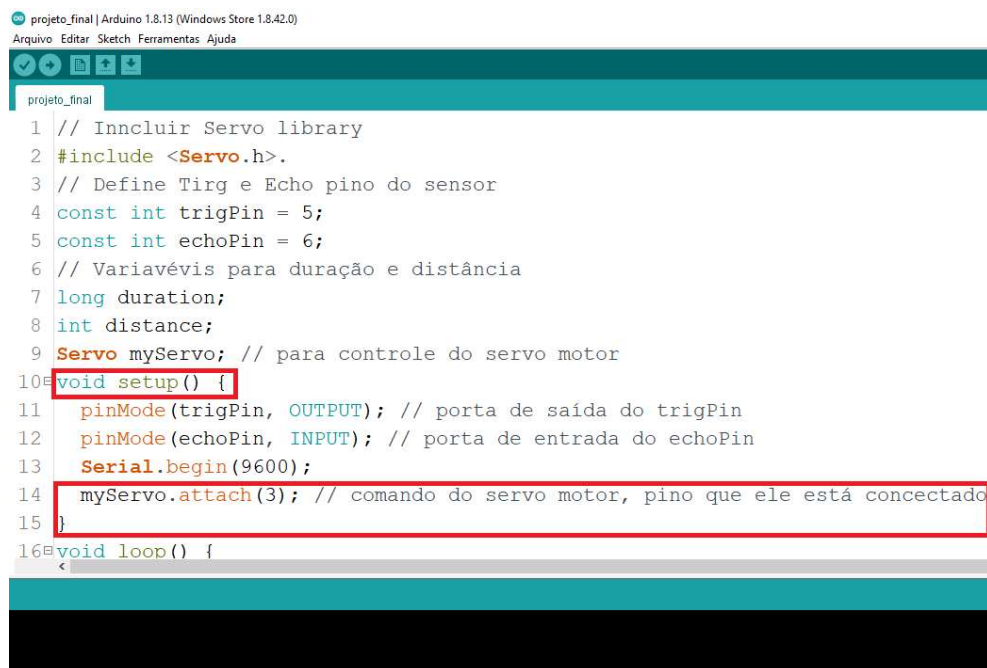
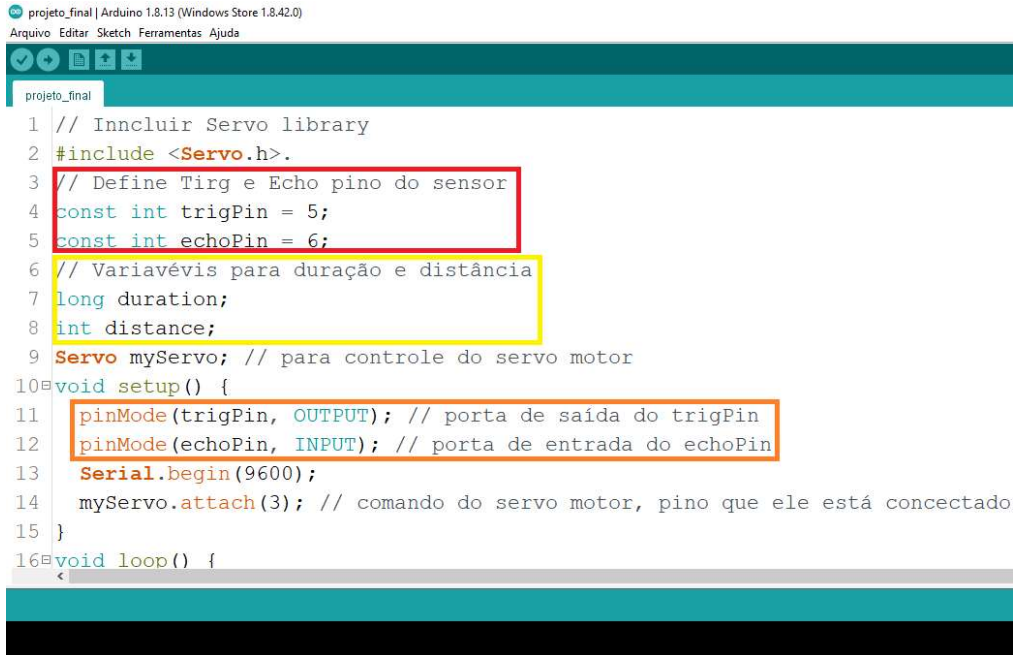


Figura 15- Programação do servo motor. (Fonte: Própria).

2. Sensor Ultrassônico

Para programar o sensor ultrassônico define-se as portas que os pinos *Echo* e *Trig* estão conectadas, também deve-se definir as variáveis para duração e distância. Na parte *Void setup* programa-se *trigPin* como *OUTPUT*, pois é uma porta de saída, ou seja, componente que envia o sinal e programa-se *echoPin* como *INPUT*, pois é uma porta de entrada, ou seja, componente que recebe o sinal.



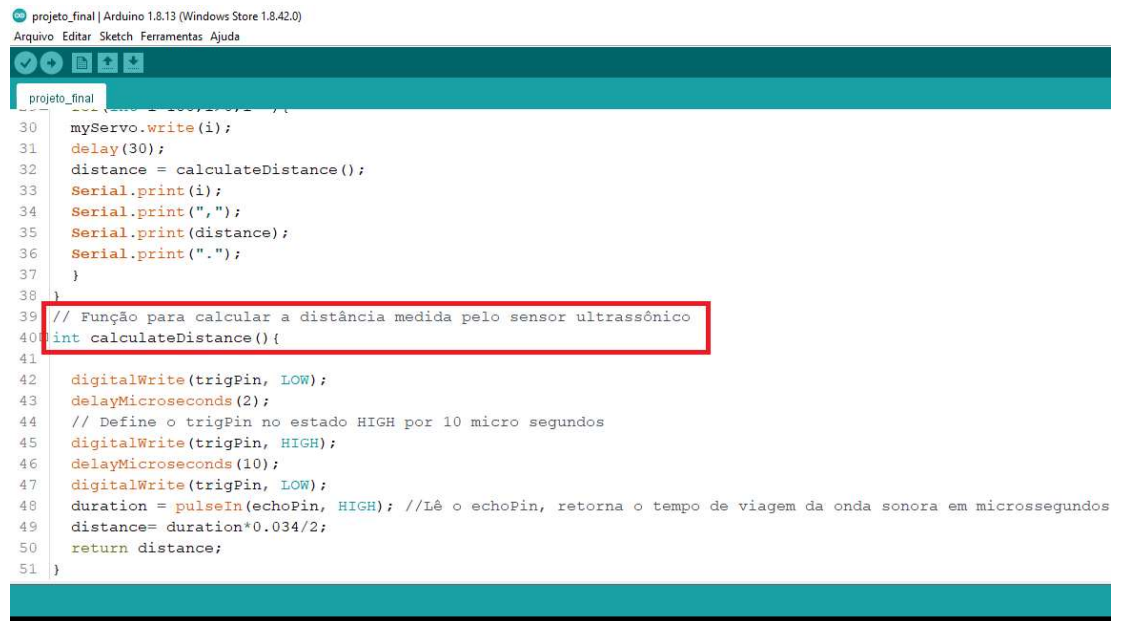
```
projeto_final | Arduino 1.8.13 (Windows Store 1.8.42.0)
Arquivo Editar Sketch Ferramentas Ajuda
projeto_final
1 // Incluir Servo library
2 #include <Servo.h>.
3 // Define Trig e Echo pino do sensor
4 const int trigPin = 5;
5 const int echoPin = 6;
6 // Variáveis para duração e distância
7 long duration;
8 int distance;
9 Servo myServo; // para controle do servo motor
10 void setup() {
11   pinMode(trigPin, OUTPUT); // porta de saída do trigPin
12   pinMode(echoPin, INPUT); // porta de entrada do echoPin
13   Serial.begin(9600);
14   myServo.attach(3); // comando do servo motor, pino que ele está conectado
15 }
16 void loop() {
<
```

Figura 16- Programação de Trig e Echo. (Fonte: Própria).

3. Serial.begin

Configura a taxa de transferência em *bits* por segundo (*baud rate*) para a transmissão serial. (Arduino, 2011).

2. Função para calcular a distância medida pelo sensor:



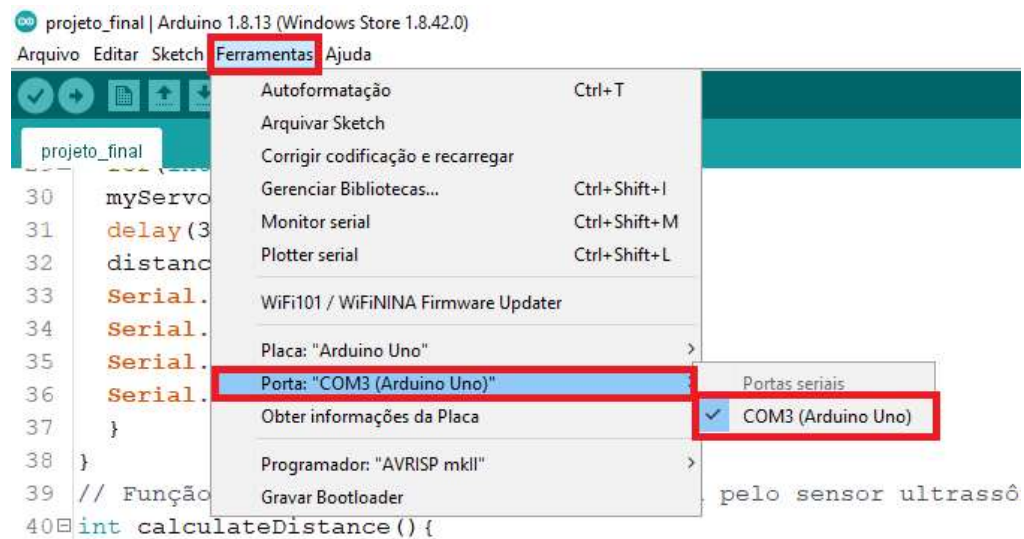
```
projeto_final | Arduino 1.8.13 (Windows Store 1.8.42.0)
Arquivo Editar Sketch Ferramentas Ajuda

projeto_final
30 myServo.write(i);
31 delay(30);
32 distance = calculateDistance();
33 Serial.print(i);
34 Serial.print(",");
35 Serial.print(distance);
36 Serial.print(".");
37 }
38 }
39 // Função para calcular a distância medida pelo sensor ultrassônico
40 int calculateDistance() {
41
42     digitalWrite(trigPin, LOW);
43     delayMicroseconds(2);
44     // Define o trigPin no estado HIGH por 10 micro segundos
45     digitalWrite(trigPin, HIGH);
46     delayMicroseconds(10);
47     digitalWrite(trigPin, LOW);
48     duration = pulseIn(echoPin, HIGH); //Lê o echoPin, retorna o tempo de viagem da onda sonora em microssegundos
49     distance= duration*0.034/2;
50     return distance;
51 }
```

Figura 19- Cálculo da distância. (Fonte: Própria).

Programando a Conexão Arduino e *Processing*:

Para estabelecer a conexão entre Arduino e *Processing*, primeiramente deve-se descobrir qual porta “COM” a placa está conectada.



```
projeto_final | Arduino 1.8.13 (Windows Store 1.8.42.0)
Arquivo Editar Sketch Ferramentas Ajuda

projeto_final
30 myServo
31 delay(3
32 distanc
33 Serial.
34 Serial.
35 Serial.
36 Serial.
37 }
38 }
39 // Função
40 int calculateDistance() {

Autoformatação Ctrl+T
Arquivar Sketch
Corrigir codificação e recarregar
Gerenciar Bibliotecas... Ctrl+Shift+I
Monitor serial Ctrl+Shift+M
Plotter serial Ctrl+Shift+L
WiFi101 / WiFININA Firmware Updater
Placa: "Arduino Uno" >
Porta: "COM3 (Arduino Uno)" >
Obter informações da Placa
Programador: "AVRISP mkII" >
Gravar Bootloader

Portas seriais
COM3 (Arduino Uno)

peelo sensor ultrassô
```

Figura 20- Porta COM. (Fonte: Própria).

No IDE do *processing* deve-se colocar a porta “COM” em que a comunicação serial se encontra.


```
projeto_final_pro | Processing 3.4
Arquivo Editar Sketch Debug Ferramentas Ajuda

projeto_final_pro
7
8 /* define algumas variaveis.. */
9 String angle="";
10 String distance="";
11 String data="";
12 String noObject;
13 float pixsDistance;
14 int iAngle, iDistance;
15 int index1=0;
16 int index2=0;
17 //fonte de texto
18 PFont myFont;
19
20 /**
21 * setup -
22 */
23 void setup() {
24
25   size (1024, 700); /* ***Alterar para resolução que desejar*** */
26   smooth();
27   myPort = new Serial(this, "COM3", 9600); /* inicia comunicação serial. */
28   myPort.bufferUntil('.'); /* Lê os dados através da porta serial (ângulo e distância) */
```

Figura 21- IDE *processing*. (Fonte: Própria).

Os códigos aqui utilizados se encontram no anexo a e anexo b. Os códigos tanto do Arduino IDE quanto do *Processing 3.4* foram retirados do site filipeflop.com. Os códigos sofreram pequenas alterações a fim de possibilitar a realização do trabalho.

4.1.3. Funcionamento

O sensor utilizado no projeto foi o HC-SR04, que possui um emissor de sinal, *trigger*, e um receptor, *echo*.

O sensor ultrassônico é capaz de emitir ondas de alta frequência e captar a sua reflexão, ou seja, um sensor emite pulsos ultrassônicos que se propagam pelo ar e ao encontrar um objeto, líquido ou sólido, são refletidos. Esse retorno é denominado de eco. Essa característica de reflexão possibilita o cálculo da distância do objeto.

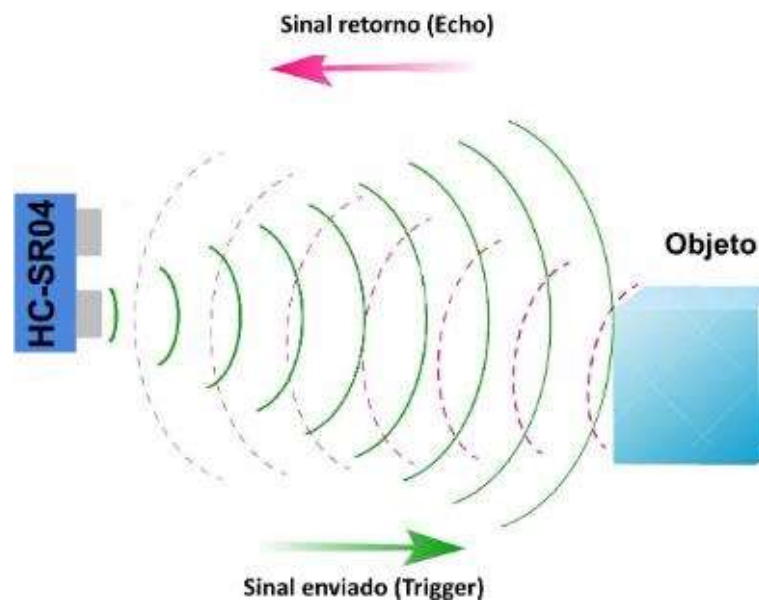


Figura 22-Esquema de emissão e recepção de sinal. (Fonte: <https://www.filipeflop.com/>).

O servo motor é responsável por girar o sensor HC-SR04 entre 0° e 180° e o sensor ao emitir sinais e receber ecos refletidos de um objeto marca na tela a posição e a distância que o objeto se encontra.

A comunicação serial entre o programa *Processing* e o Arduino permitem a marcação da distância e posição do objeto de forma gráfica.

Você pode ver o funcionamento desse projeto acessando: Radar Ultrassônico com Arduino e Processing: (<https://www.youtube.com/watch?v=bCF9RuPXrMA>)



5. DISCUSSÃO

Depois de realizar a montagem do projeto e analisar o seu funcionamento, refletiremos em sua importância para o ensino de Ciências.

No primeiro momento da montagem do projeto, a respeito da placa e do *proto-board*, o professor pode abordar assuntos como aterramento, conhecido no Arduino como GND, sigla correspondente a *ground*, que significa terra, também pode falar sobre tipos de circuitos elétricos, voltagem e corrente elétrica.

À medida que a construção avança surge temas sobre ultrassom, som e infrassom, quais suas diferenças, utilizações e importância.

O som é uma onda que pode ser classificada como audível, quando se encontra aproximadamente na faixa de 20 hertz a 20.000 hertz. Para frequências acima dessa faixa, o som é conhecido como ultrassom e para abaixo dela, infrassom, ambas são as ondas sonoras não audíveis para os humanos, no entanto, diversos animais dispõem da capacidade de escutar nessas faixas.

Ao final, o professor pode trabalhar com o tema eco, velocidade do som, distância, tempo de ida e retorno do ultrassom etc. Ou mesmo, é possível explorar o tema ângulo, arco, área varrida pelo radar ultrassônico, demonstrando assim o grande potencial de se trabalhar com robótica na sala de aula, isso por que não há somente os conteúdos de ciências, o que possibilita que o ensino seja interdisciplinar abrangendo também a matemática, por exemplo.

No marco teórico foi citado que o radar ultrassônico abrangia segundo a BNCC o tema ultrassom que está designado ao 9º ano, porém ao analisarmos a multidisciplinariedade do projeto é possível afirmar que ele também pode ser designado no ensino de medidas de ângulos para o 6º ano, cujo assunto está explícito na habilidade EF06MA27 - determinar medidas da abertura de ângulos, por meio de transferidor e/ou tecnologias digitais (BNCC, 2018).

A construção e programação de um robô exige a combinação de conhecimentos de diversas áreas, o que dá à robótica um caráter multidisciplinar. (Brito; Moita; Lopes, 2018).

6. CONCLUSÃO

Retornando a pergunta da Introdução: Educação e robótica combinam? O que se pode esperar de um projeto educacional construído com *software* e *hardware*? Ao chegar até aqui provavelmente é possível vislumbrar algumas respostas.

Educação e robótica combinam, esse conjunto ao ser bem administrado potencializa o ensino e leva a uma aprendizagem ampla e significativa. A educação é um direito de todos e o artigo 205 da constituição federal de 1988 frisa isso, sendo assim ela deve ser ofertada de forma gratuita e com qualidade e todos possuem o direito de cobrar isso.

Ao construir o radar ultrassônico apresentamos o seu funcionamento e a sua importância na educação. O projeto, que envolve *software* e *hardware*, oferece contribuições para o ensino de ciências, matemática, linguagem computacional etc, e levanta outros aspectos como resolução de problemas, demonstrando assim a sua interdisciplinaridade.

Por fim após perceber a importância do uso da robótica educacional e do Arduino no ensino, podemos e devemos ir mais além. Robótica educacional, Arduino e trabalho colaborativo em equipe combinam? Provavelmente sim, mas isso é assunto para os próximos capítulos da história educacional.

7. REFERÊNCIAS BIBLIOGRÁFICAS

Apostila Arduino: Com aplicações baseada na placa: Arduino UNO, FBS E. Disponível em: < <http://www.valdick.com/files/ApostilaArduinoIntroducao.pdf> >. Acesso em: 10/08/2020.

Arduino - *AboutUs*. <https://www.arduino.cc/en/Main/AboutUs>. Acessado 18 de abril de 2019.

BRASIL. Base Nacional Comum Curricular (BNCC). Educação é a Base. Brasília, MEC/CONSED/UNDIME, 2018. Disponível em: < http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf >. Acesso em: 07 out. 2020.

BRASIL. Constituição Federal do Brasil. Brasília: Senado, 1988.

BRASIL. MINISTÉRIO DA EDUCAÇÃO. Parâmetros Curriculares Nacionais: ensino médio. Brasília: 1999.

BRASIL. Secretaria de Educação Fundamental. Parâmetros curriculares nacionais: ciências naturais / Secretaria de Educação Fundamental. – Brasília: MEC/SEF, 1997. 136p.

CASTILHO, M.I. Robótica na Educação: Com que objetivos? 2002. Monografia de Especialização em Informática na Educação.

COELHO, F.C.R.; JUNIOR, I.C.S.; DIAS, B.H.; MARCATO, A.L.M; Metaheurística Inspirada na Ecolocalização de Morcegos: Aperfeiçoamento e Estudos de Casos. Congresso Latino-Iberoamericano de Investigación Operativa. Rio de Janeiro, 2012.

CUNHA, N.B.; CUNHA, N.C.; CUNHA, T.N.B.; Geração digital- Transformação pedagógica. Cadernos da Fucamp, v.14, n.20, p.74-83. 2015.

FEDERATIO OF AMERICAN SCIENTISTS. Introduction to Naval Weapons Engineering, 1998. Disponível em: < https://fas.org/man/dod-101/navy/docs/es310/uw_acous/uw_acous.htm >. Acesso em: 09 nov. 2020.

FORNAZA, R.; Webber, C.G.; VILLAS-BOAS, V. Kits Educacionais de Robótica: opções para o Ensino de Ciências. SCIENTIA CUM INDUSTRIA (SCI. CUM IND.), Centro de Ciências Exatas e da Tecnologia - Universidade de Caxias do Sul, v. 3, n. 3, p. 142-147, 2015.

GOMES, C.G.; SILVA, F.O.; BOTELHO, J.C.; SOUZA, A.R. A robótica como facilitadora do processo ensino-aprendizagem de matemática no ensino fundamental. PIROLA, NA. org. Ensino de ciências e matemática, IV: temas de investigação [online]. São Paulo: Editora UNESP; São Paulo: Cultura Acadêmica, 2010. 244 p. ISBN 978-85-7983-081-5. Available from SciELO Books.

HEWITT, P.G.; Fundamentos de Física Conceitual. Grupo A, 2009.

JANZEN, S.; Estudo da Geração de Infrassom com Diferentes Tipos de Transdutores. Trabalho de Conclusão de Curso. Curitiba, 2017.

JÚNIOR, M.; Introdução a informática: Hardware e Software. INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE DO NORTE CAMPUS MACAU. 2014.

NETO, R.P.B.; ROCHA, P.D.; SANTANA, A.M.; SOUZA, A.A de S.; Robótica na Educação: Uma Revisão Sistemática dos Últimos 10 Anos. Anais do XXVI Simpósio Brasileiro de Informática na Educação (SBIE 2015). P 386-393. 2015.

PROCESSING FOUNDATION. *Processing*, 2001. Página inicial. Disponível em: <<https://processing.org/>>. Acesso em: 06 jan. 2020.

RISTOW, J.P.; Estudo e Desenvolvimento de Algoritmos de Sonar Ativo para Mapeamento de Áreas Submersas. Dissertação de Mestrado. Florianópolis, 2015.

SANTOS, C.O.S.; AMARAL, W.N.; TACON, K.C.B.; A história da ultrassonografia no Brasil e no mundo. EFDeportes.com, Revista Digital. Buenos Aires, Ano 17, Nº 167, abril de 2012. <http://www.efdeportes.com/>.

SANTOS, H.C.O.; AMARAL, W.N.; A História da Ultrassonografia no Brasil. 2012. Disponível em: < <http://doutormedicamentos.com.br/wm/admin/upload/1103114623livro-ultrassonografia.pdf> >. Acesso em: 09 nov. 2020

SILVA, A.F.; RoboEduc: Uma Metodologia de aprendizado com Robótica Educacional. Natal, RN. 2009.

software. In: DICIO, Dicionário Online de Português. Porto: 7Graus, 2020. Disponível em: <<https://www.dicio.com.br/software/>>. Acesso em: 27/08/2020.

SOUSA, J.F.; GOBBI, M.C.; Geração digital: Uma reflexão sobre as relações da “juventude digital” e os campos da comunicação e da cultura. Revista GEMInIS, ano 5 - n. 1 - v. 2, p. 129-145. 2014.

THOMAZINI, D; Sensores Industriais: Fundamentos e Aplicações. 8ª Ed. Editora Érica. 2018.

THOMSEN, A.; Como conectar o Sensor Ultrassônico HC-SR04 ao Arduino, 2011. Blog. Disponível em: < <https://www.filipeflop.com/blog/sensor-ultrassonico-hc-sr04-ao-arduino/> >. Acesso em: 27 nov. 2018.

VILLAS-BOAS, S.B.; C / C++ e Orientação a Objetos em Ambiente Multiplataforma. Versão do Livro: 5.1, de 17 de agosto de 2001.

WING, J.; PENSAMENTO COMPUTACIONAL – Um conjunto de atitudes e habilidades que todos, não só cientistas da computação, ficaram ansiosos para aprender e usar. R. bras. Ens. Ci. Tecnol., Ponta Grossa, v. 9, n. 2, p. 1-10, mai./ago. 2016.

ANEXO A- CÓDIGO ARDUINO IDE

```
// Incluir Servo library
#include <Servo.h>.

// Define Trig e Echo pino do sensor
const int trigPin = 5;
const int echoPin = 6;

// Variáveis para duração e distância
long duration;
int distance;

Servo myServo; // para controle do servo motor

void setup() {
  pinMode(trigPin, OUTPUT); // porta de saída do trigPin
  pinMode(echoPin, INPUT); // porta de entrada do echoPin
  Serial.begin(9600);
  myServo.attach(3); // comando do servo motor, pino que ele está conectado
}

void loop() {
  // rotação do servo entre 0 e 180 graus
  for(int i=0;i<=180;i++){
    myServo.write(i);
    delay(30);
    distance = calculateDistance();// uma função para cálculo da distância para cada grau
    Serial.print(i); // envia o grau atual para a porta serial
    Serial.print(","); // envio de valores para o processing
    Serial.print(distance); //envia o valor da distância para a porta serial
    Serial.print("."); // envio de valores para o processing
  }

  // Repete as linhas anteriores de 180 a 0 graus
  for(int i=180;i>0;i--){
    myServo.write(i);
    delay(30);
    distance = calculateDistance();
```



```
Serial.print(i);
Serial.print(",");
Serial.print(distance);
Serial.print(".");
}
}
// Função para calcular a distância medida pelo sensor ultrassônico
int calculateDistance(){
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    // Define o trigPin no estado HIGH por 10 micro segundos
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH); //Lê o echoPin, retorna o tempo de viagem da
onda sonora em microssegundos
    distance= duration*0.034/2;
    return distance;
}
```

ANEXO B- CÓDIGO *PROCESSING*

```
/*Radar Arduino*/
import processing.serial.*; /* biblioteca para comunicação serial */
import java.awt.event.KeyEvent; /*biblioteca para leitura de dados na porta serial. */
import java.io.IOException;
Serial myPort; /* define objeto serial */
/* define algumas variáveis. */
String angle="";
String distance="";
String data="";
String noObject;
float pixsDistance;
int iAngle, iDistance;
int index1=0;
int index2=0;
//fonte de texto
PFont myFont;
/**
 * setup -
 */
void setup() {
    size (1024, 700); /* ***Alterar para resolução que desejar*** */
    smooth();
    myPort = new Serial(this,"COM3", 9600); /* inicia comunicação serial. */
    myPort.bufferUntil('.'); /* lê os dados através da porta serial (angulo e distância) */
    //orcFont = loadFont("OCRAExtended-30.vlw");
    myFont = createFont("verdana", 10); /* define tipo de fonte e tamanho. */
}
/**
 * draw -
 */
void draw() {
```

```

fill(98, 245, 31);
//textFont(orcFont);
textFont(myFont);
/* simulando o borrão de movimento e lento desaparecer da linha móvel. */
noStroke();
fill(0, 4);
rect(0, 0, width, height-height*0.065);
fill(98, 245, 31); /* cor verde */
/* chama as funções para desenhar o radar. */
drawRadar();
drawLine();
drawObject();
drawText();
}
/**
 * serialEvent --
 */
void serialEvent (Serial myPort) { /* começa a ler os dados a parti da porta serial. */
/* lê os dados na porta serial (caractere) e armazena em uma string de dados. */
data = myPort.readStringUntil('.');
data = data.substring(0, data.length()-1);
index1 = data.indexOf(","); /* encontra o caractere e armazena na variável "index1" */
angle= data.substring(0, index1); /* ler os dados da posição "0" a posição do index1 variável
ou isso é o valor do ângulo da placa Arduino enviado */
distance= data.substring(index1+1, data.length()); /* ler os dados de posição "index1" para o
final do pr de dados que é o valor da distância. */
// converte string para inteiro.
iAngle = int(angle);
iDistance = int(distance);
}
/**
 * drawRadar -
 */

```

```

void drawRadar() {
  pushMatrix();
  translate(width/2, height-height*0.074); /* move as coordenadas para o novo local. */
  noFill();
  strokeWeight(2);
  stroke(98, 245, 31);
  // draws the arc lines
  arc(0, 0, (width-width*0.0625), (width-width*0.0625), PI, TWO_PI);
  arc(0, 0, (width-width*0.27), (width-width*0.27), PI, TWO_PI);
  arc(0, 0, (width-width*0.479), (width-width*0.479), PI, TWO_PI);
  arc(0, 0, (width-width*0.687), (width-width*0.687), PI, TWO_PI);
  // draws the angle lines
  line(-width/2, 0, width/2, 0);
  line(0, 0, (-width/2)*cos(radians(30)), (-width/2)*sin(radians(30)));
  line(0, 0, (-width/2)*cos(radians(60)), (-width/2)*sin(radians(60)));
  line(0, 0, (-width/2)*cos(radians(90)), (-width/2)*sin(radians(90)));
  line(0, 0, (-width/2)*cos(radians(120)), (-width/2)*sin(radians(120)));
  line(0, 0, (-width/2)*cos(radians(150)), (-width/2)*sin(radians(150)));
  line((-width/2)*cos(radians(30)), 0, width/2, 0);
  popMatrix();
}
/**
 * drawObject -
 */
void drawObject() {
  pushMatrix();
  translate(width/2, height-height*0.074); /* move as coordenadas para o novo local. */
  strokeWeight(9);
  stroke(255, 10, 10); /* cor vermelho */

  pixsDistance = iDistance*((height-height*0.1666)*0.025); /* converte a distância entre o
sensor de cm para px */

  /* limite de faixa de 40 cms */

```

```

if (iDistance<40 & iDistance > 3) {
    /* desenha o objeto de acordo com o ângulo e a distância. */
    line(pixsDistance*cos(radians(iAngle)),    -pixsDistance*sin(radians(iAngle)),    (width-
width*0.505)*cos(radians(iAngle)), -(width-width*0.505)*sin(radians(iAngle)));
}
popMatrix();
}
/**
* drawLine -
*/
void drawLine() {
    pushMatrix();
    strokeWeight(9);
    stroke(30, 250, 60);
    translate(width/2, height-height*0.074); /* move as coordenadas iniciais para o novo local. */
    line(0,    0,    (height-height*0.12)*cos(radians(iAngle)),    -(height-
height*0.12)*sin(radians(iAngle))); /* chama a linha de acordo com o angulo */
    popMatrix();
}
/**
* drawText -
*/
void drawText() { /* desenha os textos na tela. */
    pushMatrix();
    if (iDistance > 40) {
        noObject = "Fora de alcance!";
    } else {
        noObject = "Na faixa!";
    }
    fill(0, 0, 0);
    noStroke();
    rect(0, height-height*0.0648, width, height);
    fill(98, 245, 31);

```

```

textSize(12);
text("10cm", width-width*0.3854, height-height*0.0833);
text("20cm", width-width*0.281, height-height*0.0833);
text("30cm", width-width*0.177, height-height*0.0833);
text("40cm", width-width*0.0729, height-height*0.0833);
textSize(12);
text("Objeto: " + noObject, width-width*0.875, height-height*0.0277);
text("Ângulo: " + iAngle + " °", width-width*0.48, height-height*0.0277);
text("Distância: ", width-width*0.26, height-height*0.0277);
if (iDistance < 40) {
  text("      " + iDistance + " cm", width-width*0.225, height-height*0.0277);
}
/*plano cartesiano*/
textSize(20);
fill(98, 245, 60);
translate((width-width*0.4994)+width/2*cos(radians(30)),          (height-height*0.0907)-
width/2*sin(radians(30)));
rotate(-radians(-60));
text("30°", 0, 0);
resetMatrix();
translate((width-width*0.503)+width/2*cos(radians(60)),          (height-height*0.0888)-
width/2*sin(radians(60)));
rotate(-radians(-30));
text("60°", 0, 0);
resetMatrix();
translate((width-width*0.507)+width/2*cos(radians(90)),          (height-height*0.0833)-
width/2*sin(radians(90)));
rotate(radians(0));
text("90°", 0, 0);
resetMatrix();
translate(width-width*0.513+width/2*cos(radians(120)),          (height-height*0.07129)-
width/2*sin(radians(120)));
rotate(radians(-30));

```

```
text("120°", 0, 0);
resetMatrix();
translate((width-width*0.5104)+width/2*cos(radians(150)), (height-height*0.0574)-
width/2*sin(radians(150)));
rotate(radians(-60));
text("150°", 0, 0);
popMatrix();
}
/**
 * stop -
 */
void stop() {
  super.stop();
}
```