



CENTRO PAULA SOUZA

ETEC "PROFª. ANNA DE OLIVEIRA FERRAZ"

Técnico em Mecatrônica

RICHARD EVANDER MARQUES

BASTÃO GUIA AUTOMATIZADO PARA CEGOS

Araraquara
2017

RICHARD EVANDER MARQUES

BASTÃO GUIA AUTOMATIZADO PARA CEGOS

Trabalho apresentado a FETEPS 2017 da ETEC "Prof.^a Anna de Oliveira Ferraz", do Centro Estadual de Educação Tecnológica Paula Souza, do curso Técnico em Mecatrônica sob a orientação do Professor Reinaldo Haddad.

Araraquara
2017

DEDICATÓRIA

Dedico primeiramente a meu pai, pelo incentivo a sempre estudar e me manter atualizado e por ser minha base, meu porto seguro.

Dedico a minha noiva, por estar sempre ao meu lado e por respeitar e apoiar sempre minhas decisões.

AGRADECIMENTOS

Agradeço a minha família, em especial a meu pai e minha noiva que sempre apoiaram e incentivaram meus estudos.

Agradeço a Etec Prof^a Anna de Oliveira Ferraz pela oportunidade de aprimorar meus estudos e me tornar um profissional cada dia mais capacitado.

Agradeço aos professores que ao longo do curso me agregaram novos conhecimentos, em especial a Prof^a MS. Karla Cristina Vicentine de Araujo aos Professores Reinaldo Haddad e Cley Eduardo da Rocha. Meus sinceros agradecimentos.

RESUMO

MARQUES, Richard Evander: Bastão Guia Automatizado Para Cegos. Araraquara: 2017.

A presente monografia apresenta o protótipo de um bastão guia automatizado para cegos, com o objetivo de facilitar a locomoção de deficientes visuais, possibilitando maior segurança e uma melhoria na qualidade de vida dessas pessoas. A metodologia empregada baseou-se na utilização da plataforma Arduino que se trata de uma prototipagem eletrônica open-source com hardware e software flexíveis e fáceis de usar. Utilizando a referida plataforma, sensores e atuadores, este bastão guia permite a leitura do ambiente reconhecendo obstáculos e transformando-a em um sinal de fácil identificação e interpretação do portador, possibilitando uma rápida tomada de decisão para que este desvie sem que haja riscos de acidentes. Em paralelo com a eficiência do projeto esperamos obter um produto de qualidade e baixo custo, para que se torne viável, com chances reais de ser produzido em escala industrial e que venha a ser acessível a todos que necessitem.

Palavras-chave: Bastão guia. Facilitar a locomoção. Segurança. Qualidade de vida. Baixo custo.

ABSTRACT

MARQUES, Richard Evander - Automated Guide For the Blind. Araraquara: 2017.

This monograph presents the prototype of an automated guide stick for the blind, with the objective of facilitating the locomotion of the visually impaired, enabling greater security and an improvement in the quality of life of these people. The methodology used was based on the use of the Arduino platform which is an open-source electronic prototyping with flexible and easy-to-use hardware and software. Using this platform, sensors and actuators, this guide rod allows the reading of the environment, recognizing obstacles and transforming it into a signal of easy identification and interpretation of the bearer, making possible a quick decision to deviate without the risk of accidents. In parallel with the efficiency of this project, we expect to obtain a low cost and quality product , so that it becomes usable , with real chances of being produced on an industrial scale and that will be available to all who need it.

Keywords: Guide stick. Facilitate the locomotion. Safety. Quality of life. Low cost.

Lista de Figuras

FIGURA 3.3.3 – PLACA DO ARDUINO UNO.....	11
FIGURA 3.3.5 – TELA INICIAL DO COMPILADOR.....	13
FIGURA 3.3.6(A) - SELECIONANDO A PLACA NO COMPILADOR	15
FIGURA 3.3.6 (B) - SELECIONANDO A PLACA NO COMPILADOR	15
FIGURA 3.3.7 (A) – CARREGANDO O BLINK	16
FIGURA 3.3.7 (B) – PROGRAMAÇÃO ESCRITA DO BLINK.....	17
FIGURA 3.3.7 (C) – TÉRMINO DE VERIFICAÇÃO	17
FIGURA 3.3.8 – REFERENCIA NO MENU DE AJUDA	19
FIGURA 3.3.10 (A) - ARDUINO PRO MINI FRENTE	19
FIGURA 3.3.10 (B) - ARDUINO PRO MINI VERSO.....	19
FIGURA 3.3.12 – EXEMPLO DO FUNCIONAMENTO DE UM SENSOR.....	21
FIGURA 3.3.13 (A) – SENSOR HC-SR04 LIGADO AO ARDUINO	22
FIGURA 3.3.13 (B) – HC-SR04 EM TRABALHO.....	22
FIGURA 3.3.13 (C) – HC-SR04 COMPORTAMENTO	22
FIGURA 3.3.14 – VARIAÇÃO ANALÓGICA	23
FIGURA 3.3.15 – VARIAÇÃO DIGITAL	23
FIGURA 3.3.17 – CONVERSÃO DE SINAL A/D E D/A	24
FIGURA 4.2 (A) – COMPONENTES DA ENERGIZAÇÃO.....	27
FIGURA 4.2(B) – O CABO DO BASTÃO.....	28
FIGURA 4.2 (C) – SIMULANDO A PROGRAMAÇÃO	28
FIGURA 4.2 (D) – MOTOR DE VIBRAÇÃO	29
FIGURA 4.2 (E) – EXPONDO A ELETRÔNICA INTERNA	29
FIGURA 4.2 (F) – ESFERA DE CONTATO COM O SOLO	30
FIGURA 4.2 (G) – ESTRUTURA DO PROJETO.....	30
FIGURA 4.2 (H) – SENSOR HC-SR04.....	31
FIGURA 4.2 (I) – REVESTIMENTO DA ESTRUTURA.....	31
FIGURA 4.3 – BASTÃO GUIA FINALIZADO	32

SUMÁRIO

OBJETIVO	8
DESENVOLVIMENTO.....	9
3.1 LEVANTAMENTO DEMOGRÁFICO	9
3.2 PRIMEIRO PROJETO DE UMA BENGALA ULTRASSÔNICA	9
3.3 ATUADORES, CONTROLADORES E SENSORES	10
3.3.1 - Conhecendo o arduino.....	10
3.3.2 Plataforma de desenvolvimento Arduino	10
3.3.3 Hardware do Arduino	11
3.3.4 Software do Arduino	12
3.3.5 IDE do Arduino.....	13
3.3.6 Serial monitor	14
3.3.7 - "Hello World" – Blink.....	16
3.3.8 - Analisando o Código	18
3.3.9 - Atuadores.....	19
3.3.10 - Arduino Pro Mini.....	19
3.3.11 - Visão geral.....	19
3.3.12 -Sensor Ultrassônico.....	21
3.3.13 - Sensor HC-SR04.....	21
3.3.14 - Sensores analógicos	22
3.3.15 - Sensores Digitais	23
3.3.16 - Transdutores.....	24
3.3.17 - Conversores A/D e D/A	24
4-O PROJETO.....	25
4.1 – A PROGRAMAÇÃO.....	25
4.2 COMPONENTES	27
4.3- IMAGEM DO BASTÃO GUIA FINALIZADO.....	32
5-CONCLUSÃO	33
6-CRONOGRAMA	34
REFERÊNCIASBIBLIOGRAFICAS	35
ANEXO A – TERMODE AUTORIZAÇÃO DE DIVULGAÇÃO	36
ANEXO B –DECLARAÇÃODE AUTENTICIDADE	37

INTRODUÇÃO

Ao longo da evolução tendemos a buscar meios de tornar fáceis nossas atividades, buscar um padrão e qualidade de vida melhor, mais conforto, e para pessoas deficientes tem-se essa busca como uma obrigação e dever.

O projeto “bastão guia automatizado para cegos” busca dar melhor qualidade de vida para um deficiente visual, visando segurança em sua locomoção.

E se esse bastão fosse capaz de prever um obstáculo antes de tocá-lo? Isso certamente traria mais confiança e segurança ao seu portador certo? Esse é o foco do projeto bastão guia automatizado para cegos.

Além do objetivo de produzir um projeto com qualidade, buscamos também construí-lo com um baixo valor final para garantir que todos que necessitem tenham meios de adquirir o produto.

OBJETIVO

Desenvolver e executar o projeto de um bastão que possa reconhecer objetos que se aproximem utilizando um sensor ultrassônico. O bastão emitirá constantemente sinais ultrassônicos, quando se aproximar de um objeto esses sinais irão bater e voltar ao emissor que também funciona como um receptor, ao reconhecer o retorno desse sinal o bastão o converterá para um sinal vibratório e sonoro, tornando fácil a interpretação do usuário permitindo assim identificar quando algo se aproxima antes mesmo de tocá-lo. A distância dos obstáculos será diferenciada por intensidade e duração de vibração tornando assim a interpretação do usuário simples e rápida.

Visando construir um projeto de qualidade e baixo custo, serão utilizados materiais que ofereçam resistência e durabilidade com o menor custo possível para que se torne um produto viável, com chances reais de ser produzido em escala industrial.

DESENVOLVIMENTO

3.1 Levantamento Demográfico

Dados do IBGE revelam que 6,2% da população brasileira têm algum tipo de deficiência. A Pesquisa Nacional de Saúde (PNS) considerou quatro tipos de deficiências: auditiva, visual, física e intelectual. O levantamento foi divulgado pelo Instituto Brasileiro de Geografia e Estatística (IBGE) e feito em parceria com o Ministério da Saúde.

Dentre os tipos de deficiência pesquisados, a visual é a mais representativa e atinge 3,6% dos brasileiros, sendo mais comum entre as pessoas com mais de 60 anos (11,5%). O grau intenso ou muito intenso da limitação impossibilita 16% dos deficientes visuais de realizarem atividades habituais como ir à escola, trabalhar e brincar. (Villela, 2015)

Com uma quantidade tão expressiva de deficientes visuais apenas no Brasil, ter como um projeto de conclusão de curso um equipamento de fácil acesso a todos que necessitem a um baixo custo é uma meta e motivação.

3.2 Primeiro Projeto De Uma Bengala Ultrassônica

A Universidade Técnica Estatal de São Petersburgo elaborou e produziu uma bengala ultrassônica para deficientes visuais. A nova bengala permite aos cegos contornar não só buracos e bordas de passeio, mas também os obstáculos que não tocam na terra.

A bengala foi inventada pelo cientista e colaborador da Universidade Técnica Estatal, Serguei Ierchov.

A bengala tem um pequeno radar no punho. O aparelho emite um som agudo demais para ser ouvido por humanos e capta as reflexões dessas ondas. O sinal que significa que um obstáculo está perto pode ser ouvido através de auscultadores. O ultra-som é o meio mais seguro para determinar a distância porque funciona até quando chove e neva. Pode determinar a distância com exatidão de 15 centímetros.

Infelizmente agora todos os deficientes visuais não podem receber esta bengala porque ainda produz-se uma pequena quantidade de bengalas novas. A fábrica da Universidade

Técnica Estatal consegue produzir só 50-100 bengalas por mês o que é obviamente insuficiente. (PRAVDA.Ru, 03/11/2006)

Embora a ideia de uma bengala ultrassônica tenha nascido na Rússia em 2006, e alguns outros estudantes de diversas partes do mundo tenham tentado aperfeiçoar essa esplêndida idéia atualmente ainda não tem esse produto em escala industrial e comercial, talvez por falta de incentivos de patrocinadores, ou por ainda estar com um valor de produção elevado. Assim o principal objetivo do projeto além de visar à qualidade do produto é construí-lo com o menor custo possível.

3.3 Atuadores, Controladores E Sensores

O BASTÃO GUIA AUTOMATIZADO PARA CEGOS contará com sensores, controladores e atuadores que serão responsáveis por detectar obstáculos e enviar um sinal ao portador em forma de vibração e sonoro.

Componentes do projeto:

3.3.1 - Conhecendo o arduino

“O Arduino é uma plataforma de prototipagem eletrônica open-source que se baseia em hardware e software flexíveis, livres e fáceis de usar. É destinado a artistas, designers, hobbistas e qualquer pessoa interessada em criar objetos ou ambientes interativos.

Basta uma programação adequada o arduino pode sentir o estado do ambiente que o cerca por meio da recepção de sinais de sensores e pode interagir com os seus arredores, controlando luzes, motores e outros atuadores. O microcontrolador na placa é programado com a linguagem de programação Arduino, baseada na linguagem *Wiring*, e o ambiente de desenvolvimento Arduino, baseado no ambiente *Processing*. Os projetos desenvolvidos com o Arduino podem ser autônomos ou podem comunicar-se com um computador para a realização da tarefa, com uso de software específico (ex: *Flash*, *Processing*, *MaxMSP*).”

3.3.2 Plataforma de desenvolvimento Arduino

O Arduino é formado por dois componentes principais: Hardware e Software que são livres, qualquer pessoa pode alterar a qualquer momento.

O hardware é composto por uma placa de prototipagem na qual são construídos os projetos.

O software é uma IDE, que é executado em um computador onde é feita a programação, conhecida como sketch, na qual será feita upload para a placa de prototipagem Arduino, através de uma comunicação serial. O sketch feito pelo projetista dirá à placa o que deve ser executado durante o seu funcionamento.

3.3.3 Hardware do Arduino

Existem diversas placas oficiais de Arduino e muitas outras não oficiais, por se tratar de um software e hardware livre, podem ser amplamente estudados e modificados. Será abordada a placa do Arduino Uno conforme a figura 3.3.3.



Placa Arduino UNO

FIGURA 3.3.3 – PLACA DO ARDUINO UNO

Fonte: www.circuitar.com.br/tutoriais

Conforme visto na imagem acima a placa Arduino UNO possui diversos conectores que servem para interface com o mundo externo. Veja como estão organizados os pinos na placa:

14 pinos de entra e saída digital (pinos 0-13):

Esses pinos podem ser utilizados como entradas ou saídas digitais de acordo com a necessidade do projeto e conforme foi definido no sketch criado na IDE.

6 pinos de entradas analógicas (pinos A0 - A5):

Esses pinos são dedicados a receber valores analógicos, por exemplo, a tensão de um sensor. O valor a ser lido deve estar na faixa de 0 a 5 V onde serão convertidos para valores entre 0 e 1023.

6 pinos de saídas analógicas (pinos 3, 5, 6, 9, 10 e 11):

São pinos digitais que podem ser programados para ser utilizados como saídas analógicas, utilizando modulação PWM.

A alimentação da placa pode ser feita a partir da porta USB do computador ou através de um adaptador AC. Para o adaptador AC recomenda-se uma tensão de 9 a 12 volts.

3.3.4 Software do Arduino

O software para programação do Arduino é uma IDE que permite a criação de sketches para a placa Arduino (no menu do programa há opção para muitas outras variações de arduino). A linguagem de programação é modelada a partir da linguagem Wiring. Quando pressionado o botão upload da IDE, o código escrito é traduzido para a linguagem C e é transmitido para o compilador avr-gcc, que realiza a tradução dos comandos para uma linguagem que pode ser compreendida pelo microcontrolador.

A IDE apresenta um alto grau de abstração, possibilitando o uso de um microcontrolador sem que o usuário conheça o mesmo, nem como deve ser usado os registradores internos de trabalho.

A IDE do Arduino possui uma linguagem própria baseada na linguagem C e C++.

O Ciclo de programação do Arduino pode ser dividido da seguinte maneira:

Conexão da placa a uma porta USB do computador;

Desenvolvimento de um sketch com comandos para a placa;

Upload do sketch para a placa, utilizando a comunicação USB.

Aguardar a reinicialização, após ocorrerá à execução do sketch criado.

A partir do momento que foi feito o upload o Arduino não precisa mais do computador: o Arduino executará o sketch criado, desde que seja ligado a uma fonte de energia.

3.3.5 IDE do Arduino

A IDE (figura 3.3.5 a seguir) pode ser baixada gratuitamente no site do Arduino, onde pode ser escolhida a melhor opção de download conforme plataforma utilizada.

Quando se abre o IDE do Arduino, será exibido algo semelhante à figura 3.3.5 abaixo:

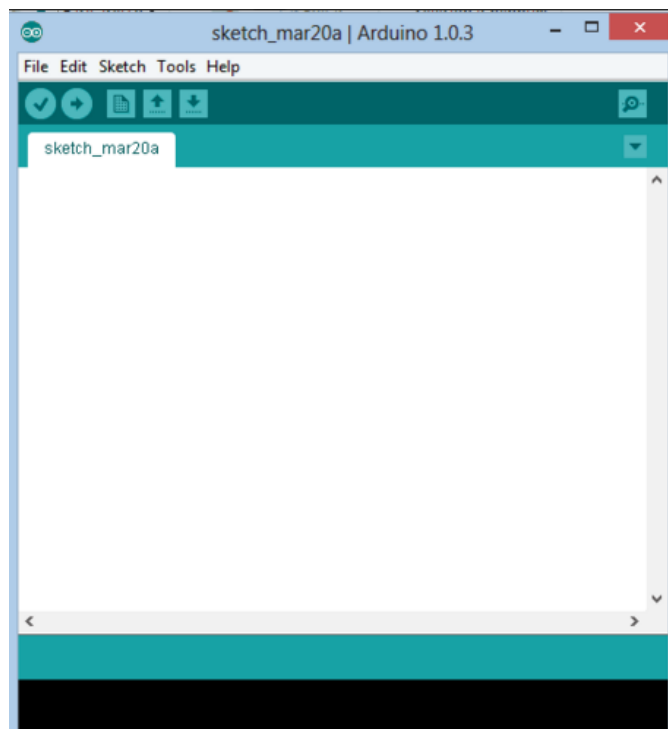


FIGURA 3.3.5 – TELA INICIAL DO COMPILADOR

Fonte: www.circuitar.com.br/tutoriais

O IDE é dividido em três partes: O Toolbar no topo, o código ou a *Sketch Window* no centro, e a janela de mensagens na base, conforme é exibido na figura 3.3.5 anterior.

No *Toolbar* há uma guia, ou um conjunto de guias, com o nome do *sketch*. Ao lado direito há um botão que habilita o serial monitor. No topo há uma barra de menus, com os itens *File*, *Edit*, *Sketch*, *Tools* e *Help*. Os botões no *Toolbar* fornecem acesso rápido às funções mais utilizadas dentro desses menus.

Abaixo são identificados os ícones de atalho da IDE:

- ***Verify***

Verifica se existe erro no código digitado.

- ***Upload***

Compila o código e grava na placa Arduino se corretamente conectada;

- ***New***

Cria um novo sketch em branco.

- ***Open***

Abre um *sketch*, presente no *sketchbook*.

- ***Save***

Salva o sketch ativo

3.3.6 Serial monitor

Os demais comandos presentes na barra de menus podem ser consultados através do menu<*help*><*Environment*> e duvidas na guia *Help*.

Após a conexão do Arduino ao computador, é atribuída a placa uma COM. A primeira vez que o programa Arduino for executado deve-se selecionar o modelo de placa utilizado, no projeto escolher-se-á o Arduino Uno, conforme figura abaixo:

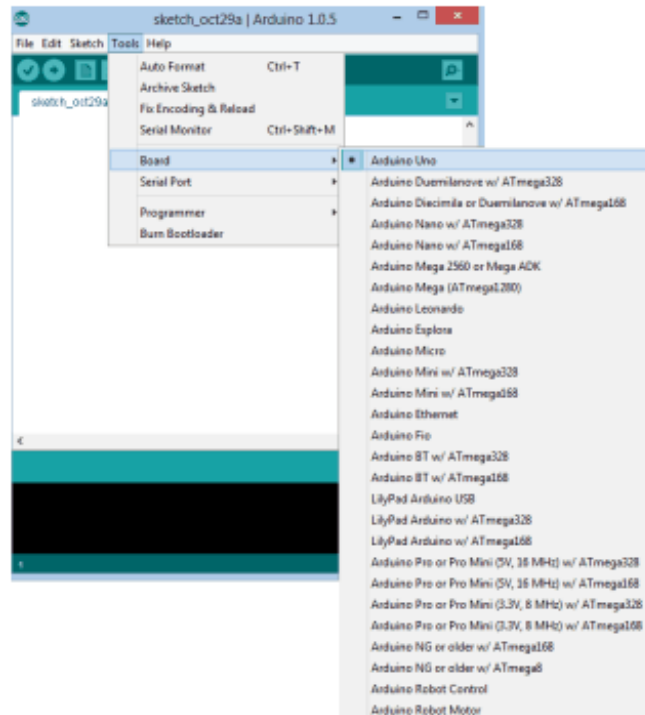


FIGURA 3.3.6(A) - SELECIONANDO A PLACA NO COMPILADOR

Fonte: www.circuitar.com.br/tutoriais

Após a definição do modelo, deve-se selecionar em qual COM a placa foi atribuída conforme a figura 3.3.6 (B) a seguir:

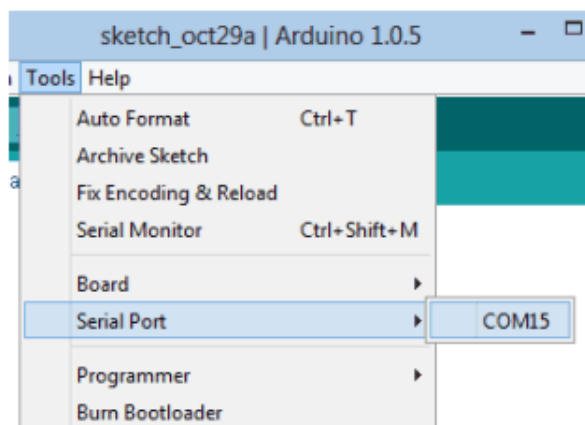


FIGURA 3.3.6 (B) - SELECIONANDO A PLACA NO COMPILADOR

Fonte: www.circuitar.com.br/tutoriais

Após estas configurações o ambiente está preparado para uso e pode-se testar qualquer um dos exemplos que acompanham a IDE ou até mesmo com um novo sketch.

3.3.7 - "Hello World" – Blink

O exemplo mais simples para iniciar a programação do Arduino, que pode ser considerado como o conhecido “Hello World” (Olá mundo) das linguagens de programação, consiste em acionar um LED através de uma saída digital (LED esse que já está presente em quase todas as placas de arduino).

A placa Arduino Uno já possui um Led ligado ao pino digital 13 que pode ser utilizado para o teste conforme a figura 3.3.7, e na IDE pode-se carregar o exemplo *Blink*:

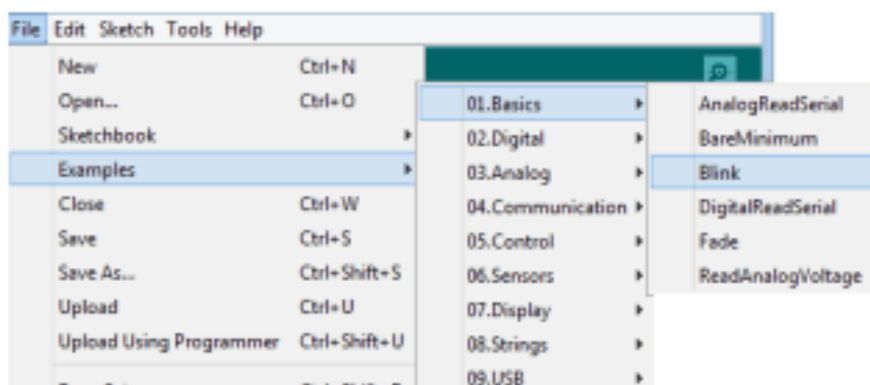


FIGURA 3.3.7 (A) – CARREGANDO O BLINK

Fonte: www.circuitar.com.br/tutoriais

```

1  /*
2  Blink
3  Turns on an LED on for one second, then off for one second, repeatedly.
4
5  This example code is in the public domain.
6  */
7
8  // Pin 13 has an LED connected on most Arduino boards.
9  // give it a name:
10 int led = 13;
11
12 // the setup routine runs once when you press reset:
13 void setup() {
14 // initialize the digital pin as an output.
15 pinMode(led, OUTPUT);
16 }
17
18 // the loop routine runs over and over again forever:
19 void loop() {
20 digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
21 delay(1000); // wait for a second
22 digitalWrite(led, LOW); // turn the LED off by making the voltage LOW
23 delay(1000); // wait for a second
24 }

```

FIGURA 3.3.7 (B) – PROGRAMAÇÃO ESCRITA DO BLINK

Fonte: www.circuitar.com.br/tutoriais

Para verificar se o código está correto deve-se clicar no ícone **verify**, após a compilação é exibida uma mensagem de status da operação e caso esteja tudo certo será exibida a quantidade de bytes gerados pelo programa no canto esquerdo inferior da IDE conforme figura 3.3.3 (C) a seguir:

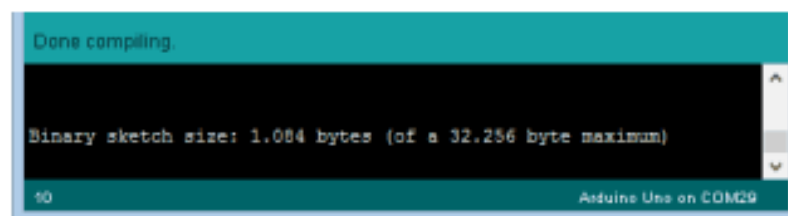


FIGURA 3.3.7 (C) – TÉRMINO DE VERIFICAÇÃO

Fonte: www.circuitar.com.br/tutoriais

Para gravar o código na memória flash do microcontrolador é necessário clicar no ícone **Upload**, será transferido o código para a placa e após alguns segundos o LED ligado ao pino 13 começará a piscar em intervalos de 1 segundo (executando a programação *Blink*).

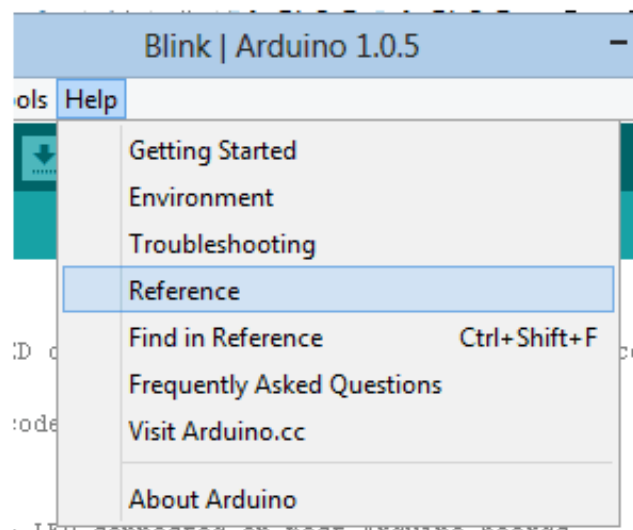
3.3.8 - Analisando o Código

O código do exemplo Blink é relativamente simples, porém apresenta a estrutura básica de um programa desenvolvido na IDE Arduino. Inicialmente nota-se que existem duas funções obrigatórias em um programa Arduino, *setup()* e *loop()*.

A função *setup()* é executada na inicialização do programa e é responsável pelas configurações iniciais do microcontrolador, tal como definição dos pinos de I/O, inicialização da comunicação serial, entre outras.

A função *loop()* será onde ocorrerá o laço infinito da programação, ou seja, onde será inserido o código que será executado continuamente pelo microcontrolador.

Dentro do *loop* principal está o código que fará o led ligado pino 13 piscar em intervalos de 1segundo.A função *digitalWrite(led, HIGH);* coloca o pino em nível lógico 1, ligando o led.A função *delay(1000);* aguarda o tempo de 1000 ms, ou seja, 1 segundo para que possa ser executada a próxima instrução.A função *digital Write (led, LOW);* coloca o pino em nível lógico 0, desligando o led. E novamente é esperado 1 segundo com a função *delay();*O *loop* é repetido infinitamente enquanto a placa estiver ligada. A referência da linguagem Arduino pode ser acessada através do menu<*help*> como mostrado na figura 3.3.8 a seguir:



Testar outros exemplos básicos é um bom começo para aprender mais sobre a plataforma Arduino e aprender a desenvolver os próprios programas. (SOUZA, 06/11/2013)

3.3.9 - Atuadores

São dispositivos que modificam uma variável controlada. Recebem um sinal proveniente do controlador e agem no sistema controlado. Geralmente trabalham com potência elevada.

3.3.10 - Arduino Pro Mini

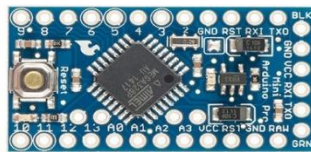


FIGURA 3.3.10 (A) - ARDUINO PRO
MINI FRENTE

Fonte: BR-ARDUINO.ORG

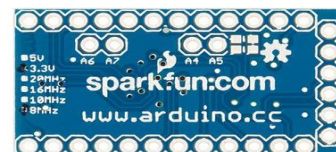


FIGURA 3.3.10 (B) - ARDUINO PRO MINI
VERSO

Fonte: BR-ARDUINO.ORG

3.3.11 - Visão geral

O arduino pro mini foi o hardware selecionado para o projeto do bastão guia por ser uma placa compacta e versátil, oferecendo todas as vantagens que qualquer outro arduino possa oferecer, apenas com um numero de portas mais limitado, porem mais que o necessário para executar o projeto. O Arduino Pro Mini é uma placa de microcontrolador baseado no ATmega328 (datasheet). Dispõe de 14 pinos digitais de entrada / saída (dos quais 6 podem ser usados como saídas PWM), 6 entradas analógicas, um ressonador on-board, um botão de reset e buracos para a montagem de *pin headers*. Um cabeçalho de seis pinos pode ser conectado a um cabo FTDI ou placa fuga *Sparkfun* para fornecer energia USB e

comunicação para o conselho. O Arduino Pro Mini é destinado para instalação semi-permanente em objetos ou exposições. A placa vem sem cabeçalhos de pré-montado, permitindo o uso de vários tipos de conectores ou de solda direta de fios. A pinagem é compatível com o Arduino Mini. Existem duas versões do Pro Mini. Corre-se em 3.3V e 8 MHz , a outra em 5V e 16 MHz .O Arduino Pro Mini foi projetado e fabricado pela *SparkFun Electronics*.

Resumo

Microcontrolador	ATmega328
Tensão operacional	3.3V ou 5V (dependendo do modelo)
Tensão de entrada	3.35 -12 V (modelo 3.3V) ou 5 - V (modelo 5V) 12
Digital pinos I / O	14 (dos quais 6 fornecer saída PWM)
Pinos de entrada analógica	6
Corrente DC por I / O Pin	40 mA
Memória flash	32 kB (dos quais 0,5 kB usado pelo bootloader)
SRAM	2 kB
EEPROM	1 kB
Velocidade do relógio	8 MHz (modelo 3.3V) ou 16 MHz (5V modelo)

(ARDUINO) FONTE: FABRICANTE ATMEGA328

3.3.12 -Sensor Ultrassônico

Sensores são dispositivos sensíveis a alguma forma de energia ambiente que pode ser luminosa, térmica, cinética, relacionando informações sobre uma grandeza que precisa ser medida, como temperatura, pressão etc., conforme indica a figura 3.3.12 a seguir:

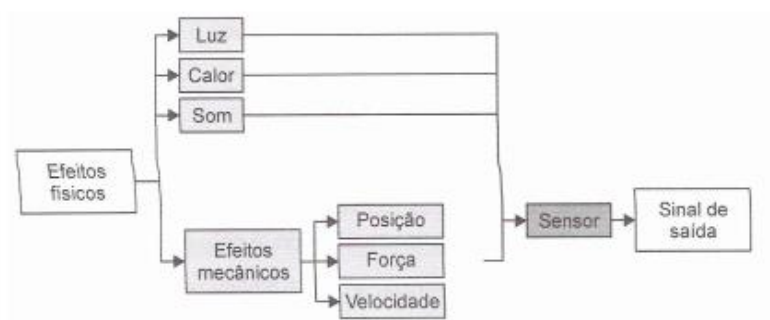


FIGURA 3.3.12 – EXEMPLO DO FUNCIONAMENTO DE UM SENSOR. FONTE: PRÓPRIO AUTOR

O sensor que efetua a medição da grandeza nem sempre tem as características elétricas necessárias para ser utilizado diretamente em um sistema de controle. Para isso faz se necessário a um tratamento deste sinal, tornando pronto para a utilização.

3.3.13 - Sensor HC-SR04

O Sensor Ultrassônico HC-SR04 (figura 3.3.13 (A)) é um componente muito comum em projetos com Arduino, e permite que você faça leituras de distâncias entre 2 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, acionar alarmes, etc.

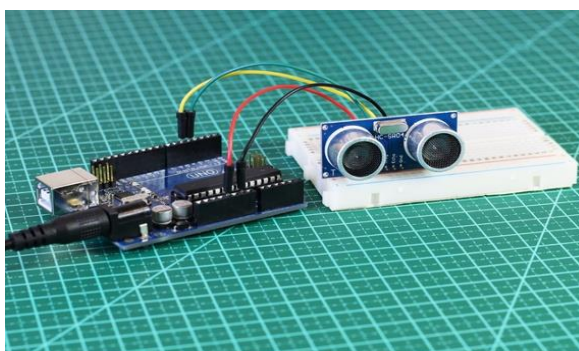


Figura 3.3.13 (A)– Sensor HC-SR04 ligado ao arduino. *FONTE: PRÓPRIO AUTOR*

O funcionamento do HC-SR04 (datasheet) se baseia no envio de sinais ultrassônicos pelo sensor, que aguarda o retorno (echo) do sinal, e com base no tempo entre envio e retorno, calcula a distância entre o sensor e o objeto detectado conforme a figura 3.3.13 (B) a seguir.

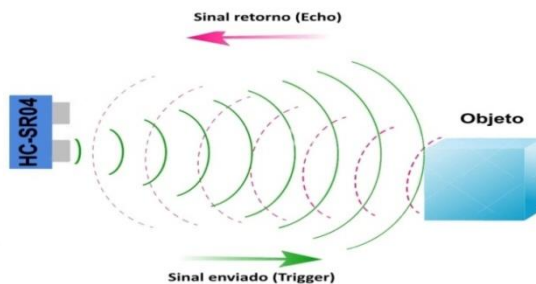


Figura 3.3.13 (B) – HC-SR04 em trabalho. *Fonte: Apostila sensores*

Primeiramente é enviado um pulso de 10µs, indicando o início da transmissão de dados. Depois disso, são enviados 8 pulsos de 40 KHz e o sensor então aguarda o retorno (em nível alto/high), para determinar a distância entre o sensor e o objeto, utilizando a equação $Distância = (Tempo\ echo\ em\ nível\ alto * velocidade\ do\ som) / 2$ como está representado na figura 3.3.13 (C) a seguir:

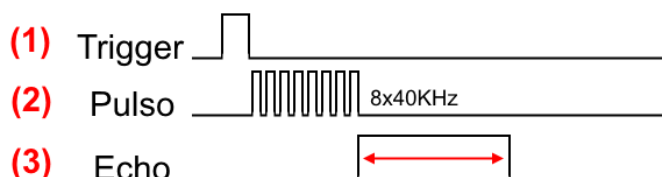


Figura 3.3.13 (C) – HC-SR04 comportamento. *FONTE: PRÓPRIO AUTOR*

Para ligação do sensor ao microcontrolador, são utilizados 4 pinos: **Vcc**, **Trigger**, **ECHO** e **GND**. (THOMSEN, 2013)

3.3.14 - Sensores analógicos

Esse tipo de sensor pode assumir qualquer valor no seu sinal de saída ao longo do tempo, desde que esteja dentro da sua faixa de operação.

Algumas grandezas físicas que podem assumir qualquer valor ao longo do tempo são: pressão, temperatura, velocidade, umidade, vazão, força, ângulo, distância, torque, luminosidade. Essas variáveis são mensuradas por elementos sensíveis com circuitos eletrônicos não digitais. A figura 3.3.14 a seguir ilustra a variação de uma grandeza física (temperatura) de uma forma analógica.

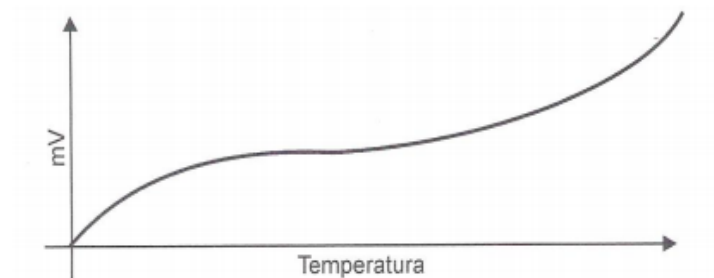


Figura 3.3.14 – Variação analógica. *FONTE: PRÓPRIO AUTOR*

3.3.15 - Sensores Digitais

Esse tipo de sensor pode assumir apenas dois valores no seu sinal de saída ao longo do tempo, que podem ser interpretados como zero (0) ou um (1). Não existem naturalmente grandezas físicas que assumem esses valores, mas eles são assim mostrados ao sistema de controle após serem convertidos pelo circuito eletrônico do transdutor. É utilizado, por exemplo, em detecção de passagem de objetos, encoders na determinação de distância ou velocidade etc. A figura 3.3.15 a seguir ilustra a variação da posição de um objeto lida por um encoder incremental.

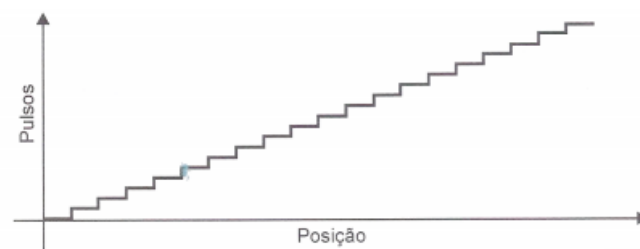


Figura 3.3.15 – Variação digital. *FONTE: PRÓPRIO AUTOR*

3.3.16 - Transdutores

Transdutor é a denominação que se aplica a um dispositivo completo, que contém o sensor, usado para transformar uma grandeza física em qualquer outra que pode ser utilizada nos dispositivos de controle. Um transdutor pode ser considerado uma interface ao formar de energia do ambiente e o circuito de controle ou, eventualmente, entre o controle e o atuador.

Os transdutores transformam uma grandeza física (temperatura, pressão, etc.) em um sinal de tensão ou corrente que pode ser facilmente interpretado por um sistema de controle.

3.3.17 - Conversores A/D e D/A

É possível converter um sinal analógico em digital ou vice-versa por meio de circuitos integrados conversores, mais vale ressaltar que após a conversão realizada, parte do sinal é perdida e podem existir pequenas distorções na grandeza realmente medida, conforme indica a figura 3.3.17 a seguir dando exemplo de digitalização e linearização de sinais.

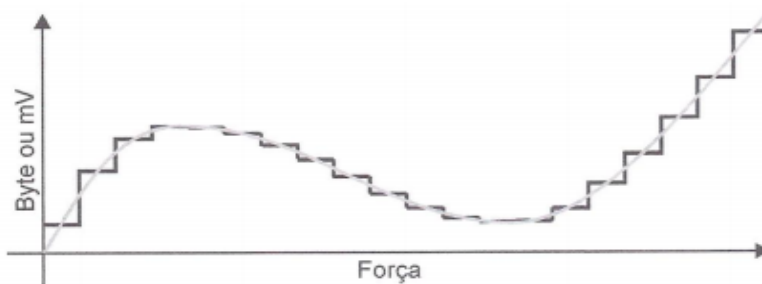


Figura 3.3.17 – Conversão de sinal A/D e D/A. *FONTE: PRÓPRIO AUTOR*

4-O PROJETO

4.1 – A Programação

```

/*****
* Projeto do TCC: BASTÃO GUIA AUTOMATIZADO PARA CEGOS
* Richard Marques - Tec. em mecatrônica
* 2016
*****/

#define trig    9//Pino 9 do Arduino será a saída de trigger
#define echo_a8 //Pino 8 do Arduino será a entrada de echo_a
#define echo_b7 //Pino 5 do arduino será a entrada de echo_b
#define buzzer6 //pino 6 aciona alarme sonoro do buzzer
#define vibra5 //Pino 7 aciona motor de vibração

void trigPulse(); //Função que gera o pulso de trigger
float pulse;      //Variável que armazena o tempo de duração do echo
float distancia_a; //Variável que armazena o valor da distância em centímetros
float distancia_b;

void setup()      //definindo o setup
{
  pinMode(trig, OUTPUT);    //Pino de trigger será saída digital
  pinMode(echo_a, INPUT);   //Pino de echo_a será entrada digital
  pinMode(echo_b, INPUT);   //Pino de echo_b será entrada digital
  pinMode(vibra, OUTPUT);   //Pino de vibra sera saída digital
  pinMode(buzzer, OUTPUT);  //Pino de buzzer será de saída digital
  digitalWrite(trig, LOW);  //Saída trigger inicia em nível baixo
  digitalWrite(vibra, LOW); //Saída de vibra inicia em nível baixo
  digitalWrite(buzzer, LOW); //Saída de buzzer inicia em nível baixo
  Serial.begin(9600);       //Inicia comunicação serial
} //fim do setup

void loop()
  //Leitura do sensor 1
  trigPulse();              //Aciona o trigger do módulo ultrassônico
  pulse = pulseIn(echo_a, HIGH); //Mede o tempo em que o pino de echo fica em nível alto
  distancia_a = pulse/58.82;    //Valor da distância em centímetros
  Serial.println(distancia_a); //Imprime o valor na serial

```

```

delay(300);           //Taxa de atualização
//Leitura do sensor 2
trigPulse();          //Aciona o trigger do módulo ultrassônico
pulse = pulseIn(echo_b, HIGH);    //Mede o tempo em que o pino de echo fica em nível alto
distancia_b = pulse/58.82;        //Valor da distância em centímetros
Serial.println(distancia_b);      //Imprime o valor na serial
delay(300);           //Taxa de atualização
// 1º laço de distancia
//se distancia entre intervalo definido
if((distancia_a > 149 && distancia_a < 199) || (distancia_b > 150 && distancia_b < 200))
//if(distancia_a > 150 && distancia_a < 200)
{
digitalWrite(vibra, HIGH); //vibra vai a nivel alto
delay(500);                //tempo em nivel alto durante loop
digitalWrite (vibra, LOW); //vibra vai a nivel baixo
delay(100);                //tempo em nivel baixo durante loop
}
// 2º laço de distancia
//se distancia entre intervalo definido
if((distancia_a > 19 && distancia_a < 148) || (distancia_b > 20 && distancia_b < 149))
//if(distancia_a > 20 && distancia_a < 149)
{
digitalWrite(vibra, HIGH); //vibra vai a nivel alto
digitalWrite(buzzer, HIGH); //buzzer vai a nivel alto
delay(200);                //tempo em nivel alto durante loop
digitalWrite (buzzer, LOW); //buzzer vai a nivel baixo
digitalWrite (vibra, LOW); //vibra vai a nivel baixo
delay(200);                //tempo em nivel baixo durante o loop
}
} //Fim do loop
//função auxiliar do trig
void trigPulse()
{
digitalWrite(trig, HIGH); //Pulso de trigger em nível alto
delayMicroseconds(10);    //duração de 10 micro segundos
digitalWrite(trig, LOW);  //Pulso de trigge em nível baixo
}

```

```
delayMicroseconds(5);    //duração de 5 micro segundos  
} //fim da função auxiliar trig  
// Richard Marques
```

FONTE: PRÓPRIO AUTOR

4.2 Componentes

O projeto contará com um sistema de alimentação por bateria de 9Volts(V), conforme a figura 4.2 (A) e que será reduzido e estabilizado para um padrão de 5V.

Na figura 4.2 (A) também se encontra o Buzzer responsável pelo aviso sonoro, o terminal de encaixe padrão para uma bateria de 9V, e o botão ON OFF de pulso com trava responsável por ligar e desligar o sistema do bastão guia.



FIGURA 4.2 (A) – COMPONENTES DA ENERGIZAÇÃO

FONTE: PRÓPRIO AUTOR

O cabo do bastão foi desenvolvido reutilizando um frasco vazio de desodorante conforme a figura 4.2 (B). Ele serve como suporte de acoplamento da bateria de 9V e também

do botão de ligar e desligar o sistema do bastão guia. Localizado dentro do cabo também está fixado o motor de vibração(figura 4.3 (D), para maximizar o alerta ao usuário do bastão.



Figura 4.2(B) – O cabo do bastão. *FONTE: PRÓPRIO AUTOR*

Para facilitar a programação, na saída do arduino foi ligado dois leds para que simulassem o motor do vibra que indica um obstáculo a primeira distância programada, e o Buzzer que indica a segunda distância programada (mais próximo). Após a programação finalizada, os leds foram substituídos pelo buzzer (figura 4.2 (A), e pelo motor de vibração(figura 4.2(D)).Na imagem 4.2 (C) a seguir, encontra-se a montagem com leds para testes da programação em bancada.

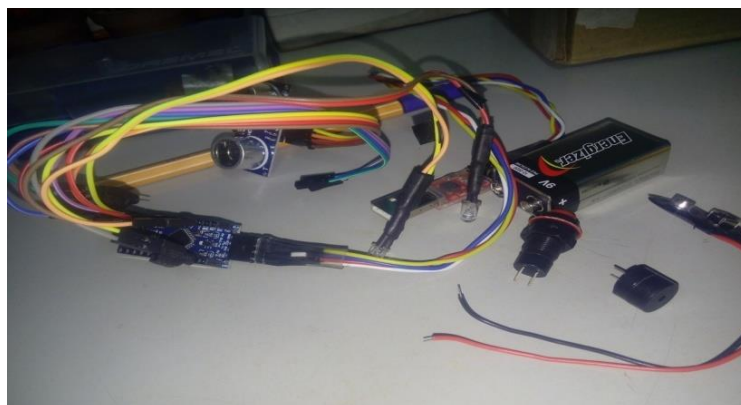


Figura 4.2 (C) – Simulando a programação.*FONTE: PRÓPRIO AUTOR*

Na figura 4.2 (D) a seguir encontra-se o motor de vibração de 5V (o mesmo motor usado em vibracall de celulares) para indicar a primeira distância programada de algum obstáculo.



Figura 4.2 (D) – Motor de vibração. *FONTE: PRÓPRIO AUTOR*

O motor vai fixado dentro do cabo do bastão guia (tubo de desodorante conforme a figura 4.2 (B)) para maximizar a sensação e percepção da vibração quando há um evento.

Na figura 4.2 (E) encontra-se parte da eletrônica que fica oculta internamente no bastão guia, e pode ser retirada facilmente para alguma eventual manutenção ou atualização da programação se necessário.



Figura 4.2 (E) – Expondo a eletrônica interna. *FONTE: PRÓPRIO AUTOR*

Na figura 4.2 (F) encontra-se uma esfera de plástico para o contato do bastão com o solo (pé de cadeira reaproveitado). Por ser de um material mais resistente que os demais componentes, ela absorve o primeiro impacto, evitando danos a outras partes do bastão guia.



Figura 4.2 (F) – Esfera de contato com o solo. *FONTE: PRÓPRIO AUTOR*

O corpo do bastão foi construído com partes reutilizadas de uma vara de pesca, tornando o projeto leve e barato, como nota-se na figura 4.2 (G) a seguir.

Por ser de uma estrutura oca, possibilitou que toda parte de circuito do projeto ficasse alojado em seu interior, apresentando uma melhor estética ao projeto finalizado.



Figura 4.2 (G) – Estrutura do projeto. *FONTE: PRÓPRIO AUTOR*

Por ficar exposta a placa do circuito eletrônico do sensor ultrassônico HC-SR04, foi revestida com fita isolante para que não pegasse unidade e não se danificasse, como encontra-se na figura 4.2 (H):



Figura 4.2 (H) – Sensor HC-SR04. *FONTE: PRÓPRIO AUTOR*

O tubo da vara de pesca foi revestido com fita isolante para reforçar a estrutura e dar uma melhor estética ao projeto como nota-se na figura 4.2 (I) a seguir. O encaixe do pé de cadeira (batedor) foi embuchado na extremidade inferior do tubo como nota-se também na figura 4.2 (I) a seguir:



Figura 4.2 (I) – Revestimento da estrutura. *FONTE: PRÓPRIO AUTOR*

4.3- Imagem Do Bastão Guia Finalizado

Na figura 4.3 a seguir, encontra-se o projeto do **bastão guia automatizado para cegos**, já finalizado.



Figura 4.3 – Bastão guia finalizado. *FONTE: PRÓPRIO AUTOR*

5-CONCLUSÃO

Para um deficiente visual, o simples direito assegurado a todos de ir e vir torna-se uma tarefa complicada e até muitas vezes perigosa por não conseguir prever obstáculos com antecedência utilizando os bastões tradicionais conhecido como bengala longa, objetos ou obstáculos que possam impedir dificultar ou até mesmo tornar perigoso o seu acesso.

Ainda não há nada em escala comercial relacionado ao bastão guia automatizado para cegos, embora muitas pesquisas e projetos modelo já tenham sido executados. O intuito principal do projeto é desenvolver um equipamento com baixo custo para se tornar de fácil acesso a todas as pessoas que venham a necessitar.

6-CRONOGRAMA

PERÍODO 2016 ATIVIDADES	MAR	ABR	MAI	JUN	JUL	AGO	SET	OUT	NOV
Definição do tema	X								
Referencial Teórico			X						
Introdução			X						
Justificativa			X						
Objetivo			X						
Metodologia				X					
Revisão / Correção				X					
Apresentação				X					
Entrega do Projeto				X					
Produção de Dados						X			
Análise e Discussão dos Dados						X	X		
Elaboração do Desenvolvimento						X	X		
Considerações Finais e Conclusão								X	X
Correção									X
Entrega Versão e Apresentação									X

REFERÊNCIAS BIBLIOGRAFICAS

VILLELA, Flávia. IBGE: 6,2% da população têm algum tipo de deficiência, 2015. Disponível em <<http://www.ebc.com.br/noticias/2015/08/ibge-62-da-populacao-tem-algum-tipo-de-deficiencia>> Acessado em 05/06/2016.

THOMSEN, Adilson. BLOG FILIPEFLOP » SENSORES » COMO CONECTAR O SENSOR ULTRASSÔNICO HC-SR04 AO ARDUINO. Disponível em <<http://blog.filipeflop.com/sensores/sensor-ultrassonico-hc-sr04-ao-arduino.html>> Acessado em 05/06/2016.

PRAVDA.Ru. 03/11/2006. Na Rússia foi inventada uma bengala ultrassônica para cegos. Disponível em <<http://port.pravda.ru/science/03-11-2006/13626-bengala-0/>> Acessado em 05/06/2016.

ARDUINO. Arduino Pro Mini. Disponível em <<https://www.arduino.cc/en/Main/ArduinoBoardProMini>> Acessado em 17/06/2016,

SOUZA, Fabio. Arduino - Primeiros Passos. Disponível em <www.embarcados.com.br/arduino/>06/11/2013.