

**CENTRO PAULA SOUZA
FACULDADE DE TECNOLOGIA
FATEC SANTO ANDRÉ**

TECNOLOGIA EM ELETRÔNICA AUTOMOTIVA

**JOEL CRISTIAN DE SOUZA PINTO
LUIS GUSTAVO PEREIRA CARNEIRO**

**SISTEMA DE DETECÇÃO DE CAÇAMBA LEVANTADA EM
CAMINHÃO BASCULANTE**

**SANTO ANRÉ – SÃO PAULO
2018**

**CENTRO PAULA SOUZA
FACULDADE DE TECNOLOGIA
FATEC SANTO ANDRÉ**

TECNOLOGIA EM ELETRÔNICA AUTOMOTIVA

**JOEL CRISTIAN DE SOUZA PINTO
LUIS GUSTAVO PEREIRA CARNEIRO**

**SISTEMA DE DETECÇÃO DE CAÇAMBA LEVANTADA EM
CAMINHÃO BASCULANTE**

Monografia apresentada ao Curso de Tecnologia em Eletrônica Automotiva da Fatec Santo André, como requisito necessário para conclusão do curso em Tecnologia em Eletrônica Automotiva.

Orientador: Prof. Wagner Massarope
Co-Orientador: Prof. Me.Eliel Marcelino

**SANTO ANDRÉ – SÃO PAULO
2018**

LISTA DE PRESENÇA

Santo André, 02 de Julho de 2018

**LISTA DE PRESENÇA REFERENTE À APRESENTAÇÃO DO
TRABALHO DE CONCLUSÃO DE CURSO COM O TEMA: “SISTEMA
DE DETECÇÃO DE CAÇAMBA LEVANTADA EM CAMINHÃO
BASCULANTE” DOS ALUNOS DO 6º SEMESTRE DESTA U.E.**

BANCA

PRESIDENTE:
PROF. WAGNER MASSAROPE _____

MEMBROS:
PROF. ELIEL WELLINGTON MARCELINO _____

PROF. FERNANDO GARUP DALBO _____

ALUNOS:
JOEL CRISTIAN DE SOUZA PINTO _____

LUIS GUSTAVO PEREIRA CARNEIRO _____

Ficha catalográfica

P659s

Pinto, Joel Cristian de Souza
Sistema de detecção de caçamba levantada em caminhões
basculante / Joel Cristian de Souza Pinto, Luis Gustavo Pereira
Cameiro. - Santo André, 2018. – 74f: il.

Trabalho de Conclusão de Curso – FATEC Santo André.
Curso de Tecnologia em Eletrônica Automotiva, 2018.

Orientador: Prof. Wagner Massarope

1. Eletrônica automotiva. 2. Sensor ultrassônico. 3. Veículos. 4.
Caminhões. 5. Sistemas embarcados. 6. Sistema de
segurança. I. Cameiro, Luis Gustavo Pereira II. Sistema de
detecção de caçamba levantada em caminhões basculante.

621.389

Dedicamos este trabalho aos nossos familiares,
amigos e a todos que estiveram ao nosso lado
durante o curso.

AGRADECIMENTOS

Agradecemos a todos que de forma direta ou indiretamente contribuíram para a realização deste trabalho, ao nosso orientador Prof. Wagner Massarope pelo direcionamento e incentivo para a conclusão do curso, ao nosso co-orientador Prof. Me. Eliel Wellington Marcelino pelo apoio e direcionamento para a idealização do protótipo, ao professor Fernando Garup Dalbo pelo apoio e sugestões de melhorias e a todo o corpo docente da Fatec Santo André.

*“Eu prefiro agir certo e não receber
agradecimentos, do que agir errado e não
receber nenhuma punição.”*

Marcus Catu

RESUMO

Este trabalho tem como objetivo estudar e desenvolver um sistema de detecção de caçamba levantada em caminhão basculante, sendo este sistema de baixo custo, eficiente e confiável. Este sistema visa alertar o motorista de forma visual e sonora quando a tomada de força for acionada e também quando a caçamba estiver sendo basculada, evitando assim que possíveis acidentes ocorram.

Será utilizado para a detecção do posicionamento da caçamba um sensor ultrassônico, que será instalado na longarina do caminhão na região abaixo da caçamba, este acionará o sistema alertando ao motorista que a caçamba está sendo levantada ou se encontra levantada.

Palavras chave: sistema de segurança, caminhão basculante, sensor ultrassônico.

ABSTRACT

The objective of this work is to study and develop a system of detection of bucket raised in a dump truck, being this system of low cost, efficient and reliable. This system aim is to alert the driver visually and sound when the power take-off is activated, also when the hopper is being tilted, thus avoiding possible accidents.

The using of an ultrasonic sensor was to detect the position of the bucket that was set up in the truck's spar at the region below the bucket. The sensor reading will trigger the system by alerting the driver that the hopper is lifting up or raised.

Key words: safety system, dump truck, ultrasonic sensor.

LISTA DE FIGURAS

Figura 1- Passarela derrubada por caminhão basculante no RJ.....	16
Figura 2 - Carro de passeio atingido pela estrutura da passarela	17
Figura 3 - Taxi atingido pela estrutura da passarela.....	17
Figura 4 - Caminhão basculante Agrale-13000 4x2	21
Figura 5 - Acionamento direto	22
Figura 6 - Acionamento indireto.....	22
Figura 7 - Componentes do Sistema indireto	23
Figura 8 - Caçamba com acionamento indireto	23
Figura 9 - Acionamento Frontal	24
Figura 10 – Tomada de força	25
Figura 11- Localização das tomadas de força	25
Figura 12 - Modo de funcionamento do sensor ultrassônico	26
Figura 13 - Formação do som	27
Figura 14 - Espectro de frequências.....	28
Figura 15 - Comprimento de onda	28
Figura 16 - Detecção de objetos.....	29
Figura 17 - Placa Arduino Uno R3.....	30
Figura 18 - Sensor HC-SR04.....	36
Figura 19 - Diagrama de tempo sensor HC-SR04.....	37
Figura 20 - Funcionamento do sensor HC-SR04.....	37
Figura 21 - Local da instalação do sensor	38
Figura 22 - Placa BlackBoard UNO R3.....	39
Figura 23 - Buzzer passivo	40
Figura 24 - LED Convencional.....	41
Figura 25 - L293D.....	42
Figura 26 - Montagem do L293D.....	42
Figura 27 - Entradas e saídas do sistema	43
Figura 28 - Conexão com a BlackBoard UNO R3	47
Figura 29 - Componentes conectados a BlackBoard Uno R3	48
Figura 30 – Dispositivo de acionamento da caçamba	48
Figura 31 - Sistema de basculamento da caçamba.....	49
Figura 32 – Local de instalação do sensor e da chave fim de curso	49

Figura 33 - Protótipo caçamba abaixada	50
Figura 34 - Protótipo tomada de força ligada.....	50
Figura 35 - Protótipo caçamba levantada	50
Figura 36 - Caminhão Basculante Magic Toys	51

LISTA DE TABELAS

Tabela 1 – Passarelas derrubadas por caminhão com caçamba levantada.....	18
Tabela 2 - Característica do Arduino Uno R3	31
Tabela 3 - Parâmetros elétricos do Sensor HC-SR04	38
Tabela 4 - Mapeamento do Programa.....	44
Tabela 5 – Resultados das distâncias detectadas pelo sensor (cm)	52

LISTA DE GRÁFICOS

Gráfico 1 - Distância da caçamba abaixada	53
Gráfico 2 – Distância da caçamba levantada pelo sistema	53
Gráfico 3 - Distância da caçamba levantada manualmente.....	54

LISTA DE ABREVIATURAS E SIGLAS

ABNT – Associação Brasileira de Normas Técnicas

CA – Corrente Alternada

CC – Corrente contínua

cm – centímetro

CNTP – Condição Normal de Temperatura e Pressão

CONTRAN – Conselho Nacional de Trânsito

ED – Tomada de força acionada pelo motor

EG – Tomada de força acionada pela caixa de mudança

EK – Tomada de força acionada pelo volante

GND – Filtro Graduado de Densidade Neutra

Hz – Hertz

I2C – Inter Integrated Circuit

ICSP - In Circuit Serial Programmer

KB – Quilobyte

KHz – Quilohertz

Km/h – Quilômetro por hora

k Ω – Quiloohm

LED – Diodo Emissor de Luz

mA – Miliampère

MHz - MegaHertz

NBR – Norma Brasileira

SPLIT – Tomada de força no trem

USB – Universal Serial Bus

V – Volt, Tensão elétrica

Ω – Ohm

SUMÁRIO

1 INTRODUÇÃO	16
1.1 Justificativa.....	19
1.2 Objetivo	19
1.3 Conteúdo.....	20
1.4 Metodologia.....	20
2 TEMAS RELACIONADOS	21
2.1 Caminhão Basculante	21
2.2 Tomada de força	24
2.3 Sensor Ultrassônico	26
2.3.1 Ondas sonoras.....	26
2.4 Arduino Uno R3	30
2.4.1 Especificações Técnicas	31
2.4.2 Alimentação.....	31
2.4.3 Memória	32
2.4.4 Entrada e Saída.....	32
2.4.5 Comunicação	33
2.4.6 Programação.....	34
2.4.7 Reset automático por software	34
2.4.8 Proteção contra sobrecorrente na USB.....	35
2.4.9 Características físicas	35
3 COMPONENTES UTILIZADOS NO PROJETO	36
3.1 Sensor Ultrassônico HC-SR04	36
3.1.1 Parâmetros elétricos	38
3.2 Placa BlackBoard UNO R3	39
3.3 Buzzer.....	40
3.4 LED	41
3.5 Ponte H - L293D	42
4 DESENVOLVIMENTO DO PROTÓTIPO	43
4.1 Funcionamento do programa	43
4.2 Montagem do protótipo	47
4.3 Programação	51
4.4 Análise dos resultados	51

5	CONSIDERAÇÕES FINAIS	55
5.1	Trabalhos futuros.....	55
6	REFERÊNCIAS BIBLIOGRÁFICAS.....	56
7	APÊNDICE – CÓDIGO DO PROGRAMA.....	58
8	ANEXO – RESOLUÇÃO N°563 DO CONTRAN	73

1 INTRODUÇÃO

Nos últimos anos têm sido divulgados pela imprensa, vários acidentes provocados por caminhões basculante que estavam trafegando com a caçamba levantada, na grande maioria os motoristas envolvidos neste tipo de acidente não perceberam que a caçamba estava levantada, seja por falta de atenção, cansaço ou falha mecânica. Quando este tipo de acidente ocorre os prejuízos são enormes, pois estes veículos chocam contra estruturas sobre as rodovias como passarelas, viadutos e outras estruturas, gerando prejuízos materiais e até vítimas fatais. Na figura 1 é mostrado o acidente ocorrido em 2014 na Linha Amarela no Rio de Janeiro.

Figura 1- Passarela derrubada por caminhão basculante no RJ



Fonte: Carlos Eduardo Cardoso/Agência O Dia/Estadão Conteúdo - 2014

Segundo o jornal O GLOBO houve cinco vítimas fatais neste acidente. O caminhão trafegava com a caçamba levantada quando colidiu contra a passarela, a mesma caiu sobre três veículos, sendo uma motocicleta, um veículo de passeio e um táxi.

Figura 2 - Carro de passeio atingido pela estrutura da passarela



Fonte: Guito Moreto / Agência O Globo - 2014

Figura 3 - Taxi atingido pela estrutura da passarela



Fonte: Marcos de Paula/Estadão Conteúdo - 2014

Na tabela 1 são relatados os acidentes ocorridos nos últimos 10 anos no Brasil, levado em conta apenas os acidentes em que passarelas foram derrubadas por caminhão e que foram noticiados pela imprensa.

Tabela 1 – Passarelas derrubadas por caminhão com caçamba levantada

Data	Local do Acidente	Vítimas Fatais	Fonte
14/07/2008	Rodovias Imigrantes - SP	1	g1.globo.com
03/11/2010	Rodovia Raposo Tavares - SP	2	sao-paulo.estadao.com.br
05/08/2011	Rodovia Padre Manoel da Nóbrega - SP	1	zonaderisco.blogspot.com
19/07/2012	Rodovia BR-101 - PB	0	jornaldaparaiba.com.br
02/05/2013	Rodovia BR-050 - Uberlândia - MG	1	g1.globo.com
28/01/2014	Linha Amarela - RJ	5	videos.band.uol.com.br
16/11/2015	Rodovia Anchieta - SP	0	g1.globo.com
21/06/2016	Rodovia Rio - Santos RJ	0	g1.globo.com
19/11/2017	BR376 - PR	0	g1.globo.com
24/01/2018	Avenida Brasil - RJ	1	g1.globo.com

Fonte: dos Autores

Devido à incidência desses tipos de acidentes o CONTRAN (Conselho Nacional de Trânsito) baixou a Resolução 563/2015 (anexo) tornando obrigatória a utilização de sistemas de segurança para a circulação de veículos e implementos rodoviários do tipo carroceria basculantes.

Foram aprovados três sistemas de segurança, sendo dispositivo de segurança primário, secundário e terciário. Esses sistemas são definidos na norma ABNT NBR 16141 – Implementos rodoviários – Sistemas de travamento e levantamento da caçamba basculante – Requisitos (ABNT NBR 16141, 2013, p.2):

“Dispositivo de segurança primário: dispositivo que impede o acionamento da tomada de força de forma involuntária e de modo que, para o acionamento, sejam necessários dois comandos de acionamentos ou um comando de dois estágios”.

“Dispositivo de segurança secundário: aviso visual e sonoro, com intuito de alertar o operador sobre o acionamento da tomada de força, sendo que o aviso visual é colocado na altura do painel e no campo visual do operador”.

“Dispositivo de segurança terciário: dispositivo eletrônico de controle do acionamento da tomada de força que objetiva garantir que o caminhão não passe 10 Km/h com a tomada de força ligada”.

Neste projeto estaremos atendendo o sistema secundário, além de alertar o motorista quando a tomada de força estiver acionada será desenvolvido um sistema

de detecção de caçamba levantada através da utilização do sensor ultrassônico, indicando os benefícios, desafios e viabilidade da sua implementação.

1.1 Justificativa

De acordo com os relatos da maioria dos motoristas envolvidos nos acidentes, eles não notaram que a caçamba estava erguida enquanto trafegavam. Nas imagens de vídeos publicadas pela imprensa é possível notar a caçamba sendo levantada enquanto o caminhão está trafegando em alta velocidade (acidente ocorrido na via - Linha Amarela - Rio de Janeiro, 28/01/2014, www.videos.band.uol.com.br).

A solução apresentada neste projeto possibilitará ao motorista do caminhão evitar acidentes, pois o sistema o alertará sobre o posicionamento da caçamba e em algumas situações o sistema abaixará a caçamba automaticamente.

1.2 Objetivo

O objetivo do trabalho é desenvolver um sistema que detecta o posicionamento da caçamba, a fim de evitar que o motorista trafegue com o caminhão sem notar que a caçamba está levantada ou sendo levantada. O motorista do caminhão será alertado através de sinais luminosos e sonoros quando:

- A tomada de força for acionada.
- A caçamba estiver sendo levantada.
- A caçamba estiver levantada.
- A caçamba estiver sendo abaixada.
- Mau funcionamento do sistema.

Ainda de forma didática, monitorar o sistema de forma que caso a caçamba erga sem o acionamento da tomada de força por parte do motorista, o sistema indique através de um display que há falha no sistema e em seguida abaixe a caçamba automaticamente.

Caso o motorista esqueça a caçamba levantada, ao ultrapassar a 10 Km/h o sistema abaixará a caçamba automaticamente. O sistema também alertará quando a tomada

de força for acionada caso o caminhão esteja trafegando com velocidade superior a 10 Km/h, nesta situação o sistema também bloqueará o basculamento da caçamba.

Teremos também um sistema de redundância utilizando uma chave fim de curso, ou seja, indicará se a caçamba está totalmente abaixada ou levantada.

1.3 Conteúdo

A divisão deste trabalho baseia-se em 5 capítulos. No primeiro capítulo é apresentada a justificativa para o desenvolvimento do sistema, uma breve descrição e o objetivo.

No segundo capítulo são abordados os temas relacionados para o funcionamento do projeto.

No terceiro capítulo são apresentados os componentes utilizados para o funcionamento do sistema.

O quarto capítulo é apresentado o desenvolvimento do protótipo e resultados dos testes realizados durante o desenvolvimento.

No quinto capítulo são realizadas as considerações finais e sugestões de trabalhos futuros.

1.4 Metodologia

Utilizando o sensor ultrassônico que será acoplado ao protótipo de forma que fique instalado abaixo da caçamba, e ligado a placa BlackBoard UNO R3 (Robocore), será criado um sistema que deverá alertar o motorista através de sinal visual e sonoro quando a caçamba estiver levantada ou sendo levantada. A fim de atender a resolução nº563 do CONTRAN referente ao dispositivo de segurança secundário, o sistema indicará quando a tomada de força for acionada.

Para realizar o sinal visual será utilizado LED e para o sinal sonoro o buzzer. O sistema também conterá um display para indicar o funcionamento do sistema.

2 TEMAS RELACIONADOS

2.1 Caminhão Basculante

Caminhão equipado com carroceria basculante de aço, que é acoplada no chassi do caminhão, geralmente utilizado para transporte de carga granel. Para realização da descarga a caçamba é basculada através de um dispositivo hidráulico, geralmente realizado para trás onde a carga escorre devido à ação da gravidade.

Para Schulz (2015) o sistema de basculamento das caçambas pode ocorrer de três formas: acionamento direto, acionamento indireto e acionamento frontal.

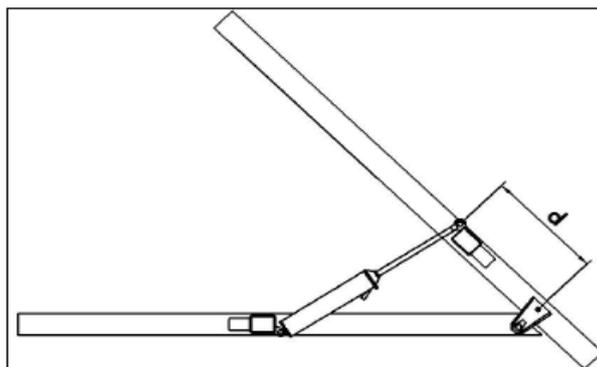
O acionamento direto é composto por um ou dois cilindros simples, ligados diretamente ao chassi do caminhão e a haste do cilindro ligada a caçamba. Este sistema é utilizado em caçambas menores, comumente utilizado por depósitos de materiais de construção para entrega de pedra e areia. Na figura 4 é mostrado um caminhão com o sistema de acionamento direto e na figura 5 o esquema do acionamento direto.

Figura 4 - Caminhão basculante Agrale-13000 4x2



Fonte: www.terraplenagem.net/caminhao-agrale-13000-548. Acessado em 05/05/2018 às 11:17.

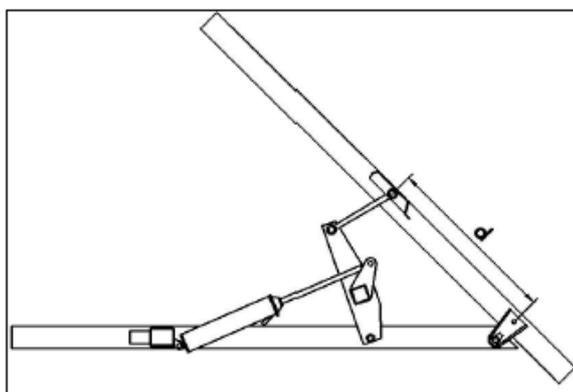
Figura 5 - Acionamento direto



Fonte: Análise e projeto de sistema de acionamento para caçamba basculante. Schulz, Freddy Johnatan,2015.

O acionamento indireto é composto por um ou dois cilindros simples conectados aos chassis inferior e superior da caçamba indiretamente, através de um mecanismo composto por uma alavanca e braços, conforme mostrado na figura 6. O acionamento indireto tem a vantagem de obter um momento maior da força de basculamento, sem que haja aumento da força exercida pelo cilindro hidráulico. Conseqüentemente caminhões com este tipo de acionamento conseguem transportar maiores cargas.

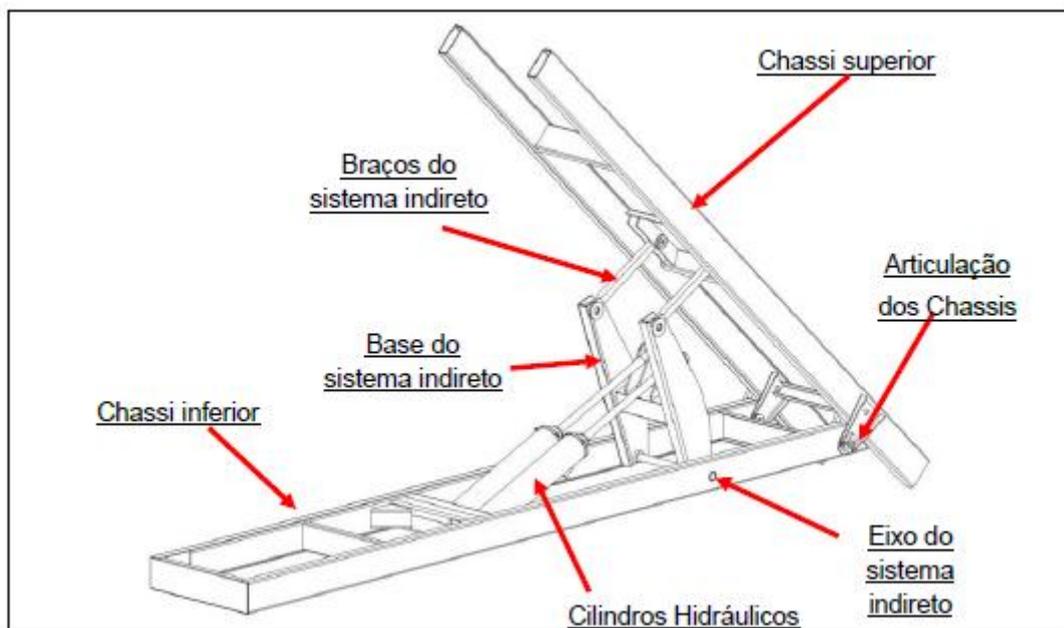
Figura 6 - Acionamento indireto



Fonte: Análise e projeto de sistema de acionamento para caçamba basculante. Schulz, Freddy Johnatan,2015.

Na figura 7 são mostrados os componentes utilizados no sistema indireto e na figura 8 a caçamba com acionamento indireto.

Figura 7 - Componentes do Sistema indireto



Fonte: Análise e projeto de sistema de acionamento para caçamba basculante. Schulz, Freddy Johnatan, 2015.

Figura 8 - Caçamba com acionamento indireto



Fonte: www.pastre.com.br. Acessado em 05/05/2018 às 11:21

No acionamento frontal é utilizado um cilindro hidráulico composto por várias hastes de diâmetros diferentes, acoplados um dentro do outro, daí o nome cilindro hidráulico telescópico. Devido ao seu tamanho inicial é possível instalar na frente da caixa de carga, conforme mostrado na figura 9.

Figura 9 - Acionamento Frontal



Fonte: www.saojorgelocadora.com.br – Acessado em 05/05/2018 às 11:25

2.2 Tomada de força

Para Manutenção & Tecnologia (2018) a tomada de força é responsável por transferir parte da potência do motor para movimentar um implemento, ela é fundamental para determinar a eficiência da operação. As tomadas de força são projetadas para acionamento direto de uma bomba hidráulica, na figura 10 é mostrada uma tomada de força.

A Scania possui diferentes tipos de tomada de força, sendo:

- Tomadas de força acionada pelo motor (ED, independente da embreagem);
- Tomadas de força acionada pelo volante (EK, independente da embreagem);

- Tomadas de força acionadas pela caixa de mudanças (EG, dependente da embreagem);
- Tomadas de força no trem de força (SPLIT, dependente da embreagem).

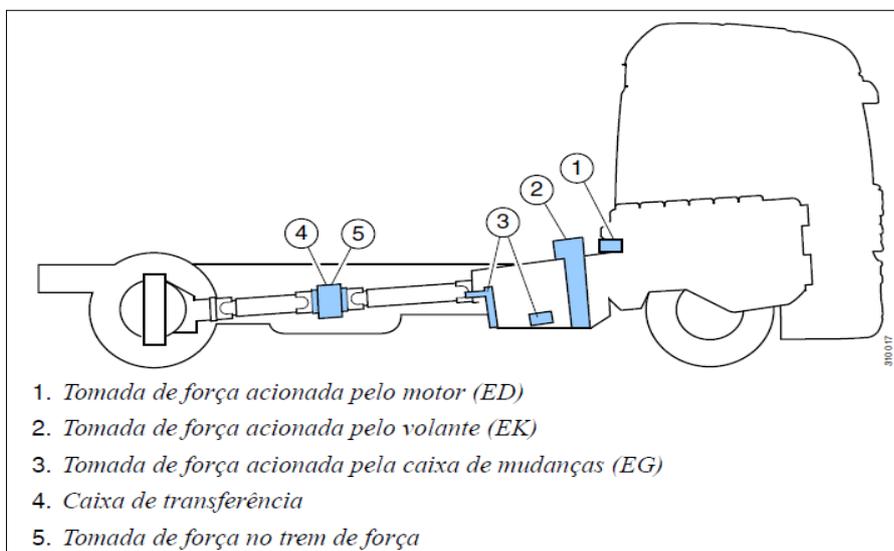
Figura 10 – Tomada de força



Fonte: www.mfrural.com.br – Acessado em 11/06/2018 às 21:30

A figura 11 mostra a localização onde pode ser instalada a tomada de força nos caminhões da Scania.

Figura 11- Localização das tomadas de força



Fonte: Scania CV AB 2016, Sweden

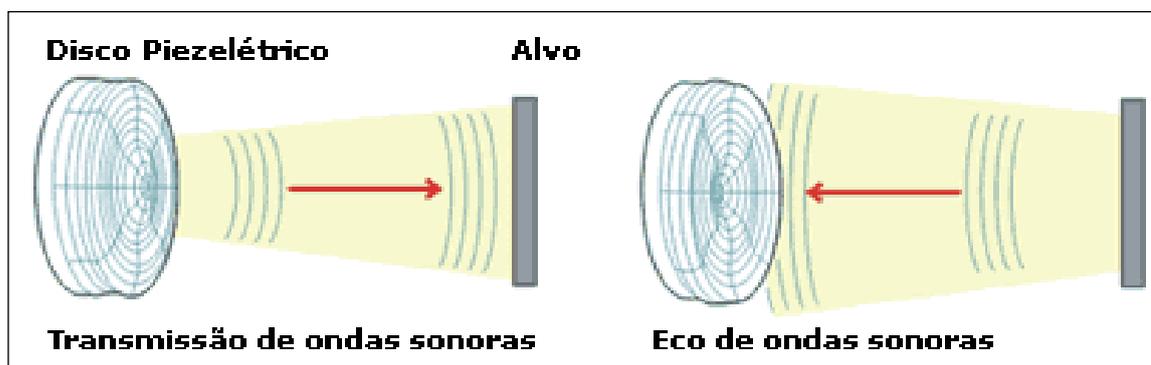
2.3 Sensor Ultrassônico

Para Mecânica Industrial (2018) sensor ultrassônico é um dispositivo que utiliza som em alta frequência para medir distâncias. Este sensor consiste de uma unidade de transceptor que é capaz de emitir e detectar o som, o pulso sonoro criado pelo dispositivo está além da faixa de audição do ouvido humano.

De acordo com Gomes (2017), o princípio de funcionamento dos sensores ultrassônicos consiste na emissão de uma onda sonora de alta frequência e na medição do tempo levado para a recepção da reflexão sonora, que é produzida quando a onda se choca com algum objeto capaz de refletir o som.

Os pulsos ultrassônicos são emitidos ciclicamente, quando a onda atinge um objeto estes pulsos são refletidos, conforme vemos na figura 12. O eco resultante é recebido e convertido em um sinal elétrico.

Figura 12 - Modo de funcionamento do sensor ultrassônico



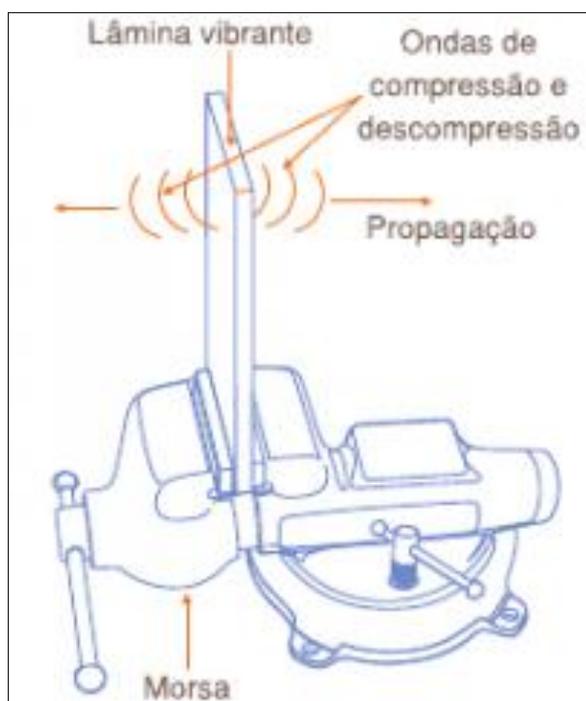
Fonte: <http://www.automatizesensores.com.br/ultrasonicos>. Acessado em 20/08/2017 às 21:15.

O sensor ultrassônico mede o tempo de propagação do eco, sendo o intervalo de tempo medido entre o impulso sonoro emitido e o eco do mesmo.

2.3.1 Ondas sonoras

Quando um objeto vibra como, por exemplo, uma lâmina de metal presa a uma morsa, conforme ilustra a figura 13, esse objeto produz ondas de compressão e descompressão do ar.

Figura 13 - Formação do som



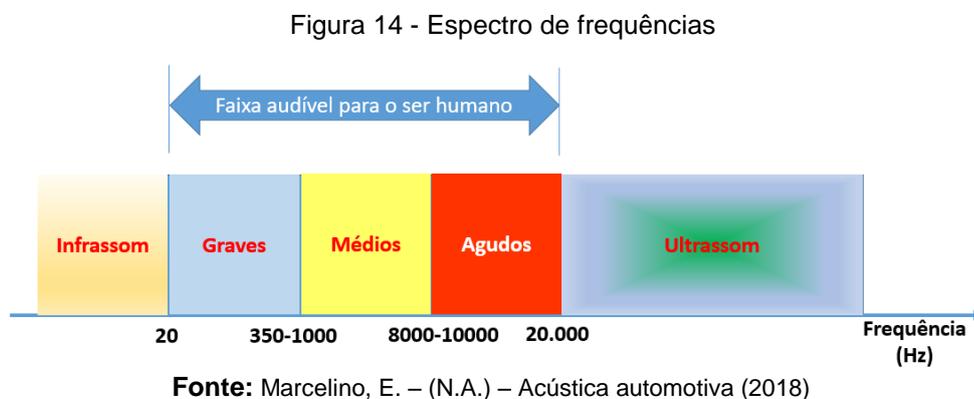
Fonte: <http://www.mecatronicaatual.com.br> – Acessado em 20/08/2017 às 21:32.

Essas ondas se propagam a uma velocidade de 343,5 metros por segundo no ar em condições normais de temperatura e pressão (CNTP). Com a elevação da temperatura, essa velocidade aumenta da ordem de 0,607 metros por segundo para cada grau Celsius. Atingindo nossos ouvidos, essas ondas alcançam o canal auditivo e em seguida vibram a membrana timpânica que irá excitar a cóclea a produzir pulsos elétricos ao nosso cérebro, produzindo finalmente a sensação sonora, para uma faixa bem definida de frequências (Mecânica Atual, 2017).

O ouvido humano está habilitado para perceber níveis de pressão sonora (NPS) na faixa de 2×10^{-5} Pa, que é considerado o limiar da audição e também conhecido como 0dB (zero dB), até aproximadamente 20Pa, que representa em torno de 130dB e considerado como o limiar da dor, na faixa de frequência de 20Hz a 20kHz (Mecânica Atual, 2017).

Teoricamente, partindo do zero dB, só podemos começar a ouvir alguma coisa quando ultrapassar o número de 20 vibrações por segundo ou 20 Hz. À medida que as vibrações vão se tornando mais rápidas, vamos tendo a sensação de sons cada vez mais agudos até que em torno de 20000 Hz (dependendo da pessoa), acima desta faixa deixamos de ter qualquer sensação auditiva. É justamente acima dessas

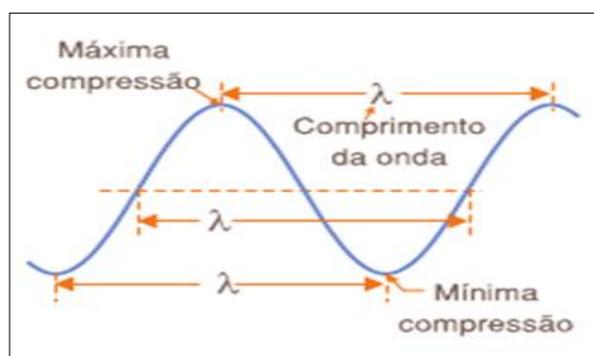
20 000 vibrações por segundo ou 20 kHz, que estão os ultrassons, conforme vemos no espectro da figura 14 (Mecânica Atual, 2017).



As vibrações sonoras na faixa do ultrassom existem, mas não podemos ouvi-las, e suas propriedades são as mesmas dos sons comuns. É claro que existem animais que podem ouvir bem acima dos 20kHz, como os morcegos, os golfinhos e até mesmo o cachorro. Alguns morcegos podem ouvir ultrassons de frequências que ultrapassam os 200 kHz. Mas, por que usar os ultrassons em sensores e não sons comuns? (Mecânica Atual, 2017).

Além de não sermos incomodados com o barulho, pois não podemos ouvir os ultrassons, há outras propriedades importantes que devem ser levadas em conta. Os sons e ultrassons, como qualquer tipo de vibração, possuem uma intensidade, frequência e comprimento de onda. O comprimento de onda, que nos interessa em especial, é a distância entre dois pontos de compressão máxima ou mínima de uma onda, observe a figura 15 (Mecânica Atual, 2017).

Figura 15 - Comprimento de onda



Fonte: <http://www.mecatronicaatual.com.br> – Acessado em 20/08/2017 às 21:32.

Se a frequência do som aumenta, a distância entre esses pontos diminui, ou seja, temos comprimentos de onda menores. O comprimento é importante, pois determina as dimensões dos objetos, os quais ele pode refletir. De fato, os sons comuns produzem o eco pela reflexão em objetos. Todavia, para refletir e retornar na direção da emissão, o objeto deve ter dimensões maiores do que o comprimento de onda emitido. Assim, quando se deseja que um sensor seja capaz de detectar pequenos objetos por reflexão, o som emitido deverá ter pequeno comprimento de onda, ou seja, deverá ter uma frequência muito alta, na faixa do ultrassom conforme se percebe pela definição de comprimento de onda da equação abaixo:

$$\lambda = \frac{c}{f}$$

Onde:

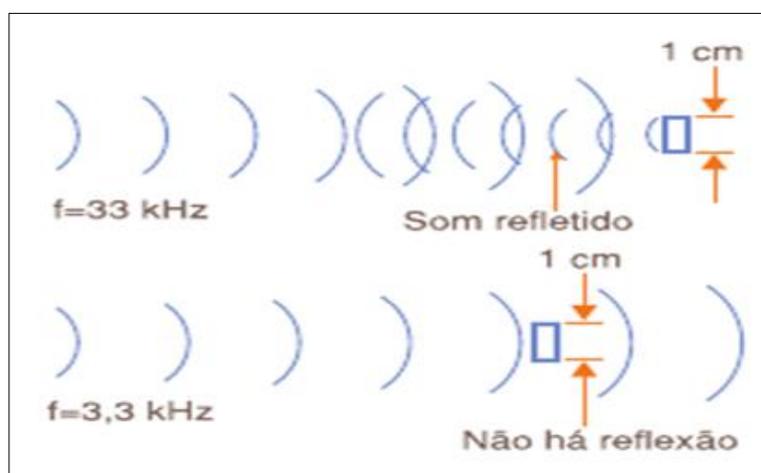
λ = Comprimento de onda (m)

c = velocidade do som (m/s)

f = frequência (Hz)

Um sinal de 33 kHz, exposto a uma temperatura ambiente de 20°C, por exemplo, tem um comprimento de onda de 1cm. Portanto, essa a ordem de grandeza do menor objeto que um sensor que use um sinal desta frequência pode detectar, conforme exhibe a figura 16 (Mecânica Atual, 2017).

Figura 16 - Detecção de objetos



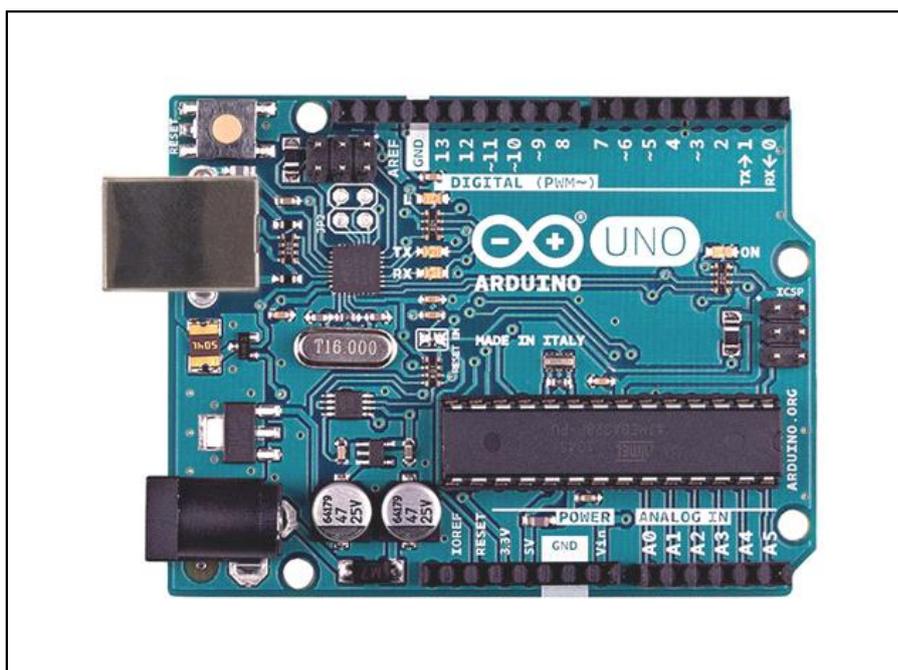
Fonte: <http://www.mecatronicaatual.com.br> – Acessado em 20/08/2017 às 21:32.

A distância em que o objeto se encontra pode ser facilmente determinada pelo tempo que o som demora em ir e voltar. Utilizando o exemplo da velocidade do som no ar à 20°C, se o tempo de retorno for de 0,01 segundos, teremos 3,3 metros de distância à fonte, e se for 0,001 segundos, teremos 33cm. Veja que os tempos envolvidos neste processo são muito maiores do que se utilizássemos o radar (Mecânica Atual, 2017).

2.4 Arduino Uno R3

O Arduino Uno é uma placa de microcontrolador baseado no ATmega328. Ele tem 14 pinos de entrada/saída digital (dos quais 6 podem ser usados como saídas PWM), 6 entradas analógicas, um cristal oscilador de 16MHz, uma conexão USB, uma entrada de alimentação uma conexão ICSP (In Circuit Serial Programmer) e um botão de reset. Ele contém todos os componentes necessários para suportar o microcontrolador, basta conectá-lo a um computador com um cabo USB ou ligá-lo com um adaptador AC-CC ou bateria para começar (ARDUINO, 2018).

Figura 17 - Placa Arduino Uno R3



Fonte: <https://multilogica-shop.com/arduino-uno-r3> - Acessado em 05/05/2018 às 11:36.

2.4.1 Especificações Técnicas

Na tabela 2 são apresentadas as características técnicas do Arduino Uno R3.

Tabela 2 - Característica do Arduino Uno R3

Microcontrolador	ATmega328
Tensão Operacional	5V
Tensão de entrada (recomendada)	7-12V
Tensão de entrada (limites)	6-20V
Pinos E/S digitais	14 (dos quais 6 podem ser saídas PWM)
Pinos de entrada analógica	6
Corrente CC por pino E/S	40 mA
Corrente CC para o pino 3,3V	50 mA
Flash Memory	32 KB (ATmega328) dos quais 0,5KB são utilizados pelo bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Velocidade de Clock	16 MHz

Fonte: dos autores

2.4.2 Alimentação

O Arduino Uno pode ser alimentado pela conexão USB ou com uma fonte de alimentação externa. A alimentação é selecionada automaticamente.

Alimentação externa (não USB) pode ser tanto de um adaptador CA para CC ou bateria. Há um conector para alimentação de 2,1mm com o positivo no centro. Cabos vindos de uma bateria podem ser inseridos diretamente nos pinos Gnd e Vin do conector de alimentação (ARDUINO, 2018).

Esta placa pode funcionar com uma fonte de alimentação externa de 6 a 20 volts. No entanto se a alimentação for inferior a 7V, o pino 5V pode fornecer menos de cinco volts e a placa pode se mostrar instável. E se a alimentação for maior do que 12V o regulador de voltagem pode superaquecer e danificar a placa. A faixa recomendada é de 7 a 12 volts.

Os pinos de alimentação são os seguintes:

- VIN - A entrada de alimentação para a placa Arduino quando se está utilizando uma fonte de alimentação externa (em oposição à conexão USB ou outra fonte de alimentação regulada). Você pode fornecer alimentação através deste pino, ou se estiver utilizando o conector de alimentação acessar esta voltagem aqui.
- 5V - A fonte de alimentação regulada usada para o microcontrolador e para outros componentes na placa. Pode vir tanto do VIN através do regulador embarcado ou da conexão USB ou outra fonte regulada em 5V.
- 3V3 - Uma fonte de 3,3V gerada pelo regulador embarcado. A corrente máxima suportada é de 50mA.
- GND - Pinos terra.

2.4.3 Memória

O ATmega328 tem 32KB (dos quais 0,5 são utilizados pelo bootloader) e inclui 2KB de SRAM e 1KB de EEPROM.

2.4.4 Entrada e Saída

Cada um dos 14 pinos digitais do Arduino Uno podem ser utilizados como uma entrada ou uma saída utilizando-se as funções `pinMode()`, `digitalWrite()`, e `digitalRead()`. Eles operam a 5V. Cada pino pode fornecer ou receber um máximo de 40mA e tem um resistor pull-up interno (desconectado por padrão) de 20-50kΩ.

Além disso alguns pinos tem funções especializadas:

- Serial - 0 (RX) e 1 (TX) Usados para receber (RX) e transmitir (TX) dados seriais TTL. Estes pinos são conectados aos pinos correspondentes do chip serial USB-para-TL ATmega8U2.
- Interruptores Externos 2 e 3 - Estes pinos podem ser configurados para disparar uma interrupção de acordo com alguma variação sensível pelo circuito.
- SPI 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK) - Estes pinos dão suporte à comunicação SPI utilizando a biblioteca SPI.

- LED 13 - Há um LED integrado ao pino digital 13. Quando este pino está no valor HIGH este LED está aceso, quando o pino está em LOW o LED está apagado.
- I2C 4 (SDA) e 5 (SCL) - Fornecem suporte a comunicação I2C (TWI) utilizando a biblioteca Wire.
- AREF - Voltagem de referência para as entradas analógicas. Utilizado com a função `analogReference()`.
- Reset - Envio o valor LOW para esta linha para resetar o microcontrolador. Tipicamente usado para adicionar um botão de reset para shields montados sobre a placa original.
-

O Arduino Uno tem 6 entradas analógicas, etiquetadas de A0 a A5, cada uma tem 10 bits de resolução (i.e. 1024 valores diferentes). Por padrão elas medem de 0 a 5V, embora seja possível alterar o limite superior utilizando o pino AREF e a função `analogReference()` (ARDUINO, 2018).

2.4.5 Comunicação

O Arduino Uno possui uma série de facilidades para se comunicar com um computador, outro Arduino, ou outros microcontroladores. O ATmega328 fornece comunicação serial UART TTL (5V) que está disponível nos pinos digitais 0 (RX) e 1 (TX). Um ATmega8U2 na placa canaliza esta comunicação para a USB e aparece como uma porta virtual para o software no computador. O firmware do '8U2 utiliza os drivers padrão USB COM e nenhum driver externo é necessário. Entretanto, no Windows, um arquivo `.inf` é necessário. O software do Arduino inclui um monitor serial que permite dados textuais ser enviados e recebidos da placa. LEDs conectados ao RX e TX piscarão enquanto dados estiverem sido transmitidos pelo chip USB-para-serial e pela conexão USB (mas não para comunicação serial nos pinos 0 e 1).

Uma biblioteca de `SoftwareSerial` permite comunicação serial em qualquer dos pinos digitais do Arduino Uno.

O ATmega328 também suporta comunicação I2C (TWI) e SPI. O software do Arduino inclui uma biblioteca `Wire` para simplificar o uso do bus I2C (ARDUINO, 2018).

2.4.6 Programação

O Arduino Uno pode ser programado com o software Arduino. O ATmega328 no Arduino Uno vem pré-gravado com um bootloader que permite que seja enviado um código novo para ele sem a utilização de um programador de hardware externo. Ele se comunica utilizando o protocolo original STK500 (ARDUINO, 2018).

O código fonte do firmware do ATmega8U2 também está disponível. Este chip é carregado com um bootloader DFU, que pode ser ativado conectando o jumper de solda na parte posterior da placa e depois resetando o 8U2. Pode-se utilizar o software FLIP da Atmel (Windows) ou o programador DFU (Mac OS X e Linux) para carregar um novo firmware. Ou ainda utilizar um programador externo (sobrescrevendo o bootloader DFU).

2.4.7 Reset automático por software

Ao invés de necessitar do pressionamento físico de um botão antes de um upload, o Arduino Uno é desenvolvido de maneira que permita que esta operação seja feita por meio do software rodando em um computador. Uma das linhas de controle de fluxo do hardware (DTR) do ATmega8U2 é conectado à linha de reset do ATmega328 através de um capacitor de 100nF. Quando esta linha é declarada (rebaixada) a linha de reset cai o suficiente para resetar o chip. O software do Arduino utiliza esta capacidade para permitir o envio de código novo simplesmente pressionando o botão de upload na IDE. Isto significa que o bootloader pode ter um intervalo mais curto, uma vez que o rebaixamento do DTR pode ser melhor coordenado com o início do upload (ARDUINO, 2018).

Esta configuração tem outras implicações. Quando o Arduino Uno é conectado a um computador rodando Mac OS X ou Linux, ele é resetado cada vez que uma conexão é estabelecida com o software (via USB). Durante os próximos 500ms, o bootloader estará rodando no Arduino Uno. Uma vez que ele está programado para ignorar

dados malformados (qualquer coisa diferente do upload de um novo código), ele irá interceptar os primeiros bytes de informação após a abertura da conexão. Se um programa rodando na placa recebe alguma configuração ou outra informação quando começa a rodar esteja seguro de que o software com o qual ela se comunica espere por um segundo antes de começar a enviar dados.

O Arduino Uno contém uma trilha que pode ser interrompida (cortada fisicamente) para desabilitar o auto-reset. Os conectores de cada lado da trilha podem ser soldados para reabilitar esta função (ARDUINO, 2018).

2.4.8 Proteção contra sobrecorrente na USB

O Arduino Uno possui um polyfuse resetável que protege a porta USB do seu computador contra sobrecorrente e curtos circuitos. Embora muitos computadores tenham sua própria proteção interna, o fuso fornece uma camada a mais de proteção. Se mais de 500mA forem aplicados a porta USB ele automaticamente irá interromper a conexão até que o curto ou a sobrecarga seja removido (ARDUINO, 2018).

2.4.9 Características físicas

A largura e o comprimento máximos do PCB do Arduino Uno são 68,58 e 53,34mm respectivamente (2,7 x 2,1 polegadas), com os conectores USB e de alimentação estendendo-se além destas dimensões. Quatro orifícios para parafusos permitem que a placa seja fixada a uma superfície ou encapsulamento. Verifique que a distância entre os pinos digitais 7 e 8 é de 160 milésimos de polegada (ARDUINO, 2018).

3 COMPONENTES UTILIZADOS NO PROJETO

3.1 Sensor Ultrassônico HC-SR04

Para Costa (2017) o sensor ultrassônico HC-SR04 permite que seja feita leituras de distâncias entre 3 cm e 4 metros, com precisão de 3 mm. Pode ser utilizado simplesmente para medir a distância entre o sensor e um objeto, como para acionar portas do microcontrolador, desviar um robô de obstáculos e acionar alarmes.

Figura 18 - Sensor HC-SR04

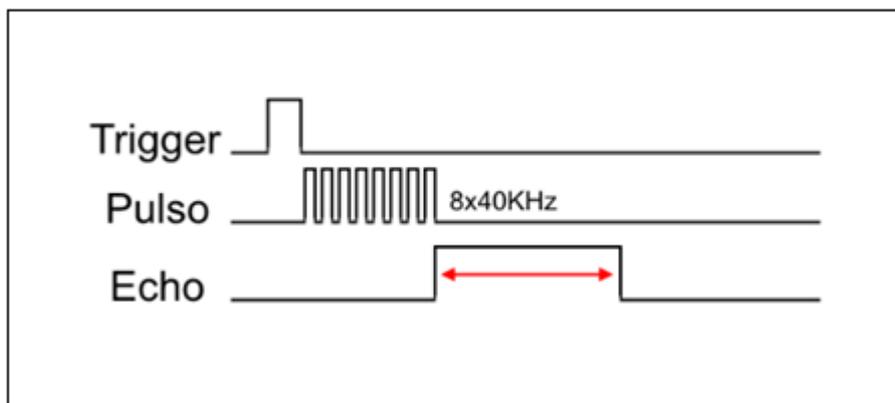


Fonte: dos autores

Seu funcionamento baseia-se no envio de sinais ultrassônicos (Trigger), o mesmo aguarda o retorno do sinal (Echo) e com base no tempo de envio e retorno do sinal é calculada a distância entre o sensor e o objeto detectado.

Primeiramente é enviado um pulso de 10 μ s, indicando o início da transmissão de dados. Em seguida são enviados 8 pulsos de 40 KHz e o sensor fica aguardando o retorno em nível alto.

Figura 19 - Diagrama de tempo sensor HC-SR04



Fonte: DataSheet HC-SR04

Para determinar a distância entre o sensor e o objeto, é utilizada a equação:

$$D = \frac{Techo * Vsom}{2}$$

Onde:

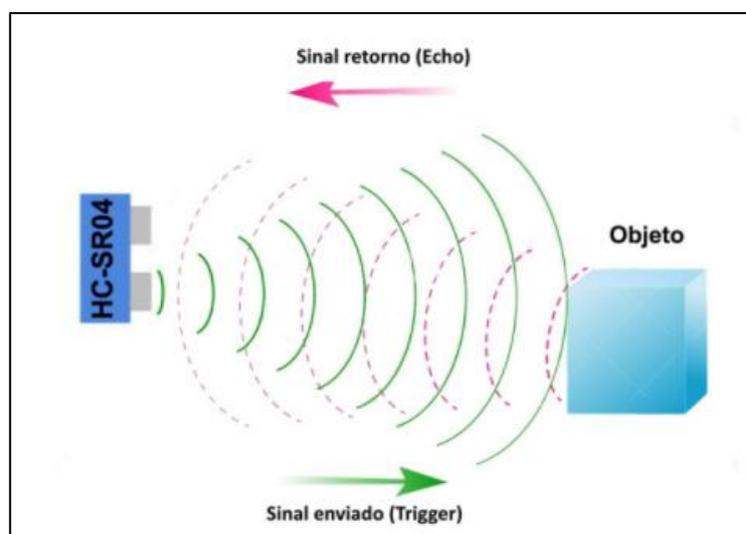
D = distância entre o sensor e o obstáculo;

Techo = tempo echo em nível alto;

Vsom = velocidade do som no ar (m/s)

A divisão por 2 é devido ao tempo de ida e volta do sinal.

Figura 20 - Funcionamento do sensor HC-SR04



Fonte: [http:// www.filipeflop.com/blog/sensor-ultrassonico-hc-sr04-ao-arduino/](http://www.filipeflop.com/blog/sensor-ultrassonico-hc-sr04-ao-arduino/)

Acessado em 05/05/2018 às 11:40.

3.1.1 Parâmetros elétricos

Na tabela 3 são mostrados os parâmetros elétricos do sensor HC-SR04.

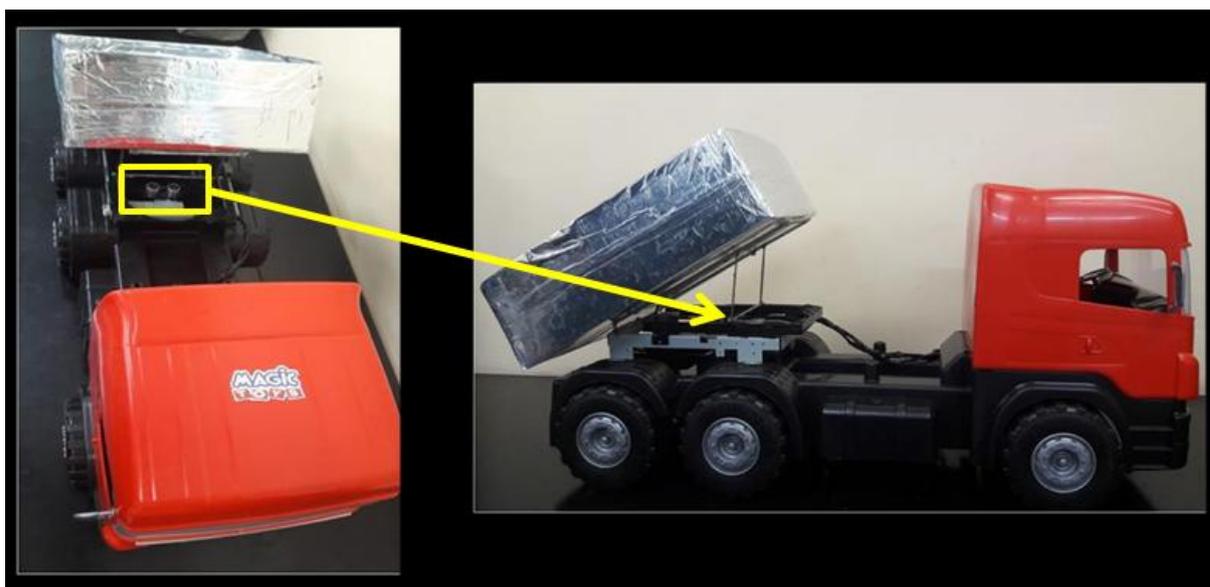
Tabela 3 - Parâmetros elétricos do Sensor HC-SR04

Tensão:	DC 5 V
Corrente:	15mA
Frequência:	40Hz
Máxima distância:	4m
Mínima distância:	2cm
Ângulo de trabalho:	15 Graus
Sinal de entrada Trigger:	Pulso TTL de 10uS.
Sinal de saída Echo:	Entrada sinal de movimento TTL e a distância em proporção.
Dimensões:	45x20x15mm

Fonte: dos autores

Neste projeto o sensor HC-SR04 tem a função de monitorar o posicionamento da caçamba do caminhão, este instalado na longarina abaixo da caçamba, conforme mostrado na figura 21.

Figura 21 - Local da instalação do sensor

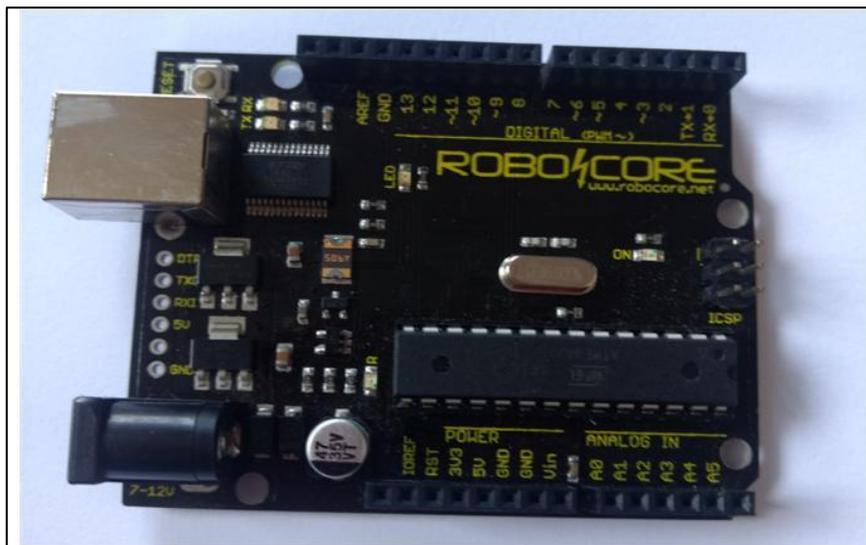


Fonte: dos autores

3.2 Placa BlackBoard UNO R3

A placa BlackBoard UNO R3 é compatível com o Arduino, fabricada pela RoboCore empresa brasileira onde foram realizadas algumas melhorias em relação ao Arduino UNO R3.

Figura 22 - Placa BlackBoard UNO R3



Fonte: dos autores

Segundo a RoboCore (2018) as vantagens desta placa para a Arduino UNO R3, são:

- Os LEDs de comunicação (Tx e Rx) estão dispostos no canto da placa, próximos ao botão de reset, o que permite verificar facilmente se existe uma comunicação serial quando um shield é colocado sobre a placa;
- Possui um LED vermelho para indicar se a fonte usada está com polaridade invertida, sistema também que não permite o funcionamento da placa caso a tensão de entrada esteja invertida;
- A conversão do sinal do computador para o ATmega328 é feita através de um chip FTDI. Este chip é mais robusto e confiável que o microcontrolador usado na conversão de sinais usado na placa Arduino UNO. Esta vantagem faz com que não existam mais problemas de compatibilidade de driver da placa com os diversos sistemas operacionais, como no Windows 8;

- Ela possui acesso direto ao ATmega328 via conector para Placa FTDI, ou seja, se por algum motivo o chip FTDI parar de funcionar a placa não precisa ser descartada, pode continuar sendo gravada externamente;
- Fabricada no Brasil.

A placa BlackBoard em nosso projeto tem a função de executar o programa, controlando as entradas e saídas digitais.

3.3 Buzzer

Buzzer é um componente de geração de ruídos sonoros a partir da excitação elétrica de componentes eletromecânicos ou piezoelétricos. As principais aplicações dos buzzers são alarmes, relógios despertadores, computadores, campainhas, brinquedos e sistemas automatizados. Quando utilizados em sistemas programados é capaz de gerar melodias a partir da alteração da frequência de trabalho (Baú da Eletrônica, 2018).

O buzzer pode ser ativo ou passivo. O ativo tem um circuito oscilador que produz o som, o mesmo não varia. O passivo é um transdutor, como se fosse um pequeno alto falante, o som produzido pode ser controlado a partir da alteração da frequência.

Neste projeto estamos utilizando o buzzer passivo, a fim de termos controle sobre os tons emitidos. A função dele é chamar a atenção do motorista de forma sonora sobre o sistema de acionamento da caçamba, alertando quando a tomada de força for acionada, quando a caçamba estiver subindo ou descendo.

Figura 23 - Buzzer passivo



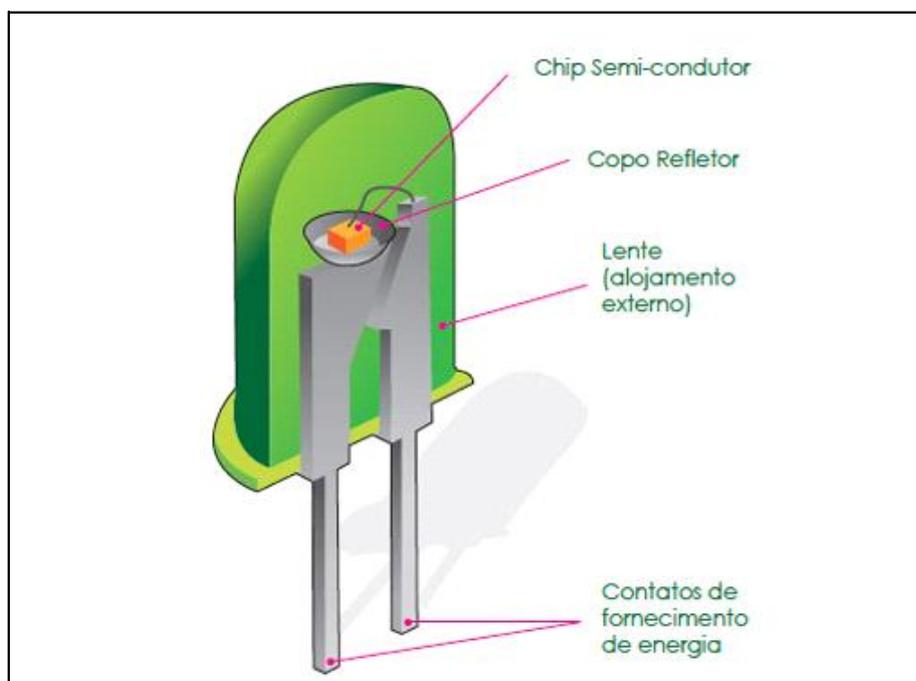
Fonte: dos autores

3.4 LED

A abreviatura LED vem do inglês (Light Emitting Diode) que significa diodo emissor de luz. Componente eletrônico semicondutor, composto de cristal semicondutor de silício ou germânio. O LED é um componente do tipo bipolar, tem dois terminais anodo e catodo. Dependendo de como for polarizado, permite ou não a passagem de corrente elétrica, gerando ou não a emissão de luz (Mundo da Eletrônica, 2018).

A emissão de luz no LED ocorre quando a corrente elétrica percorre o material de junção PN (diodo semicondutor), emitindo radiação. O componente mais importante é o chip semicondutor, responsável pela geração de luz.

Figura 24 - LED Convencional



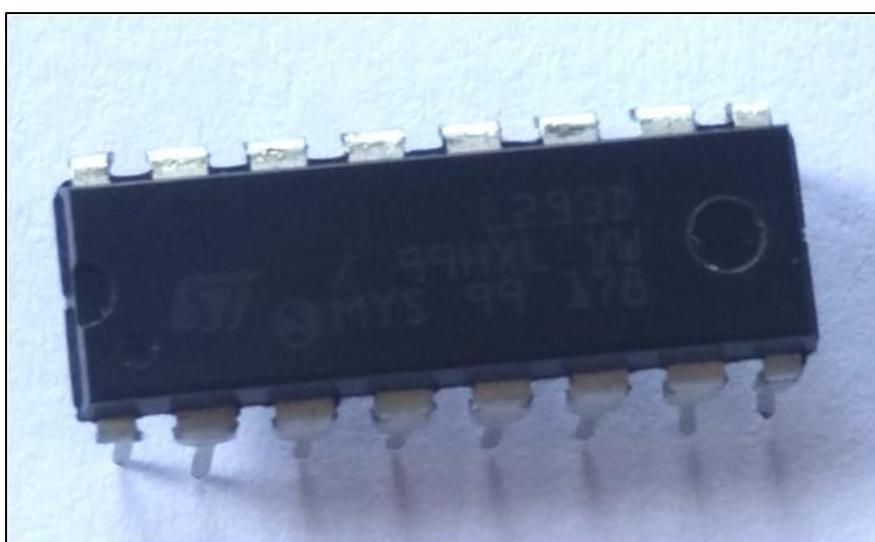
Fonte: <http://www.conexled.com.br/sobre/led/historia/> Acessado em 05/05/2018 às 11:48

Neste projeto estamos utilizando LEDs, com a função de alertar o motorista de forma visual quando a tomada de força for acionada, quando a caçamba estiver subindo ou descendo. Também está sendo utilizado para monitoramento dos acionamentos dos botões, utilizados na simulação do sistema.

3.5 Ponte H - L293D

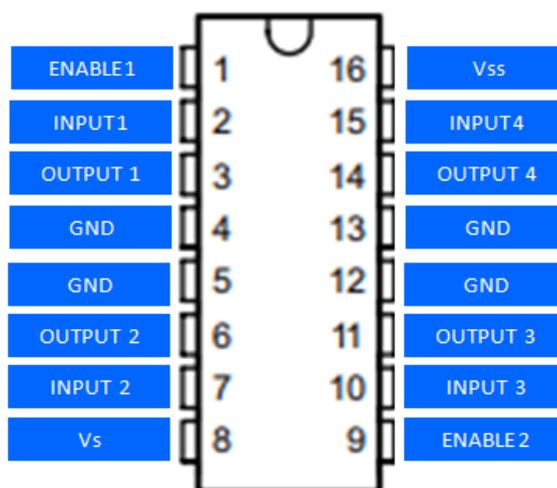
Neste projeto estamos utilizando o Circuito Integrado Ponte H - L293D responsável pelo controle da direção da rotação do motor da caçamba (levanta/abaixa). Na figura 25 é mostrado o circuito integrado L293 e na figura 26 é demonstrado como deve ser realizado sua ligação.

Figura 25 - L293D



Fonte: dos autores

Figura 26 - Montagem do L293D



Fonte: dos autores

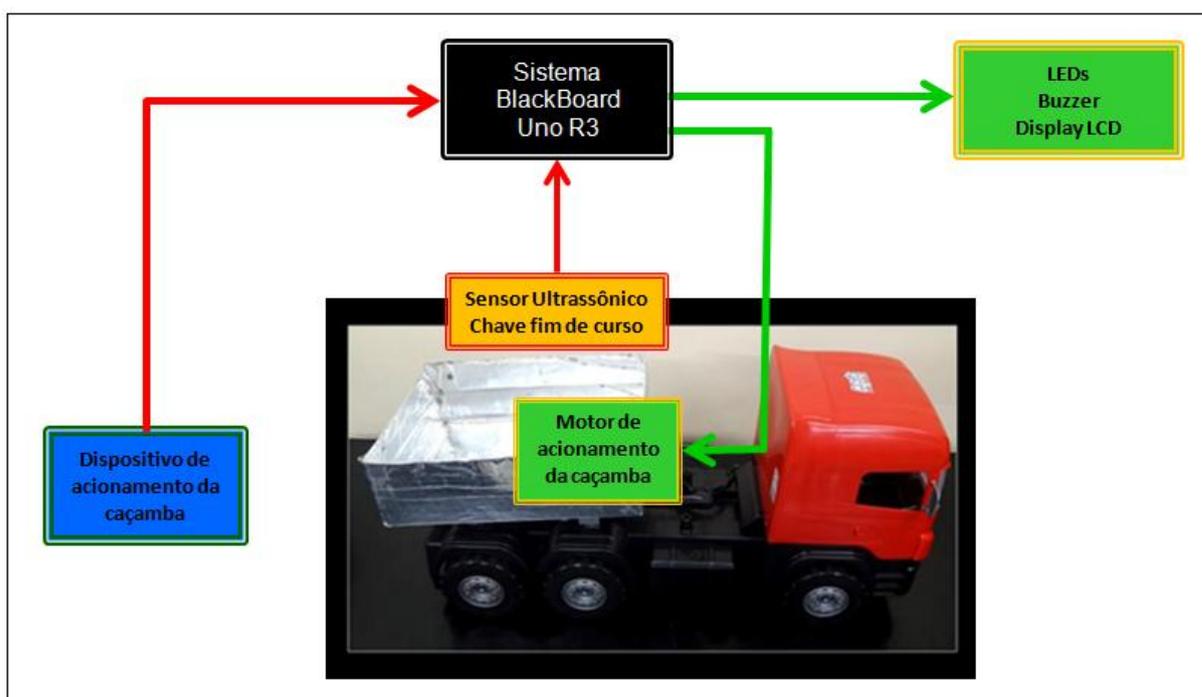
4 DESENVOLVIMENTO DO PROTÓTIPO

4.1 Funcionamento do programa

Na figura 27 é mostrado o esquema das entradas e saídas que serão monitoradas e controladas pelo sistema, sendo eles: entradas (dispositivo de acionamento da caçamba, sensor ultrassônico e chave fim de curso) e as saídas (motor de acionamento da caçamba, Leds, buzzer e display LCD).

No dispositivo de acionamento da caçamba temos: botão da tomada de força, botão aciona (levanta a caçamba), botão abaixa (abaixa a caçamba) e simulador de velocidade.

Figura 27 - Entradas e saídas do sistema



Fonte: dos autores

O funcionamento do programa é descrito na tabela 4, onde é realizado o monitoramento das variáveis: Tomada de força, botão aciona, botão abaixa, sensor,

trava mecânica e velocidade. De acordo com as combinações das variáveis é tomada uma ação.

Tabela 4 - Mapeamento do Programa

Mapeamento do Funcionamento do Programa						
Tomada de força	Botão Aciona	Botão Abaixa	Sensor (cm)	Botão Trava	Velocidade (Km/h)	Ação
0	0	0	≤ 4	1	----	Comando "0"
1	0	0	≤ 4	1	≤ 10	Comando "1"
1	1	0	≤ 4	1	≤ 10	Comando "2"
1	1	0	> 4 e < 10	0	≤ 10	Comando "3"
1	1	0	≥ 10	0	≤ 10	Comando "4"
0	0	1	> 4	0	≤ 10	Comando "5"
0	0	1	≤ 4	1	≤ 10	Comando "6"
0	0	0	> 4 e ≤ 8	0	> 10	Comando "7"
0	0	1	> 4	0	> 10	Comando "8"
0	0	1	≤ 4	1	> 10	Comando "9"
1	0	0	≤ 4	1	> 10	Comando "10"
1	1	0	≤ 4	1	> 10	Comando "10"
0	0	0	≥ 9	0	> 10	Comando "11"
0	0	0	> 4	0	≤ 10	Comando "4"
0	0	0	> 4	1	----	Comando "12"

Fonte: dos autores

A seguir são descritos os comandos que serão utilizados pelo programa de acordo com as combinações da tabela 4.

Comando 0: Mensagem “Sistema Detecta, altura da caçamba e velocidade”, led verde fica piscando indicando que a chave mecânica está acionada. A caçamba está abaixada;

Comando 1: Mensagem “Tomada de Força Ligada e velocidade”. Led vermelho fica piscando, o buzzer fica ligado intermitente. A caçamba está abaixada;

Comando 2: Mensagem “Subida acionada e altura da caçamba”. Led verde fica aceso, led vermelho fica piscando e o buzzer fica ligado intermitente. A caçamba começa a ser basculada;

Comando 3: Mensagem “Caçamba subindo e altura da caçamba”. Led verde fica aceso, led vermelho fica piscando e o buzzer fica ligado intermitente. A caçamba está sendo basculada;

Comando 4: Mensagem: “Caçamba levantada e altura da caçamba”. O botão aciona e a tomada de força são desligados, led verde e vermelho ficam piscando e o buzzer fica ligado intermitente. A caçamba está levantada;

Comando 5: Mensagem: “Descida acionada e altura da caçamba”. Led verde fica apagado, o led verde fica piscando e o buzzer fica ligado intermitente. A caçamba começa a ser abaixada;

Comando 6: Mensagem: “Caçamba abaixada e altura da caçamba”. O botão abaixa é desligado, led verde e vermelho são apagados e o buzzer fica desligado. A caçamba está abaixada;

Comando 7: Mensagem: “Atenção caçamba levantando”, “altura da caçamba”. Leds verde e vermelho ficam piscando, o buzzer fica ligado intermitente; Em seguida aparece outra mensagem: “Descida acionada automaticamente”, “contagem”, é realizado uma contagem regressiva de 5 a 0. Em seguida o estado do botão abaixa é colocado em nível alto. A caçamba está levantada parcialmente;

Comando 8: Mensagem: “Descida acionada e altura da caçamba”. Leds verde e vermelho ficam piscando e o buzzer fica ligado intermitente. A caçamba começa a ser abaixada;

Comando 9: Mensagem: “Caçamba abaixada e altura da caçamba”. O botão abaixa é colocado em nível baixo, led verde e vermelho são apagados e o buzzer fica desligado; Em seguida aparece outra mensagem: “Verifique o sistema”. A caçamba está abaixada;

Comando 10: Mensagem “Tomada de Força Ligada e velocidade”, led verde fica aceso, led vermelho fica piscando, o buzzer fica ligado intermitente, o estado do

botão aciona é colocado em nível baixo evitando que a caçamba seja basculada; Em seguida aparece outra mensagem: “Desligue a tomada, velocidade >10Km/h”. A caçamba está abaixada;

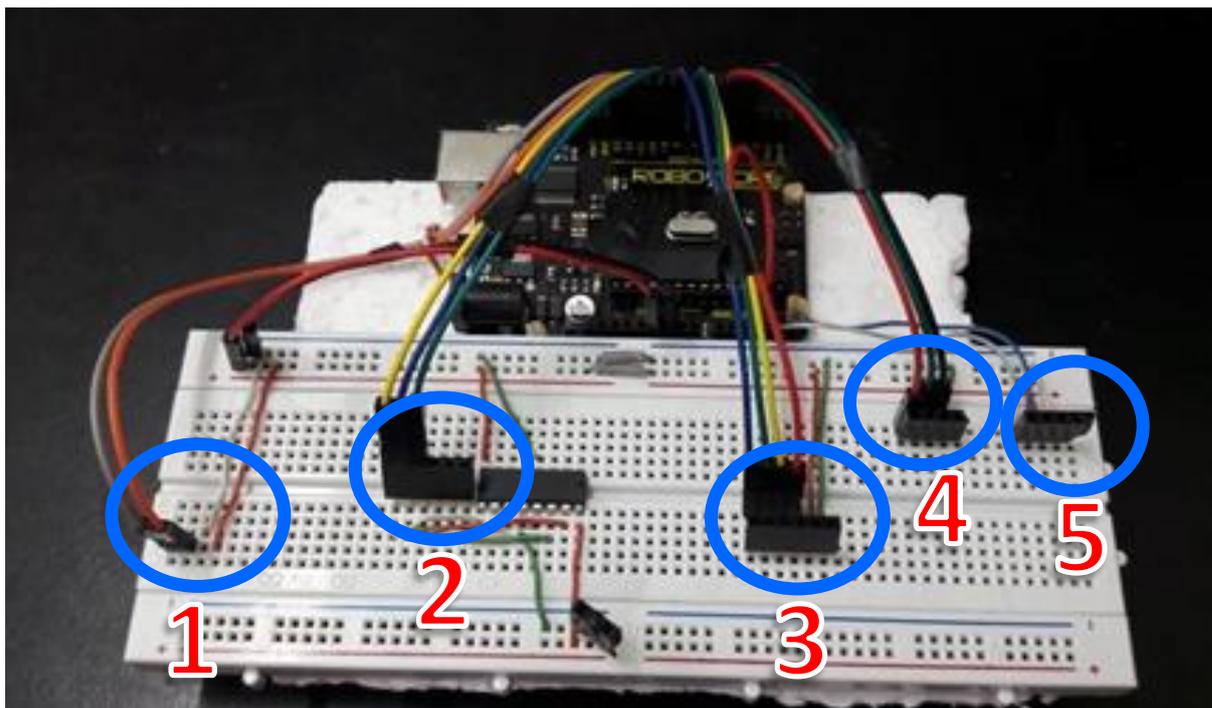
Comando 11: Mensagem: “Velocidade > 10Km/h”, “velocidade”. Leds verde e vermelho ficam piscando, o buzzer fica ligado intermitente; Em seguida aparece outra mensagem: “Descida acionada automaticamente”. Em seguida o estado do botão abaixa é colocado em nível alto. A caçamba é abaixada;

Comando 12: Mensagem: “Caçamba levantada e altura da caçamba”. O botão aciona e a tomada de força são desligados, led verde e vermelho ficam piscando e o buzzer fica ligado intermitente. Em seguida aparece outra mensagem: “ Falha no sistema”, “Verifique a chave fim de curso”. A caçamba está levantada;

4.2 Montagem do protótipo

Na montagem do protótipo foi utilizada uma protoboard que tem a função de fazer a conexão e distribuição das entradas e saídas entre os componentes e a placa BlackBoard UNO R3, conforme mostrado na figura 28.

Figura 28 - Conexão com a BlackBoard UNO R3



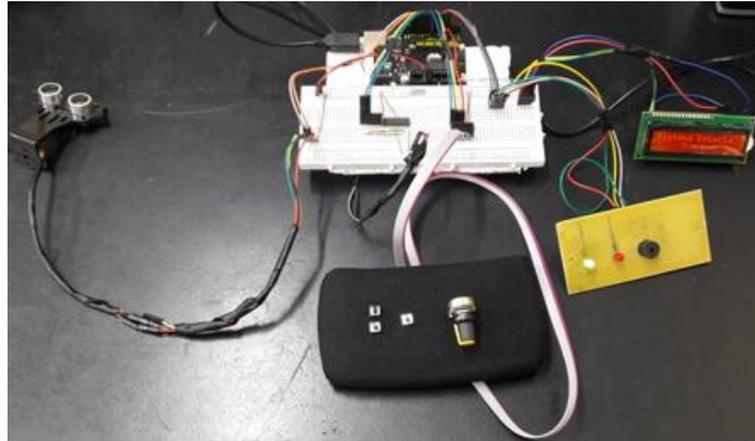
Fonte: dos autores

As entradas e saídas estão distribuídas da seguinte forma:

- 1: Entradas e saídas dos cabos do sensor ultrassônico;
- 2: Entradas e saídas do motor da caçamba;
- 3: Entradas e saídas do controle da caçamba, botões e potenciômetro;
- 4: Saídas dos Leds e buzzer;
- 5: Saídas do display LCD.

Na figura 29 são mostrados os componentes conectados a BlackBoard Uno R3.

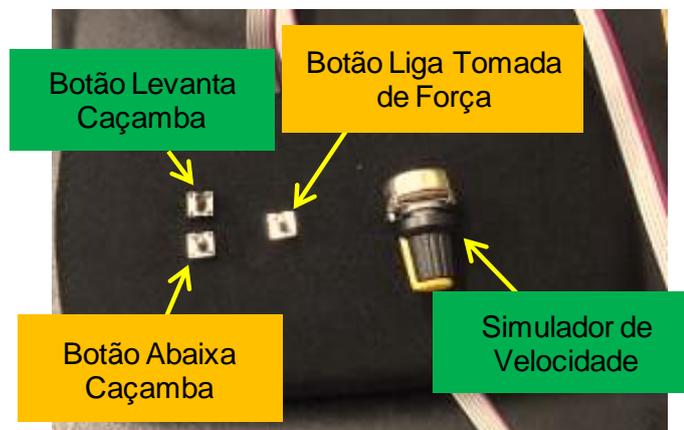
Figura 29 - Componentes conectados a BlackBoard Uno R3



Fonte: dos autores

Na figura 30 é mostrado o dispositivo de acionamento da caçamba com os botões no qual tem a função de ligar a tomada de força, acionar a caçamba, abaixar a caçamba e simular a velocidade do veículo através do potenciômetro.

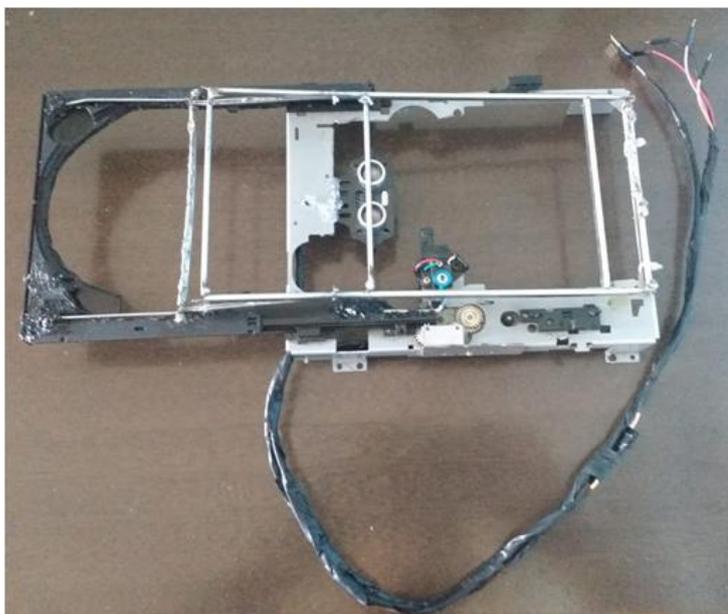
Figura 30 – Dispositivo de acionamento da caçamba



Fonte: dos autores

Para fazermos a simulação do basculamento da caçamba utilizamos um driver de CD juntamente com uma estrutura metálica, construído para simular o acionamento direto por dois cilindros. Este sistema é mostrado na figura 31.

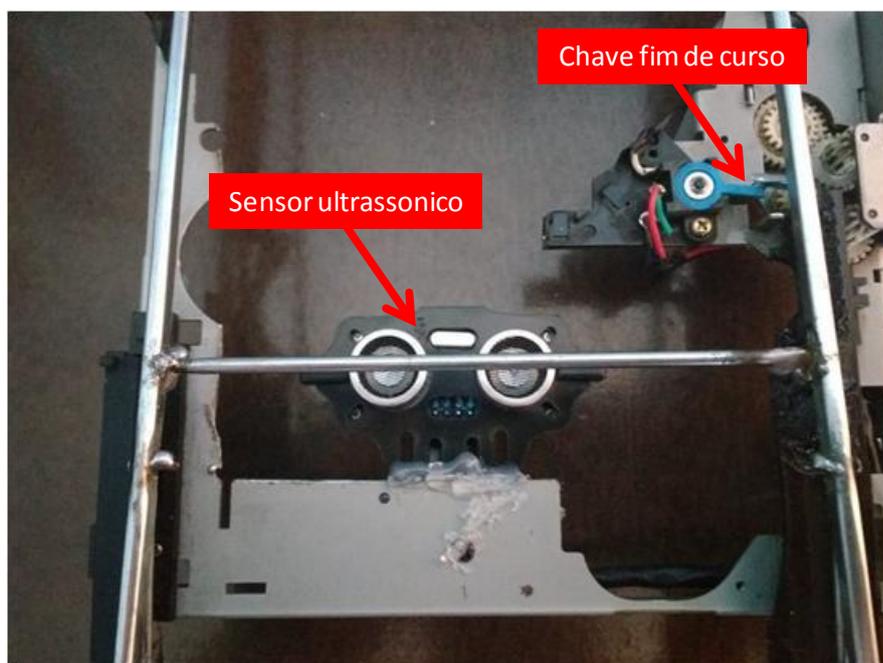
Figura 31 - Sistema de basculamento da caçamba



Fonte: dos autores

Na figura 32 é mostrado o local onde foram instalados o sensor ultrassônico e a chave fim de curso, os mesmos ficarão abaixo da caçamba.

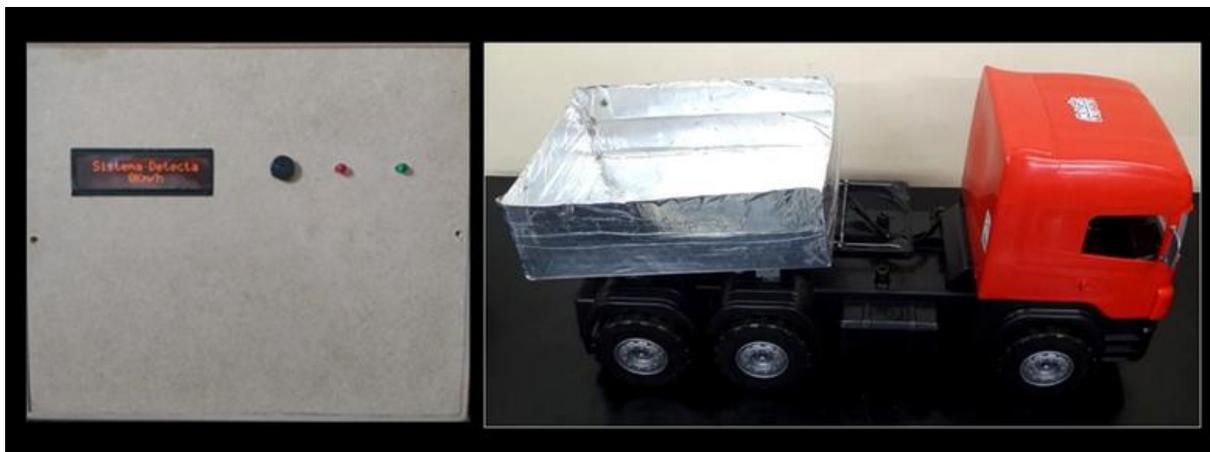
Figura 32 – Local de instalação do sensor e da chave fim de curso



Fonte: dos autores

Nas figuras 33 a 35 é apresentado o protótipo utilizado na apresentação final deste trabalho.

Figura 33 - Protótipo caçamba abaixada



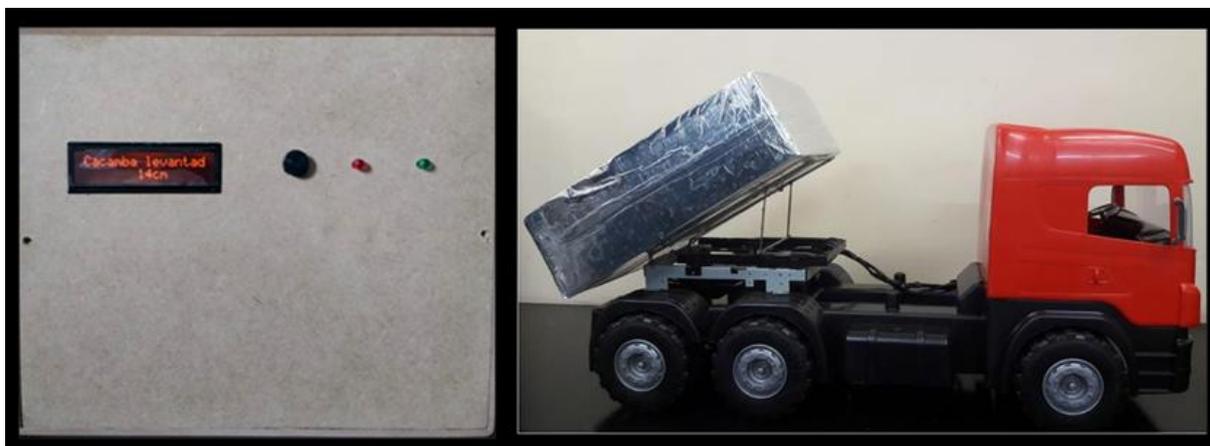
Fonte: dos autores

Figura 34 - Protótipo tomada de força ligada



Fonte: dos autores

Figura 35 - Protótipo caçamba levantada



Fonte: dos autores

4.3 Programação

A programação utilizada neste projeto para o funcionamento do sistema e para o acionamento da caçamba foi a linguagem C. O roteiro completo utilizado neste projeto se encontra no apêndice.

4.4 Análise dos resultados

Para demonstração do funcionamento do sistema adquirimos um caminhão caçamba da Magic Toys conforme mostrado na figura 36.

Figura 36 - Caminhão Basculante Magic Toys



Fonte: dos autores

Para nos aproximar de uma simulação real e ter a certeza que o sistema funcionasse, decidimos que a caçamba fosse basculada pelo sistema e não manualmente. Porém, os recursos obtidos não suportaram erguer a caçamba que pesa aproximadamente 700g. Para solucionarmos este problema substituímos a caçamba por uma pasta plástica que foi encapada com papel laminado, conforme já demonstrado nas figuras anteriores 33 a 35.

Após este ajuste o sistema funcionou conforme havíamos planejado, mostrando-se robusto, seguro e atendendo a Resolução 563 do CONTRAN, referente ao sistema secundário.

Realizamos três amostragens (A1, A2 e A3) cada um contendo 10 testes para verificar a reprodutibilidade da distância detectada pelo sensor no sistema. Os testes foram realizados em três situações: caçamba abaixada, caçamba levantada pelo sistema e caçamba levantada manualmente. Na tabela 5 constam os valores encontrados nos testes.

Tabela 5 – Resultados das distâncias detectadas pelo sensor (cm)

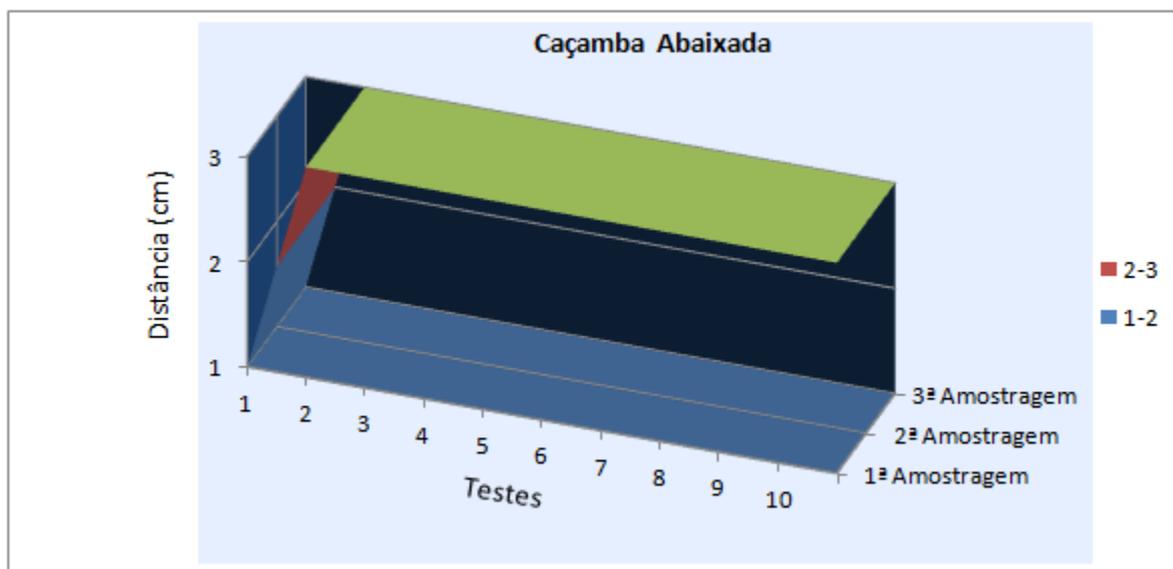
Testes	Caçamba Abaixada			Levantada Pelo Sistema			Levantada Manualmente		
	A1	A2	A3	A1	A2	A3	A1	A2	A3
1	3	3	3	12	13	13	13	14	14
2	3	3	3	13	13	12	14	13	14
3	3	3	3	12	13	13	14	14	14
4	3	3	3	12	12	13	14	14	13
5	3	3	3	12	12	13	14	13	14
6	3	3	3	13	13	12	14	14	13
7	3	3	3	12	13	12	13	13	13
8	3	3	3	13	13	13	14	14	14
9	3	3	3	12	12	13	14	13	14
10	3	3	3	12	12	13	14	14	14

Fonte: dos autores

Com base nos resultados obtidos chegamos a três conclusões:

1ª conclusão: A ausência de variações nas distâncias detectadas pelo sensor ultrassônico quando a caçamba está abaixada é devido ao posicionamento do sensor, nesta situação o sensor está paralelo com a caçamba, ou seja, o sinal emitido pelo sensor é refletido quase que 100%. Podemos visualizar esse efeito no gráfico 1.

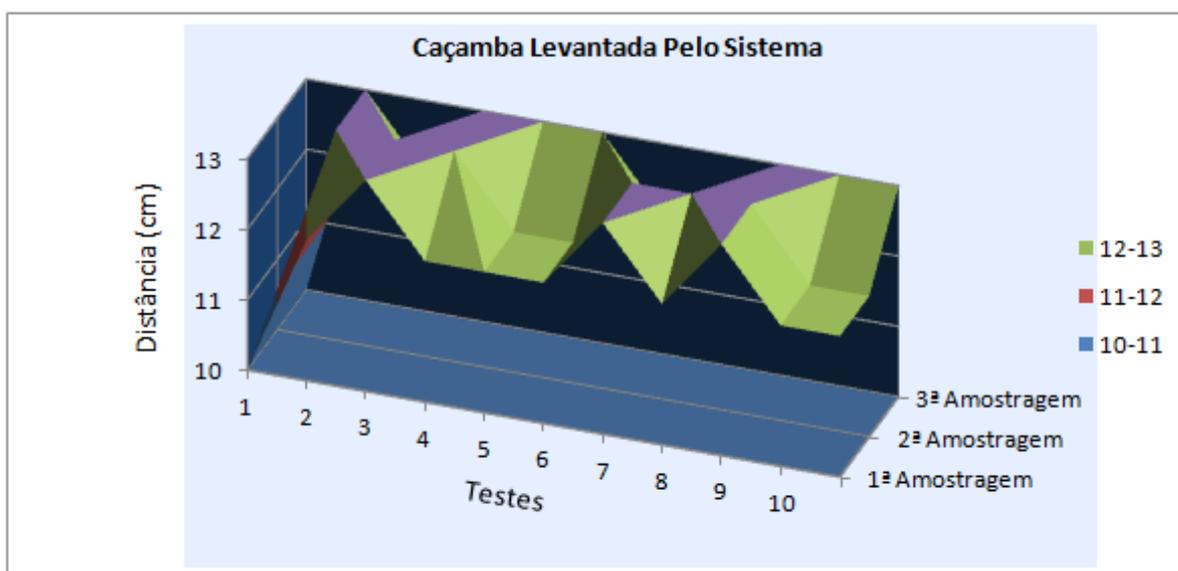
Gráfico 1 - Distância da caçamba abaixada



Fonte: dos autores

2ª conclusão: Com relação às variações nas distâncias detectadas pelo sensor ultrassônico quando a caçamba foi levantada pelo sistema, podemos atribuir a dois fatores. Primeiro: O posicionamento do sensor em relação à caçamba, nesta situação a caçamba está inclinada, ou seja, parte do sinal emitido pelo sensor é perdida na reflexão. Segundo: Os arredondamentos utilizados na conversão do sinal do sensor para distância. Podemos visualizar esse efeito no gráfico 2.

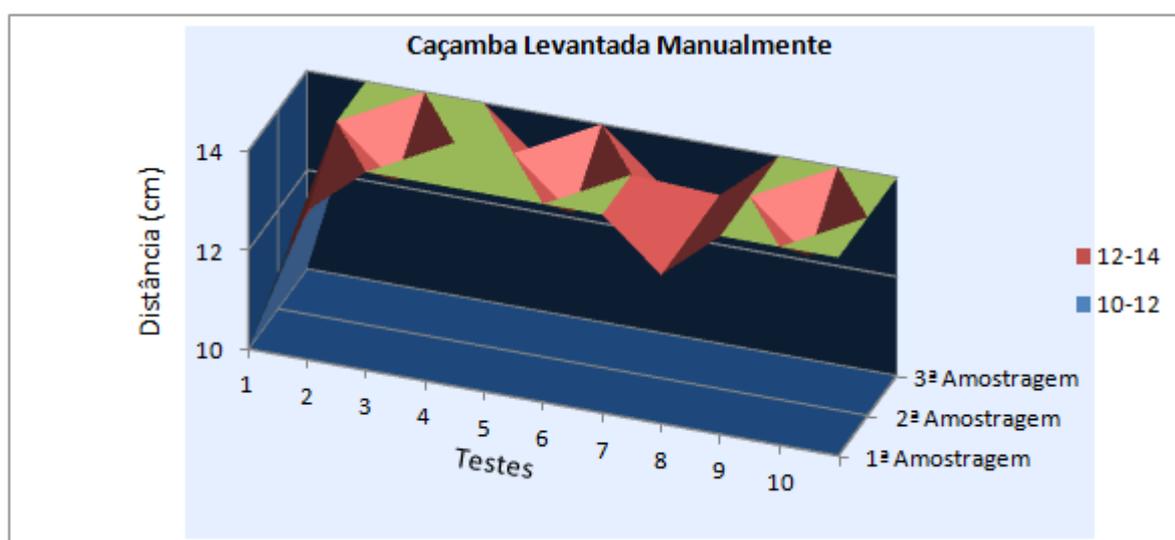
Gráfico 2 – Distância da caçamba levantada pelo sistema



Fonte: dos autores

3ª conclusão: As variações nas distâncias detectadas pelo sensor ultrassônico quando a caçamba foi levantada manualmente, podemos atribuir aos mesmos fatores da conclusão 2. A diferença da distância máxima entre as duas situações, é devido ao fato de no caso anterior a subida da caçamba estar programada para parar ao atingir a distância ≥ 10 cm, já neste caso onde a caçamba foi levantada manualmente o valor considerado foi da caçamba totalmente levantada. Podemos visualizar esse efeito no gráfico 3.

Gráfico 3 - Distância da caçamba levantada manualmente



Fonte: dos autores

5 CONSIDERAÇÕES FINAIS

Através do desenvolvimento deste projeto Detecção de Caçamba Levantada em Caminhão Basculante, adquirimos maior experiência em sistemas embarcados aprendidos em sala de aula, onde hardware e software se comunicam.

Os resultados propostos no início foram obtidos e com algumas melhorias sugeridas pelos nossos professores orientadores Wagner Massarope, Eliel Wellington Marcelino e Fernando Garup Dalbo.

A placa BlackBoard Uno R3 (Unidade de controle) foi de grande importância para realização deste projeto, por ser didática e de fácil aplicação, onde colocamos em prática a matéria de programação envolvendo código e lógica.

5.1 Trabalhos futuros

As nossas sugestões de propostas futuras são para que o sistema seja testado com outros tipos de sensores como, por exemplo, sensor tipo Radar utilizados em veículos atuais.

Testar o sistema em um ambiente externo para verificar o comportamento do sensor em ambientes hostis.

Para dar continuidade a este projeto testar o sistema em um veículo real, a fim de verificar a sua real implementação.

6 REFERÊNCIAS BIBLIOGRÁFICAS

Schulz, Freddy Johnatan, 1982. Análise e projeto de sistema de acionamento para caçamba basculante, Universidade Federal de Uberlândia - 2015.

Marcelino, E., Acústica Automotiva, Notas de aulas, ministradas no 1º Sem.2018 na Faculdade de Tecnologia, FATEC – Santo André.

McRoberts, Michael, Arduino Básico 2ª Edição, Editora Novatec 03/2015.

Informações gerais sobre tomada de força introdução - Scania CV AB 2016, Sweden – Edição 11 pt-BR

Manutenção & Tecnologia, Tomada de força. 27 abr. de 2012.

<http://www.revistamt.com.br/Materias/Exibir/a-forca-do-motor-para-a-movimentacao-do-implemento> - Acessado em 04/04/2018 às 19:30).

CONTRAN RESOLUÇÃO N°563, de 25 de Novembro de 2015.

Norma ABNT NBR 16141 – Implementos rodoviários – Sistemas de Travamento e levantamento da caçamba basculante – requisitos (Primeira edição 14/01/2013).

<http://www.pastre.com.br> – Acessado em 10/03/2018 às 19:15.

DataSheet HC-SR04

<http://www.mecatronicaatual.com.br/educacao/1598-sensores-ultra-sonicos>
(Acessado em 20/08/2017 às 21:10).

<https://www.mecanicaindustrial.com.br/598-o-que-e-um-sensor-ultrassonico/>
(Acessado em 03/04/2018 às 19:25).

Gomes; Sinésio Raimundo. Sensores Ultrassônicos,atualização: 20/03/2017.
<http://controleeautomacaoindustrial3.blogspot.com/2017/> (Acessado em 03/04/2018 às 20:05).

<https://www.filipeflop.com/blog/sensor-ultrassonico-hc-sr04-ao-arduino/>
(Acessado em 03/04/2018 às 20:10).

<https://multilogica-shop.com/arduino-uno-r3> (Acessado em 16/04/2018 às 19:45).

Placa BlackBoard Robocore

<https://www.robocore.net/loja/produtos/arduino-blackboard.html>

(Acessado em 28/04/2018 às 09:40).

<http://www.baudaeletronica.com.br/componentes-eletronicos/buzzer>

(Acessado em 28/04/2018 às 10:53).

<https://www.mundodaeletrica.com.br/o-que-e-um-led/>

(Acessado em 28/04/2018 às 12:10).

<http://www.conexled.com.br/sobre/led/historia/>

(Acessado em 28/04/2018 às 12:38).

DataSheet Ponte H – L293D

Arduino & Cia – Arduino Uno

<http://www.arduinoecia.com.br> (Acessado em: 04/2017)

Arduino Uno. 29 nov. 2013.

<https://www.embarcados.com.br/arduino-uno/> (Acessado em 14/04/2018 às 20:45)

Macedo, Felipe - material e vídeos

<https://www.filipeflop.com> (Acessado em: 05/2018)

Costa, Rodrigo. Como medir distâncias com sensor ultrassônico HC-SR04 e Arduino. 06 abr. 2017. <https://eletronicaparatodos.com/como-medir-distancias-com-sensor-ultrassonico-hc-sr04-e-arduino/> (Acessado em 21/04/2018 às 11:10).

7 APÊNDICE – CÓDIGO DO PROGRAMA

```

#include "Ultrasonic.h"
#include "Wire.h"
#include "LiquidCrystal_I2C.h"

LiquidCrystal_I2C lcd(0x20,16,2); // LCD de 16x2 no endereço 0x20

//Declaração das constantes pinos digitais e analógicos.
const int trigPin = 13; //Pino digital 13 setado Trigger
const int echoPin = 12; //Pino digital 12 setado Echo

int enable1 = 11; // Pino digital 11 Habilita Ponte H
int input1 = 10; // Pino digital 10 Dir1 do motor
int input2 = 9; // Pino digital 9 Dir2 do motor

int botao_tomada=8; // Pino digital 8 botão tomada de força
int var_tomada=0; // valor instantaneo enviado pelo botão
int var2_tomada=0; // valor guardado
int estado_tomada=0; // guarda o valor 0 ou 1 (HIGH ou LOW)

int botao_aciona=7; // Pino digital 7 botão aciona caçamba
int var_aciona=0; // valor instantaneo enviado pelo botão
int var2_aciona=0; // valor guardado
int estado_aciona=0; // guarda o valor 0 ou 1 (HIGH ou LOW)

int botao_abaixa=6; // Pino digital 6 botão abaixa caçamba
int var_abaixa=0; // valor instantaneo enviado pelo botão
int var2_abaixa=0; // valor guardado
int estado_abaixa=0; // guarda o valor 0 ou 1 (HIGH ou LOW)

const int pinoBuzzer = 5; // Pino digital 5 BUZZER
const int ledVermelho = 4; // Pino digital 4 Led vermelho
const int ledVerde = 3; // Pino digital 3 Led verde

const int pinoPot = A3; // Pino analógico 3 Potenciometro
int leituraA3; //Variável que armazena o valor lido na porta analógica A3
int velocidade;
int calc;

int botao_trava=2; // Pino digital 2 Botão trava mecânica
int var_trava=0; // valor instantaneo enviado pelo botão
int var2_trava=0; // valor guardado
int estado_trava=0; // guarda o valor 0 ou 1 (HIGH ou LOW)

Ultrasonic ultrasonic(trigPin,echoPin); // Inicializando os pinos do sensor
int distancia;
String result;

```

```

int comando=0;

void setup()
{
  Serial.begin(9600); //Iniciando o serial monitor
  pinMode(echoPin, INPUT); // Define o pino 12 como entrada
  pinMode(trigPin, OUTPUT); //Define o pino 13 como saída
  pinMode(pinoBuzzer, OUTPUT); //Define o pino 5 como saída
  pinMode(ledVerde,OUTPUT); //Define o pino 3 como saída.
  pinMode(ledVermelho,OUTPUT); //Define o pino 4 como saída.

  pinMode(botao_tomada,INPUT); //Define o pino 8 como entrada
  pinMode(botao_aciona,INPUT); //Define o pino 7 como entrada
  pinMode(botao_abaixa,INPUT); //Define o pino 6 como entrada
  pinMode(botao_trava,INPUT); //Define o pino 2 como entrada
  pinMode(pinoPot, INPUT); //Define o pino A3 como entrada

  pinMode(enable1, OUTPUT); // Define o pino 11 como saída
  pinMode(input1, OUTPUT); // Define o pino 10 como saída
  pinMode(input2, OUTPUT); // Define o pino 9 como saída

  lcd.init();      // Inicializando o LCD
  lcd.backlight(); // Ligando o BackLight do LCD

  lcd.clear();
  lcd.setCursor(4,0);
  lcd.print("Carregando");
  lcd.setCursor (5,1);
  lcd.print("sistema");
  delay(5000);

  Serial.println("Carregando sistema");
  digitalWrite(ledVerde,HIGH);
  tone(pinoBuzzer,1500);//Liga o BUZZER
  digitalWrite(ledVermelho, HIGH); // liga o led
  digitalWrite(ledVerde, HIGH);
  delay(400);
  digitalWrite(ledVermelho, LOW); // desliga o led
  digitalWrite(ledVerde, LOW);
  delay(400);
  noTone(pinoBuzzer);//Desliga o BUZZER
}

```

```
void loop()
{

  hcsr04(); // Chama a função do sensor HCSR04

  Serial.print(distancia);
  Serial.println(" cm");
  delay(100);

  //Botão tomada de força
  var_tomada=digitalRead(botao_tomada); // ler o estado do botão tomada de força
  if ((var_tomada == HIGH) && (var2_tomada == LOW))
  {
    estado_tomada = 1 - estado_tomada;
    delay(20); // de-bouncing
  }

  var2_tomada=var_tomada;
  if (estado_tomada == 1)
  {
    digitalWrite(ledVermelho, HIGH);
  }

  else
  {
    digitalWrite(ledVermelho, LOW);
  }

  //Botão aciona caçamba
  var_aciona=digitalRead(botao_aciona); // ler o valor enviado pelo botão: "HIGH" ou "LOW"
  if ((var_aciona == HIGH) && (var2_aciona == LOW))
  {
    estado_aciona = 1 - estado_aciona;
    delay(20); // de-bouncing
  }

  var2_aciona=var_aciona;
  if (estado_aciona == 1)
  {
    digitalWrite(ledVermelho, HIGH);
  }
  else
  {
    digitalWrite(ledVermelho, LOW);
  }
}
```

```

//Botão abaixa caçamba
var_abaixa=digitalRead(botao_abaixa); // ler o valor enviado pelo botão: "HIGH" ou "LOW"
if ((var_abaixa == HIGH) && (var2_abaixa == LOW))
{
    estado_abaixa = 1 - estado_abaixa;
    delay(20); // de-bouncing
}

var2_abaixa=var_abaixa;
if (estado_abaixa == 1)
{
    digitalWrite(ledVermelho, HIGH);
}
else
{
    digitalWrite(ledVermelho, LOW);
}

//Botão chave mecânica
estado_trava = digitalRead (botao_trava);
if (estado_trava == HIGH)
{
    digitalWrite(ledVerde, HIGH);
    delay(200);
    digitalWrite(ledVerde, LOW);
    delay(200);
    digitalWrite(ledVerde, HIGH);
    delay(100);
    digitalWrite(ledVerde, LOW);
    delay(100);
}
else
{
    digitalWrite(ledVerde,LOW);
}

//Verificando as variaveis
if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==0)&&(distancia<=4 && estado_trava==1))
{
    comando=0;
}

else if((estado_tomada==1)&&(estado_aciona==0)&&(estado_abaixa==0)&&(distancia<=4 && estado_trava==1)&&(velocidade <=10))
{
    comando=1;
}

```

```

else          if((estado_tomada==1)&&(estado_aciona==1)&&(estado_abaixa==0)&&(distancia<=4      &&
estado_trava==1)&&(velocidade <=10))
{
    comando=2;
}

else  if((estado_tomada==1)&&(estado_aciona==1)&&(estado_abaixa==0)&&(distancia  >3  &&  distancia<10  &&
estado_trava==0)&&(velocidade <=10))
{
    comando=3;
}

else          if((estado_tomada==1)&&(estado_aciona==1)&&(estado_abaixa==0)&&(distancia>=10      &&
estado_trava==0)&&(velocidade <=10))
{
    comando=4;
}

else          if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==1)&&(distancia>4      &&
estado_trava==0)&&(velocidade <=10))
{
    comando=5;
}

else if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==1)&&(distancia<=4 || estado_trava==1)&&(velocidade
<=10))
{
    comando=6;
}

else  if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==0)&&(distancia>4  &&  distancia<=8  &&
estado_trava==0)&&(velocidade >10))
{
    comando=7;
}

else          if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==1)&&(distancia>4      &&
estado_trava==0)&&(velocidade >10))
{
    comando=8;
}

else if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==1)&&(distancia<=4 || estado_trava==1)&&(velocidade
>10))
{
    comando=9;
}

```

```

else          if((estado_tomada==1)&&(estado_aciona==0)&&(estado_abaixa==0)&&(distancia<=4
estado_trava==1)&&(velocidade >10))          &&
{
  comando=10;
}

else          if((estado_tomada==1)&&(estado_aciona==1)&&(estado_abaixa==0)&&(distancia<=4
estado_trava==1)&&(velocidade >10))          &&
{
  comando=10;
}

else          if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==0)&&(distancia>=9
estado_trava==0)&&(velocidade >10))          &&
{
  comando=11;
}

else          if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==0)&&(distancia>4
estado_trava==0)&&(velocidade <=10))          &&
{
  comando=4;
}

else if((estado_tomada==0)&&(estado_aciona==0)&&(estado_abaixa==0)&&(distancia>4 && estado_trava==1))
{
  comando=12;
}

switch (comando)
{
  case 0:
    delay(100);
    Serial.println("Sistema Detecta Caçamba");
    lcd.clear();
    lcd.setCursor(1, 0);
    lcd.print("Sistema Detecta");
    lcd.setCursor (7,1);
    lcd.print(distancia);
    lcd.println("cm   ");
    delay(800);
    lcd.setCursor (6,1);
    lcd.print(velocidade);
    lcd.println("Km/h   ");
    delay(800);
    break;

  case 1:
    delay(100);

```

```

lcd.clear();
lcd.setCursor(1, 0);
lcd.print("Tomada Ligada");
lcd.setCursor(6, 1);
lcd.print(velocidade);
lcd.println("Km/h   ");
delay(300);
tone(pinoBuzzer, 1500); //Liga o BUZZER
digitalWrite(ledVermelho, HIGH); // liga o led
delay(400);
digitalWrite(ledVermelho, LOW); // desliga o led
delay(400);
noTone(pinoBuzzer); //Desliga o BUZZER
break;

```

```

case 2:
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Subida acionada!");
lcd.setCursor(7, 1);
lcd.print(distancia);
lcd.println("cm   ");
delay(300);
digitalWrite(ledVerde, HIGH);
tone(pinoBuzzer, 1500);
digitalWrite(ledVermelho, HIGH);
delay(300);
digitalWrite(ledVermelho, LOW);
delay(300);
noTone(pinoBuzzer);
rotateLeft();
parar();
break;

```

```

case 3:
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Cacamba subindo!");
lcd.setCursor(7, 1);
lcd.print(distancia);
lcd.println("cm   ");
delay(300);
tone(pinoBuzzer, 1500);
digitalWrite(ledVerde, HIGH);
digitalWrite(ledVermelho, HIGH);
delay(200);
digitalWrite(ledVermelho, LOW);
noTone(pinoBuzzer);

```

```
delay(200);
rotateLeft();
parar();
break;

case 4:
estado_aciona = 0;
estado_tomada = 0;
parar();
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Cacamba levantada");
lcd.setCursor (7,1);
lcd.print(distancia);
lcd.println("cm    ");
delay(300);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
tone(pinoBuzzer,1500);
digitalWrite(ledVermelho,HIGH);
delay(50);
digitalWrite(ledVermelho, LOW);
delay(50);
noTone(pinoBuzzer);
digitalWrite(ledVerde,HIGH);
break;

case 5:
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Descida acionada");
lcd.setCursor (7,1);
lcd.print(distancia);
lcd.println("cm    ");
delay(300);
digitalWrite(ledVerde,LOW);
tone(pinoBuzzer,1500);
digitalWrite(ledVermelho,HIGH);
delay(200);
digitalWrite(ledVermelho,LOW);
noTone(pinoBuzzer);
rotateRight();
parar();
break;

case 6:
estado_abaixa = 0;
parar();
```

```
noTone(pinoBuzzer);
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Cacamba abaixada");
lcd.setCursor (7,1);
lcd.print(distancia);
lcd.println("cm   ");
delay(3000);
digitalWrite(ledVermelho,LOW);
break;
```

```
case 7:
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Atencao Cacamba");
lcd.setCursor (0,1);
lcd.print ("Levantando");
delay(300);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Distancia:");
lcd.setCursor(7, 1);
lcd.print (distancia);
lcd.println("cm   ");
tone (pinoBuzzer,262,200);
digitalWrite(ledVerde,HIGH);
digitalWrite(ledVermelho,HIGH);
delay(200);
tone (pinoBuzzer,960,200);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
delay (200);
tone (pinoBuzzer,262,200);
digitalWrite(ledVerde,HIGH);
digitalWrite(ledVermelho,HIGH);
delay(200);
tone (pinoBuzzer,960,200);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
delay (200);
tone (pinoBuzzer,262,200);
digitalWrite(ledVerde,HIGH);
digitalWrite(ledVermelho,HIGH);
delay(200);
tone (pinoBuzzer,960,200);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
delay (200);
```

```
tone (pinoBuzzer,262,200);
digitalWrite(ledVerde,HIGH);
digitalWrite(ledVermelho,HIGH);
delay(200);
tone (pinoBuzzer,960,200);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
delay (200);
tone(pinoBuzzer,262,200);
delay(400);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Descida Acionada");
lcd.setCursor (0,1);
lcd.println("Automaticamente ");
delay(3000);
lcd.clear();
lcd.setCursor(3, 0);
lcd.print ("Contagem:");
lcd.setCursor(7, 1);
lcd.print ("5 ");
noTone(pinoBuzzer);
delay(300);
tone(pinoBuzzer, 1500);
delay(300);
lcd.clear();
lcd.setCursor(3, 0);
lcd.print ("Contagem:");
lcd.setCursor(7, 1);
lcd.print ("4 ");
noTone(pinoBuzzer);
delay(300);
tone(pinoBuzzer, 1500);
delay(300);
lcd.clear();
lcd.setCursor(3, 0);
lcd.print ("Contagem:");
lcd.setCursor(7, 1);
lcd.print ("3 ");
noTone(pinoBuzzer);
delay(300);
tone(pinoBuzzer, 1500);
delay(300);
lcd.clear();
lcd.setCursor(3, 0);
lcd.print ("Contagem:");
lcd.setCursor(7, 1);
lcd.print ("2 ");
noTone(pinoBuzzer);
delay(300);
```

```
tone(pinoBuzzer,1500);
delay(300);
lcd.clear();
lcd.setCursor(3, 0);
lcd.print ("Contagem:");
lcd.setCursor(7, 1);
lcd.print ("1 ");
noTone(pinoBuzzer);
delay(300);
tone(pinoBuzzer,1500);
delay(300);
lcd.clear();
lcd.setCursor(3, 0);
lcd.print ("Contagem:");
lcd.setCursor(7, 1);
lcd.print ("0 ");
tone(pinoBuzzer,1500);
delay(300);
estado_abaixa = 1;
parar();
break;

case 8:
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Descida acionada");
lcd.setCursor (7,1);
lcd.print(distancia);
lcd.println("cm ");
delay(300);
digitalWrite(ledVerde,LOW);
tone(pinoBuzzer,1500);
digitalWrite(ledVerde, HIGH);
digitalWrite(ledVermelho,HIGH);
delay(200);
digitalWrite(ledVerde, LOW);
digitalWrite(ledVermelho,LOW);
noTone(pinoBuzzer);
rotateRight();
parar();
break;

case 9:
estado_abaixa = 0;
parar();
noTone(pinoBuzzer);
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
```

```
lcd.print("Cacamba abaixada");  
lcd.setCursor(7,1);  
lcd.print(distancia);  
lcd.println("cm  ");  
delay(3000);  
lcd.clear();  
lcd.setCursor(4, 0);  
lcd.print("Verifique");  
lcd.setCursor(3,1);  
lcd.print(" o sistema ");  
delay(5000);  
digitalWrite(ledVermelho,LOW);  
break;
```

```
case 10:  
delay(50);  
estado_aciona = 0;  
parar();  
lcd.clear();  
lcd.setCursor(1, 0);  
lcd.print("Tomada Ligada");  
lcd.setCursor(6,1);  
lcd.print(velocidade);  
lcd.println("Km/h  ");  
delay(200);  
digitalWrite(ledVerde,HIGH);  
tone(pinoBuzzer,1500);  
digitalWrite(ledVermelho, HIGH);  
delay(200);  
digitalWrite(ledVermelho, LOW);  
delay(200);  
noTone(pinoBuzzer);  
delay(50);  
tone(pinoBuzzer,1500);  
delay(100);  
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print("Desligue a Tomad");  
lcd.setCursor(0,1);  
lcd.print("Velocid >10 Km/h");  
delay(2000);  
noTone(pinoBuzzer);  
break;
```

```
case 11:  
delay(100);  
lcd.clear();  
lcd.setCursor(0,0);  
lcd.print("Velocid >10 Km/h");  
lcd.setCursor(6,1);
```

```
lcd.print(velocidade);
lcd.println("Km/h   ");
delay(600);
tone (pinoBuzzer,262,200); // Toca o buzzer em tonalidade de dó, em (262Hz)
digitalWrite(ledVerde,HIGH);
digitalWrite(ledVermelho,HIGH);
delay(200);
tone (pinoBuzzer,960,200);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
delay (200);
tone (pinoBuzzer,262,200);
digitalWrite(ledVerde,HIGH);
digitalWrite(ledVermelho,HIGH);
delay(200);
tone (pinoBuzzer,960,200);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
delay (200);
tone (pinoBuzzer,262,200);
digitalWrite(ledVerde,HIGH);
digitalWrite(ledVermelho,HIGH);
delay(200);
tone (pinoBuzzer,960,200);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
delay (200);
tone(pinoBuzzer,262,200);
delay(400);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Descida Acionada");
lcd.setCursor (0,1);
lcd.println("Automaticamente ");
delay(3000);
noTone(pinoBuzzer);
delay(200);
tone(pinoBuzzer, 1500);
delay(200);
noTone(pinoBuzzer);
delay(200);
tone(pinoBuzzer, 1500);
delay(200);
noTone(pinoBuzzer);
delay(200);
tone(pinoBuzzer, 1500);
delay(200);
noTone(pinoBuzzer);
delay(200);
tone(pinoBuzzer, 1500);
```

```
delay(200);
noTone(pinoBuzzer);
delay(200);
tone(pinoBuzzer,1500);
delay(200);
tone(pinoBuzzer,1500);
rotateRight();
delay(300);
parar();
noTone(pinoBuzzer);
delay(400);
break;

case 12:
estado_aciona = 0;
estado_tomada = 0;
parar();
delay(100);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Cacamba levantada");
lcd.setCursor (7,1);
lcd.print(distancia);
lcd.println("cm ");
delay(1000);
digitalWrite(ledVerde,LOW);
digitalWrite(ledVermelho,LOW);
tone(pinoBuzzer,1500);
digitalWrite(ledVermelho,HIGH);
delay(50);
digitalWrite(ledVermelho, LOW);
delay(50);
noTone(pinoBuzzer);
digitalWrite(ledVerde,HIGH);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Falha no sistema");
lcd.setCursor (7,1);
lcd.print(distancia);
lcd.println("cm ");
delay(1000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print ("Verifique Chave");
lcd.setCursor (2,1);
lcd.print("fim de curso");
delay(1000);
break;
}
```

```
}
```

```
void hcsr04() //Função que calcula a distância
{
  digitalWrite(trigPin, LOW); //Seta o pino 13 com pulso baixo "LOW"
  delayMicroseconds(2); //Delay de 2 microsegundos
  digitalWrite(trigPin, HIGH); //Seta o pino 13 com pulso alto "HIGH"
  delayMicroseconds(10); // Delay de 10 microsegundos
  digitalWrite(trigPin, LOW); //Seta o pino 13 com pulso baixo "LOW"
  distancia = (ultrasonic.Ranging(CM)); // VARIÁVEL GLOBAL RECEBE O VALOR DA DISTÂNCIA MEDIDA
  delay(100);
  // Função RANGING, faz a conversão do tempo de resposta do ECHO em
  //centímetros e armazena na variável "distancia"

  leituraA3 = analogRead(pinoPot); //Faz a leitura da porta analógica(Valor lido em bits de 0 A 1023 bits)
  calc= leituraA3/51,1; //(51,1 relação da quantidade de BITS por Km/h)
  velocidade=calc;
  Serial.print(velocidade);
  Serial.println(" Km/h");
}
}
```

```
void rotateLeft() //Função de Giro do Motor
{
  digitalWrite(enable1, HIGH);
  digitalWrite(input2, LOW);
  digitalWrite(input1, HIGH);
  delay(75);
}
}
```

```
void rotateRight()
{
  digitalWrite(enable1, HIGH);
  digitalWrite(input1, LOW);
  digitalWrite(input2, HIGH);
  delay(50);
}
}
```

```
void parar()
{
  digitalWrite(enable1, LOW);
  digitalWrite(input1, LOW);
  digitalWrite(input2, LOW);
}
}
```

8 ANEXO – RESOLUÇÃO Nº563 DO CONTRAN

RESOLUÇÃO Nº 563, DE 25 DE NOVEMBRO DE 2015

Dispõe sobre o sistema de segurança para a circulação de veículos e implementos rodoviários do tipo carroceria basculante.

O CONSELHO NACIONAL DE TRÂNSITO (CONTRAN), usando da competência que lhe confere o art. 12, inciso I, da Lei nº 9.503, de 23 de setembro de 1997, que instituiu o Código de Trânsito Brasileiro (CTB), e conforme o Decreto nº 4.711, de 29 de maio de 2003, que trata da coordenação do Sistema Nacional de Trânsito (SNT);

Considerando o disposto no art. 103 do CTB, que determina que o veículo só poderá transitar pela via quando atendidos os requisitos e as condições de segurança estabelecidos no Código de Trânsito Brasileiro e em normas do CONTRAN;

Considerando a necessidade de regulamentar a circulação de veículos e implementos rodoviários do tipo de carroceria basculante;

Considerando o disposto nos processos de número 80000.003354/2014-59, 80000.005901/2014-31, 80000.010253/2014-34 e 80020.001175/2014-49;

RESOLVE:

Art. 1º Esta Resolução dispõe sobre o sistema de segurança para a circulação de veículos e implementos rodoviários do tipo carroceria basculante.

Art. 2º O disposto nesta Resolução não se aplica ao caminhão-trator sem sistema hidráulico, não destinado à operação com basculante.

Art. 3º Os seguintes sistemas de segurança são definidos na norma ABNT NBR 16141 e apresentados a seguir:

I – dispositivo de segurança primário – dispositivo que impede o acionamento da tomada de força de forma involuntária e de modo que, para o acionamento, sejam necessários dois comandos de acionamentos ou um comando de dois estágios;

II – dispositivo de segurança secundário – aviso visual e sonoro, com intuito de alertar o operador sobre o acionamento da tomada de força, sendo que o aviso visual deverá ser colocado na altura do painel e no campo visual do operador;

III – dispositivo de segurança terciário – dispositivo eletrônico de controle do acionamento da tomada de força que objetiva garantir que o caminhão não passe de 10 km/h com a tomada de força ligada.

Art. 4º O veículo do tipo carroceria basculante deverá possuir sistema hidráulico que utilize o sistema de segurança Tipo A, que é composto pelos dispositivos de segurança primário e secundário, ou o Tipo B, composto pelos dispositivos de segurança primário e terciário.

Art. 5º Os veículos do tipo carroceria basculante deverão possuir fixados no para-brisa os avisos de alerta e segurança sobre a operação dos dispositivos.

Parágrafo único. A apresentação do Certificado de Segurança Veicular (CSV) será exigida anualmente para o licenciamento destes veículos.

Art. 6º Cabe ao implementador fornecer o manual de operação do sistema de basculamento e a descrição do sistema de segurança juntamente com o implemento, sendo obrigatória, pelo menos, a utilização do Tipo A.

Art. 7º O Departamento Nacional de Trânsito (DENATRAN) poderá, a qualquer tempo, solicitar ao implementador ou ao instalador do conjunto hidráulico a apresentação dos resultados de ensaios que comprovem o atendimento das exigências estabelecidas nesta Resolução.

Art. 8º Os caminhões e implementos nacionais e importados do tipo carroceria basculante, a partir de 1º de janeiro de 2017, somente poderão transitar nas vias terrestres abertas a circulação se atenderem aos requisitos desta Resolução.

Parágrafo único. Faculta-se a adoção desta Resolução a partir da data de sua publicação.

Art. 9º A não observância dos preceitos desta Resolução sujeita o infrator às penalidades previstas nos incisos IX ou X do artigo 230 do Código de Trânsito Brasileiro.

Art. 10. Esta Resolução entrará em vigor na data da sua publicação.

Alberto Angerami
Presidente

Guilherme Moraes Rego
Ministério da Justiça

Alexandre Euzébio de Moraes
Ministério dos Transportes

Himário Brandão Trinas
Ministério da Defesa

José Maria Rodrigues de Souza
Ministério da Educação