

**UNIVERSIDADE FEDERAL DE SANTA CATARINA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA  
COMPUTAÇÃO**

**Carlos Eduardo Gonçalves**

**TADEU - SISTEMA SIMULADOR DE TECLADO  
PARA DEFICIENTES FÍSICOS**

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação

João Bosco da Mota Alves


Florianópolis Novembro 2001

# TADEU – SISTEMA SIMULADOR DE TECLADO PARA DEFICIENTES FÍSICOS

Carlos Eduardo Gonçalves

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação Área de Concentração Sistemas de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

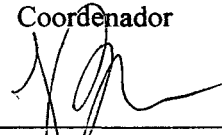
Banca Examinadora



---

Prof. Dr. Fernando A. Ostuni Gauthier

Coordenador



---

Prof. Dr. João Bosco da Mota Alves

Orientador



---

Prof. Dr. Mauro Roisenberg



---

Prof. Dr. Luiz Fernando Jacinto Maia



---

Prof. Alberto Angel Mazzoni

“Aquele que não sabe e não sabe que não sabe é um tolo, evite-o.

Aquele que não sabe e sabe que não sabe é um estudioso, ensine-o.

Aquele que sabe e não sabe que sabe é um sonâmbulo, acorde-o

Aquele que sabe e sabe que sabe é um sábio, siga-o.”

(Provérbio chinês)

Aos meus amados pais Carlos Alberto e Regina Estela, fontes inesgotáveis de inspiração e sabedoria.

Agradeço aos amigos que em diversas oportunidades compartilharam comigo suas idéias e experiências, em especial à Alberto, Elizabeth e à meu amigo e orientador João Bosco.

# SUMÁRIO

<b>SUMÁRIO</b> .....	<b>1</b>
<b>LISTA DE FIGURAS</b> .....	<b>3</b>
<b>LISTA DE TABELAS</b> .....	<b>4</b>
<b>RESUMO</b> .....	<b>5</b>
<b>ABSTRACT</b> .....	<b>6</b>
<b>1. INTRODUÇÃO</b> .....	<b>7</b>
<b>1.1 OS COMPUTADORES FRENTE A DIVERSIDADE HUMANA</b> .....	<b>7</b>
<b>1.2 TECNOLOGIAS ASSISTIVAS</b> .....	<b>8</b>
1.2.1. <i>Tecnologias alternativo/aumentativas de acesso a informação.</i> .....	9
1.2.2. <i>Tecnologias de acesso.</i> .....	9
1.2.3. <i>Tecnologias alternativo/aumentativas para a comunicação.</i> .....	9
1.2.4. <i>Tecnologias de mobilidade.</i> .....	9
1.2.5. <i>Tecnologias de controle ambiente.</i> .....	10
<b>1.3 TADEU</b> .....	<b>10</b>
1.3.1. <i>Objetivos.</i> .....	10
1.3.2. <i>Materiais e métodos.</i> .....	11
<b>1.4 ESTRUTURA DO TRABALHO</b> .....	<b>11</b>
<b>2. NECESSIDADES ESPECIAIS E AS BARREIRAS PARA INTERAÇÃO COM OS COMPUTADORES</b> .....	<b>13</b>
<b>2.1 CONCEITOS</b> .....	<b>13</b>
2.1.1. <i>Quem tem necessidades especiais?</i> .....	13
2.1.2. <i>Novas classificações.</i> .....	15
2.1.3. <i>Características das deficiências.</i> .....	17
<b>2.2 DEFININDO BARREIRA E ACESSIBILIDADE</b> .....	<b>18</b>
2.2.1. <i>Barreira.</i> .....	18
2.2.2. <i>Acessibilidade.</i> .....	18
<b>2.3 DEFICIÊNCIAS E AS BARREIRAS PARA ACESSIBILIDADE</b> .....	<b>18</b>
2.3.1. <i>Deficiência visual.</i> .....	18
2.3.2. <i>Deficiência auditiva.</i> .....	19
2.3.3. <i>Deficiência física.</i> .....	20
2.3.4. <i>Deficiência cognitiva e de linguagem.</i> .....	21
<b>3. ERGONOMIA DE INTERFACES ESPECIAIS</b> .....	<b>24</b>
<b>3.1 CONCEITOS</b> .....	<b>24</b>
3.1.1. <i>Ergonômica de interface homem - computador.</i> .....	24
3.1.2. <i>Avaliação ergonômica de IHC.</i> .....	24
3.1.3. <i>Critérios para qualidade ergonômica em IHC.</i> .....	25
<b>3.2 INTERFACES PARA PORTADORES DE NECESSIDADES ESPECIAIS</b> .....	<b>27</b>
3.2.1. <i>Interfaces especiais.</i> .....	27
3.2.2. <i>Características pertinentes para uma interface especial.</i> .....	28
3.2.3. <i>Interfaces multimodais.</i> .....	29
<b>4. CARACTERÍSTICAS GERAIS DOS SISTEMAS SIMULADORES DE TECLADO</b> .....	<b>31</b>
<b>4.1 SELEÇÃO DE TECLAS</b> .....	<b>31</b>
4.1.1. <i>Seleção direta.</i> .....	31
4.1.2. <i>Seleção indireta.</i> .....	32

<b>4.2</b>	<b>POSICIONAMENTO</b> .....	<b>33</b>
<b>4.3</b>	<b>RECURSOS ESPECIAIS</b> .....	<b>33</b>
4.3.1.	<i>Predição de palavras</i> .....	34
4.3.2.	<i>Emulação de mouse</i> .....	34
4.3.3.	<i>Múltiplos padrões de teclado</i> .....	34
<b>4.4</b>	<b>OS SISTEMAS SIMULADORES DE TECLADO PESQUISADOS</b> .....	<b>34</b>
4.4.1.	<i>UniControl 1.0</i> .....	34
4.4.2.	<i>SoftType</i> .....	35
4.4.3.	<i>TotiPM 1.0.38</i> .....	35
4.4.4.	<i>TcSoft</i> .....	36
4.4.5.	<i>Simulador de Teclado</i> .....	37
<b>5.</b>	<b>DESENVOLVIMENTO DA INTERFACE</b> .....	<b>38</b>
<b>5.1</b>	<b>OS ESTUDOS DE CASO</b> .....	<b>38</b>
5.1.1.	<i>O perfil do grupo estudado</i> .....	38
5.1.2.	<i>As constatações</i> .....	39
<b>5.2</b>	<b>AVALIAÇÃO DAS INTERFACES DOS SIMULADORES DE TECLADO</b> .....	<b>40</b>
5.2.1.	<i>Itens observados durante a avaliação</i> .....	40
5.2.2.	<i>Resultados da avaliação</i> .....	40
<b>5.3</b>	<b>A PROPOSTA DE INTERFACE DO SISTEMA</b> .....	<b>42</b>
<b>6.</b>	<b>DESENVOLVIMENTO DO SISTEMA</b> .....	<b>45</b>
<b>6.1</b>	<b>A MODELAGEM UTILIZADA</b> .....	<b>45</b>
<b>6.2</b>	<b>VARREDURA AUTOMÁTICA DE TECLAS</b> .....	<b>46</b>
6.2.1.	<i>Varredura de linhas</i> .....	48
6.2.2.	<i>Varredura de colunas</i> .....	49
6.2.3.	<i>Varredura especial para predição de palavras</i> .....	50
<b>6.3</b>	<b>SIMULANDO A DIGITAÇÃO DAS TECLAS</b> .....	<b>51</b>
<b>6.4</b>	<b>ACIONANDO O MOUSE</b> .....	<b>54</b>
6.4.1.	<i>Utilizando os Hooks</i> .....	54
6.4.2.	<i>Macros de mouse</i> .....	55
<b>6.5</b>	<b>PREDIÇÃO DE PALAVRAS</b> .....	<b>57</b>
6.5.1.	<i>Filtrando as palavras</i> .....	57
6.5.2.	<i>Digitando as palavras</i> .....	58
<b>6.6</b>	<b>COMUNICAÇÃO COM HARDWARE</b> .....	<b>58</b>
6.6.1.	<i>Porta Serial</i> .....	58
6.6.2.	<i>Microfone</i> .....	59
<b>7.</b>	<b>DESENVOLVENDO DISPOSITIVOS ACIONADORES</b> .....	<b>61</b>
<b>7.1</b>	<b>DISPOSITIVOS DE ENTRADA DE DADOS</b> .....	<b>61</b>
7.1.1.	<i>Dispositivos de acesso direto</i> .....	61
7.1.2.	<i>Dispositivos de acesso indireto</i> .....	61
<b>7.2</b>	<b>COMUNICAÇÃO SERIAL</b> .....	<b>62</b>
7.2.1.	<i>O padrão RS232</i> .....	62
7.2.2.	<i>A porta serial</i> .....	63
<b>7.3</b>	<b>COMUNICAÇÃO COM O SISTEMA</b> .....	<b>64</b>
7.3.1.	<i>Conexão</i> .....	64
7.3.2.	<i>Driver</i> .....	65
7.3.3.	<i>Princípio geral do acionamento</i> .....	65
<b>7.4</b>	<b>DISPOSITIVO ACIONADOR POR PROXIMIDADE</b> .....	<b>66</b>
<b>8.</b>	<b>RESULTADOS</b> .....	<b>69</b>
<b>9.</b>	<b>CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS</b> .....	<b>70</b>
	<b>REFERÊNCIAS BIBLIOGRAFICAS</b> .....	<b>72</b>
	<b>ANEXO</b> .....	<b>75</b>

## LISTA DE FIGURAS

FIGURA 1 INTERAÇÃO ENTRE OS COMPONENTES APRESENTADOS PELA ICIDH-2.....	17
FIGURA 2 SENTENÇAS PICTOGRÁFICAS NAS LINGUAGENS PCS E MAKATON.....	22
FIGURA 3 MODELO CLÁSSICO DE INTERAÇÃO HOMEM - COMPUTADOR.....	25
FIGURA 4 EXEMPLO DO FUNCIONAMENTO DA VARREDURA GRUPAL.....	32
FIGURA 5 POSICIONAMENTO EM PARTE DA TELA.....	33
FIGURA 6 JANELA DO UNICONTROL.....	35
FIGURA 7 SIMULADOR DE TECLADO SOFTTYPE.....	35
FIGURA 8 JANELA DO SIMULADOR DE TECLADO TOTIPM.....	36
FIGURA 9 INTERFACE DO TCSOFT.....	36
FIGURA 10 DISPOSIÇÃO DOS TECLADOS EM CAMADAS.....	42
FIGURA 11 PADRÃO DE TECLADO QWERTY.....	43
FIGURA 12 INTERFACE DO EDITOR DE PROPRIEDADES.....	44
FIGURA 13 COMPOSIÇÃO DO CÓDIGO DE VARREDURA.....	47
FIGURA 14 TRANSIÇÃO ENTRE OS MODOS DE VARREDURA.....	51
FIGURA 15 ENCADEAMENTO DAS FUNÇÕES FILTRO IMPLEMENTADA A PARTIR DO WINDOWS 3.1.....	56
FIGURA 16 PINAGEM DAS PORTAS SERIAIS 25 PINOS E 9 PINOS.....	63
FIGURA 17 ACIONADOR FEITO COM UMA CHAVE ELÉTRICA.....	66
FIGURA 18 DIAGRAMA DE BLOCOS DO ACIONADOR POR PROXIMIDADE.....	67
FIGURA 19 FILTRAGEM DOS MOVIMENTOS INVOLUNTÁRIOS PELO ACIONADOR.....	68



## LISTA DE TABELAS

TABELA 1 ESCOPO DA ERGONOMIA.....	24
TABELA 2 DISTRIBUIÇÃO DOS CARACTERES GRAFÊMICOS. ....	37
TABELA 3 PADRÃO PARA A CONEXÃO SERIAL COM O SISTEMA. ....	65

## RESUMO

Este trabalho tem como proposta o estudo da diversidade humana, no sentido de identificar necessidades especiais ocasionadas por deficiências físicas, e oferecer soluções tecnológicas que permitam as pessoas portadoras de tais necessidades, transpor eventuais barreiras encontradas durante a interação com sistemas computacionais.

A tecnologia proposta pelo trabalho é um sistema simulador de teclado, destinado ao grupo de usuários portadores de deficiências motoras (e.g. tetraplegia, paralisia cerebral), incapazes de utilizar plenamente os teclados convencionais.

O trabalho parte da exposição dos conceitos sobre as necessidades especiais associadas com determinadas deficiências físicas, relacionando as principais tecnologias encontradas atualmente na área de informática para atender estas necessidades. Abrange também os principais conceitos referentes a acessibilidade em informática, e aos cuidados ergonômicos relativos a concepção de interfaces para deficientes físicos, bases para o desenvolvimento do sistema simulador de teclado.

Este trabalho mostra a importância do desenvolvimento de sistemas computacionais mais inteligentes e adaptáveis, capazes de atender a um perfil mais heterogêneo de usuários, permitindo uma maior qualidade na interação com os sistemas, tanto para os usuários portadores de necessidades especiais, quanto para os usuários comuns.

## ABSTRACT

This work has the proposition of study the human diversity, in the meaning to identify special needs occasioned by physical deficiencies, and to offer technological solutions which permit the people that have those needs, to cross eventual barriers found while the interaction with computing systems.

The technology proposed by this work is an onscreen keyboard system, destined for the users' group that have motor deficiencies (e.g. tetra palsy, cerebral palsy), disable to use fully the conventional keyboards.

The work starts from exposition of the concepts about the special needs associated with determined physical deficiencies, making the relation with the main technologies find actually in the computing area to attend those needs. It ranges too the main concepts about the accessibility in computing, and the ergonomics' cares relative with the interface design for disabled people bases for the onscreen keyboard development.

This work shows the importance of development of more intelligent and adaptable computing systems, able to attend a more heterogeneous profile of users, permitting a greater quality in the interaction with the systems, both for the users with special needs and the common users.

# 1. INTRODUÇÃO

## 1.1 OS COMPUTADORES FRENTE A DIVERSIDADE HUMANA

Segundo um estudo realizado na escola de medicina da Universidade de Stanford, apresentado por BIELER (1), é possível representar proporcionalmente a população do planeta ao final do século XX, utilizando um grupo de 100 pessoas, segundo o estudo haveriam:

- 57 asiáticos, 21 europeus, 14 dos hemisférios ocidentais norte e sul e 8 africanos;
- 52 mulheres e 48 homens;
- 70 não descendentes de europeus e 30 descendentes de europeus;
- 70 não cristãos e 30 cristãos;
- 89 heterossexuais e 11 homossexuais;
- 6 teriam se apossado de 59% das riquezas mundiais e estes 6 viveriam nos Estados Unidos da América;
- 80 viveriam em habitações de má qualidade;
- 70 seriam analfabetos;
- 50 passariam fome;
- 1 estaria para morrer e 1 estaria para nascer;
- 1 teria educação em nível universitário;
- 1 teria um computador.

Ao nos depararmos com estas estatísticas, que representam o mundo onde vivemos de maneira tão reduzida, percebemos melhor a diversidade humana e a necessidade de igualdade de condições para todos os seres humanos.

Se transferirmos esta diversidade humana para dentro da computação, constataremos que a maioria dos sistemas computacionais e dispositivos de hardware

não estão preparados para lidar com espectro tão diverso de usuários, limitando-se a atender um único tipo de usuário, um usuário padrão, cujo perfil pode ser descrito como o de um indivíduo do sexo masculino com 25 anos de idade, anglo-saxão, formado em ciência da computação e viciado em tecnologia, NEWELL (13).

Esta incapacidade dos sistemas computacionais, em atender uma faixa mais ampla de pessoas, se mostra especialmente limitante, ao considerarmos o grupo de usuários formado pelos portadores de deficiências físicas, que tem uma relativa dificuldade em encontrar tecnologias de computação capazes de atendê-los.

Tomando como referência inovações tecnológicas como a caneta esferográfica e o controle remoto, ambas criadas originalmente para portadores de deficiência e que hoje estão integradas no cotidiano de todos nós, percebemos como os deficientes físicos podem ajudar os computadores a se tornarem mais eficientes, NEWELL (13).

Deste modo é preciso oferecer aos portadores de deficiência física, considerados de forma mais ampla como portadores necessidades especiais, tecnologias capazes de auxiliá-los em tarefas onde os dispositivos e sistemas tradicionais fracassam.

Algumas destas tecnologias, podem ser classificadas como tecnologias assistivas para computação.

## 1.2 TECNOLOGIAS ASSISTIVAS

As tecnologias assistivas, também conhecidas como ajudas técnicas, são constituídas por qualquer dispositivo ou artefato capaz de facilitar a execução de uma ou mais tarefas. Dentro deste amplo espectro, podemos encontrar uma cadeira de rodas, sistemas de reconhecimento de voz e até mesmo uma faca elétrica.

Uma das definições formais para as tecnologias assistivas é de que são dispositivos que correspondem a qualquer item, peça de equipamento ou produto, seja ele adquirido comercialmente ou não, modificado ou construído, que é utilizado para aumentar, manter ou aperfeiçoar capacidades funcionais de indivíduos com deficiências físicas, DATI (5).

As tecnologias assistivas podem ser aplicadas praticamente em qualquer área de atuação, oferecendo a pessoas portadoras de necessidades especiais a possibilidade de

executar tarefas de forma mais agradável e produtiva. Com base em KOON (9), dentro da área da computação, podemos dividir as tecnologias assistivas em 5 grupos distintos:

#### 1.2.1. Tecnologias alternativo/aumentativas de acesso a informação.

Grupo formado por tecnologias destinadas a pessoas com limitações sensoriais, e que por este motivo necessitam de sistemas especializados para superar tais déficits. Conta com sistemas de reconhecimento de voz, multimídia, aplicações de computação móvel e conectividade.

#### 1.2.2. Tecnologias de acesso.

Grupo caracterizado por sistemas adaptados, capazes de oferecer a usuários portadores de deficiências físicas a possibilidade de interação com computadores. Dentre as tecnologias que se destacam neste grupo estão os magnificadores de tela, sistemas simuladores de teclado, telas sensíveis ao toque, linhas Braille e sintetizadores de voz.

#### 1.2.3. Tecnologias alternativo/aumentativas para a comunicação.

São as tecnologias destinadas as pessoas que não conseguem estabelecer uma comunicação plena através do código verbal - oral. A diferença básica entre a comunicação alternativa e aumentativa, é que comunicação alternativa não se utiliza a linguagem falada como base para a comunicação, já a comunicação aumentativa conta com recursos de apoio para permitir a comunicação verbal.

#### 1.2.4. Tecnologias de mobilidade.

Grupo formado por tecnologias de computação capazes de auxiliar uma pessoa a se movimentar dentro de um determinado ambiente. Muitas destas tecnologias possuem forte apelo futurista, vinculadas a áreas da medicina permitindo o desenvolvimento de implantes cibernéticos e próteses inteligentes.

### 1.2.5. Tecnologias de controle ambiente.

Todas as tecnologias que permitem o controle de um ambiente, como o abrir e fechar de portas e o acionamento de eletrodomésticos. As aplicações em realidade virtual são destaque nesta área.

Em geral as tecnologias assistivas procuram atender as necessidades específicas de um determinado perfil de deficientes físicos, isto porque os custos envolvidos no desenvolvimento de tecnologias personalizadas, tornam esta abordagem bastante restritiva.

O desenvolvimento de tecnologias assistivas com base nas necessidades de um único indivíduo é um ideal a ser alcançado, principalmente nos casos onde várias tecnologias precisam ser aplicadas simultaneamente.

## 1.3 TADEU

Este trabalho tem a finalidade de desenvolver uma Tecnologia Assistiva Desenvolvida em Universidade (*Tadeu*), caracterizada por um sistema simulador de teclado, destinado as pessoas portadoras de deficiências motoras, que em virtude de suas limitações, encontram dificuldade em utilizar os teclados convencionais.

### 1.3.1. Objetivos.

O trabalho possui como principais objetivos:

- Desenvolver um sistema simulador de teclado que agregue os principais recursos oferecidos por sistemas do gênero, capazes de auxiliar os usuários nas tarefas de digitação e acesso à informação;
- Oferecer uma proposta diferenciada de interface, em relação as tradicionalmente aplicadas em sistemas simuladores de teclado, na tentativa de oferecer maior ergonomia aos usuários em questão;

- Embora o foco principal do trabalho esteja voltado ao desenvolvimento do sistema simulador de teclado, o mesmo possui um projeto complementar de hardware, que visa a criação de um dispositivo que possa ser utilizado em conjunto com o sistema, ampliando as possibilidades de utilização do mesmo.

### 1.3.2. Materiais e métodos.

O sistema trabalha sobre plataforma *Microsoft Windows* ®, atendendo as versões 95 e superiores. Todo o software foi desenvolvido através da linguagem Object Pascal, utilizando o ambiente *Borland Delphi 3* ®, com a utilização de funções residentes na *Microsoft Win32* ® Application Program Interface.

O desenvolvimento da interface do sistema, foi feito a partir da avaliação heurística dos sistemas simuladores de teclado *TcSoft* desenvolvido pelo E.U.P. Diplomatura de Informática de Córdoba, *Simulador de Teclado* desenvolvido em parceria pelo grupo de Tecnologias de Reabilitação da Universidade Politécnica de Madri e pelo grupo CIES/EDUCOM/FACED da Universidade Federal do Rio Grande do Sul, *UniControl* © Ricardo Lastra Olsen, *SoftType* © George Leotti e *TotiPm* © Jordi Lagares Roset.

Foram realizados estudos de caso junto a alunos portadores de paralisia cerebral da APAE de Florianópolis, a fim de observar as limitações dos mesmos durante a interação com computadores, o que permitiu o desenvolver a interface do sistema e determinar as formas de acionamento a serem contempladas.

## 1.4 ESTRUTURA DO TRABALHO

O capítulo 2 apresenta os principais conceitos relacionados com as necessidades especiais e as barreiras impostas pelas deficiências sensoriais, cognitivas e físicas para interação com sistemas computacionais.

O capítulo 3 abrange a utilização da ergonomia de IHC, no desenvolvimento de sistemas adequados as necessidades de usuários portadores de deficiências.



As características dos sistemas simuladores de teclado pesquisados durante o trabalho é o tema central do capítulo 4.

O capítulo 5 mostra a o desenvolvimento da interface do sistema simulador de teclado, com base em estudos de caso e avaliação ergonômica das interfaces dos sistemas estudados.

O capítulo 6 apresenta a implementação do sistema através da linguagem Object Pascal.

Por fim o capítulo 7 aborda o desenvolvimento de dispositivos acionadores capazes de operar o sistema, apresentando o acionador desenvolvido durante o trabalho.

## 2. NECESSIDADES ESPECIAIS E AS BARREIRAS PARA INTERAÇÃO COM OS COMPUTADORES

### 2.1 CONCEITOS

#### 2.1.1. Quem tem necessidades especiais?

Vejam os inicialmente algumas classificações dadas em 1980 pela OMS (*Organização Mundial da Saúde*), que fazem parte do documento conhecido como ICIDH (*International Classification of Impairments, Disabilities, and Handicap*), apresentadas aqui com base em MONTROYA (12):

- Deficiência (*Impairment*): perda ou anormalidade de uma estrutura ou função psicológica, fisiológica ou anatômica;
- Incapacidade (*Disability*): restrição ou ausência da capacidade de realizar uma atividade na forma ou dentro da margem que se considera normal para o ser humano;
- Desvantagem (*Handicap*): é a situação desvantajosa em que se encontra um indivíduo, em consequência de uma deficiência ou de uma incapacidade, que lhe limita e impede de desempenhar um rol de atividades que seria considerado normal para pessoas da mesma idade, sexo e nível sociocultural.

Com base nas definições acima, constatamos que os termos deficiência e incapacidade não são sinônimos, embora estejam intimamente relacionados e causem algum tipo de desvantagem as pessoas que as possuem.

Por este motivo podemos encontrar pessoas, que mesmo sem possuir nenhuma deficiência, apresentam algum tipo de incapacidade ao realizar uma determinada tarefa. Tomemos como exemplo um atleta paraolímpico, que mesmo com sua deficiência é capaz de participar de uma prova de natação, uma atividade que muitas pessoas que não possuem deficiência são incapazes de realizar, pelo simples fato de não saberem nadar.

É extremamente importante reconhecer, que é impossível estabelecer uma fronteira, capaz de separar linearmente as pessoas que não possuem deficiências, daquelas consideradas incapazes, uma vez que a medição de performance ou habilidade para a execução de uma determinada atividade, não pode ser representada de forma bivalente, mas sim através de uma função contínua, VANDERHEIDEN (21).

Segundo NEWELL (13), os seres humanos podem ser representados como pontos em um espaço multidimensional, formado por múltiplos eixos. Cada um destes eixos, representa uma das diversas características de um ser humano:

- Características físicas: altura, peso, idade, etc.;
- Habilidades sensoriais: percepção de cores, habilidade auditiva, etc.;
- Funcionalidade intelectual e emocional: QI, QE, etc.

Deste modo, cada ser humano pode ser plotado neste espaço multidimensional, em um dado momento do tempo.

O fato é que uma pessoa pode ter uma incapacidade permanente, em consequência de alguma deficiência, ou uma incapacidade temporária, em função de acidentes, doenças, ou de fatores associados ao estresse, carga de trabalho, distúrbio do sono, ambiente, entre outros fatores. Isso sem considerar-mos o fato de que uma deficiência não é necessariamente congênita, podendo ser adquirida em consequência de traumas, acidentes ou doenças.

Portanto, cada um de nós se move dentro deste gráfico multidimensional, a cada dia, hora, ou até mesmo a cada minuto. Isso significa que a classificação de um indivíduo, como capaz ou incapaz, pode mudar inúmeras vezes ao longo de sua vida.

Uma incapacidade, seja ela consequência de uma deficiência ou não, é o agente causador de uma desvantagem, e conseqüentemente de uma necessidade especial, que precisa ser trabalhada, a fim de suplantar esta desvantagem.

Deste modo qualquer pessoa tem, teve ou pode vir a ter uma incapacidade ao longo da vida, que a inclui dentro do grupo dos portadores de necessidades especiais.

Se considerarmos a natureza nebulosa desta classificação, podemos concluir que todos nós pertencemos ao conjunto dos portadores de necessidades especiais, com

graus de pertinência distintos (e.g. uma pessoa cega, possui grau de pertinência elevado, ao passo que uma pessoa que usa óculos, possui grau de pertinência baixo).

### 2.1.2. Novas classificações.

A OMS elaborou mais recentemente um novo documento chamado ICIDH-2 (*International Classification of Functioning, Disability and Health*), no qual a entidade altera o escopo da primeira versão do documento, baseada na classificação das conseqüências da doença, para a classificação dos componentes da saúde.

Dentre as classificações componentes do ICIDH-2, destacamos aquelas particularmente úteis para o desenvolvimento deste trabalho, apresentadas aqui com base em WORLD HEALTH ORGANIZATION - WHO (23):

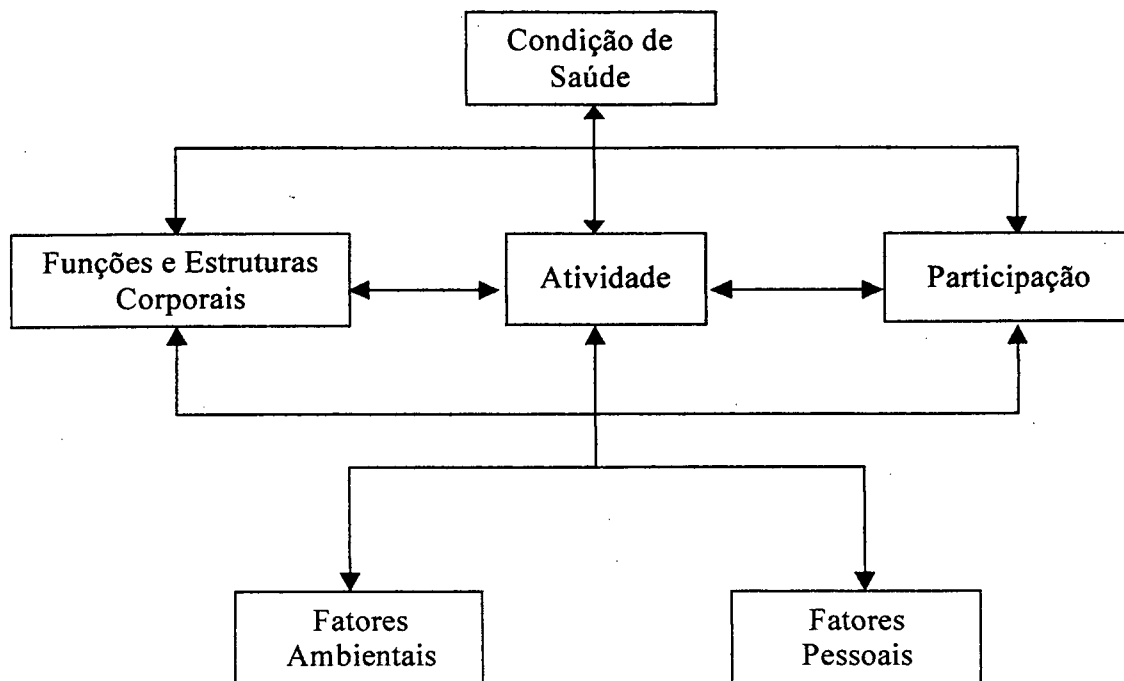
- Funcionamento (*Functioning*): é um termo genérico para funções corporais, estruturas corporais, atividades e participação. Representa os aspectos positivos da interação entre um indivíduo e seus fatores contextuais;
  - **Funções corporais:** são as funções fisiológicas e psicológicas do sistema corporal;
  - **Estruturas corporais:** são as partes anatômicas do corpo, como os órgãos e membros;
  - **Atividade:** é a execução de uma tarefa ou ação por um indivíduo;
  - **Participação:** é o envolvimento em uma determinada situação da vida;
  - **Fatores contextuais:** são os fatores que em conjunto constituem o contexto completo da vida de um indivíduo. É composto pelos fatores ambientais e pessoais;
  - **Fatores ambientais:** refere-se a todos os aspectos do mundo externo que influenciam a vida de um indivíduo, tais como, o mundo físico, políticas, regras, leis, sistemas sociais e serviços;
  - **Fatores pessoais:** refere-se aos fatores relacionados ao indivíduo, tais como, idade, sexo, posição social e experiência de vida.

- Incapacidade (*Disability*): é um termo genérico para deficiências, limitações nas atividades e restrições na participação. Representa os aspectos negativos da interação entre um indivíduo e seus fatores contextuais;
  - **Deficiência:** são problemas nas funções ou nas estruturas corporais, referentes a perdas ou desvios significativos;
  - **Limitações na atividade:** são dificuldades que um indivíduo pode ter na execução de uma atividade. Uma limitação na atividade pode variar de um pequeno a um grande desvio, referente a qualidade ou quantidade na execução da atividade, na maneira ou na extensão que é esperada para uma pessoa sem as mesmas condições de saúde;
  - **Restrições na participação:** são problemas que um indivíduo pode experimentar no envolvimento em situações da vida. A presença na restrição da participação, é determinada pela comparação da participação do indivíduo, com aquela que é esperada para um indivíduo sem incapacidade, naquela mesma cultura ou sociedade.
  
- Condição de saúde: é um termo genérico para doença, desarranjo, ferimento ou trauma. A condição de saúde pode abranger outros fatores como gravidez, envelhecimento, estresse, anomalia congênita ou predisposição genética.

Estes conceitos são importantes, uma vez que precisamos detectar quais as limitações na atividade e as restrições na participação de um determinado grupo de pessoas, para que possamos desenvolver tecnologias computacionais, capazes de ampliar a participação destas pessoas com seus fatores contextuais, principalmente no caso dos portadores de deficiência, que possuem uma participação bastante restrita, podendo ainda auxiliar na melhora da condição de saúde destas pessoas.

A Fig.1 ilustra graficamente a forma como estes componentes se relacionam e influenciam a condição de saúde de um indivíduo.

**FIGURA 1** Interação entre os componentes apresentados pela ICIDH-2.



Fonte: WHO. ICIDH-2 Final Draft. Disponível na Internet. <http://www.who.int/icidh/prefinaldec2000.htm>. 07 junho 2001.

### 2.1.3. Características das deficiências.

Podemos encontrar na ICIDH-2, características mais particulares das deficiências, WHO (23):

- Podem ser temporárias ou permanentes; progressivas, regressivas ou estáticas; intermitentes ou contínuas;
- Podem ser classificadas em quatro categorias distintas, (a) Perda ou Falta, (b) Redução, (c) Adição ou Excesso, (d) Desvio;
- Representam um desvio dentro de um padrão, aceito para uma determinada população, em termos biomédicos relacionados ao corpo e suas funções;
- Podem ocasionar outras deficiências (e.g. a perda da força nos músculos, pode ocasionar deficiência na locomoção);

- Podem ser uma parte ou uma expressão da condição de saúde de uma pessoa, mas não indicam necessariamente que exista uma doença.

## **2.2 DEFININDO BARREIRA E ACESSIBILIDADE**

### **2.2.1. Barreira.**

Com base em WHO (23), podemos definir uma barreira como sendo fatores em um ambiente, cuja ausência ou presença, limitam e criam uma incapacidade. As barreiras incluem aspectos como um ambiente físico inacessível, carência de tecnologias assistivas, reação negativa de terceiros diante da incapacidade e ainda serviços e políticas ineficientes, ou que impeçam o envolvimento de todas as pessoas em todas as áreas da vida.

### **2.2.2. Acessibilidade.**

Com base em UNIVERSIDADE SÃO PAULO - USP (19), podemos definir a acessibilidade como a possibilidade e condição de alcance para a utilização de elementos com segurança e autonomia. Um elemento acessível é aquele que pode ser alcançado, visitado e utilizado por qualquer pessoa, inclusive as portadoras de deficiência.

## **2.3 DEFICIÊNCIAS E AS BARREIRAS PARA ACESSIBILIDADE**

Passemos agora para uma análise das deficiências sensoriais, motoras e cognitivas, no sentido de identificar as possíveis barreiras que podem incapacitar os portadores destas deficiências e detectar soluções para a transposição das mesmas. As informações apresentadas aqui estão baseadas em VANDERHEIDEN (22).

### **2.3.1. Deficiência visual.**

A deficiência visual abrange uma ampla faixa de casos, desde pessoas com visão muito pobre, dita sub-normal, passando por aquelas que conseguem perceber a luz sem distinguir formas, até aquelas totalmente cegas.

Para estas pessoas a maior dificuldade, refere-se a interação com mostradores de informação, como displays e monitores. Estas pessoas também encontram sérias limitações no uso de controles e dispositivos baseados na orientação visual, como no caso do mouse, cuja operação baseia-se na identificação da posição do cursor na tela.

Muitas das pessoas que possuem deficiência visual ainda possuem preservadas uma parte da capacidade visual, podendo ser auxiliadas por magnificadores de tela, iluminadores e redutores de brilho. Nos casos de visão sub-normal, o uso de fontes com dimensões grandes, tipografia com serifa e cores que reforcem o contraste, são medidas particularmente úteis.

Os portadores de deficiências visuais mais severas, podem utilizar estratégias tecnológicas baseadas em Braille e em interfaces sonoras.

### 2.3.2. Deficiência auditiva.

A deficiência auditiva representa um grau ou nível de desordem auditiva, sendo a surdez a incapacidade extrema de discriminar níveis de fala através das orelhas, portanto, uma pessoa surda é incapaz de utilizar a audição para se comunicar. Uma pessoa pode ser considerada surda, quando só é capaz de ouvir sons superiores a 90 decibels.

A principal dificuldade destas pessoas, esta em interagir com dispositivos que utilizam o meio sonoro para transmitir informações. Este problema poderia ser superado, adotando-se uma representação redundante da informação sonora, através de meios visuais e/ou táteis.

Uma alternativa mais factível, seria oferecer um mecanismo através do qual o usuário pudesse selecionar dispositivos alternativos de saída.

Outras estratégias baseadas na flexibilização da faixa de volume e frequências dos equipamentos comerciais, a utilização de linguagem de sinais e de tecnologias de telecomunicação para surdos (e.g. Closed Caption), são outras formas de transpor as barreiras impostas pela deficiência auditiva.



### 2.3.3. Deficiência física.

As deficiências físicas abrangem um grande universo de deficiências, cujas origens e características podem ser extremamente distintas. As deficiências físicas podem ser divididas em dois grande grupos:

a) Deficiências Neuromusculares: caracterizadas por três tipos distintos:

Paralisia (perda total do controle muscular, em determinada parte ou no corpo inteiro);

Fraqueza (paresia; perda da força muscular, fraqueza muscular ou dor);

Interferência no controle em conseqüência de Espasmo (quando os músculos ficam tensos), Ataxia (problema na coordenação e programação motora), ou Atetose (movimento extra, involuntário ou incontrolável).

b) Deficiências do Esqueleto: caracterizadas por problemas na movimentação das juntas, atrofia ou perda de membros.

Dentre as várias deficiências físicas conhecidas, podemos citar:

- **Artrite:** dor nas juntas que causa redução nos movimentos e fraqueza;
- **Paralisia cerebral:** é definida como dano nas áreas motoras do cérebro. Pode ser causada por altas temperaturas, falta de oxigenação e traumas cranianos. Existem vários quadros de paralisia cerebral, como as do tipo espástica, atáxica e atetóide;
- **Lesão medular:** pode ocasionar paralisia ou paresia, sendo a extensão da mesma determinada pela altura do trauma na região medular. A tetraplegia é causada por trauma acima da região cervical, ocasionando em danos motores nos quatro membros do corpo, já as lesões abaixo da região cervical denotam em paraplegia, que afeta apenas os membros inferiores;
- **Acidente vascular cerebral (AVC):** são ocasionados por trombose (coágulo nos vasos sanguíneos que bloqueia o fluxo de sangue), hemorragia (sangramento no tecido cerebral em conseqüência do aumento da pressão

sangüínea ou ruptura de aneurisma) e embolia (coágulo que se rompe e bloqueia uma artéria);

- **Perda de membros e dedos:** pode ocorrer por motivo de amputação, em consequência de um trauma ou cirurgia, ou por motivos congênitos;
- **Mal de Parkinson:** enfermidade progressiva caracterizada pela rigidez dos músculos, lentidão dos movimentos e tremores característicos;
- **Esclerose múltipla:** é uma doença progressiva do sistema nervoso central que degenera o material isolante que recobre as fibras nervosas. Ocasiona problemas no controle motor, fraqueza, dificuldade para andar, ver e manipular objetos;
- **Esclerose lateral amiotrófica:** é uma doença degenerativa fatal que ocorre no sistema nervoso central, caracterizada pela evolução lenta da paralisia muscular;
- **Distrofia muscular:** caracterizada por um grupo de doenças hereditárias, que acarretam em enfraquecimento muscular progressivo, perda do controle motor, contrações, dificuldade para andar, respirar e manipular objetos.

Os portadores de deficiência física possuem dificuldades conseqüentes de problemas no controle muscular, fraqueza e fadiga; dificuldade para caminhar, ver, falar e sentir; dificuldade para alcançar objetos e efetuar manipulações complexas.

Estas pessoas necessitam de tecnologias assistivas, projetadas tomando como base suas aptidões, a fim de auxiliá-las na execução de tarefas relacionadas a mobilidade, manipulação e comunicação.

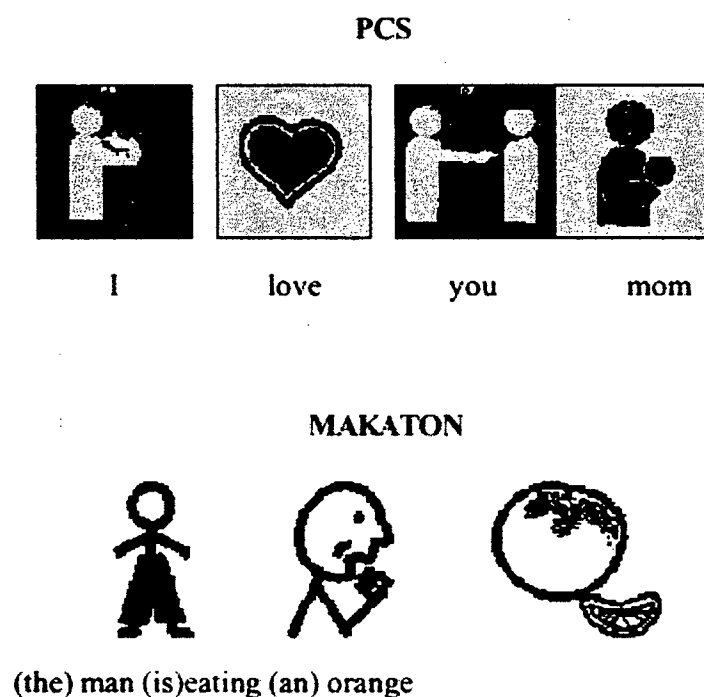
#### 2.3.4. Deficiência cognitiva e de linguagem.

As deficiências cognitivas são muitíssimo variadas, mas podem ser caracterizadas como problemas de memória (dificuldade de reconhecer e recuperar informações), percepção (dificuldade de discriminar informações sensoriais), resolução de problemas (dificuldade na identificação de um problema, seleção de método para solução e interpretação de resultados) e conceituação (dificuldades relacionadas a identificação de fenômenos de ação e consequência, conceitos abstratos, compreensão, etc.).

As deficiências de linguagem ocasionam dificuldade na compreensão e/ou expressão da linguagem oral e/ou escrita.

As tecnologias destinadas a estas pessoas, embora ainda bastante escassas, baseiam-se no uso de mostradores simples, linguagens de baixa complexidade, Fig. 2, uso de padrões e seqüências de operação simples e óbvias.

**FIGURA 2** Sentenças pictográficas nas linguagens PCS e Makaton.



Fonte: LOPES, João B. Designing User Interfaces for Severely Handicapped Persons. Disponível na Internet. <http://virtual.inesc.pt/wuau01/procs/papers-list.html>. 18 maio 2001.

Embora a análise tenha trabalhado com as deficiências de forma isolada, não podemos ignorar a possibilidade da ocorrência de múltiplas deficiências em um mesmo indivíduo, já que uma das características das deficiências é o fato de poderem ocasionar outras deficiências, WHO (23).

Isto pode ocorrer em casos como o de diabéticos, que podem ter a visão comprometida pela doença, juntamente com a perda da sensibilidade nos dedos, o que dificulta o uso de tecnologias baseadas no sistema Braille, ou ainda em casos de

paralisia cerebral, que além de problemas motores, pode ocasionar problemas de visão, audição, fala e comprometimento cognitivo.

Estes casos em particular são especialmente desafiadores para os pesquisadores da área, no sentido de oferecer soluções híbridas, que combinem várias tecnologias assistivas em uma única.

### 3. ERGONOMIA DE INTERFACES ESPECIAIS

#### 3.1 CONCEITOS

##### 3.1.1. Ergonômica de interface homem - computador.

Uma das definições para ergonomia, dada por Wisner (1972), apud CYBIS (4), é a de conjunto de conhecimentos científicos relativos ao homem e necessários à criação de instrumentos, máquinas e dispositivos, que possam ser utilizados com o máximo de conforto, segurança e eficácia.

A Tabela 1 mostra a classificação geral da ergonomia segundo seus objetivos, áreas de atuação e domínio:

**TABELA 1** Escopo da ergonomia.

Domínio das Atividades	Objetivos	Atuação
Ergonomia física	Ergonomia de concepção e projeto	Ergonomia de produto
Ergonomia cognitiva	Ergonomia de verificação e correção	Ergonomia de processo

Fonte: CYBIS, Walter de Abreu. Ergonomia de Interfaces Humano - Computador. Disponível na Internet. <http://www.labiutil.inf.ufsc.br/apostila/apostila.htm>. 21 junho 2000

A ergonomia de interfaces homem - computador (*IHC*), se aplica no âmbito das tarefas informatizadas onde os processos cognitivos são preponderantes, CYBIS (4).

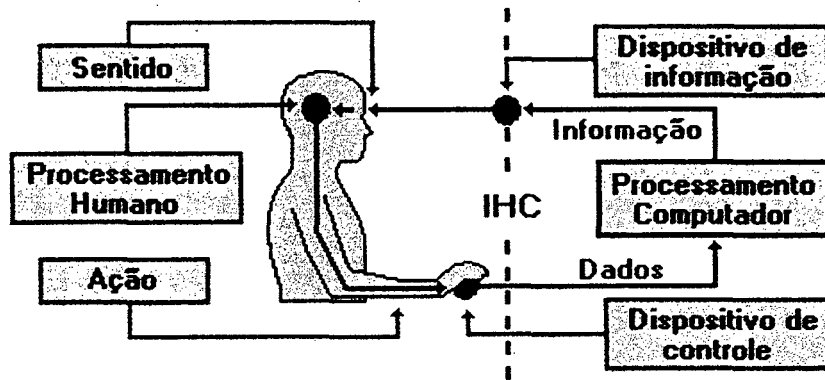
A ergonomia de IHC leva em consideração fatores relacionados a memória, níveis de percepção, raciocínio e comunicação, Fig. 3. Através dela é possível criar interfaces, que ofereçam todos os objetos de interação pertinentes à realização de uma determinada tarefa, por um determinado grupo de usuários.

##### 3.1.2. Avaliação ergonômica de IHC.

A avaliação ergonômica tem como finalidade detectar problemas de usabilidade, definida pela norma ISO 9241 como a capacidade que apresenta um sistema interativo

de ser operado, de maneira eficaz, eficiente e agradável, em um determinado contexto de operação, para a realização das tarefas de seus usuários, CYBIS (4).

FIGURA 3 Modelo clássico de interação homem - computador



Fonte: DUL, J., WEERDMEESTER, B. Ergonomia Prática. São Paulo, Editora Edgard Blücher LTDA, 1998, 147p. Informação e Operação, p.55-84

Os problemas detectados através da avaliação de usabilidade, são:

- Barreira: aspecto da interface na qual o usuário esbarra sucessivas vezes, sem conseguir suplantá-la;
- Obstáculo: aspecto da interface na qual o usuário esbarra, mas consegue superar;
- Ruído: aspecto da interface que afeta prejudicialmente o desempenho do usuário.

### 3.1.3. Critérios para qualidade ergonômica em IHC.

Existem diversos critérios utilizados para definir a qualidade ergonômica de uma interface, segundo Bastien e Scapin (1993), apud CYBIS (4), é possível avaliar a qualidade ergonômica de um software interativo através de 8 critérios principais:

- Condução: um programa ergonômico é capaz de aconselhar o usuário, conduzindo-o até o seu objetivo. A condução é composta por características como:

- **Presteza:** diz respeito as informações oferecidas ao usuário, para que este tome conhecimento do estado e contexto nos quais ele se encontra;
  - **Resposta imediata:** refere-se a capacidade de um software em produzir respostas as ações do usuário, de forma rápida e clara;
  - **Legibilidade:** refere-se a apresentação das informações levando em consideração as características cognitivas e perceptivas do usuários;
  - **Agrupamento e distinção de itens:** diz respeito a organização visual dos itens de informação relacionados uns com os outros, levando em consideração o posicionamento e características gráficas;
- Carga de trabalho: um programa ergonômico exige pouco trabalho cognitivo para o usuário. É composta por características como:
    - **Brevidade:** refere-se a capacidade do software em respeitar a capacidade de trabalho perceptivo e cognitivo do usuário, tornando sucinto o número de itens e passos durante a interação;
    - **Densidade informacional:** diz respeito a redução da quantidade total de itens de informação exibidos simultaneamente ao usuário;
  - Controle explícito: um software ergonômico é obediente ao controle do usuário;
  - Adaptabilidade: um software ergonômico tem a capacidade de reagir conforme o contexto e conforme as necessidades do usuário. É composta por características como:
    - **Flexibilidade:** refere-se aos meios através dos quais o usuário pode personalizar a interface;
    - **Consideração da experiência do usuário:** refere-se a capacidade de respeitar o nível de conhecimento do usuário;

- **Gestão de erros:** um software ergonômico possui mecanismos, capazes de evitar ou reduzir a ocorrência de erros;
- **Homogeneidade e coerência:** um software ergonômico possui códigos, formatos e procedimentos idênticos em contextos idênticos;
- **Significado dos códigos e denominações:** um software ergonômico possui perfeita adequação entre o objeto ou informação e sua referência;
- **Compatibilidade:** um software ergonômico apresenta procedimentos compatíveis com as características psicológicas do usuário.

## **3.2 INTERFACES PARA PORTADORES DE NECESSIDADES ESPECIAIS**

### **3.2.1. Interfaces especiais.**

Os usuários portadores de necessidades especiais, necessitam de cuidados ergonômicos mais refinados, em virtude de suas limitações físicas e/ou cognitivas.

Embora os avanços tecnológicos, obtidos nas áreas de software e hardware, já ofereçam alternativas para interação com ambientes WIMP (*Windows, Icons, Menus, Pointer*), baseadas na fala, gestos, postura, em sensores biométricos, entre outros avanços, existem muitos desafios relacionados ao desenvolvimento de interfaces especiais, capazes de atender a uma faixa mais ampla de usuários e habilidades, BRANCO (2).

Os desafios encontrados no desenvolvimento de interfaces especiais, estão relacionados a fatores como:

- A dificuldade de coletar dados capazes de representar as aptidões e preferências dos usuários destas interfaces;
- A dificuldade em permitir que a mesma informação possa ser manipulada por usuários com diferentes habilidades e impedâncias, permitindo que os mesmos



interajam entre si (e.g. permitir que um indivíduo surdo e outro cego se comuniquem);

- Problemas com as arquiteturas tradicionais dos computadores, que limitam o uso de interfaces multimodais.

Segundo LOPES (10), os principais fatores envolvidos no desenvolvimento de interfaces voltadas aos portadores de necessidades especiais são:

- Limitação do uso de dispositivos de entrada: muitos usuários com incapacidade física, possuem muita limitação no uso das mãos, fato que dificulta o uso de dispositivos como o mouse e o teclado. Isto se deve a dificuldade dos usuários em apontar os elementos na tela, no caso de dispositivos apontadores, e também da dificuldade de acionar os botões e teclas;
- Tamanho, cor e animação: estes itens são muito importantes, visto que muitas pessoas possuem problemas visuais associados com suas limitações físicas. Muitas pessoas possuem problemas em distinguir planos de ação, sem a utilização do contraste apropriado e dificuldade em trabalhar com objetos animados;
- Coordenação: os usuários possuem dificuldade de coordenar a ação que eles exercem através dos dispositivos de entrada, com aquilo que ocorre na tela do computador;
- Leitura e escrita: a leitura e escrita não são tarefas fáceis para estes usuários, uma vez que envolvem tarefas complexas para o cérebro e exigem em alguns casos acuidade visual para a identificação das palavras.

### 3.2.2. Características pertinentes para uma interface especial.

Segundo LOPES (10), uma interface desenvolvida para portadores de necessidades especiais deve:

- Permitir a adaptação do tamanho, cor e quantidade de objetos exibidos na tela, as necessidades do usuário;
- Permitir o uso do número mais amplo possível de dispositivos de entrada (e.g. mouse, trackball, joystick, chaves, comandos de voz);
- Permitir o uso mais amplo possível de formas de saída, incluindo gráficos, sons e voz;
- Utilizar um desenho simplificado;
- Promover a interação e reter a atenção do usuário;
- Permitir que o retorno das ações do usuário, promovam grande alteração na interface.

Muitas destas características privilegiam não apenas os usuários portadores de deficiência física, como também os idosos, que apresentam perdas e limitações provenientes do envelhecimento. A diferença básica existente entre estes dois grupos de usuários, está no nível e quantidade de incapacidades apresentadas, estando os deficientes físicos propensos a apresentar índices maiores em ambos os casos.

### 3.2.3. Interfaces multimodais.

Com base em OVIATT (15), podemos definir as interfaces ou sistemas multimodais, como interfaces capazes de serem operadas através de diferentes modos, permitindo com isso que os usuários possam escolher a modalidade de operação compatível com suas habilidades, preferências ou condições de uso.

As principais razões para o desenvolvimento de interfaces multimodais são:

- A expansão potencial de acessibilidade aos computadores oferecida por estas interfaces;
- A possibilidade de se criar novas formas de interação com os computadores não previstas anteriormente e das quais outros usuários poderão se beneficiar, como no caso de sistemas operados por voz;

- A capacidade de suportar um perfil mais diversificado de usuários, no que se refere a vários aspectos como a idade, nível de habilidade, desenvolvimento lingüístico, estilo cognitivo e capacidade sensorial;
- Prevenir danos físicos provenientes do uso prolongado do computador, as chamadas lesões por esforço repetitivo (*LER*);
- Expansão do contexto de utilização viável dos computadores (e.g. aplicações móveis operadas sem o uso das mãos).

Esta capacidade de operação múltipla, é de extrema importância para sistemas destinados a pessoas portadoras de deficiência física, uma vez que muitas das barreiras para o acesso aos computadores encontram-se nos dispositivos e modos tradicionais de operação das interfaces.

## **4. CARACTERÍSTICAS GERAIS DOS SISTEMAS SIMULADORES DE TECLADO**

Segundo KOON (9) os sistemas simuladores de teclado pertencem ao grupo das tecnologias assistivas para computação destinadas ao acesso de informação, que permitem que pessoas portadoras de necessidades especiais, possam interagir com computadores.

Estes sistemas de maneira geral, representam na tela de um computador uma interface semelhante a um teclado, permitindo que o usuário manipule o sistema ao invés do dispositivo. Estes sistemas são particularmente úteis no auxílio de pessoas com limitações motoras, em virtude de deficiências neuromusculares ou do esqueleto.

Como qualquer outro tipo de sistema, os simuladores de teclado podem possuir abordagens distintas, diferindo na maneira como os sistemas implementam determinadas características gerais de funcionamento e/ou na presença de recursos adicionais.

Veremos agora uma explanação das principais características apresentadas pelos sistemas simuladores de teclado pesquisados.

### **4.1 SELEÇÃO DE TECLAS**

A seleção dos elementos que compõem a interface de um simulador de teclado pode ser feita de forma direta ou indireta.

#### **4.1.1. Seleção direta.**

Requer que o usuário aponte e ative os elementos de interação através de meios próprios, utilizando para isto um dispositivo apontador, normalmente o mouse.

Em casos de limitações físicas mais severas o mouse pode ser substituído por dispositivos mais sofisticados, geralmente operados através da movimentação dos olhos e da cabeça, como o dispositivo apresentado por EVANS (8).

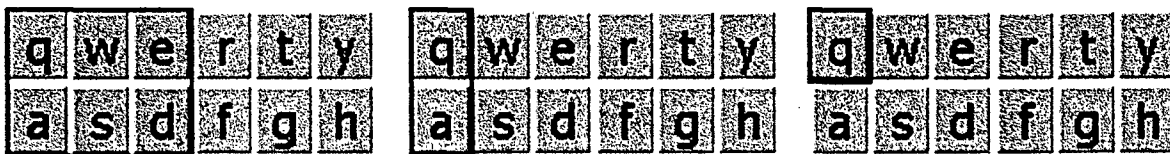
#### 4.1.2. Seleção indireta.

Permite que o usuário apenas confirme o acionamento das teclas, uma vez que o próprio sistema se encarrega de selecioná-las. O elemento que indica a seleção das teclas pode ser o próprio cursor do mouse, ou elementos gráficos especiais. A forma como o sistema efetua a seleção das teclas é chamada de varredura, sendo as principais formas:

- **Seqüencial:** as teclas são selecionadas seguindo uma seqüência previamente estabelecida, normalmente da esquerda → direita / superior → inferior;
- **Matricial:** dividi-se em dois ciclos interdependentes, um ciclo inicial que efetua a varredura de linhas, que aponta grupos de teclas dispostas horizontalmente, e um ciclo subsequente dedicado a varredura de colunas, que seleciona seqüencialmente as teclas da linha;
- **Grupal:** esta varredura seleciona grupos iniciais de teclas, de forma que o usuário possa selecionar um grupo e dividi-lo em subgrupos, que serão selecionados separadamente no próximo ciclo da varredura. O processo se repete ate que o subgrupo resultante contenha apenas uma tecla, que pode então ser acionada, fazendo com que a varredura retome a seleção dos grupos iniciais, Fig.4.

A seleção indireta é utilizada em conjunto com dispositivos de seleção binária (liga/desliga), chamados de acionadores, que substituem os dispositivos apontadores, permitindo que o usuário acione as teclas do sistema. Dispositivos analógicos como o microfone, também podem ser utilizados como acionadores.

FIGURA 4 Exemplo do funcionamento da varredura grupal.



## 4.2 POSICIONAMENTO

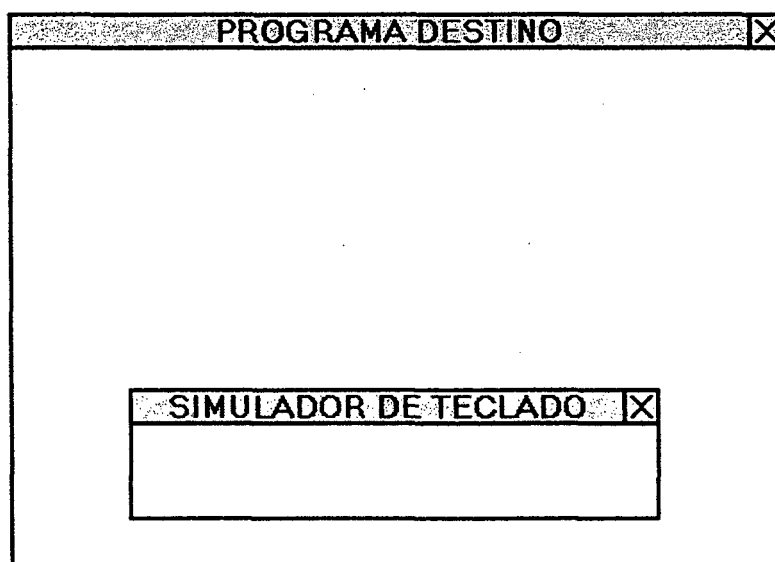
O posicionamento refere-se a maneira como o sistema se apresenta na tela do computador, e define o nível de acessibilidade que o usuário possui para interagir com os programas do computador.

A maior parte dos sistemas utiliza um posicionamento em primeiro plano, utilizando uma pequena parte da tela, permitindo que o usuário visualize e utilize outros programas posicionados em segundo plano, destino das informações “digitadas” no simulador de teclado, Fig. 5. Esta abordagem é a mais utilizada devido ao alto nível de acessibilidade oferecido aos usuários.

Alguns sistemas utilizam um posicionamento que utiliza toda a tela, fazendo com que todas as informações digitadas sejam processadas apenas pelo sistema simulador de teclado, que neste caso passa a funcionar com uma espécie de editor de texto.

Esta abordagem oferece baixo nível de acessibilidade aos usuários, uma vez que os mesmos estão limitados ao escopo do sistema.

FIGURA 5 Posicionamento em parte da tela.



## 4.3 RECURSOS ESPECIAIS

Os sistemas simuladores de teclado podem possuir vários recursos especiais, criados para oferecer aos usuários mais comodidade e conforto na tarefa de digitação. Dentre os diversos recursos empregados por sistemas do gênero destacam-se:

#### 4.3.1. Predição de palavras.

Este recurso permite que o usuário selecione palavras preestabelecidas a medida em que aciona as teclas do simulador de teclado. As palavras são filtradas de dicionários, com base nos caracteres digitados, permitindo que o usuário ganhe tempo na digitação de palavras de uso frequente.

Outra possibilidade deste recurso é permitir que o usuário utilize dicionários mais específicos, contendo outros idiomas ou palavras reservadas para linguagens de programação.

#### 4.3.2. Emulação de mouse.

A emulação de mouse consiste em uma forma alternativa para a movimentação e operação do mouse, compreendendo uma série de alternativas distintas, muitas vezes implementadas em sistemas específicos. Este recurso permite que o usuário tenha acesso mais ágil a funções e comandos acessáveis via mouse, como o caso dos itens de menu de programas.

#### 4.3.3. Múltiplos padrões de teclado.

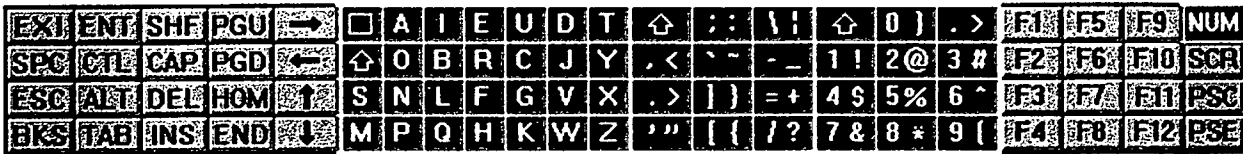
Com este recurso os usuários são capazes de alterar o padrão de teclado utilizado pelo sistema, permitindo a utilização do padrão correto para o tipo de tarefa a ser realizada e tornando o sistema mais receptivo ao conhecimento prévio do usuário.

### **4.4 OS SISTEMAS SIMULADORES DE TECLADO PESQUISADOS**

#### 4.4.1. UniControl 1.0.

Programa desenvolvido no Brasil, Fig. 6, que possui como características a adoção da varredura grupal e uso de um padrão de teclado bastante particular.

FIGURA 6 Janela do UniControl.

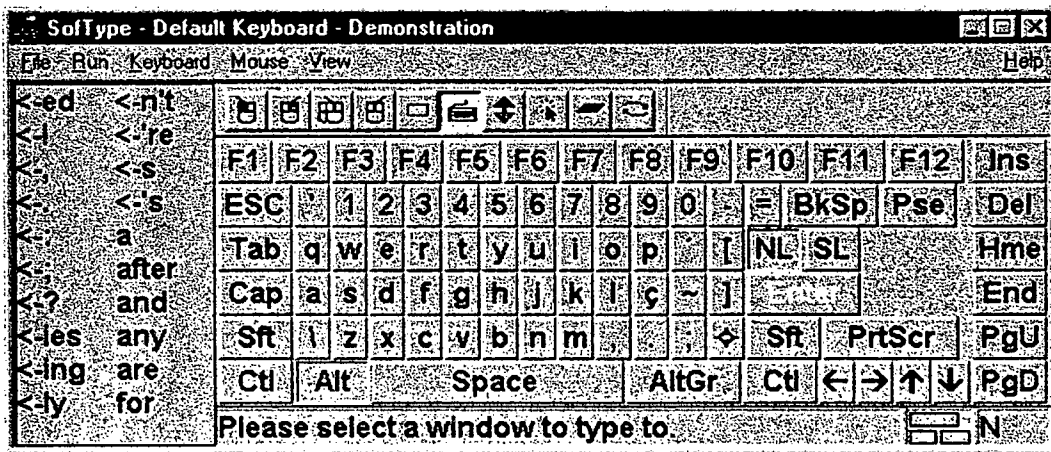


Disponível na Internet. [http://kidslink.bo.cnr.it/usl/cat\\_hw/soft/unictrl1.html](http://kidslink.bo.cnr.it/usl/cat_hw/soft/unictrl1.html).

4.4.2. SoftType.

Programa desenvolvido nos Estados Unidos, que possui como características gerais o uso da seleção direta para digitação, recurso de predição de palavras, emulação de mouse e múltiplos padrões de teclado, Fig. 7.

FIGURA 7 Simulador de Teclado SoftType.



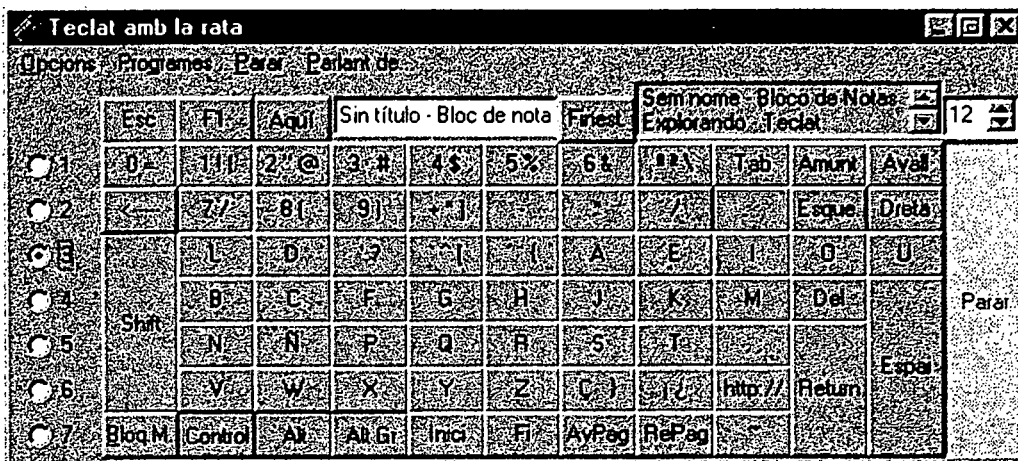
Disponível na Internet. <http://www.orin.com>

4.4.3. TotiPM 1.0.38.

Simulador de teclado desenvolvido na Espanha, que utiliza seleção direta e indireta de teclas, Fig. 8.



FIGURA 8 Janela do simulador de teclado TotiPM.

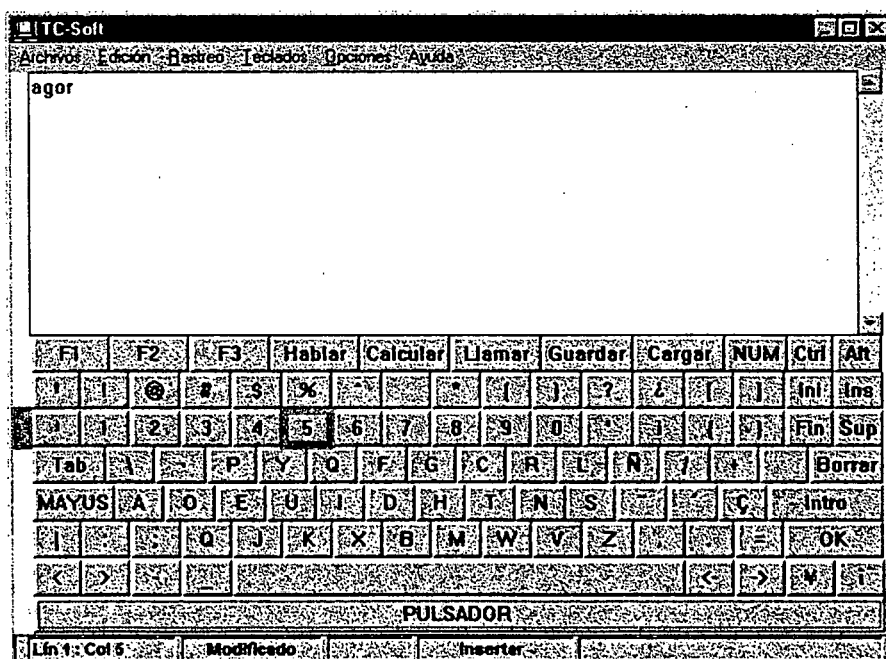


Disponível na Internet. <http://www.xtec.es/~jlagares>

#### 4.4.4. TcSoft.

Programa desenvolvido na Espanha, Fig. 9, sendo o único programa pesquisado que adota o posicionamento em tela toda, permite a seleção direta e indireta de teclas e permutação entre os padrões de teclado QWERTY e DVORAK.

FIGURA 9 Interface do TcSoft.



Disponível na Internet. <http://www.angelfire.com/id2/infodis/soft.html>.

#### 4.4.5. Simulador de Teclado.

Programa brasileiro para o ambiente Microsoft DOS ®, adaptado a partir da versão espanhola original. Possui janelas específicas para determinados conjuntos de teclas, e disposição de teclas baseada no estudo de Poersch, Becker e Lopes (1990), apud UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL - UFRGS (20), referente ao uso percentual dos caracteres gráficos na língua portuguesa, Tabela 2. Disponível na Internet <http://www.niee.ufrgs.br/software/st/st.html>.

**TABELA 2** Distribuição dos caracteres grafêmicos.

1 - A	2 - E	3 - O	4 - S	5 - I	6 - R	7 - N	8 - D	9 - T
10 - M	11 - U	12 - C	13 - L	14 - P	15 - V	16 - G	17 - Q	18 - F
19 - B	20 - H	21 - Ç	22 - Z	23 - X	24 - J	25 - W	26 - K	27 - Y

Fonte: UFRGS. Curso de Informática Educativa Especial. Simulador de Teclado versão 1.0 Manual do Usuário. Porto Alegre, 69p. p59.

## **5. DESENVOLVIMENTO DA INTERFACE**

Através de estudos de caso realizados com portadores de paralisia cerebral, foi possível constatar a dificuldade dos mesmos em interagir com as interfaces dos simuladores de teclado utilizados nos estudos, fato que motivou a avaliação heurística das interfaces dos simuladores de teclado pesquisados, a fim de desenvolver uma proposta diferenciada, capaz de atender melhor as necessidades especiais dos usuários.

### **5.1 OS ESTUDOS DE CASO**

Durante a fase inicial da pesquisa, foram realizados três estudos de caso com alunos portadores de paralisia cerebral da Associação dos Pais e Amigos dos Excepcionais (*APAE*) de Florianópolis, com o intuito de identificar as necessidades especiais destes alunos durante a interação com computadores.

Neste estudo foram utilizados os simuladores de teclado TcSoft e TotiPm, escolhidos com base na diferença existente entre suas formas de posicionamento e indicação de varredura de teclas, e pelo fato de ambos permitirem o acionamento através do mouse, que embora não fosse o dispositivo ideal, era a única possibilidade viável de acionador no momento.

#### **5.1.1. O perfil do grupo estudado.**

Embora o número de pessoas acompanhadas tenha sido relativamente pequeno, o grupo propiciou um quadro bastante rico para a observação, devido a diferença de faixas etárias e dos quadros de paralisia cerebral apresentados. O grupo estudado era composto por:

- Um adulto com quadro de paralisia cerebral espástica progressiva, com pouca amplitude de movimentos nos membros superiores e cadeirante;

- Uma criança com comprometimento total da fala e presença de movimentos involuntários de baixa intensidade;
- Um adolescente com bom controle motor nos membros superiores, cadeirante e com visão sub-normal.

#### 5.1.2. As constatações.

Desconsiderando-se o imprevisto realizado para o acionamento dos sistemas, os estudos de caso permitiram a constatação de várias dificuldades dos usuários durante a interação com o computador, sendo que as principais dificuldades foram provenientes da inadequação das interfaces dos simuladores de teclado as necessidades dos usuários estudados. As principais dificuldades identificadas durante a interação com as interfaces foram:

- A dificuldade em visualizar as teclas dos sistemas, em virtude do tamanho pequeno das letras utilizadas na interface. Esta dificuldade ocorreu principalmente pelo fato da cadeira de rodas distanciá-los do monitor e pela presença de problemas visuais;
- A dificuldade em acompanhar a velocidade da varredura de teclas. Os usuários frustravam-se seguidamente ao acionar as teclas do sistema, devido a pouca flexibilidade nos valores de velocidade da varredura de teclas;
- A dificuldade em trabalhar com uma grande quantidade de elementos de interação simultaneamente. A reprodução integral do teclado na tela do computador, causou confusão nos usuários.

Outra característica importante observada durante os estudos de caso, se refere a forma de acionamento a ser utilizada, ficando evidente a dificuldade de se conceber uma forma única de acionamento, capaz de atender ao amplo universo de limitações físicas.

A medida mais eficiente seria permitir múltiplas formas de acionamento, que possam se adequar aos diferentes casos, permitindo principalmente a utilização de dispositivos acionadores especiais, que podem ampliar significativamente o conforto e a autonomia durante a interação.

## 5.2 AVALIAÇÃO DAS INTERFACES DOS SIMULADORES DE TECLADO

As interfaces dos simuladores de teclado pesquisados, foram submetidas a uma avaliação heurística, que se trata de um julgamento de valor sobre as qualidades ergonômicas das interfaces humano-computador, baseado na experiência e competência do avaliador, CYBIS (4).

A avaliação heurística avalia um sistema interativo e diagnostica os problemas ou barreiras que os usuários poderão encontrar durante a interação.

### 5.2.1. Itens observados durante a avaliação.

Os principais itens observados para a avaliação das interfaces, segundo as constatações obtidas através dos estudos de caso realizados e os critérios de qualidade ergonômica definidos por Bastien e Scapin (1993), apud CYBIS (4), foram:

- Tamanho das teclas e rótulos;
- Disposição das teclas;
- Uso de códigos de cores para organização das teclas;
- Utilização de elementos de interação de difícil acesso;
- Quantidade de elementos de interação apresentados simultaneamente;
- Forma de varredura de teclas;
- Indicação da varredura de teclas.

### 5.2.2. Resultados da avaliação.

Através da avaliação das interfaces dos simuladores de teclado foi possível concluir que:

- Sistemas que utilizam a forma de varredura grupal, exigem grande esforço cognitivo e físico dos usuários, devido ao grande número de acionamentos necessários para a seleção das teclas;

- Sistemas que utilizam a varredura seqüencial, podem causar tédio aos usuários, principalmente em situações onde é necessário acionar teclas localizadas nas posições finais da seqüência de varredura;
- Sistemas de varredura sinalizados através do cursor do mouse, cobrem o plano de visão do usuário, impedindo a visualização clara das teclas. Isto se deve ao fato do cursor utilizar um plano de projeção superior em relação as teclas. O problema se agrava se o cursor utilizado tiver dimensões maiores que as tradicionais;
- A utilização de padrões de teclado pouco difundidos, desconsidera a possibilidade dos usuários utilizarem, ainda que de forma precária, os teclados tradicionais, que trabalham com padrões universais, como o QWERTY. A utilização de padrões diferenciados só deve ser adotada, em casos onde exista a possibilidade de escolha entre múltiplos padrões de teclado;
- Sistemas que utilizam o posicionamento em tela toda, limitam e impõem barreira para que o usuário utilize outros programas, premissa para sistemas simuladores de teclado, por se tratarem de sistemas de acesso a informação;
- A utilização de elementos do tipo menu, mostra-se ineficaz. Os menus são ruídos na interface dos simuladores de teclado, já que não podem ser operados diretamente pela ampla maioria dos usuários e aumentam a densidade informacional da interface;
- A reprodução fidedigna do teclado na tela do computador, é uma prática comum e equivocada, uma vez que o dispositivo possui dimensões que excedem o tamanho normal dos monitores, o que obriga a redução do tamanho dos objetos de interação. Isto acarreta em uma interface com pouca legibilidade e agrupamento confuso, principalmente para os usuários com limitações visuais;
- Teclas com dimensões reduzidas, são difíceis de serem identificadas por usuários com problemas na visão, postura e por aqueles que utilizam cadeiras de rodas, que normalmente obriga um maior afastamento em relação ao computador;

- Teclas que utilizam rótulos compostos para indicar segunda função, dificultam a identificação por parte do usuário. Exibir somente o rótulo com a função atual da tecla, facilita a legibilidade.

### 5.3 A PROPOSTA DE INTERFACE DO SISTEMA

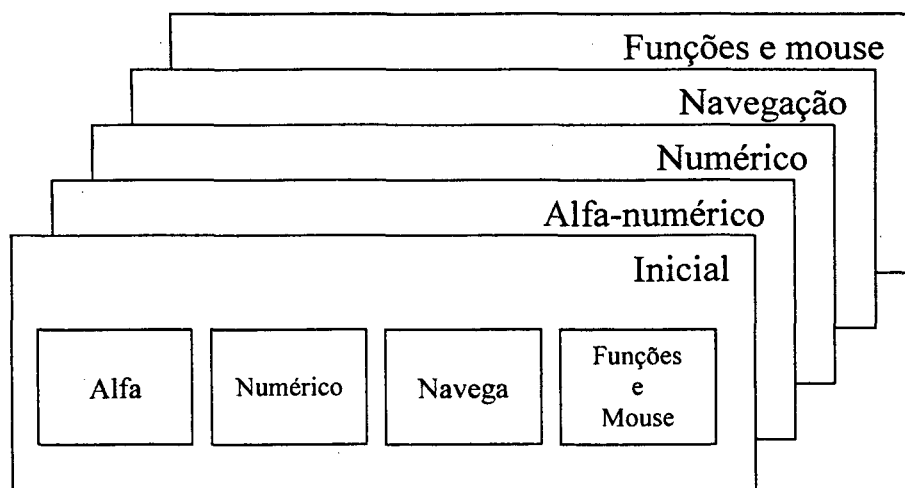
As conclusões alcançadas através da avaliação ergonômica das interfaces, encorajaram o desenvolvimento de uma interface diferenciada.

A proposta central da interface em termos de organização, é a separação física das teclas alfanuméricas, numéricas, de navegação e de funções, em teclados distintos, aproveitando a distinção contextual das mesmas e oferecendo ao usuário um teclado especializado para cada tipo de tarefa a ser executada.

A interface utiliza o posicionamento em parte da tela, com os teclados dispostos em camadas, Fig. 10, de maneira que possam ser acessados através de um teclado inicial, que funciona como um menu. Esta abordagem permite:

- A diminuição da densidade informacional;
- O aumento do tamanho das teclas;
- O aumento da área útil para os programas operados pelo simulador de teclado.

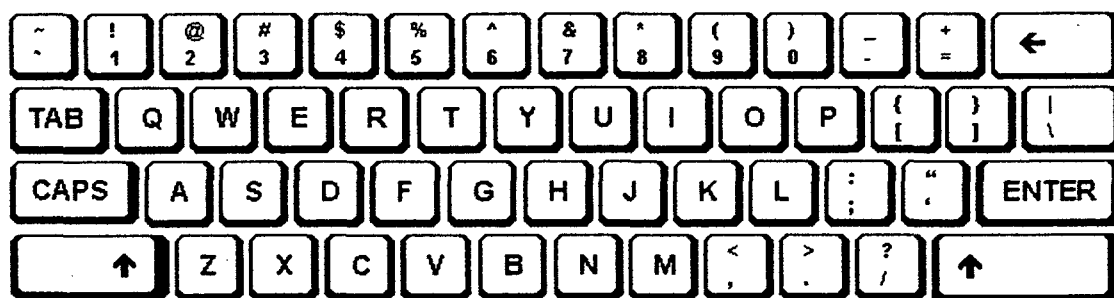
FIGURA 10 Disposição dos teclados em camadas.



As teclas do teclado alfanumérico, foram dispostas segundo o padrão QWERTY, Fig. 11, padrão bastante difundido e mais adaptável a possível experiência prévia dos usuários. Nele também foi adicionado uma área especial, através da qual os usuários podem utilizar o recurso de predição de palavras implementado no sistema.

O teclado de funções por sua vez, foi utilizado para a inserção de teclas especiais de mouse, através das quais os usuários podem utilizar o recurso de emulação de mouse implementado no sistema. Por este motivo o teclado de funções, passou a ser chamado de teclado de funções e mouse.

FIGURA 11 Padrão de teclado QWERTY.



Fonte: MICROSOFT. Disponível na Internet. [http:// www.microsoft.com/enable](http://www.microsoft.com/enable). 15 setembro 2000.

Em relação a varredura de teclas, foi escolhido o método matricial, por combinar o baixo número de acionamentos, invariavelmente dois, e a velocidade de acesso as teclas. A varredura de teclas do sistema, pode ser operada através de microfone ou dispositivos seriais, permitindo uma faixa relativamente ampla de formas de acionamento, além da seleção direta através de dispositivo apontador.

Outra medida importante, foi a substituição dos menus do simulador de teclado, por uma interface dedicada exclusivamente a tarefa de configuração do sistema, Fig. 12.

Esta substituição, esta baseada no fato de que muitos usuários dos simuladores de teclado, em virtude das limitações físicas, necessitam do apoio de uma outra pessoa durante o desempenho de suas tarefas diárias.

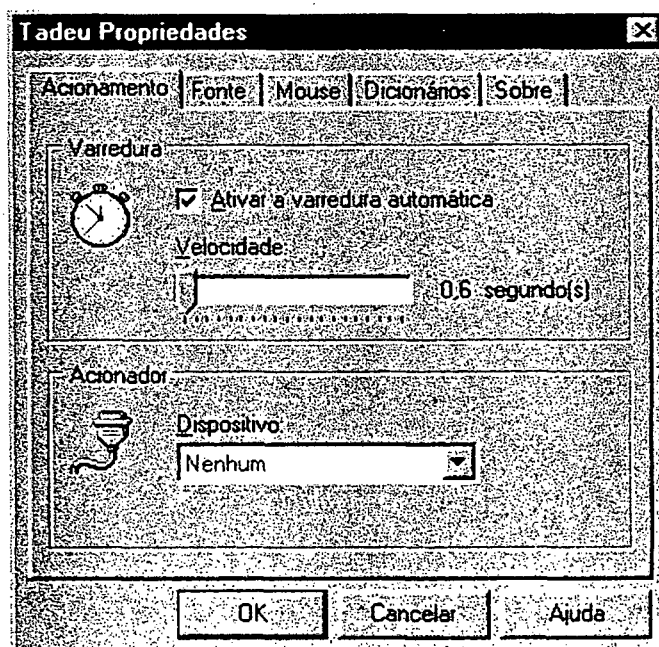
Estes facilitadores, são fisicamente capazes de operar uma interface tradicional, através da qual é possível ajustar as características do simulador de teclado, segundo as preferências e aptidões do usuário. Contudo, isto não impede que os usuários do sistema operem a interface de configuração, através do próprio simulador de teclado, ou em alguns casos de forma direta. Através do editor de propriedades é possível:



- Editar e criar dicionários de palavras usados pelo recurso de predição de palavras;
- Gravar macros de mouse associadas as teclas disponíveis no teclado de funções e mouse;
- Alterar características da fonte usada nos rótulos das teclas;
- Alterar a velocidade de varredura;
- Alterar o tipo de acionador usado para seleção de teclas.

O projeto detalhado da interface do simulador de teclado, baseado na análise de tarefa, está disponível em (ANEXO).

FIGURA 12 Interface do editor de propriedades.



## 6. DESENVOLVIMENTO DO SISTEMA

### 6.1 A MODELAGEM UTILIZADA

A base para o desenvolvimento do sistema Tadeu, dentro do ambiente Delphi 3.0, foi a criação da classe *TVirtualKeyBoard*, que reúne as principais propriedades do sistema simulador de teclado, sem se preocupar com as implementações de métodos e interface.

A adoção da língua inglesa para denominação da classe e de suas respectivas propriedades, segue o padrão sugerido para a VCL (*Visual Component Library*) do ambiente Delphi, CARVALHO (3). A classe *TVirtualKeyBoard* é composta pelas seguintes propriedades:

- *Keyboard (TPanel)*: Refere-se ao teclado atual usado pelo simulador de teclado. Ao todo o sistema utiliza 5 teclados distintos;
- *ActiveKey (TControl)*: Refere-se a tecla ativa do teclado atual;
- *LineSelector (TImage)*: Refere-se ao elemento seletor de linhas do teclado atual;
- *ColumnSelector (TImage)*: Refere-se ao elemento seletor de colunas do teclado atual;
- *Activator (Inteiro)*: Número que identifica o dispositivo utilizado para acionar o simulador de teclado;
- *Lines (Inteiro)*: Número de linhas do teclado atual;
- *Columns (Inteiro)*: Número de colunas de uma linha;
- *ActiveLine (Inteiro)*: Linha atualmente ativa, usada na varredura de teclas;
- *ActiveColumn (Inteiro)*: Coluna atualmente ativa, usada na varredura de teclas;
- *WordsDictionary (TStringList)*: Refere-se ao dicionário de palavras usado no recurso de predição de palavras;

- *TypeWritten (String)*: Propriedade utilizada como acumulador dos caracteres que são digitados através do simulador de teclado;
- *Accent (Char)*: Propriedade usada para armazenar o último acento digitado;
- *Status (TKeyboardStatus)*: Criado para indicar o estado das teclas Shift, CapsLock, Alt e Control, usado para manter a redundância da interface;
- *ScanMode (TScanMode)*: Criado para indicar o modo atual da varredura automática de teclas.

A classe *TVirtualKeyboard* é uma classe de componente não visual derivada da classe *TCustomControl*, isto significa, que esta classe serve somente como elemento de organização para o sistema.

Os objetos de interação do sistema foram divididos em 5 teclados distintos, implementados a partir da classe *TPanel*, sendo que cada um destes teclados possui basicamente um conjunto de controles do tipo *TSpeedButton*, que fazem o papel das teclas, juntamente com dois objetos *TImage*, responsáveis pela indicação da varredura de linha e coluna.

O controle do simulador de teclado consiste na manipulação indireta dos elementos da interface, a partir da manipulação do objeto *vkbTadeu*, instância da classe *TVirtualKeyboard*, cujas propriedades são instanciadas com os objetos que desejamos manipular na interface.

## 6.2 VARREDURA AUTOMÁTICA DE TECLAS

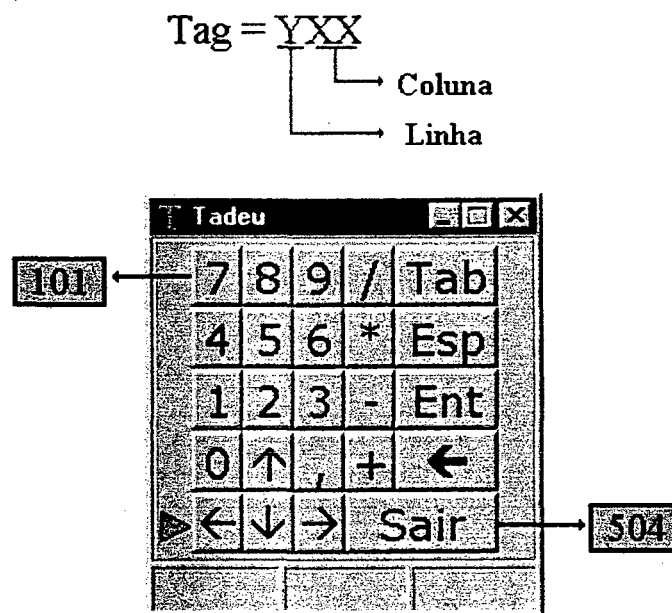
A implementação da varredura automática de teclas, consiste no momento inicial, na identificação unívoca de cada uma das teclas que compõem um teclado. As teclas aqui mencionadas, tratam-se de objetos pertencentes a classe *TSpeedButton*, escolhida por ser uma classe que não permite a permanência de foco sobre os objetos, fato que prejudicaria a implementação gráfica da varredura.

No caso do sistema proposto, a identificação das teclas baseia-se na atribuição de uma propriedade capaz de expressar a linha e a coluna que cada tecla ocupa dentro

de um teclado, uma vez que o esquema de varredura utilizado é matricial, baseado em linhas e colunas.

A melhor forma encontrada para a identificação das teclas, foi a criação de um código de varredura, aplicado aos elementos de interação em tempo de projeto, utilizando-se para isso a propriedade *Tag* já disponível na classe usada para implementação das teclas.

FIGURA 13 Composição do código de varredura.



A Fig.13 mostra a composição do código, baseado em dois campos distintos, sendo o primeiro utilizado para identificação de linhas e o segundo para identificação de colunas. A composição do código de varredura obedece a disposição das teclas dentro da interface, partindo da posição superior esquerda para inferior direita.

Com as teclas corretamente identificadas, é possível implementar o algoritmo responsável pela varredura automática das teclas. Este algoritmo é implementado dentro do evento *OnTimer* de um objeto da classe *TTimer*, de forma que o mesmo seja executado periodicamente dentro de um intervalo que controla a velocidade da varredura.

O sistema possui 3 formas diferentes de varredura, para linhas, colunas e uma forma especial criada para a seleção de palavras retornadas pelo recurso de predição de palavras.

#### 6.2.1. Varredura de linhas.

Esta varredura efetua a seleção de linhas utilizando como base as propriedades *Lines* e *ActiveLine*, sendo definida pelo valor *smLine* da propriedade *ScanMode*. Uma das funções do algoritmo é atualizar a posição do objeto imagem especificado na propriedade *LineSelector*, de maneira que este sinalize para o usuário o andamento da varredura:

```

...
if ActiveLine = Lines then
  begin
    LineSelector.Top := 9;
    ActiveLine := 1;
  end
else
  begin
    LineSelector.Top := LineSelector.Top + 32;
    Inc(ActiveLine);
  end;
...

```

A segunda função do algoritmo é a de atualizar o valor das propriedades *ActiveColumn* e *Columns*, de maneira a preparar as informações para uma eventual transição para a varredura de colunas:

```

...
if Keyboard = pnlAlfa then
  case ActiveLine of

```

```

1: if alfPred.Enabled then
    Columns := 16
else
    Columns := 15;
3: Columns := 13;
5: Columns := 6;
else Columns := 14;
end;
if Keyboard = pnlNumerico then
    case ActiveLine of
        5: Columns := 4;
        else Columns := 5;
    end;
...
ActiveColumn := ActiveLine * 100;
...

```

A atualização da propriedade *Columns* se faz necessária, pelo fato de que em alguns casos, as linhas de um teclado não possuem as mesmas quantidades de teclas.

Observe a alteração na propriedade *ActiveColumn*, necessária para a o funcionamento correto da varredura de colunas.

### 6.2.2. Varredura de colunas.

Nesta varredura o algoritmo encarrega-se de localizar dentre as teclas do teclado ativo, aquela que deve ser selecionada. É definida pelo valor *smColumn* da propriedade *ScanMode*.

Para efetuar a seleção de teclas, o algoritmo utiliza a informação contida na propriedade *ActiveColumn* e no código de varredura indicado na propriedade *Tag* das teclas:

...

```

for Cnt:= 0 to Keyboard.ControlCount - 1 do
  if Keyboard.Controls[Cnt].Tag = ActiveColumn + 1 then
    begin
      ActiveKey:= Keyboard.Controls[Cnt];
      ActiveColumn:= ActiveKey.Tag;
      DrawColumn;
      ...
      Exit;
    end;
end;

```

O procedimento *DrawColumn* se encarrega posicionar o objeto imagem especificado em *ColumnSelector*, de maneira que o usuário possa então visualizar a tecla que esta sendo selecionada.

O algoritmo utiliza a informação contida na propriedade *Columns* para detectar o momento de retomar a varredura de linhas, garantindo que o usuário possa corrigir um erro cometido durante a seleção de linhas:

```

...
if ActiveColumn >= Columns + (ActiveLine * 100) then
  begin
    ScanMode:= smLine;
    ColumnSelector.SetBounds(630, 140, 18, 18);
    ColumnSelector.Picture.Bitmap.FreeImage;
  end;
...

```

O algoritmo se encarrega de apagar a indicação efetuada pelo elemento *ColumnSelector*, preparando-o para o próximo ciclo da varredura de colunas, através do método *FreeImage*.

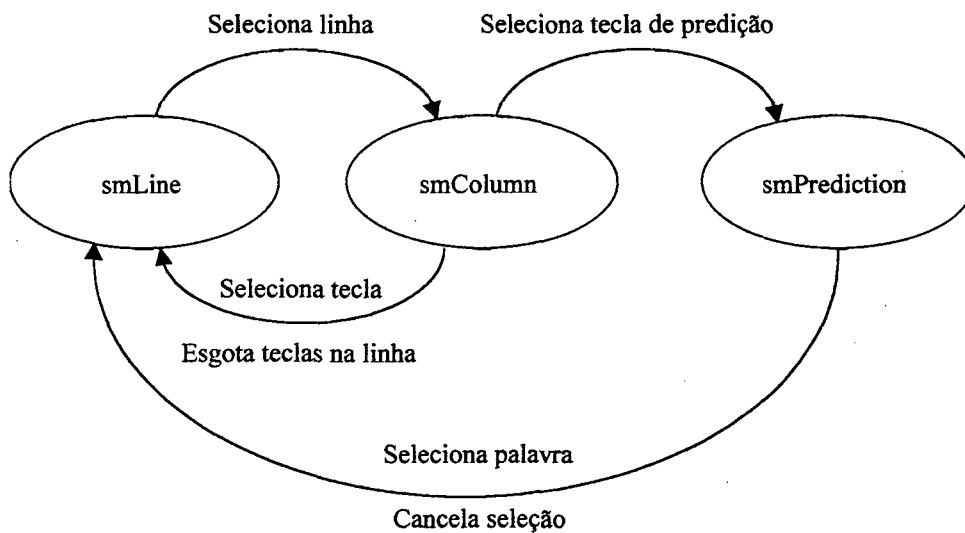
### 6.2.3. Varredura especial para predição de palavras.

Esta varredura é semelhante a varredura de colunas, sendo usada exclusivamente para uma área específica da interface, referente aos elementos responsáveis pela seleção de palavras filtradas pelo recurso de predição de palavras.

Esta varredura é ativada quando o usuário aciona a tecla de predição de palavras no teclado alfanumérico do sistema, e desativada quando o usuário cancela ou confirma a seleção de uma palavra contida na lista de palavras filtradas. É definida através do valor *smPrediction* na propriedade *ScanMode* da classe *TVirtualKeyBoard*.

A Fig. 14 representa a transição das formas de varredura, através dos estados referentes a propriedade *ScanMode*.

**FIGURA 14** Transição entre os modos de varredura.



### 6.3 SIMULANDO A DIGITAÇÃO DAS TECLAS

A digitação das teclas do simulador de teclado, foi implementada a partir da função *keybd\_event*, disponível na Microsoft Win32® Application Program Interface (API). A função *keybd\_event* é utilizada para simular o acionamento de uma tecla, gerando uma mensagem *WM\_KEYDOWN* para indicar o que uma tecla foi pressionada, ou uma mensagem *WM\_KEYUP* para indicar que uma determinada tecla foi liberada.



O protótipo para a chamada da função *keybd\_event* na API do Windows é o seguinte, NORTON (14):

```
VOID keybd_event(
    BYTE bVk,           // código virtual da tecla desejada
    BYTE bScan,        // código de varredura de hardware da tecla desejada
    DWORD dwFlags,     // flags para definição de opções
    DWORD dwExtraInfo // dados adicionais
);
```

Os parâmetros *bVk* e *bScan* podem ser utilizados de forma mutuamente exclusiva, servindo para a identificação da tecla a ser acionada, já o parâmetro *dwFlags* é utilizado para definir se a tecla deve ser pressionada ou liberada.

O sistema Tadeu, utiliza a função *keybd\_event* indiretamente, através de dois procedimentos distintos.

O procedimento *TypeChar* é utilizado para a digitação de caracteres, recebendo a propriedade *Caption* do controle *ActiveKey* como parâmetro, e convertendo-o para o código virtual da tecla correspondente, através da função *VkKeyScan*, disponível na API do Windows.

```
procedure TypeChar(lpszKey: PChar);
begin
    ...
    keybd_event(VkKeyScan(Char(lpszKey)), 0, 0, 0);
    keybd_event(VkKeyScan(Char(lpszKey)), 0, KEYEVENTF_KEYUP, 0);
end;
```

Um segundo procedimento *TypeByte*, foi criado para a digitação de teclas especiais, como Delete, Enter e Alt, recebendo como parâmetro um número usado como índice de acesso ao array *SpecialKey*, que contem os códigos virtuais das teclas especiais.

```

procedure TypeByte(nKey: Integer);
begin
    ...
    keybd_event(SpecialKey[nKey], 0, 0, 0);
    keybd_event(SpecialKey[nKey], 0, KEYEVENTF_KEYUP, 0);
end;

```

Além destes dois procedimentos foram criados outros dois chamados *HoldKey* e *UnHoldKey*, que nada mais são do que variações do procedimento *TypeByte*, usados para manter uma tecla pressionada e liberar uma tecla pressionada respectivamente.

Para permitir o acionamento das teclas através de seleção indireta, foi desenvolvido o procedimento *ClickKey*, através do qual é possível gerar um evento de clique sobre a tecla selecionada, sem a utilização do mouse ou qualquer dispositivo apontador:

```

procedure TfrmPrincipal.ClickKey;
begin
    ...
    ActiveKey.Invalidate;
    ActiveKey.Perform(WM_LBUTTONDOWN, 0, 0);
    ActiveKey.Invalidate;
    ActiveKey.Perform(WM_LBUTTONUP, 0, 0);
    ...
end;

```

A base do procedimento são os métodos *Perform* e *Invalidate*, disponíveis na classe *TControl* contida na VCL do Delphi 3.0. O método *Perform* permite que um controle responda como se tivesse recebido uma mensagem gerada pelo ambiente Windows. O método *Invalidate* é utilizado para atualizar um controle, depois que este recebe uma mensagem.

No caso do procedimento *ClickKey*, o método *Perform* é utilizado para gerar sobre o objeto *ActiveKey* as mensagens *WM\_LBUTTONDOWN* e *WM\_LBUTTONUP*, que

geradas nesta ordem simulam um evento de clique com o botão esquerdo do mouse.

## 6.4 ACIONANDO O MOUSE

### 6.4.1. Utilizando os Hooks.

Segundo MARSH (11), um hook é um mecanismo promovido pelo ambiente Microsoft Windows ®, através do qual uma função é capaz de interceptar um evento, antes do mesmo alcançar o seu destino. As funções que recebem estes eventos são chamadas de funções filtro e podem ser classificadas de acordo com o tipo de evento que processam, as principais são:

- **WH\_MSGFILTER:** é capaz de processar ou modificar todas as mensagens referentes a caixas de diálogo, caixas de mensagem, barras de rolagem e menus, dentro de uma aplicação;
- **WH\_SYSMSGFILTER:** capaz de processar ou modificar todas as mensagens referentes a caixas de diálogo, caixas de mensagem, barras de rolagem e menus, dentro do Windows;
- **WH\_GETMESSAGE:** capaz de processar ou modificar qualquer mensagem do ambiente, sempre que as funções GetMessage e PeekMessage são chamadas;
- **WH\_CALLWNDPROC:** processa ou modifica qualquer mensagem sempre que a função SendMessage é chamada;
- **WH\_JOURNALRECORD:** capaz de gravar qualquer evento de teclado ou mouse;
- **WH\_JOURNALPLAYBACK:** capaz de reproduzir eventos gravados com uma função filtro WH\_JOURNALRECORD;
- **WH\_KEYBOARD:** capaz de remover, processar ou modificar eventos de teclado;
- **WH\_MOUSE:** capaz de remover, processar ou modificar eventos de mouse;

- WH\_CBT: responde a determinadas ações do sistema, permitindo o chamado treinamento baseado em computador (*computer-based training CBT*);
- WH\_DEBUG: monitora outros hooks já instalados.

Todas as funções filtro possuem o mesmo protótipo, NORTON (14):

```
LRESULT CALLBACK FilterFunc(
int nCode,           // código que identifica o hook
Word wParam,        // flag do processo corrente
Dword lParam        // endereço da estrutura com dados da mensagem
);
```

A base para a utilização dos hooks, são as funções *SetWindowsHookEx*, *UnhookWindowsHookEx* e *CallNextHookEx*, também pertencentes a API do Windows.

A função *SetWindowsHookEx* permite a associação de uma função filtro, contida em um aplicativo ou dll, a um determinado tipo de hook. Todo o controle de encadeamento das funções filtro é feito internamente pelo Windows, de maneira que os aplicativos não necessitam se preocupar com questões de endereçamento, Fig. 15.

Uma vez que a função filtro é instalada, a função *CallNextHookEx* deve ser usada para a chamada da próxima função filtro encadeada, o que permite a aplicação de recursividade.

Finalmente para a remoção de uma função da corrente do hook, é utilizada a função *UnhookWindowsHookEx*.

Este recurso foi de extrema importância no simulador de teclado, permitindo o êxito na implementação do recurso de acionamento de mouse via macros.

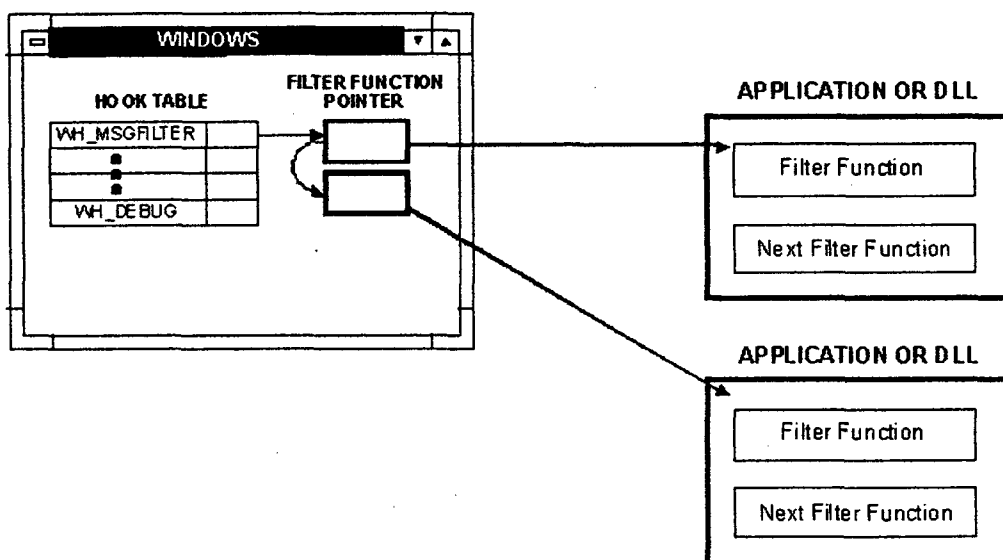
#### 6.4.2. Macros de mouse.

O recurso de macros de mouse permite que procedimentos executados através do mouse, possam ser gravados em arquivos, permitindo que os mesmos possam ser reproduzidos futuramente pelos usuários.

Este recurso necessita de dois estágios, um estágio responsável pela gravação das macros contendo os comandos e um segundo estágio que efetua a reprodução das macros gravadas.

O registro dos procedimentos de mouse em um arquivo, é feito através do monitoramento das mensagens geradas pelo mouse dentro do ambiente Windows, utilizando uma função filtro do tipo WH\_JOURNALRECORD.

**FIGURA 15** Encadeamento das funções filtro implementada a partir do Windows 3.1



Fonte: MARSH, Kyle. Win32 Hooks. Disponível na Internet. [http://msdn.microsoft.com/library/techart/msdn\\_hooks32.htm](http://msdn.microsoft.com/library/techart/msdn_hooks32.htm). 20 abril 2001.

A função criada, monitora todas as mensagens de mouse e registra as informações em um arquivo, que no término do procedimento é associado a uma das 12 teclas (M1 à M12) que o simulador de teclado oferece para a reprodução das macros. A associação é feita através da atribuição do nome do arquivo, utilizando-se o mesmo nome da tecla que deverá utilizá-lo (e.g. a tecla identificada pelo rótulo M1 utiliza o arquivo chamado M1.~tms).

A reprodução das macros é feita através de uma função filtro do tipo WH\_JOURNALPLAYBACK, que lê as informações contidas no arquivo e reproduz todos os eventos armazenados.

## 6.5 PREDIÇÃO DE PALAVRAS

### 6.5.1. Filtrando as palavras.

O recurso de predição de palavras, permite que o usuário selecione palavras obtidas de dicionários, a medida em que seleciona as teclas do sistema, fazendo com que este digite os caracteres restantes para o término da palavra.

Para isto é necessário, inicialmente, que o simulador de teclado consiga filtrar dentro dos dicionários selecionados, todas as palavras que iniciem com os caracteres digitados pelo usuário até um determinado momento.

O procedimento *GetWords* foi criado com esta finalidade, utilizando as propriedades *TypeWritten*, *Accent* e *WordsDictionary* do objeto *vkbTadeu*, usado para controlar o sistema simulador de teclado. A propriedade *Accent* é utilizada para armazenar o último acento digitado pelo usuário, permitindo assim a busca de palavras acentuadas.

O núcleo do procedimento utiliza uma cláusula do tipo *for* para realizar a busca:

```

procedure TfrmPrincipal.GetWords(Letra: Char);
  var
    Cnt: Integer; Inicio: String;
  begin
    ...
    vkbTadeu.TypeWritten := vkbTadeu.TypeWritten + Letra;
    for Cnt := 0 to vkbTadeu.WordsDictionary.Count - 1 do
      begin
        Inicio := Copy(vkbTadeu.WordsDictionary.Strings[Cnt], 0,
          Length(vkbTadeu.TypeWritten));
        if StrComp(PChar(vkbTadeu.TypeWritten), PChar(Inicio)) = 0 then
          lstPalavras.Items.Add(vkbTadeu.WordsDictionary.Strings[Cnt]);
        end;
      end;
    end;
  
```

As palavras obtidas através da busca, são armazenadas na propriedade *Items* da lista *lstPalavras*, permitindo que o usuário possa selecionar alguma das palavras.

A propriedade *TypeWritten* possui seu conteúdo apagado, sempre que o usuário digita caracteres especiais, como pontos, vírgulas e espaços, permitindo o perfeito funcionamento do procedimento *GetWords*.

A propriedade *WordsDictionary* é instanciada, no início da execução do programa, com todas as palavras contidas em todos os dicionários selecionados através do editor de propriedades do sistema. O procedimento que insere as palavras em *WordsDictionary*, impede a ocorrência de palavras repetidas, evitando duplicidade no resultado da busca.

#### 6.5.2. Digitando as palavras.

Ao selecionar uma palavra e confirmar a seleção através do comando apropriado, o usuário dispara o procedimento *TypeWord*, que digita os caracteres finais da palavra selecionada, com base no que já foi digitado pelo usuário, armazenado na propriedade *TypeWritten*.

O procedimento *TypeWord* se assemelha ao procedimento *TypeChar* visto anteriormente, com a diferença que o procedimento *TypeWord* digita vários caracteres ao invés de um único, além de decompor os caracteres acentuados, que correspondem a dois caracteres digitados pelo usuário.

## 6.6 COMUNICAÇÃO COM HARDWARE

A capacidade do sistema em estabelecer comunicação com determinados dispositivos de hardware, permite que o mesmo se torne um sistema multimodal, isto é capaz de ser operado de diferentes modos. Para o simulador de teclado desenvolvido, foram contempladas as comunicações com a porta serial e com o microfone.

#### 6.6.1. Porta Serial.

As portas seriais podem ser utilizadas pelo simulador de teclado, para se comunicar com dispositivos acionadores especiais.

Para que o sistema possa efetuar esta comunicação, inicialmente é feita uma espécie de validação, para identificar se uma porta está disponível para a comunicação. Dentro da Win32 (nome dado a API de 32 bits do Windows) as portas seriais são manipuladas como arquivos, de modo que para acessá-las é necessário usar a função *CreateFile* da API, DENVER (6).

Em caso de uma porta não estar disponível no computador, ou já estiver sendo utilizada, a função *CreateFile* retorna a constante `INVALID_HANDLE_VALUE`.

Neste ponto não existe forma de distinguir uma porta inexistente de uma porta ocupada, já que o valor retornado pela função, em ambos os casos, é o mesmo. Deste modo optou-se simplesmente, por utilizar esta validação para notificar o operador de que, na melhor das hipóteses, a porta serial indicada pode estar sendo utilizada por outro processo ou dispositivo.

Esta medida toma como base o fato de que muitas vezes, haverá a necessidade de se compartilhar uma porta serial entre o acionador especial e outro dispositivo já instalado.

Para permitir o acesso absoluto a uma porta serial, foi utilizado o recurso oferecido pelo ambiente Delphi, que permite criar procedimentos em linguagem Assembler, dentro do código Object Pascal.

Estes procedimentos utilizam os endereços de memória das portas seriais para acessar e manipular os seus conteúdos diretamente, permitindo que o sistema trabalhe com uma porta serial que esteja sendo ocupada dentro do ambiente Windows.

Estes procedimentos Assembler foram reunidos para formar um driver, usado para a comunicação entre o simulador de teclado e um acionador serial.

#### 6.6.2. Microfone.

O microfone é manipulado, a fim de permitir o acionamento das teclas através da emissão de sons.

Basicamente, para que o sistema acesse o microfone é necessário a utilização da função *waveInOpen* contida na API do Windows, NORTON (14):



```
MMRESULT waveInOpen(  
LPHWAVEIN phwi, // manipulador do dispositivo  
UINT uDeviceID, // identificador do dispositivo  
LPWAVEFORMATEX pwfx, // estrutura que define o formato de áudio usado  
DWORD dwCallback, // destino das notificações de mensagem  
DWORD dwCallbackInstance, // instancia do usuário mecanismo de chamada  
DWORD fdwOpen // flag definindo forma de abertura  
);
```

Através do manipulador de dispositivo *phwi*, retornado pela função *waveInOpen*, é possível preparar o dispositivo e então iniciar o monitoramento, através da função *waveInStart*. Com a chamada desta função a aplicação passa a receber mensagens do tipo *MM\_WIM\_DATA*, que trazem em seus componentes informações referentes a entrada de dados pelo microfone.

Uma vez que o simulador de teclado é capaz de receber e manipular tais mensagens, é possível operar a interface através do microfone, fazendo-se o ajuste de volume para obtenção da sensibilidade adequada.

## **7. DESENVOLVENDO DISPOSITIVOS ACIONADORES**

Uma vez que o simulador de teclado é capaz de monitorar a porta serial de comunicação, é possível criar dispositivos capazes de acionar suas teclas, ampliando com isso suas possibilidades de utilização.

Este capítulo mostra as características da comunicação entre o sistema desenvolvido e um dispositivo serial, juntamente com o dispositivo acionador desenvolvido durante o trabalho.

### **7.1 DISPOSITIVOS DE ENTRADA DE DADOS**

Segundo PASTOR (16), a relação com os computadores está baseada na entrada dos dados através de algum dispositivo, que estabelece o contato e permite que o computador receba mensagens.

É possível agrupar a grande diversidade de dispositivos de entrada de dados em dois grandes grupos, os de acesso direto e indireto.

#### **7.1.1. Dispositivos de acesso direto.**

Grupo constituído por dispositivos convencionais como os teclados e mouse, e por adaptações que permitam a utilização dos mesmos, como carcaças, que são coberturas colocadas no teclado permitindo o aumento da precisão na pressão e manoplas que auxiliam no controle de movimentos involuntários.

Existem dispositivos de acesso direto muito mais sofisticados, como o dispositivo que permite o controle do mouse via infravermelho, baseado na movimentação da cabeça, apresentado por EVANS (8).

#### **7.1.2. Dispositivos de acesso indireto.**

Os dispositivos de acesso indireto permitem que os sinais emitidos pelo usuário sejam decodificados através de programas específicos, permitindo que os mesmos possam ser utilizados pelo computador.

Dentre os dispositivos de acesso indireto estão os dispositivos de acesso binário, que operam a partir de comutadores, permitindo ou não a passagem do sinal mediante um estímulo do usuário.

Existe uma grande quantidade de dispositivos comutadores, controlados através de pressão, posição, tato, proximidade, luz e diversas outras formas, permitindo que qualquer pessoa utilizando um movimento voluntário possa operá-los.

Este tipo de dispositivo de entrada, pode ser utilizado para acionar as teclas do simulador de teclado Tadeu, através de uma porta de comunicação serial.

## 7.2 COMUNICAÇÃO SERIAL

### 7.2.1. O padrão RS232.

Segundo STRANGIO (18), o padrão RS232 (*Recommended Standard 232*) foi criado na década de 60, com a finalidade de permitir a segurança durante a comunicação serial, e estabelecer a compatibilidade entre equipamentos de diferentes fabricantes.

O padrão RS232 é utilizado para a comunicação serial entre um Data Terminal Equipment (*DTE*), geralmente um computador ou terminal, e um Data Circuit-terminating Equipment (*DCE*), que pode ser um modem, mouse ou qualquer outro dispositivo.

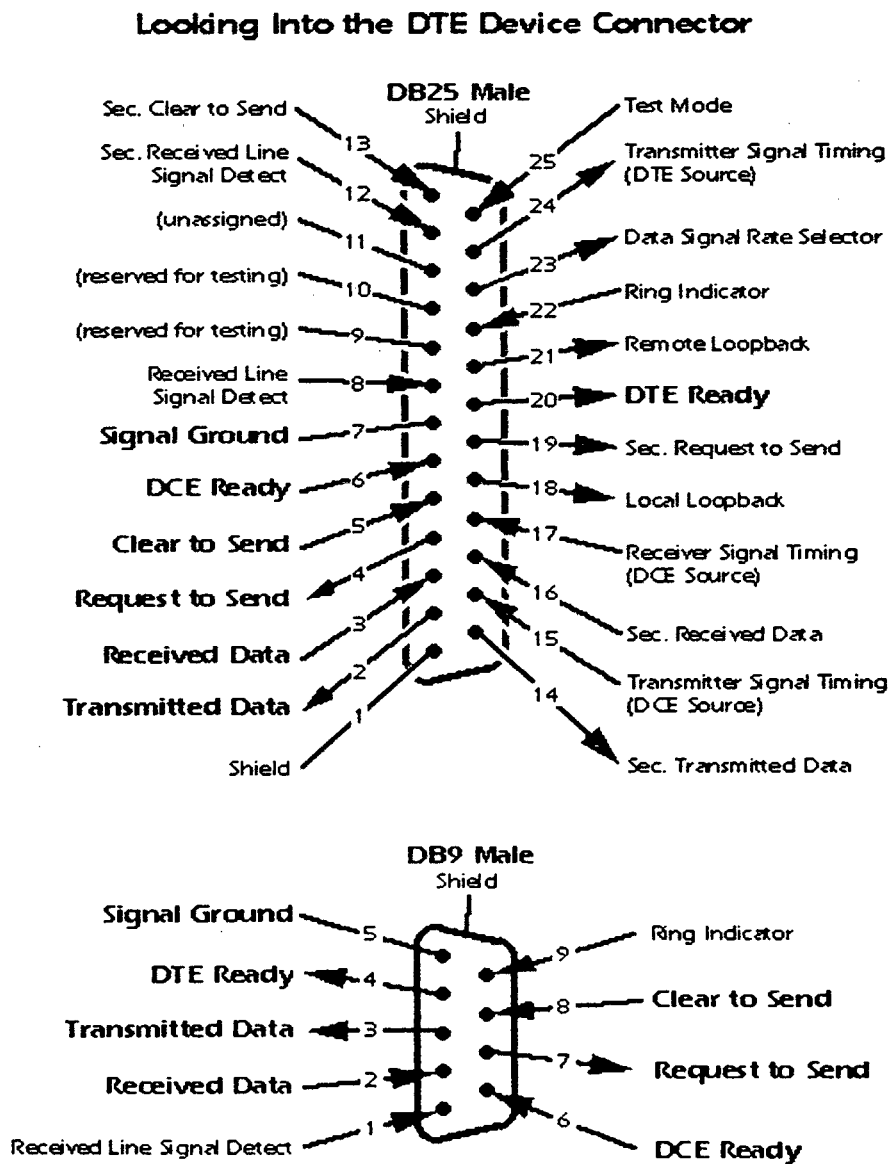
O padrão RS232 estabelece características referentes aos níveis de tensão, tempo, função dos sinais, protocolo para troca de dados e conexão mecânica para uma comunicação serial.

Existem padrões mais recentes, como o EIA232 (*Electronic Industries Association 232*), que apresentam pequenas alterações em relação ao padrão RS232 original.

### 7.2.2. A porta serial.

As portas de comunicação serial dos computadores podem possuir 25 ou 9 pinos, sendo que cada um destes pinos possui uma função dentro do processo de comunicação, como mostra a Fig. 16.

FIGURA 16 Pinagem das portas seriais 25 pinos e 9 pinos.



Fonte: STRANGIO, Christopher E. The RS232 Standard. Disponível na Internet. [http://www.camiresearch.com/Data\\_Com\\_Basics/RS232\\_standard.html](http://www.camiresearch.com/Data_Com_Basics/RS232_standard.html). 10 julho 2001.

Segundo SHUFORD (17), os pinos mais importantes para uma comunicação serial, podem ser divididos de acordo com suas funções da seguinte forma:

- Received Data (*RxD*) e Transmitted Data (*TxD*): são as linhas portadoras de dados, usadas para recepção e transmissão de dados, respectivamente;
- Request to Send (*RTS*) e Clear to Send (*CTS*): usados para iniciar ou interromper a comunicação, controlando o fluxo de dados;
- DTE Ready ou Data Terminal Ready (*DTR*) e DCE Ready ou Data Set Ready (*DSR*): usados no início da comunicação para indicar a presença dos dispositivos DTE e DCE, respectivamente;
- Received Line Signal Detect ou Data Carrier Detect (*DCD*): usado pelo modem para indicar a detecção do sinal de chamada na linha telefônica;
- Ring Indicator (*RI*): usado pelo modem para indicar que ocorreu uma chamada;
- Signal Ground (*GND*): ponto para referência de tensão dos demais pinos. Para o padrão RS232 o nível lógico 1, corresponde a um sinal elétrico de -12 volts e o nível lógico 0, a um sinal de +12 volts, ambos os níveis em valores típicos de tensão;

Para as linhas de status, que incluem todas as linhas acima, exceto as linhas *RxD*, *TxD* e *GND*, o nível lógico 0 indica o estado ativo.

## 7.3 COMUNICAÇÃO COM O SISTEMA

### 7.3.1. Conexão.

Uma comunicação serial é tradicionalmente utilizada para efetuar a transmissão / recepção de um grande volume de bits, como ocorre no caso dos modems, no entanto é possível utilizar esta comunicação para transmitir um único bit.

Esta foi exatamente a forma adotada para permitir que um dispositivo de seleção binária, indique ao simulador de teclado, a ocorrência de uma ativação por parte do usuário.

O padrão criado para a conexão entre o dispositivo serial e o sistema, trabalha com 5 pinos, que oferecem todos os recursos necessários para a transmissão do bit de estado, tanto para portas seriais de 25 pinos, quanto para portas seriais de 9 pinos, como mostra a Tabela 3.

**TABELA 3** Padrão para a conexão serial com o sistema.

Pino	Função	Sentido
DTR	Alimentação positiva (nível 0).	DTE → DCE
RTS	Alimentação positiva (nível 0).	DTE → DCE
TxD	Alimentação negativa (nível 1).	DTE → DCE
CTS	Via de indicação do acionamento.	DTE ← DCE
GND	Referência de tensão.	-

### 7.3.2. Driver.

O sistema conta com um programa conhecido como *driver*, utilizado para efetuar o controle lógico de conexões entre dispositivos de hardware, efetuando o processamento dos dados presentes nestas conexões .

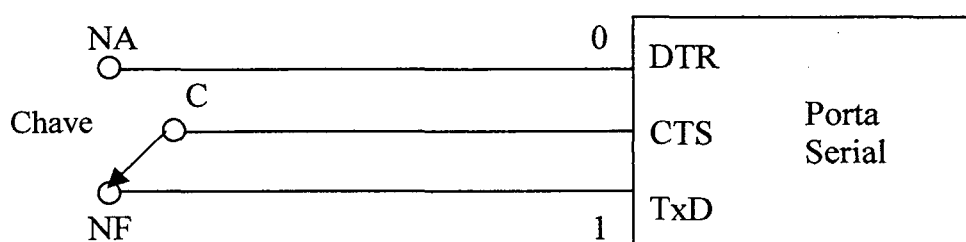
O driver do sistema permite monitorar um dispositivo conectado a porta serial, desde que este obedeça o padrão de conexão estabelecido. O driver possui como principais funções, garantir sinal de alimentação para o acionador, atuando nos pinos DTR e RTS, e monitorar o bit presente no pino CTS. O monitoramento do pino CTS é feito periodicamente, em intervalos de 250 ms.

### 7.3.3. Princípio geral do acionamento.

De posse das características estabelecidas para conexão e controle lógico, é possível demonstrar o princípio geral do acionamento do sistema através da porta serial, baseado na comutação de estado do pino CTS.

A Fig. 17 ilustra um dispositivo acionador bastante simples, composto unicamente por uma chave elétrica de um pólo e duas posições, inspirado no modelo apresentado em UFRGS (20).

**FIGURA 17** Acionador feito com uma chave elétrica.



A chave possui um terminal comum (C), um terminal normalmente fechado (NF) e um terminal normalmente aberto (NA).

Quando a chave se encontra em repouso, a conexão dos pinos CTS e TxD da porta serial é mantida através dos terminais C e NF da chave, isto garante que o pino CTS apresente o nível lógico 1, indicando que o acionador não está ativado.

Ao se acionar a chave, ocorre a conexão entre os pinos CTS e DTR através dos terminais C e NA da chave, fazendo com que o pino CTS passe a apresentar o nível lógico 0, indicando que o acionador foi ativado.

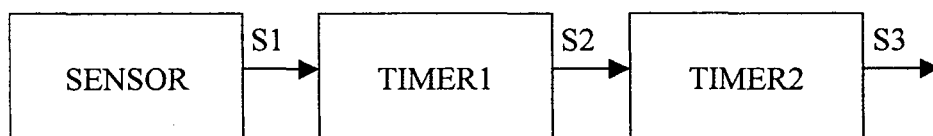
Embora este seja o princípio geral do funcionamento de qualquer acionador que trabalhe com o sistema, é possível criar soluções mais elaboradas de chaveamento.

#### 7.4 DISPOSITIVO ACIONADOR POR PROXIMIDADE

Na tentativa de oferecer um dispositivo que opere de forma mais sofisticada, foi desenvolvido durante o trabalho um acionador por proximidade, que permite que o usuário ative-o através da aproximação de alguma parte do corpo.

O dispositivo conta ainda com uma estratégia de comutação, que permite que usuários com interferência no controle, em virtude de movimentos involuntários, possam operá-lo normalmente. A Fig. 18 ilustra o diagrama de blocos do acionador.

**FIGURA 18** Diagrama de blocos do acionador por proximidade.



O bloco Sensor representa um sensor de campo eletromagnético, que detecta através de uma minúscula antena a presença de ondas eletromagnéticas induzidas no corpo humano. Este fenômeno ocorre pelo fato do corpo humano ser rico em água e líquidos, elementos que conduzem facilmente a corrente elétrica, o que permite que pessoas expostas a campos eletromagnéticos, provenientes de condutores localizados em edificações e até mesmo nas ruas, emanem um sinal eletromagnético de baixíssima potência, mas capaz de ser detectado pelo circuito do bloco Sensor.

O circuito presente no bloco Sensor possui sensibilidade para detectar campos eletromagnéticos a uma distância média de 5cm, bem como o toque em sua antena.

O Sensor envia o sinal captado e amplificado para a entrada de Timer1, que dispara e inicia uma temporização, período em que sua saída S2 é ativada apresentando nível de tensão. A temporização de Timer1 pode ser ajustada externamente no intervalo de 0 a 20 segundos.

Assim que Timer1 dispara, este envia um sinal até a entrada de Timer2 que também dispara e inicia sua temporização, fazendo com que sua saída S3 seja ativada por um período de 100 ms. O sinal proveniente de S3 é enviado então para o pino CTS da porta serial, que passa a apresentar nível lógico 0 indicando a ativação do dispositivo. Ao término do intervalo de 100 ms, a saída de Timer2 é desativada, fazendo com que o pino CTS apresente nível lógico 1, indicando a inatividade do dispositivo.

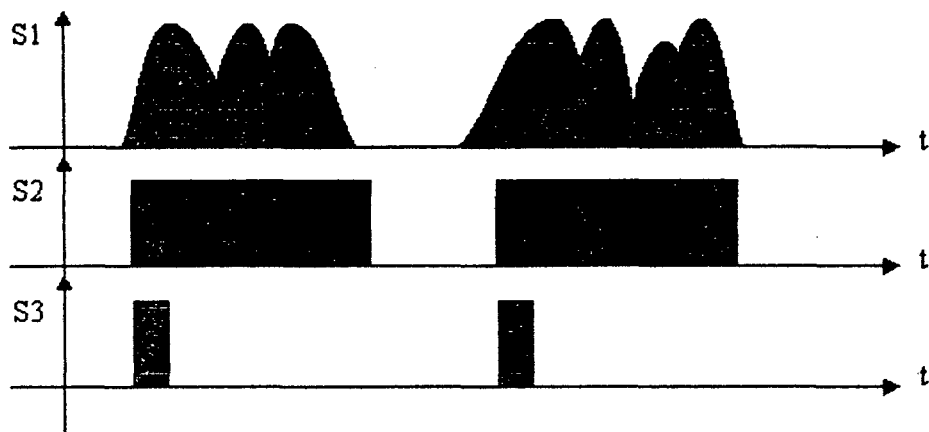
O temporizador Timer2 só poderá ser disparado novamente quando o temporizador Timer1 encerrar seu ciclo de temporização. A presença de Timer1 permite que oscilações na entrada do Sensor, não se perpetuem até a saída de Timer2, isto



ocorre porque o “gatilho” de disparo do temporizador uma vez ativado, só pode ser reativado quando o circuito terminar o seu ciclo de temporização, Fig. 19.

Uma vez que o intervalo de Timer1 é ajustado para zero, sua presença se torna nula, fazendo com que Timer2 seja disparado sempre que o Sensor detectar a aproximação de algum elemento.

**FIGURA 19** Filtragem dos movimentos involuntários pelo acionador.



## 8. RESULTADOS

Os resultados obtidos até o presente momento são favoráveis ao desenho adotado na interface e aos recursos implementados no sistema, embora existam pequenos problemas de adaptação.

A principal dificuldade encontrada pelos usuários é operar comodamente a varredura de teclas, em função da impossibilidade do uso de um acionador mais adequado. A dificuldade em se construir ou adquirir um simples acionador binário, fica evidente em uma das solicitações feitas, sugerindo que o mouse fosse utilizado como dispositivo de acesso indireto, permitindo assim o acionamento da varredura.

O dispositivo acionador proposto, foi implementado com sucesso, no entanto seu sensor de proximidade apresentou baixa sensibilidade, tornando difícil a operação do dispositivo por usuários com comprometimento motor. O circuito do sensor está passando por uma reestruturação, que possibilite um arranjo mais sensível, mantendo o dispositivo simples e barato de modo a torná-lo acessível aos usuários.

Por se tratar de uma tecnologia assistiva desenvolvida para atender uma ampla faixa de pessoas, os problemas particulares não podem ser priorizados, sendo que resultados mais expressivos, só poderão ser obtidos a partir do uso mais disseminado do sistema e da incidência de problemas comuns reportados pelos usuários.

## 9. CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

O sistema simulador de teclado desenvolvido no trabalho mostrou-se bastante adequado ao seu público de usuários, principalmente em virtude de sua interface com desenho simplificado e objetivo, que pode ser operada sem exigir das habilidades sensoriais, físicas e cognitivas dos usuários.

Considerando-se puramente a limitação motora dos usuários, fica evidenciada a necessidade dos sistemas simuladores de teclado em trabalhar conjuntamente com dispositivos alternativos para entrada de dados, aptos a trabalhar com os poucos recursos motores dos deficientes físicos e sem os quais os simuladores de teclado perdem significativa flexibilidade. A dificuldade dos usuários e instituições em adquirirem dispositivos adequados as suas necessidades ainda é um obstáculo, que pode ser transposto através de incentivos e parcerias que possibilitem a oferta de dispositivos baratos e acessíveis.

Se considerar-mos ainda as possíveis limitações sensoriais apresentadas pelos usuários, fica evidenciada, ainda que de forma secundária, a necessidade dos simuladores de teclado em trabalhar paralelamente com outras tecnologias assistivas, como os sistemas magnificadores de tela.

O trabalho mostrou como pode ser simples a criação de alternativas tecnológicas, capazes de atender com qualidade as pessoas portadoras de deficiência. Muitos dos avanços conseguidos nesta área, trarão inúmeros progressos para a computação, oferecendo aos deficientes físicos a possibilidade de desempenhar atividades para as quais encontram-se incapacitados e proporcionando ao homem tecnologias inovadoras, capazes de serem assimiladas por todos no desempenho de tarefas até então inimagináveis.

Como sugestões para trabalhos futuros, recomendo a implementação de um recurso que permita o uso de múltiplos padrões para o teclado alfanumérico, privilegiando o padrão DVORAK e suas variantes. Esta medida permitiria um aumento bastante significativo na adaptabilidade e flexibilidade do sistema.

Um recurso de varredura sonora de teclas, permitiria que usuários com comprometimento na visão pudessem interagir mais facilmente com o sistema, sem a necessidade do apoio de sistemas magnificadores de tela ou lupas. Outra possibilidade bastante interessante para usuários com comprometimento visual, seria o desenvolvimento de uma varredura de teclas com o recurso de lupa embutido, ampliando as dimensões das teclas a medida em que estas fossem selecionadas.

A criação de um teclado capaz de operar o cursor do mouse, seria uma estratégia de emulação bastante flexível, sem a limitação de uma quantidade definida de comandos, como ocorre no caso das macros. O desenvolvimento de dispositivos acionadores, também pode ser explorado, como a criação de acionadores baseados em sensores infravermelhos.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] BIELER, Rosângela Bermam. **Inclusão**. Disponível na Internet. <http://iee.inf.ufsc.br>. 20 outubro 2000.
- [2] BRANCO, Pedro. **Challenges for Multi-Modal Interfaces Towards Anyone Anywhere Accessibility: A Position Paper**. In: Workshop on Universal Accessibility of Ubiquitous Computing: Providing for the Elderly, 1, 2001.
- [3] CARVALHO, Faíçal Farhat de. **Programação orientada a objetos usando Delphi 3. 2. ed.** São Paulo: Érica, 1998. 505p.
- [4] CYBIS, Walter de Abreu. **Ergonomia de Interfaces Humano - Computador**. Disponível na Internet. <http://www.labiutil.inf.ufsc.br/apostila/apostila.htm>. 21 junho 2000.
- [5] DATI. **FAQ: Frequently Asked Questions**. Disponível na Internet. <http://www.asel.udel.edu/dati/atdef.html>. 20 março 2001.
- [6] DENVER, Allen. **Serial Communication in Win32**. Disponível na Internet. [http://msdn.microsoft.com/library/techart/msdn\\_serial.htm](http://msdn.microsoft.com/library/techart/msdn_serial.htm). 04 junho 2001.
- [7] DUL, J., WEERDMEEESTER, B. **Ergonomia Prática**. São Paulo, Editora Edgard Blücher LTDA, 1998, 147p. Informação e Operação, p.55-84.
- [8] EVANS, Gareth D., DREW, Roger, BLENKHORN, Paul. Controlling Mouse Pointer Position Using an Infrared Head-Operated Joystick. **IEEE Transactions on Rehabilitation Engineering**, v. 8, n.º 1, p.107-117, março 2000.

- [9] KOON, Ricardo A, VEGA, Maria Eugenia de la. **El impacto tecnológico en las personas com discapacidad.** In: Congreso Internacional de Informática Educativa Especial, 2, 2000.
- [10] LOPES, João B. **Designing User Interfaces for Severely Handicapped Persons.** In: Workshop on Universal Accessibility of Ubiquitous Computing: Providing for the Elderly, 1, 2001.
- [11] MARSH, Kyle. **Win32 Hooks.** Disponível na Internet. [http://msdn.microsoft.com/library/techart/msdn\\_hooks32.htm](http://msdn.microsoft.com/library/techart/msdn_hooks32.htm). 20 abril 2001.
- [12] MONTOYA, Rafael S. **Los sistemas de ayudas basados en la tecnología de la información.** In: Congreso Internacional de Informática Educativa Especial, 1, 1998, Neuquén – Argentina.
- [13] NEWELL, Alan F, GREGOR, Peter. Human Computer Interfaces for People with Disabilities. In: ELSEVIER SCIENCE. **Handbook of Human-Computer Interaction.** 1997, p.813-824.
- [14] NORTON, Peter, YAO, Paul. **Programando em Borland C++ para Windows.** Tradução por Kátia de Almeida Guimarães. Rio de Janeiro: Berkely, 1992. 584p. Tradução de: Borland C++ for Windows.
- [15] OVIATT, Sharon. **Designing Robust Multimodal Systems for Diverse Users and Environments.** In: Workshop on Universal Accessibility of Ubiquitous Computing: Providing for the Elderly, 1, 2001.
- [16] PASTOR, Carmen Alba. Utilização didática de recursos tecnológicos em resposta à diversidade. In: SANCHO, Juana M. **Para uma tecnologia educacional.** Porto Alegre, Artmed, 1998.

- [17] SHUFORD, S. Richard. **Serial Port**. Disponível na Internet. [http://www.cs.utk.edu/~shuford/terminal/blum\\_serial\\_port.txt](http://www.cs.utk.edu/~shuford/terminal/blum_serial_port.txt) 10 julho 2001.
- [18] STRANGIO, Christopher E. **The RS232 Standard**. Disponível na Internet. [http://www.camiresearch.com/Data\\_Com\\_Basics/RS232\\_standard.html](http://www.camiresearch.com/Data_Com_Basics/RS232_standard.html). 10 julho 2001.
- [19] UNIVERSIDADE DE SÃO PAULO. Faculdade de Saúde Pública. **Acessibilidade**. Disponível na Internet. <http://www.fsp.usp.br/acessibilidade/acessibilidade1.htm>. 28 junho 2001.
- [20] UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL. Curso de Informática Educativa Especial. **Simulador de Teclado versão 1.0 Manual do Usuário**. Porto Alegre, 69p.
- [21] VANDERHEIDEN, Gregg C. **Thirty-Something (Million): Should They Be Exceptions?** Disponível na Internet. [http://trace.wisc.edu/docs/30\\_some/30\\_some.htm](http://trace.wisc.edu/docs/30_some/30_some.htm). 14 março 2001.
- [22] VANDERHEIDEN, Gregg C, VANDERHEIDEN, Katherine R. **Guidelines for the design of consumer products to increase their accessibility to persons with disabilities or who are aging**. Disponível na Internet. [http://trace.wisc.edu/docs/consumer\\_product\\_guidelines/toc.htm](http://trace.wisc.edu/docs/consumer_product_guidelines/toc.htm). 14 março 2001.
- [23] WORLD HEALTH ORGANIZATION. **ICIDH-2 Final Draft**. Disponível na Internet. <http://www.who.int/icidh/prefinaldec2000.htm>. 07 junho 2001.

## ANEXO

### A INTERFACE DO SIMULADOR DE TECLADO

#### 1. Definição funcional

Simular um teclado alfanumérico padrão QWERTY;

Simular um teclado numérico;

Simular as teclas de navegação para permitir a navegação em textos.

Simular as teclas de funções e de acionamento de mouse;

#### 2. Modelo conceitual da interação

##### DESCRIÇÃO BÁSICA:

Interagir com o programa

    Selecionar a área do simulador de teclado que deseja utilizar

        Utilizar área selecionada

    Finalizar o programa

##### DESCRIÇÃO DETELHADA:

Selecionar a área do simulador de teclado que deseja utilizar

*Selecionar área alfanumérica (ALT)*

*Selecionar área numérica (ALT)*

*Selecionar área de navegação (ALT)*

*Selecionar área de funções e acionamento de mouse (ALT)*

Utilizar área selecionada

*Digitar na área alfanumérica (SEQ)*

*Utilizar recurso de predição de palavras (OPC)*

*Digitar na área numérica (SEQ)*

*Digitar na área de navegação (SEQ)*



*Digitar na área de teclas funções e acionamento de mouse (SEQ)*

*Finalizar o programa*

### **3. Definição dos cenários para as interações**

*Selecionar área alfanumérica*

**Convite da interface:** Na **janela principal** do programa é oferecido o painel de menu contendo o botão de ativação da área alfanumérica e o ícone apontador de foco da varredura.

**Ação do usuário:** Aciona o botão de ativação da área alfanumérica através de seleção direta, utilizando o ícone apontador de foco da varredura e um acionador de seleção binária, ou através de seleção indireta, utilizando dispositivo apontador.

**Resposta da interface:** Exibe na janela principal a barra de ajuda e o painel da área alfanumérica.

*Selecionar área numérica*

**Convite da interface:** Na **janela principal** do programa é oferecido o painel de menu contendo o botão de ativação da área numérica e o ícone apontador de foco da varredura.

**Ação do usuário:** Aciona o botão de ativação da área numérica utilizando o ícone apontador de foco da varredura e acionador de seleção binária, ou dispositivo apontador.

**Resposta da interface:** Exibe na janela principal a barra de ajuda e o painel da área numérica.

*Selecionar área de navegação*

**Convite da interface:** Na **janela principal** do programa é oferecido o painel de menu contendo o botão de ativação da área de navegação e o ícone apontador de foco da varredura.

**Ação do usuário:** Aciona o botão de ativação da área de navegação utilizando o ícone apontador de foco da varredura e acionador de seleção binária, ou dispositivo apontador.

**Resposta da interface:** Exibe na janela principal a barra de ajuda e o painel da área de navegação.

### *Selecionar área de funções e acionamento de mouse*

**Convite da interface:** Na **janela principal** do programa é oferecido o painel de menu contendo o botão de ativação da área de funções e acionamento de mouse e o ícone apontador de foco da varredura.

**Ação do usuário:** Aciona o botão de ativação da área de funções e acionamento de mouse utilizando o ícone apontador de foco da varredura e acionador de seleção binária, ou dispositivo apontador.

**Resposta da interface:** Exibe na janela principal a barra de ajuda e o painel da área de funções e acionamento de mouse.

### *Digitar na área alfanumérica*

**Convite da interface:** Na **janela principal** é oferecido a barra de ajuda e o painel da área alfanumérica contendo o conjunto de botões representativos das teclas alfanuméricas, o ícone apontador de foco da varredura de linhas e o ícone apontador de foco da varredura de colunas.

**Ação do usuário:** Aciona um dos botões do conjunto de botões representativos das teclas alfanuméricas através de seleção direta, utilizando os ícones apontadores de foco da varredura de linha e coluna e um acionador de seleção binária, ou através de seleção indireta, utilizando um dispositivo apontador.

**Resposta da interface:** Atualiza a posição dos ícones de apontamento de foco para o acionamento de um novo botão, caso o recurso de varredura automática esteja ativado, e atualiza as informações de estado na barra de ajuda.

### *Utilizar recurso de predição de palavras*

**Convite da interface:** Na **janela principal** é oferecido a barra de ajuda e o painel da área alfanumérica contendo o conjunto de botões representativos das teclas alfanuméricas, o ícone apontador de foco da varredura de linhas, o ícone apontador de foco da varredura de colunas, a lista de palavras filtradas do dicionário de palavras, o botão de seleção de próxima palavra, o botão de seleção de palavra anterior, o botão de confirmação de seleção e o botão de cancelamento de seleção.

**Ação do usuário:** Aciona o botão Pre contido no conjunto de botões representativos das teclas alfanuméricas para dirigir-se a área de predição de palavras, aciona o botão de

seleção de próxima palavra e o botão de seleção de palavra anterior para obter a palavra desejada, contida na lista de palavras filtradas do dicionário de palavras, confirmando ou cancelando a seleção através do botão de confirmação de seleção e o botão de cancelamento de seleção respectivamente, utilizando os ícones apontadores de foco da varredura de linha e coluna e acionador de seleção binária, ou dispositivo apontador

#### *Digitar na área numérica*

**Convite da interface:** Na **janela principal** é oferecido a barra de ajuda e o painel da área numérica do programa contendo o conjunto de botões representativos das teclas numéricas, o ícone apontador de foco da varredura de linhas e o ícone apontador de foco da varredura de colunas.

**Ação do usuário:** Aciona um dos botões do conjunto de botões representativos das teclas numéricas utilizando os ícones apontadores de foco da varredura de linha e coluna e acionador de seleção binária, ou dispositivo apontador.

**Resposta da interface:** Atualiza a posição dos ícones de apontamento de foco de varredura para o acionamento de um novo botão, caso o recurso de varredura automática esteja ativado, e atualiza as informações de estado na barra de ajuda.

#### *Digitar na área de navegação*

**Convite da interface:** Na **janela principal** é oferecido a barra de ajuda e o painel da área de navegação contendo o conjunto de botões representativos das teclas de navegação, o ícone apontador de foco da varredura de linhas e o ícone apontador de foco da varredura de colunas.

**Ação do usuário:** Aciona um dos botões do conjunto de botões representativos das teclas de navegação utilizando os ícones apontadores de foco da varredura de linha e coluna e acionador de seleção binária, ou dispositivo apontador.

**Resposta da interface:** Atualiza a posição dos ícones de apontamento de foco para o acionamento de um novo botão, caso o recurso de varredura automática esteja ativado, e atualiza as informações de estado na barra de ajuda.

#### *Digitar na área de teclas de função e de acionamento de mouse*

**Convite da interface:** Na **janela principal** é oferecido a barra de ajuda e o painel da área de funções e acionamento de mouse contendo o conjunto de botões representativos das teclas de funções e acionamento de mouse, o ícone apontador de foco da varredura de linhas e o ícone apontador de foco da varredura de colunas.

**Ação do usuário:** Aciona um dos botões do conjunto de botões representativos das teclas de funções e acionamento de mouse utilizando os ícones apontadores de foco da varredura de linha e coluna e acionador de seleção binária, ou dispositivo apontador.

**Resposta da interface:** No caso da seleção de uma botão correspondente a um comando de mouse, a descrição do comando é exibida na barra de ajuda. Atualiza a posição dos ícones de apontamento de foco para o acionamento de um novo botão, caso o recurso de varredura automática esteja ativado, e atualiza as informações de estado na barra de ajuda.

#### *Finalizar o programa*

**Convite da interface:** Na **janela principal** do programa é oferecido o painel de menu contendo o botão de finalização do programa e o ícone apontador de foco da varredura.

**Ação do usuário:** Aciona o botão de finalização do programa utilizando o ícone apontador de foco da varredura e acionador de seleção binária, ou dispositivo apontador.

## **4. Definição preliminar da apresentação**

Unidade de apresentação 1: Janela Principal

### **Componentes:**

painel de menu

painel da área alfanumérica

painel da área numérica

painel da área de navegação

painel da área de funções e mouse

barra de ajuda

Painel de Menu

### **Componentes:**

botão de ativação da área alfanumérica

botão de ativação da área numérica

botão de ativação da área de navegação

botão de ativação da área de funções e acionamento de mouse

botão de finalização do programa

ícone apontador de foco da varredura

Painel da Área Alfanumérica

**Componentes:**

conjunto de botões representativos das teclas alfanuméricas

lista de palavras filtradas do dicionário de palavras

botão de seleção de próxima palavra

botão de seleção de palavra anterior

botão de confirmação de seleção

botão de cancelamento de seleção

ícone apontador de foco da varredura de linhas

ícone apontador de foco da varredura de colunas

Painel da Área Numérica

**Componentes:**

conjunto de botões representativos das teclas numéricas

ícone apontador de foco da varredura de linhas

ícone apontador de foco da varredura de colunas

Painel da Área de Navegação

**Componentes:**

conjunto de botões representativos das teclas de navegação

ícone apontador de foco da varredura de linhas

ícone apontador de foco da varredura de colunas

Painel da Área de Funções e Acionamento de Mouse

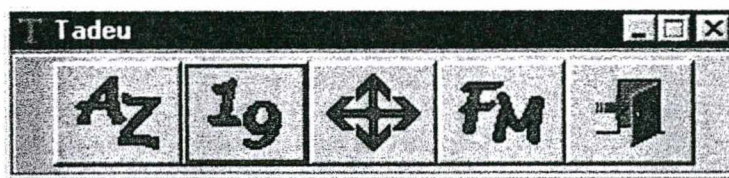
**Componentes:**

conjunto de botões representativos das teclas de funções e acionamento de mouse

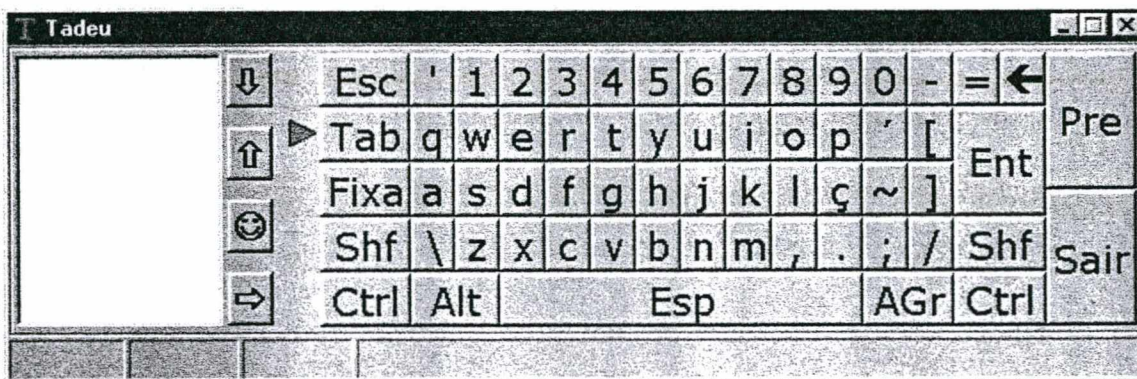
ícone apontador de foco da varredura de linhas

ícone apontador de foco da varredura de colunas

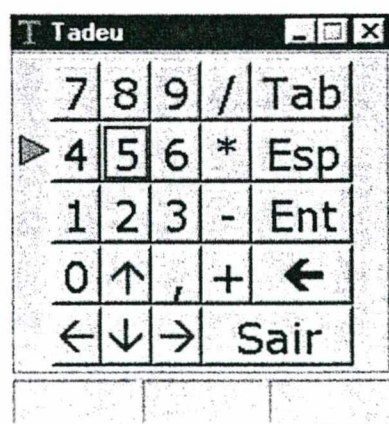
## 5. Definição final da apresentação



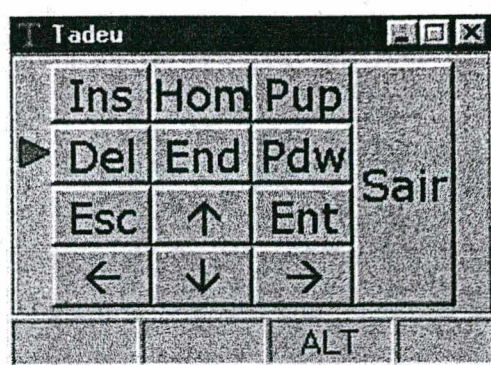
Janela Principal - painel de menu



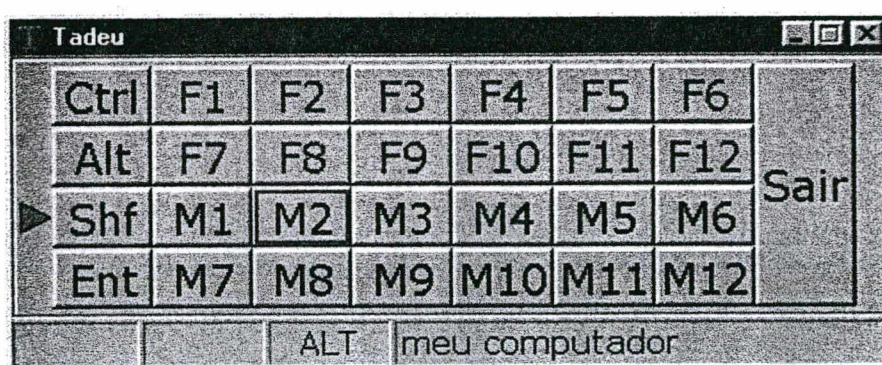
Janela Principal - painel da área alfanumérica



Janela Principal - painel da área numérica



Janela Principal - painel da área de navegação



Janela Principal - painel da área de funções e acionamento de mouse