Claudio Alberto Ikeda

PROJETO E IMPLEMENTAÇÃO DE UM SISTEMA AUTÔNOMO DE COMANDO VOCAL (VC) UNIVERSAL

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de mestre em Engenharia

Área de Concentração

Engenharia Mecatrônica

Orientador

Prof. Dr. Marcos Ribeiro Pereira Barretto

São Paulo
1999

Claudio Alberto Ikeda

PROJETO E IMPLEMENTAÇÃO DE UM SISTEMA AUTÔNOMO DE COMANDO VOCAL (VC) UNIVERSAL

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de mestre em Engenharia

Área de Concentração

Engenharia Mecatrônica

Orientador

Prof. Dr. Marcos Ribeiro Pereira Barretto

São Paulo
1999

Ikeda, Claudio Alberto
Projeto e implementação prática de um sistema autônomo de comando vocal (vc) universal, São Paulo, 1999 111p

Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecânica - Mecatrônica

" O amor é sem sombra de dúvidas o grande agente capaz de transformar simples homens em verdadeiros heróis. O enamorado, perante a sua amada, enchesse de coragem e de valentia sendo então capaz de superar seus próprios medos e lutar como um verdadeiro herói. Dai-me um exército de enamorados e eu conquistarei o mundo..."

Francis Bacon

AGRADECIMENTOS

Ao Prof. Dr. Marcos Ribeiro Pereira Barretto, por sua demonstração de carinho, atenção, compreensão e amizade na orientação deste trabalho.

Ao meu amigo Eng. Jorge Szanto de Toledo, pelo seu companheirismo e amizade, sem os quais não seria possível a execução deste trabalho.

Ao Prof. Dr. Giorgio Eugênio Oscare Giacaglia, pelo seu incentivo na execução desta pós graduação..

À Prof. Dra. Edith Ranzini, por sua solicitude e dedicação na definição de meios práticos para execução da minha tese .

Aos meus pais e a minha querida Silvia, que sempre incentivaram a conclusão deste projeto

A todo o departamento de Mecatrônica, pelo auxílio técnico indispensável .

A todos que cooperaram para a conclusão deste trabalho.

ABSTRACT

The complexity of speech recognition systems results in a great variety of scientific research on its subsystems, including training algorithms, recognition or electronic sound acquisition subsystem. This reality difficults a global vision of a real speech recognition system. Regarding this reality it was decided to develop a real speech recognition system integrating usual speech recognition subsystems. For this propose we selected a pre-programmed DSP chip integrated to a Man Machine Interface developed on PC environment.

The prototype was submited to different operational conditions :

- three users
- seven sets of words
- three types of environment noise

and all commands were trained and recognized on portuguese language

DSP Chip nominal recognition rate and response time were also measured..

Resumo

A complexidade dos sistemas de reconhecimento de comandos vocais tem resultado em intensa pesquisa em temas específicos tais quais algoritmos de treinamento, algoritmos de reconhecimento, técnicas de aquisição de sinais sonoros , etc. Esta realidade segmentada dificulta ao pesquisador uma visão global da implementação de um sistema real de reconhecimento de comandos vocais. Deste modo, optou-se pelo desenvolvimento de um sistema, baseado em subsistemas de tecnologias usualmente empregadas em sistemas desta classe . Para este propósito selecionou-se um chip DSP pré-programado, integrado a uma interface Homem Máquina desenvolvida em ambiente PC.

O protótipo implementado foi submetido a diferentes condições de operação:

- Três usuários
- Sete conjuntos de palavras
- Três tipos de ruídos ambientais

E todos os comandos foram treinados e reconhecidos pelo protótipo, na Lingua Portuguesa. Os valores nominais de taxa de reconhecimento e tempo de resposta foram também medidos.

345

1 Introdução

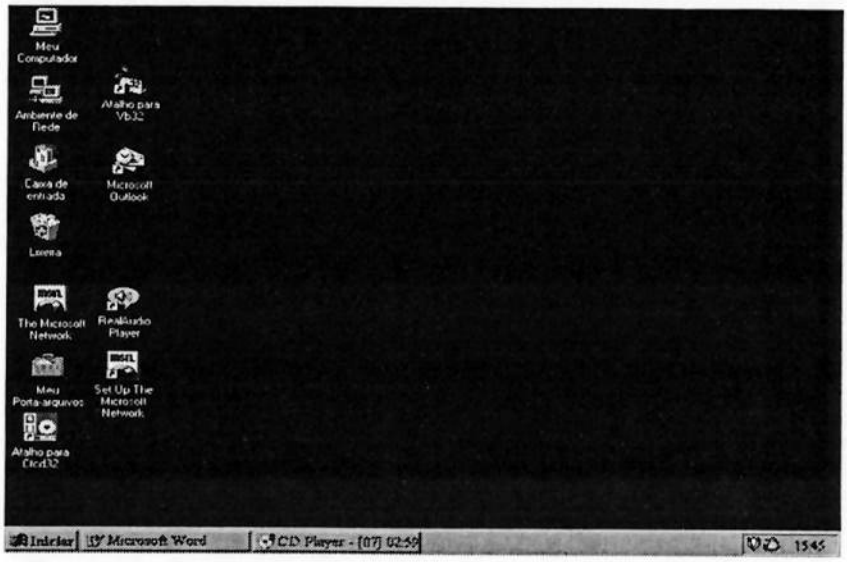


Fig.1 A Interface Homem Máquina I

O homem, desde o início de seus tempos, vem buscando a descoberta e a melhoria de ferramentas e instrumentos que melhorem a sua qualidade de vida. Começamos pela moradia em cavernas, passamos pela descoberta da pedra lascada, do ferro, do bronze e passamos hoje para a disponibilidade de diferentes facilidades da vida moderna tais quais o automóvel, os escritórios inteligentes, etc. Enfim, em todos os ramos da ciência e da tecnologia o homem avançou em passos largos rumo à comodidade, funcionalidade; inventando e aprimorando máquinas e tecnologias que alargassem os limites da capacidade humana. [17, 18, 19, 20, 21, 22, 23, 24, 25, 26]

No ramo dos sistemas mecânico-eletrônicos, ou "mecatrônicos", a história não foi diferente. Partimos da descoberta da energia elétrica, passamos pelos sistemas comandados por válvulas, pelos transistores e chegando hoje aos sistemas microcontrolados programáveis, extremamente compactos e flexíveis, capazes de comandar desde um simples brinquedo até um moderno avião de combate ou então uma usina nuclear.

Frente a todo este avanço tecnológico, notamos uma área de destaque, a tecnologia das IHM (Interface Homem-Máquina) aplicadas no comando de todos os sistemas

'mecatrônicos" em que haja necessidade de interação com o comando ou julgamento humano. Sejam em aparelhos civis ou militares, a aplicação de IHMs adequadas é imprescindível para o correto aproveitamento de seus recursos . O ser humano, após uma estranha tendência de implementar IHMs com um número infindável de botões buscando uma falsa impressão de poderio tecnológico, tem buscado o caminho da funcionalidade, simplificando ao máximo a interface entre máquina e homem, para que assim seja possível uma maior familiaridade do usuário com o sistema. Um bom exemplo desta tendência é o novo padrão de IHM para microcomputadores chamado "Windows". [18,19]

Hoje, o uso da comunicação por meio de ícones intuitivos, janelas ("Windows") e "mouse" é sinônimo de sistema funcional e de interface amigável. Não importa o programa, seja ele um editor de texto, são todos similares e amigáveis de modo que caso o usuário esteja familiarizado com o padrão Windows, qualquer programa que siga este padrão terá o seu tempo de familiarização com o usuário extremamente reduzido.

É inegável que o uso de interfaces universais e intuitivas, que sirvam para diferentes tipos de sistemas, deverá ser também uma tendência nas interfaces dos sistemas "mecatrônicos". Deverá surgir um padrão, intuitivo e flexível, que

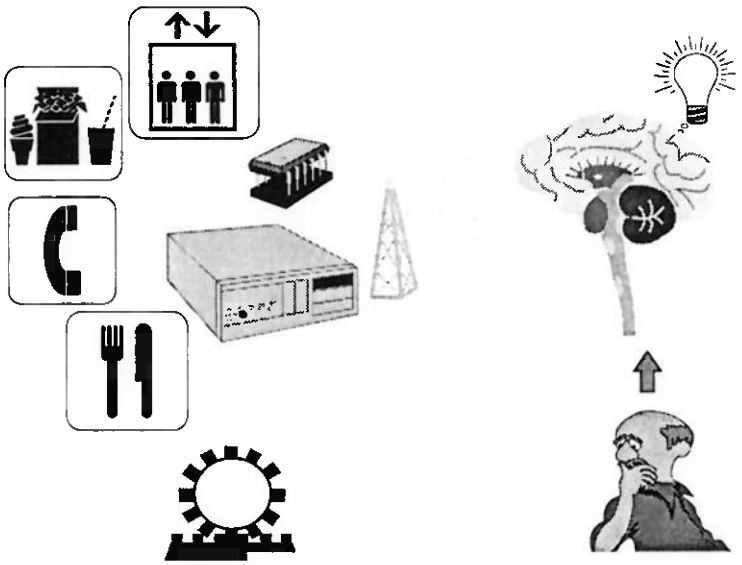


Fig. 2. A Interface Homem Máquina III

permita que diferentes equipamentos e sistemas sejam operados por interfaces padronizadas. Mas qual será esta nova interface, capaz de rivalizar com o sucesso do padrão "Windows"?

Rapidamente poderíamos dizer: "Com certeza seria uma interface que lesse as vontades humanas, fizesse a devida filtragem de eventuais distorções do pensamento e que comandasse diferentes equipamentos ou sistemas, de tal forma que a vontade ou necessidade do operador fosse satisfeita.". Neste ponto haveríamos de concordar que este sistema comandado por ondas cerebrais está muito longe de nossa realidade, visto que para a viabilização deste projeto precisaríamos desenvolver um sistema capaz de 'ler' o pensamento humano. Na verdade seria uma interface direta entre o cérebro humano e o comando do equipamento. Apesar de ideal, esta interface ainda deverá esperar algumas décadas para se tornar realidade. Uma interface com grande potencial, e realmente implementável com a tecnologia hoje disponível é o comando vocal. [25]

O uso do comando vocal nos IHM (Interface Homem Máquina) parece ser hoje um caminho irreversível,

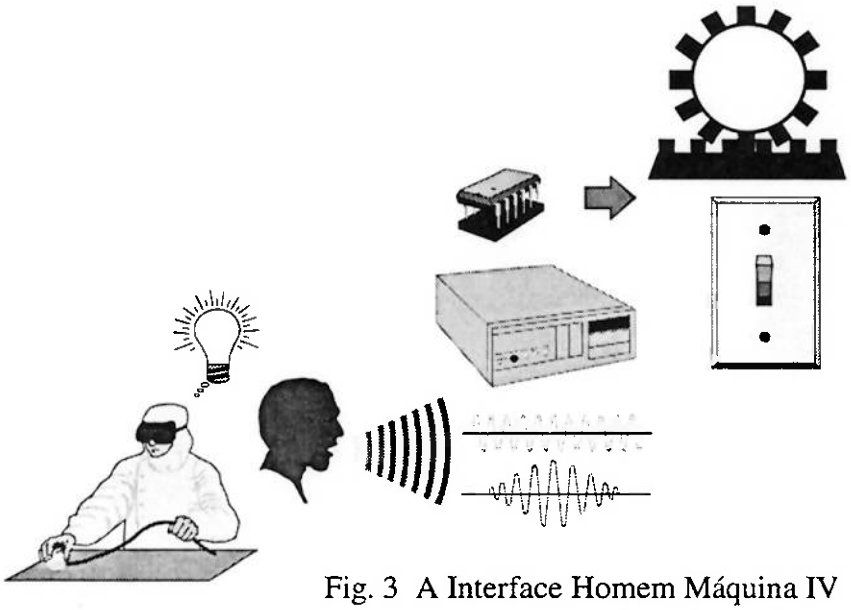


Fig. 3 A Interface Homem Máquina IV

representando assim um campo potencial para o estudo e a pesquisa de novas aplicações para a tecnologia de reconhecimento da fala humana.

Tanto em aplicações, designadas "low- end" como por exemplo o recurso de comando em um eletrodoméstico, com uma dezena de comandos, ou em aplicações designadas "high-end" como por exemplo um avião de combate, com milhares de instruções, painéis de leitura, táticas de combate, comando de armas; podemos notar claramente uma tendência dos projetos de IHM no sentido de aproximar a funcionalidade de sistemas informatizados em níveis cada vez mais próximos à própria interface humana. [25].

Um bom exemplo disso é o projeto TRON. Este projeto, desenvolvido na Universidade de Tóquio e financiado por um consórcio formado pelas empresas Mazda, Renault e Pegeout, tem como meta o desenvolvimento de um sistema integrado, formado por subsistemas que simulam os sentidos visual, auditivo e a presença de um copiloto, identificando situações de perigo, colisão, auxílio de navegação em neblina e chuva. Além disso, o sistema também é capaz de avisar ao motorista sobre o status de variáveis de controle do carro tais como autonomia, temperatura, pressão dos pneus, etc, além de atualizar algumas destas variáveis através de comando vocal tais como ligar ou desligar os faróis, abrir ou fechar os vidros, discar automaticamente o telefone celular e interrogar um auxílio de navegação ao sistema GPS do carro. [12, 13, 14, 15] O sistema de comunicação com o motorista é baseada na comunicação verbal tanto de interpretação quanto de síntese.

Inspirado neste tipo de aplicação da tecnologia de reconhecimento de comandos vocais desenvolveu-se esta dissertação. A meta deste projeto é, portanto, incorporar a tecnologia de reconhecimento de comandos vocais em Interfaces Homem Máquina.

Tendo em vista este potencial da "speech technology" vamos estudar a teoria e a aplicação desta tecnologia no sentido de criar um novo tipo de IHM ou seja uma "INTERFACE HOMEM MÁQUINA REPRESENTADA PELO COMANDO VOCAL (IHM-VC)".

OBJETIVOS DO TRABALHO

Após uma análise de "papers" de diversos pesquisadores especialistas na tecnologia de reconhecimento de voz, concluiu-se que o reconhecimento de voz é a integração das tecnologias de:

- tratamento digital do sinal de voz [1, 2, 3, 4, 6, 7, 8]
- modelagem e parametrização da voz [6, 7, 8]
- algoritmos de treinamento (cruzamento dos parâmetros obtidos com os padrões pré-estabelecidos) [7,8]
- modelos de reconhecimento [7,8]

A exploração científica aprofundada de cada um destes temas está fora do escopo deste trabalho. Neste trabalho, apresentou-se uma visão geral sobre o "estado da arte" de cada um destes tópicos específicos. Vale lembrar que não individualizamos nenhum dos tópicos específicos, visto que a intenção principal deste trabalho é a integração destes diversos subsistemas, de tal forma que se possa apresentar uma solução global integrada e implementável para se ter uma real perspectiva da funcionalidade deste tipo de sistema.

Tendo por base esta meta, ao invés de se buscar o desenvolvimento ótimo de todos os subsistemas, optou-se pela implementação de um sistema integrado. Deste modo, foi

implementado um sistema de reconhecimento de comandos vocais baseado em uma plataforma flexível (programável) e de alto desempenho.

As características básicas de projeto do sistema automático de reconhecimento de comando vocal devem ser:

- Baixa Complexidade

  O sistema deve ser ter uma operação intuitiva e amigável, de modo que o usuário não necessite de treinamento ou de conhecimento específico sobre o sistema

- Baixo Tempo de resposta

  O sistema deve possuir baixo tempo de resposta de modo que a de desempenho da aplicação queda de performance do sistema seja imperceptível ao usuário.

- Alta Taxa de Reconhecimento

  O sistema de reconhecimento de voz deve ter alta taxa de acerto de comandos vocais superior, para que o usuário tenha a percepção de uma interface confiável

- Robustez à Variação do Nível de Qualidade do Som

  O sistema será utilizado sob diversas situações de ruído ambiente de modo que, o sistema seja resistente a eles

- Sistema dependente do usuário

  O sistema deverá responder ao usuário responsável pela inserção dos comandos vocais na fase de treinamento

2 Revisão da Literatura

Neste capítulo aborda-se, sob o enfoque teórico, os diversos subsistemas do reconhecedor de comandos vocais destacando-se:

- Tratamento digital do sinal de voz
- Modelagem e extração de parâmetros
- Algoritmos de treinamento
- Modelos de reconhecimento

2.1 Tratamento digital do sinal de voz

Neste item apresenta-se o subsistema de aquisição, digitalização e filtragem do comando vocal, conforme descrito na figura abaixo:

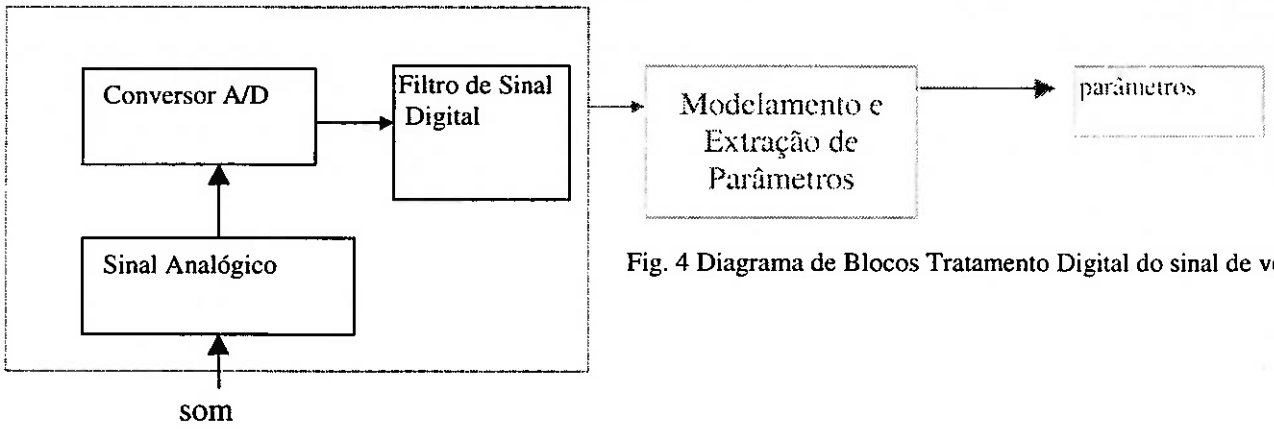
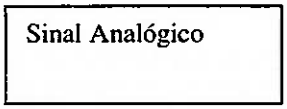


Fig. 4 Diagrama de Blocos Tratamento Digital do sinal de voz

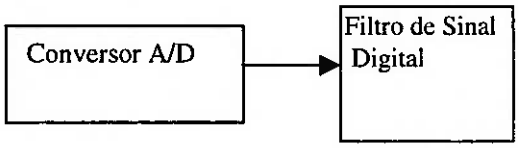
O sinal sonoro é captado pelo reconhecedor de comandos vocais e transformado em sinal analógico. O sinal analógico é convertido em sinal digital no conversor A/D e então tratado em um filtro de sinal digital. Técnicas de modelagem, análise , predição e algoritmos de reconhecimento e treinamento de sinais vocais serão abordados a seguir.

2.1.1 Aquisição



Aquisição é o subsistema responsável pela captação do sinal vocal físico no formato de ondas sonoras e convertido em tempo real em impulsos elétricos. Este subsistema é conhecido como sistema piezelétrico e operacionalizado no sistema pelo componente microfone.

2.1.2 Digitalização e Filtragem do Sinal Acústico



O sinal analógico adquirido precisa ser convertido em um sinal digital, para que assim possa ser utilizado no processo de treinamento e reconhecimento de comandos vocais [7]. Para este fim é necessária a inclusão de um subsistema de conversão A/D (Analógico/Digital) e de um filtro digital . Estes subsistemas integrados formatam o comando vocal para o processamento da etapa de modelagem e parametrização que está descrita a seguir.

2.2 Modelagem e extração de parâmetros

A modelagem é uma das formas usadas para simplificar o processo de análise ou síntese de sinais analógicos de forma a simplificar o processo representado por operações diretas sobre a própria forma de onda. [1,2,3]

Se um modelo de uma fonte de sinais pode ser implantado, apenas os parâmetros do modelo precisam ser processados, transmitidos ou armazenados.

O modelamento no domínio do tempo tenta adequar uma amostra finita do sinal para um conjunto de coeficientes que podem predizer o comportamento futuro a curto prazo, tal como em uma interpolação.

O modelamento, no domínio da frequência, ajusta parâmetros tais como: a frequência, a amplitude e os picos dominantes do espectro, permitindo a regeneração do sinal.

Os processos conjugados de modelagem e predição estão relacionados com os algoritmos de compressão de dados, que procuram reduzir o número de bits necessários para a transmissão, processamento e armazenamento da informação digital.

Para a modelagem e a extração de parâmetros de comandos vocais, a demanda de largura de banda é relativamente baixa (cerca de 2.400 bits/segundo) e o método usualmente empregado para esta função é conhecida como LPC (Linear Predictive Coding). [5,22, 28, 31]

2.2.1 Predição Linear (LPC)

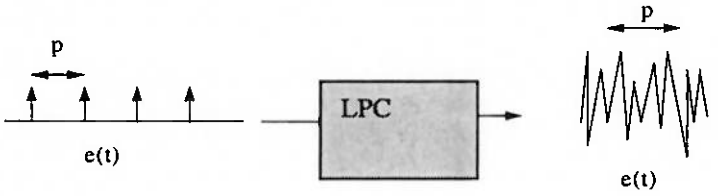


Figura 5 Predição Linear

O sistema de codificação LPC prevê o valor do sinal de entrada baseado nos valores dos pesos das n amostras anteriores, onde n é o número de coeficientes de filtragem (a ordem do filtro de síntese). A diferença entre a entrada atual e o valor previsto é definido como " erro de predição linear (E)". O problema da predição linear é determinar um conjunto de n coeficientes que minimiza o erro quadrático médio de predição (E) ao longo de um curto segmento (janela) do sinal de entrada. [1, 5, 6]

2.3 Algoritmo de Treinamento

Na etapa conhecida como treinamento, os parâmetros LPCs, obtidos na fase de modelamento e extração de parâmetros, são transformados em vetores que posteriormente serão utilizados na fase de reconhecimento do comando vocal (quantização vetorial). Trata-se, portanto, das referências a serem utilizadas na fase de reconhecimento.

2.3.1 Quantização Vetorial

A quantização pode ser definida como uma operação de característica não linear onde uma variável contínua pode apenas assumir valores discretos.[1],[5],[6]

Na prática, é exatamente a etapa subsequente à fase de filtragem, janelamento e modelamento do sinal de voz, na qual os comandos vocais já modelados são submetidos a um processo de compressão no sentido de reduzir a quantidade de bytes necessários para a transmissão e armazenamento dos dados, sendo sua qualidade avaliada por critérios de fidelidade ou de distorção.[5]

A quantização vetorial possui as etapas de Treinamento ou Aglutinação ("clustering") e de Quantização propriamente dita.[6]

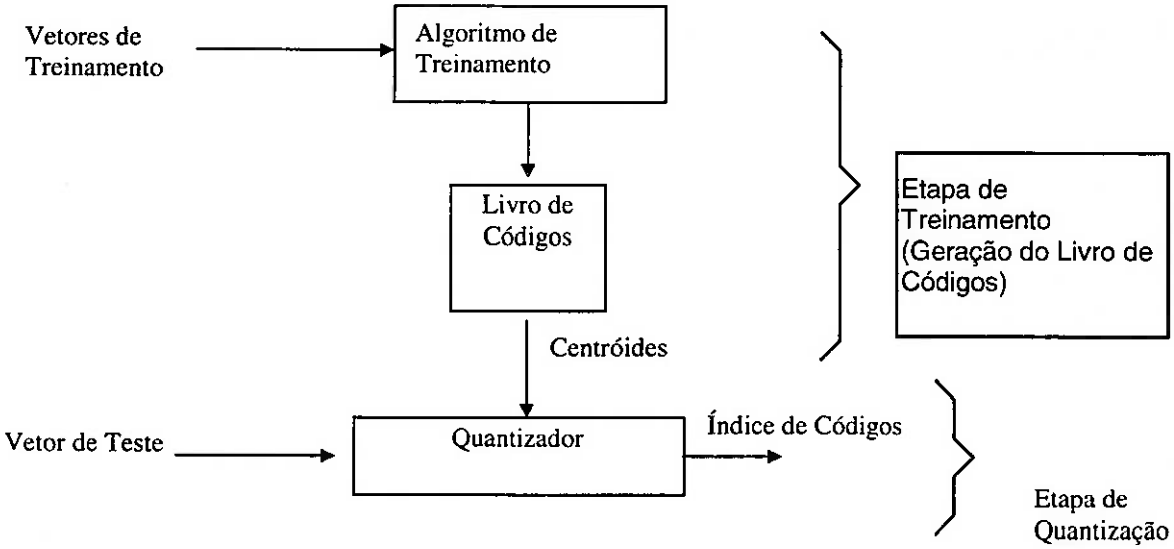


Fig. 6 Quantização Vetorial

Na etapa de treinamento, executa-se um algoritmo para o dimensionamento (projeto) de um alfabeto ou livro de códigos("codebook"); cada vetor deste livro de códigos é denominado centróide, vetor código, palavra código ou vetor protótipo.[5]

Na etapa de quantização, mapea-se o nível de similaridade entre o vetor de teste e os centróides do livro de códigos. Este mapeamento segue a regra de mínima distorção (NN - nearest neighbor) , através de medida de distorção.[5]

Define-se como vetor de teste de dimensão-M:

$x = [x_1,x_2,...,x_M]$ $(x \in R^M)$
e o livro de códigos S com extensão K, com cada vetor código (centróide):

$y_i = [y_{1i},y_{2i},y_{3i},...,y_{Mi}]$ $(y_i \in R^M,\ 1 \le i \le K)$

Pode-se representar a distorção entre o vetor de teste e o vetor código como:

$d(x,y_i)$

uma distorção $d(x,y_i)$ será mínima, para o vetor de teste x se e somente se

$d(x,y_i) \le d(x,y_j);\ j \ne i,\ 1 \le j \le k$

e assim $y=y_i$ será a palavra código de x_i . Pode-se então escrever:

$y=y_i=q(x)$

se e somente se

$d(x,y_i) \le d(x,y_j);\ j \ne i\ \ 1 \le j \le k$

onde q( ) representa a operação de quantização

Para uma melhor visualização do significado físico do processo de quantização vetorial imagine o processo no caso bidimensional (M=2)

Seja x=(x1,x2) o vetor de teste e seja yi de K níveis ($1 \leq i \leq K$) o livro de códigos, como mostra a figura:

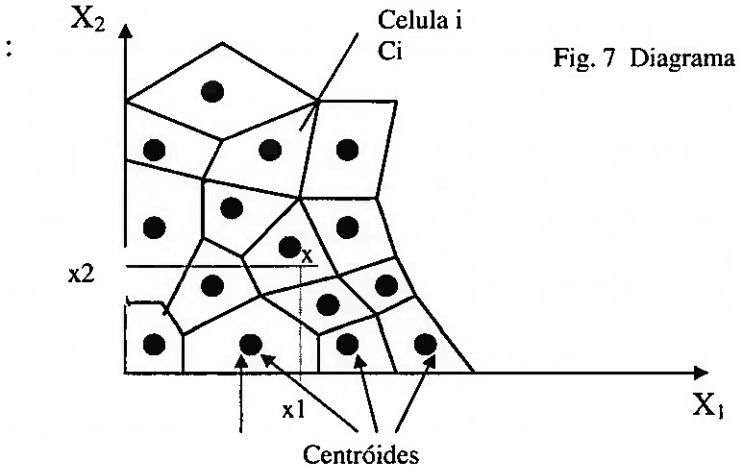


Fig. 7 Diagrama de Centróides

Nesta figura, os centróides y_i são representados por círculos escuros, as fronteiras de distorção são representadas por linhas que delimitam os círculos e o vetor de teste é representado por um "x". O conjunto formado pelo centróide e suas linhas limítrofes são designadas como regiões de Voronoi. [5]

A técnica de quantização vetorial é extremamente útil no processo de compressão e equalização das sequências de parâmetros obtidos através da aplicação do modelamento LPC a série de amostras (digitais) do sinal contínuo da voz (analógico)

2.3.2 Treinamento: Geração de Livro Código

O processo de quantização vetorial poderia ser descrito como um mapeamento num espaço M-dimensional de um vetor aleatório x para uma célula C_i de um livro de códigos (ou alfabeto) A de K níveis ($C_i \in A$, $1 \leq i \leq K$) [5],[6]. O quantizador fornece o vetor código y_i se x está contido em C_i.

Na geração do livro de códigos A de um quantizador vetorial, deve-se procurar obter uma distorção mínima, o que equivale a dizer que se deve procurar um quantizador ótimo. Numa implementação prática, utiliza-se um conjunto finito de vetores .

Algoritmo LBG

Um método para a geração do livro de códigos é um algoritmo interativo de aglutinação . Este algoritmo foi originalmente apresentado por Lloyd [5], sendo que a generalização para várias medidas de distorção e aplicação em quantização vetorial foi feita por Linde, Buzo e Gray [5], sendo denominado algoritmo de Lloyd generalizado ou algoritmo LBG. O algoritmo LBG pode ser dividido em quatro etapas:

- Inicialização
- Classificação
- Atualização
- Teste de Parada.

Na inicialização, escolhem-se os vetores iniciais arbitrários como centróides. Na classificação, todos os vetores de treinamento são testados e mapeados nas células definidas pelos centróides atuais. Na atualização, após o término da classificação, cada célula terá um subconjunto de vetores de treinamento e recalculam-se todos os centróides, levando-se em consideração estes vetores de treinamento. No teste de parada, verifica-se a distorção total e , sendo esta aceitável, o procedimento termina; caso contrário, retorna-se à etapa de classificação e assim iterativamente.

Como os vetores de treinamento formam um conjunto amostral estocástico e na etapa de inicialização utiliza-se um critério arbitrário, pode-se inferir que o algoritmo fornecerá um resultado sub-ótimo, sendo a distorção total localmente mínima. Linde faz sugestões de como se obter um livro de códigos ótimo. [5]

2.4 Modelos de Reconhecimento

Após a etapa de Aquisição e Pré-Processamento Acústico, incluindo etapas de filtragem , codificação e compressão dos frames digitalizados (do sinal analógico da voz), apresenta-se a etapa subseqüente que é exatamente, executar a correlação entre o sinal de voz amostrado e um conjunto de palavras ou fonemas pré - determinados na etapa de treinamento do sistema..[14,15, 16, 20, 25, 28, 33]

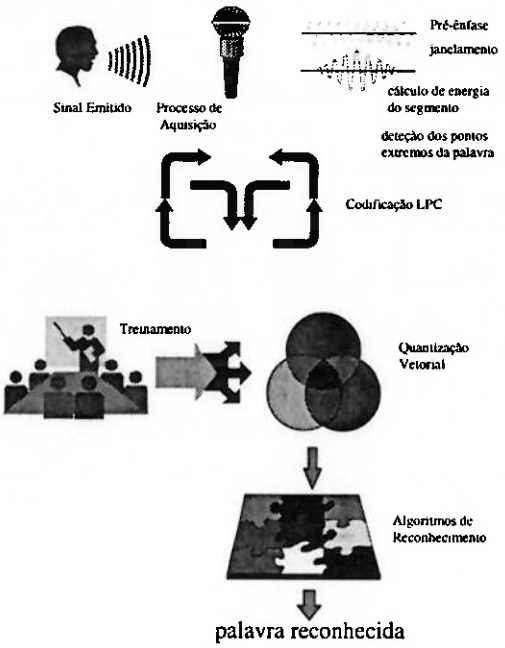


Fig. 8 Sistema de Reconhecimento de Voz

Estudou-se de modo abrangente os diferentes modelos de reconhecimento, destacando-se os reconhecedores baseados nos modelos HMM (Hidden Markov Models) e ANN (Artificial Neural Network)

2.4.1 Unidades de Reconhecimento

Uma classificação básica feita entre as diversas variantes dos reconhecedores de voz é baseada na unidade mínima de reconhecimento adotada por cada um destes modelos.

Um reconhecedor de palavras, como o próprio nome indica, adota como unidade mínima de reconhecimento a palavra. Computacionalmente representa uma forma com maior requisito de processamento e armazenagem. O seu vocabulário, via de regra, restringe-se a um conjunto limitado de palavras (ordem de centenas de palavras). [11],[22],[39],[40],[41],[43]

Um reconhecedor de palavras, baseado em fonemas, é uma evolução dos sistemas baseados em reconhecimento de palavras. Exatamente por não ser vinculado a um conjunto finito de

palavras e sim a uma combinação quase infinita de fonemas, o sistema é capaz de reconhecer milhares de palavras. Basicamente, este tipo de sistema trabalha com duas formas de reconhecimento. A primeira é o reconhecimento de suas unidades ou seja, os fonemas, e depois a combinação dos fonemas de modo a formar as palavras, num processo denominado Construção por Níveis ("Level Building").

2.4.2 Reconhecedores de Comandos Contínuos ou Individualizados

Além da classificação por unidades mínimas de reconhecimento, pode-se classificar os reconhecedores de voz em reconhecedores de palavras isoladas ou seja, palavras separadas por intervalos de silêncio, ou como reconhecedores de discursos contínuos em que se buscam palavras conhecidas no meio da sequência de fonemas . [42],[9],[37],[40],[22],[38]

2.4.3 Algoritmos de Reconhecimento

A perspectiva atual indica que existe a possibilidade de utilização da tecnologia de reconhecimento de voz em uma quantidade crescente de aplicações. Os algoritmos baseados no modelo HMM (Hidden Markov Models) têm sido utilizados em sistemas de reconhecimento com desempenho impressionante.

Vale lembrar que, os reconhecedores de voz não representam de forma alguma uma tecnologia dominada. Enquanto que a tecnologia hoje existente serve para um número finito de aplicações comerciais específicas, as pessoas esperam que os sistemas de reconhecimento de voz se comportem de modo similar ao sistema humano, pessoas podem reconhecer palavras contendo acentuação incorreta, ruído ambiental, problemas de acústica

(ecos), palavras gramaticalmente incorretas, e palavras não familiares. Apesar das adversidade, somos capazes de identificar o que está sendo dito. O desempenho do reconhecimento humano , em situações realísticas, é ainda muito melhor do que o dos reconhecedores automáticos. Para tarefas simples, sob condições ambientais controladas, os sistemas de reconhecimento de comando vocal podem ter desempenho aceitável.

Muita evolução e pesquisa serão necessários aos sistemas de reconhecimento de comando vocal . É necessário considerar soluções alternativas, tais como os modelos baseados em redes neurais. Ressalta-se entretanto, que dada a forte base matemática para reconhecedores de voz estatísticos, é muito mais inteligente modificar apenas alguns aspectos das aproximações existentes do que começar do "zero".

Na tabela abaixo mostra-se os prós e os contras do paradigma dominante de reconhecimento de voz, nos quais os HMM (Hidden Markov Models) são usados

HMM Clássico em Reconhecedores de voz	
Prós	**Contras**
rica base de ferramentas matemáticas	baixa discriminação
Poderosos métodos de treinamento e decodificação	Pré-condições básicas para hipóteses distributivas
Boa abstração para sequências, aspectos temporais	Modelos de Markov de primeira ordem para os estados de fonema e subfonema
Flexibilidade de topologia para fonologia e sintaxe estatística	usualmente ignora a correlação entre vetores acústicos.

Tabela 2 Prós e Contras dos HMM Clássico em ASRs

Os HMM (Hidden Markov Models) também apresentam como característica básica a ausência de necessidade de segmentação explícita (manual) em termos de unidades de fala (tipicamente fonemas) usado como base para reconhecimento e treinamento de

reconhecedores contínuos de fala. Além disso, eles também assimilam diferentes níveis de relações (fonológica e semântica), uma vez que estas podem ser representadas pelo mesmo formalismo estatístico.

Vale entretanto lembrar que, para que se possa tirar proveito das vantagens da representação, os algoritmos, implicitamente ou explicitamente, assumem hipóteses que são em alguns casos irreais. Por exemplo, é sempre necessário assumir que vetores retirados de segmentos fonéticos precisam sempre ser correlacionados com um outro vetor.

Apesar do HMM (Hidden Markov Models) não apresentar uma boa correlação com as condições usuais destes sistemas, eles são muito populares, visto que sua forte base de ferramentas matemáticas facilita a exploração de suas potencialidades para este tipo de aplicação.

O ANN (Artificial Neural network) apresenta-se como um método de reduzir a dependência dos sistemas a hipóteses irreais sobre a fala e seu ambiente. Apesar disso, não existe uma unanimidade sobre a melhor técnica, visto que observa-se o surgimento de diversos sistemas com desempenhos similares, baseados em cada uma das técnicas, ou então, utilizando-se de ambas as técnicas em diferentes funções. Desta observação também nota-se uma significativa melhora de desempenho pelo uso do ANN em grandes modelos estatísticos. A maioria dos grandes laboratórios de pesquisa nesta área, tal como o IMT (Laboratório Lincoln), os Laboratórios Bell, têm publicado artigos de uma forte linha de trabalho em que o ANN é usado para gerar as probabilidades necessárias para os modelos HMM. [46,54]

Artificial Neural Networks (ANN)

Multilayer Perceptrons (MLPs)

A discussão sobre redes neurais para análise de fala será abordada sobre os "Multilayer Perceptron (MLP)" que é a ANN mais usada nos reconhecedores de voz. Vale lembrar que todas as conclusões básicas sobre a utilidade destas estruturas para a estimativa de probabilidades HMM também faz uso de outros tipos de ANN, tais como rede neural recursiva, ou uma Rede neural com Defasagem Temporal (TDNN). O MLP tem uma estrutura de camadas subdividindo-se em:

Camadas de entrada, intermediárias (Hidden layers) e saída. Cada camada computa um conjunto de funções lineares de discriminação (via matriz de pesos) seguida de uma função não linear, descrita por uma função não linear conhecida como função sigmóide.

Os MLPs com suficientes unidades intermediárias (hidden), podem, em princípio, prover mapeamentos arbitrários g(x) entre a entrada e a saída. Parâmetros MLP (os elementos da matriz de pesos) são treinados para associar um vetor de saída "esperada" com um vetor de entrada. Isto é conseguido através de um algoritmo de minimização da propagação de Erro a posteriori (Error Back Propagation (EBP)) para redes neurais de múltiplos níveis intermediários, como também para redes neurais com apenas um nível intermediário que usa um procedimento para minimizar, interativamente a estimativa de probabilidades nos HMM. [46,54].

ANN no Papel de Estimadores Estatísticos

O ANN pode ser usado para classificar unidades de fala, tais como fonemas ou palavras, mapeando representações temporais em espaciais, ou então usando recorrências. Esta era a forma na qual os ANNs eram inicialmente utilizados nos problemas de reconhecimento de palavras primárias. Todavia, os ANNs usados como classificadores de sequências temporais completas não apresentam bom desempenho no processo de reconhecimento contínuo de fala. Isto se deve basicamente pelo fato de que o número de sequências possíveis de fonemas é infinito. Além disso, não existe nenhum meio conhecido de traduzir uma sequência de vetores acústicos em uma sequência de unidades de fala usando apenas um ANN. Por outro lado, os HMMs provêem de uma estrutura razoável para representar sequências de sons de fala ou palavras. Assumindo este fato, podería-se dizer que uma boa função para os ANN é estimar a distância de um determinado vetor dos diversos centróides do universo de treinamento.

Para sistemas de reconhecimentos estatísticos, a regra para estimadores locais é a aproximação pelo princípio de probabilidades. Em particular, dadas às equações básicas dos HMMs, pode-se querer estimar algo como a probabilidade $p(x_n!q_k)$ que representa a probabilidade de um vetor de dados ser observado, dados os estados HMM hipotetizados (correspondente a uma determinada palavra ou fonema).

Todavia, os HMMs são baseados em um formalismo muito restrito que é muito difícil de se modificar sem a perda das bases teóricas ou então a eficiência dos algoritmos de treinamento e reconhecimento.. Felizmente, os ANNs podem estimar estas probabilidades de emissão , e que portanto, podem ser perfeitamente integrado a um sistema baseado em

HMM. Em particular, os ANNs podem ser treinados para produzir probabilidade a posteriori $p(q_k!x_n)$ de um determinado estado HMM dado um vetor acústico. Se cada saída do ANN puder ser associado a um estado HMM específico então esta propriedade de predição pode ser convertida para probabilidades a priori (emissão) usando a regra de Bayes.

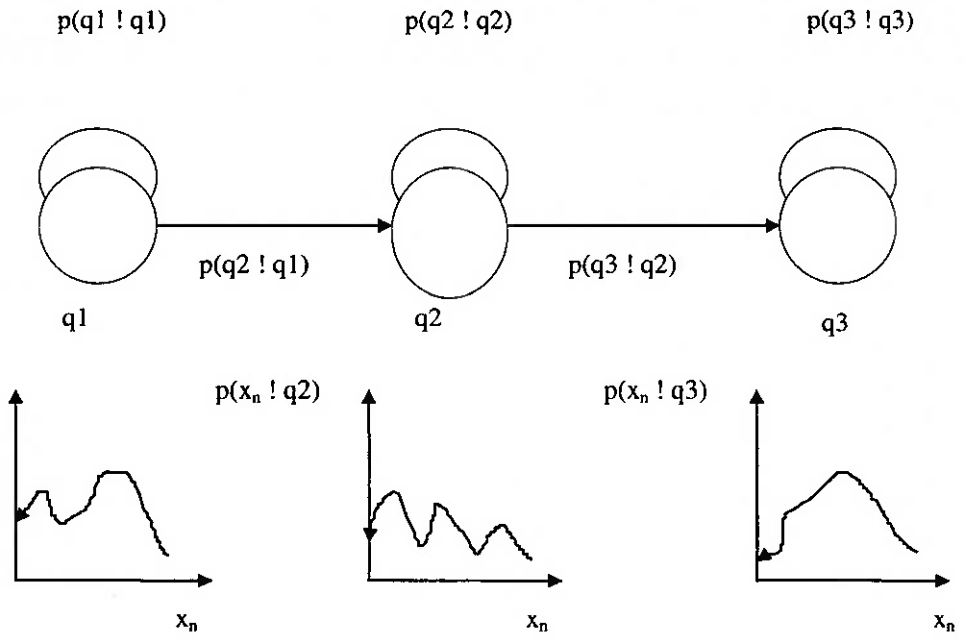


Fig 9 Diagrama ANN

Existe a tese de que as saídas ANN usadas no modo classificação pode ser interpretada como uma estimativa de probabilidades a posteriori de classes de saída condicionadas às entradas. Pode-se representar graficamente o princípio:

A figura mostra que um equilíbrio pode ser atingido no caso ideal, que na verdade corresponde a uma saída da rede g que é igual a probabilidade a posteriori p. Considerando uma área no espaço de formatos ao redor de um padrão de treinamento x_n, e assumindo que só se pode considerar duas classes:

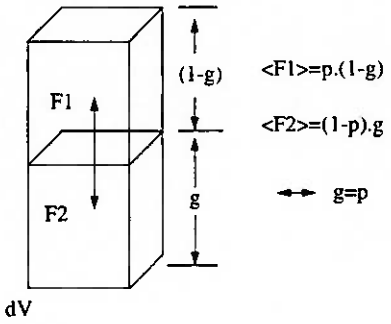


Fig. 10 Divisão de forças

- a classe-objeto para este padrão
- a classe-objeto de todos os formatos que não pertencem a esta classe. Os vetores pertencentes a área selecionada irão incitar uma força ascendente correspondente ao gradiente do termo de erro para todos os "features" com um target de 1; o termo de erro quadrático neste caso é ½ $*(1-g)^2$, com derivada igual a (1-g), e neste caso isto assumirá valor de probabilidade p por uma fração de segundo. A força ascendente média aplicada à região é de p(1-g). A força descendente é definida de forma similar como (1-p)g, força esta que só se balanceia no modo equilibrado.

As condições básicas para que um ANN seja apropriado para a função de estimadores de probabilidades de Bayes são:

1. O sistema deve conter parâmetros suficientes para ser treinado Isto garantirá uma boa aproximação da função de mapeamento entre as classes de entrada e de saída;

2. O sistema deve ser treinado para um padrão onde o erro quadrático médio e a entropia relativa servirão como critérios de erro.

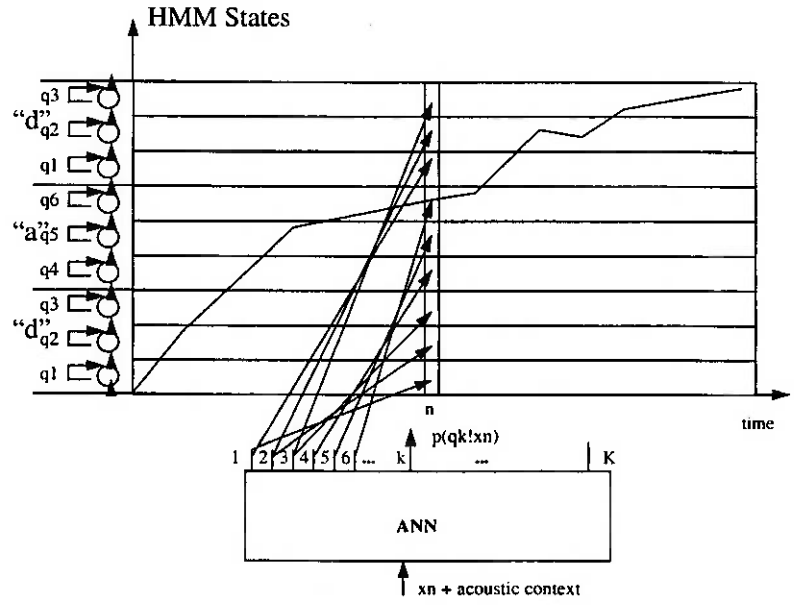
Observa-se que, experimentalmente, os sistemas treinados com uma grande quantidade de amostras de fala fornece boas aproximações de probabilidades a posteriori .

As probabilidades a priori dos HMMs podem ser obtidos das saídas do ANN (probabilidades a posteriori) aplicando-se para isso a regra de Bayes. Comumente usa-se:

$$\frac{p(x_n!q_k)}{p(x_n)} = \frac{p(q_k!x_n)}{p(q_k)}$$

Isto significa que pode-se identificar as estimativas a posteriori baseadas nas saídas do ANN das classes a priori. A vizinhança dimensionada do ANN pode ser usada como função probabilidade a priori de um HMM sendo que, durante o reconhecimento, o fator de dimensionamento $p(x_n)$ é uma constante para todas as classes, portanto, não influi na definição das probabilidades a priori.

Fig.11 Modelo Híbrido



A figura acima representa o esquema híbrido básico, no qual o ANN gera probabilidades a posteriori que são transformadas em probabilidades a priori., e então, usadas para programação dinâmica . [46,54]

Comparação do ANN e métodos Convencionais

Uma vez que pode-se, essencialmente, derivar a mesma probabilidade com um ANN ou com um estimador convencional (por exemplo uma distribuição gaussiana), existem basicamente duas vantagens que devem ser observadas.

Reconhecedores estatísticos clássicos requerem "fortes" hipóteses sobre as características estatísticas do sinal de entrada.

Este tipo de hipótese não é necessária para um estimador ANN. Isto representa uma vantagem principalmente se uma mistura de diferentes tipos de padrões é utilizada. Especificamente, HMM clássicos assumem que os vetores acústicos consecutivos não possuem correlação. Para o estimador MLP, múltiplas entradas podem ser utilizadas para uma faixa de passos de tempo, e a rede irá aprender algumas coisas sobre a correlação entre as entradas acústicas. Note que o uso de tal rede irá levar a uma representação muito mais genérica das probabilidades de emissão. Isto significa que, se (c+d+1) frames de vetores acústicos $X_{n-c}^{n=d} = \{x_{n-c},...,x_n,...,x_{n+d}\}$ são usados como entrada para prover informações contextuais para a rede, as saídas da ANN irão estimar $p(q_k!X_{n-c}^{n+d})$, para qualquer k (k=1...K). O ANN apresenta-se bastante apropriados para funções de minimização do erro quadrático médio. Portanto as probabilidades serão otimizadas para a distinção entre classes de som.

Prós ANN	Contras ANN
Apropriado para cálculo da distância entre o vetor de teste e os centróides do sistema	Não apresenta uma boa abstração para sequências e aspectos temporais
Não exige hipóteses estatísticas necessárias nos HMM	

Tabela 3. Prós e Contras do ANN

Teoricamente, pode-se concluir que um modelo hibrido composto pela representação de sequências do modelo HMM com estimadores estatísticos baseados em ANN pode representar uma evolução sobre o modelo convencional HMM com estimadores clássicos gausseanos. Para a implementação deste protótipo, foi adotado o modelo HMM clássico com estimadores gausseanos clássicos Deixou-se os estimadores probabilisticos baseados em ANN como um possível ponto de aprimoramento do sistema . [46,54]

3 Implementação do Protótipo

3.1 Descrição do Projeto

Os objetivos básicos no processo de implementação de um sistema de reconhecimento de voz é a minimização dos fatores:

1. tamanho;
2. custo ;
3. consumo de energia ;
4. complexidade tecnológica ;

aliado à maximização de fatores tais quais:

1. taxas de reconhecimento de sinais vocais ;
2. velocidade de resposta a excitação vocal

Com o avanço das pesquisas de aperfeiçoamento da tecnologia de reconhecimento de voz, aliado aos investimentos em projetos de chips DSP dedicados ao reconhecimento de voz, equipamentos comandados por instruções vocais têm se tornado cada vez mais populares.

Em sintonia com esta realidade , neste experimento, desenvolveu-se hardware e software necessários para implementação de um protótipo conectável à porta de comunicação paralela de equipamentos tipo PC . Este chip DSP D6106 é baseado nos chips DSP programáveis da marca ALTERA . Todos os algoritmos de treinamento, reconhecimento e síntese de sinais vocais são pré-gravados em seu chipset .

As características básicas do sistema são:

Número de usuários

Para a definição deste parâmetro deve-se considerar as variáveis flexibilidade e desempenho. Idealmente o chip de comando vocal poderia suportar um volume ilimitado de usuários e comandos.

Para isso bastaria aumentar no projeto do sistema o dimensionamento da capacidade de memória RAM do sistema. Mas quanto mais usuários e comandos tivermos, menor serão a taxa de reconhecimento do sistema e a velocidade de resposta do mesmo .

Tendo em vista esta relação direta entre número de usuários ativos com o desempenho do sistema, utilizou-se um recurso conhecimento como paginação de memória RAM.

Por este princípio a memória é dividida de modo que, para cada usuário selecionado, tenha-se no máximo 20 comandos ativos. Este procedimento maximiza a taxa de reconhecimento do sistema, assim como a velocidade de resposta do sistema.

Dependência do usuário - treinador

O sistema só é capaz de reconhecer os comandos vocais do mesmo usuário responsável pelo procedimento de treinamento do mesmo. Ou seja um comando treinado pelo usuário só pode ser comandado por ele mesmo. Para que o sistema reconheça os comando de outro usuário, será necessário informar ao sistema a mudança de usuário, assim como a página ativa da memória RAM.

Seleção de Ambiente

O desempenho do sistema , medido na sua forma de taxa de reconhecimento e tempo de resposta, depende basicamente do ruído ambiental em que se procede o treinamento e o reconhecimento do sinal vocal. O chip DSP D6106 prevê a escolha de três níveis de ruído ambiental:

- Home - ambiente silencioso (exemplo: ambientes internos, sem emissões de comandos vocais secundários e sem fontes geradoras de ruído aparente tais quais rádios, alarmes, etc)
- Medium - ambiente compartilhado com outras fontes de ruído de baixa e média intensidade (exemplo: ambientes onde hajam outros emissores de comandos vocais e/ou fontes geradoras de ruído aparente tais quais rádios , alarmes, etc)
- Car - ambiente ruidoso com fontes de ruído de alta intensidade tal qual é o interior de um automóvel

A seleção do ambiente de operação evita que alterações ambientais impactem de modo sensível a taxa de reconhecimento do sistema . Isto significa que a taxa nominal de 97% pode ser atingida em diferentes condições ambientais. É evidente que isso é válido dentro de alguns limites; em ambiente com ruído intenso tal como em uma discoteca, ou bolsa de valores, o sistema apresentará taxas de reconhecimento muito inferiores a sua taxa nominal.

Taxa de Reconhecimento Nominal

A taxa de reconhecimento dependende do ruído ambiental e do número de comandos ativos. Mas em condições normais de ruído para um vocabulário de 20 comandos ativos o desempenho do sistema atinge a taxa nominal de 97%.

3.2 Conceitos Básicos Aplicados

Neste ítem , os conceitos desenvolvidos anteriormente na sessão teórica , foram retomandos sob o enfoque da implementação do protótipo.

3.2.1 Aquisição

O microfone é um fator chave na melhora da taxa de reconhecimento do sistema. A escolha do microfone é sempre uma relação entre custo e desempenho. Para os testes desenvolvidos com o nosso protótipo selecionou-se um microfone direcional de baixo custo marca TMS modelo M-11 utilizado por atendentes de telemarketing (a curva de sensibilidade do microfone está descrita no gráfico abaixo). Este microfone apresentou boa sensibilidade ao comando vocal e moderada captação de ruídos ambientais incrementando a relação Sinal-Ruído do sinal de entrada.

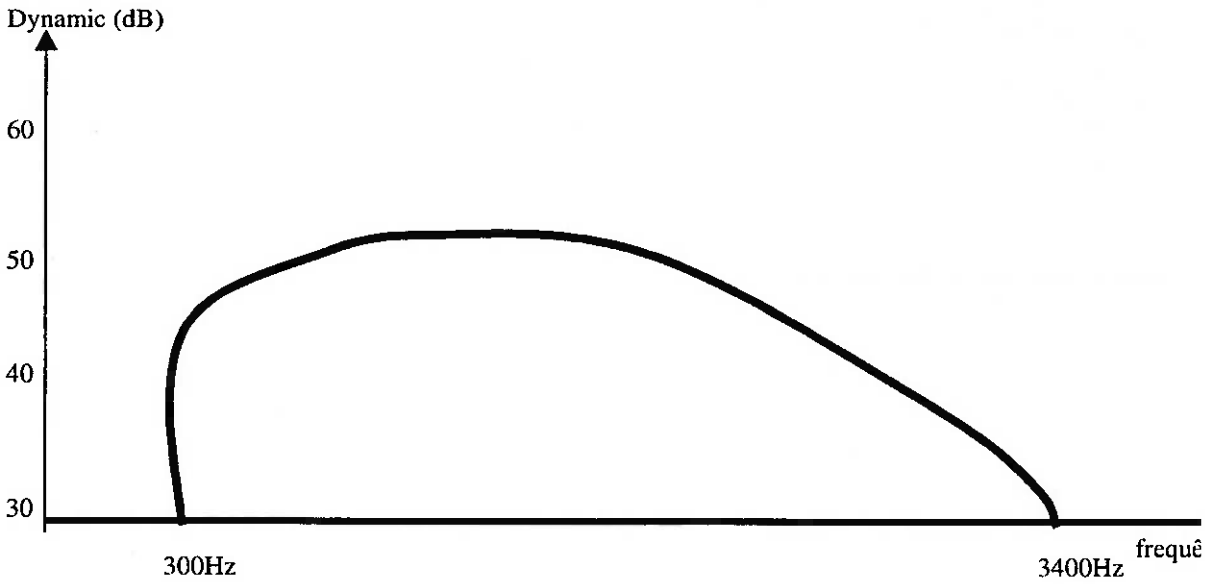


Fig.12 Curva de Sensibilidade do Microfone TMS M11 (informado na embalagem do microfone)

3.2.2 Pré-Processamento Acústico

O tratamento de filtragem e condicionamento do sinal de voz, seguido pela compactação e codificação do sinal de voz é executado através do TCM129C13 da Texas Instruments. O chip possui recursos de aplicação de filtro passa-banda, conversão digital de 16 bits com modelamento e extração de parâmetros nos padrões LPC

3.2.3 Modelamento e extração de parâmetros

O conceito utilizado para o modelamento e extração de parâmetros dos comandos vocais é o LPC. Os algoritmos de LPC estão programadas no chipset do chip TCM129C13 da Texas Instruments

3.2.4 Treinamento

Na etapa de treinamento, os parâmetros LPCs obtidos na fase de modelamento e extração de parâmetros, são transformados em vetores. Trata-se, portanto, da definição das referências as quais são medidos os níveis de similaridade na fase de reconhecimento.

Na prática, a fase de treinamento já é executada pelo chip DSP D6106 através de subrotinas internas ao processador dedicado DSP.

Os comandos vocais de teste são adquiridos, parametrizados e então vetorizados formando os centróides e respectivas Regiões de Voronoi.

O livro código é então gerado utilizando-se o algoritmo LBG.

3.2.5 Reconhecimento dos Comandos Vocais

O processamento do reconhecimento dos Comandos Vocais é executado pelo próprio chip D6106, seguindo o algoritmo baseado no modelo HMM com estimadores probabilísticos clássicos.

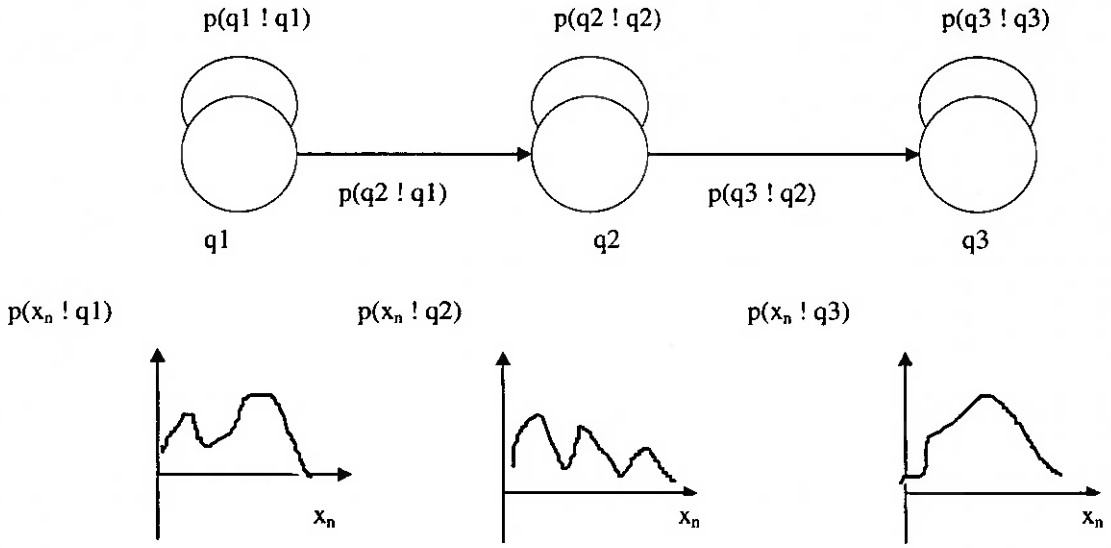


Fig.13 Ilustração algoritmo de reconhecimento HMM

3.3 Interface Digital

Em se tratando de programação de um software de comunicação entre o chip DSP e uma CPU do tipo PC, a primeira questão foi a opção existente entre desenvolver uma solução local do tipo "stand alone" ou então, desenvolver uma solução compartilhada do tipo "networking solution".

A maior vantagem da solução em rede é o compartilhamento de uma única CPU DSP de reconhecimento de voz. Para executar um comando de voz o usuário deveria se conectar a CPU servidora e proceder a transmissão de dados conforme o protocolo específico de cada rede. Os mais usuais seriam TCP/IP , IPX, Netbios.

Em se tratando de aplicações cotidianas, as de interesse comercial seriam: Transações Bancárias Via Internet, Servidores de Alta Segurança para transações de Cartão de Crédito, Sistema de Identificação em Caixas Eletrônicos, Sistemas de Acesso a áreas restritas e residências inteligentes.

A maior desvantagem deste tipo de solução é que anteriormente a procedimentos de treinamento e reconhecimento de sinal vocal, os sinais de voz têm de ser adquiridos e transmitidos através de arquivos sonoros compactados, estando portanto, sujeito à perda de qualidade do sinal sonoro . Esta perda resulta em modificações sensíveis das taxas de acerto do sistema de reconhecimento do sinal vocal.

Como o propósito deste projeto é a implementação de um sistema de reconhecimento de voz baseado em chip DSP dedicado de baixo custo individual, não se considerou o estudo de uma solução em rede.

Sendo "stand-alone" , pode-se optar por projetar uma placa em que se adicionou o chip DSP na própria motherboard do processador ("on board") ou então, optar pela manutenção da concepção da placa DSP como um acessório universal compatível a vários tipos de motherboards.

Na verdade, ambas as concepções são igualmente viáveis e importantes, mas neste caso em que se pretendia utilizar como instrumento de comando e operação um equipamento do tipo PC, optou-se pela configuração universal e conexão de comunicação via porta paralela.

3.4 Modos de Operação do Chip D6106

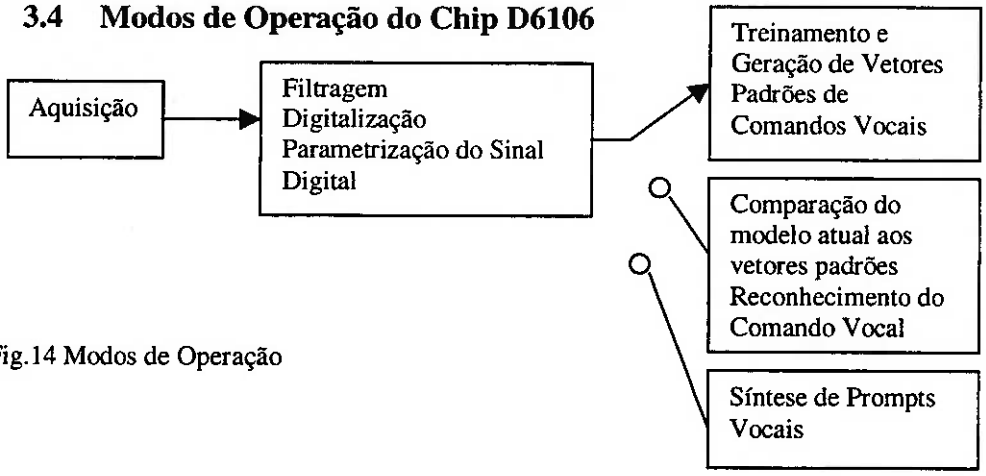


Fig.14 Modos de Operação

3.4.1 Modo de Treinamento

O modo de treinamento dá ao usuário a flexibilidade de criar seu diretório com os comandos desejados. O treinamento de um sistema, dependente de usuário, torna o mesmo

independente de língua e sotaque regional ou nacional. O modo de treinamento influi diretamente na taxa de reconhecimento .

O procedimento de treinamento precisa apenas ser executado uma única vez para cada comando desejado. Mas caso o usuário deseje poderá modificar a lista de comando vocal, deletá-lo ou então adicionar um novo comando.

Procedimentos de treinamento com baixo ruído ambiental e alta taxa SNR (relação sinal / ruído) são essenciais para se atingir altas taxas de reconhecimento.

No modo de treinamento, o sinal vocal é adquirido e filtrado através do chip CODEC e então transmitido ao chip DSP D6106, onde modelos do sinal vocal são criados e armazenados na SRAM. Para medir a qualidade do sinal, adquirido na fase de treinamento, o DSP D6106 calcula o fator SNR (relação sinal ruído), que representa a proporção entre o sinal vocal e o ruído ambiental, e o transmite ao servidor. Além disso o DSP retorna um código de erro caso os níveis de SNR não sejam aceitáveis .

Quando o procedimento de treinamento for falho o sistema requisita, ao usuário, um novo procedimento de treinamento para o comando vocal em questão.

Além disso o DSP D6106 também pode ser instruído para calcular o nível de similaridade entre dois modelos do mesmo comando vocal.

O modo de treinamento é usado para gerar os modelos para a fase de reconhecimento. É possível se estabelecer internamente até oito usuários. O modo de treinamento também pode gerar uma forma comprimida de cada palavra treinada, tendo uma taxa de compressão de 7.7 kHz.

Um comando vocal pode ter uma duração entre 0.2 e 1.1 segundos. Cada vocabulário pode incluir mais de 127 comandos. Como vocabulários menores resultam em maior acuidade de reconhecimento e melhor tempo de resposta é aconselhável subdividir vocabulários em sub vocabulários menores , como por exemplo subvocabulários de 16 comandos.

No modo de treinamento, um usuário pronuncia os comandos individualmente, constituindo um conjunto de comandos .

O usuário especifica uma janela de tempo no qual o sinal Audio In é captado. Durante esta janela de tempo esperasse que o usuário emita um comando vocal. O D6106 indica a duração da janela usando uma linha de interface especial LSTN. A duração da janela de tempo é informada ao usuário através de um sinal luminoso (led) representando a espera do sistema por um comando vocal.

O comando vocal é adquirido, filtrado e digitalizado pelo CODEC LPC e processado pelo chip DSP D6106.

O D6106 possui algumas funcionalidades que podem ser usadas na verificação e otimização do processo de treinamento.:

- Indicação no caso em que nenhum comando vocal é emitido durante a janela de treinamento, ou quando o sinal vocal excede os valores máximo ou mínimo desta janela.

- Indicação do caso em que o sinal vocal não é satisfatório no que se refere ao nível absoluto do sinal (muito baixo ou muito alto) ou quando a relação sinal/ruído está abaixo de um valor pré-definido (threshold)

- Função de Comparação de padrões. A função COMPARE compara uma dado modelo com outros modelos especificados pelo servidor e provê um parâmetro proporcional as diferenças encontradas entre os mesmos. Este parâmetro pode ser usado para detetar procedimentos de treinamento que resultem em uma alta taxa de reconhecimentos incorretos

3.4.2 Modo de Reconhecimento

O chip DSP D6106, no modo de reconhecimento, compara o comando vocal a ser reconhecido aos modelos pré-armazenados no subdiretório do usuário ativo.

O chip DSP D6106 reconhece o modelo com maior similaridade ao sinal vocal adquirido e ativa o comando pré-programado.

O algoritmo neste caso é o HMM (Hidden Markov Models), com estimadores probabilísticos clássicos.

Além do modelo de maior similaridade, o D6106 também identifica um ranking dos três modelos suplementares com maior taxa de similaridade ao sinal vocal análise.

Para esta função, além das informações do nível de similaridade dos modelos, o chip DSP transmite o valor SNR do sinal vocal atual conjuntamente a um código de erro. Deste modo, caso o sinal vocal tenha um SNR muito baixo ou o nível de similaridade caia abaixo

de um nível de similaridade aceitável, o DSP D6106 retorna um código de erro e requisita ao usuário uma nova entrada de comando.

Os valores limites de SNR e Similaridade podem ser pré-definidos ou modificados assim que o usuário achar necessário.

Os algoritmos necessários para a operacionalização do reconhecimento do sinal podem variar. Para este caso os algoritmos pré-implementados no chip DSP D6106 são:

VAD - "Voice Activation Detection" , algoritmo para deteção de pontos de silêncio e pontos de emissão sonora real, além da deteção do início e fim do comando vocal [10]

LP - "Linear Prediction" , algoritmo para análise de predição linear para a extração de características espectrais [10]

LTW - "Linear Time Warping" algoritmo para truncagem do sinal vocal quando a duração dos mesmos excede os padrões pré-selecionados. [10]

DTW - "Dynamic - Time Warping" algoritmo para adaptação e suavização de pequenos erros de modo a otimizar o desempenho em ambientes ruidosos. [10]

No modo de reconhecimento o D6106 provê uma comparação entre o comando vocal atual e os modelos pré analisados na fase de treinamento. O resultado desta comparação é expresso na forma de um parâmetro numérico. O parâmetro numérico é um índice que indica o nível de discrepância entre o comando vocal e os modelos da fase de treinamento. Quanto menor é o valor numérico deste parâmetro, maior é o nível de similaridade do comando vocal ao modelo em análise.

O D6106 pode lidar com uma ampla faixa de ruído no comando vocal (SNR), permitindo que se obtenha resultados confiáveis mesmo em ambientes ruidosos.

Vale lembrar que o tempo de resposta aumenta proporcionalmente ao aumento do ruído, ou seja, redução do SNR. Para minimizar os efeitos destes ruídos ambientais o D6106 possui opção de se selecionar o nível de ruído de operação do sistema

Seleção	**Descrição**
Home	Provê o menor tempo de resposta, mas só pode ser utilizado em ambientes silenciosos com SNR superior a 18 dB
Medium	Provê seus melhores resultados em ambientes com níveis de ruído intermediários. (Entre 5 e 18 dB)
Car	Provê boa acuidade quando o ruído ambiental é alto. Aconselhável para SNR menor que 5dB. Seu limite está próximo aos 3dB quando então , nem mesmo neste modo de operação, o sistema apresenta bons resultados

Tabela 4 modos de operação

Um conjunto completo de resultados de reconhecimento incluem:

A identificação do padrão com maior nível de similaridade ao comando vocal em análise
Os scores de identificação dos três padrões com melhor nível de similaridade ao modelo em análise
O SNR e o nível relativo do comando vocal
Código de Erro do Processo. Valores não nulos indicam diferentes tipos de falha no processo de reconhecimento

Antes de iniciar o processo de abertura da janela de tempo, correspondente a fase de captação do comando vocal, o servidor deve especificar parâmetros específicos .

O processo de reconhecimento segue uma determinada sequência de eventos:

Seguindo o comando do servidor, o D6106 abre uma janela de aquisição e espera que o comando vocal seja recebido pelo CODEC

Quando uma entrada é detetada o comando vocal é digitalizado e analisado no sentido de comparar seu nível de similaridade a padrões previamente analisados na fase de treinamento

Caso o comando vocal atenda a critérios de qualidade do sinal vocal, então a melhor escolha é selecionada e o resultado é transferido ao servidor, assim como os valores de SNR, nível relativo do sinal, parâmetro numérico do padrão escolhido assim como um ranking dos três modelos com miar nível de similaridade (menor parâmetro numérico de comparação)

3.4.3 Modo de Síntese

O modo de síntese permite que o sistema reproduza os modelos previamente armazenados.

Dois tipos básicos de prompt de síntese podem ser considerados :

O primeiro é o prompt original do sistema e o segundo poderia ser definido como o prompt definido pelo usuário.

Os prompts do sistema são pré-armazenados em memória EPROM. O segundo tipo de prompt são os modelos gravados na SRAM através do procedimento de treinamento.

3.5 Arquitetura Teórica

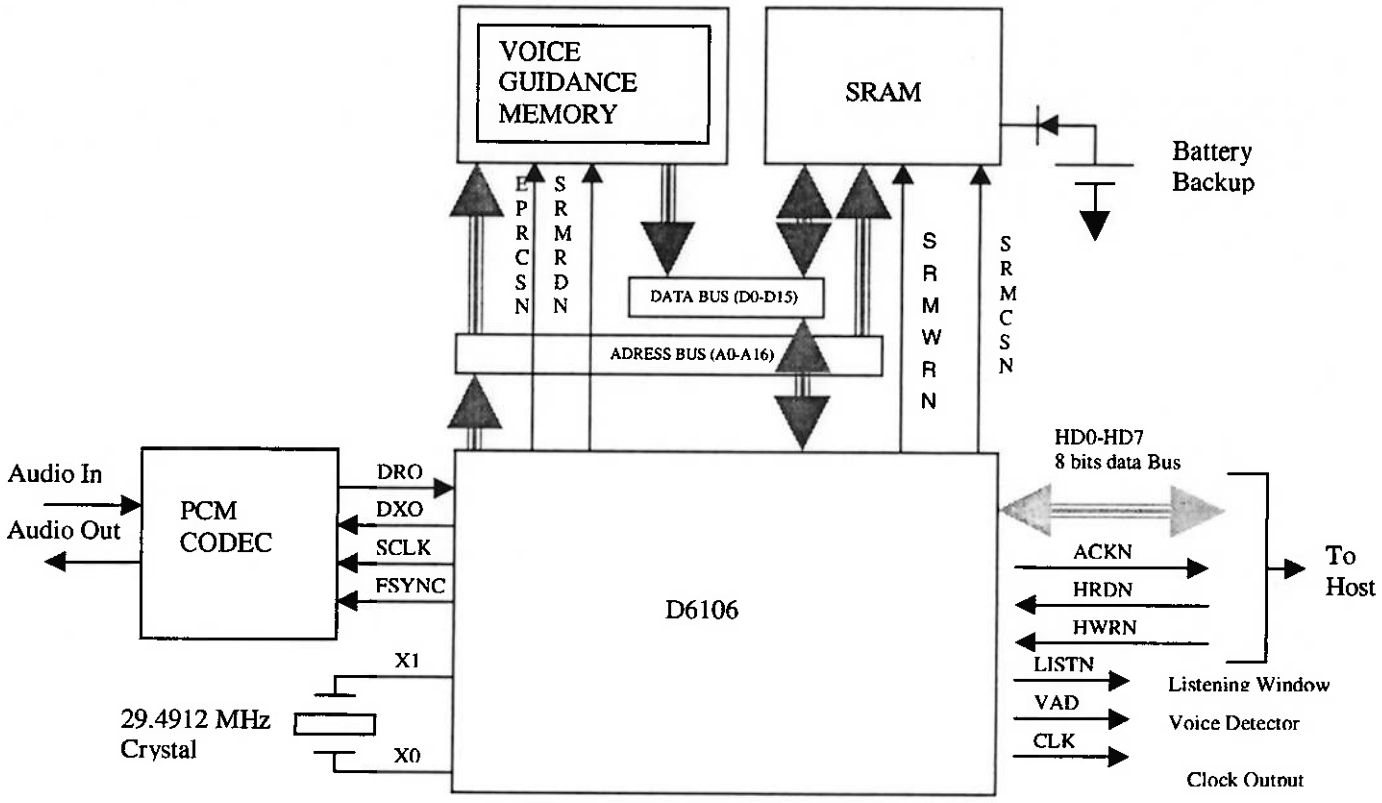


Fig 15 : Arquitetura do Sistema

O projeto de um típico sistema comandado por voz inclue um processador DSP de comando vocal D6106, codec, SRAM, EPROM, microfone, e alto falantes.

O D6106 é controlado por um computador do tipo PC através de uma porta paralela bidirecional padrão.

A entrada de áudio analógica, capturada através do microfone, é codificada no CODEC e então submetida ao DSP D6106 onde a função de reconhecimento é executada. A saída do chip DSP é decodificada pelo CODEC e re-transformada em um sinal analógico de voz.

Configurado com uma memória de 32Kb ou de 128Kb o DSP D6106 suporta até 128 comandos vocais distintos.

A capacidade da memória SRAM do sistema pode ser aumentada ou reduzida, de modo a se adaptar ao tamanho de vocabulário desejado. Em se escolhendo a quantidade de memória SRAM e EPROM para um projeto específico, um projetista deve balancear sua relação com o nível de reconhecimento do sistema e a velocidade de resposta do mesmo. Uma bateria de backup também é aconselhável para se evitar perda de dados adquiridos em processo de treinamento com o usuário.

O fabricante do chip adverte que é aconselhável a limitar o vocabulário ativo em 20 palavras para que se mantenha um percentual de confiabilidade do sistema em 97%.

Como o DSP D6106 aceita partição de memória, pode-se criar diversos diretórios respeitando o limite máximo de 20 palavras ativas e consequentemente a confiabilidade do sistema.

3.5.1 Processador de Comando Vocal D6106

O coração do sistema é um processador de comando Vocal D6106. O D6106 é um chip VLSI especial que inclui um DSP proprietário e funções adicionais que garantem um sistema eficiente e de boa relação custo-benefício.

Desempenho

- Tamanho do Vocabulário
  - Máximo 127 comandos por usuário
  - Recomendado 16 comandos organizados em subvocabulários
- Tempo de Resposta
  - 0.5 segundo modos Home/medium
  - Entre 0.5 segundo e 01 segundo modo car
- Taxa de Reconhecimento
  - 97% vocabulário de 16 palavras e SNR superior a 3dB

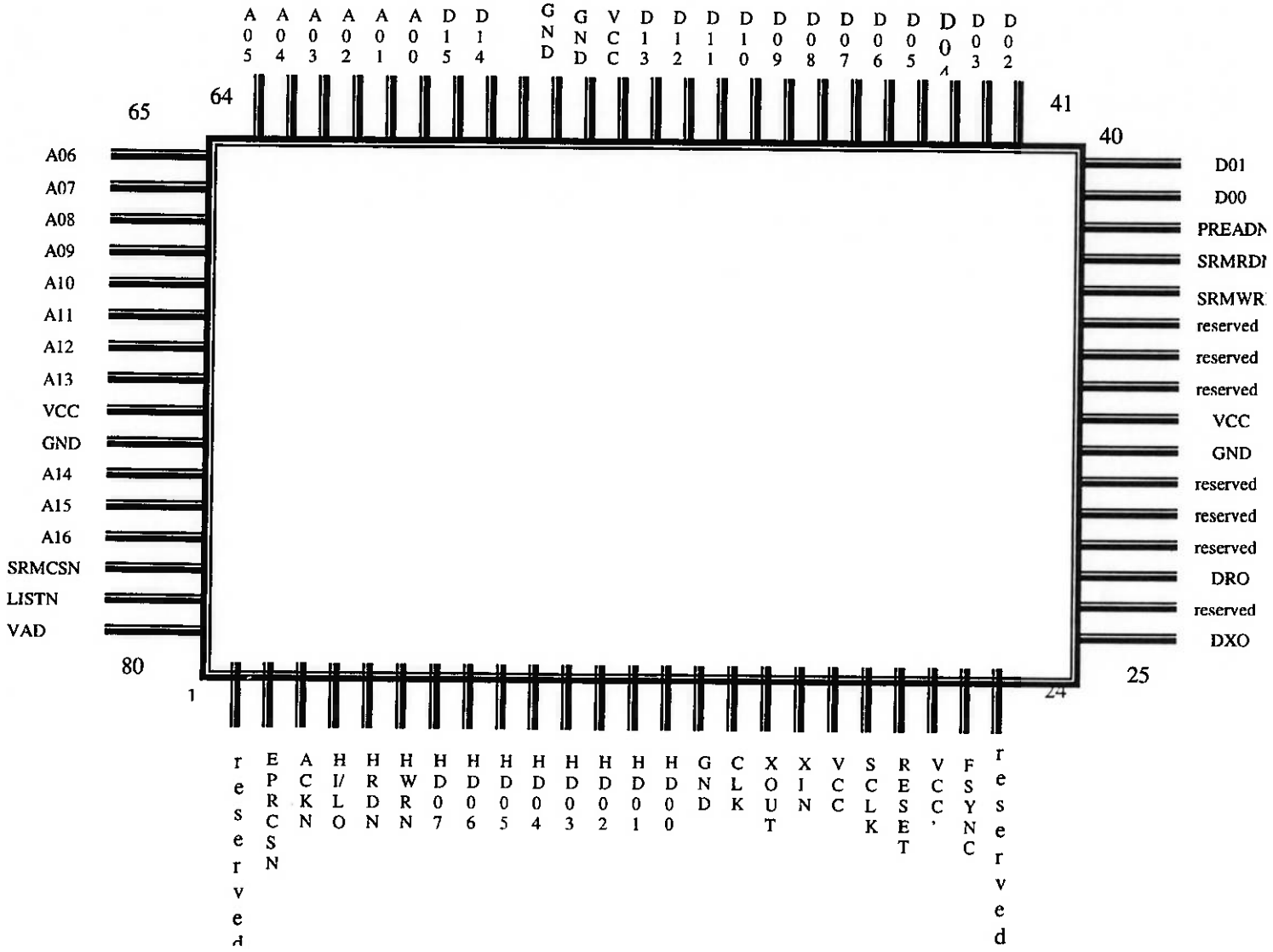
Geração de Prompt Vocal

- Taxa de 7.7 Kbps
- 32Kbyte ROM para 34 segundos de prompts vocais
- Duração de prompt de 1.1 segundos

Múltiplos Usuários/Partição de Memória

- Até 08 usuários/partições
- Apenas 01 usuário ativo simultaneamente

Pinagem D6106



Abreviação

- I Entrada de Sinal Digital
- O Saída de Sinal Digital
- I/O Barramento de Dados Bidirecional
- P Suprimento de Energia ou linhas de terras
- (PD) resistores "pull-down" (100kΩ)
- Z Estado de alta impedância

Fig 16. Pinagem do chip

3.5.2 PCM codec Interface

Usado para transferir dados e sinais de temporização entre o D6106 e o CODEC

3.5.3 Interface de memória

Usado para ler dados da SRAM e a memória de armazenagem vocal (ROM ou EPROM), assim como para gravar dados na SRAM. A interface inclue "buses" independentes de endereço e dados, e gera os sinais de controle necessários para acessar a memória. Um gerador interno de "wait state" permite ajustar o tempo de acesso de acordo com a velocidade da memória.

3.5.4 Oscilador de Clock

Gera o sinal interno de clock do D6106, usando um cristal conectado aos pinos X0 e X1 do D6106.

3.5.5 Host Interface

Usado para transferir comandos, status e outros dados entre o D6106 e o microcontrolador servidor. A interface também inclue as linhas de handshake.

O D6106 inclue uma função de auto-teste que pode ser comandado pelo servidor. A função testa o código interno da ROM, assim com da SRAM, EPROM/ROM externas.

O D6106 é encapsulado em um chip de 80 pinos quadrado e requer apenas uma alimentação de 5V DC para operação.

3.5.6 CODEC LPC

O D6106 requer um Codec do tipo LPC que serve como "front end" analógico para o D6106. O CODEC LPC é usado para converter o sinal de voz analógico, Audio IN, em dados de 64kbps codificados no padrão LPC, DRO, que será o dado de entrada do DSP, assim como para converter o sinal DXO LPC 64kbps em sinal analógico Audio Out usado para sintetizar sinais vocais pré-gravados.

Os valores nominais e a faixa de espectro dos sinais analógicos audio in e audio out, são determinados pelas características do codec LPC.

O D6106 provê um sinal de clock SCLK, e um sinal de framing, FSYNC para o codec LPC. Dois tipos de CODEC são suportados:

- TI TCM29C16
- OKI MSM7508

3.5.7 SRAM

A memória SRAM é usada para armazenar os parâmetros de reconhecimento, e os modelos de comandos vocais treinados. Como uma opção, o D6106 pode armazenar os comandos vocais, no formato compactado, de modo a maximizar o número de comandos vocais armazenáveis.

O D6106 pode endereçar as configurações de memórias SRAM de 1x32kbytes até 2x128kbytes. Isto corresponde a um vocabulário mínimo de 16 comandos vocais ao máximo de 127 comandos.

O D6106 pode trabalhar com SRAM de tempo de acesso de 35 a 150ns

ROM/EPROM

O dimensionamento de ROM/EPROM depende da duração do prompt vocal. O D6106 suporta as seguintes configurações de ROM e EPROM

- 1 x 32k x 8bits: aproximadamente 34s
- 2 x 32k x 8bits: aproximadamente 68s
- 1 x 128k x 8bits: aproximadamente 136s
- 2 x 128k x 8bits: aproximadamente 272s

A memória ROM/EPROM pode ser dimensionada pela fórmula:

O D6106 necessita de memória PROM/EPROM com tempo de acesso inferior a 300 ns

$$Memo\ (bytes) = (19 + 2 * N + 480 * T) * 2$$

3.5.8 ROM/EPROM

A memória ROM ou EPROM é um componente opcional que pode ser usado para gravar comandos vocais compactados.

O D6106 pode acessar diretamente a memória ROM ou EPROM de 1x32kbytes até 2x128kbytes com velocidade mínima requerida é de 300 ns. Uma vez que a taxa de digitalização requerida pela função interna de síntese é de 7.7kbps, estas configurações correspondem de 0.5 a 4.5 minutos de dados compressos. O total de tempo de prompts pré-gravados pode aumentar utilizando-se EPROM e ROMs adicionais controlados pelo servidor.

3.5.9 Cristal

O D6106 requer um cristal de 29.4912Mhz com acuidade de 100 ppm

Quando for necessária maior capacidade de memória , bancos de memória adicionais podem ser usadas sob controle do servidor . O D6106 suporta memórias SRAM com velocidades entre 35 e 150ns.

Uma vez que os dados armazenados na SRAM são geradas pelo usuário, a SRAM precisa incluir um bateria de backup, de modo a prever perda de dados armazenados e uma nova sessão de treinos pode ser requerida.

Configuração de Memória Externa

SRAM

O tamanho da SRAM depende basicamente do número de usuários e do tamanho do vocabulário. O D6106 suporta as seguintes configurações de SRAM

- 1 x 32k x 8 bits
- 8 x 32k x 8 bits
- 1 x 128k x 8 bits
- 2 x 128k x 8 bits

Para calcular o tamanho de SRAM requerida para uma determinada aplicação, a fórmula abaixo descrita pode ser utilizada :

$$Memo\ (bytes\ ) = 10{,}000 + \frac{N_{UD}}{M_{TW}} + P_D * 28 + S_D * 28 + (N_{UD} + N_{PD}) * 196$$

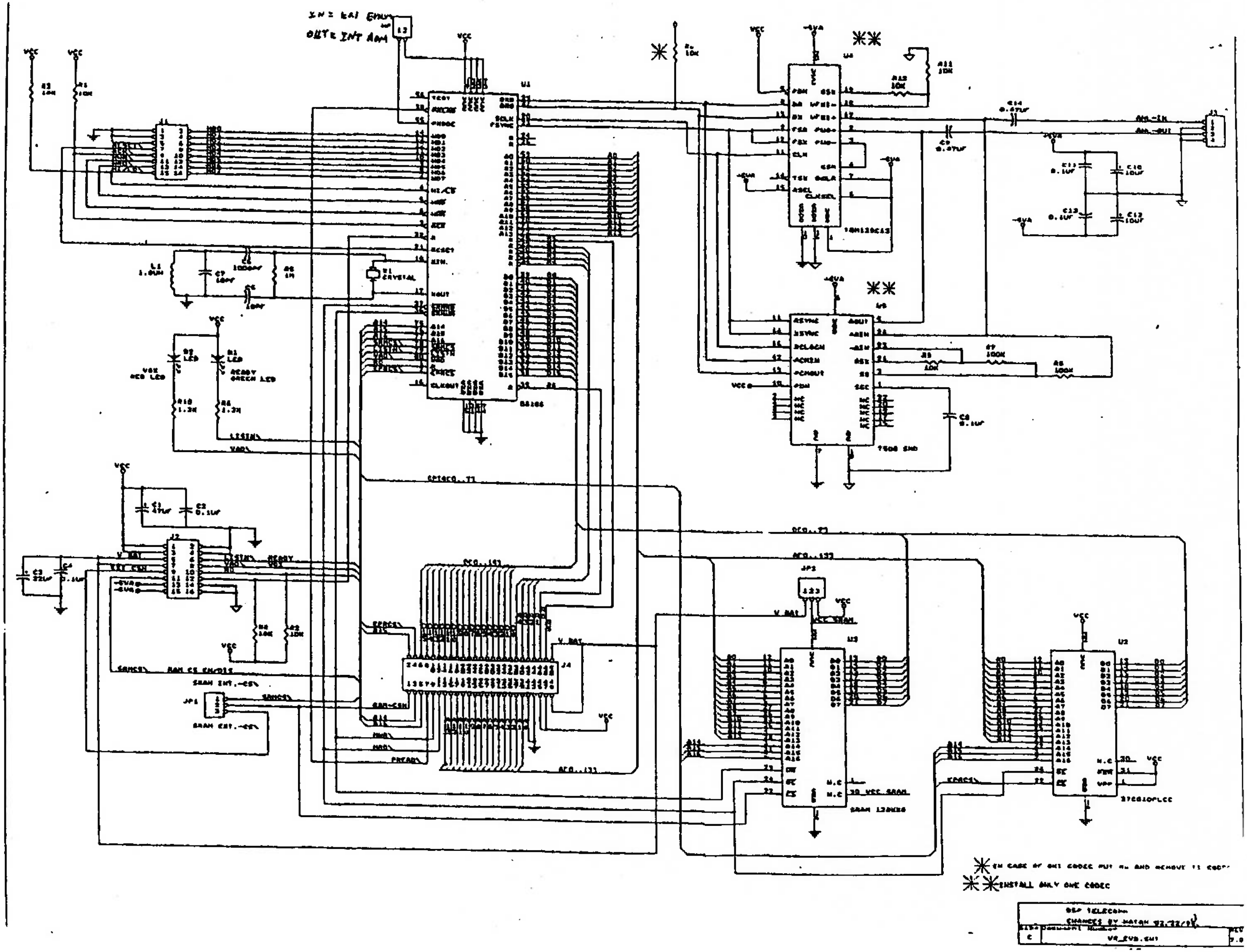
Onde:

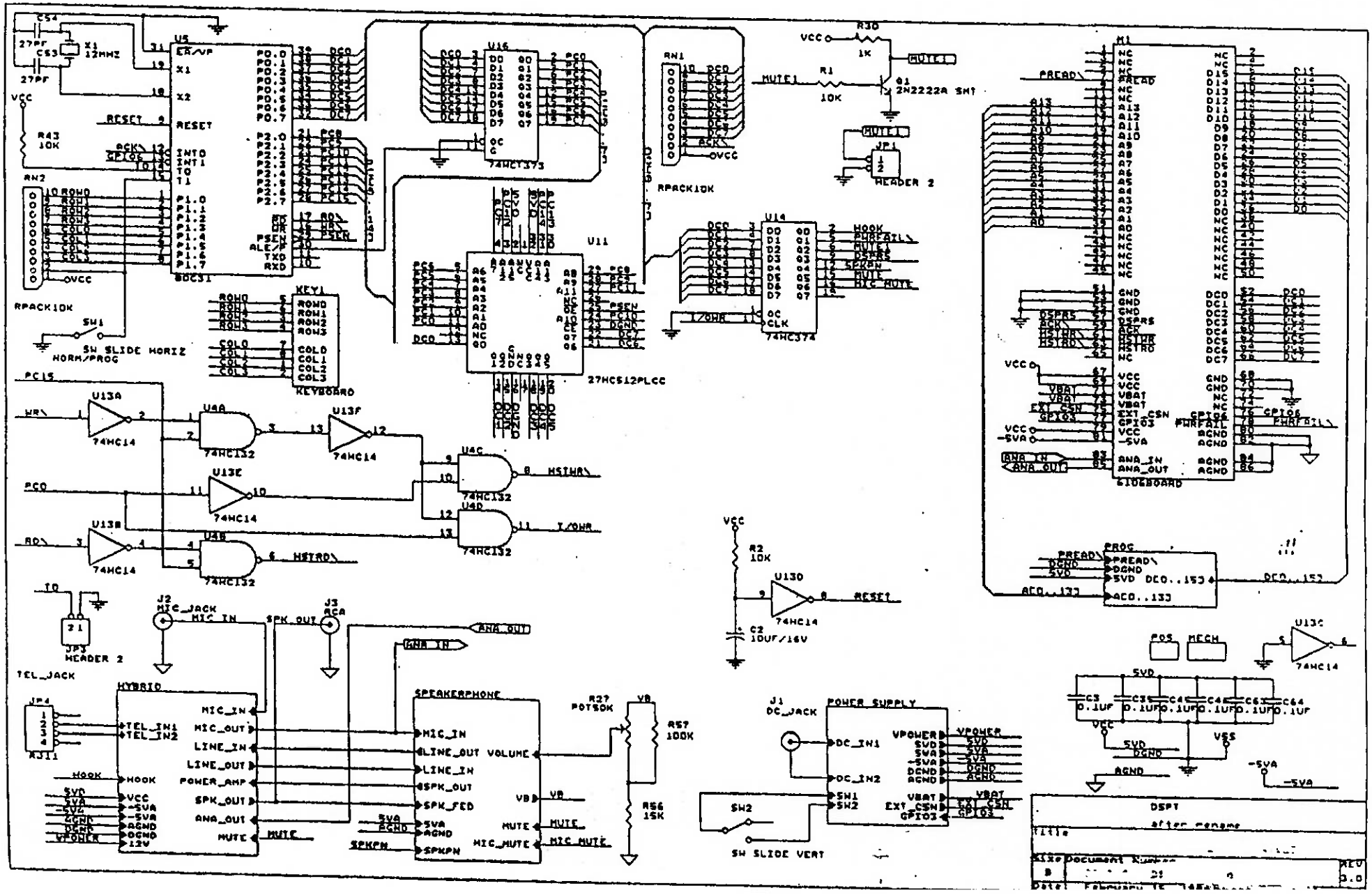
$N_{UD} - Número\_de\_Usuários$
$M_{TW} - Número\_de\_Templates\_por\_Palavra\_do\_Vocabulário$
$N_{PD} - Número\_de\_Templates\_Pré\_definido$
$P_D - Duração\_do\_Prompt\_em\_unidades\_de\_29ms$
$S_D - Duração\_da\_Janela\_de\_Retreinamento\_em\_unidades\_de\_29ms$

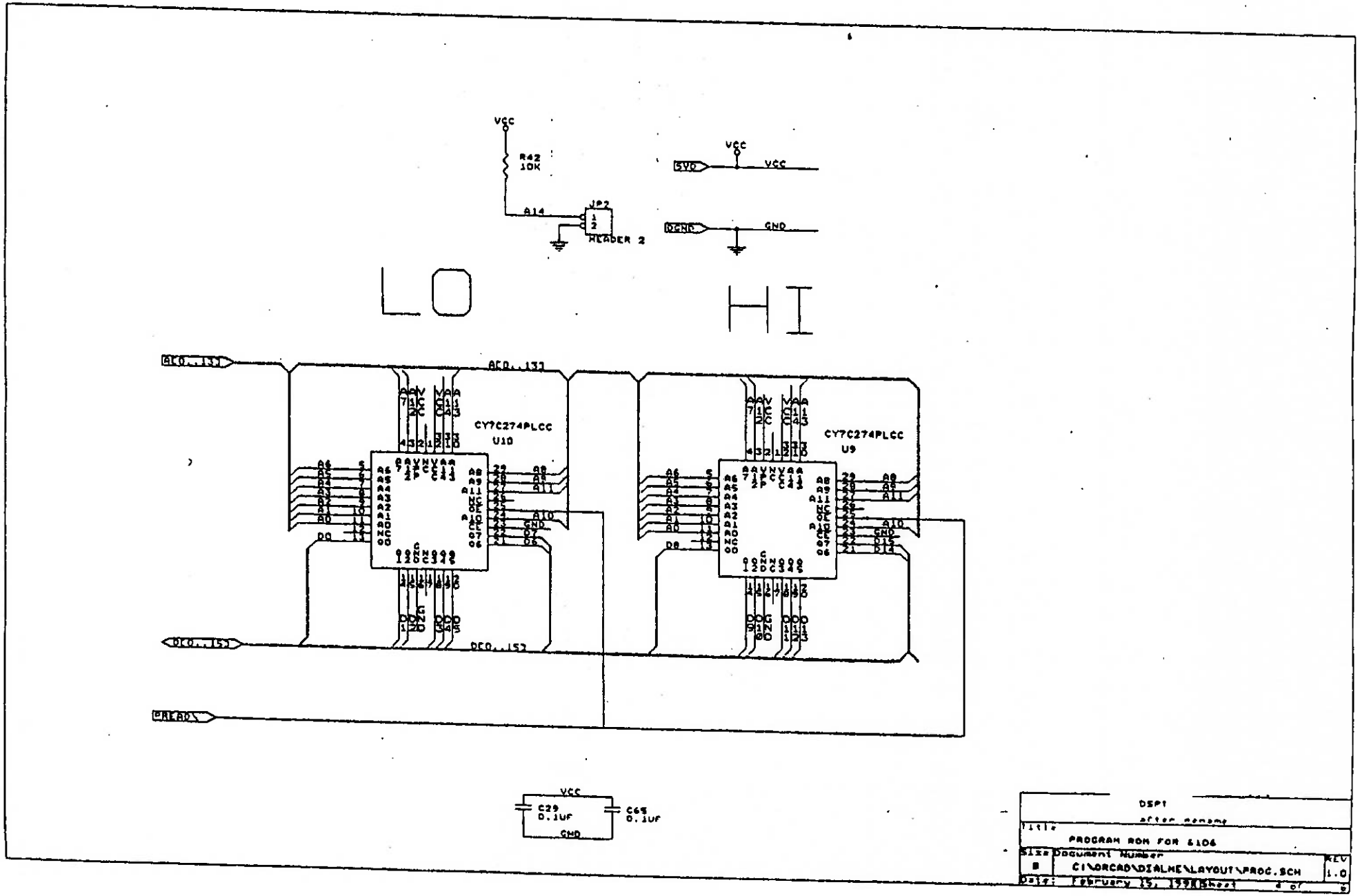
3.5.10 Microcontrolador Host

Qualquer microcontrolador pode ser utilizado como host do D6106. A interface com o host consiste de um bus de dados bidirecional de 8 bits e sinais de controle.

3.6 Diagrama da Placa


U1
D8166
U2
27C010PLCC
U3
SRAM 128K8
U4
TCM129C13
U5
7508 SND
INSTALL ONLY ONE CODEC
DSP TELECOM
VR_EVB.SHT


C54
27PF
C53
27PF
X1
12MHZ
U5
80C31
RESET
R43
10K
RPACK10K
RN2
SW1
SW SLIDE HORIZ
HORM/PROG
KEY1
KEYBOARD
U16
74HCT373
RN1
RPACK10K
U11
27HC512PLCC
U14
74HC374
R30
1K
MUTE1
R1
10K
Q1
2N2222A SMT
JP1
HEADER 2
M1
SLIDEBOARD
U13A
74HC14
U4A
74HC132
U13F
74HC14
U4C
HSIWR\
U13E
74HC14
U4D
I/OWR\
U13B
74HC14
U4B
HSIRD\
J2
MIC_JACK
J3
RCA
JP3
HEADER 2
TEL_JACK
ANA_OUT
ANA_IN
HYBRID
JP4
RJ11
SPEAKERPHONE
R27
POTS0K
R57
100K
R56
15K
SW2
SW SLIDE VERT
J1
DC_JACK
POWER SUPPLY
R2
10K
U13D
74HC14
C2
10UF/16V
PROG
U13C
74HC14
C3
0.1UF
C35
0.1UF
C4
0.1UF
C44
0.1UF
C6
0.1UF
C64
0.1UF
DSPI
after rename


VCC
R42
10K
A14
JP2
HEADER 2
VCC
5VD
VCC
DGND
GND
LO
HI
A[0..13]
A[0..13]
CY7C274PLCC
U10
CY7C274PLCC
U9
D[0..15]
D[0..15]
PREAD
VCC
C29
0.1uF
C65
0.1uF
GND
DSP1
after memory
PROGRAM ROM FOR &106
Size
B
Document Number
C:\ORCAD\DIALME\LAYOUT\PROG.SCH
REV
1.0
Date: February 15, 1994
Sheet of

3.7 Software

3.7.1 Descrição Geral

O software necessário ao comando do nosso hardware D6106 pode ser subdividido nos componentes:

1. Sistema Operacional da Máquina Servidora
2. Ambiente de Desenvolvimento das Rotinas de Comunicação, Comando e Interface Gráfica
3. Definição de Variáveis Globais para a Aplicação
4. Rotinas de Baixo Nível para Comunicação via Porta Paralela 8 bits
5. Rotinas de Baixo Nível para Comunicação com Bancos de Dados
6. Rotinas de Baixo Nível de Comando do D6106 que utilizando as rotinas de comunicação de baixo nível proporciona a efetiva comunicação com o D6106
7. Rotinas de Lógicas de Alto Nível para Processar as respostas fornecidas pelo D6106
8. Interfaces Gráficas de Alto Nível para Iteração com o usuário
9. Interfaces Gráficas de Alto Nível para Setagem dos parâmetros da placa

3.7.2 Protocolo de Comunicação com o Processador Servidor

O protocolo de comunicação para a interface entre o chip D6106 e o Intel PC Pentium 133 pode ser implementada como segue:

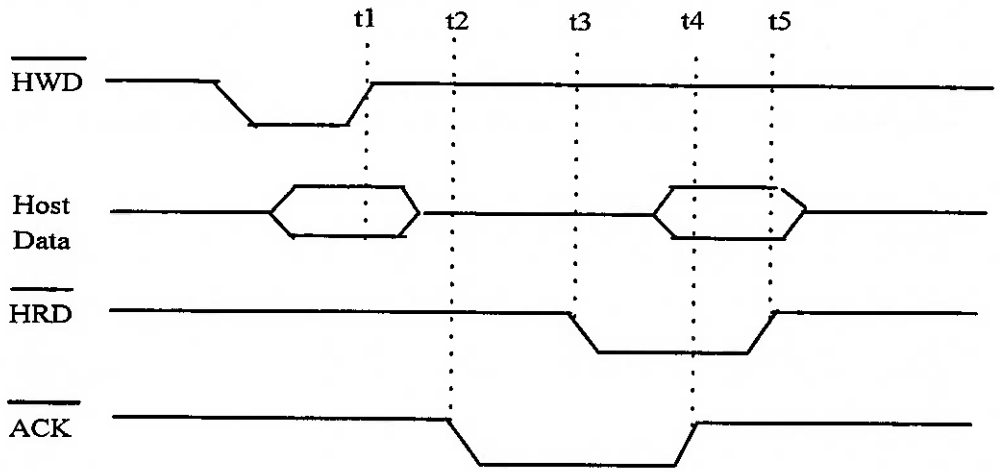


Fig. 17 Protocolo de Comunicação

t1 O servidor escreve o comando

t2 O DSP D6106 responde zerando o sinal ACK

t3 O servidor responde e lê o status

t4 O D6106 seta ACK

t5 O servidor lê o status

Sequência:

01. O servidor escreve na porta de servidor do D6106 executando o "strobing" do pino HWRN

02. O D6106 responde escrevendo o status para o servidor. Com a resposta do D6106 o sinal ACKN é zerado

03. O servidor lê a resposta do D6106 que também seta o sinal ACKN

Para este projeto, o software que emula o protocolo de comunicação no servidor é implementado com funções de dois níveis:

Send_Byte

Send_Command

A função send_byte é necessária para que o processador servidor envie um byte ao chip D6106.

O processador servidor aguarda pelo sinal ACKN e então lê como resposta o byte de resposta enviado pelo D6106.

As linhas de código a seguir exemplificam como se pode implementar esta função send_byte:

```
BYTE send_byte(BYTE data)
{
out_D6106_data(data);
wait_D6106_ack();
return(in_D6106_data());
}
```

A função send_command é usada para escrever bytes do processador servidor no D6106 e lê a resposta do D6106 e informa na forma de dados hexa ao servidor. Se o comando e a resposta não forem coincidentes, esta função trata o erro (isto é resetando o D6106). Esta função é usada quando o chip DSP ecoa o sinal enviado pelo servidor.

Um exemplo de linhas de código da função send_command:

```
void send_command(BYTE command)
{
```

```
BYTE respons;
respons=send_byte(command);
if (respons!=command)
handle_error();
}
```

3.7.3 Sequências do Software Servidor

O software de controle a ser executado no processador servidor tipicamente consiste de 04 sequências operacionais:

01. Inicialização do Sistema

02. Síntese de Palavras

03. Treinamento de Palavras

04. Reconhecimento de Palavras

A interface humana e o software de controle no servidor integram um conjunto que confere um alto desempenho aos sistemas comandados por voz.

3.7.4 Inicialização do Sistema

A sequência de inicialização do D6106 a ser comandado pelo servidor é ilustrado abaixo:

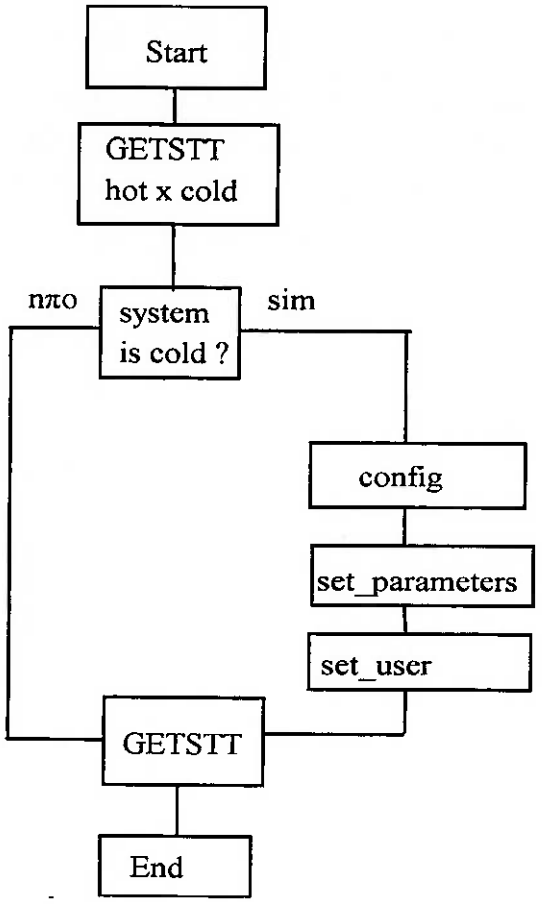


Fig.18 Inicialização

```
void
{
send_command(SETPRMC|0x20|0x10); /* send train SNR
send_command(0xF
send_command(2000 >>8) ;          /* send acceptance threshold -
send_command(2000 & 0x00FF) ;      /* low
send_command(3000>>8) ;           /* send rejection threshold -
send_command(3000 & 0x00ff);      /* low
send_command(-3*4+128);           /* send recognition
send_command(0*4+128);            /* send train
send_command(45);                 /* send min. level
send_command(65);                 /* send max. level
send_command((15*1000)/233);      /* listen_window duration is in units
```

Posteriormente ao procedimento de energização da placa o servidor deve ressetar o D6106 por pelo menos 100ms. Após este procedimento o servidor precisa executar um procedimento de inicialização que determina se o sistema foi ou não previamente configurado. Se o status do D6106 for "cold status" ou seja não existe configuração ativa no servidor o servidor precisa configurar o sistema. Caso o status do sistema seja diferente do "cold start" nenhuma configuração é necessária.

O comando GETSTT deve ser implementado para ler o status do usuário e do sistema. O servidor precisa destas informações para determinar o número de palavras previamente treinadas(para um cold start este número será zero)

As subrotinas de inicialização são descritos a seguir:

1. Initiate_dsp

Esta função inicializa o D6106 após sua energização

Se o status do sistema for "cold status", a função configura as características básicas do D6106

A função retorna o número de palavras pré-definidas do sistema em operação

O grupo seguinte de subrotinas é usado na inicialização do sistema

2. Config

Configura o D6106

03. Set_params

Seta parâmetros para o D6106

04 Set_user

Seta o usuário ativo para o D6106

05. Get_system_status

Retorna o status do sistema ("cold ou "warm") e o número de palavras pré-definidas

3.7.5 Word Synthesis

O DSP pode prover síntese de voz de alta qualidade.

A função de síntese pode ser usada para:

1. Fazer o playback de comandos pré-gravados na EPROM .

2. Fazer o playback de comandos adquiridos na fase de treinamento e armazenadas na SRAM

Neste caso a subrotina pode ser tão simples quanto se queira:

synt. Sintetiza o comando da SRAM ou EPROM. O indexador é o ponteiro para o comando

3.7.6 Treinamento de Comando

A função de treinamento do servidor (TRAIN) deve instruir o chip DSP a criar um novo modelo baseado na entrada vocal do comando feita pelo usuário durante a "janela de aquisição" aberta pelo chip D6106 no codec.

O servidor também prove o index do modelo, além de informações necessárias para o chip controlar o espaço alocado na SRAM para cada modelo assim como a mínima qualidade aceitável para o sinal.

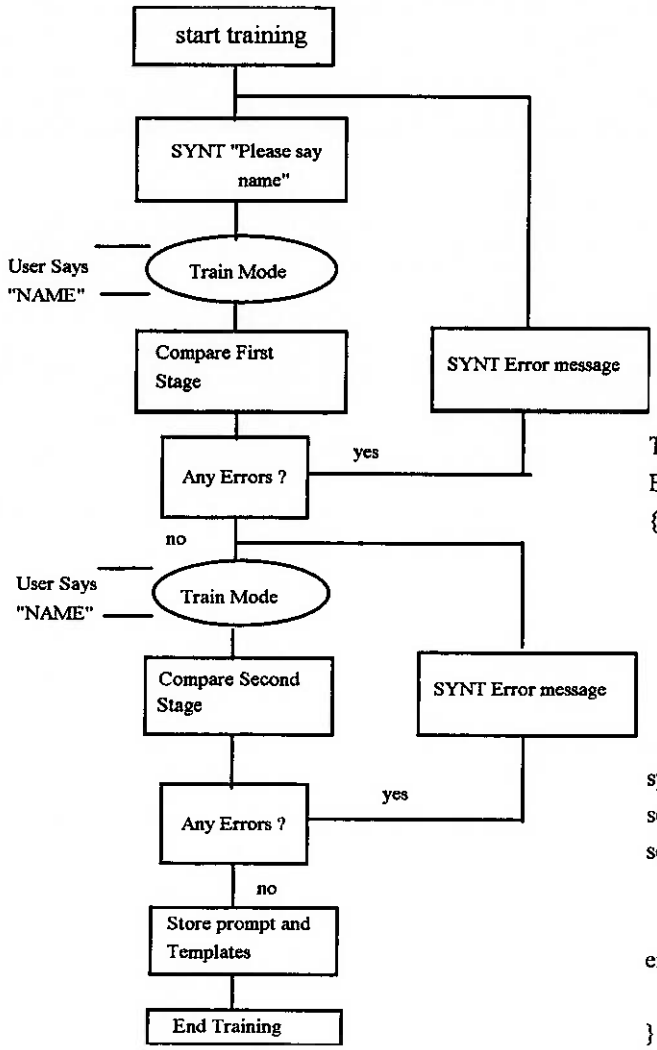
O chip D6106 também pode ser instruído para gerar um versão compactada do prompt se desejado. Prompts compactados são armazenados na SRAM o que significa que eles podem ser reproduzidos quando o comando SYNT é acionado.

A função COMPARE pode ser usada para ativar a comparação do modelo corrente (recém-treinado) aos modelos pré-armazenados. O comando retorna o index do modelo com o melhor nível de similaridade e um código de erro como especificado pelo servidor

Como explicado anteriormente existem tipicamente duas formas de se utilizar a rotina de comparação para checar a qualidade dos modelos.

01. Comparar os dois modelos de um mesmo comando. Neste caso um parâmetro numérico de comparação alto para um dos modelos indica problemas na fase de treinamento. O servidor pode ser programado para alertar o usuário para retreinar o modelo falho;

02. Comparar modelos treinados para diferentes comandos. Neste caso um parâmetro numérico de comparação muito baixo significa um alto nível de similaridade




```
Train_template;
BYTE train_template(BYTE index, BYTE t_stage)
{
      BYTE error_code = 0;
      BYTE compress;
     if (t_stage= =1)
      compress=1;
     else
      compress=0;
synt (P_BEEP,SYS);          /* synthesize BEEP */
send_command((BYTE)(TRAINC|compress<<4));
send_command(index);
     error_code=get_response();
     if error_code) return (error_code);
error_code = send_byte(STOREC);    /* store template */
          return (error_code);
}
```

Fig.19 Treinamento

3.7.7 Reconhecimento de Comandos Vocais

A função RECOG especifica os modelos ativos durante a etapa de reconhecimento e instrui o D6106 a "reconhecer" um comando durante uma "janela de escuta". O resultado do procedimento de reconhecimento pode então ser transferido ao servidor para uma ação final.

O D6106 retorna um conjunto de resultados ao servidor:

1. O modelo pré-treinado que apresenta o maior nível de similaridade com o modelo ativo

2. Os três modelos que apresentam na sua ordem decrescente os maiores índices de similaridade ao modelo ativo.

3. O valor SNR (Signal to Noise Ratio) que é uma relação de significatividade do sinal vocal em relação ao ruído ambiental

4. Um código de erro - no caso de um valor diferente de zero existe uma sinalização de algum tipo de erro durante a fase de reconhecimento

A função recog envia ao D6106 a lista de modelos ativos e lê de volta o status do algoritmo de reconhecimento.

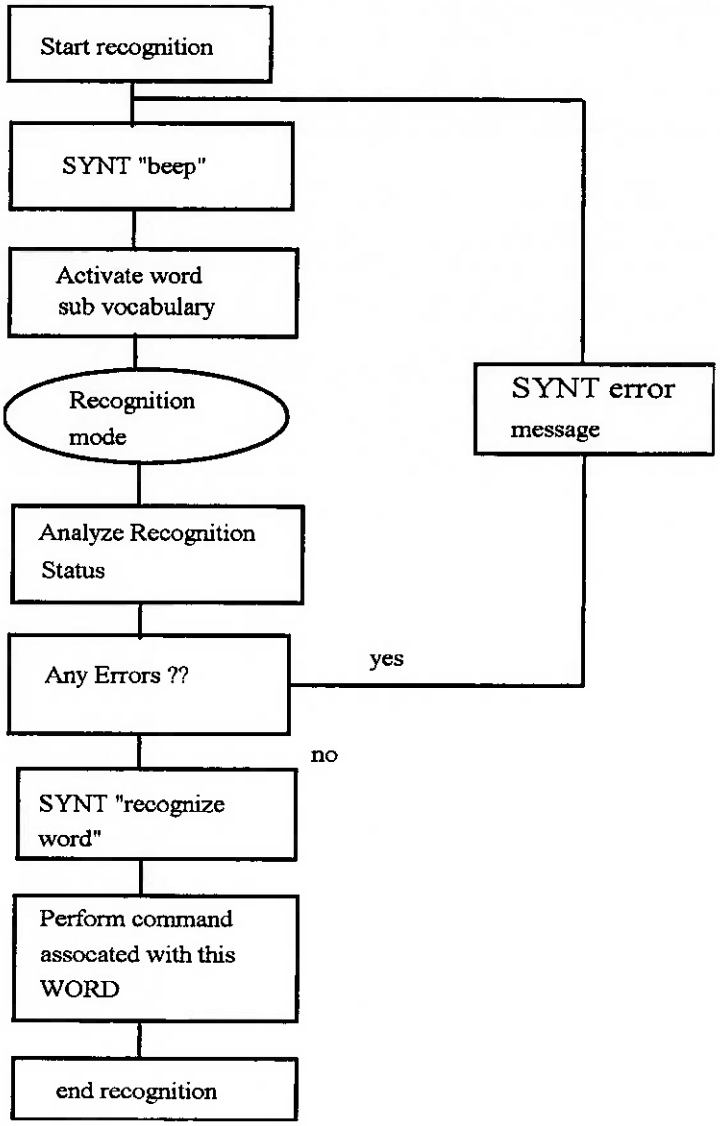


Fig.20 Algoritmo de Reconhecimento

3.8 Interface Gráfica

3.8.1 Introdução

A aplicação desenvolvida para exploração dos recursos do sistema D6106 utiliza tecnologia GUI de interface gráfica iterativa com o usuário.

Além disso utilizou-se de recursos adicionais do Visual Basic viabilizados através de bibliotecas OCX da GMS que podem ser consultados no site (http://www.globalmajic.com) . São interfaces gráficas na forma de gráficos animados xy, botões animados, leds virtuais e mostradores automotivos e aeronáuticos.

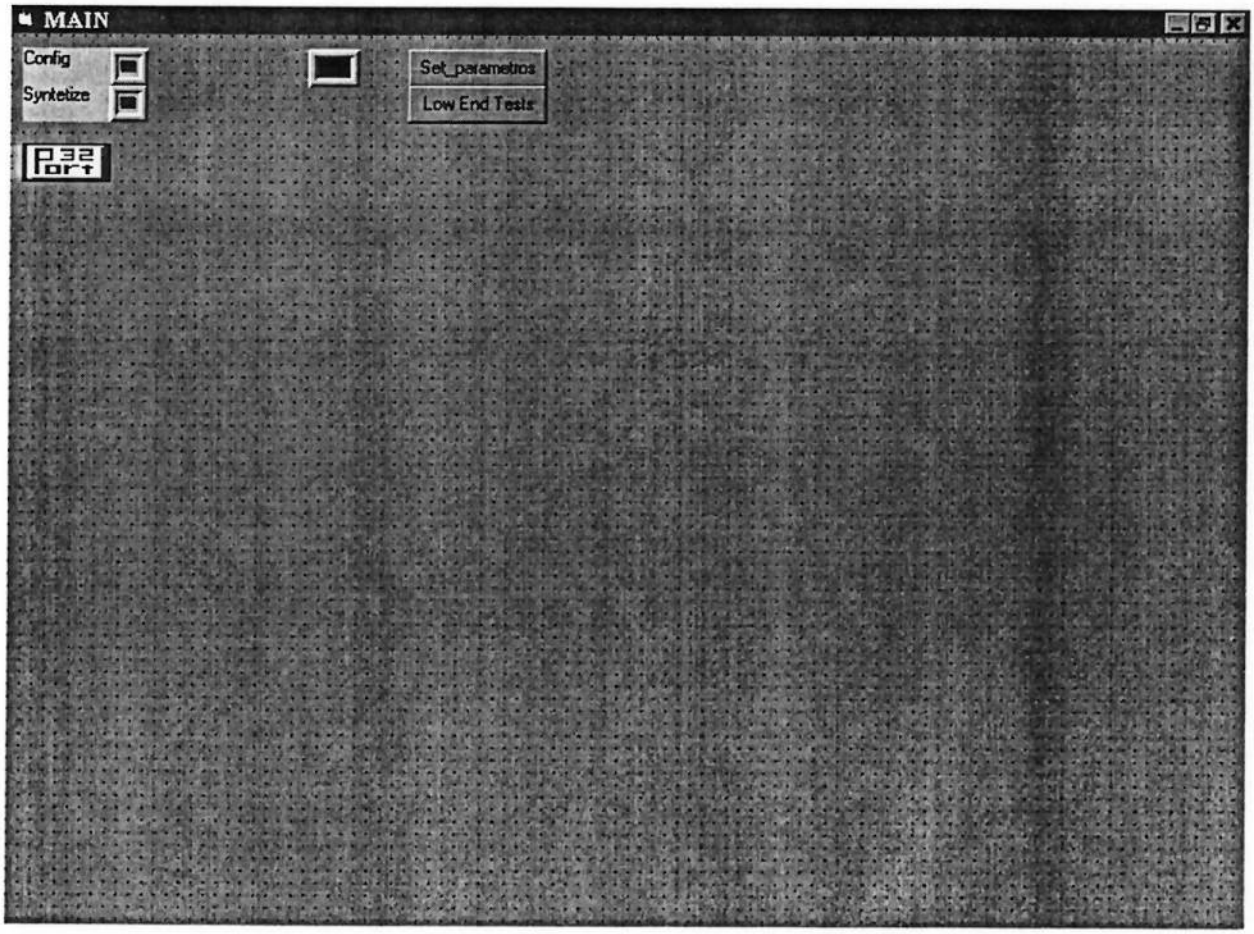


Fig.27 Main screen

3.8.2 Parâmetros de Setup

Ao executarmos o programa aplicativo de interface com o subsistema de comando vocal, o primeiro ponto de checagem será o setup de nosso chip D6106. Para isso clique sobre o ícone set_parametros.

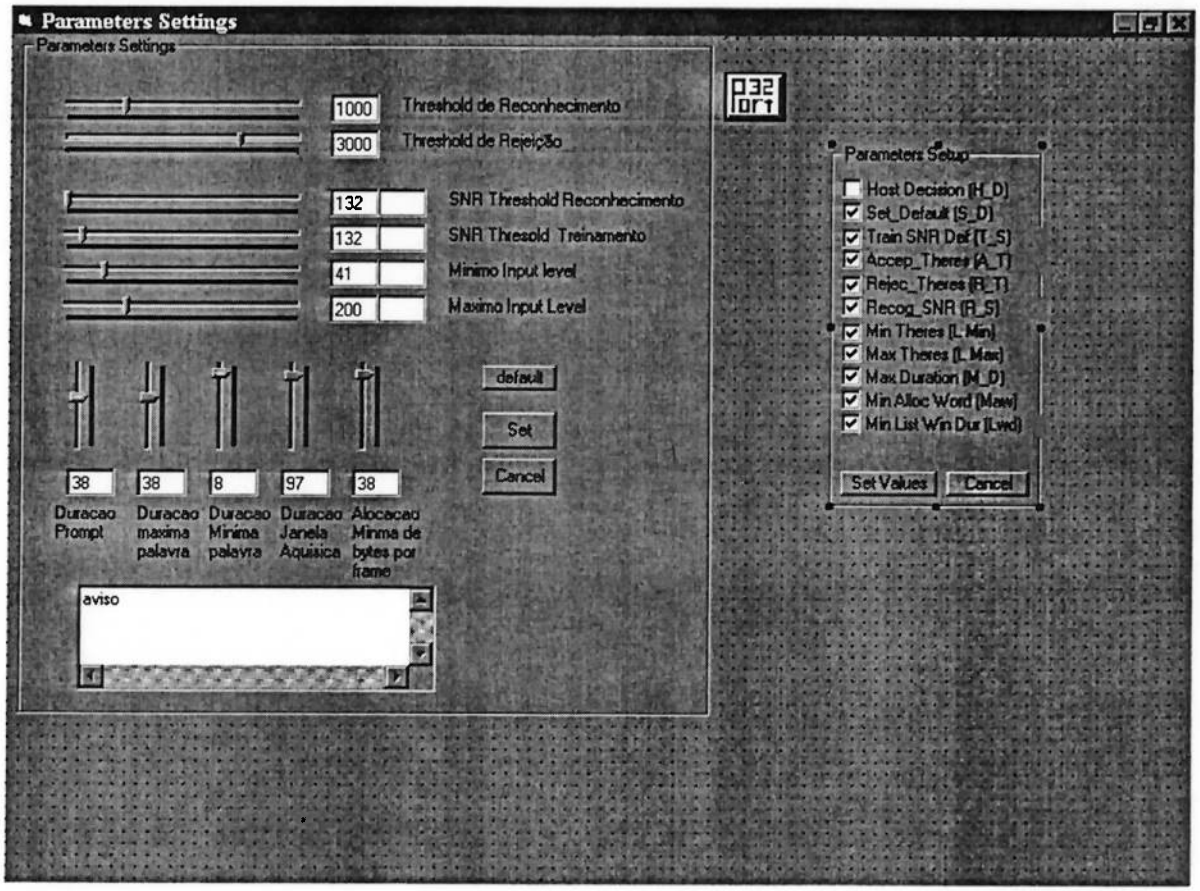


Fig.21 tela de parâmetros

Duração Janela de Aquisição

Duração (em segundos) da janela de tempo na qual o chipset permanece aguardando uma entrada de comando vocal do usuário

Min. Input Level

Mínima Intensidade para que um comando de voz seja considerado aceitável

Max Input Level

Máxima Intensidade para que um comando de voz não sature o CODEC e que portanto seja considerado aceitável

Threshold de Reconhecimento

Valor de corte para o parâmetro numérico de diferenciação abaixo do qual o sistema considera que o comando vocal pode ser reconhecido

Threshold de Rejeição

Valor de corte para o parâmetro numérico de diferenciação acima do qual o sistema rejeita um comando vocal

SNR Threshold Reconhecimento

Valor de SNR abaixo do qual o sistema que o processo de reconhecimento não pode ser executado com segurança

SNR Threshold Treinamento

Valor de SNR abaixo do qual o sistema que o processo de treinamento não pode ser executado com segurança

Duração do Prompt

Duração (em segundos) do prompt do sistema

Alocação Mínima por frame

Unidade de Alocação (em bytes) por frame

Default

O sistema carrega os valores de variáveis consideradas ótimas através de processo empírico

Set

O host envia ao D6106 os parâmetros selecionados pelo usuário

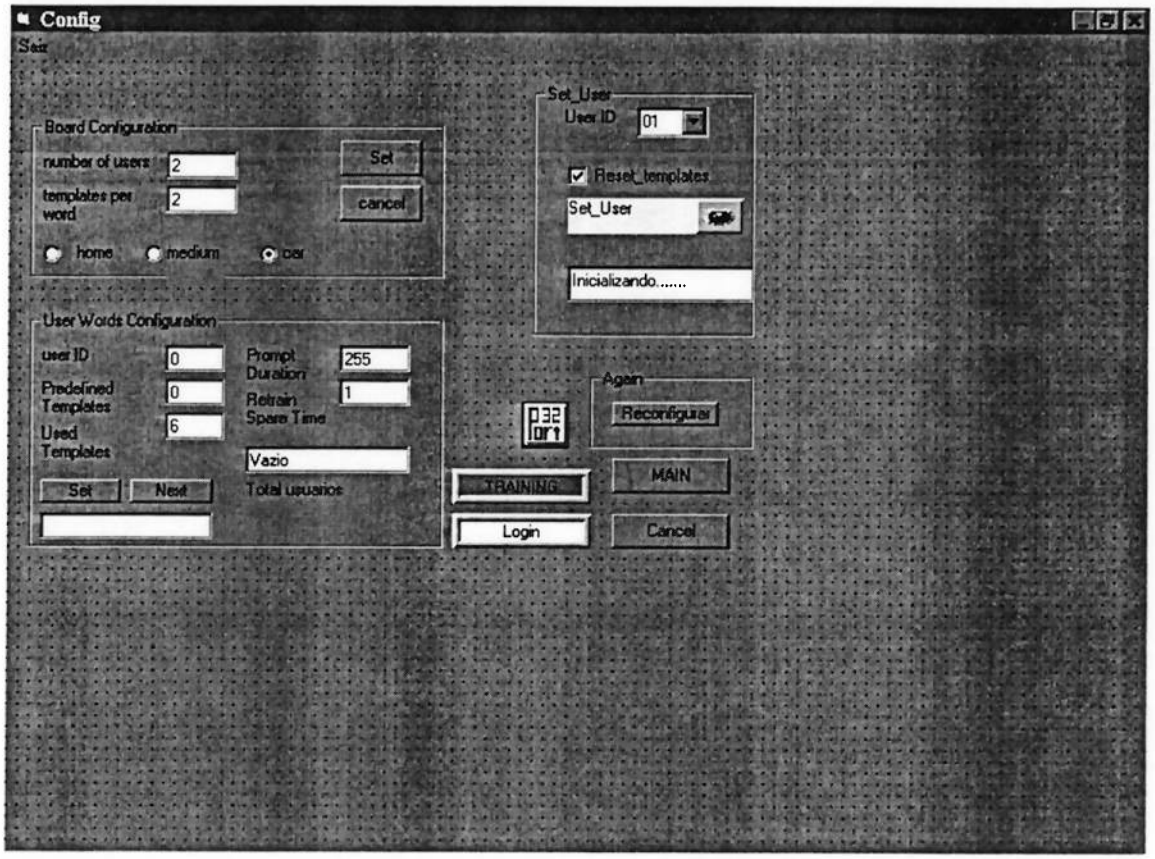
Cancel

O processo de setagem do D6106 é cancelado

3.8.3 Config

Após a setagem dos parâmetros operacionais do sistema o usuário deve configurar dados operacionais do sistema, tais como número de comandos vocais a serem ativados assim como o número de usuários ativos

Fig 22 Tela de Configuração



3.8.4 Board Configuration

Selecione o número total de usuários para o sistema assim como o número de modelos por comando vocal

Além disso selecione o ambiente de operação do sistema:

- Home
- Medium
- Car

User Words Configuration

Preencha o número de modelos pré-definidos, número de modelos utilizados, a duração do prompt e o retrain spare time. Ao terminar pressione next e repita a operação até que tenha sido completado o total de usuários.

Set User

Selecione o usuário ativo

Train/Login
Selecione o tipo de procedimento que desejar. Login ou Treinamento

3.8.5 Modo de Treinamento

Neste modo o usuário poderá fazer o processo de treinamento dos comandos vocais armazenados no banco de dados. Caso se deseje modificar o comando vocal, ou então a sequência de bits digitais (08 bits) correspondente a este comando tecle DATABASE no menu principal.

Caso deseje reconfigurar parâmetros da placa pressione config no menu principal.

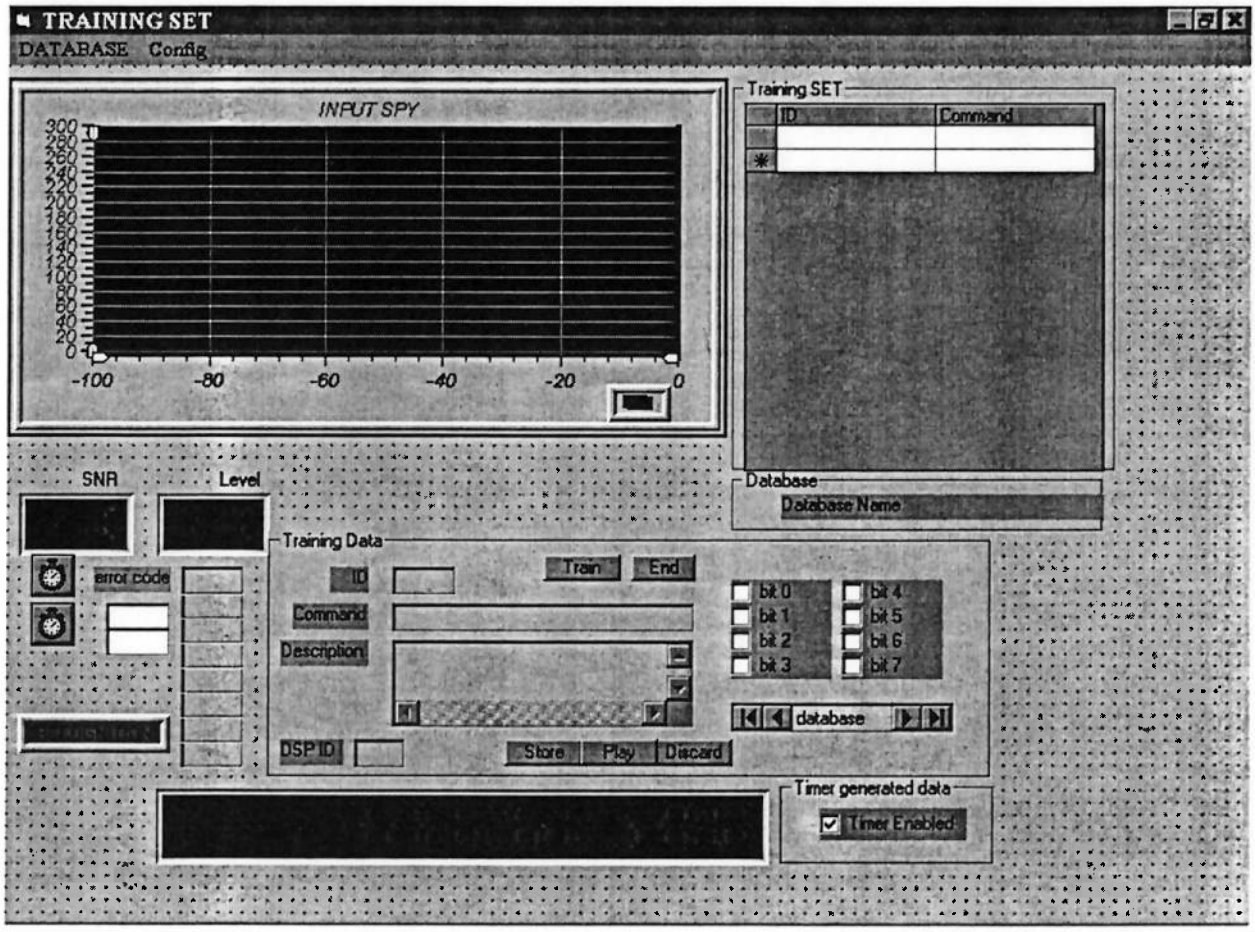


Fig.23 tela de treinamento

input spy
Monitoramento XY animado da intensidade sonora na entrada do CODEC.

training set
Apresenta a lista de comandos vocais disponível no banco de dados do sistema

database name
Apresenta o display do nome do arquivo de banco de dados utilizado

Snr/Level
Indicam num interessante display de leds vermelhos a intensidade média do comando vocal assim como a relação sinal ruído do processo de treinamento do modelo ativo

Error Code
Indica o código binário de oito bits do código de erro retornado pelo D6106

Training Data
Apresenta o Comando de Seleção de Comandos Vocais do Banco de Dados

Recognition
O usuário deve pressionar esta tecla quando desejar iniciar a etapa de reconhecimento de comandos vocais

Store
O usuário deve ativar esta opção quando considera que o modelo treinado é valido e que pode ser usado no processo de reconhecimento

Play
Acionando esta tecla, o usuário terá a sua disposição o playback do modelo treinado. Deve ser usado quando o usuário além dos dados de intensidade e SNR deseja ter uma análise auditiva da qualidade do modelo

Discard
O usuário deseja adquirir um novo modelo para determinado comando vocal e deseja descartar o modelo atual

3.8.6 Database editor

Edição do banco de dados do sistema

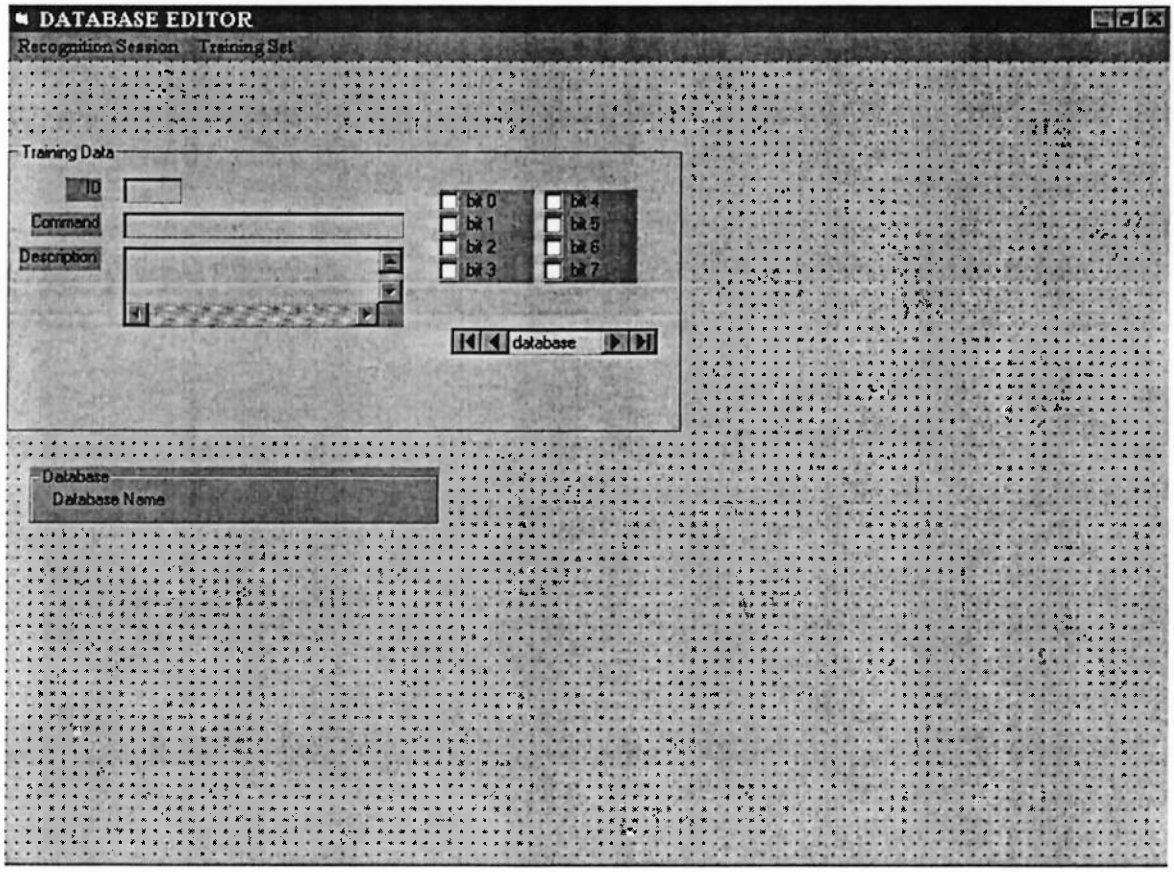
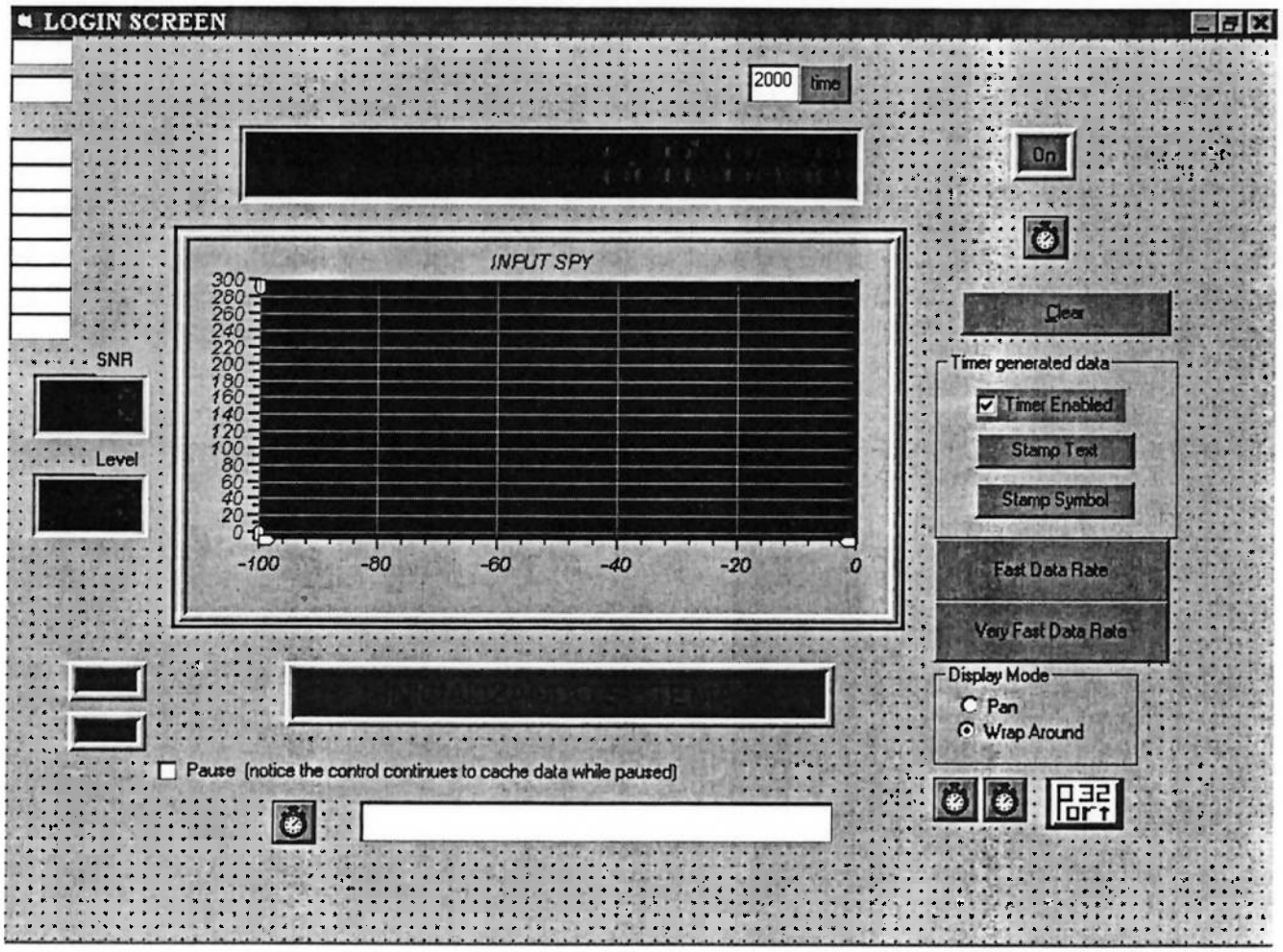


Fig 24. Editor de banco de dados

3.8.7 Login Interface (Interface de Ativação do sistema)

Na tela de login o usuário pode sentir a potencialidade do D6106 de sensoriar o nível de intensidade e de SNR do comando vocal e proporcionar ao usuário um procedimento de auto-ajuste do ganho do amplificador do sinal sonoro, assim como monitorar o nível de ruído ambiental presente.

Fig.25 Tela de Inicialização do Sistema



3.8.8 Reconhecimento de Comando Vocal

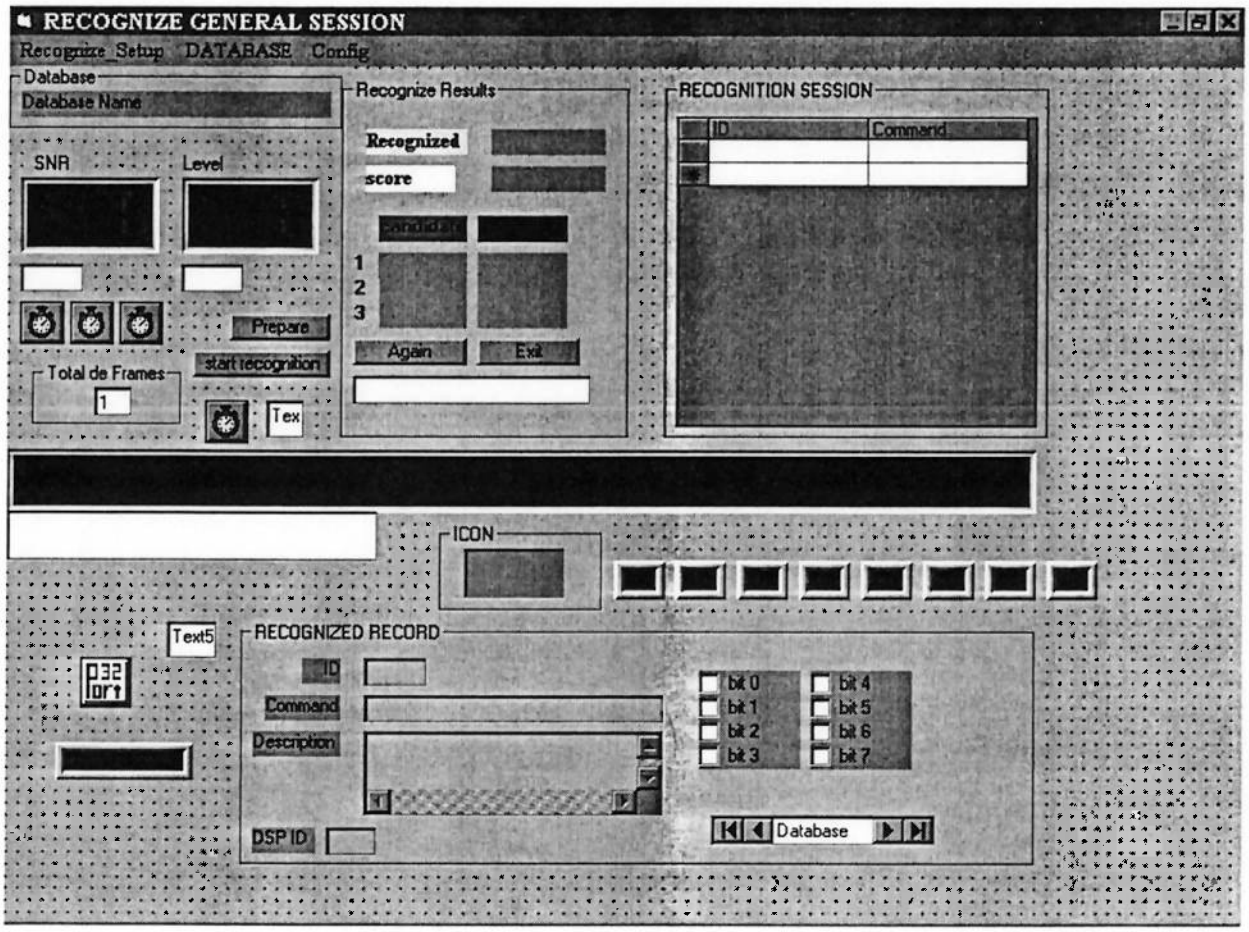


Fig. 26 Tela de reconhecimento de comandos vocais

3.8.9 Menu Principal

database command

Ativa a sessão de edição do banco de dados de comandos vocais

config command

Ativa a sessão de config do D6106

recognize setup

Ativa a sessão de setup da fase de reconhecimento

Displays da Tela

SNR/LEVEL
Apresentam o display da intensidade média do comando vocal e da relação sinal ruído do ambiente

Total de frames
Número total de frames ativos para o reconhecimento

Recognized Record
Display dos detalhes do comando ativado vocalmente

Leds 08 bits
Indicação Visual do ativamento de um protocolo de 08 bits através do comando vocal

Recognition Session
Display de todos os comandos vocais ativos

Database Name
Nome do banco de dados Ativo para a fase de reconhecimento

Start Recognition
Abre janela de tempo para aquisição e reconhecimento de comando vocal

Recognition Results

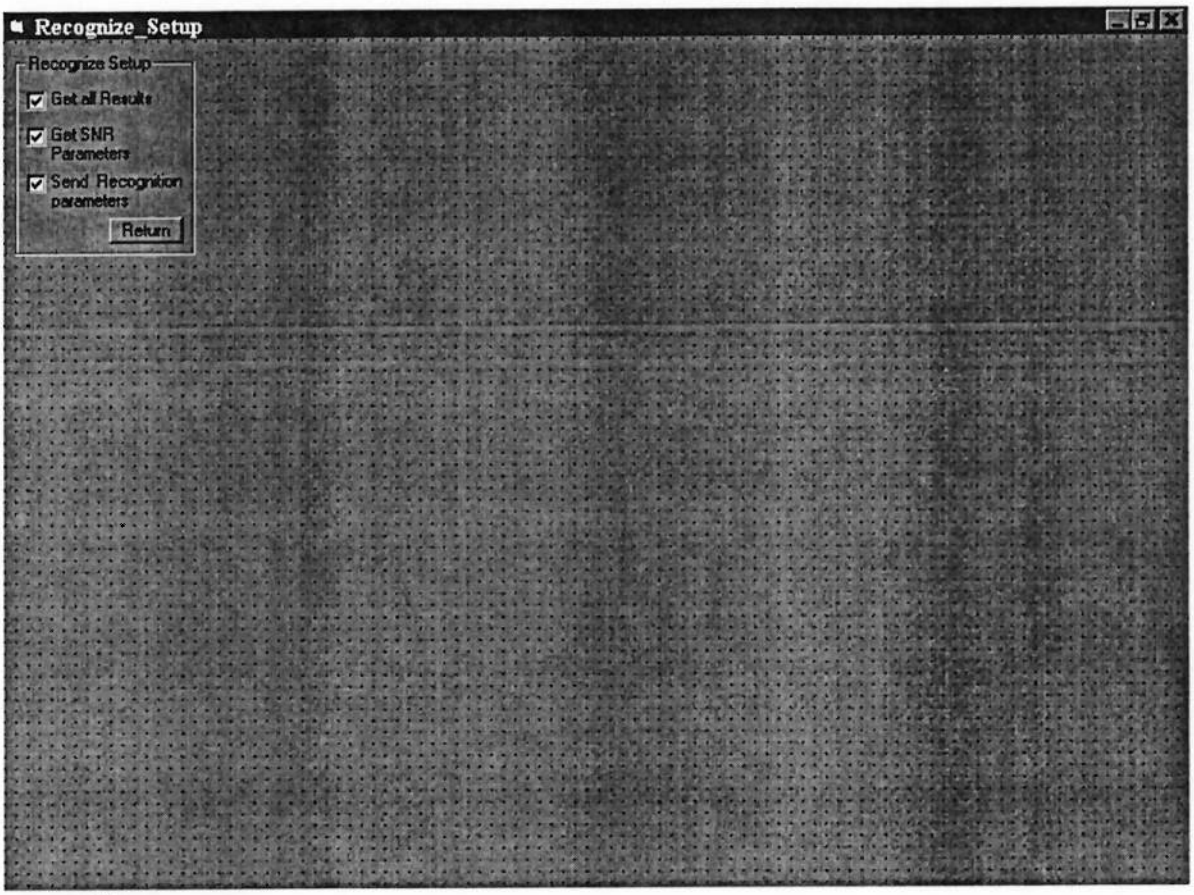
Proporciona o display do modelo reconhecido com o mais similar ao comando vocal adquirido e o ranking dos 3 modelos mais similares ao modelo selecionado

Prepare

Inicializa a janela de resultados , SNR/Intensidade para novo reconhecimento

3.8.10 Setup da fase de reconhecimento

Fig.27 tela de setup



Get All Commands

Caso o usuário selecione esta opção o D6106 envia ao host a informação do comando vocal com maior nível de similaridade ao comando vocal presente assim como o ranking dos outros três modelos mais similares

Get SNR parameters

Envia os dados de intensidade e SNR para a fase de reconhecimento

Send Recognition Results

O D6106 envia além do ID dos modelos o score do processo de reconhecimento do comando vocal

3.8.11 SINTETIZAÇÃO DE PROMPTS

O D6106 pode sintetizar modelos pré-gravados na memória EPROM ou na SRAM

Index

Selecione o ID do prompt desejado

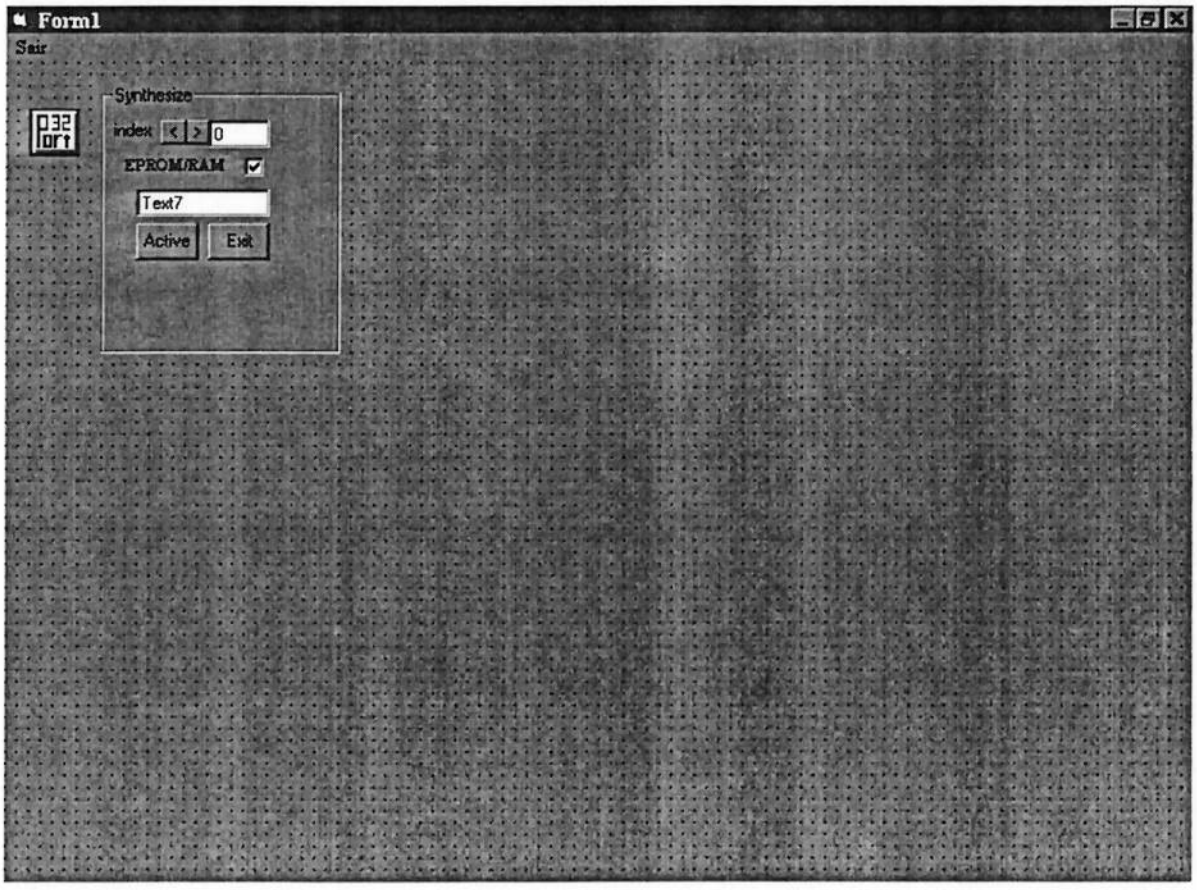
EPROM/RAM

Selecione se deseja sintetizar a EPROM ou a SRAM

Active

Inicializa processo de síntese

Fig.28 Tela de Prompt



3.8.12 Tela low end tests

Executa testes de comunicação Básicos com o D6106

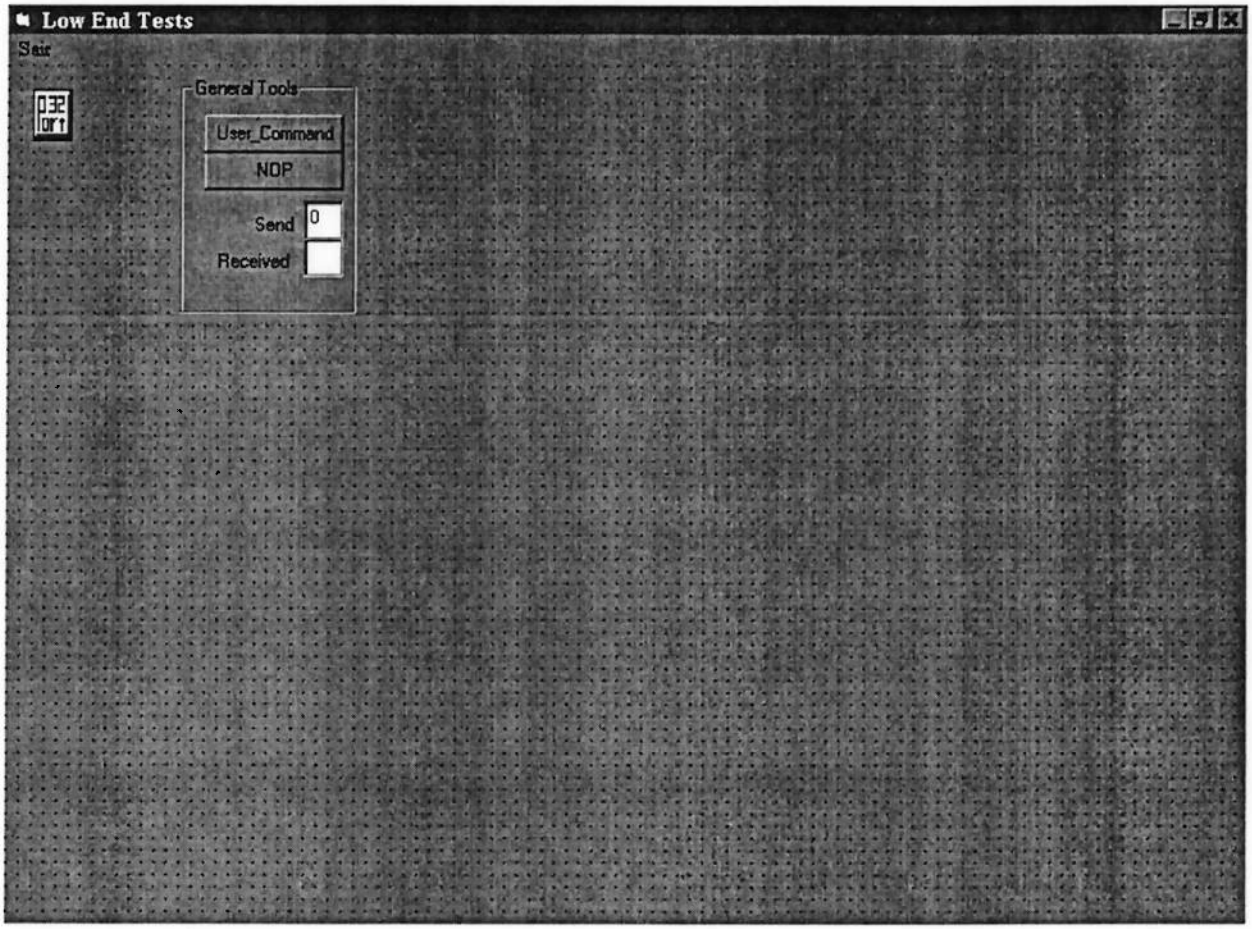


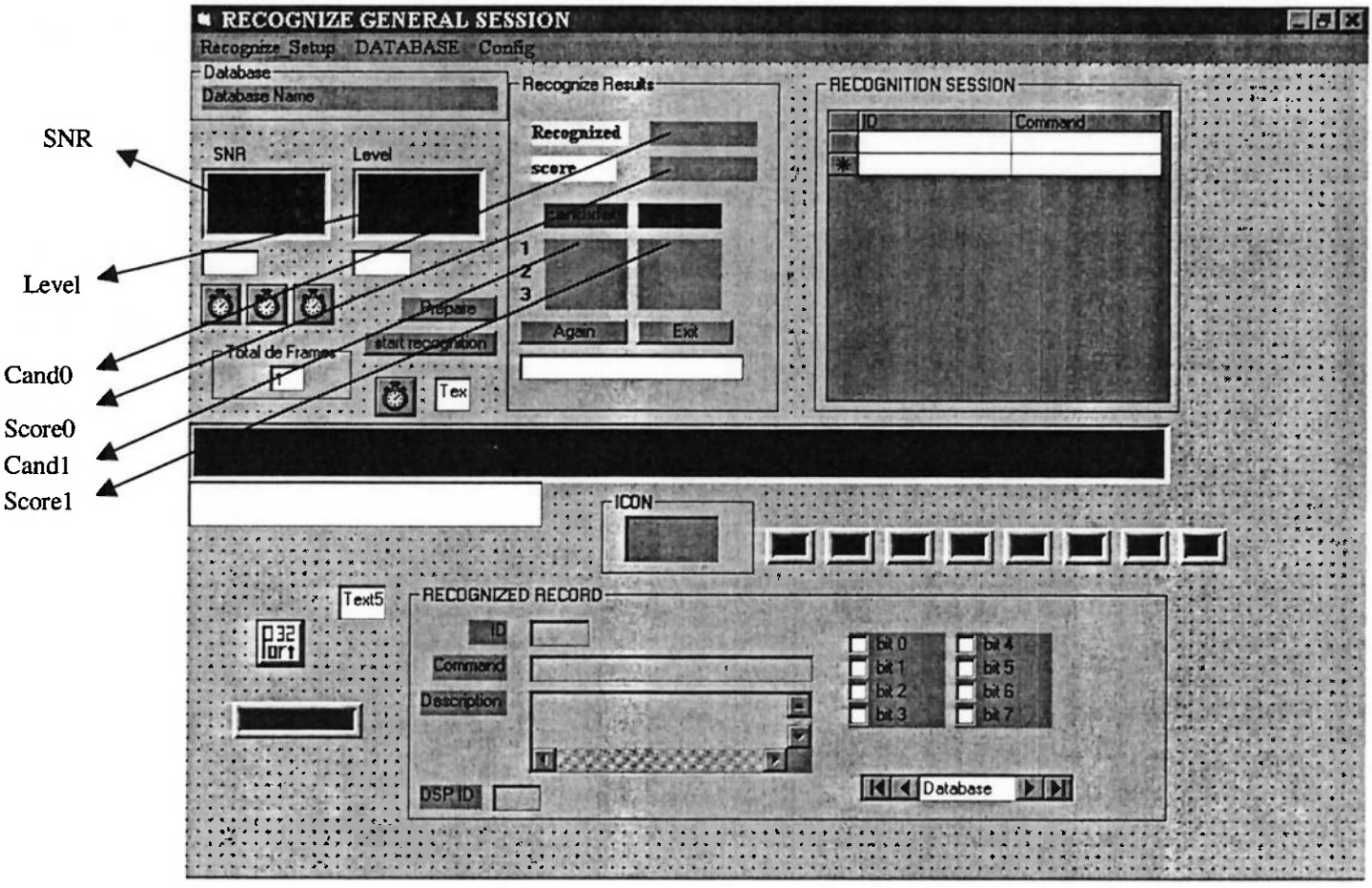
Fig.29 Tela de Testes Básicos do do Sistema

4 Resultados

Para o sistema desenvolvido, foram executados testes com diferentes conjuntos de comandos vocais sob diversas condições de operação.

Parâmetros medidos

Para o teste de performance do sistema de reconhecimento de comando vocal foram utilizados os seguintes parâmetros: SNR (relação Sinal Ruído), Level (Nível), score1, cand1, score2, cand2, score3, cand3, score4, cand4, dif1, dif2 dif3, dif4, tempo de resposta.



Snr/Level

Level (Nível) indica a intensidade média do comando vocal em decibéis(dB); e SNR a relação sinal x ruído do processo.

Cand(x), Score(x), Dif(x)

Cand(x) Variando-se x de 1 até 4, indica a ordem de similaridade decrescente dos comandos vocais tratados na fase de treinamento ao comando vocal em análise.

Score(x) Indica a distância euclidiana entre o comando vocal analisado e os comandos vocais armazenados na fase de treinamento. Quanto menor o score(x), maior é a similaridade entre o comando vocal analisado e os comandos vocais armazenados.

Dif(x) é a diferença aritmética entre o score do candidato x + 1 e o score do candidato x, onde x é menor que o número total de frames treinados.

Tempo de Resposta

É o tempo, em segundos, que o sistema precisa para analisar um comando vocal e retornar todos os valores comparativos necessários para deteção daquele com o maior nível de similaridade.

Conjuntos de Comandos Vocais Testados

Para teste do sistema implementado, foram escolhidos sete (07) conjuntos de palavras com diferentes números de sílabas e diferentes níveis de similaridade fonética.

Conjunto 1

- luz
- som
- ar
- TV
- dormir
- timer

Conjunto 2

- um
- dois
- três
- quatro
- cinco

Conjunto 3 (Agenda de Nomes)

- Ana
- Carlos
- Claudio
- Luis
- Marcelo
- Roberto
- Adalberto
- Julio
- Rulio
- Silvia
- Zelia

Conjunto 4

- Casa
- Caça
- Capa
- Cama
- Cala
- Cata
- Caca
- Lata
- Coca

Conjunto 5

- Beijo
- Queijo
- Queixo
- Eixo
- Gueixa
- Brejo
- Tejo

Conjunto 6

- Tomate
- Boate
- Malote
- Mascate
- Mascote

Conjunto 7

- Triciclo
- Biociclo
- Quadriciclo
- Monociclo
- Pentaciclo
- Hexaciclo

4.1 Testes

4.1.1 Conjuntos 1 e 2

Utilizou-se para o teste dos conjuntos de comandos vocais 1 e 2 os seguintes recursos:

- 03 diferentes usuários de modo a checar a acuidade do sistema
- Ambiente Silencioso (Home) e Ambiente Ruidoso (Car).
- Lista dos comandos vocais com nível decrescente de similaridade.
- Diferença aritmética do nível de similaridade entre o primeiro e segundo candidatos na lista (dif 1) entre o segundo e terceiro candidatos (dif2) e entre o terceiro e quarto candidatos (dif3).

Tabela 5: Resultados dos Testes com Conjunto 1

Teste	14.10.98	0	luz	Acuidade Aferida	100%		Treinamento				
		1	som	Acuidade Nominal	97%				comando	SNR	Level
		2	ar					0	luz	248	55
		3	TV					1	som	237	53
		4	dormir					2	ar	237	51
		5	timer					3	TV	245	54
								4	dormir	246	54
								5	timer	237	50

SNR	Level	Detetado	Score	2nd cand	3rd cand	4th cand	dif1	ruído	time	user	result
230	46	ar	1125	timer	TV	dormir	3008	silêncio	6,3	claudio	OK
231	48	ar	1084	dormir	timer	TV	2198	silêncio	6,2	claudio	OK
204	47	ar	1447	timer	TV	dormir	2264	silêncio	6,1	claudio	OK
228	50	ar	1461	TV	timer	dormir	1756	silêncio	6	claudio	OK
231	48	ar	1114	timer	dormir	TV	2719	silêncio	6,1	claudio	OK
245	51	dormir	713	timer	dormir	luz	1022	silêncio	6,2	claudio	OK
248	52	dormir	728	timer	dormir	luz	891	silêncio	6,1	claudio	OK
248	52	dormir	914	timer	dormir	luz	887	silêncio	6,3	claudio	OK
243	51	dormir	809	timer	dormir	luz	997	silêncio	6,2	claudio	OK
240	51	dormir	847	timer	dormir	luz	944	silêncio	6,1	claudio	OK
255	55	luz	706	ar	dormir	TV	1664	silêncio	6	claudio	OK
251	55	luz	766	dormir	TV	ar	1402	silêncio	6,3	claudio	OK
248	52	luz	720	TV	dormir	ar	1365	silêncio	6,04	claudio	OK
243	51	luz	1212	ar	TV	dormir	830	silêncio	6,02	claudio	OK
245	52	luz	1319	TV	dormir	ar	813	silêncio	6	claudio	OK
242	50	som	990	dormir	TV	timer	2385	silêncio	6,12	claudio	OK
233	49	som	963	dormir	TV	timer	2158	silêncio	6,14	claudio	OK
233	48	som	629	dormir	TV	timer	3508	silêncio	6,16	claudio	OK
234	49	som	672	dormir	TV	timer	2397	silêncio	6,17	claudio	OK
237	49	som	919	dormir	TV	timer	1517	silêncio	6,19	claudio	OK
246	51	timer	848	TV	timer	luz	1065	silêncio	6,21	claudio	OK
242	50	timer	780	timer	TV	luz	1153	silêncio	6,22	claudio	OK
224	49	timer	624	timer	TV	luz	1384	silêncio	6,31	claudio	OK
233	48	timer	1003	timer	TV	luz	871	silêncio	6,21	claudio	OK
228	48	timer	643	timer	TV	luz	1091	silêncio	6,3	claudio	OK
248	51	tv	705	timer	dormir	luz	1488	silêncio	6,25	claudio	OK
246	51	tv	882	timer	dormir	luz	1081	silêncio	6,22	claudio	OK
249	51	tv	739	timer	luz	dormir	1779	silêncio	6,23	claudio	OK
240	51	tv	612	timer	dormir	luz	1776	silêncio	6,24	claudio	OK
240	51	tv	702	timer	luz	dormir	1721	silêncio	6,22	claudio	OK
									6,171667	Media	

Tabela 6: Resultados dos Testes com o Conjunto 2

Teste	14.10.98			Acuidade Aferida	99%		Treinamento				
		1	Um	Acuidade Nominal	97%			comando	SNR	Level	
		2	Dois					um	255	54	
		3	Tres					dois	254	55	
		4	Quatro					tres	255	53	
		5	Cinco					quatro	245	52	
								cinco	255	55	

SNR	Level	Detetado	Score	2nd cand	3rd cand	4th cand	Dif1	Ruído	time	user	result
255	50	tres	390	cinco	quatro	dois	810	Silêncio	6,1	Silvia	OK
245	49	tres	652	dois	quatro	dois	577	Silêncio	6	Mauricio	OK
255	47	tres	735	cinco	um	dois	227	Silêncio	6,2	Mauricio	OK
254	48	tres	695	cinco	quatro	dois	731	Silêncio	6,4	Mauricio	OK
244	49	tres	598	cinco	quatro	um	391	Silêncio	6,2	Mauricio	OK
245	51	tres	565	cinco	quatro	dois	468	Silêncio	6,1	Mauricio	OK
247	47	quatro	832	dois	tres	cinco	515	Silêncio	6,3	Claudio	OK
250	49	quatro	869	dois	tres	cinco	53	Silêncio	6,21	Claudio	OK
251	52	quatro	741	dois	tres	cinco	226	Silêncio	6,24	Claudio	OK
252	53	quatro	658	dois	tres	cinco	733	Silêncio	6,4	Claudio	OK
253	51	quatro	814	dois	tres	cinco	151	Silêncio	6,45	Claudio	OK
245	50	quatro	604	cinco	tres	dois	2122	Silêncio	6,44	Silvia	OK
247	53	quatro	598	cinco	tres	dois	2052	Silêncio	6,01	Silvia	OK
253	52	quatro	766	cinco	tres	dois	1874	Silêncio	6,04	Silvia	OK
255	51	quatro	713	dois	tres	um	1851	Silêncio	6,3	Silvia	OK
253	55	quatro	667	cinco	tres	um	1806	Silêncio	6,2	Silvia	OK
252	53	quatro	837	dois	tres	um	1163	Silêncio	6,4	Mauricio	OK
251	52	quatro	670	dois	tres	um	962	Silêncio	6,1	Mauricio	OK
252	54	quatro	804	dois	tres	um	639	Silêncio	6,2	Mauricio	OK
255	51	quatro	621	dois	tres	um	877	Silêncio	6,3	Mauricio	OK
255	52	quatro	693	dois	tres	um	907	Silêncio	6,2	Mauricio	OK
245	53	cinco	1512	tres	dois	quatro	-605	Silêncio	6,1	Claudio	OK
247	54	cinco	728	tres	dois	quatro	881	Silêncio	6,3	Claudio	OK
255	55	cinco	574	tres	dois	quatro	1700	Silêncio	6,4	Claudio	OK
245	51	cinco	553	tres	dois	quatro	1842	Silêncio	6,2	Claudio	OK
253	52	cinco	796	um	dois	quatro	1222	Silêncio	6,1	Claudio	OK
252	53	cinco	639	tres	dois	quatro	1270	Silêncio	6,4	Silvia	OK
251	54	cinco	738	tres	dois	quatro	922	Silêncio	6,32	Silvia	OK
255	55	cinco	767	tres	dois	quatro	494	Silêncio	6,33	Silvia	OK
251	52	cinco	737	tres	dois	quatro	1041	Silêncio	6,34	Silvia	OK
249	51	cinco	717	tres	dois	quatro	1093	Silêncio	6,32	Silvia	OK
248	52	cinco	613	um	tres	dois	246	Silêncio	6,12	Mauricio	OK
247	54	cinco	595	um	tres	dois	314	Silêncio	6,23	Mauricio	OK
255	53	cinco	553	um	tres	dois	466	Silêncio	6,25	Mauricio	OK
251	52	cinco	655	um	dois	tres	116	Silêncio	6,26	Mauricio	OK
237	51	tres	632	cinco	dois	um	131	Silêncio	6,21	Mauricio	Erro
									6,240833	Media	

4.1.2 Conjunto 3

Neste ensaio, a intenção foi simular a situação de um telefone inteligente com discagem de nomes pré – agendados através de comando vocal, sob diferentes tipos de ruído ambiente. Os ruídos ambeinte abaixo descritos (pop, sfx, symphony) foram gerados através do MIDI Player da Creative labs ®.

Abaixo estão descritos os espectros de frequência médios dos diversos tipos de ruídos ambientais, simulador no MIDI Player :

	63Hz	160Hz	400Hz	1Khz	2,5Khz	6,3Khz	16Khz
Pop	7	10	8	4	2	1	1
Sfx	5	8	10	7	4	2	1
Symphony	10	8	7	3	2	1	1

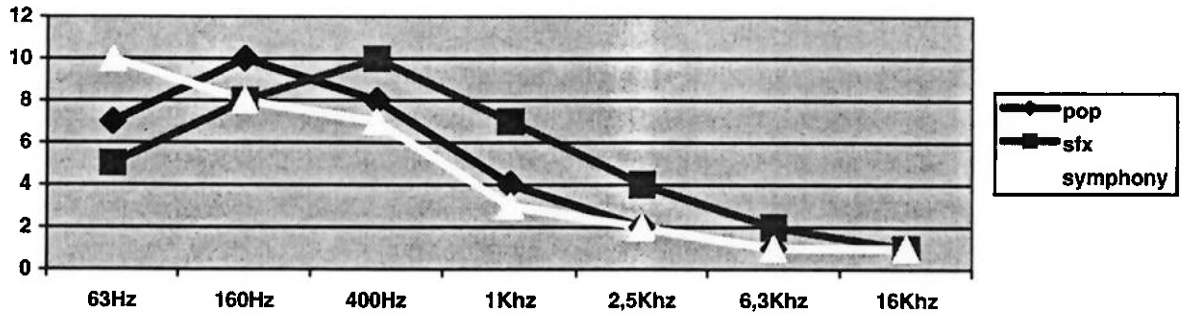


Tabela 7 – Ruídos Gerados no MIDI Player

O sistema, no resultado do reconhecimento do comando vocal, também oferece a lista de comparação dos demais candidatos em relação ao comando vocal em análise.

Foram mensurados os valores de SNR e Level dos sinais para a análise da qualidade dos comandos vocais a serem reconhecidos.

Calculou-se também, a diferença de similaridade entre o primeiro e segundo candidatos da lista do processo de reconhecimento (dif 1)

E finalmente, foi mensurada a relação existente entre o tempo de resposta do sistema e o nível de sensibilidade do sistema sob diferentes tipos de ruído ambiental (sfx, pop, symphony)

Teste	31.3.99	0	ana	Acuracidade Aferida	100%
		1	carlos	Acuracidade Nominal	97%
		2	claudio		
		3	luis		
		4	marcelo		
		5	roberto		
		6	adalberto		
		7	julio		
		8	rulio		
		9	silvia		
		10	zelia		

Treinamento

	comando	SNR	Level
0	ana	255	60
1	carlos	255	61
2	claudio	255	60
3	luis	255	68
4	marcelo	255	59
5	roberto	255	60
	adalberto	255	63
	julio	255	62
	rulio	255	65
	silvia	255	62
	zelia	255	67

SNR	Level	Detetado	Score	2nd cand	:	3rd cand	4th cand	dif1	ruido	time	Result	User
255	57	ana	778	zelia		rulio	carlos	2237	silencio	6,3	OK	Claudio
255	68	carlos	1361	ana		claudio	rulio	2430	silencio	6,32	OK	Claudio
255	58	claudio	928	rulio		adalberto	roberto	2330	silencio	6,2	OK	Claudio
254	58	luis	1461	rulio		zelia	julio	1018	silencio	6	OK	Claudio
255	54	marcelo	1114	roberto		ana	zelia	2090	silencio	6,26	OK	Claudio
255	58	roberto	713	adalberto		claudio	rulio	1383	silencio	5,87	OK	Claudio
246	55	adalberto	728	roberto		marcelo	claudio	594	silencio	6,22	OK	Claudio
255	57	julio	914	luis		silvia	rulio	1723	silencio	5,88	OK	Claudio
255	61	rulio	809	roberto		ana	claudio	2288	silencio	6,04	OK	Claudio
255	57	silvia	847	julio		zelia	luis	1231	silencio	6,04	OK	Claudio
255	58	zelia	706	ana		julio	silvia	2132	silencio	5,92	OK	Claudio
										6,0955	Media	

MODO CAR

SNR	Level	Detetado	Score	2nd cand	:	3rd cand	4th cand	dif1	ruido	time	Result	User
255	64	julio	896	silvia		ana	rulio	1765	silencio	6,32	OK	Claudio
255	66	rulio	1999	ana		luis	roberto	591	silencio	5,98	OK	Claudio
255	65	rulio	1650	ana		marcelo	adalberto	630	silencio	5,92	OK	Claudio
254	65	julio	1010	ana		roberto	rulio	1685	silencio	6,1	OK	Claudio
255	60	milho	3372	silvia		zelia	rulio	198	silencio	6,09	OK	Claudio
255	60	milho	2992	julio		julio	rulio	247	silencio	5,98	OK	Claudio
246	64	zilio	2466	silvia		silvia	ana	1187	silencio	6,1	OK	Claudio
255	65	julis	1483	ana		zelia	silvia	1686	silencio	6,21	OK	Claudio
255	63	rulia	2115	roberto		ana	claudio	14	silencio	6,05	OK	Claudio
										6,84375	Media	

MODO CAR

SNR	Level	Detetado	Score	2nd cand	:	3rd cand	4th cand	dif1	ruido	time	Result	Usuário
255	64	julio	896	silvia		ana	rulio	1765	silencio	6,32	OK	Claudio
255	66	rulio	1999	ana		luis	roberto	591	silencio	5,98	OK	Claudio
255	65	rulio	1650	ana		marcelo	adalberto	630	silencio	5,92	OK	Claudio
254	65	julio	1010	ana		roberto	rulio	1685	silencio	6,1	OK	Claudio
255	60	milho	3372	silvia		zelia	rulio	198	silencio	6,09	OK	Claudio
255	60	milho	2992	julio		julio	rulio	247	silencio	5,98	OK	Claudio
246	64	zilio	2466	silvia		silvia	ana	1187	silencio	6,1	OK	Claudio
255	65	julis	1483	ana		zelia	silvia	1686	silencio	6,21	OK	Claudio
255	63	rulia	2115	roberto		ana	claudio	14	silencio	6,05	OK	Claudio
										6,84375	Media	

Tabela 8: Resultados dosTestes com o conjunto 3

MODO CAR

SNR	Level	Detetado	Score	2nd cand	:	3rd cand	4th cand	dif1	ruido	time	Result	User
255	64	ana	1169	marcelo		rulio	zelia	3137	sfx	5,5	OK	Claudio
255	66	carlos	1445	ana		claudio	adalberto	2576	afx	5,39	OK	Claudio
255	64	claudio	928	rulio		adalberto	julio	2463	afx	5	OK	Claudio
255	65	luis	2559	zelia		rulio	ana	729	afx	4,83	OK	Claudio
255	60	marcelo	1488	ana		roberto	zelia	1579	pop	5,05	OK	Claudio
255	60	roberto	2575	ana		julio	marcelo	351	pop	5,6	OK	Claudio
245	65	adalberto	2835	roberto		marcelo	zelia	230	pop	5,34	OK	Claudio
255	64	adalberto	1631	roberto		marcelo	zelia	1537	sfx	5,5	OK	Claudio
255	66	roberto	1605	zelia		adalberto	ana	1327	sfx	4,67	OK	Claudio
255	65	julio	1388	ana		zelia	silvia	1821	Symphony	4,95	OK	Claudio
255	66	rulio	2299	ana		julio	zelia	761	symphony	4,78	OK	Claudio
255	65	silvia	1225	julio		zelia	ana	1780	symphony	4,89	OK	Claudio
255	63	zelia	1024	ana		julio	silvia	2486	symphony	4,89	OK	Claudio
										5,106923	Media	

Tabela 8: Resultados dos Testes com o Conjunto 3

4.1.3 Conjunto 4, 5 e 6

Utilizou-se para o teste dos conjuntos de comandos vocais 4, 5 e 6 os seguintes recursos:

- Ambiente Silencioso (Home)
- Lista dos comandos vocais com nível decrescente de similaridade.
- Diferença aritmética do nível de similaridade entre o primeiro e segundo candidatos na lista (dif 1)

Tabela 9: Resultados dos testes com o conjunto 4

Teste	31.3.99	0	casa	Acuracidade Aferida	100%
		1	caça	Acuracidade Nominal	97%
		2	capa		
		3	cama		
		4	cala		
		5	cata		
		6	caca		
		7	lata		
		8	coca		

Treinamento

	comando	SNR	Level
0	casa	249	61
1	caça	255	62
2	capa	255	60
3	cama	255	63
4	cala	252	63
5	cata	255	60
6	caca	255	60
7	lata	255	60
8	coca	255	63

SNR	Level	Detetado	Score	2nd cand	:	3rd cand	4th cand	dif1	ruido	time	Result	User
240	61	casa	719	caça		cata	caca	977	silencio	6,15	OK	Claudio
252	58	caça	975	cata		caca	capa	197	silencio	6,15	OK	Claudio
242	57	capa	1051	cata		caca	caça	263	silencio	5,94	OK	Claudio
255	61	cama	1193	casa		lata	coca	1813	silencio	6,03	OK	Claudio
255	60	cala	1326	casa		capa	cata	955	silencio	5,88	OK	Claudio
239	57	cata	1034	caça		capa	caca	178	silencio	5,83	OK	Claudio
255	59	caca	898	capa		cata	caça	516	silencio	6,37	OK	Claudio
243	58	lata	1083	cata		coca	capa	1112	silencio	5,83	OK	Claudio
255	62	coca	974	capa		lata	caça	1632	silencio	6,1	OK	Claudio
										6,031111	Media	

Tabela 10: Resultados dos testes com o conjunto 5

Teste	31.3.99	0	beijo	Acuracidade Aferida	100%
		1	queijo	Acuracidade Nominal	97%
		2	queixo		
		3	eixo		
		4	gueixa		
		5	brejo		
		6	tejo		

Treinamento

	comando	SNR	Level
0	beijo	255	61
1	queijo	255	62
2	queixo	255	61
3	eixo	255	62
4	gueixa	255	60
5	brejo	255	62
6	tejo	255	63

SNR	Level	Detetado	Score	2nd cand	Score2	3rd cand	4th cand	dif1	ruido	time	Result	User
255	60	beijo	763	queixo	1411	queijo	eixo	648	silencio	5,83	OK	Claudio
255	61	queijo	810	queixo	855	eixo	tejo	45	silencio	5,71	OK	Claudio
255	61	queixo	700	queijo	869	eixo	tejo	169	silencio	5,83	OK	Claudio
255	63	eixo	898	queixo	1273	queijo	tejo	375	silencio	5,76	OK	Claudio
255	60	gueixa	881	eixo	3012	beijo	queijo	2131	silencio	5,88	OK	Claudio
255	61	brejo	944	tejo	2020	beijo	queixo	1076	silencio	5,73	OK	Claudio
255	65	tejo	854	queijo	1393	queixo	eixo	539	silencio	5,66	OK	Claudio
										5,771429	Media	

Tabela 11 : Resultados dostestes com o conjunto 6

Teste	31.03.99	0	tomate	Acuracidade Aferida	100%
		1	boate	Acuracidade Nominal	97%
		2	malote		
		3	mascate		
		4	mascote		

Treinamento

	comando	SNR	Level
0	tomate	255	60
1	boate	255	64
2	malote	255	61
3	mascate	255	57
4	mascote	255	60

SNR	Level	Detetado	Score	2nd cand	:	3rd cand	4th cand	dif1	ruido	time	Result	User
255	62	tomate	1057	mascate		boate	malote	1022	silencio	5,6	OK	Claudio
255	63	boate	1331	malote		tomate	mascote	586	silencio	5,49	OK	Claudio
255	61	malote	770	boate		mascote	tomate	1247	silencio	5,38	OK	Claudio
255	58	mascate	799	mascote		tomate	malote	1031	silencio	5,76	OK	Claudio
255	61	mascote	921	malote		mascate	tomate	558	silencio	5,61	OK	Claudio
										5,568	Media	

4.1.4 Conjunto 7

Utilizou-se para o teste do conjunto 7 de comandos vocais os seguintes recursos:

- Ambiente Silencioso (Home)
- Lista dos comandos vocais com nível decrescente de similaridade.
- Diferença aritmética do nível de similaridade entre o primeiro e segundo candidatos na lista (dif 1)

Tabela 12: Resultados dos testes com o conjunto 7

Teste	20.03.99	0	triciclo	Acuracidade Aferida	100%
		1	biociclo	Acuracidade Nominal	97%
		2	quadriciculo		
		3	monociclo		
		4	pentaciclo		
		5	hexaciclo		

Treinamento

	comando	SNR	Level
0	triciclo	255	56
1	biociclo	255	61
2	quadriciculo	255	61
3	monociclo	255	63
4	pentaciclo	252	62
5	hexaciclo	255	59

SNR	Level	Detetado	Score	2nd cand	3rd cand	4th cand	dif1	ruido	time	Result	User
248	55	triciclo	1278	biociclo	pentaciclo	quadriciclo	1093	silencio	5,61	OK	Claudio
255	60	biociclo	1569	triciclo	pentaciclo	monociclo	907	silencio	5,66	OK	Claudio
243	58	quadricick	1169	triciclo	biociclo	hexaciclo	589	silencio	5,88	OK	Claudio
255	63	monocick	1104	biociclo	quadriciclo	pentaciclo	1141	silencio	5,6	OK	Claudio
245	61	pentacick	1739	biociclo	triciclo	hexaciclo	515	silencio	6,21	OK	Claudio
242	58	hexaciclo	1341	pentaciclo	quadriciclo	triciclo	1057	silencio	5,89	OK	Claudio
									5,792	Media	

5 Discussões

Conjuntos 1 e 2

Os testes realizados nos conjuntos 1 e 2, por três diferentes usuários, apresentaram uma taxa de acerto do comando vocal de 99%. O tempo médio de resposta do sistema foi de seis segundos. A diferença do nível de similaridade entre o comando vocal reconhecido e o segundo classificado, ficou superior a 800 (número absoluto).

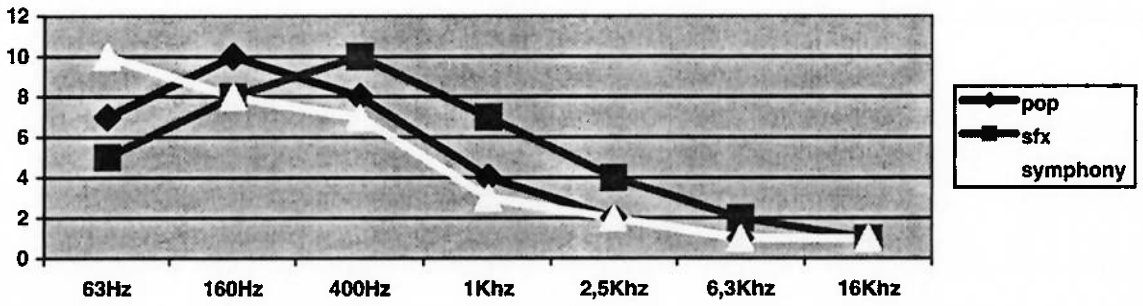
Conjunto 3

Nos testes realizados com o conjunto 3 de comandos vocais , simulou-se o funcionamento de uma agenda eletrônica com comando vocal.

Teste de identificação de palavras foneticamente semelhantes

Basicamente, observou-se o comportamento do sistema com a inserção das palavras "Júlio" e "Rulio" no conjunto de comandos ativos. O sistema não apresentou nenhuma dificuldade para identificar as duas palavras apesar da similaridade fonética entre ambas.

Influência de Ruídos Ambientais na performance do sistema



Conforme citado anteriormente, o sistema foi submetido à operação em ambientes com os ruídos sfx, pop e symphony. Os ruídos sfx e o symphony, apesar de reduzirem o nível de acuidade do sistema, ainda permitiram um alto índice de acerto. Surpreendentemente o tempo de resposta do sistema foi reduzido, descendo para uma média de 5,1 segundos. O ruído pop afetou de modo crítico a performance do sistema, prejudicando o reconhecimento dos comandos vocais.

Este comportamento pode ser explicado pelas características do microfone utilizado para captação dos comandos vocais . Por exemplo o ruído pop apresenta amplitude máxima em seu sinal sonoro na faixa dos 160Hz, que é exatamente a faixa de frequência de maior sensibilidade do microfone. Os ruídos sfx, com pico a 400Hz e symphony com pico a 60Hz ativam faixas de frequência aos quais o microfone é menos sensível.

Influência do Número de Comandos vocais ativos no tempo de resposta do sistema

Para um conjunto de comandos entre 1 a 16 , a variação do tempo de resposta do sistema foi de 1 segundo. Ou seja, quanto menor é o conjunto de comandos vocais a serem analisados, menor é o tempo de resposta do sistema.

Diferenças do tempo de resposta e da performance do sistema nos modos de operação silêncio (home) e ruidoso (car)

A performance em condições de baixo ruído ambiental, alternando-se o modo de operação car ou home, permaneceu inalterada mas o tempo de resposta do sistema teve uma alteração de aproximadamente 1 segundo, subindo de uma faixa de 6 segundos para uma faixa de 7 segundos.

Em condições de ruído ambiental nas formas sfx, pop e symphony, o sistema apresentou performance surpreendente no modo car. O tempo de resposta, que no modo home era de aproximadamente 7 segundos, caiu para a faixa dos 5 segundos . A taxa de reconhecimento manteve-se bastante elevada, mas a diferença do parâmetro de similaridade entre os comandos vocais sofreu uma forte redução. Nestas mesmas condições de ruído ambiental, o sistema, ajustado no modo home, apresentou índice de acerto de comandos vocais reduzido e tempo de resposta alto.

<u>Conjuntos 4, 5 e 6</u>

Teste de identificação de palavras foneticamente semelhantes

Os conjuntos de comandos vocais 4, 5 e 6 foram elaborados basicamente para o teste de performance do sistema na identificação de palavras foneticamente semelhantes.

Nos três testes efetuados detetamos uma alta capacidade do sistema de identificar palavras foneticamente semelhantes.

Conjunto 7

Neste caso, foram testados comandos vocais com mesma terminação fonética. A performance do sistema manteve-se elevada com um alto índice de acerto e um baixo tempo de resposta . Identificou-se aqui uma peculiaridade deste tipo de vocabulário. O sistema só consegue identificar os comandos quando o usuário faz uma forte acentuação na partícula de diferenciação, tornando assim a partícula foneticamente comum uma partícula átona. Este fato pode ser detetado pela baixa taxa Sinal – Ruído (SNR) apresentado pelos comandos vocais em que a regra não foi respeitada.

6 Conclusão

De modo geral, pode-se dizer que a performance apresentada pelo sistema, no que se refere a taxa de acerto, ficou acima das expectativas. A taxa nominal de acerto de 97% (informada pelo fabricante) foi superada, ficando muito próximo dos 100%.

O tempo de resposta do sistema, na faixa dos 5 segundos, apesar de aceitável, mostrou-se acima dos padrões esperados (na faixa de 1 segundo). Este fato se deve muito mais ao tempo de processamento da CPU para a interface gráfica e em processos de consulta ao banco de dados do sistema do que o tempo de resposta da placa DSP ao comando vocal captado. Um aprimoramento da programação da CPU , aliado a utilização de uma CPU superior a utilizada (CPU Intel Pentium 133) pode reduzir, de modo sensível, o tempo de resposta do sistema.

O sistema apresentou altas taxas de acerto quando submetido a diversos conjuntos de comandos vocais em diferentes níveis de dificuldade. Pode-se ressaltar a forte habilidade do sistema em identificar comando foneticamente semelhantes.

Sob condições de ruído ambiente, a operação do sistema, no modo car, apresentou sensível melhora de performance em relação a operação no modo home.

Foi comprovado que o sistema só responde ao usuário responsável pela entrada dos comandos vocais na fase de treinamento.

A qualidade do processo de treinamento, executado durante a operação inicial do chipset, é essencial para a otimização da taxa de reconhecimento do sistema. Quanto mais cuidadosa for a fase de treinamento e quanto menor for o ruído ambiente nesta fase, melhores serão os resultados na fase de reconhecimento

O trabalho foi concluído atingindo todos os objetivos projetados. O desempenho superou as referências de tempo de resposta e acuidade nominais informados pelo fabricante.

Tempo Médio de Resposta Aferido (do chip)	0,453seg
Tempo Médio de Resposta Aferido (da interface gráfica)	5,000seg
Tempo de Resposta Nominal	0,500seg
Taxa de Acuidade de Reconhecimento Aferido	98,86%
Taxa de Acuidade de Reconhecimento Nominal	97,00%

Tabela 13 Medições de Performance do Sistema

6.1 Proposição de temas para desenvolvimento de estudos futuros de melhoria do sistema implementado

Uma evolução para o sistema desenvolvido seria substituir o chip D6106 dependente do usuário por um chip experimental D306 independente do usuário. Nesta nova versão, a fase de treinamento independeria do usuário final do sistema, o que permitiria ter uma única fase de treinamento para inúmeros usuários.

O microfone e o sistema de filtragem do comando vocal também poderiam ser aprimorados, mudando a frequência de sensibilidade e de atenuação conforme o ruído ambiente.

A Interface Homem – Máquina também precisa ser aprimorada de modo a tornar o sistema totalmente intuitivo e controlado por comandos vocais.

7 Referências Bibliográficas

1) Rabiner, L.R., Schafer, R.W. 1978 Digital Processing of Speech Signals, Englewood Cliffs, N.J.; Prentice-Hall, Inc.
2) Higgins, R. J., Digital Signal Processing in Analog Devices, Prentice Hall,1990)
3) Levinson, S.E.and ROE.D.B. A Perspective on Speech Recognition, IEEE Comm.Magazine p.28.34 Jan 1990
4) Levinson S. E. Structural Methods in Automatic Speech Recognition Proc.Ieee,vol.73,n.11,p.1625-1650,nov.1985
5) Markel, J. D., and Gray, A. H. Jr. 1980. Linear Prediction of Speech. New York: Spring-Verlag
6) Minami, M., Reconhecedor de Palavras Isoladas, Independente do Falante Usando HMM Discreto, Dissertação de Tese de mestrado , Poli-Usp- 1993
7) Robinson, T., Speech Recognition with Associative Networks, Cambridge University, 1986
8) Michaelmas, Introduction to Speech Recognition, 1995
9) Entropic, ASR Development Package, unix Toolset for Speech Recognition Development.
10) DSPC 6106 Application Notes
11) Salvador, A.C.; Sundstrom, G.A., Information organization, access and management advantages of a task-based HMI for telecommunications network monitoring, Conference Proceedings. 1993 International Conference on Systems, Man and Cybernetics. Systems Engineering in the Service of Humans (Cat. No.93CH3242-5) p. 357-63 vol.2
12) Mori, H.; Sakamura, K., Complexity optimization technique for sound synthesis on digital sound processing architectures, Proceedings. 10th TRON Project International Symposium p. 113-25
13) Ikeda, H.; Kondo, K.; Sakata, M.; Kataoka, Y.; Fujie, K.; Tanaka, TRON HMI design analysis by the semiotic approach M. Proceedings. 10th TRON Project International Symposium p. 31-6
14) Fujii, T. : Problems of HMI guidelines for 3-D user interface , Proceedings. 10th TRON Project International Symposium p. 29-30
15) Sakamura, K., Bibliography of the TRON project (1984-1994), Proceedings of the 11th TRON Project International Symposium (Cat. No.94TH8027) p. 146-73
16) Kirkpatrick, D., Developing a commercial engineering application inSmalltalk, Object Magazine, Vol: 4 Iss: 6 p. 33-4, 36-8 Date: Oct. 1994
17) Tavatia, S.; Porayath, R.; Doherty, J.F.,Lattice CELP for low bit rate speech coding, 1994 IEEE MILCOM

18) Jey-Hsin Yao; Tanaka, Y., Low-bit-rate speech coding with mixed-excitation and interpolated LPC coefficients, ICSLP 94. 1994 International Conference on Spoken Language Processing p. 2079-82 vol.4

19) McCree, A.V.; Barnwell, T.P., III , A mixed excitation LPC vocoder model for low bit rate speech coding, IEEE Transactions on Speech and Audio Processing Vol: 3 Iss: 4 p. 242-50 Date: July 1995

20) Karema, T.; Ofner, E, A single-chip GSM vocoder with analog front end, ESSCIRC '94. Twentieth European Solid-State Circuits Conference. Proceedings p. 132-5

21) Basztura, C., Speech signal transmission rate compression using time parameters coding, Archives of Acoustics Vol: 19 Iss: 1 p. 5-35 Date: 1994

22) Poornaiah, D.V.; Mohan, P.V.A.; Chadchan , Design and implementation of a programmable bit-rate multipulse excited LPC vocoder for digital cellular radio applications, 1994 IEEE International Conference on Personal Wireless Communications. Conference Proceedings (Cat. No.94TH0666-8) p. 209-15

23) Waldemar, P.; Ramstad, T.A., Design of gain optimized perfect reconstruction regular lattice filter banks, Proceedings of the SPIE - The International Society for Optical Engineering Vol: 2308 Iss: pt.2 p. 963-70 Date: 1994

24) Atkinson, I.A.; Kondoz, A.M.; Evans, B.G , Time envelope vocoder, a natural sounding speech coding algorithm operating at 1.6 kbits/sec, Fifth IEE Conference on Telecommunications' (Conf. Publ. No.404) p. 215-19

25) Narendranath, M.; Murthy, H.A.; Rajendran, S.;Yegnanarayana, B., Transformation of formants for voice conversion using artificial neural networks, Speech Communication Vol: 16 Iss: 2 p. 207-16 Date: Feb. 1995

26) Moulines, E.; Laroche, J., Non-parametric techniques for pitch-scale and time-scale modification of speech, Speech Communication Vol: 16 Iss: 2 p. 175-205 Date: Feb. 1995

27) Vanzieleghem, E.; Schelfhout, K.; Dartois, L.; Wenin, J.; Vanwelsenaers, A.; Rabaey, D.H., A compact and power efficient GSM vocoder, ICC '93 Geneva. IEEE International Conference on Communications '93. Technical Program, Conference Record(Cat.No.93CH3261-5) p. 207-11 vol.1

28) Atkinson, I.A.; Kondoz, A.M.; Evans, B.G., 1.6 kbit/s LP vocoder using time envelope, Electronics Letters Vol: 31 Iss: 7 p. 517-19 Date: 30 March 1995

29) Nurmi, J.; Eerola, V.; Ofner, E.; Gierlinger, A.; Jernej, J.; Karema, T.; Raita-aho, T.,A DSP core for speech coding applications, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. II/429-32 vol.2

30) Gao Yang; Leich, H., High-quality harmonic coding at very low bit rates, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/181-4 vol.1

31) Tanaka, Y.; Kimura, H., Low-bit-rate speech coding using a two-dimensional transform of residual signals and waveform interpolation, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat. No.94CH3387-8) p. I/173-6 vol.1

32) McDonough, J.; Chienchung Chang; Kantak, P.; Sakamaki, C.; Singh, R.; Ming-Chang Tsai , A single chip QCELP vocoder for CDMA digital cellular, Proceedings of the IEEE 1994 Custom Integrated Circuits Conference (Cat. No.94CH3427-2) p. 211-14

33) Atkinson, I.A.; Kondoz, A.M.; Evans, B.G.,Time envelope vocoder, a new LP based coding strategy for use at bit rates of 2.4 kb/s and below, IEEE Journal on Selected Areas in Communications Vol: 13 Iss: 2 p. 449-57 Date: Feb. 1995

34) Wright, M., Examples of ZIPI applications,Computer Music Journal Vol: 18 Iss: 4 p. 81-5 Date: Winter 1994

35) Hansen, J.H.L.; Nandkumar, S.,Objective speech quality assessment and the RPE-LTP coding algorithm in different noise and language conditions, Journal of the Acoustical Society of America Vol: 97 Iss: 1 p. 609-27 Date: Jan. 1995

36) Wada, C.; Ifukube, T.; Ino, S.; Izumi, T., Proposal of a new tactile display method of speech signals as a nonverbal communication for the profoundly hearing impaired, Proceedings. 3rd IEEE International Workshop on Robot and Human Communication. RO-MAN '94 Nagoya (Cat.No.94TH0679-1) p. 95-100

37) Gimenez de los Galanes, F.M.; Savoji, M.H.; Pardo, J.M., New algorithm for spectral smoothing and envelope modification for LP-PSOLA synthesis, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat. No.94CH3387-8) p. I/573-6 vol.1

38) Gao Yang; Leich, H., High-quality harmonic coding at very low bit rates, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/181-4 vol.1

39) Jankowski Jr., Hoang-Doan H. Vo, and R. P. Lippmann, A Comparison of Signal Processing Front Ends for Automatic Word Recognition", ", IEEE Transactions of Speech and Audio Processing, vol 3, n.4, July 1995

40) D. Hakerman, J. S. Breese, and K. Rommelse, "Decision-Theoric Troubleshooting", Communications of the ACM, vol.38, n.5. March 1995

41) M. Hochberg, S. Renals, and A. Robinson, "ABBOT: The CUED hybrid connectinist-HMM large vocabulary recognition system", in Proc. ARPA Spoken Language Technology Workshop, 1994

42) X. D. Huang, K. F. Lee, and A. Waibel, "Connectionist speaker normalization and its application to speech recognition", IEEE Workshop pn Neural Networks for Signal Processinig, pp.357-366 IEEE Press, 1991

43) P. Woodland and S. Young, "The HTK tied-state continuous speech recognizer", Eurospeech 93 pp.2207-2210, 1993

8 Bibliografia Recomendada

1) Levy, M. Coyle A. DSP-Chip Directory, EDN March 1, 1996 p. 40 – 106
2) Keil Elektronik GmbH, 8051 utilities
3) Pentek, Innovative DSP systems for VMEbus and Multibus
4) Analog Devices, DSP Applications
5) TI DSP Applications
6) DSPC 306 Application Notes
7) Roy, D.M.; Erenshteyn, R.; Panayi, M.; Harwin, W.S.; Foulds, R.; Fawcus, R., Computer recognition of imprecise dynamic arm gestures of people with severe motor impairment, World Congress on Neural Networks-San Diego. 1994 International Neural Network Society Annual Meeting p. II/191-6 vol.2
8) Saito, H.; Ishiwaka, T.; Okabayashi, S., Applications of driver's line of sight to automobiles what can driver's eye tell, 1994 Vehicle Navigation and Information Systems, Conference Proceedings (Cat. No.94CH35703) p. 21-6
9) Tohjo, H.; Yoda, I.; Kimura, T.; Fujii, N., CMIP-based OpS-WS interface supporting graphical user interface, IEICE Transactions on Communications Vol: E78-B Iss: 1 p. 74-81 Date: Jan. 1995
10) Pleczon, P.; Chalard, S., The human-machine interface of ProLab2 copilot, Proceedings of the Intelligent Vehicles '94 Symposium (Cat. No.94TH8011) p. 461-6
11) JooHun Lee; HongYeol Jeon; MyungJin Bae; SouGuil Ann, A fast pitch searching algorithm using correlation characteristics in CELP vocoder, 1994 IEEE MILCOM. Conference Record (Cat. No.94CH34009) p. 699-702 vol.3
12) Ferrer-Ballester, M.A.; Figueiras-Vidal, A.R., Efficient adaptive vector quantization of LPC parameters, IEEE Transactions on Speech and Audio Processing Vol: 3 Iss: 4 p. 314-17, July 1995
13) McCree, A.V.; Barnwell, T.P., III, A mixed excitation LPC vocoder model for low bit rate speech coding, IEEE Transactions on Speech and Audio Processing Vol: 3 Iss: 4 p. 242-50, July 1995
14) Kwon, C.H.; Un, C.K., Improving the adaptive source model for CELP coding withlong analysis frame size, Speech Communication, Vol: 16 Iss: 4 p. 423-33 June 1995
15) Miki, S.; Moriya, T.; Mano, K.; Ohmuro, H.,Pitch synchronous innovation code excited linear prediction (PSI-CELP), Journal: Electronics and Communications in Japan [Fundamental Electronic Science] Vol: 77 Iss: 12 p. 36-49, Part 3, Dec 1994
16) Ma Hong Fei; Fan Changxin ,A method to encode speech at 1200 bps, Journal of Xidian University, Vol: 20 Iss: suppl.issue p. 31, 53-7, 1993
17) Usagawa, T.; Iwata, M.; Ebata, M., Speech parameter extraction in noisy environment using a masking model, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. II/81-4 vol.2, 1994

18) Jianping Pan; Fischer, T.R., Vector quantization-lattice vector quantization of speech LPC coefficients, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/513-16 vol.1

19) Hagen, R.,Spectral quantization of cepstral coefficients, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat. No.94CH3387-8) p. I/509-12 vol.1

20) Bruhn, S., Efficient interblock noiseless coding of speech LPC parameters,ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/501-4 vol.1

21) Skinnemoen, H.; Perkis, A., Efficient vector quantisation of LPC parameters for noisy channels, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat. No.94CH3387-8) p. I/497-500 vol.1

22) Kwok-Wah Law; Cheung-Fat Chan , A novel split residual vector quantization scheme for low bit rate speech coding,ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/493-6 vol.1

23) Kohata, M.; Takagi, T., Vector quantization with hyper-columnar clusters, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/489-92 vol.1

24) Kobatake, H.; Matsunoo, Y , Degraded word recognition based on segmental signal-to-noise ratio weighting, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat. No.94CH3387-8) p. I/425-8 vol.1

25) Ricart, R.; Cupples, J.; Fenstermacher, L., Speaker recognition in tactical communications, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/329-32 vol.1

26) Ozawa, K.; Serizawa, M.; Miyano, T.; Nomura, T., M-LCELP speech coding at 4 kbps, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/269-72 vol.1

27) Sun-Won Park , Speech compression using ARMA model and wavelet transform, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat. No.94CH3387-8) p. I/209-12 vol.1

28) Thyssen, J.; Nielsen, H.; Hansen, S.D , Non-linear short-term prediction in speech coding,ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/185-8 vol.1

29) Kondo, K.; Picone, J.; Wheatley, B.,A comparative analysis of Japanese and English digit recognition, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat.No.94CH3387-8) p. I/101-4 vol.1

30) Savic, M.; Huiqin Gao; Sorensen, J.S.,Co-channel speaker separation based on maximum-likelihood deconvolution, ICASSP-94. 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (Cat. No.94CH3387-8) p. I/25-8 vol.1

31) Miki, S.; Moriya, T.; Mano, K.; Ohmuro, H.,Basic algorithm of pitch synchronous innovation CELP (PSI-CELP) speech coding, NTT Review, Vol: 6 Iss: 6 p. 53-60, Nov. 1994

32) Mano, K.; Moriya, T.; Miki, S.; Ohmuro, H.; Ikeda, K.; Ikedo, J., Design of a pitch synchronous innovation CELP coder for mobile communications, IEEE Journal on Selected Areas in Communications, Vol: 13 Iss: 1 p. 31-41, Jan. 1995

33) Asanuma, N.; Nagabuchi, H.,A new reference signal for evaluating the quality of speech coded at low bit rates, Electronics and Communications in Japan, Part 3 [Fundamental Electronic Science] Vol: 77 Iss: 5 p. 39-45, May 1994

34) Ozawa, K.; Serizawa, M.; Miyano, T.; Nomura, T.; Ikekawa, M.; Taumi, S., M-LCELP speech coding at 4 kb/s with multi-mode and multi-codebook, IEICE Transactions on Communications, Vol: E77-B Iss: 9 p. 1114-21, Sept. 1994

35) Watson, D. Kewley-Port, D.J. Reed, and D. Maki, "The Indiana Speech training Aid (ISTRA) I: Comparison between human and computer-based evaluation of speech quality", J. Speech Hearing Res., vol. 32, p 245-251, 1989

36) Anderson, Kewley-Port, "Evaluation of Speech Recognizers for Speech Training Applications", IEEE Transactions of Speech and Audio Processing, vol 3, n.4, July 1995

37) K. Kwan, "A fuzzy neural network and its Application to Pattern Recognition", IEEE Transactions of Speech and Audio Processing, vol 2 n.3 August 1994

38) V. McCree, and T. P. Barnwell III, "A Mixed Excitation LPC Vocoder Model for Low Bit Rate Speech Coding, IEEE Transactions of Speech and Audio Processing, vol 3, n.4, July 1995

39) A. Ferrer Ballester, and A. R. Figueiras-Vidal, "Effecient Adaptive Vector Quantization of LPC Parameters", IEEE Transactions of Speech and Audio Processing, vol 3, n.4, July 1995

40) H. Kuo, Cheng-I Kao, Jiahn-Jung Chen, "A Fuzzy Neural Network Model and Its Hardware Inplementation", IEEE Transactions of Speech and Audio Processing, vol 1, n.3, August 1993

41) Chang, and S. -H Chen, "Isolated Mandarin Syllable Recognition Using Segmental Features", IEE Proc. -Vis. Image Signal Process., Vol 142, no. 1 , February 1995

42) M. Peinado, J. C. Segura, A. J. Rubio, V. E. Sánchez, P. Garcia, Use of multiple Vector Quantisation for Semicontinuous-HMM Speech Recognition, IEE Proc. -Vis. Image Signal Process., Vol 141, no. 6 , December 1994

43) K. Ong, A. M. Kondoz, B. G. Evans, "Enhanced Channel Coding Using Source Criteria in Speech Coders", IEE Proc. -Vis. Image Signal Process., Vol 141, no. 3 , June 1994

44) D. Hackerman, A. Mandani, and M. P. Wellman, " Real-World Applications of Bayesian Networks", Communications of the ACM, vol.38, n.3. March 1995

45) R. Fung, B. Del Farvero, "Applying bayesian Networks to Information Retrieval", Communications of the ACM, vol.38, n.3. March 1995

46) N. Morgan, and H. Bourland, "Continuous Speech Recognition", IEEE Signal Processing Magazine, pag. 25, May 1995

47) S. Amari, "A theory of Adaptive Pattern Classifiers", IEEE Trans. on Elec. Com., vol EC 16, pp 279-307, 1967

48) G. Zavaliagkos, Y. Zhao, R. Schwartz, and Makhoul, "A hybrid segmental neural net/hidden markov model system for continuous speech recognition" IEEE Trans. on Speech and Audio Processing, vol.2, n.1, pp 151-160, 1994

49) H. Bourland Y. Konig and N. Morgan, "REMAP: Recursive estimation and maximization of a posterior probabilities Application to transition-based connectionist speech recognition, "ISCSI Tecnical Report TR 94-064, 1994

50) H. Bourlard and N. Morgan, Connectionist Speech recognition - A Hybrid Approach, Kluwer Academic Press, 1994

51) M. Cohen, H. Franco, N. Morgan, D. Rumelhart, and V. Abrash, "Context-Dependent Multiple Distribution Phonetic Modeling", in Advances in Neural Information Processing Systems 5, pp. 649-657, 1993

52) H. Franco, M. Cohen, N. Morgan, D. Rumelhart, and V. Abrash, "Context-dependent connectionist probability estimation in a hybrid hidden Markov model-neural net speech recognition system", Computer Speech and Language, vol. 8, n. 3. pp. 211-222, july 1994

53) O. Ghitza and M.M. Sondhi, "Hidden Markov Models with templates as non-stationary states: an appplication to speech recognition", Computer Speech and Language, 2: 101-119 , 1993

54) H. Hermansky, "Perceptual Linear Predictive (PLP) analysis of speech", Jounal of the Acoust. Soc. Am., vol. 87, n.4, 1990

55) H. Hermansky, and N. Morgan, "RASTA processing of Speech", IEEE Transactions on Speech and Audio Processing, special issue on Robust Speech Recognition, vol.2 n.4, pp 578-589, Oct 1994

56) D. Jurafsky, C. Wooters, G. Tajchman, J. Segal, A. Stocke, and N. Morgan, "The Berkeley restaurant Project", in Proc. Intl. Conf. on Spoken Language Processing (Yokohama Japan)

57) N. Morgan, H. Hermansky, H. Bourlard, P. Kohn, and C. Wooters, "Continuous Speech Recornition Using PLP analysis with Multilayer perceptrons, "Proc. IEEE Trans. on Acoustics, Speech, and Signal Processing, pp 49-52, Toronto, Canada, 1991

58) L. R. Rabiner, "A tutorial on hidden Markov models aond selected applications in speech recognition", Proceedings of IEEE, vol. 77, n. 2, p 257-285, 1989

59) A. Poritz, "linear Predictive Hidden Markov Models and the speech signal", proc. IEEE Proc. Intl. Conf. on Acoustic, Speech and Signal Processing, pp.1291-1294, Paris, 1982

60) R. Schaefer and L. Rabiner, "digital representations of speech signals", Proceedings of the IEEE, vol 63, n4, p662-667, 1975

61) H. Lucke, "Bayesian Belief Networks as a tool for stochastic parsing", Speech Communication v. 16 p 89-118 ,1995

62) S. B. Davis and P. Mermelstein, "Comparison of parametric Representations for monosyllabic word recognition in continuousy spoken sentences", IEEE Trans. Acoust. , Speech, Signal Processing, vol. ASSP-28, n.4, p 357-366, 1980

63) O. Ghitza, "Auditory nerve representation as a baisis for speech processing", in Advances in Speech Signal Processing (S. Furui and M. M. Sondhi, eds.) New York: Marcel Dekker, 1992, p.453-486

64) R. M. Stern, F. -H. Liu, Y. Ohshima, T. M. Sullivan, and A. Acero, "Multiple Approaches to Robust speech recognition", in proc. DARPA Speech & Natural language Workshop, Harriman, NY, 1992, pp- 274-279

65) M. J. Hunt and C. Lefebvre, "A comparison of several acoustic representations for speech recognition with degraded and undergraded speech", in proc. Int. Conf. Acoust., Speech, Signal Processing, Glasgow, 1989, p 262-265

66) B. Delgutte and N. Y. Kiang, "Speech Coding in the Auditory Nerve", J. Acoust. Soc. Amer. , v. 75, pp. 866-919, 1984

67) M. B. Sachs and E. D. Young, "Encoding of Steady State vowels in the auditory nerve: Representation in terms of discharge rate", J. Acoust Soc. Amer. , vol.66 pp 470-479, 1979

68) W. M. Siebert, "Frequency discrimination in the auditory system: Place or Periodic Mechanism ? , proc. IEEE, v. 58, p 723-730, 1970

69) P. J. Rajesekaran, G. R. Doddington, and J. W. Picone, "Recognition of Speech under stress and in noise" in Proc. int. Conf. Acoust. , Speech, Signal Processing, Tokyo, Japan, 1986, p. 733-736

70) D. B. Paul, "Speech Recognition using Hidden Markov Models" , Lincoln Lab. J., vol.3, n.1, pp41-62, 1990

71) L. R. Rabiner and R. Schafer, Digital Processing of Speech Signals, Englewood Cliffs, NJ, Prentice-Hall, 1979

72) D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus" in Proc. DARPA Speech and Natural Language Workshop, Harriman, NY , 1992, pp. 357-362

73) K. Shikano, "Evaluation of LPC spectral matching measures for phonetic unit recognition", tech. rep. Comput. Sci. Dept. Carnegie-Mellon Univ., Pittburg, PA, 1986

74) S. Amari, "A theory of Adaptive Pattern Classifiers", IEEE Trans. on elec. Com., vol. EC16, pp. 279-307, 1967

75) J. Baker, "The Dragon System - An overview", IEEE Trans. on Acoustics Peech and Signal Processing, vol. 23, n.1, pp.24-29, 1975

76) L. Baum, "An inequality and associated maximation techniques in statistical estimation of probabilistic functions of Markov processes", Inequalities, n.3, pp 1-8, 1972

77) Y. Bengio, R. de Mori, G. Flammia, R. Kompe, "Global Optimization of a neural network-Hidden Markov Model hybrid", IEEE trans. on Neural Networks, vol. 3, n.2, pp 252-259, 1992

78) H. Bourlard and N. Morgan, "A continuous speech recognition System embedding MLP into HMM, "in Advances in Neural Information Processing Systems 2 (D. S. Touretzky,Ed.) pp.413-416, Morgan Kaufmann, San mateo CA, 1990

79) H. Bourlard and N. Morgan, "CDNN: A context dependent neural network for noncontinuous speech recognition", IEEE Proc. Intl. Conf. on Acoustic, Speech, and Signal Processing (San francisco, CA), pp. II:349-352, 1992

80) H. Bourlard and N. Morgan, Connectionist Speech Recognition - A Hybrid Approach, Kluwer Academic Press, 1994 (LIVRO)

81) H. Bourlard and C.J. Wellekens, "links between Markov models and multilayer perceptrons", IEEE transactions on Pattern Analysis and Machine Intelligence, vol. 12, pp. 1167-1178, 1990

82) J. S. Bridle, "Probabilistic Interpretation of feedforward classification networks outputs, with relationships to statistical pattern recognition", in Neurocomputing: Algorithms, Architectures and Applications. F. Fogelman Soulié and J. Hérault(Eds.) NATO ASI Series, pp 227-236, 1990

83) J. S. Bridle, "Alpha-Nets: a recurrent neural network architeture with a hidden Markov model interpretation", Speech Communication, v.9 p. 83-92, 1990

84) D. Broomhead, D. Lowe , "Multi-variable functional interpolation and adaptive networks", Complex Systems, v.2, pp.321-355, 1988

85) P. F. Brown, "The Acoustic Modelling Problem in Automatic Speech Recognition", PhD Thesis, School of Computer Science, Carnegie Mellon University, 1987

86) Y. Chow, M. Dunham, O. Kimball, M. Krasner, G. Kubala, J. Makhoul, P. Price, S. Roucos, and R. Schwarz, "BYBLOS: The BBN continuous speech recognition system", Proc, IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing, Dallas, Texas, pp 89-92, 1987

87) M. Cohen, 'Phonological Structures for Speech Recognition", PhD Thesis, University of California at Berkeley, 1989

88) M. Cohen, H. Murveit, J. bermstein, P. Price and M. weintraub, "The DECIPHER speech recognition system", in proc. IEEE Intl. Conf. on Acoustic, Speech and Signal Processing (Albuquerque, NM), p 77-80, 1990

89) L. Deng, "A generalized hidden markov model with state-conditioned trend functions of time for the speech signal", Signal Processing,27:65-78, 1992

90) V.V. Digalakis, J. R. Rohlicek, and M. Ostenford, "Segment-based stochastic models of spectral dynamics for continuous speech recognition", IEEE Trans. on Speech and Audio Processing, 1(4):431-442, October 1993

91) R.O. Duda and P. E. Hart, Pattern Classification and Scene Analysis, Wiley Interscience, New york, 1973

92) M. Franzini, K. F. Lee, and A. Waibel, "Connectionist Viterbi training: a new hybrid method for continuous speech recognition"IEEE Proc. Intl. Conf. on Acoustics, Speech and Signal Processing, pp. 425-428, Albuquerque, NM, 1990

93) S. Furui, "Speaker independent isolated word recognizer using dynamic features of speech spectrum", IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. 34, n. 1, pp. 52-59, 1986

94) O. Guitza and M.M. Sondhi, "Hidden Markov models with templates as non-stationary states: an application to speech recognition", Computer Speech and Language,2:101-119, 1993

95) H. Gish, "A probabilistic approach to the understanding and training of neural network classifiers", in Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (Albuquerque, NM), pp. 1361-1364, 1990

96) R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition", IEEE Proc. Intl. Conf. on Acoustics, Speech and Signal Processing, pp.1-13-16, San Francisco, CA, 1992

97) J. Hamshire and A. Waibel, "Connectionist architectures for multi-speaker phoneme recognition", in Advances in Neural Information Processing 2 (D.S. Touretszky, Ed.), Morgan Kaufamnn, CA, 1990

98) J. Hertz, A. Krogh, and R. Palmer, Introduction to the Theory of Neural Networks, Addison Wesley, 1991

99) R. Jacobs and M. Jordan M., "Linear piecewize control strategies in a modular neural network architecture", IEEE Trans. on Systems, Man, and Cybernetics, March/April 1993, vol. 23, n.2, pp. 337-345, 1993

100) F. Jelinek, "Continuous speech recognition by statistical methods", Proceedings of the IEEE, v. 64, n.4, pp. 532-555, 1976

101) F. Jelinek, Self Organized modeling for speech recognition", in Readings in Speech Recognition, A. Waibel and K. Lee (eds.), pp. 450-503, Morgan Kaufmann, 1990

102) L. Jiang and E. Barnard, "Choosing contexts for neural networks", Oregoon Graduate institute Technical Report, 1994

103) T. Kohonen, "The "neural"phonetic typewriter", IEEE Computer: 11-22 , 1988

104) Y. Konig, N. Morgan, C. Wooters, V. Abrash. , M. Cohen, and H. Franco, "Modeling consistency in a speaker independent continuous speech recognition system", in Advances in Neural Information Processing Systems 5 (S. J. Hanson, J. D. Cowan, and C.L. Giles, Eds.), pp. 682-687, 1993

105) K. F. Lee, Large Vocabulary Speaker-Independent continuous speech recognition: The SPHINX system, Kluwer Academic Publishers, 1988

106) E. Levin, "Speech recognition Using Hidden Control Neural Network Architecture", in Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (Albuquerque, NM), pp-433-436, 1990

107) R. P. Lippmann, "Review of neural networks for speech recognition", neural Computation, v.1, n.1, p 1-38, 1989

108) R. P. Lippmann, and E. Singer, "Hybrid neural-network/HMM approaches to wordspotting", Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing, Minneapolis, Minn, pp. 1-565-568, 1993

109) R. Lippmann, personal communication, 1994

110) D. M. Lubensky, A.O. Asadi, and J. M. Naik, "Connected digit recognition using connectionist probability estimators and mixture-gaussian densities", IEEE Proc. of the Intl. Conf. on Spoken Language Processing, pp. 295-298, Yokohama, Japan, 1994

111) S. Makino, T. Kawabata, and K. Kido, "Recognition of consonant based on the perceptron model", Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing, Boston, Mass., pp. 738-741, 1983

112) M. Minsky, S. Papert, Perceptrons Cambridge, MA: MIT Press, 1969

113) N. Mirghafori, N. Morgan, H. Bourlard, "parallel training of MLP probability estimators for speech recognition: a gender- based approach", IEEE Workshop on Neural Networks for Signal Processing, Greece, pp.289-298, 1994

114) D. P. Morgan and C. L. Scofield, Neural Networks and Speech Processing, Kluwer Academic Publishers, 1991

115) N. Morgan, "Big Dumb Neural Nets (BDNN): a working brute force approach to speach recognition", Proceedings of the ICNN, vol. VII, pp. 4462-4465, 1994

116) N. Morgan, J. Beck, P. Kohn, J. Bilmes, E. Allman, and J. Beer, "The Ring Array Processor (RAP): a multiprocessing peripheral for connectionist applications", Journal of Parallel and Distributed Computing. Special Issue on Neural Networks, v. 14, p. 248-259, 1992

117) N. Morgan and H. Bourlard , "Generalization and parameter estimation in feedforward nets: some experiments", in Advances in Neural Information Processing System2, San Mateo, CA: Morgan Kaufmann, p. 630-637, 1990

118) N. Morgan, H. Bourlard, S. Greenberg, and H. Hermansky, "Stochastic Perceptual Auditory-Event-Based Models (SPAM) for Speech Recognition", Intl. Conference on Spoken Language Processing, pp. 1943-1946, 1994

119) N. Morgan, H. Hermansky, H. Bourlard, P. Kohn, and C. Wooters, "Continuous Speech Recognition Using PLP analysis with multilayer perceptrons", Proc IEEE Intl Conf. on Acoustics, Speech, and Signal Processing, pp.49-52, Toronto Canada, 1991

120) H. Ney, "The use of a one stage dynamic programming algorithm for connected word recognition", IEEE trans. on Acoustics, Speech, and Signal Processing, 32:263-271, 1984

121) D. Parker, "Learning Logic", Technical Report TR-47, Center for Computational Research in Economics and management Science, MIT, Cambridge, MA, 1985

122) S. M. Peeling, R. K. Moore, "Isolated digit recognition experiments using the multi-layer perceptron", Speech Communication, v. 7, pp. 403-409, 1988

123) A. Poritz, and A. L. Richter "On Hidden Markov models in isolated word recognition", Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing, pp. 14.3.1-4, Tokyo, Japan, 1986

124) S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist Optimization of tied mixture Hidden Markov Models", in Advances in Neural Information Processing Systems 4 (J. Moody, S. Hanson, and R. Lippmann, Eds.), San Mateo, CA: Morgan Kaufmann, pp. 167-174, 1992

125) S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist probability estimators in HMM speech recognition", IEEE trans. on Acoustics, Speech, and Signal Processing, v.2, n.1, pp 161-174, 1994

126) M. D. Richard and R. P. Lippmann, "Neural Network classifieres estimate Bayesian a posteriori probabilities", Neural Computation, n. 3, pp. 461-483, 1991

127) T. Robinson, L. Almeida, J. M. Boite, H. Bourlard, F. Fallside, M. Hochberg, D. Kershaw, P. Kohn, Y. Konig N. Morgan, J. P. Neto, S. Renals, M. Saerens, C. Wooters, "A neural network based, speaker independent, large vocabulary, continuous speech recognition system: The WERNICKE Project", Proc. EUROSPEECH 93 (berlim, Germany), pp. 1941-1944, 1993

128) F. Rosenblatt, Principles of Neurodynamics, Perceptrons and the Theory of Brain Mechanisms, Spartan Books, Washington, 1962

129) J. Tebelskis and A. Waibel, "Large vocabulary recognition using linked predictive neural networks", in proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing (Albuquerque, NM), pp. 437-440, 1990

130) P. Werbos, 'Beyond regression: new tools for prodiction and analysis in the behavioral science", PhD Thesis, harvard University, Cambridge, MA, 1974