

SISTEMA DE MEDIÇÃO DE DISTÂNCIA ULTRASSÔNICO COM ARDUÍNO

Caio Vinicius Cardoso¹ – Faculdade de Tecnologia de Carapicuíba

Francisco de Assis² – Faculdade de Tecnologia de Carapicuíba

Prof. Esp. Renato de Pierri³ – Faculdade de Tecnologia de Carapicuíba

Profa. Me. Magali Amorim Mata⁴ – Faculdade de Tecnologia de Carapicuíba

Profa. Dra. Silvia M. Farani Costa⁵ – Faculdade de Tecnologia de Carapicuíba

RESUMO

Em decorrência da expressiva evolução da eletrônica, pode-se encontrar hoje no mercado diversos modelos de microprocessadores e microcontroladores com um alto nível de processamento e um baixo custo. A maior acessibilidade de aquisição desses componentes contribuiu para a criação de novos produtos. Um exemplo de novo produto desenvolvido a partir de tais componentes são os equipamentos de medição de distância, como é o caso da trena eletrônica. Este artigo tem como objetivo geral relatar o desenvolvimento de um sistema de medição de distância ultrassônico utilizando uma placa de Arduino. Trata-se de uma pesquisa de caráter exploratório e experimental de natureza aplicada com abordagem dedutiva e natureza de dados qualitativa. A partir das bibliotecas *LiquidCrystal* e *Ultrasonic.h*, juntamente com a plataforma Arduino uno, sensor ultrassônico HC-SR04, pôde-se chegar num protótipo ultrassônico para medição de distâncias. Foi possível a constatação de alguns parâmetros que podem limitar a área de aplicação desse dispositivo, o que nada implica na eficiência ou ineficiência das bibliotecas estudadas, deixando como sugestão a possibilidade de que estudos futuros de otimização de código possam ser realizados.

Palavras-chave: Arduino. Biblioteca ultrassônica. Sensor HC-SR04. Medição ultrassônica.

ABSTRACT

*As a result of the expressive evolution of electronics, several models of microprocessors and microcontrollers with a high level of processing and a low cost can be found on the market today. The increased accessibility of these components contributed to the creation of new products. An example of a new product developed from such components are distance measuring equipment, as is the case with electronic tape. This article aims to report the development of an ultrasonic distance measurement system using an Arduino board. It is an exploratory and experimental research of applied nature with deductive approach and qualitative data nature. From the libraries *LiquidCrystal* and *Ultrasonic.h*, together with the platform Arduino uno, ultrasonic sensor HC-SR04, it was possible to achieve an ultrasonic prototype for measurement of distances. It was possible to verify some parameters that can limit the area of application of this device, which not impact the efficiency or inefficiency of the studied libraries, leaving as a suggestion the possibility that future studies of code optimization might be performed.*

Keywords: Arduino. Ultrasonic libraries. Sensor HC-SR04. Ultrasonic measurement.

1 - Discente do CST em Análise e Desenvolvimento de Sistemas – e-mail: caioviniciuscardsoso@hotmail.com

2 - Discente do CST em Análise e Desenvolvimento de Sistemas – e-mail: francis.tender@yahoo.com.br

3 - Docente do CST em Análise e Desenvolvimento de Sistemas – e-mail: renato.pierri@fatec.sp.gov.br

4 - Docente do CST em Análise e Desenvolvimento de Sistemas – e-mail: magali.mata@fatec.sp.gov.br

5 - Docente do CST em Análise e Desenvolvimento de Sistemas – e-mail: silvia.costa01@fatec.sp.gov.br

1 INTRODUÇÃO

Em decorrência da expressiva evolução da eletrônica, pode-se encontrar hoje no mercado diversos modelos de microprocessadores e microcontroladores com um alto nível de processamento e um baixo custo comercial tornando-se mais acessíveis aos profissionais de diversas áreas como da tecnologia da informação, da robótica, das engenharias, ou até para aqueles que apreciem produzir *gadgets* por *hobbie*. A maior acessibilidade de aquisição desses componentes contribuiu para a criação de novos produtos, sendo estes usados nas mais diferentes áreas e com os mais diversos objetivos. Um exemplo de novo produto desenvolvido a partir de tais componentes são os equipamentos de medição de distância, como é o caso da trena eletrônica.

A automatização da trena convencional para a trena eletrônica permite que uma única pessoa consiga fazer a medição de distâncias sem a necessidade de deslocamento entre os pontos numa dada área, tornando assim a medição mais prática e fácil, além de ter uma maior precisão uma vez que tais equipamentos têm em seu conjunto de desenvolvimento, cálculos já programados, que garantem a independência da visão e habilidade do usuário.

A medição de espaços é rotina de diferentes profissionais: ora o arquiteto em seus projetos de decoração de interiores ou em projetos mais arrojados e faraônicos, ora o serralheiro ou o carpinteiro para atender sua clientela, ora o organizador de shows para a medição de palco, entre tantas outras necessidades que possam demandar a medição de distâncias num dado espaço.

Este artigo tem como objetivo geral relatar o desenvolvimento de um sistema de medição de distância ultrassônico utilizando uma placa de Arduino. Os objetivos específicos são: apresentar os componentes utilizados bem como as bibliotecas *LiquidCrystal* e *Ultrasonic*, assim como seu uso em outros projetos. Tem-se como problema de pesquisa: é possível a montagem e o desenvolvimento de uma trena eletrônica para medição de espaços? Como as bibliotecas funcionam e qual a sua relevância para o projeto?

2 REFERENCIAL TEÓRICO

2.1 Medidores Eletrônicos de Distância(MED)

Surgido em 1943, projetado pelo cientista sueco E. Bergstrand, o primeiro medidor eletrônico de distância, chegou ao mercado em 1950, conhecido pelo nome comercial de *Geodimeter NASM-2*, tendo o seu uso inicialmente voltado para medições topográficas, os

medidores eletrônicos foram se difundindo para muitas áreas tanto científica, industrial e médica. Para Tavella, Burdelis e Tenyi (2007) com o surgimento dos medidores eletrônicos de distância (MED), solucionou-se um problema da Geodésia: a medição simultânea de direções angulares e distâncias em tempo real. Reduzindo de forma significativa a quantidade de cálculos efetuados no processo de levantamento das medidas e em seu tempo de execução.

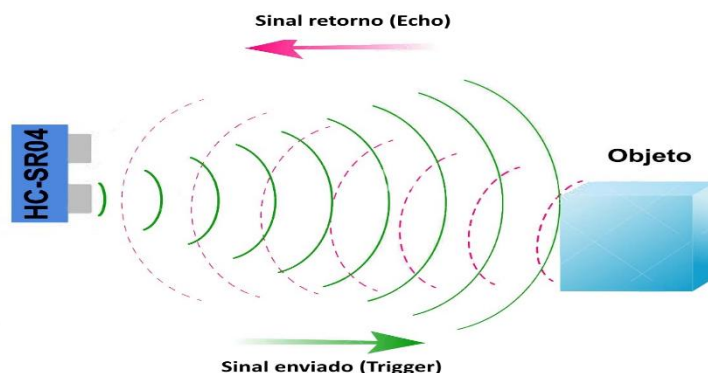
Ainda segundo o autor, embora tenham sido inventados na década de 1940, os medidores eletrônicos ainda estão em contínua evolução. De dispositivos pesados e de difícil manuseio, hoje eles estão cada vez mais leves e didáticos, fazendo parte dos chamados teodolitos eletrônicos.

O funcionamento de um dispositivo MED é parecido como um sistema de sonar, onde ondas eletromagnéticas são lançadas por um sensor em uma determinada direção e assim que encontram um obstáculo, essa onda é refletida de volta a esse mesmo sensor por onde ela é capturada e através do Arduino é feita a leitura e enviados para um *display*, onde as medidas são mostradas. É possível ressaltar ainda que esses dispositivos geralmente apresentam uma certa instabilidade em suas medidas a partir do momento em que a distância aumenta, carecendo de alguns ajustes e calibragem (TAVELLA, BURDELIS E TENYI (2007)).

2.1.1 Sensor ultrassônico de distância HC SR04

Para Nakatani, Guimarães e Neto (2013) o sensor ultrassônico de distância é um item essencial para a medição de pequenas distâncias, bastante utilizado na eletrônica, pois trata-se de um dispositivo composto por um circuito que controla a saída e entrada de ondas sonoras de alta frequência (40 khz) por meio de um transmissor e outro receptor, chegando segundo o próprio fabricante fazer medidas entre 20 a 4000 mm com margem de erro de 3mm, como mostra a Figura 1.

Figura - 1 Sensor ultrassônico de distância HC SR04 em funcionamento

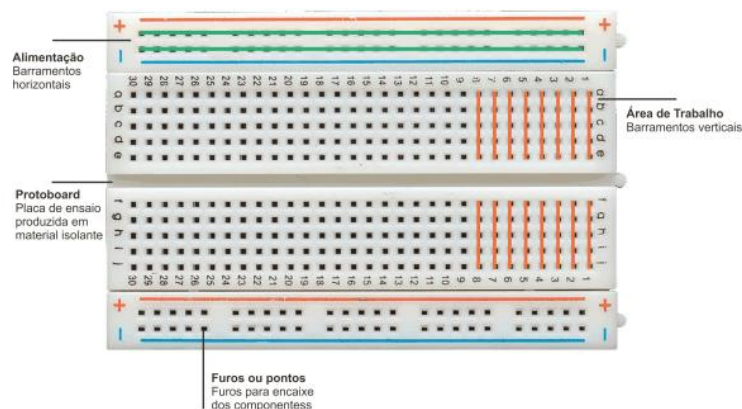


Fonte: Adilson Thomsen, 2018.

2.1.2 Protoboard 400 pontos

Uma das ferramentas mais didáticas existente no meio eletrônico, a *protoboard* é uma placa didática usada para fazer conexões de fios elétricos de baixa voltagem, para experimentos, como acendimento de LEDs e conexões de periféricos, ela permite que qualquer pessoa com um conhecimento mínimo de eletrônica possa fazer seus experimentos, criando pequeno circuitos sem a necessidade do uso da solda. Nesse projeto, foi aplicado o protoboard 400 pontos para fazer a interligação do nosso sistema conforme a Figura 2.

Figura 2 - *Protoboard* 400 pontos



Fonte: Allan Mota, 2018.

2.1.3 *Jumpers* macho-fêmea x 40 unidades

Segundo o Guia do Hardware (2018), *jumpers* são pequenos fios com conectores alternativos, utilizados para fazer a ligação de um periférico ao outro. Antigamente eles eram usados para fazerem conexões entre a placa mãe e os componentes de um computador, porém com o passar dos anos e a evolução das placas, foram surgindo novas tecnologias e atualmente seu uso é empregado na prototipação de projetos com o uso da tecnologia Arduino. A Figura 3 abaixo mostra como são os *jumpers*.

Figura 3 - *Jumpers* macho-fêmea x40 unidades

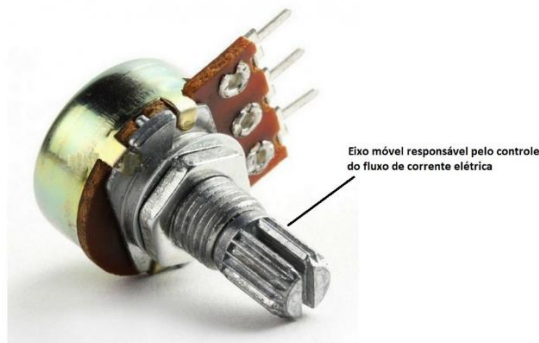


Fonte: Arduino Manaus, 2018.

2.1.4 Potenciômetro Linear Rotativo 10k

O potenciômetro é um componente eletrônico, responsável pelo controle do fluxo de eletricidade de um item para outro no dispositivo. Eles funcionam como resistores, porém, possuem um eixo móvel que permite o ajuste manual de corrente elétrica (COMO FAZER AS COISAS, 2018). A Figura 4 apresenta o potenciômetro.

Figura 4 - Potenciômetro Linear Rotativo 10k

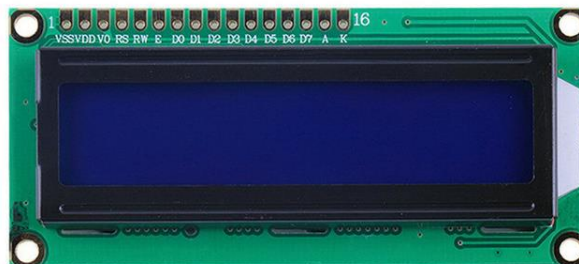


Fonte: Mercado Livre, 2018.

2.1.5 Display de LCD 16x2 Backlight Azul

De acordo com Mota (2015) todo sistema já desenvolvido até então necessita de interação com o ser humano, esse processo é chamado de interface homem máquina (IHM) e para que isso seja possível, é utilizado o *display*, onde são mostrados números, imagens, mensagens de erro pelo qual nos dá uma ideia do funcionamento ou não do nosso sistema. Para o presente projeto foi aplicado o *Display* de LCD 16x2 *Backlight* Azul como mostrado na Figura 5.

Figura 5 - Display LCD 16x2 (Tela Azul)



Fonte: Baú da Eletrônica, 2018.

3 PROCEDIMENTOS METODOLÓGICOS

Este trabalho acadêmico teve início com o levantamento dos materiais necessários e seus custos, como mostra a Tabela 1.

Tabela 1 – Tabela Material x Custo

Peça	Preço
Sensor ultrassônico HC-SR04	R\$ 6,50
Arduino (UNO R3 ou compatível) + Cabo Usb	R\$ 25,50
Potenciômetro 10K ohms (x10)	R\$ 12,00
Protoboard	R\$ 11,25
Display LCD 16x2	R\$ 15,90
Jumpers Macho x Macho (x60)	R\$ 12,50
Jumpers Macho x Fêmea (x60)	R\$ 12,50
Total	R\$ 96,50
Frete	R\$ 28,35
Caixa de Aço	R\$ 50,00
Total	R\$ 174,85

Fonte: Elaborado pelos autores.

A partir desse levantamento foi feito o teste dos componentes e por fim foi feita a montagem da trena tanto pela parte física quanto pela operacional, ambas serão detalhadas a seguir. A partir disso o trabalho foi dividido em dois seguimentos. O primeiro visando a construção da trena eletrônica com Arduino. E o segundo visando o estudo e entendimento das bibliotecas utilizadas no projeto, assim como sua aplicação em outros projetos.

As ferramentas utilizadas na construção deste trabalho foram: para o desenvolvimento da linguagem C foi utilizado o Notepad++ v7.5.5 (64-bit); para o código do *hardware* foi utilizado o *software*, *Integrated Development Environment* (IDE), de código aberto do Arduino chamado de Arduino 1.8.5. A pesquisa bibliográfica para a coleta de dados não se limitou a artigos científicos trouxe porém referências de fabricantes das peças de *Liquid Crystal Display* e sensor ultrassônico.

Trata-se de uma pesquisa de caráter exploratório e experimental. Trata-se ainda de uma pesquisa aplicada, isto é, a partir de conhecimentos obtidos, possa-se chegar a aplicação prática, para a solução de problemas específicos. Tem como abordagem o método dedutivo em que os dados são analisados do macro para o microambiente de forma descendente.

Quanto à natureza dos dados a pesquisa tem abordagem qualitativa (LAKATOS e MARCONI, 2009).

4 DESENVOLVIMENTO

4.1 Os Dispositivos Utilizados

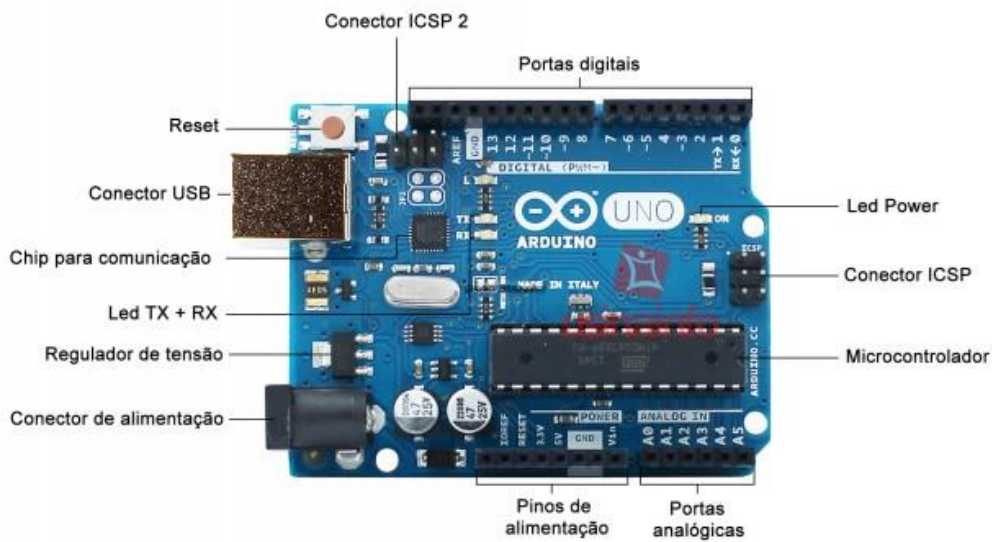
Foi utilizado como plataforma o *kit* Arduino. Surgido na Itália no ano 2005, criado pelo professor italiano Massimo Banzi e seu aluno David Mellis, O Arduino é uma plataforma eletrônica de baixo custo, utilizado no controle e desenvolvimento de sistemas das mais diversas áreas, uma das maiores vantagens do Arduino, é que ele é uma plataforma didática e de código aberto.

4.1.1 A placa Arduino uno R3

O Arduino uno é uma plataforma de prototipagem eletrônica de código aberto, destinado a qualquer pessoa que tenha interesse em criar sistemas controlados, sem a necessidade de um conhecimento técnico avançado em eletrônica ou programação. Por ser uma plataforma simples e possuir um software de fácil acesso, o Arduino uno pode ser empregado desde de sistemas simples quanto em projetos mais complexos, graças a sua flexibilidade (PRIETCH e MACEDO, 2018).

Existem diversos modelos de placas de Arduino, cada uma com a especificidade que se adequa melhor ao seu sistema ou projeto, neste trabalho foi utilizado a *Placa Arduino Uno R3*. Como pode ser observado na Figura 6, é possível destacar à sua direita o microcontrolador ATmega328, da família (AVR) da Atmel que é como se fosse um microcomputador como um único circuito integrado inserido na placa, ele é responsável por toda a parte de controles de entrada e saída de dados, controle de LEDs e automação do Arduino (GAIER, 2011).

Figura 6 - Placa Arduino Uno R3



Fonte: Equibancada, 2018.

Na figura 7 pode-se ver em detalhes o *microcontrolador ATmega328*

Figura 7 - *Microcontrolador ATmega328*



Fonte: Robson Machado, 2012

4.2 Teste dos componentes

O objetivo dessa primeira fase foi testar separadamente o sensor ultrassônico e a tela *LCD* para verificar se estavam funcionando corretamente. Para isso foram utilizados os exemplos existentes dentro da IDE do Arduino, para o teste do sensor ultrassônico entramos em Arquivo > Exemplos > HCS04 > simples, abrimos o código, testamos o sensor e visualizamos os resultados através da porta serial. Para testar o *LCD* seguiu-se o mesmo procedimento abrindo o código referente a ele pelo caminho Arquivo > Exemplos > LiquidCrystal > Display e visualizamos o resultado pela tela do *LCD*.

Uma vez em que os componentes foram testados individualmente e foi verificado que estão funcionando corretamente, partiu-se para a etapa seguinte que foi a construção da trena eletrônica.

4.3 Montagem da trena

Componentes utilizados para a montagem da trena foram: Placa Arduino Uno R3; Protoboard 400 Pontos; Sensor Ultrassônico de Distância HC-SR04; Display LCD 16x2 (Tela Azul); Potenciômetro Linear Rotativo 10k; 17 jumpers Macho x Fêmea; 2 jumpers Macho x Macho; 1 jumper Fêmea x Fêmea.

Para se fazer a trena utilizou-se as mesmas ligações dos testes, mudando apenas as ligações dos pinos *Trig* e *Echo* do sensor que estavam ligados aos pinos 11 e 12 do Arduino, que agora estão sendo usados pelo LCD:

- Pino Vcc do sensor ficará ligado ao pino 5V do Arduino;
- Pino Trig do sensor ficará ligado ao pino 8 do Arduino;
- Pino Echo do sensor ficará ligado ao pino 9 do Arduino;
- Pino Gnd do sensor ficará ligado ao pino Gnd do Arduino;
- Pino Vss do lcd ficará ligado ao pino “-” da protoboard;
- Pino Vdd do lcd ficará ligado ao pino “+” da protoboard;
- Pino Vo do lcd ficará ligado ao pino do meio do potenciômetro;
- Pino Rs do lcd ficará ligado ao pino 12 do Arduino;
- Pino Rw do lcd ficará ligado ao pino “-” da protoboard;
- Pino En do lcd ficará ligado ao pino 11 do Arduino;
- Pino D4 do lcd ficará ligado ao pino 5 do Arduino;
- Pino D5 do lcd ficará ligado ao pino 4 do Arduino;
- Pino D6 do lcd ficará ligado ao pino 3 do Arduino;
- Pino D7 do lcd ficará ligado ao pino 2 do Arduino;
- Pino A do lcd ficará ligado ao pino “+” da protoboard;
- Pino K do lcd ficará ligado ao pino “-” da protoboard;
- Primeiro pino do potenciômetro ficará ligado ao pino “-” da protoboard;
- Terceiro pino do potenciômetro ficará ligado ao pino “+” da protoboard;
- Pino GND do Arduino ficará ligado ao pino “-” da protoboard;
- Pino 5V do Arduino ficará ligado ao pino “+” da protoboard;

Uma vez que o *hardware* esteja montado e pronto, partimos para o entendimento do código (representado pelas Figuras 8 e 9) que será o responsável por fazer todos os cálculos e exibir as informações para o usuário.

O código faz a chamada da biblioteca Ultrasonic.h inicializando com os valores dos pinos Trig e Echo (8 e 9). A biblioteca faz os cálculos da distância e retorna os valores obtidos salvando-os nas variáveis cmMsec e inMsec, correspondentes aos valores centímetros e polegadas. Uma vez que o programa tenha obtido esses valores ele utiliza as funções da biblioteca LiquidCrystal para acessar o lcd e mostrar os resultados na tela.

Figura 8 – Código da Trena 1/2

```
trena | Arduino 1.8.6 Hourly Build 2018/05/03 04:33
Arquivo Editar Sketch Ferramentas Ajuda

trena $
#include <Ultrasonic.h> //Carrega a biblioteca Ultrasonic
#include <LiquidCrystal.h> //Carrega a biblioteca LCD

//Define o pino do Arduino a ser utilizado com o pino Trigger do sensor
#define PINO_TRIGGER 8

//Define o pino do Arduino a ser utilizado com o pino Echo do sensor
#define PINO_ECHO 9

//Inicializa o sensor ultrasonico
Ultrasonic ultrasonic(PINO_TRIGGER, PINO_ECHO);

//Define os pinos que serão ligados ao LCD
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  Serial.begin(9600); //Inicializa a serial
  lcd.begin(16,2); //Inicializa LCD
  lcd.clear(); //Limpa o LCD
}

void loop()
{
  float cmMsec, inMsec;

  //Le os dados do sensor, com o tempo de retorno do sinal
  long microsec = ultrasonic.timing();
```

Fonte: Próprios autores

Figura 9 – Código da Trena 2/2

```
//Calcula a distancia em centimetros
cmMsec = ultrasonic.convert(microsec, Ultrasonic::CM);

//Calcula a distancia em polegadas
inMsec = ultrasonic.convert(microsec, Ultrasonic::IN);

//Apresenta os dados, em centimetros, no LCD e na Serial
lcd.setCursor(0,0);
lcd.print("Cent.: ");
lcd.print(" ");
lcd.setCursor(7,0);
lcd.print(cmMsec);

Serial.print("Cent: ");
Serial.print(cmMsec);

//Apresenta os dados, em polegadas, no LCD e na Serial
lcd.setCursor(0,1);
lcd.print("Pol. : ");
lcd.print(" ");
lcd.setCursor(7,1);
lcd.print(inMsec);

Serial.print(", Pol. : ");
Serial.println(inMsec);

delay(1000);
}
```

Fonte: Próprios autores

4.4 As bibliotecas utilizadas

4.4.1 *LiquidCrystal.h*

Esta biblioteca permite que uma placa Arduino controle as telas *LiquidCrystal* (LCDs) baseadas no chipset Hitachi HD44780 (ou compatível), que é encontrado na maioria dos LCDs baseados em texto. A biblioteca trabalha no modo de 4 ou 8 bits (isto é, usando 4 ou 8 linhas de dados além das linhas **rs**, **en**, e opcionalmente, **rw**). Para melhor entender qual a função de cada pino do LCD, o quadro 1, mais adiante, trará cada função.

Existem quatro maneiras de instanciar o display, dependendo das configurações de *hardware* utilizadas.

1-) Instanciando o display modo 8 bits, modo Read e Write:

```
1 LiquidCrystal lcd( pino_rs, pino_rw, pino_enable, d0, d1, d2, d3, d4, d5, d6, d7);
```

2-) Instanciando o display modo 8bits, modo Write:

```
1 LiquidCrystal lcd( pino_rs, pino_enable, d0, d1, d2, d3, d4, d5, d6, d7);
```

3-) Instanciando o display modo 4bits, modo Read e Write:

```
1 LiquidCrystal lcd( pino_rs, pino_rw, pino_enable, d0, d1, d2, d3);
```

4-) Instanciando o display modo 4bits, modo Write:

```
1 LiquidCrystal lcd( pino_rs, pino_enable, d0, d1, d2, d3);
```

Um fator importante sobre a biblioteca *LiquidCrystal* é que ela já vem pré-instalada com o software Arduino, sem que haja a necessidade de baixar nada a mais para poder utilizá-la. Ela está presente em todos os projetos que utilizam o lcd para exibir informações, desde jogos em que exista a necessidade de informar o placar para o jogador até em calculadoras feitas com o Arduino.

Uma função importante da biblioteca é a `LCD.setCursor()`, a partir dela é possível definir a linha e a coluna em que o primeiro caractere será exibido. Essa função entende as linhas e colunas do LCD como uma matriz se iniciando sempre em (0,0). O Quadro 1 apresenta outras funções referentes a esta biblioteca.

Quadro 1 – Funções da biblioteca *LiquidCrystal*

Função	Descrição
<code>begin(int coluna, int linha)</code>	Define as dimensões do display
<code>clear()</code>	Reinicia e limpa o display
<code>home()</code>	Posiciona o cursor no canto superior esquerdo do display
<code>setCursor(int coluna, int linha)</code>	Posiciona o cursor em uma posição solicitada
<code>write(byte value)</code>	Escreve um caractere na posição atual do cursor
<code>print(dados)</code>	Imprime um texto na tela, pode ser char, byte, int, long ou string
<code>cursor()</code>	Exibe um caractere sublinhado na posição atual do cursor
<code>noCursor()</code>	Esconde o cursor
<code>blink()</code>	Exibe o cursor piscando
<code>noBlink()</code>	Desabilita o piscar do cursor
<code>display()</code>	Liga o display
<code>noDisplay()</code>	Apaga o display, salvando o texto atual
<code>scrollDisplayLeft()</code>	Rola o texto um espaço para a esquerda
<code>scrollDisplayRight()</code>	Rola o texto um espaço para a direita
<code>autoscroll()</code>	Rola automaticamente o texto, deslocando o caractere anterior uma posição para esquerda ou para direita
<code>noAutoscroll()</code>	Desabilita a rolagem automática
<code>leftToRight()</code>	Define a direção do texto que está sendo exibido
<code>rightToLeft()</code>	Define a direção do texto que está sendo exibido
<code>createChar(int num, byte[] jcharData)</code>	Define um caractere de 5 x 8 personalizado

Fonte: Silvatronics, 2018.

4.4.2 Ultrasonic.h

Diferentemente da *LiquidCrystal.h* a biblioteca *Ultrasonic* não vem pré-instalada no software Arduino, sendo necessário fazer o download dela na internet e instalá-la manualmente.

Everton e Romualdo explica que a biblioteca ultrassônica foi desenvolvida pelo espanhol J. Rodrigo no ano de 2015.

Ela serve para fazer o cálculo do tempo entre a emissão e a recepção de uma onda, lançada pelos sensores ultrassônicos. (CARVALHO, Bruno F, 2016).

E para fazer isso é usado a seguinte fórmula:

$$d = \frac{(V*t)}{2} \quad (1)$$

Onde:

d = a distância percorrida pela a onda até o obstáculo;

V = velocidade do som pelo ar (343 m/s aproximadamente);

t = tempo em que a onda leva para ir até o obstáculo e voltar;

E tudo isso é dividido por dois (tempo de ida e tempo de volta);

O cálculo da distância em centímetros na biblioteca funciona dessa forma:

```
1 | long distancia = duration / 29 / 2 ;
```

Onde a variável “duration”, representa o tempo em microssegundos que o pulso leva para ir até um obstáculo e voltar até o sensor.

```
1 | long duration = pulseIn(echoPin,HIGH);
```

A velocidade média do som no ar é de aproximadamente 340m/s (metros por segundo), portanto em 1.000.000 microssegundos o som percorre 340m.

Então para sabermos a distância, devemos fazer a seguinte conta:

$distancia = duration * 1.000.000 \text{ ms} / 340 \text{ m/s} / 2$ (Onde a variável “duration” representa o tempo de ida e volta, por isso se faz necessário a divisão por 2 no final).

$1.000.000 \text{ ms} / 340 \text{ m/s} = 2,94$ que equivale a distância em metros, como a trena faz a medição em centímetros, o resultado 2,92 é multiplicado por 100 o que dá 29.

Após isso a biblioteca também faz o cálculo da distância em polegadas e retorna os valores correspondentes para a trena que ficará responsável em chamar as funções da biblioteca *LiquidCrystal* e exibir as informações.

5 RESULTADOS E DISCUSSÃO

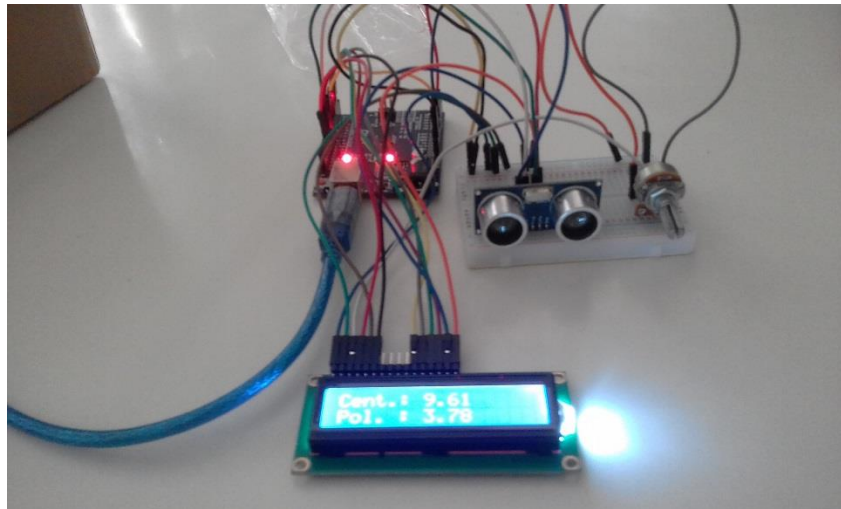
Esta pesquisa além do desenvolvimento e montagem de um sistema medidor de distância ultrassônico com Arduino versou sobre o estudo das bibliotecas pelo meio prático, mostrando que é possível a construção de uma trena eletrônica com Arduino e apresentando suas principais vantagens em relação a uma trena comum.

No decorrer da pesquisa por meio das atividades realizadas e do estudo das tecnologias e ferramentas utilizadas observou-se que o medidor ultrassônico trazia uma certa instabilidade e pouca precisão aos resultados. Como o foco com o final do projeto foi conseguir uma boa medição em centímetros pode-se contornar esse problema eliminando as casas decimais.

O fator custo benefício também deve ser citado uma vez que, com o Arduino conseguiu-se desenvolver todo o projeto tendo um custo final baixo, quando levado em consideração o custo do *kit* Arduino.

No entanto este projeto está em constante aperfeiçoamento e está longe de alcançar a precisão desejada e a variedade de funcionalidades que as trenas eletrônicas de hoje têm no mercado. A Figura 11 apresenta o versão final do desenvolvimento.

Figura 11 – Protótipo da Trena



Fonte: Próprio autor

6 CONSIDERAÇÕES FINAIS

No decorrer do período de monitoria na Faculdade de Tecnologia de Carapicuíba, nos foi proposto o desenvolvimento de um projeto com o Arduino. Por meio de pesquisas em monografias e artigos científicos na biblioteca da própria instituição, constatou-se que nenhum outro grupo havia utilizado o Arduino para realizar a medição de distâncias.

O que começou como um projeto de estudo acabou ganhando uma maior relevância ao perceber que a biblioteca Ultrasonic.h, da qual a trena faz uso das suas funções para fazer o cálculo da distância, tinha poucas informações na *internet* e por conta disso, merecia mais estudo. Tendo isto em mente, este trabalho não se limitou somente na criação de um protótipo, mas teve também o objetivo de servir como material de apoio para trabalhos futuros sobre o tema. A pesquisa teve a finalidade de trazer informações relevantes as bibliotecas estudadas tendo como objetivo servir de fonte para pesquisas futuras. Desta forma, o objetivo geral foi alcançado e os problemas de pesquisa respondidos.

REFERÊNCIAS

ASSIS, Mario. **Explorando sensores de ultrassom**. 2013. disponível em: < <http://automatobr.blogspot.com/search?q=ultrassom>>. Acessado em 2 jun. 2018.

CARVALHO, Bruno F. et al. **Evaluation of an Arduino-based IoT Person Counter**. 2016. Disponível em < <http://blog.baudaeletronica.com.br/conhecendo-biblioteca-liquidcrystal/>>. Acesso em 28 mai. 2018

GAIER, Micael Bronzatti. **Aprendendo a programar em Arduíno**. Instituto Federal de Educação, Ciência e Tecnologia do Mato Grosso (IFMT), Cuiabá, 2011. Disponível em <http://www.arduinoetecnologia.com.br/upload/apostilas/aprendendo-a-programar-em-arduino-2011.pdf> Acesso em 02 JUN 2018.

LAKATOS, E. M.; MARCONI, M. A. **Fundamentos da Metodologia Científica**. 6 ed. São Paulo: Atlas, 2009

MOTA, Allan Deangelle. 7. **Display de Cristal Líquido (LCD)**. In: Apostila Arduino Básico: Vol. 2. Serra – ES: Vida de Silício, 2015. 65p. Apostila. Acessível em: <<http://promo.vidadesilicio.com.br/apostila-arduino-basico-vol2/>>. Pesquisado em: 01/06/2018

NAKATANI, Alessandro Massayuki; GUIMARÃES, Anderson Valenga; NETO, Vicente Machado. **Medição com Sensor Ultrassônico Hc-Sr04**. 2013. Disponível em http://www.energiapura.net.br/Trabalhos%20Publicados/2014/sensor_ultrassom_arduino_cim_mec_2014.pdf Acesso em 03 JUN 2018.

PIERRI, Renato de; LiquidCrystal.h. 2017. Disponível em <<https://blog.silvatronics.com.br/utilizando-o-display-lcd-16x2-com-arduino-e-sensor-ultrassonico/>>. Acesso em 30 mai. 2018

PRIETCH, Soraia Silva. MACEDO Renato Soares de. **Utilizando a Placa Arduino como Ferramenta de Apoio ao Ensino de Programação**. Anais da IV Escola Regional de Informática da Sociedade Brasileira de Computação (SBC) – Regional de Mato Grosso, 2013. Fundação Universidade Federal de Mato Grosso do Sul, Faculdade de Computação. Disponível em https://sites.ufmt.br/eri2013/arquivos/ERI_2013_Completos.pdf . Acesso em 01 JUN 2018

TAVELLA, Anna Catarina; BURDELIS, Marcelo Jorge Parente; TENYI, Thomas Takats. **Trena eletrônica autocalibrável baseada em módulo ultrasom de baixo custo**. Revista de Engenharia de Computação e Sistemas Digitais, n. 3, p. 79-81, 2007. Disponível em <http://revista.pcs.usp.br/n3/r003c002.pdf> Acesso em 29 MAIO 2018.

“O conteúdo expresso no trabalho é de inteira responsabilidade do(s) autor(es).”