

**Unioeste - Universidade Estadual do Oeste do Paraná**  
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS  
Colegiado de Ciência da Computação  
*Curso de Bacharelado em Ciência da Computação*

**Desenvolvimento de uma Técnica de Ativação de Pontos Táteis para a Aplicação  
em *Displays Braille***

*Heitor Faccioni*

**CASCADEL  
2012**

**HEITOR FACCIÓNI**

**DESENVOLVIMENTO DE UMA TÉCNICA DE ATIVAÇÃO DE PONTOS  
TÁTEIS PARA A APLICAÇÃO EM *DISPLAYS* BRAILLE**

Monografia apresentada como requisito parcial  
para obtenção do grau de Bacharel em Ciência da  
Computação, do Centro de Ciências Exatas e Tec-  
nológicas da Universidade Estadual do Oeste do  
Paraná - Campus de Cascavel

Orientador: Prof. Marcio Seiji Oyamada

CASCADEL  
2012

**HEITOR FACCIÓNI**

**DESENVOLVIMENTO DE UMA TÉCNICA DE ATIVAÇÃO DE PONTOS  
TÁTEIS PARA A APLICAÇÃO EM *DISPLAYS* BRAILLE**

Monografia apresentada como requisito parcial para obtenção do Título de Bacharel em  
Ciência da Computação, pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel,  
aprovada pela Comissão formada pelos professores:

---

Prof. Marcio Seiji Oyamada (Orientador)  
Colegiado de Ciência da Computação,  
UNIOESTE

---

Prof. Jorge Bidarra  
Colegiado de Ciência da Computação,  
UNIOESTE

---

Prof. Edmar Bellorini  
Colegiado de Ciência da Computação,  
UNIOESTE

Cascavel, 10 de dezembro de 2012

## **AGRADECIMENTOS**

Agradeço a todos os professores que me deram base para que pudesse desenvolver esse trabalho, a meu orientador por ter me ajudado no caminho das pedras, aos servidores e técnicos da Unioeste que sempre estiveram a disposição, aos meus pais, minha esposa e minha filha que sempre estiveram ao meu lado e que me motivam a crescer mais a cada dia, aos meus amigos e colegas que sempre estavam dispostos a ajudar em momentos difíceis. E principalmente agradeço a Deus por cada dia.

A fé é um dos maiores instrumentos de motivação e temos que focar no que nos fortalece.

# Lista de Figuras

2.1	Representação Braille de caracteres [1] . . . . .	4
2.2	Exemplo de palavra escrita em Braille - Supimpa . . . . .	5
2.3	Dimensões cela Braille [2] . . . . .	5
2.4	Pessoa utilizando um <i>display</i> Braille integrado a um computador [3] . . . . .	7
2.5	<i>Display</i> Braille Focus 40 . . . . .	8
2.6	Campo magnético gerado em uma bobina quando existe corrente elétrica. Adaptado de [4]. . . . .	8
2.7	Funcionamento de um relé [5] . . . . .	9
2.8	Relé de ativação. Fonte: Adaptado de [5] . . . . .	9
2.9	Ilustração dos relés posicionados. Adaptado de [6] . . . . .	10
2.10	Relés posicionados [6] . . . . .	10
2.11	Estrutura do ativador de bobina com fecho magnético. Adaptado de [7] . . . . .	11
2.12	Diagrama do sistema de controle. Adaptado de [7] . . . . .	11
2.13	Ativador piezoelétrico. Adaptado de [8] . . . . .	12
2.14	Ativador piezoelétrico em camadas [9] . . . . .	13
2.15	Ativador polimérico . . . . .	13
2.16	Placa de ativadores poliméricos. Adaptado de [10] . . . . .	14
3.1	Esboço do protótipo . . . . .	16
3.2	Caneta Esferográfica Retrátil . . . . .	16
3.3	Trava elétrica de carro . . . . .	18
3.4	Estrutura do protótipo construída sobre carcaça de impressora Epson FX-2180 .	18
3.5	Travas elétricas de carro posicionadas no suporte que será fixado na mesa de transporte . . . . .	19

3.6	Circuito de controle do <i>display</i> Braille . . . . .	20
3.7	Diagrama de Classes . . . . .	22
3.8	Diagrama de Sequência . . . . .	24
3.9	Conector fêmea porta paralela. Adaptado de [31] . . . . .	25
3.10	Suporte dos ativadores posicionados sobre a mesa de transporte . . . . .	27
3.11	Canetas posicionadas na plataforma de suporte . . . . .	28
3.12	Vista lateral das canetas na plataforma de suporte . . . . .	28
3.13	Visão inferior da plataforma de suporte das canetas . . . . .	29
3.14	Tampa de madeira com a plataforma de suporte das canetas posicionada . . . . .	29
3.15	Testes realizados nos pontos da coluna Braille . . . . .	30

# Lista de Tabelas

3.1	Valores aproximados dos materiais utilizados na construção do protótipo. . . .	32
-----	--	----

# Sumário

<b>Lista de Figuras</b>	<b>v</b>
<b>Lista de Tabelas</b>	<b>vii</b>
<b>Sumário</b>	<b>viii</b>
<b>Resumo</b>	<b>x</b>
<b>1 Introdução</b>	<b>1</b>
1.0.1 Objetivos gerais . . . . .	1
1.0.2 Objetivos específicos . . . . .	2
1.0.3 Organização do texto . . . . .	2
<b>2 Conceitos</b>	<b>3</b>
2.1 Cegueira . . . . .	3
2.2 Braille . . . . .	4
2.3 <i>Display</i> Braille . . . . .	6
2.3.1 Microsolenoides . . . . .	7
2.3.2 Relés . . . . .	8
2.3.3 Bobina e Fecho Magnético . . . . .	10
2.3.4 Piezoelétricos . . . . .	11
2.3.5 Poliméricos . . . . .	13
<b>3 O Protótipo</b>	<b>15</b>
3.1 Descrição do protótipo e suas partes . . . . .	15
3.1.1 Canetas esferográficas retráteis . . . . .	16
3.1.2 Motor de passo . . . . .	16
3.1.3 Ativadores . . . . .	17
3.1.4 Trilho de deslocamento . . . . .	17



3.2	Ativação das canetas . . . . .	18
3.3	Circuito de Controle . . . . .	19
3.4	O Software e o funcionamento do dispositivo . . . . .	21
3.5	Comunicação . . . . .	25
3.6	Montagem do Protótipo . . . . .	27
3.7	Procedimentos para realização dos testes . . . . .	30
3.8	Resultados Obtidos . . . . .	31
<b>4</b>	<b>Conclusões e Trabalhos Futuros</b>	<b>33</b>
<b>A</b>	<b>Código fonte de conversão</b>	<b>35</b>
<b>B</b>	<b>Código fonte de controle dos motores</b>	<b>45</b>
<b>C</b>	<b>Código fonte da classe Controller</b>	<b>47</b>
	<b>Referências Bibliográficas</b>	<b>49</b>

# Resumo

Este trabalho propõe o desenvolvimento de uma técnica de ativação de pontos Braille para a utilização em *displays* Braille, visando o baixo custo, e também o desenvolvimento de um protótipo que utiliza o sistema desenvolvido. O *display* Braille é um dispositivo que auxilia deficientes visuais, principalmente cegos, na leitura de textos em computadores ou aparelhos celulares, em uma linha Braille dinâmica. A partir do estudo das soluções existentes no mercado, verificou-se o uso excessivo de ativadores, um para cada ponto da cela Braille. Foi observado que o ativador é parte determinante no custo dos dispositivos comercializados. Com base nesta informação foi projetado um dispositivo que trabalha com três ativadores fixados a uma mesa deslocável que percorre todas as celas Braille. Para manter os pontos ativos foram utilizadas canetas esferográficas com ativação tipo retrátil posicionadas verticalmente abaixo dos pontos das celas Braille. Foi construído um protótipo contendo uma coluna com 3 pontos Braille e com isso foi possível validar a utilização de ativadores com travamento mecânico em pontos táteis da cela Braille.

**Palavras-chave:** *Display* Braille, acessibilidade, inclusão digital, deficiência visual e cegueira.

# Capítulo 1

## Introdução

A acessibilidade é um assunto recorrente em vários trabalhos. Pode ser provida de diversas formas, seja ela pela adequação do ambiente para o tráfego de cadeirantes ou pela adaptação de uma colher que auxilia uma pessoa com paralisia cerebral nas suas refeições.

Os portadores de deficiência estão cada vez mais inseridos nas atividades e na comunidade como um todo. A inclusão social dessas pessoas pode ocorrer pelo meio digital, mas para a inclusão ocorrer é necessário a acessibilidade ao meio digital [11].

Existem muitas ferramentas de acessibilidade voltadas para deficientes visuais, como exemplo o leitor de tela Jaws [12], o sistema Dosvox [13] e os *displays* Braille Focus [14]. O sistema Dosvox é gratuito e está disponível para *download* na internet, já o leitor de tela Jaws é cobrado o valor de U\$ 1.095,00 (Mil e noventa e cinco dólares), mais de R\$ 2.000,00 (dois mil reais).

O Braille é um sistema de leitura com o tato, uma importante ferramenta de acessibilidade. Com o Braille pessoas cegas podem ler textos em papel, como por exemplo livros de história, biologia e muitos outros, porém existe a limitação de esse material ser impresso em equipamentos específicos para a impressão em Braille e o papel tem custo elevado.

Uma saída para não utilizar papel nem impressora Braille é o uso de aparelhos destinados à leitura de texto no formato tátil, ou *displays* Braille, porém a aquisição de tal ferramenta se torna inviável pelo seu custo elevado, mais de R\$15.000,00 (Oito mil reais), preço cotado no distribuidor da *Freedom Scientific* no Brasil, para o modelo mais baratos.

### 1.0.1 Objetivos gerais

Desenvolver um protótipo para a validação da técnica de ativação de pontos táteis Braille para a aplicação em *displays* Braille.

## 1.0.2 Objetivos específicos

O desenvolvimento do protótipo de validação, tratado neste trabalho, tem como objetivos específicos:

- Estudar as tecnologias utilizadas na ativação de pontos de um *display* Braille.
- Especificação e prototipação de um software de comunicação com o *display* Braille.
- Projeto de uma interface de comunicação entre o computador e o *display* Braille, via porta paralela.
- Minimizar o número de ativadores utilizados para a representação do Braille quando comparado à *diplays* vendidos atualmente.
- Validação da técnica de ativação com travamento mecânico.

## 1.0.3 Organização do texto

O texto apresenta, no Capítulo 2, conceitos básicos para o entendimento do trabalho, apresentação dos diferentes tipos de ativação utilizados nos *displays* Braille existentes, são eles: Microsolenoides, Relés, Bobina e Fecho Magnético, Piezoelétricos e os Poliméricos. O Capítulo 3 apresenta o projeto do protótipo, suas partes separadamente, o funcionamento de cada parte, a forma de ativação usada, o *software* de controle e gerenciamento, o circuito de controle, a forma de comunicação entre o dispositivo e o computador, a montagem do protótipo, os procedimentos executados para a realização dos testes e por fim os resultados obtidos. O Capítulo 4 apresenta a conclusão e os trabalhos futuros. Os apêndices A, B e C abrigam respectivamente os códigos fonte de conversão de um caractere de texto em uma estrutura denominada Caracter que contém as características de qual motor deve ser ativado para a representação do símbolo a ele referido, o código fonte de controle dos motores, e o código fonte da classe Controller, onde todo o processo é gerenciado.

# Capítulo 2

## Conceitos

Neste capítulo será dada uma introdução sobre os principais itens relacionados ao trabalho. Primeiramente serão dadas definições para cego, linguagem escrita Braille e por fim será apresentado o *display* Braille e as técnicas empregadas na ativação dos pontos.

### 2.1 Cegueira

Em 1966 a OMS (Organização Mundial da Saúde) registrou 66 definições para a cegueira. Em 1972 foram definidas normas para a padronização da medida de acuidade visual. A definição de cegueira da OMS [15] considera como cego o indivíduo que tem a melhor correção visual a partir de óculos ou por cirurgia, com acuidade menor ou igual a um décimo da visão normal, ou quando o campo de visão é menor ou igual a 20 graus. De maneira genérica pode-se observar a definição gerada por Antônio João Menescal Conde [16], professor do Instituto Benjamin Constant,

“Uma pessoa é considerada cega se corresponde a um dos critérios seguintes: a visão corrigida do melhor dos seus olhos é de 20/200 ou menos, isto é, se ela pode ver a 20 pés (6 metros) o que uma pessoa de visão normal pode ver a 200 pés (60 metros), ou se o diâmetro mais largo do seu campo visual subtende um arco não maior de 20 graus, ainda que sua acuidade visual nesse estreito campo possa ser superior a 20/200. Esse campo visual restrito é muitas vezes chamado “visão em túnel” ou “em ponta de alfinete”, e a essas definições chamam alguns “cegueira legal” ou “cegueira econômica”.”



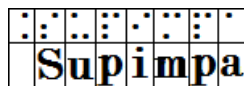


Figura 2.2: Exemplo de palavra escrita em Braille - Supimpa

De acordo com Associação Brasileira de Normas Técnicas (ABNT), em manual publicado em 2004 [2], o espaçamento entre os pontos da cela Braille deve seguir as seguintes restrições:

- diâmetro do ponto na base: 2 mm;
- espaçamento vertical e horizontal entre pontos - medido no centro dos pontos: 2,7 mm;
- largura da cela Braille: 4,7 mm;
- altura da cela Braille: 7,4 mm;
- separação horizontal entre as celas Braille: 6,6 mm;
- separação vertical entre as celas Braille: 10,8 mm;
- altura do ponto: 0,65 mm;

A Figura 2.3 mostra as especificações mencionadas.

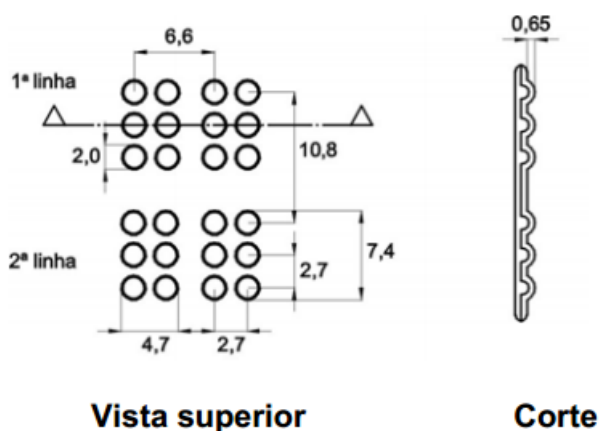


Figura 2.3: Dimensões cela Braille [2]

Para o protótipo é utilizado o espaçamento horizontal e vertical entre pontos é de 3,2 cm, pois é uma restrição física do tamanho dos ativadores. Para se obter o espaçamento de 2,7

mm entre pontos seria preciso utilizar outro mecanismo de ativação, onde os pontos possam ser mais próximos por alguma forma de agrupamento dos ativadores ou pelo tamanho reduzido dos mesmos. Outro ponto que limita a diminuição do espaçamento é a largura das canetas, as quais tem diâmetro de 10mm.

## 2.3 *Display Braille*

O *display* Braille, ou linha Braille, é um dispositivo eletromecânico que trabalha em sincronia com um *software* de gerenciamento e que representa em uma linha os caracteres em Braille [19]. Esta linha apresenta as celas Braille que possuem pinos que se movimentam verticalmente dando assim a noção tátil quanto ao caractere formado.

O *display* Braille tem a vantagem de representar os caracteres dinamicamente, não necessitando de papel, como no caso da impressoras, que tem um custo elevado. Outra vantagem é a sua utilização por pessoas surdo-cegas, em que não cabe a utilização de leitores e ampliadores de tela.

O *display* Braille trabalha mostrando uma sequência de caracteres, com número limitado pela quantidade de celas presentes no dispositivo, onde o sistema tem, como uma de suas funções, pré-processar as palavras e calcular a quantidade de caracteres necessários para a sua representação.

O *software* de gerenciamento pode ser um leitor de tela que transmite as informações obtidas da tela e as envia para o dispositivo representar ou ainda pode ser um leitor de texto, onde o usuário informa um arquivo de texto e o sistema aguarda o comando de envio das palavras do texto para o *display* Braille. O *display* também pode ser utilizado na aprendizagem da linguagem Braille, pois além de ouvir a palavra o usuário pode verificar a representação em Braille.

O *display* Braille é utilizado em conjunto ou integrado a sistemas que lhe possibilitem interação, podendo ser um computador, como mostra a Figura 2.4, e ainda, nos *displays* Braille mais recentes, aparelhos celulares [20]. Por exemplo, o *display* Braille Brailliant BI 32 [21] pode ser integrado com o iPhone 4G.

Na sequência são apresentadas as tecnologias de ativação para pontos da cela Braille existentes no mercado, em seguida, no capítulo 3 é apresentado o projeto do protótipo e seu funcionamento.





Figura 2.4: Pessoa utilizando um *display* Braille integrado a um computador [3]

Um dos fatores que influenciam diretamente o preço é a forma com que os pontos da cela Braille são ativados. Existem diferentes maneiras de os pontos serem ativados. Dentre as técnicas mais usadas encontram-se as ativações realizadas por Microsolenoide, Relés, Bobina e Fecho Magnético, Piezoelétricos e Poliméricos cujas características e funcionalidades serão abaixo descritas.

### 2.3.1 Microssolenoides

As técnicas de ativação utilizando ativadores piezoelétricos, seção 2.3.4, e microssolenoides podem ser consideradas atualmente as principais na construção de *displays* Braille. Os *displays* Focus 40 e Focus 80 [14] são exemplos de *displays* que utilizam micro solenóides como ativadores de pontos da cela Braille. A imagem 2.5 mostra o *display* Braille Focus 40.

O solenoide é uma bobina cilíndrica de fio enrolado que quando percorrida por uma corrente gera um campo eletromagnético. O campo gerado é responsável por movimentar um núcleo metálico que se localiza no interior da bobina. Uma mola faz o trabalho de retornar o núcleo ao local de partida. A Figura 2.6 (b) mostra as características do campo magnético e a Figura 2.6 (a) mostra um exemplo de solenoide com uma mola, bobina e núcleo metálico [4].

O preço do *display* Focus 40, que conta com 40 celas Braille, é de 3.895,00 dólares e o valor do Focus 80, com 80 celas, é de 7,795 dólares [14].



Figura 2.5: *Display Braille Focus 40*

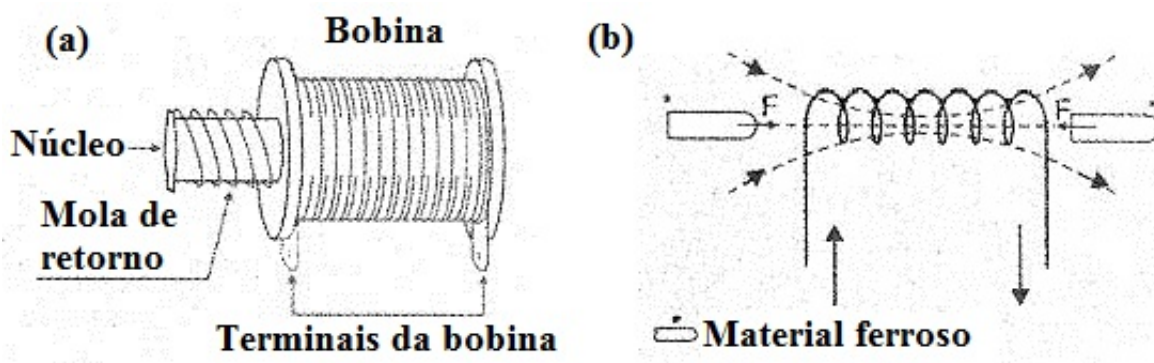


Figura 2.6: Campo magnético gerado em uma bobina quando existe corrente elétrica. Adaptado de [4].

### 2.3.2 Relés

Abordagem apresentada por Srikanthan e Subramanian em “Braille *Display* for Personal Computers” [22]. O ponto Braille é controlado por um relé. Nesta solução existe um circuito de alimentação do relé contendo capacitores, com a finalidade de manter os relés energizados.

O relé é composto por uma bobina, um núcleo, uma armadura, a qual o eletroímã exerce força, uma mola e um conjunto de contatos elétricos, conforme descrito na Figura 2.7. O princípio de funcionamento é equivalente ao do solenoide. Quando existe corrente passando pela bobina o núcleo se torna um eletroímã que atrai a armadura 2.7 (b). Quando não existe corrente a mola tenciona a armadura no sentido contrário ao do núcleo 2.7 (a). Os contatos B e C da Figura 2.7 são utilizados com a finalidade de continuidade em circuitos, onde a armadura é parte do circuito e o relé tem a função de escolha entre os circuitos. A utilização do relé nesta abordagem é totalmente mecânica, pois não trata de escolher a qual circuito será dada a conti-

nuidade e sim, em qual posição será colocado o ponto da cela Braille, alternado entre ativado e não ativado, considerando que sua posição seja rotacionada 180 graus (de cabeça para baixo) e que exista uma haste na extremidade onde existiam os contatos da armadura. Pode-se descrever seu funcionamento, quando não existe corrente passando pela bobina, como sendo o estado não ativo e quando existe corrente passando pela bobina, que faz com que a armadura seja atraída pelo núcleo e a haste seja projetada verticalmente para cima, o estado de ativo. A Figura 2.8 mostra o esquema com o motor rotacionado 180 graus.

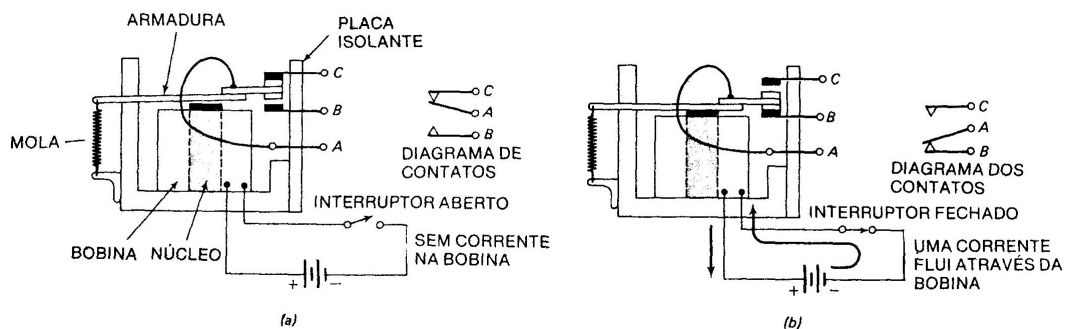


Figura 2.7: Funcionamento de um relé [5]

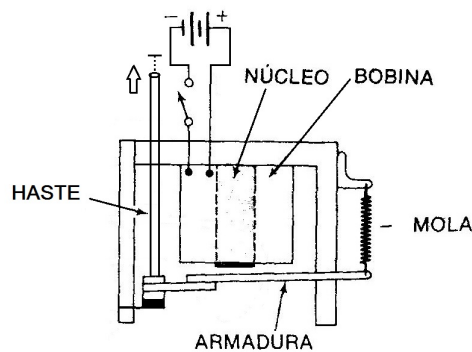


Figura 2.8: Relé de ativação. Fonte: Adaptado de [5]

No trabalho de Hey, Tsay e Liang [6] a abordagem de utilização de relés foi empregada para o desenvolvimento de um *display*. A Figura 2.9 apresenta os relés posicionados abaixo da cela Braille. A Figura 2.10 mostra a cela montada com os relés ligados aos chips de controle, que fazem a decodificação dos sinais recebidos selecionando os relés que devem ser ativados.

A utilização de relés agrega complexidade arquitetural com circuitos de controle e o gasto

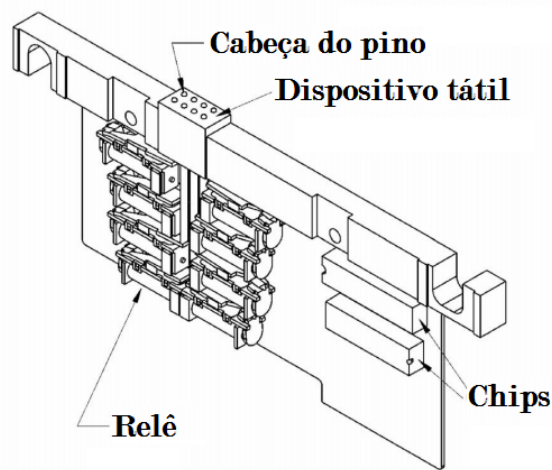


Figura 2.9: Ilustração dos relés posicionados. Adaptado de [6]

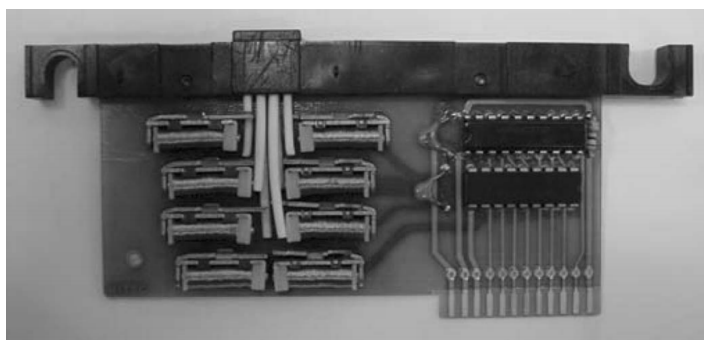


Figura 2.10: Relés posicionados [6]

na aquisição de relés é diretamente proporcional à quantidade de celas Braille que o *display* apresentará. O relé tem a vantagem de ser rápido.

### 2.3.3 Bobina e Fecho Magnético

O funcionamento do sistema de bobina com fecho magnético se dá quando a bobina SMA (*Shape Memory Alloy* - Liga com memória de forma), é aquecida pela passagem de corrente elétrica, a bobina estende-se, fazendo com que os pinos se movam para cima ou para baixo. O fecho magnético mantém os pinos nos estado ativo ou não ativo, resolvendo o problema de aquecimento e consumo de energia, pois não é necessário manter os ativadores energizados para manter a posição. A Figura 2.11 exemplifica o seu funcionamento.

O sistema com bobinas e fecho magnético é controlado por um circuito onde um módulo é

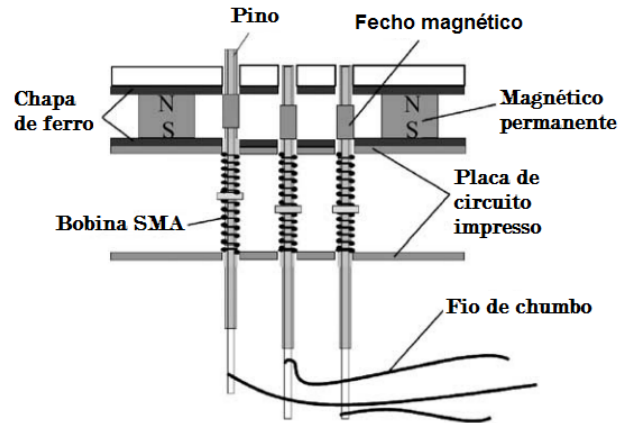


Figura 2.11: Estrutura do ativador de bobina com fecho magnético. Adaptado de [7]

constituído por uma matriz de transistores, quatro fechos endereçáveis e um demultiplexador. O controle é feito pelo computador. O demultiplexador seleciona um fecho endereçável e este envia sinais para os ativadores. A Figura 2.12 mostra um diagrama do sistema de controle.

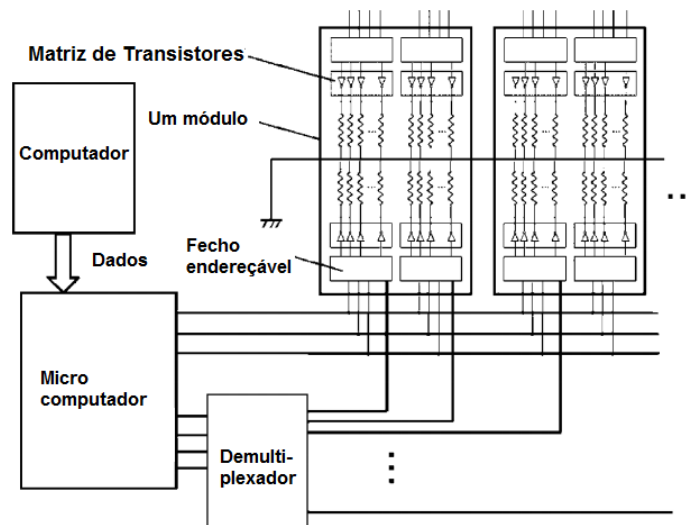


Figura 2.12: Diagrama do sistema de controle. Adaptado de [7]

O custo, considerando materiais, é relativamente baixo e sua resposta é rápida, menos de 1 segundo [7].

### 2.3.4 Piezoelétricos

De acordo com [23], a piezoelectricidade pode ser definida como:

“O termo *piezo* é derivado da palavra grega que significa pressão. Em 1880, Jacques e Pierre Curie descobriram que um potencial elétrico poderia ser gerado aplicando-se pressão a cristais de quartzo, a sais de Rochelle, e até a cristais de cana de açúcar. Nomearam este fenômeno de “o efeito piezo”.

A piezoelectricidade inversa foi deduzida matematicamente dos princípios fundamentais da termodinâmica por Lippmann em 1881. Os Curies confirmaram imediatamente a existência do “efeito piezo inverso” (quando expostos a determinados potenciais elétricos, tais materiais mudavam sua forma, se expandindo ou se contraindo), e continuaram os estudos para obter a prova quantitativa da reversibilidade completa das deformações eletro-elasto-mecânicas em cristais piezoelétricos.”

O motor piezoelétrico é utilizado na ativação dos pontos em *displays* Braille. Quando aplicada uma grande tensão o material do motor sofre alterações de dilatação ou contração, dependendo do material, e o ponto Braille é projetado. O comportamento e a utilização de elementos piezoelétricos pode ser observada no vídeo “Piezoelectric Braille system with PC parallel port control and HV transistors” [24], onde é apresentado um *display* Braille com duas celas.

A Figura 2.13 ilustra os ativadores piezoelétricos posicionados, onde está localizada a área de deformação e o movimento quando ativado. Para o modelo de cela Braille da Figura 2.13 a tensão de alimentação dos ativadores é de 200 Volts. A Figura 2.14 mostra em detalhes o ativador piezoelétrico e suas camadas.

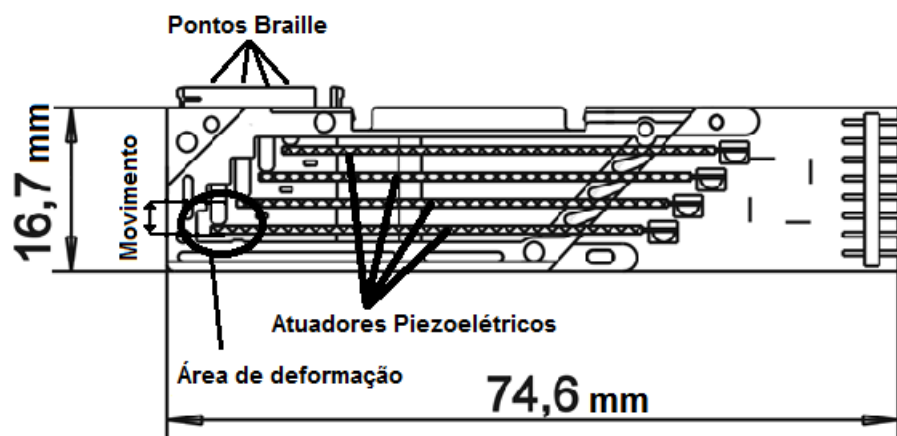


Figura 2.13: Ativador piezoelétrico. Adaptado de [8]

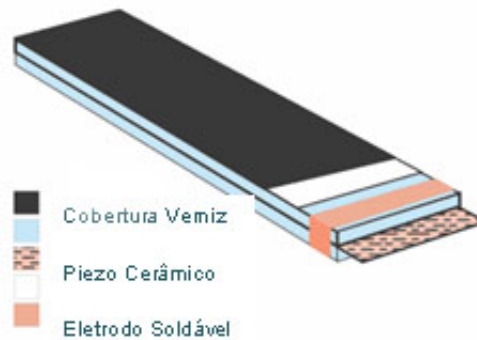


Figura 2.14: Ativador piezoelétrico em camadas [9] .

O “mouse Braille” ou, como foi batizado, Portáctil [25] é um exemplo na utilização de piezoelétricos como ativadores, onde são utilizados três celas Braille. Segundo os criadores da invenção a utilização dos piezoelétricos foi cerca de vinte vezes mais eficiente que os solenoides com relação a duração de bateria.

### 2.3.5 Poliméricos

Os ativadores poliméricos são feitos de um metal de polímero iônico composto (IPMC) [26]. A Figura 2.15, retirada de [10], mostra a estrutura do ativador polimérico.

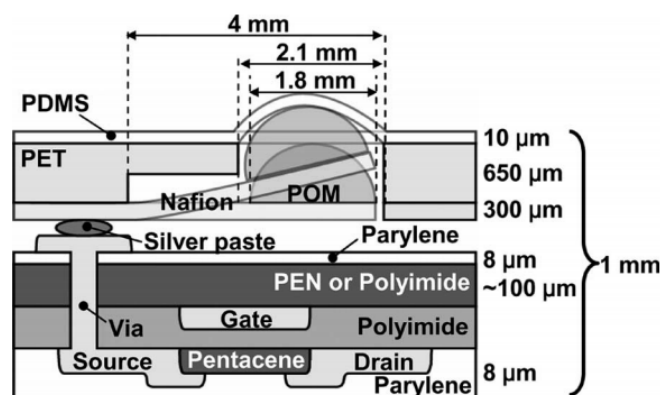


Figura 2.15: Ativador polimérico

O controle de ativação dos pontos é feito através da matriz de transistores, onde o fluxo de ativação ocorre linha-a-linha, selecionando-se as colunas da linha de maneira similar ao endereçamento de memórias de computador. O processo de ativação ocorre quando é aplicada

uma tensão sobre o ativador por um tempo igual a 0,9 segundos. O ativador desloca-se e após os 0,9 segundos a tensão é zerada. O processo é repetido linha-a-linha, ativando todos os pontos necessários. O atributo de retenção presente nos materiais do ativador fazem com que ele se mantenha ativo [10].

As placas dos ativadores poliméricos têm como vantagens a flexibilidade e a espessura fina. São constituídas de três camadas, a primeira é uma camada de cobertura, a segunda é a camada dos ativadores e a última é a camada dos transistores. A Figura 2.16 (a) mostra uma placa sendo flexionada, (b) mostra a placa em camadas e a Figura 2.16 (c) mostra a matriz de transistores.

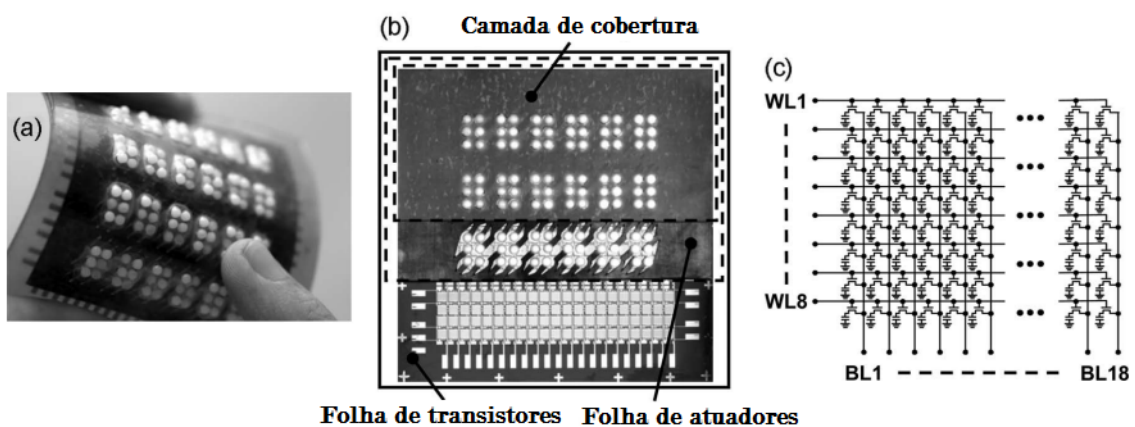


Figura 2.16: Placa de ativadores poliméricos. Adaptado de [10]

Essa tecnologia é nova no mercado de representação Braille e uma grande vantagem dos ativadores poliméricos, em relação aos ativadores microssolenoides e ativadores piezoelétricos, é a sua miniaturização mais simplificada. De acordo com os autores a folha de ativadores tem espessura de 1 mm e peso de 5,3 gramas e pode ser facilmente implantada em dispositivos móveis como celulares.



# Capítulo 3

## O Protótipo

O trabalho tem como objetivo principal o desenvolvimento de um protótipo de um *display* Braille de baixo custo. Após análise das informações relacionadas aos dispositivos existentes no mercado foi projetado um *display* Braille capaz de representar uma linha Braille a partir de sinais vindos de um computador.

Analisando os mecanismos existentes podemos perceber que para cada ponto da cela Braille existe um ativador dedicado. Este fator motivou o projeto de um mecanismo que é capaz de diminuir o número de ativadores necessários para a ativação de todos os pontos Braille de uma linha (abordagem descrita com detalhes na seção 3.2). Uma decisão de projeto foi a de controlar o dispositivo via teclado do computador, executando as ações de controle do *software* sem a necessidade da existência de botões de controle no dispositivo, o que agregaria custo e tempo ao projeto.

### 3.1 Descrição do protótipo e suas partes

O protótipo foi montado dentro de uma caixa de madeira com dimensões 60,0 cm de comprimento por 40,0 cm de largura e 40,0 cm de altura.

A Figura 3.1 mostra um esboço de como as partes do protótipo estão dispostas. Nas próximas subseções serão apresentados os componentes que são utilizados para a construção do protótipo, tais como canetas esferográficas retráteis, motores de passo, ativadores de travas elétricas e o trilho de deslocamento.

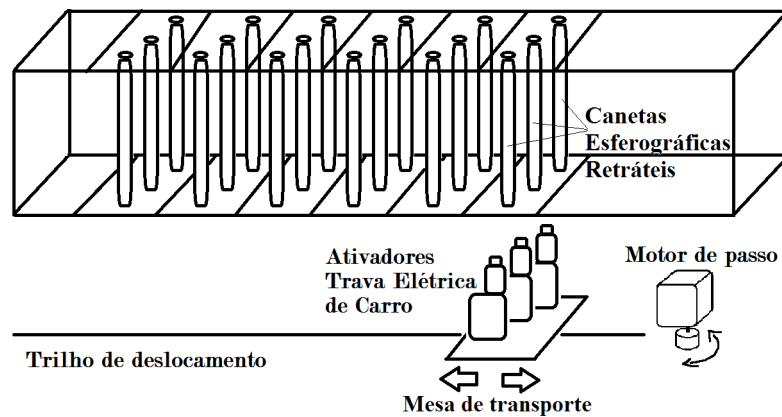


Figura 3.1: Esboço do protótipo

### 3.1.1 Canetas esferográficas retrátéis

As canetas esferográficas retrátéis são dispositivos de travamento mecânico. Quando são pressionadas, um mecanismo de travamento interno é condicionado a manter a ponta para fora da estrutura da caneta. Quando a caneta é pressionada novamente o mecanismo interno faz com que a ponta da caneta seja recolhida. A caneta exerce a função de representar o ponto da cela Braille. Quando a caneta é ativada o ponto é exibido e quando ela é desativada o ponto é recolhido. A Figura 3.2 mostra um exemplo de caneta esferográfica retrátil.



Figura 3.2: Caneta Esferográfica Retrátil

### 3.1.2 Motor de passo

Motores de passo são Máquinas de Relutância Variável (MRV) que são energizados sequencialmente de modo passo a passo, fazendo com que o motor gire por um ângulo específico [27].

“Uma característica importante do motor de passo é a sua compatibilidade com os sistemas eletrônicos digitais. Esses sistemas são comuns em uma ampla variedade de aplicações e continuam a tornar-se mais potentes e menos dispendiosos. Por

exemplo, o motor de passo é usado frequentemente em sistemas de controle digital em que o motor recebe comandos a laço aberto de forma de um trem de pulsos para girar um eixo ou mover um objeto por uma distância específica. Aplicações típicas incluem motores de alimentação de papel e posicionamento de cabeça de impressão em impressoras e plotters, motores de acionamento e posicionamento da mesa de trabalho e da ferramenta em equipamentos de usinagem controlados numericamente. Em muitas aplicações, a informação de posição pode ser obtida simplesmente deixando registrado o número de pulsos enviados ao motor, caso em que não são necessários sensores de posição nem controle realimentado.”[27]

O motor de passo utilizado nesse trabalho conta com cinco fios, quatro deles destinados aos passos (terra) e um para a alimentação, no caso 5 Volts. O motor foi obtido de uma impressora EPSON FX-2180.

O motor de passo faz com que a mesa de transporte dos ativadores percorra o trilho de deslocamento, posicionando os ativadores corretamente sob as canetas.

### **3.1.3 Ativadores**

Os ativadores foram montados com travas elétricas de carro, onde podemos observar na Figura 3.3. O seu papel é ativar as canetas que representarão os pontos Braille na cela. São utilizados três ativadores posicionados lado a lado em um suporte sobre a mesa de transporte. O sistema de gerenciamento do *display* Braille envia um sinal que é decodificado pelo circuito e os motores selecionados, a partir da representação do sinal tátil de Braille definida pelo *software*, são ativados. Um ciclo de movimentação para a ativação de um ponto da cela Braille, descrito com mais detalhes na seção 3.3, consiste de uma subida e uma descida da cabeça de ativação, localizada na porção superior do ativador, apresentada na Figura 3.3.

### **3.1.4 Trilho de deslocamento**

Trilho de deslocamento é uma barra metálica cilíndrica por onde a mesa de suporte dos ativadores é deslocada, com a atuação do motor de passo auxiliado por uma correia fixada à mesa de transporte. O trilho da impressora EPSON FX-2180 foi mantido e sobre ele foi construída uma base para se fixar os ativadores, esta será chamada de mesa de transporte.



Figura 3.3: Trava elétrica de carro

A Figura 3.4 mostra o trilho de deslocamento posicionado dentro da caixa de madeira.

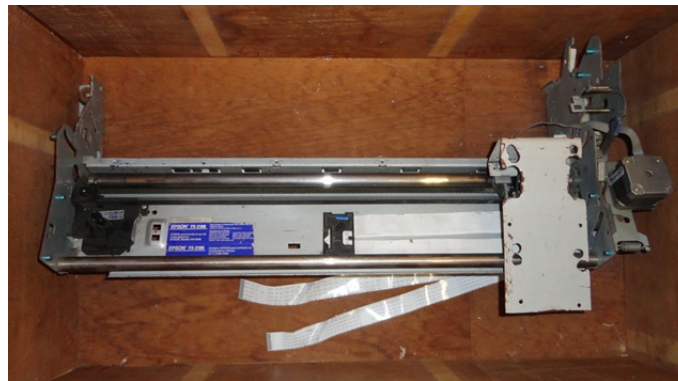


Figura 3.4: Estrutura do protótipo construída sobre carcaça de impressora Epson FX-2180

## 3.2 Ativação das canetas

Para reduzir o número de ativadores foram utilizados mecanismos de travamento mecânico fixos em uma plataforma, as canetas esferográficas retráteis, e os três ativadores são deslocados pela mesa de transporte através do trilho de deslocamento. Foram utilizadas canetas esferográficas retráteis - Figura 3.2 - para a representação do ponto Braille. Uma vez ativada ela exhibe a ponta da caneta para representação do ponto tátil. Foram posicionadas verticalmente entre duas placas de acrílico, uma superior e uma inferior, fixadas por pilares de madeira. Para ocorrer a ativação do ponto tátil, a trava elétrica é posicionada embaixo da caneta a ser ativada e pressiona o dispositivo retrátil da mesma, esta exhibe cerca de 1mm de sua ponta ativa, dando a

representação tátil do ponto Braille na placa de acrílico superior, quando a trava elétrica ativa o mecanismo retrátil novamente a ponta ativa da caneta é recolhida e a representação tátil do ponto Braille desaparece.

A mesa que transporta os ativadores foi construída a partir de uma impressora Epson FX-2180 [28] sucateada. Foram removidas todas as partes no entorno do carro de impressão e fixado a esse um suporte onde estão localizados os ativadores. Foi mantido o motor de passo que faz a movimentação da mesa de transporte e a fonte de energia que a alimenta é uma fonte de computador, onde é usada a tensão de 5 volts.

A Figura 3.5 mostra as travas elétricas de carro posicionadas no suporte que será fixado na mesa de transporte.

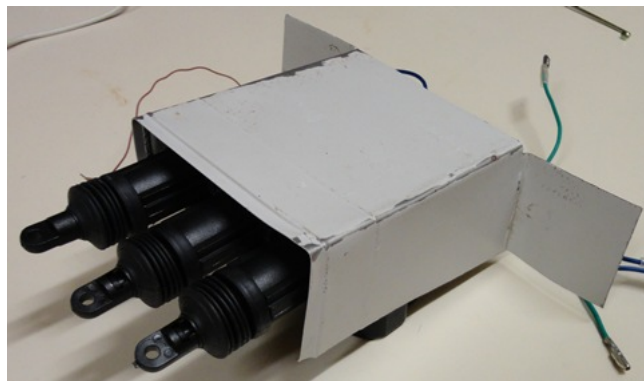


Figura 3.5: Travas elétricas de carro posicionadas no suporte que será fixado na mesa de transporte

### 3.3 Circuito de Controle

Esta seção apresenta o projeto do circuito responsável pela decodificação dos sinais vindos da porta paralela. Para simplificar e baratear o projeto buscou-se utilizar componentes de prateleira, baratos e de fácil acesso.

A Figura 3.6 mostra o circuito construído utilizado no controle dos ativadores e da mesa de transporte. São utilizados transistores para a alimentação de tensão nos relés, pois a tensão da porta paralela não é suficientemente grande. O relé X2 tem a função de chaveamento dos fios Branco e Marrom, fazendo com que um deles seja aterrado. O aterramento dos fios Branco ou Marrom faz com que a Centralina envie tensão para a movimentação de subida ou descida

das Travas Elétricas de Carro (Ativadores). O relés ReleMotor1, ReleMotor2 e ReleMotor3 são utilizados para o chaveamento e com isso a continuidade no envio do sinal de subida ou descida vindo da Centralina, representando que a caneta esferográfica retrátil posicionada sobre o motor será ativada se o relé estiver “ligado” e não ativada se o relé estiver “desligado”. Para o controle do motor de passo foi utilizado o *driver* ULN2003. Foram utilizadas 2 tensões externas, onde a primeira é a fonte de alimentação da Centralina, que faz a ativação das travas elétricas, e a outra para a alimentação do motor de passo e dos relés. Na seção 3.5 a comunicação será detalhada, bem como a utilidade de cada pino da porta paralela conectado ao circuito.

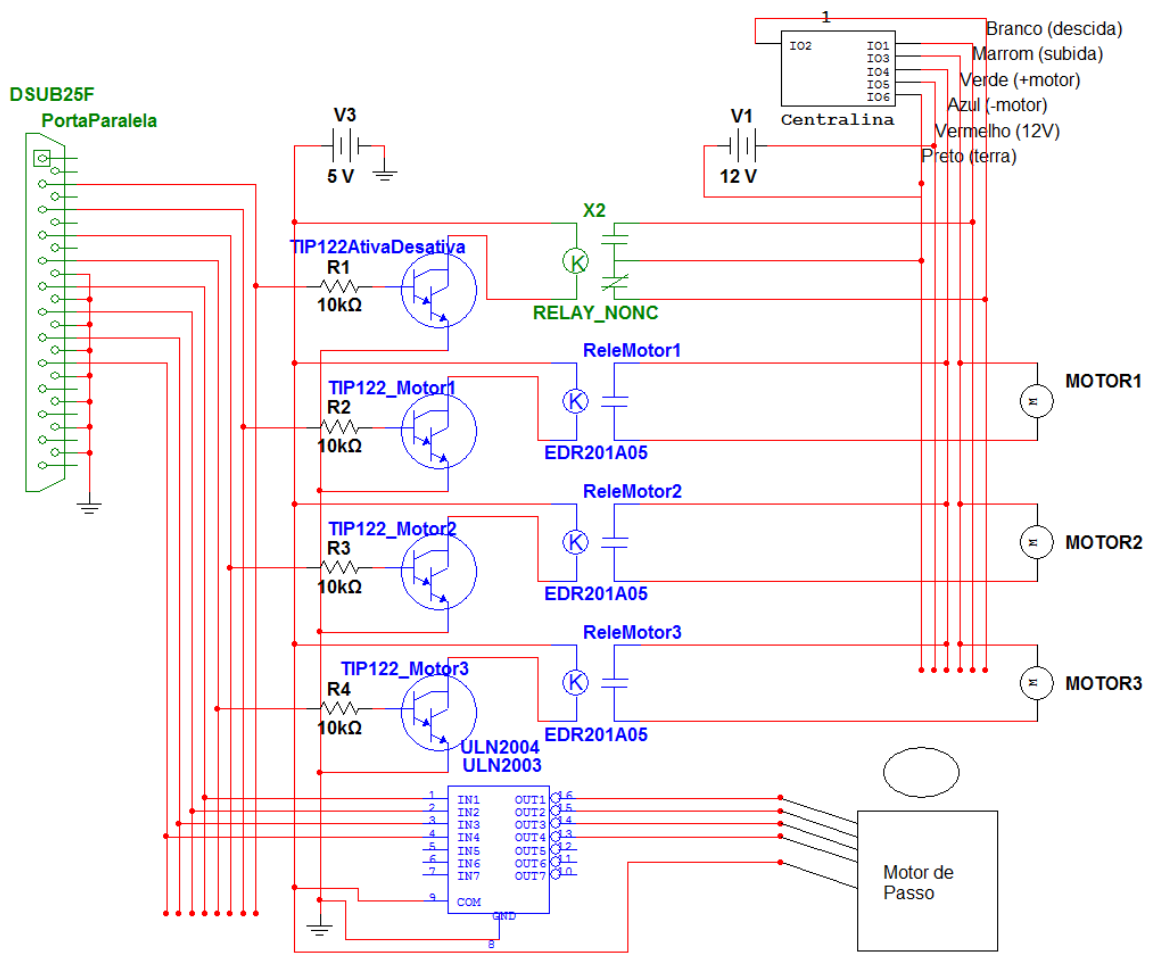


Figura 3.6: Circuito de controle do *display* Braille

## 3.4 O Software e o funcionamento do dispositivo

Foi implementado um *software* de controle em Java [29] para o *display* Braille. O *software* faz o controle de transmissão de informação do computador para o dispositivo através da porta paralela, esta descrita com maiores detalhes na subseção 3.5. O código fonte da codificação dos caracteres que serão enviados ao *display* Braille pode ser visto no Apêndice A.

O sistema foi adaptado às condições do *hardware* (*display* Braille) para realizar testes de validação em uma única coluna da cela Braille para o dispositivo, pois o protótipo conta apenas com três pontos. Tomando como premissa que o protótipo conta com mais de uma cela Braille, pode-se descrever os passos para a sua execução como sendo:

- O usuário abre arquivos no formato texto e o sistema exibe o mesmo em uma área da tela principal.
- O texto aberto será enviado em partes para a representação no *display* com a utilização de botões na interface do sistema. Para cada conjunto de palavras que serão enviados o sistema gera uma sequência de caracteres com tamanho compatível ao número de celas Braille presentes no *display*.

Quando é requisitado o envio de um conjunto de palavras o sistema manda comandos para limpar o que estava escrito no *display* e passa a enviar comandos de escrita do novo conjunto, ou se não existe nada escrito no *display* ele apenas escreve o conjunto de palavras.

Tomando um texto como entrada, o sistema o codifica em sinais e o envia ao dispositivo, que por sua vez interpreta e executa as ações determinadas.

A Figura 3.7 mostra um diagrama de classes que representa a estrutura do *software*. O sistema está separado em 3 pacotes principais:

- **Visão** - onde estão localizadas as classes que fazem a interação com o usuário.
- **Modelo** - onde estão localizadas as classes que definem objetos, no caso a classe *Caractere* que é utilizada para o mapeamento de quais pontos de uma cela Braille devem ser ativos.
- **Controle** - onde estão as classes que fazem o controle de todo o sistema. A classe *Controller* pode ser vista no Apêndice C, e é ela quem trata todas as conversões, sendo do texto vindo da visão até o envio dos caracteres mapeados em Braille para o *display*.

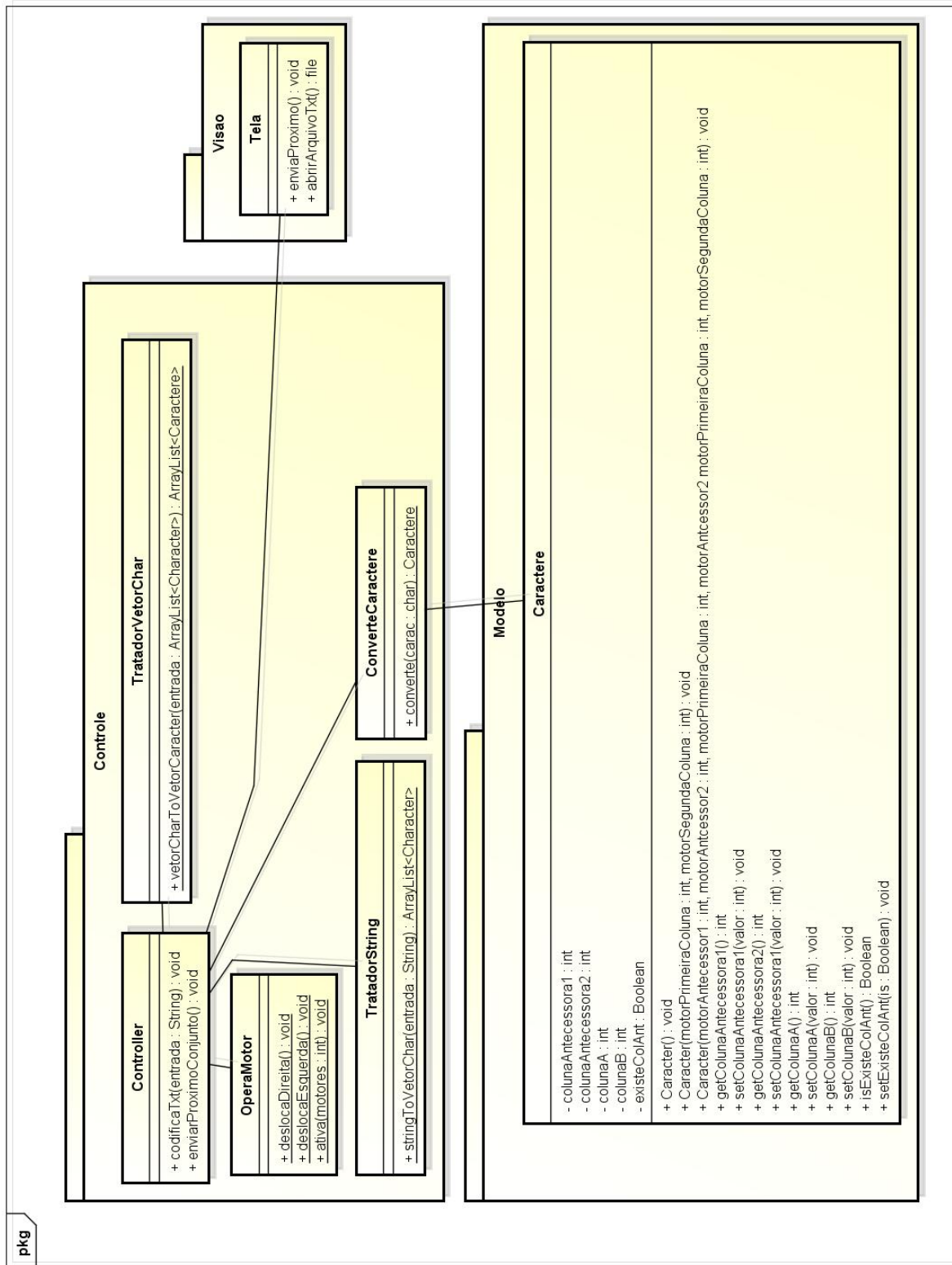


Figura 3.7: Diagrama de Classes



A Figura 3.8 mostra um diagrama de sequência do sistema, o diagrama pode ser explicado em forma de passos:

1. A partir da Tela principal é aberto um arquivo de texto que é enviado ao Controller.*display*.
2. A classe Controller envia o texto recebido para ser separado em um vetor de palavras.
3. Após separado em palavras é feito o processo de separação em Caractere, onde este define, em forma de valores inteiros que serão enviados pela porta paralela, quais os motores que devem ser ativos para a formação do caractere em questão.
4. O processo de separação em Caractere consiste em passar todas as letras da palavra convertendo cada caractere em valores inteiros que representam os pontos das colunas que devem ser ativos.
5. Depois de feito o processamento de todas as palavras do texto o envio dos caracteres pode ser feito pelo sistema. O usuário seleciona o comando para enviar próximo conjunto de informação e a classe Controller faz o controle dos motores com o envio dos valores inteiros via porta paralela. O processo de envio se repete até não existir mais texto para ser representado no *display*.

Uma restrição do sistema implementado é que a análise do texto é feita pontualmente em apenas 1 caractere por vez, o que gera problemas para a representação de palavras escritas com todas as letras maiúsculas e para números que tenham mais de um caractere de comprimento. Por exemplo o número 12, a representação gerada pelo sistema corresponde a uma indicação de número, o número 1, outra indicação de número e o número 2.

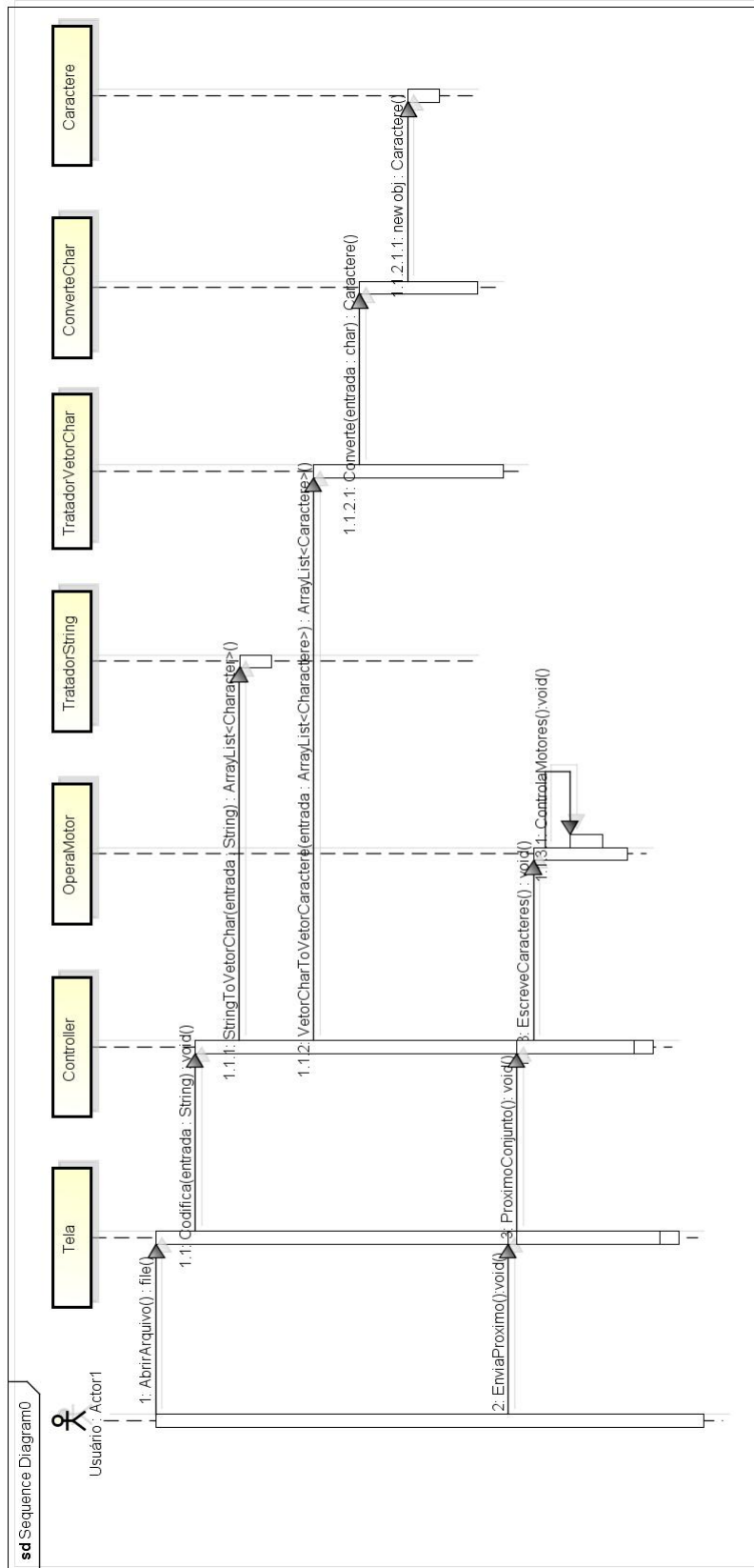


Figura 3.8: Diagrama de Sequência

A implementação da classe `Character` leva em conta a possibilidade de existir um caractere que antecede a representação, sendo este usado para representar se o valor a seguir é um número ou uma letra maiúscula. O problema consiste na inserção do sinal de número ou de maiúscula sempre ao detectar uma das características, pois a análise é focada no caractere e não na palavra inteira. Por exemplo a palavra "SOFTWARE" gera como saída a sequência de símbolos em Braille: MAIÚSCULO - S - MAIÚSCULO - O - MAIÚSCULO - F - MAIÚSCULO - T - MAIÚSCULO - W - MAIÚSCULO - A - MAIÚSCULO - R - MAIÚSCULO - E. Para a palavra "SOFTWARE" a saída deveria ser: MAIÚSCULO - MAIÚSCULO - S - O - F - T - W - A - R - E.

### 3.5 Comunicação

A interface de comunicação escolhida foi via porta paralela. Foi utilizada a biblioteca Java `ParallelPort` [30] para se ter acesso a escrita na porta paralela. "A porta paralela é uma interface de comunicação entre o computador e um periférico"[31], neste caso, o periférico é o *display* Braille. A porta paralela, Figura 3.9, conta com 25 pinos, onde 8 pinos são destinados para o envio de dados (D0 à D7), 8 pinos são para o aterramento (GND), e os outros são pinos de controle. Para esse trabalho foram utilizados os pinos de dados e de aterramento. O código fonte que faz o controle dos ativadores e do motor de passo pode ser visto no apêndice B.

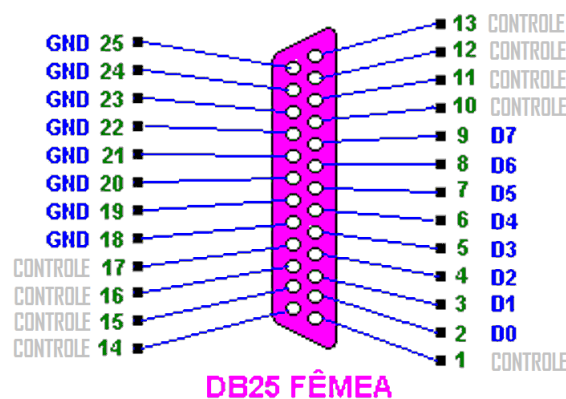


Figura 3.9: Conector fêmea porta paralela. Adaptado de [31]

A implementação da comunicação via porta paralela, neste caso, não requer a construção de nenhum tipo de *demultiplexador* já que todos os comandos são interpretados diretamente pelo dispositivo, além de ter a vantagem na simplicidade de implementação.

São utilizados 8 (oito) bits para o controle de todos os motores. Onde existem 3 (três) motores responsáveis pela ativação e 1 (um) motor de passo responsável pelo movimento da mesa de transporte dos ativadores. Dos oito bits de dados, numerados de 0 a 7, quatro são dedicados ao controle de passo do motor de passo, três são responsáveis pela escolha dentre os três motores ativadores e um bit pelo sentido de movimentação dos ativadores, que é de subida ou descida. Abaixo é mostrado o número do bit e sua função.

- **D0** - Controle de subida e descida dos ativadores. Quando o bit é falso (tensão próxima de zero Volts) os ativadores selecionados sobem, quando o bit tem valor verdadeiro (tensão próxima de cinco Volts) os ativadores selecionados descem.
- **D1** - Quando ativo libera para a ativação o motor da primeira linha.
- **D2** - Quando ativo libera para a ativação o motor da segunda linha.
- **D3** - Quando ativo libera para a ativação o motor da terceira linha.
- **D4** - Primeiro passo do motor de passo.
- **D5** - Segundo passo do motor de passo.
- **D6** - Terceiro passo do motor de passo.
- **D7** - Quarto passo do motor de passo.

A seleção dos motores serve para marcar pontos específicos de uma coluna, onde o movimento dos motores selecionados é de subida e descida. A mesa trabalha fazendo deslocamentos de tamanho igual a distância entre pontos de uma coluna da cela Braille, posicionando os ativadores sob as canetas que serão ativadas.

## 3.6 Montagem do Protótipo

A mesa de transporte foi montada sobre uma carcaça de uma impressora Epson FX-2180. Sobre a mesa foi fixado um suporte onde estão os 3 ativadores. O suporte, que fixa as travas elétricas de carro à mesa de transporte - Figura 3.10-, foi confeccionado com metal de gabinete de computador.

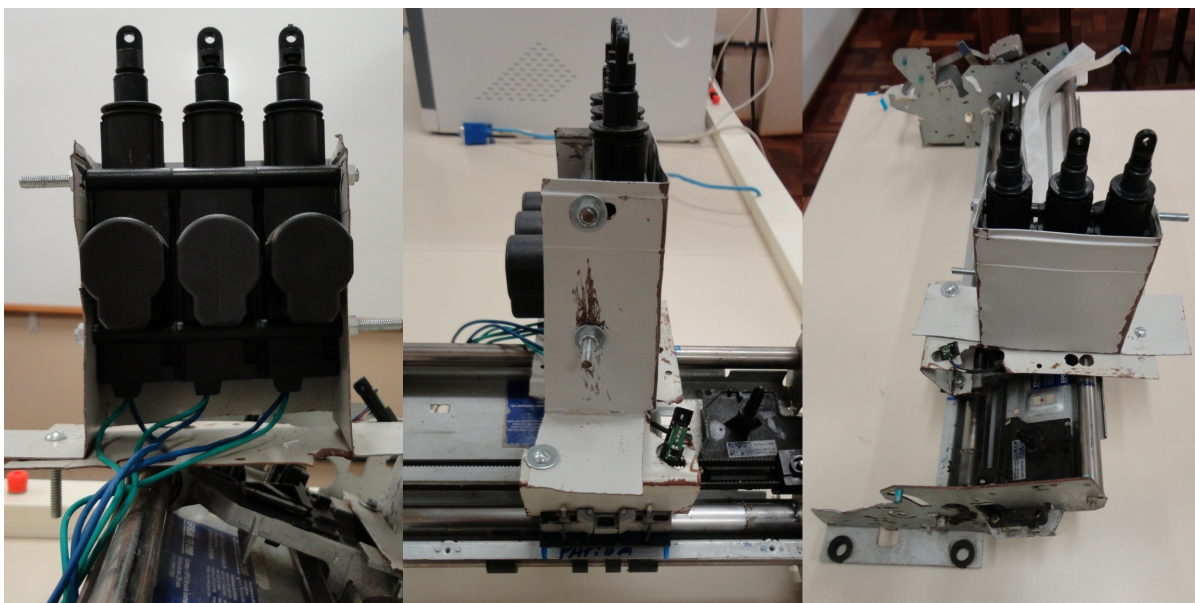


Figura 3.10: Suporte dos ativadores posicionados sobre a mesa de transporte

O circuito foi montado em uma *protoboard* (placa com furos e conexões condutoras para montagem de circuitos experimentais) e, para a versão protótipo, é fixada externamente, junto à fonte de alimentação, à caixa de madeira onde estão os ativadores e as canetas.

Os fios que ligam as travas elétricas e os fios do motor de passo estão ligados diretamente na *protoboard*, onde foi montado o circuito de controle do *display* Braille.

A estrutura da impressora foi fixada na caixa de madeira. Sobre a mesa de transporte está a plataforma de suporte contendo duas colunas com 3 canetas cada. Esta plataforma é feita com acrílico retangular na parte superior e inferior, e são usados 4 pilares de madeira, um em cada extremidade. As Figuras 3.11 e 3.12 mostram a primeira coluna de canetas posicionadas na plataforma de suporte.

A Figura 3.13 mostra a parte inferior da plataforma de suporte onde as canetas estão fixadas, sendo essa a parte de contato entre os ativadores e as canetas.



Figura 3.11: Canetas posicionadas na plataforma de suporte

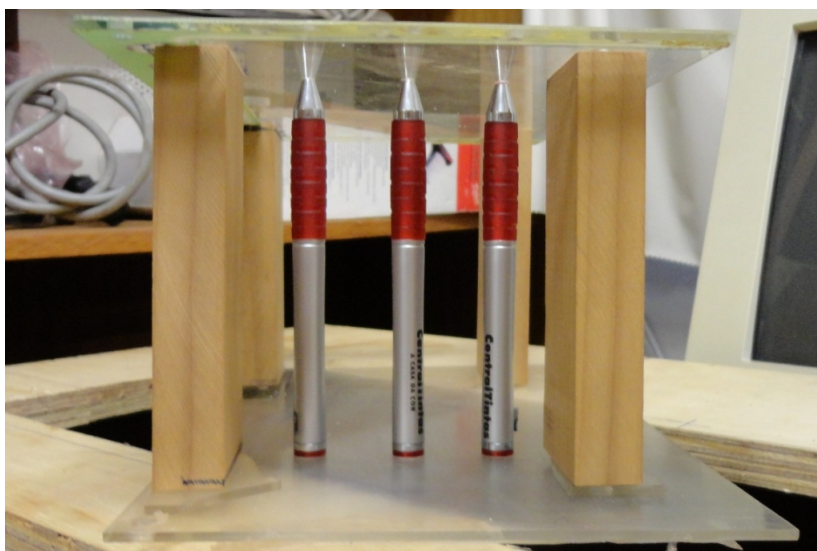


Figura 3.12: Vista lateral das canetas na plataforma de suporte

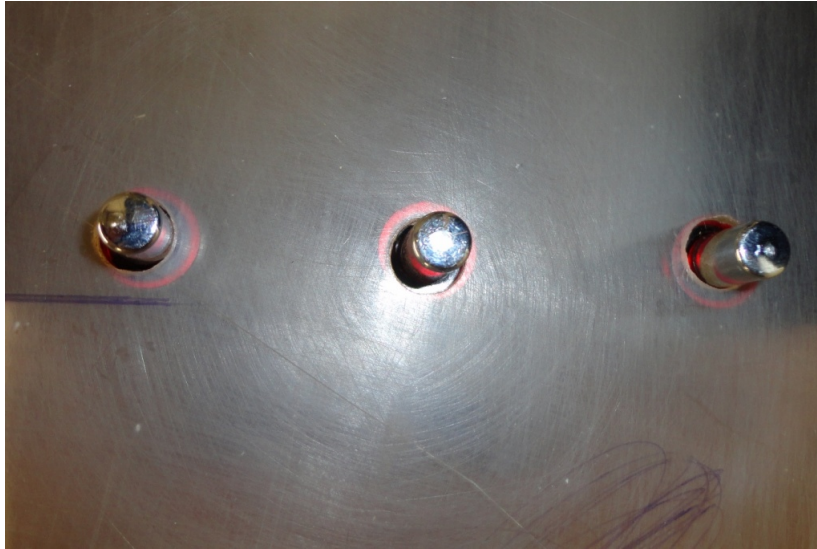


Figura 3.13: Visão inferior da plataforma de suporte das canetas

A plataforma de suporte das canetas foi fixada à tampa de madeira da caixa, como mostrado na Figura 3.14.

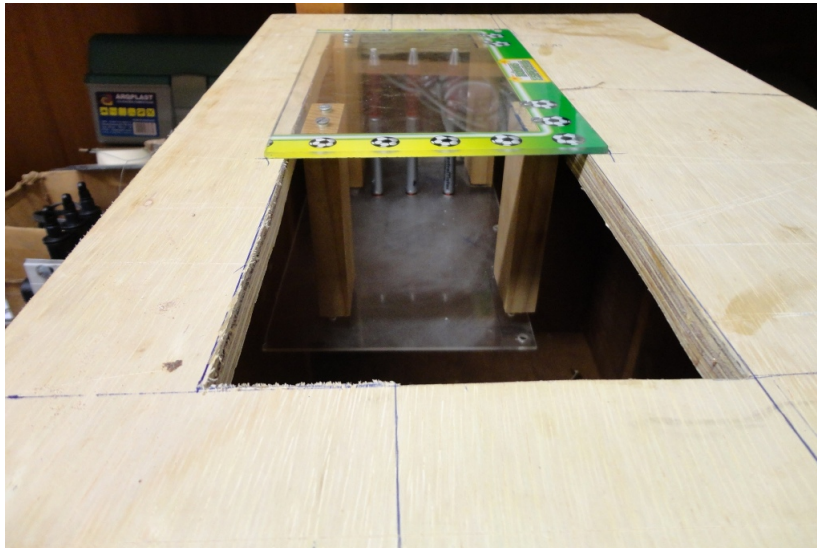


Figura 3.14: Tampa de madeira com a plataforma de suporte das canetas posicionada

Após o término da montagem dos componentes foram realizados os testes.

### 3.7 Procedimentos para realização dos testes

Para a realização dos testes, o sistema de controle do *display* Braille foi adaptado para o envio de somente uma coluna, três pontos da celda Braille, pois é uma restrição do protótipo.

A validação dos testes ocorre com a ativação correta de um ponto Braille da coluna, onde um ponto tem que ser ativado e posteriormente desativado, consistindo da execução de 7 testes, estes fazem todas as combinações de ativação para 3 pontos. A Figura 3.15 mostra os teste e os pontos testados.

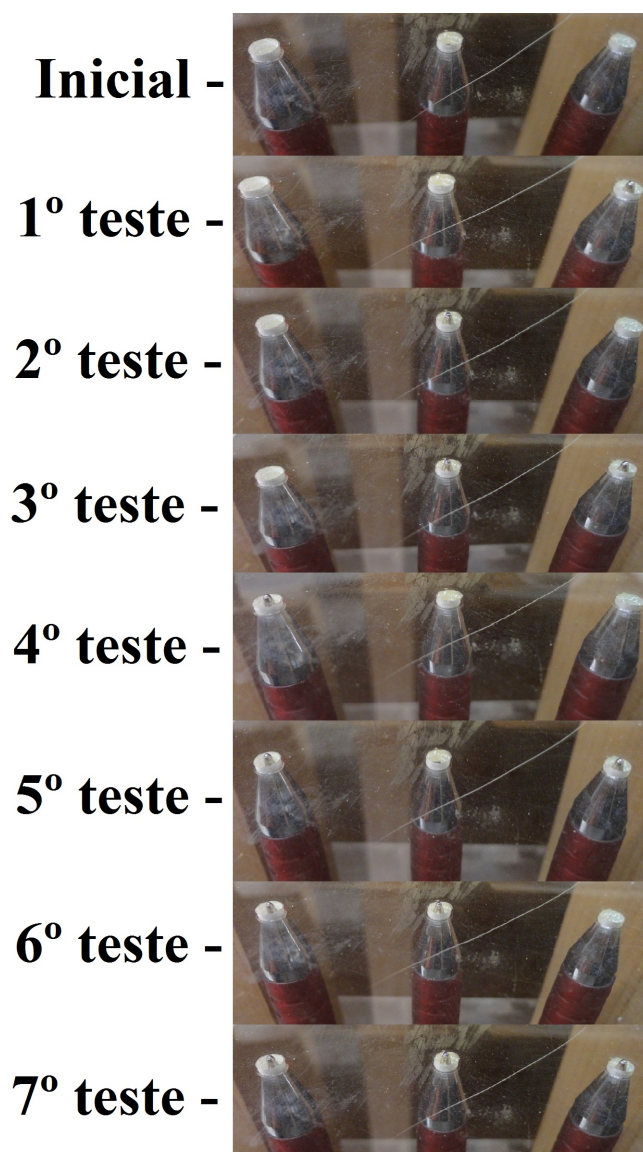


Figura 3.15: Testes realizados nos pontos da coluna Braille



## 3.8 Resultados Obtidos

Os testes demonstraram que a utilização de ativadores e de travamento mecânico é válida, provado pelo funcionamento correto do *display* na representação dos pontos de 1 coluna da cela Braille, onde os ativadores são acionados e as canetas permanecem ativas até a sua desativação.

Os espaçamentos entre pontos foram de 3,2 cm, o que está muito além da especificação da Associação Brasileira de Normas Técnicas, que estabelece o espaçamento entre pontos horizontais e verticais é de 2,7 mm, diferença de 29,7 mm.

A eficiência energética pode ser melhorada com a miniaturização do protótipo. Os ativadores são muito fortes e consomem muita energia. Com a substituição por ativadores menores o consumo será reduzido. Com travamento mecânico não é necessário energia para manter os pontos ativos, possibilitando uma diminuição do consumo energético do protótipo.

Com a realização dos testes foi possível observar que o protótipo gera muito ruído, mesmo sem a movimentação da mesa de transporte, que é outra fonte de ruído. As partes são grandes e exercem mais força que o necessário para o funcionamento e com isso o ruído é gerado. O ruído pode ser controlado com a miniaturização e a partir de processos industriais de produção.

Também foi possível verificar que a velocidade de ativação dos pontos é limitada pelo curso que o ativador deve percorrer para ativar a caneta esferográfica retrátil. Para o caso onde a mesa é deslocada, os ativadores devem ser ativados com a mesa posicionada abaixo das canetas, com isso ocasionando um retardamento no processo de ativação de todas as canetas.

A miniaturização do protótipo se torna indispensável, visto que vários dos problemas serão resolvidos ou minimizados com a miniaturização. O próprio tamanho e o peso do protótipo torna o seu transporte uma tarefa difícil.

O baixo custo, quando comparado a outros *displays* Braille vendidos no mercado, foi obtido com a utilização de materiais baratos e de fácil acesso. A Figura 3.1 mostra valores aproximados para a obtenção dos materiais utilizados na construção do protótipo. Para a estimativa dos valores de alguns materiais que não foram comprados foi feita uma pesquisa em sites de compra e venda de produtos, são esses:

- Cabo porta paralela
- Caixa de madeira

- Canetas
- Fonte
- Placa de acrílico

Tabela 3.1: Valores aproximados dos materiais utilizados na construção do protótipo.

<b>Material</b>	<b>Valor aproximado</b>	<b>Qnt</b>	<b>Total</b>
Cabo porta paralela	R\$ 4,00	1	R\$ 4,00
Caixa de madeira 60x40x40 cm	R\$ 50,00	1	R\$ 50,00
Canetas esferográfica retrátil	R\$ 0,40	6	R\$ 2,40
Driver ULN2003	R\$ 0,80	1	R\$ 0,80
Fonte de computador	R\$ 30,00	1	R\$ 30,00
Impressora Epson FX-2180 *	R\$ -	1	R\$ -
Kit trava elétrica de carro	R\$ 25,00	1	R\$ 25,00
Lata de gabinete de computador **	R\$ -	1	R\$ -
Placas de acrílico	R\$ 10,00	2	R\$ 20,00
Protoboard	R\$ 20,00	1	R\$ 20,00
Relê	R\$ 4,50	4	R\$ 18,00
Transistor TIP122	R\$ 1,50	4	R\$ 6,00
<b>Total</b>	<b>R\$</b>		<b>176,20</b>

\* - doação. \*\* - conseguida em um local onde existem computadores velhos.

## Capítulo 4

# Conclusões e Trabalhos Futuros

Com os preços dos *displays* Braille elevados o mercado começa a buscar por ferramentas mais baratas. Neste trabalho foi proposto a utilização método de ativação utilizando trava mecânica para se reduzir os custos. Outras soluções podem ser utilizadas para o barateamento, como por exemplo a diminuição do número de celas Braille, técnica adotada no “Portáctil”(ou “mouse Braille”) [25], onde existe apenas uma cela Braille e o movimento do dispositivo sobre uma superfície plana emula a passagem do dedo do usuário sobre o Braille.

Os resultados apresentados na seção 3.8 mostram a viabilidade da utilização do mecanismo de travamento mecânico, bem como o baixo custo para a sua criação, no entanto é importante consideram alguns aspectos:

- Distância entre os pontos precisa ser reduzida para atender as normas da ABNT;
- Redução de ruído emitido;
- Diminuição do tamanho do dispositivo.

Os objetivos foram alcançados em parte, visto que não foi possível a construção do protótipo com uma cela Braille completa, porém uma única coluna com três pontos foi testada e mostrou a viabilidade na utilização do mecanismo de travamento mecânico para o ponto da cela Braille.

As travas mecânicas apresentam um baixo custo quando comparadas a outros ativadores como, por exemplo, os ativadores piezoelétricos e os microssolenoides. Com tudo, é possível afirmar que o processo de ativação será demorado, com base no tempo necessário para as impressoras percorrerem toda a extensão do trilho de deslocamento. Desta forma, o uso desta técnica mais apropriada para o ensino do Braille, onde a velocidade não é um ponto crítico para

a sua utilização. Outro problema encontrado, é com relação ao ruído produzido pelo protótipo, que pode causar desconforto quando utilizado por um longo tempo.

O software apresentou limitações com relação a conversão de textos onde todas as letras são maiúscula e para números com mais de um dígito. Para a representação de uma palavra onde todas as letras são maiúsculas, dois símbolos antecedendo a palavra indicam essa característica, mas para isto é necessária uma análise sobre toda a palavra. O mesmo ocorre com relação a números com mais de um dígito, onde um símbolo adicionado representa que os próximos dígitos que seguem serão números.

Os trabalhos futuros podem ser divididos em duas partes. A primeira com relação ao software, realizando melhoramentos na codificação de mais símbolos e o tratamento de palavras com todas as letras maiúsculas e números com mais de um dígito, e na interação com um sistema sintetizador de voz. A segunda com relação ao *hardware*, com a montagem completa de uma linha braile e calibração do trilho de deslocamento, inserção de sensores de início e fim de carro, com a miniaturização das travas mecânicas, dos ativadores e do trilho de deslocamento, com a diminuição do ruído, o aumento da velocidade de ativação, bem como a validação completa do sistema que foi obtida em partes no trabalho.

Uma possível aplicação para o dispositivo seria a sua implantação em bibliotecas. Em conjunto a uma câmera e um sistema de reconhecimento caracteres, pode-se representar em Braille qualquer material escrito.

# Apêndice A

## Código fonte de conversão

```
public class converte_Caracter {

    static int m1 = 2;
    static int m2 = 4;
    static int m3 = 8;
    //nesta classe vamos fzer a conversão do caracter para
    //código binário;

    public static Character Converte(char tica) {
        Character ai = new Character();
        //Aqui verifico se o caracter é Maiusculo.
        if (Character.isUpperCase(tica)){
            ai.setExisteColsAnt(true);
            ai.setColunaAntecessora1(0);
            ai.setColunaAntecessora2(m1+m3);
        }
        //Verifico se é Número
        if (Character.isDigit(tica)){
            ai.setExisteColsAnt(true);
            ai.setColunaAntecessora1(m3);
            ai.setColunaAntecessora2(m1+m2+m3);
        }
        //Aqui transformo a letra em minusculo
        tica=Character.toLowerCase(tica);

        switch (tica) {
            case 'a':
                ai.setColunaA(m1);
                ai.setColunaB(0);
                break;
        }
    }
}
```

```
case 'b':
    ai.setColunaA(m1+m2);
    ai.setColunaB(0);
    break;

case 'c':
    ai.setColunaA(m1);
    ai.setColunaB(m1);
    break;

case 'd':
    ai.setColunaA(m1);
    ai.setColunaB(m1+m2);
    break;

case 'e':
    ai.setColunaA(m1);
    ai.setColunaB(m2);
    break;

case 'f':
    ai.setColunaA(m1+m2);
    ai.setColunaB(m1);
    break;

case 'g':
    ai.setColunaA(m1+m2);
    ai.setColunaB(m1+m2);
    break;

case 'h':
    ai.setColunaA(m1+m2);
    ai.setColunaB(m2);
    break;

case 'i':
    ai.setColunaA(m2);
    ai.setColunaB(m1);
    break;

case 'j':
    ai.setColunaA(m2);
    ai.setColunaB(m1+m2);
    break;
```

```
case 'k':
    ai.setColunaA(m1+m3);
    ai.setColunaB(0);
    break;

case 'l':
    ai.setColunaA(m1+m2+m3);
    ai.setColunaB(0);
    break;

case 'm':
    ai.setColunaA(m1+m3);
    ai.setColunaB(m1);
    break;

case 'n':
    ai.setColunaA(m1+m3);
    ai.setColunaB(m1+m2);
    break;

case 'o':
    ai.setColunaA(m1+m3);
    ai.setColunaB(m2);
    break;

case 'p':
    ai.setColunaA(m1+m2+m3);
    ai.setColunaB(m1);
    break;

case 'q':
    ai.setColunaA(m1+m2+m3);
    ai.setColunaB(m1+m2);
    break;

case 'r':
    ai.setColunaA(m1+m2+m3);
    ai.setColunaB(m2);
    break;

case 's':
    ai.setColunaA(m2+m3);
    ai.setColunaB(m1);
    break;
```

```
case 't':
    ai.setColunaA(m2+m3);
    ai.setColunaB(m1+m2);
    break;

case 'u':
    ai.setColunaA(m1+m3);
    ai.setColunaB(m3);
    break;

case 'v':
    ai.setColunaA(m1+m2+m3);
    ai.setColunaB(m3);
    break;

case 'w':
    ai.setColunaA(m2);
    ai.setColunaB(m1+m2+m3);
    break;

case 'y':
    ai.setColunaA(m1+m3);
    ai.setColunaB(m1+m2+m3);
    break;

case 'x':
    ai.setColunaA(m1+m3);
    ai.setColunaB(m1+m3);
    break;

case 'z':
    ai.setColunaA(m1+m3);
    ai.setColunaB(m2+m3);
    break;
```

// Números

```
case '0':
    ai.setColunaA(m2);
    ai.setColunaB(m1+m2);
    break;

case '1':
    ai.setColunaA(m1);
    ai.setColunaB(0);
```



```
        break ;

    case '2':
        ai . setColunaA (m1+m2);
        ai . setColunaB (0);
        break ;

    case '3':
        ai . setColunaA (m1);
        ai . setColunaB (m1);
        break ;

    case '4':
        ai . setColunaA (m1);
        ai . setColunaB (m1+m2);
        break ;

    case '5':
        ai . setColunaA (m1);
        ai . setColunaB (m2);
        break ;

    case '6':
        ai . setColunaA (m1+m2);
        ai . setColunaB (m1);
        break ;

    case '7':
        ai . setColunaA (m1+m2);
        ai . setColunaB (m1+m2);
        break ;

    case '8':
        ai . setColunaA (m1+m2);
        ai . setColunaB (m2);
        break ;

    case '9':
        ai . setColunaA (m2);
        ai . setColunaB (m1);
        break ;

    case 'é':
        ai . setColunaA (m1+m2+m3);
        ai . setColunaB (m1+m2+m3);
```

```

        break ;

    case 'á':
        ai . setColunaA (m1+m2+m3);
        ai . setColunaB (m2+m3);
        break ;

    case 'è':
        ai . setColunaA (m2+m3);
        ai . setColunaB (m1+m3);
        break ;
    case 'à':
        ai . setColunaA (m1+m2);
        ai . setColunaB (m1+m3);
        break ;

    case 'â':
        ai . setColunaA (m1);
        ai . setColunaB (m3);
        break ;

    case 'ê':
        ai . setColunaA (m1+m2);
        ai . setColunaB (m3);
        break ;
    case 'ç':
        ai . setColunaA (m1+m2+m3);
        ai . setColunaB (m1+m3);
        break ;

    case 'ú':
        ai . setColunaA (m2+m3);
        ai . setColunaB (m1+m2+m3);
        break ;

    case 'ì':
        ai . setColunaA (m1);
        ai . setColunaB (m1+m3);
        break ;
    case 'ô':
        ai . setColunaA (m1);
        ai . setColunaB (m1+m2+m3);
        break ;

    case 'ù':

```

```

        ai . setColunaA (m1);
        ai . setColunaB (m2+m3);
        break ;

    case 'ï':
        ai . setColunaA (m1+m2);
        ai . setColunaB (m1+m2+m3);
        break ;
    case 'ü':
        ai . setColunaA (m1+m2);
        ai . setColunaB (m2+m3);
        break ;

    case 'õ':
        ai . setColunaA (m2);
        ai . setColunaB (m1+m3);
        break ;

    case 'ò':
        ai . setColunaA (m2);
        ai . setColunaB (m1+m2+m3);
        break ;
    case 'í':
        ai . setColunaA (m3);
        ai . setColunaB (m1);
        break ;

    case 'ã':
        ai . setColunaA (m3);
        ai . setColunaB (m1+m2);
        break ;

    case 'ó':
        ai . setColunaA (m3);
        ai . setColunaB (m1+m3);
        break ;
    case ',':
        ai . setColunaA (m2);
        ai . setColunaB (0);
        break ;

    case ';':
        ai . setColunaA (m2+m3);
        ai . setColunaB (0);
        break ;

```

```

case '!':
    ai.setColunaA(m2+m3);
    ai.setColunaB(m2);
    break;
case '*':
    ai.setColunaA(m3);
    ai.setColunaB(m2);
    break;

case '$':
    ai.setColunaA(0);
    ai.setColunaB(m2+m3);
    break;

case ':':
    ai.setColunaA(m2);
    ai.setColunaB(m2);
    break;
case '?':
    ai.setColunaA(m2);
    ai.setColunaB(m3);
    break;

case '#':
    ai.setColunaA(m3);
    ai.setColunaB(m1+m2+m3);
    break;

case '.':
    ai.setColunaA(m3);
    ai.setColunaB(0);
    break;
case '-':
    ai.setExisteColsAnt(true);
    ai.setColunaAntecessora1(m3);
    ai.setColunaAntecessora2(m3);
    ai.setColunaA(m3);
    ai.setColunaB(m3);
    break;

case '"':
    ai.setColunaA(m2+m3);
    ai.setColunaB(m3);
    break;

```

```

case '(' :
    ai . setExisteColsAnt ( true );
    ai . setColunaAntecessora1 ( m1+m2 );
    ai . setColunaAntecessora2 ( m3 );
    ai . setColunaA ( m3 );
    ai . setColunaB ( 0 );
    break ;
case ')' :
    ai . setExisteColsAnt ( true );
    ai . setColunaAntecessora1 ( 0 );
    ai . setColunaAntecessora2 ( m3 );
    ai . setColunaA ( m3 );
    ai . setColunaB ( m1+m2 );
    break ;

case '[' :
    ai . setColunaA ( m1+m2+m3 );
    ai . setColunaB ( m2+m3 );
    break ;

case ']' :
    ai . setColunaA ( m2+m3 );
    ai . setColunaB ( m1+m2+m3 );
    break ;
case '&' :
    ai . setColunaA ( m1+m2+m3 );
    ai . setColunaB ( m1+m3 );
    break ;

case '/' :
    ai . setExisteColsAnt ( true );
    ai . setColunaAntecessora1 ( m2 );
    ai . setColunaAntecessora2 ( 0 );
    ai . setColunaA ( 0 );
    ai . setColunaB ( m3 );
    break ;

case '|' :
    ai . setColunaA ( 0 );
    ai . setColunaB ( m1+m2+m3 );
    break ;
case '%' :
    ai . setExisteColsAnt ( true );
    ai . setColunaAntecessora1 ( 0 );

```

```

        ai . setColunaAntecessora2 (m1+m2+m3);
        ai . setColunaA (m3);
        ai . setColunaB (m2+m3);
        break ;

    case '+' :
        ai . setColunaA (m2+m3);
        ai . setColunaB (m2);
        break ;

    case '=' :
        ai . setColunaA (m2+m3);
        ai . setColunaB (m2+m3);
        break ;
    case ' ' :
        ai . setColunaA (0);
        ai . setColunaB (0);
        break ;

    case '\n' :
        ai . setColunaA (0);
        ai . setColunaB (0);
        break ;

    case '\'' :
        ai . setColunaA (m3);
        ai . setColunaB (0);
        break ;
    }
    return ai ;
}
}

```

## Apêndice B

### Código fonte de controle dos motores

```
import parport.ParallelPort;

private static long tempoEntrePassos = 500;

public class OperandoMotor {
    public static void marca_PontoColuna(int valorLinhas) {
        ParallelPort porta = new ParallelPort(0x378);
        int valorSaida = valorLinhas;
        int tempo = 500;
        porta.write(0 + valorSaida);
        try {
            Thread.sleep(tempo);
        } catch (Exception e) {
        }

        porta.write(1 + valorSaida);
        try {
            Thread.sleep(tempo / 2);
        } catch (Exception e) {
        }

        porta.write(0 + valorSaida);
        try {
            Thread.sleep(tempo / 2);
        } catch (Exception e) {
        }
    }

    public static void desloca_Motor_Mesa_Direita() {
        int Deslocamento = 100;
        ParallelPort porta = new ParallelPort(0x378);
```

```

int passo = 128;
for (int i = 0; i < Deslocamento; i++) {
    passo = passo / 2;
    passo = passo < 16 ? 128 : passo;
    porta.write(passo);
    try {
        Thread.sleep(tempoEntrePassos);
    } catch (Exception e) {
    }
}
}

public static void desloca_Motor_Mesa_Esquerda() {
    int Deslocamento = 100;
    ParallelPort porta = new ParallelPort(0x378);
    int passo = 16;
    for (int i = 0; i < Deslocamento; i++) {
        passo = passo * 2;
        passo = passo > 128 ? 16 : passo;
        porta.write(passo);
        try {
            Thread.sleep(tempoEntrePassos);
        } catch (Exception e) {
        }
    }
}
}

```



# Apêndice C

## Código fonte da classe Controller

```
package Controle ;

import Modelo.Character ;
import java.util.ArrayList ;
import java.util.Stack ;

public class Controller {

    private static ArrayList<Character> listaCharacter =
    new ArrayList<Character>();
    private static Stack<Character> pilhaExibicao =
    new Stack();
    private static int numeroColunasDisplay = 1;
    private static int posicaoNaListaCharacter = 0;

    public static void codificaTxt(String entrada){

        ArrayList<Character> listaCharacter =
        new ArrayList<Character>();
        listaCharacter =
        TratadorString.StringToVetorChar(entrada);

        listaCharacter =
        TratadorVetorChar.VetorCharToVetorCharacter(listaCharacter);
    }

    public static void enviaProximoConjunto(){

        //são removidos os pontos ativados da display.
```

```

for(int k=0;k<pilhaExibicao.size();k++){

Caracter sai = pilhaExibicao.pop();
OperandoMotor.desloca_Motor_Mesa_Esquerda();
OperandoMotor.marca_PontoColuna(sai.getColunaB());
OperandoMotor.desloca_Motor_Mesa_Esquerda();
OperandoMotor.marca_PontoColuna(sai.getColunaA());

if(sai.isExisteColsAnt()){

OperandoMotor.desloca_Motor_Mesa_Esquerda();
OperandoMotor.marca_PontoColuna(sai.getColunaAntecessora2());
OperandoMotor.desloca_Motor_Mesa_Esquerda();
OperandoMotor.marca_PontoColuna(sai.getColunaAntecessora1());

}
}

//São adicionados pontos em conformidade ao número de colunas
for(int i=0;i<numeroColunasDisplay;i++){

Caracter entra = listaCaracter.get(posicaoNaListaCaracter);
pilhaExibicao.push(entra);
posicaoNaListaCaracter++;

if(entra.isExisteColsAnt()){

OperandoMotor.marca_PontoColuna(entra.getColunaAntecessora1());
OperandoMotor.desloca_Motor_Mesa_Direita();
OperandoMotor.marca_PontoColuna(entra.getColunaAntecessora2());
OperandoMotor.desloca_Motor_Mesa_Direita();

}

OperandoMotor.marca_PontoColuna(entra.getColunaA());
OperandoMotor.desloca_Motor_Mesa_Direita();
OperandoMotor.marca_PontoColuna(entra.getColunaB());
OperandoMotor.desloca_Motor_Mesa_Direita();

}
}
}

```

# Referências Bibliográficas

- [1] ARAÚJO, A. Ana Paula de. *Alfabeto Braille*. Disponível em: <http://www.infoescola.com/portugues/braille/>. Acessado em: 03 Julho 2012.
- [2] ABNT - Associação Brasileira de Normas Técnicas. *Acessibilidade a edificações, mobiliário, espaços e equipamentos urbanos*. Rio de Janeiro - RJ, 2004.
- [3] SYDNEY. *Resources for Students with Disabilities*. Disponível em: <http://resourcesswd.blogspot.com.br/2010/10/braille-terminal.html>. Acessado em: 03 Julho 2012.
- [4] BRAGA, N. C. *Como funciona o solenóide*. Disponível em: <http://www.newtoncbraga.com.br/index.php/como-funciona/3890-mec095.html>. Acessado em: 03 Julho 2012.
- [5] AG3WS. *Como Funciona o Relé?* Disponível em: <http://eletronicos.etc.br/como-funciona-o-rele/>. Acessado em: 03 Julho 2012.
- [6] YEH,F. H. et al. Application of an adaptive network based fuzzy inference system for the optimal design of a chinese braille display. *Biomedical Engineering Applications, Basis and Communications*, v. 17, n. 1, 2005.
- [7] HAGA, Y. et al. Dynamic Braille display using SMA coil actuator and magnetic latch. *Sensors and Actuators A: Physical*, v. 119, n. 2, p. 316 – 322, 2005. ISSN 0924-4247. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0924424704007514>>.
- [8] METEC. *Fabricante de displays Braille*. Disponível em: <http://www.metec-ag.de/braillemodulb11.html>. Acessado em: 03 Julho 2012.

- [9] GLOBAL EYE. *Aplicações Piezo Elétricas - The Corporate Global Source Solution*. Disponível em: <http://www.global-eye.com.br/public/newsletter/1202/>. Acessado em: 08 Setembro 2012.
- [10] KATO, Y. et al. Sheet-type braille displays by integrating organic field-effect transistors and polymeric actuators. *IEEE TRANSACTIONS ON ELECTRON DEVICES*, v. 54, n. 2, p. 202–209, 2007.
- [11] MONTARDO, S. P.; PASSERINO, L. Inclusão social via acessibilidade digital: proposta de inclusão digital para pne. *E-Compós*, v. 8, p. 1–18, 2007. Disponível em: <<http://compos.org.br/seer/index.php/e-compos/article/view/144/145>. Acessado em: 16 de Outubro 2012>.
- [12] FREEDOM SCIENTIFIC. *Fabricante do software JAWS*. Disponível em: <http://www.freedomscientific.com/products/fs/jaws-product-page.asp>. Acessado em: 03 Julho 2012.
- [13] NÚCLEO DE COMPUTAÇÃO ELETRÔNICA - Universidade Federal do Rio de Janeiro. *Projeto Dosvox*. Disponível em: <http://intervox.nce.ufrj.br/dosvox/>. Acessado em: 16 Outubro 2012.
- [14] FREEDOM SCIENTIFIC. *Displays Braille Focus*. Disponível em: <http://www.freedomscientific.com/products/fs/focus-product-page.asp>. Acessado em: 03 Julho 2012.
- [15] WORLD HEALTH ORGANIZATION - Organização Mundial da Saúde (OMS). *The WHO Family of International Classifications*. Disponível em: <http://www.who.int/classifications/en/>. Acessado em: 08 Setembro 2012.
- [16] CONDE, A. J. M. *Definindo a Cegueira e a Visão Subnormal*. Disponível em: <http://www.ibr.gov.br/?itemid=94>. Acessado em: 03 Julho 2012.
- [17] IBGE - Instituto Brasileiro de Geografia e Estatística. *Comunicação Social 27 de abril de 2012 - Dados sobre o resultado do Censo realizado em 2010*. Disponível em:

- [http://www.ibge.gov.br/home/presidencia/noticias/noticia\\_visualiza.php?id\\_noticia=2125&id\\_pagina=1](http://www.ibge.gov.br/home/presidencia/noticias/noticia_visualiza.php?id_noticia=2125&id_pagina=1). Acessado em: 03 Julho 2012.
- [18] BAPTISTA, J. A. L. S. *A invenção do Braille e a sua Importância na Vida dos Cegos*. 1. ed. R. Sacadura Cabral, 89-91 A 1495 Cruz Quebrada: Gráfica 2000, 2000.
- [19] LARATEC. *Display Braille*. Disponível em: <http://www.laratec.org.br/DisplayBraille.html>. Acessado em: 03 Julho 2012.
- [20] QUEIROZ, M. A. de. *O que é um Display Braille*. Disponível em: <http://acessibilidadelegal.com/33-display-braille.php>. Acessado em: 03 Julho 2012.
- [21] HUMANWARE. *Display Braille Brailliant*. Disponível em: [http://www.humanware.com/en-usa/products/blindness/braille\\_displays](http://www.humanware.com/en-usa/products/blindness/braille_displays). Acessado em: 03 Julho 2012.
- [22] SRISKANTHAN, N. et al. Braille display terminal for personal computers. *IEEE Trans. Consumer Electron*, Singapore, v. 36, n. 2, p. 121–128, 1990.
- [23] NATAL, G. S. *Nanoposicionamento de precisão por controle adaptativo binário de atuadores piezoelétricos*. Dissertação (Dissertação de Mestrado) — COPPE – Universidade Federal do Rio de Janeiro, Rio de Janeiro - RJ, Maio 2008.
- [24] KRASNOW, B. *Piezoelectric Braille system with PC parallel port control and HV transistors*. Disponível em: <http://www.youtube.com/watch?v=Ss-Fux86doQ>. Acessado em: 03 Julho 2012.
- [25] GONÇALVES, L. *Patente cearense disponibiliza tablet em braile*. Disponível em: <http://diariodonordeste.globo.com/materia.asp?codigo=1065041>. Acessado em: 17 Outubro 2012.
- [26] SHAHINPOOR, M. et al. Ionic polymer-metal composites (IPMCs) as biomimetic sensors, actuators and artificial muscles-A review. *Smart Mater. Struct.*, v. 7, n. 6, 1998.
- [27] FITZGERALD, A. E.; KINGSLEY, C. Jr.; UMANS, S.D. *Máquinas Elétricas*. 6. ed. Porto Alegre: Bookman, 2006.

- [28] EPSON. *Informações sobre impressora Epson FX-2180*. Disponível em: [http://www.epson.com/cmc\\_upload/0/000/174/653/specs\\_lq2180.pdf?BV\\_UseBVCookie=yes](http://www.epson.com/cmc_upload/0/000/174/653/specs_lq2180.pdf?BV_UseBVCookie=yes). Acessado em: 03 Julho 2012.
- [29] ORACLE. *Java*. Disponível em: <http://www.oracle.com/br/technologies/java/index.html>. Acessado em: 03 Julho 2012.
- [30] PARALLELPORT. *How To Use the Parallel Port with Matlab (via JAVA)*. Disponível em: <http://www.ioi.knaw.nl/heimel/computers/parport/>. Acessado em: 15 Outubro 2012.
- [31] MESSIAS, A. R. *Introdução à Porta Paralela*. Disponível em: <http://www.rogercom.com/pparalela/introducao.htm>. Acessado em: 03 Julho 2012.