

LEANDRO CÉSAR DELFINO

**CONTROLE ATIVO DE RUÍDO EM DUTOS
UTILIZANDO
PROCESSADORES DIGITAIS DE SINAIS**



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA MECÂNICA**

2005

LEANDRO CÉSAR DELFINO

**CONTROLE ATIVO DE RUÍDO EM DUTOS UTILIZANDO
PROCESSADORES DIGITAIS DE SINAIS**

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Mecânica da Universidade Federal de Uberlândia, como parte dos requisitos para a obtenção do título de **MESTRE EM ENGENHARIA MECÂNICA**.

Área de Concentração: Mecânica dos Sólidos e Vibrações

Orientador: Prof. Dr. Francisco Jose Ribeiro

UBERLÂNDIA – MG
2005

Dados Internacionais de Catalogação na Publicação (CIP)

D349c Delfino, Leandro César, 1979-
Controle ativo de ruído em dutos utilizando processadores
digitais de sinais / Leandro César Delfino. - 2006.
159 f. : il.

Orientador: Francisco Jose Ribeiro.

Dissertação (mestrado) – Universidade Federal de Uberlândia,
Programa de Pós-Graduação em Engenharia Mecânica.
Inclui bibliografia.

1. Acústica - Teses. 2. Controle de ruído - Teses. 2.
Processamento de Sinais - Técnicas digitais. I. Ribeiro, Francisco
José. II. Universidade Federal de Uberlândia. Programa de Pós-
Graduação em Engenharia Me-cânica. IV. Título.

CDU: 531.77

Elaborada pelo Sistema de Bibliotecas da UFU / Setor de Catalogação e Classificação

LEANDRO CÉSAR DELFINO

**CONTROLE ATIVO DE RUÍDO EM DUTOS UTILIZANDO
PROCESSADORES DIGITAIS DE SINAIS**

Dissertação _____ pelo Programa
de Pós-graduação em Engenharia Mecânica
da Universidade Federal de Uberlândia.

Área de Concentração: Mecânica dos Sólidos
e Vibrações

Banca Examinadora:

Prof. Dr. Francisco Jose Ribeiro – UFU – Orientador

Prof. Dr. Marcus Antonio Viana Duarte – UFU

Prof. Dr. Eduardo Bauzer Medeiros – UFMG

Uberlândia, 28 de Outubro de 2005.

Agradecimentos

À Universidade Federal de Uberlândia e à Faculdade de Engenharia Mecânica pela oportunidade de realizar este Curso.

À Deus.

À minha mãe Regina, por sempre acreditar.

À minha família, pelo apoio prestado.

Ao Professor José Francisco Ribeiro, pela dedicação, apoio e orientação.

Aos Professores Valder Steffen Jr. e Francisco Paulo Lepore Neto pelas colaborações prestadas durante o trabalho.

Aos colegas de laboratório Israel Jorge Cárdenas Nuñez e Patrick Magalhães Cardoso.

À minha namorada Aretuza.

Aos amigos de Uberlândia e Uberaba.

À todos os demais colegas do programa de pós-graduação.

À CAPES - Coordenação de Aperfeiçoamento de Pessoas de Nível Superior, pelo apoio financeiro.

Sumário

Lista de Figuras e Tabelas	ix
Lista de Símbolos	xi
RESUMO	xiii
ABSTRACT	14
1 Capítulo I : Introdução	15
2 Capítulo II : Acústica Básica	21
2.1 Ondas Acústicas	21
2.1.1 Velocidade do Som nos Fluídios	22
2.1.2 Nível de Pressão Sonora e Intensidade Sonora	22
2.1.3 Propagação Sonora em Dutos Retilíneos	23
2.2 Efeitos do Ruído no Ser Humano	25
2.2.1 O Ouvido Humano	25
2.2.2 Curvas de Compensação	27
2.2.3 Perda de Audição e Efeito nos Sistemas Extra-auditivos	28
3 Capítulo III : Controle Ativo de Ruído	31
3.1 Sistema de Controle em Malha Aberta de Banda Larga	32
3.2 Sistema de Controle em Malha Aberta de Banda Estreita	35
3.3 Sistema de Controle em Malha Fechada	37
3.4 Sistema de Controle Híbrido	39
4 Capítulo IV : Algoritmos para Controle Ativo de Ruído	41
4.1 Algoritmos para Sistemas ANC em Malha Aberta de Banda Larga	42
4.1.1 Algoritmo Adaptativo LMS	43
4.1.2 Efeitos do Caminho Secundário	44
4.1.3 Algoritmo Adaptativo FXLMS	45
4.1.4 Efeitos da Realimentação Acústica	47
4.1.5 Algoritmo Adaptativo FBFXLMS	47
4.2 Algoritmos para Sistemas ANC em Malha Aberta de Banda Estreita	48
4.2.1 Método da Síntese de Ondas	49
4.2.2 Filtros Notch Adaptativos	50
4.3 Algoritmos para Sistemas ANC em Malha Fechada	53
4.4 Algoritmos para Sistemas ANC em Proposta Híbrida	55
4.5 Estimativa do Caminho Secundário	56
4.5.1 Estimativa <i>off-line</i> do caminho secundário	56
4.5.2 Estimativa <i>on-line</i> do caminho secundário	58
4.5.2.1 Método Direto (Algoritmo LMS)	58
4.5.2.2 Proposta de Fan	60
4.5.2.3 Proposta de Eriksson	61
4.5.2.4 Proposta de Bao	64
4.5.2.5 Proposta de Kuo	65
4.5.2.6 Proposta de Zhang	67
4.6 Modelagem do Caminho de Realimentação	70
4.6.1 Estimativa <i>off-line</i> do Caminho de Realimentação	70
4.6.2 Estimativa <i>on-line</i> do Caminho de Realimentação	71
4.6.2.1 Método Direto (Algoritmo LMS)	71
4.6.2.2 Algoritmo Adaptativo FURLMS	73
5 Capítulo V : Projeto de Controladores Ativos de Ruídos (ANC)	77
5.1 Projeto do Sistema	77

5.1.1 Seleção dos Transdutores Eletroacústicos	77
5.1.2 Causalidade	79
5.1.3 Taxa de Amostragem e Filtros Analógicos	80
5.1.4 Limitações Práticas e Soluções	82
5.2 Projeto do Software	83
5.2.1 Processadores Digitais de Sinais (DSPs)	83
5.2.2 Benefícios dos DSPs	85
5.2.3 Projetos Baseados em DSPs	85
5.2.4 Arquitetura dos DSPs	87
5.2.5 Fator de Vazamento “Leakage”	88
6 Capítulo VI : Avaliação das Metodologias de Controle Ativo de Ruído	89
6.1 A Bancada Experimental	89
6.2 Recursos utilizados pelas diferentes metodologias avaliadas	92
6.3 Testes com simples tons: Identificação <i>off-line</i>	93
6.4 Testes com simples tons: Identificação <i>on-line</i>	97
6.5 Testes com múltiplos tons	101
7 Capítulo VII : Conclusões e Sugestões	105
8 Referência Bibliográfica	109
I Anexo I : Método do Gradiente Descendente	115
I.1 Regra Delta de Widrow-Hoff	115
I.2 Passo Adaptativo	118
I.3 Algoritmo do Método da Máxima Descida	119
I.4 Exemplo	120
II Anexo II : DSP TMS320LF2407A	123
II.1 Arquitetura do TMS320LF2407A	123
II.1.1 Organização da Memória	124
II.1.2 Osciladores	125
II.1.3 Entradas e Saídas Digitais	126
II.1.4 Módulos administradores de eventos (EVA, EVB)	127
II.1.5 Conversores A/D	129
II.2 Aritmética de Ponto Fixo	130
II.3 Biblioteca de Processamento de Sinais da Texas Instruments	133
II.3.1 Filtro FIR genérico utilizando <i>buffer</i> linear	134
II.3.2 Filtro FIR genérico utilizando <i>buffer</i> circular	137
III Anexo III : Foto da Bancada Experimental	141
IV Anexo IV : Algoritmo ANC – Controle Híbrido	143
V Anexo V : Diagrama dos Circuitos Eletrônicos	153

Lista de Figuras e Tabelas

Figura 1.1 - Conceito físico de cancelamento ativo de ruído.....	16
Figura 2.1 - Propagação em Dutos Retilíneos.....	23
Figura 2.2 - O ouvido humano.....	26
Figura 2.3 – Curvas de Compensação A e C	28
Figura 3.1 - Controle Ativo de Ruído em malha aberta de banda larga	33
Figura 3.2 - Diagrama de blocos para um ANC malha aberta de banda larga.....	33
Figura 3.3 - Controle Ativo de Ruído em malha aberta de banda estreita.	35
Figura 3.4 - Diagrama de blocos para um ANC malha aberta de banda estreita.....	36
Figura 3.5 - Controle Ativo de Ruído em malha fechada.....	37
Figura 3.6 - Diagrama de blocos para um ANC em malha fechada.....	38
Figura 3.7 – Controle Ativo Híbrido de Ruído.....	39
Figura 4.1 - Fluxograma da organização do capítulo.....	41
Figura 4.2 - Diagrama de um ANC de banda larga.....	42
Figura 4.3 - Sistema ANC com caminho secundário $S(z)$	44
Figura 4.4 - Diagrama de blocos do algoritmo FXLMS.....	45
Figura 4.5 - Sistema ANC com cancelamento da realimentação acústica.....	48
Figura 4.6 - Controle Ativo de Ruído com Filtro <i>Notch</i> Adaptativo.....	51
Figura 4.7 - Múltiplos Filtros <i>Notch</i> Adaptativos.	53
Figura 4.8 - Diagrama de Blocos do Sistema Malha Fechada.....	54
Figura 4.9 - Sistema Híbrido de Controle Ativo de Ruído.....	56
Figura 4.10 - Esquema de modelagem <i>off-line</i> do caminho secundário.	57
Figura 4.11 - Identificação <i>on-line</i> do caminho secundário através da modelagem direta....	59
Figura 4.12 - Modelagem do caminho secundário através da proposta de H. Fan.....	60
Figura 4.13 - Estimativa <i>on-line</i> de $S(z)$ através da técnica de Eriksson.	62
Figura 4.14 - Estimativa <i>on-line</i> de $S(z)$ através da técnica de Bao.....	65
Figura 4.15 - Estimativa <i>on-line</i> de $S(z)$ através da técnica de Kuo.....	66
Figura 4.16 - Estimativa <i>on-line</i> de $S(z)$ através da técnica de Zhang.....	69
Figura 4.17 - Identificação <i>on-line</i> do caminho de retorno através da modelagem direta....	72
Figura 4.18 - Sistema ANC com algoritmo FURLMS.	74
Figura 5.1 - Montagem dos microfones para redução da turbulência.....	82
Figura 5.2 - Arquitetura de Von Neumann e Havard.	88
Figura 6.1 - Sistema Físico do Controle Ativo de Ruído.....	90
Figura 6.2 - Funções de Transferência dos Caminhos do Sistema Físico.	92
Figura 6.3 – Estimativas dos Caminhos através do MatLab®.....	94
Figura 6.4 – Controle FXLMS para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.	94
Figura 6.5 – Controle FBFXLMS para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.	95
Figura 6.6 – Controle Feedback para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.	95
Figura 6.7 – Controle Híbrido para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.....	96
Figura 6.8 – Controle Fan para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.....	98

Figura 6.9 – Controle Eriksson para a freqüência de 220Hz. (a) Sinal do Microfone de Erro.	98
(b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura 6.10 – Controle Zhang para a freqüência de 220Hz. (a) Sinal do Microfone de Erro.	99
(b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura 6.11 – Controle FXLMS multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de	101
Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura 6.12 – Controle FBFXLMS multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de	101
Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura 6.13 – Controle Feedback multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de	102
Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura 6.14 – Controle Híbrido multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de	102
Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura 6.15 – Controle Fan multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de	103
Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura 6.16 – Controle Eriksson multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de	103
Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura 6.17 – Controle Zhang multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de	104
Cancelamento. (c) Densidade Espectral de Potência do erro.....	
Figura I.1 - Rede com valor desejado D e erro ε	115
Figura I.2 - Minimização do erro quadrático.	117
Figura I.3 - Interpretação geométrica do método do gradiente.....	117
Figura I.4 - Resultados do exemplo. (a) Sinal desejado e saída do filtro FIR.....	121
Figura II.1 - Arquitetura do Dispositivo 240xA (SPU357B, 2001).	123
Figura II.2 - Mapa de memória do TMS320LF2407A (SPRS145G, 2002).....	126
Figura II.3 - Configuração dos pinos compartilhados (SPRS145G, 2002).	127
Figura II.4 - Diagrama de Blocos do EVA (SPRS145G, 2002).	128
Figura II.5 - Diagrama de blocos do módulo ADC (SPRS145G, 2002).	129
Figura II.6 - Rodas Numéricas (Chassaing <i>et al.</i> , 1990).....	131
Figura II.7 - Estrutura do filtro FIR.	135
Figura II.8 - Computação do filtro FIR com <i>buffer</i> linear.....	137
 Tabela 2.1 - Limites do NPS - Portaria 3124/1978.	29
Tabela 6.1 - Recursos utilizados pelo processador na implementação dos algoritmos	93
Tabela 6.2 - Ensaios experimentais dos algoritmos com identificação <i>off-line</i>	97
Tabela 6.3 - Ensaios experimentais dos algoritmos com identificação <i>on-line</i>	99
Tabela II.1 - Tabela das variáveis do objeto FIRFILT_GEN.	135
Tabela II.2 - Tabela das variáveis do objeto FIRFILT_CGEN.	138

Lista de Símbolos

σ^2	Estimativa da Potência
\hat{F}	Estimativa do Caminho de Realimentação
∂_E	Atraso elétrico
∂_W	Atraso de grupo do filtro digital $W(z)$.
∂_T	Atraso total na eletrônica analógica
∂_A	Atraso acústico entre o microfone de entrada e o alto-falante
ω	Freqüência Radial
α	Fator de esquecimento na estimativa da potência
μ	Passo adaptativo
τ	Atraso dado pelo arranjo alto-falante e microfone
BW	Banda de 3dB
C_0	Velocidade da som
d	Sinal da fonte de ruído no microfone de erro
dB	Decibel (Capítulo II)
e	Sinal de erro ou resíduo
f	Freqüência
$F(z)$	Função entre o atuador e o microfone de referência (Caminho de Realimentação)
FIR	Finite Impulse Response
F_s	Freqüência de Amostragem
$G(z)$	Função de transferência entre a fonte de ruído e o microfone de referência
I	Intensidade Acústica (Capítulo II)
k	Número de Onda Acústica
L	Distância entre o microfone e o alto-falante
LMS	Least Mean Square
n	Índice Temporal
NIS	Nível de Intensidade Acústica
NPS	Nível de Pressão Sonora
P	Pressão Sonora
$P(z)$	Função entre a fonte de ruído e o microfone de erro (Caminho Primário)

\hat{S}	Estimativa do Caminho Secundário
$S(z)$	Função entre o atuador e o microfone de erro (Caminho Secundário)
T	Temperatura (Capítulo II)
t	Tempo
T	Período de Amostragem
u	Sinal do Microfone de Referência
$W(z)$	Controlador – Filtro Digital Adaptativo que estima o Caminho Primário
x	Sinal de referência (entrada) do controlador
x'	Sinal de entrada filtrado
y	Sinal de saída do controlador
y, z	Coordenadas Cartesianas (Capítulo II)
z	Freqüência discreta (Transformada Z)
Δ	Atraso de tempo das amostras entre a saída do sinal do sintetizador e sua chegada ao microfone de erro

Delfino, L. C., 2005, **Controle Ativo de Ruído Utilizando Processadores Digitais de Sinais**, Dissertação de Mestrado, Universidade Federal de Uberlândia, Minas Gerais, Brasil.

RESUMO

Ruídos acústicos são conhecidos como fontes de poluição sonora que podem causar efeitos adversos na vida humana. Para solucionar estes problemas, interesse considerável tem sido mostrado em Controle Ativo de Ruído.

O intuito deste trabalho é estudar e analisar os principais algoritmos de Controle Ativo de Ruído presentes na literatura, incluindo algoritmos de malha aberta (*Feedforward*) e de malha fechada (*Feedback*), bem como um sistema híbrido que utilize os dois conceitos. Os problemas relacionados aos caminhos secundário e de realimentação são apresentados e algumas soluções são discutidas. Neste âmbito, metodologias de modelagem *off-line* e *on-line* são apresentadas.

Uma pequena introdução à acústica básica em dutos é apresentada e alguns efeitos que o ruído acústico pode causar ao ser humano são discutidos.

Uma discussão é realizada a respeito do arranjo físico do sistema, incluindo escolha e posicionamento dos transdutores eletroacústicos. Problemas de causalidade e do condicionamento de sinais também são discutidos. Uma introdução a respeito dos Processadores Digitais de Sinais (DSPs) e algumas particularidades encontradas durante o desenvolvimento deste trabalho são apresentadas.

Para validar a performance de alguns algoritmos de controle, montou-se uma bancada experimental constituída de um duto hidráulico de PVC utilizado como duto acústico, onde estes algoritmos foram implementados em linguagem C em uma plataforma DSP da Texas Instruments do tipo TMS320LF240A. Uma análise é realizada com respeito à dificuldade e recursos utilizados por cada algoritmo implementado. Resultados e discussões são apresentados com respeito à performance dos sistemas de controle.

Delfino, L. C., 2005, **Active Noise Control Using Digital Signal Processors**, M. Sc. Dissertation, Universidade Federal de Uberlândia, Minas Gerais, Brazil.

ABSTRACT

Acoustical noises are known as pollution sources that cause adverse effects in human life. Considerable investigations have been done to development of the new technologies in Active Noise Control.

This work presents and experimentally analyses algorithms of Active Noise Control in Ducts presented in literature, including Feedforward algorithms, Feedback algorithms and Hybrid algorithms that uses both concepts. The identification of secondary path and feedback path is presented and solutions are discussed. In this way, methods of off-line and on-line modeling are presented.

A short introduction about acoustics in ducts is presented and some effects that the acoustical noise can cause in human being are also discussed.

Problems about the sensors and actuators displacement in the duct system, causality and signal conditioning are also argued here. An introduction about Digital Signal Processors (DSPs) e some particularities found in the development of this works are presented.

In order to evaluate the control algorithms performance, an experimental acoustic duct using a standard PVC water pipe was built, where those algorithms were implemented in a DSP platform TMS320LF240A from Texas Instruments®. An analysis is done about the difficulty and recourses used for which algorithm implemented. This work ends presenting and discussing the obtained results for the different control procedures studied and pointing to some future works.

Key Words: Active Noise Control, Digital Signal Processing, DSP, LMS.

CAPÍTULO I

Introdução

Ruídos acústicos são reconhecidos como fontes de poluição sonora que podem causar efeitos adversos na vida humana, uma vez que além de interferirem na performance de trabalhos que exigem concentração e comunicação sonora têm, em algumas situações de exposição indevida, provocado sérios transtornos psicológicos e/ou fisiológicos (Tokhi et al., 1992).

Problemas de ruído acústico no meio-ambiente tornaram-se mais observáveis nas últimas décadas devido ao aumento no número de equipamentos industriais utilizados; ao crescimento da densidade populacional que aumentou a exposição da população ao ruído e ao uso de materiais leves para construção e fabricação de equipamentos, resultado das limitações nos custos.

São conhecidos alguns tipos de ruído acústico no meio-ambiente. O primeiro é causado por turbulência, sendo totalmente aleatório. Este tipo de ruído distribui a energia igualmente ao longo das bandas de freqüência, sendo conhecido como ruído de banda larga. Exemplos são sons de baixa freqüência de aviões a jato e o ruído impulsivo de uma explosão. Outro tipo, chamado de ruído de banda estreita, concentra a maior parte da energia em freqüências específicas. Este tipo de ruído pode ser encontrado em máquinas rotativas ou repetitivas, contendo características periódicas ou quase periódicas (Tokhi et al., 1992).

Há duas formas de se atenuar o ruído acústico. A forma tradicional utiliza técnicas passivas tais como cercas, barreiras e silenciadores. Normalmente, os silenciadores são classificados como passivos, resistivos ou reativos. Os silenciadores passivos usam o conceito de mudança da impedância para atenuar o ruído, enquanto que os resistivos utilizam o conceito de perda de energia causada pela propagação do som em dutos alinhados com materiais absorventes. Já os silenciadores reativos utilizam a combinação de diafragmas e tubos para atenuar o som indesejado. Os reativos são comumente utilizados como escapamentos em máquinas de combustão interna, enquanto que os resistivos são

utilizados em dutos de ventilação. Estes silenciadores são eficientes em freqüências altas entretanto, são volumosos, caros e ineficientes em baixas freqüências.

Para solucionar o problema de ruído em baixa freqüência, interesse considerável tem sido observado na área de controle ativo de ruídos. O sistema de controle ativo de ruído contém um dispositivo eletroacústico que atenua o som indesejado pela geração de uma onda sonora (anti-ruído) de igual amplitude e fase oposta. O ruído primário, indesejado, e o anti-ruído se combinam acusticamente, resultando no cancelamento de ambos. A Figura 1.1 mostra as formas de ondas do ruído indesejado, do anti-ruído e do ruído residual quando eles se superpõem. O desempenho do cancelamento do ruído primário depende da precisão em amplitude e fase do anti-ruído gerado.

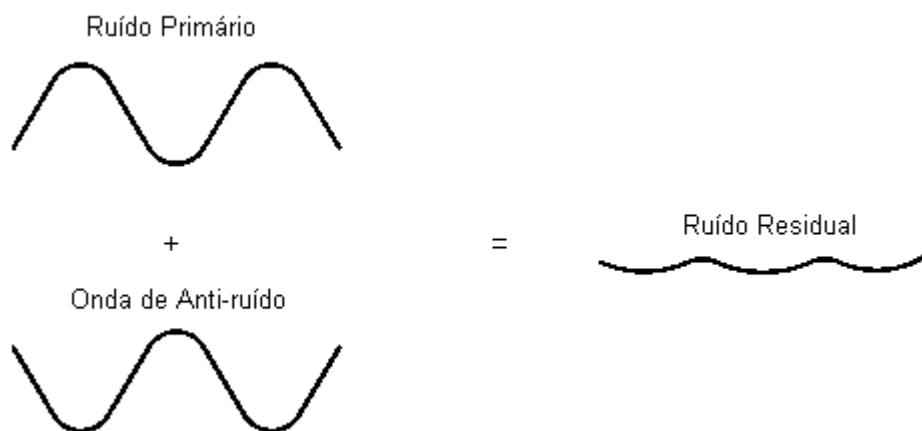


Figura 1.1 - Conceito físico de cancelamento ativo de ruído.

O controle ativo de ruído foi uma das primeiras aplicações eletrônicas de controle aplicadas a sistemas físicos. Lueg registrou a patente No. 2043416 em 1936 (Lueg, 1936). O autor fez uso de dois princípios básicos da acústica para idealizar um sistema ANC: interferência e absorção. O princípio da interferência resulta da combinação de ondas acústicas levando a interferências construtivas e destrutivas, que causam intensificação e atenuação do campo sonoro, respectivamente.

O autor manipulou o princípio da superposição de tal forma que o anti-ruído gerado produzia uma interferência destrutiva das ondas sonoras, eliminando o ruído indesejado. Ele introduziu o conceito da atenuação ativa do som pela geração artificial de ondas deslocadas em fase de 180° . Lueg também utilizou o princípio da absorção pela sincronia dos movimentos do diafragma de um alto-falante defasados de 180° em relação ao ruído indesejado. O esquema proporcionava a absorção da energia do ruído pelo alto-falante. Lueg ilustrou o fenômeno físico que provê a possibilidade do controle ativo de ruído da seguinte maneira (Lueg, 1936):

“Uma onda acústica com freqüência específica tem velocidade relativamente menor que um sinal elétrico de mesma freqüência. Isto implica que, enquanto a onda sonora está viajando do ponto onde foi detectada até o ponto onde será atenuada, há tempo suficiente para o circuito eletrônico processar o sinal e ativar os elementos de controle, em um grau maior ou menor, dependendo da freqüência, tipo de ruído e características físicas do sistema”.

Somente após vinte anos do desenvolvimento de Lueg, o interesse pelo controle ativo de ruído voltou a ser discutido por Olson. Este introduziu o absorvedor acústico eletrônico (Olson *et al.*, 1953). Em seu artigo, o autor propôs a localização de redutores de som para ocupantes de veículos e operadores de máquinas. A onda sonora era detectada por um microfone e passada através de um amplificador. O amplificador, então, controlava o alto-falante de tal forma que a pressão sonora na região próxima ao microfone era reduzida. A estrutura de *feedback* do dispositivo de Olson foi uma grande contribuição, porém sua utilização possuía várias limitações práticas.

Nos anos cinqüenta, Simshauser e Hawley (1955) propuseram o desenvolvimento de um defensor ativo de ouvido. O dispositivo reduzia o ruído do ambiente no ouvido pelo uso de um capacete que gerava uma pressão sonora igual em magnitude e com fase oposta ao ruído. O dispositivo desenvolvido era um sistema de dois canais, consistindo de dois microfones e dois alto-falantes montados em ambos fones de ouvido de um capacete militar convencional, com um amplificador e rede de deslocamento de fase para cada canal. Atualmente esta é uma das áreas de aplicações do controle ativo de ruído de maior sucesso (Chaplin *et al.*, 1987).

Conover (1957) da *General Electric* discutiu o controle ativo do ruído em transformadores. Seu trabalho foi baseado em um transformador de 15MVA. O autor não confinou seu trabalho ao laboratório, desenvolvendo resultados expressivos no campo. O esquema de Conover foi localizar alto-falantes próximos à superfície do transformador, cancelando a pressão sonora nas proximidades. A vibração do transformador produzia um espectro periódico com harmônicos pares múltiplos da freqüência da rede. Conover foi o primeiro a tratar do assunto, desde então outros autores investigaram o problema.

O interesse acelerado pelo controle ativo de ruídos iniciou-se nos anos sessenta com publicações de Jessel e co-autores na França, e Kido e associados no Japão. O trabalho de Jessel foi concentrado em ruídos em dutos (Jessel *et al.*, 1972). A contribuição de Jessel foi largamente baseada no Princípio de Huygens¹. O trabalho de Kido (1975) foi concentrado

¹ O princípio de Huygens é uma construção puramente geométrica que permite determinar a posição futura de uma frente de onda a partir de sua posição em um instante dado.

em ruídos de transformadores. Jessel e Kido descobriram que a principal vantagem de controladores ativos é a sua habilidade de atenuar ruídos de baixa freqüência.

Jessel e outros (Kuo *et al.*, 1996) também descobriram alguns problemas associados com a redução de ruídos em dutos. Estes ofereciam realimentação acústica devido aos componentes refletidos, tendendo a confundir os controladores, uma vez que o microfone de referência não conseguia distinguir o ruído primário dos componentes refletidos. Isto levava o sistema à instabilidade ou a baixa atenuação em algumas bandas de freqüências. Há considerável pesquisa voltada ao ruído em dutos desde a patente de Lueg e do trabalho de Jessel (Kuo *et al.*, 1996; Potta *et. al.*, 2000).

Por causa do fato das características da fonte de ruído acústico e do ambiente não serem constantes, estes tipos de sistemas são ditos fortemente não-estacionários (variantes no tempo). Um sistema de controle deveria então ser adaptativo para cobrir as mudanças nas características da planta.

Para compensar as variações ocasionadas pelas características não-lineares e não-estacionárias do sistema, foram propostos os controladores automaticamente adaptáveis. Tais controladores são filtros adaptativos que possuem a propriedade de modificarem a função de transferência ao longo do tempo através da mudança de seus coeficientes (Goodwin *et al.*, 1984; Clarkson, 1993). Nestes procedimentos os filtros mais comumente usados são os do tipo resposta finita ao impulso ("Finite Impulse Response - FIR") e resposta infinita ao impulso ("Infinite Impulse Response - IIR"). O mecanismo mais comumente usado no ajuste dos coeficientes dos filtros é o algoritmo LMS – (*Least Mean Square*) (Widrow, 1985).

Os filtros FIR possuem apenas zeros em sua equação característica, portanto são filtros totalmente estáveis. No entanto, para modelar o sistema acústico, estes devem possuir um grande número de coeficientes, o que pode proporcionar um grande tempo de processamento no sistema de adaptação. Já os filtros IIR possuem pólos e zeros. Devido aos pólos podem se tornar instáveis durante o processo de adaptação, ou mesmo na quantização dos coeficientes. São filtros com pequeno número de coeficientes. Um sistema de cancelamento de ruídos em dutos baseado em filtros adaptativos foi desenvolvido por Burgess (1981).

Embora as potencialidades do controle ativo do ruído tenham sido demonstradas há várias décadas, havia grandes dificuldades de realização devido às limitações técnicas dos processadores digitais para realizarem os cálculos do controlador, particularmente para sistemas que alterassem suas características rapidamente.

Toda a idéia dos controladores ativos de ruído se baseia no Processamento Digital de Sinais (*Digital Signal Processing - DSP*), onde cálculos de convoluções são exigidos a

todo momento. Estes tipos de cálculos são realizados em processadores específicos, pois o processamento em tempo real é uma característica inerente destes controladores.

As implementações iniciais de DSPs em grandes computadores não eram muito práticas, contudo a disponibilidade de processadores digitais de sinais de baixo custo a partir da década de 80 causou uma revolução na área. Deste modo, os DSPs embarcados num único chip estão mudando o processamento digital de sinais da mesma maneira que os microprocessadores mudaram o processamento de dados.

O primeiro DSP em um único chip foi lançado pela *Texas Instruments* em 1982. Desde já estes possibilitaram a implementação de poderosos algoritmos adaptativos a um baixo custo (Kuo *et al.*, 1996) e encorajaram inúmeros desenvolvimentos e aplicações na área (Eriksson *et al.*, 1987).

Nesta linha este trabalho procura estudar as diferentes e mais promissoras técnicas de controle ativo de ruído em dutos, discutir suas vantagens e desvantagens e avaliar experimentalmente algumas delas quando implementadas em um processador digital de sinais.

Para atender a estes objetivos esta dissertação está dividida como se segue. O Capítulo seguinte introduz a teoria básica de acústica envolvida no cancelamento ativo de ruído, bem como os efeitos danosos aos seres humanos devido à exposição excessiva ao ruído acústico. O Capítulo III trata das principais estratégias presentes na literatura atual, com potencialidades de implementação em DSPs, para se realizar o controle adaptativo. Já o Capítulo IV apresenta alguns algoritmos utilizados para o cancelamento e discute os problemas dos efeitos do caminho secundário e de realimentação. O Capítulo V trata de alguns problemas a respeito da instrumentação e ao aparato físico envolvendo o controle ativo de ruído. A discussão a respeito da plataforma DSP também é realizada neste mesmo capítulo. O Capítulo VI trata da avaliação de algumas metodologias estudadas, a fim de avaliar a performance e capacidade das mesmas. Por fim, o Capítulo VII apresenta as devidas conclusões a respeito do trabalho, bem como futuros desdobramentos.

CAPÍTULO II

Acústica Básica

O som se caracteriza por flutuações de pressão em um meio compressível (Samir, 2000). No entanto, não são todas as flutuações de pressão que produzem a sensação de audição quando atingem o ouvido humano. Esta sensação só ocorre quando a amplitude e freqüência destas flutuações estiverem dentro de determinada faixa de valores. Flutuações com amplitudes inferiores a certos valores não são audíveis (limiar de audição), no entanto, ondas de alto nível, tais como nas proximidades de turbinas a gás, podem produzir sensação de dor ao invés de som. Ainda, há ondas cujas freqüências de repetição estão fora da faixa de geração de sensação auditiva, que no ser humano esta vai de 20Hz a 20KHz.

Estes valores sofrem variações fisiológicas entre as pessoas e ainda podem sofrer influência de fatores subjetivos psicológicos. As respostas subjetivas de humanos incluem percepção, sentimento de desconforto, apreensão e dor. Como vários fatores estão envolvidos, como amplitude, freqüência, modo de aplicação, duração e a situação da pessoa, os resultados são de difícil caracterização.

Neste capítulo serão apresentados alguns conceitos físicos de interesse em acústica, tais como pressão e intensidade sonora, velocidade do som e escala decibel. Os efeitos do ruído no ser humano serão abordados de forma concisa.

2.1 Ondas Acústicas

Na prática, a geração do som é causada pela variação da pressão ou da velocidade das moléculas no meio. O som é uma forma de energia que é transmitida pelas sucessivas colisões das moléculas no meio, umas contra as outras. Portanto, o som pode ser representado por uma série de compressões e rarefações do meio em que se propaga.

É importante ficar claro que não há deslocamento permanente de moléculas, ou seja, não há transferência de matéria, apenas de energia (com exceção nas proximidades da fonte sonora).

Uma pequena variação de pressão acústica é suficiente para produzir um ruído desconfortável (aproximadamente 0,1 milibar). Por outro lado, a sensibilidade do ouvido é tal que, uma variação de pressão de 2×10^{-7} milibar pode ser detectada, caso a freqüência de oscilação esteja na faixa mais sensível de audição, que situa-se na escala de 1KHz a 4KHz (Samir, 2000).

2.1.1 Velocidade do Som nos Fluídos

As ondas acústicas se propagam através do meio e sua velocidade, supondo um modelo simplificado, depende apenas da temperatura do meio (Samir, 2000). Uma fórmula aproximada para determinação da velocidade do som no ar, em uma temperatura T ($^{\circ}\text{C}$) é:

$$C_0 = 331,4 + 0,6T \text{ [m/s]} \quad (2.1)$$

Para o ar a 25°C , a velocidade do som é de 346,4m/s.

2.1.2 Nível de Pressão Sonora e Intensidade Sonora

O ouvido humano responde a uma larga faixa de intensidade acústica, desde o limiar de audição até o limiar da dor. Por exemplo, a 1000Hz a intensidade acústica capaz de causar dor é 10^{12} vezes a intensidade capaz de causar sensação de audição. Devido a dificuldade de se expressar números de ordens de grandeza tão diferentes numa mesma escala linear, adota-se a escala logarítmica. O nível de intensidade acústica N/S é dado por:

$$N/S = 10 \log \frac{I}{I_0}, \text{ com } I_0 = 10^{-12} \text{ Watt/m}^2 \quad (2.2)$$

Onde I_0 corresponde, aproximadamente, a intensidade de um tom de 1000Hz, que é levemente audível pelo ouvido humano normal (valor de referência).

A intensidade acústica é proporcional ao quadrado da pressão acústica, assim o nível de pressão sonora é dado por:

$$NPS = 20 \log \frac{P}{P_0}, \text{ com } P_0 = 2 \times 10^{-5} \text{ N/m}^2 \quad (2.3)$$

Onde P_0 é o valor de referência correspondente ao limiar de audição em 1000Hz.

Outro aspecto importante da escala dB é que ela apresenta uma correlação melhor com a audibilidade humana. Um (1) dB é a menor variação que o ouvido humano pode perceber. Uma redução de 6dB no nível de pressão sonora equivale a reduzir pela metade a pressão sonora na escala linear.

2.1.3 Propagação Sonora em Dutos Retilíneos

Considere um duto retilíneo infinito com paredes internas rígidas (sem revestimento com materiais de absorção). Se o duto for excitado por uma fonte sonora em uma das extremidades, as ondas propagadas obedecem à equação da onda acústica (Samir, 2000). Considerando o problema em duas dimensões y e z , conforme a Figura 2.1, o campo sonoro obedece a seguinte equação diferencial:

$$\frac{\partial^2 P}{\partial y^2} + \frac{\partial^2 P}{\partial z^2} - \frac{1}{c^2} \frac{\partial^2 P}{\partial t^2} = 0 \quad (2.4)$$

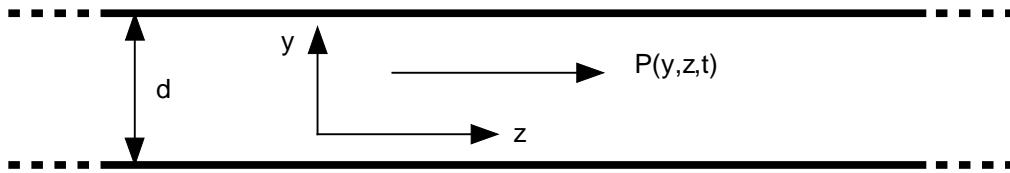


Figura 2.1 - Propagação em Dutos Retilíneos.

As condições de contorno de velocidade nula da partícula nas paredes rígidas são dadas por:

$$\frac{\partial P}{\partial y} = 0 \text{ em } y = 0 \text{ e } y = d \quad (2.5)$$

Considerando a solução da equação da onda com variação harmônica simples no tempo, tem-se:

$$P(y, z, t) = P(y, z) e^{i\omega t} \quad (2.6)$$

Substituindo (2.6) em (2.4), resulta:

$$\frac{\partial^2 P}{\partial y^2} + \frac{\partial^2 P}{\partial z^2} + k^2 P = 0 \quad (2.7)$$

onde $k = \omega / c$ é o número de onda acústica.

Utilizando o método de separação de variáveis, pode-se escrever:

$$P(y, z) = P_1(y)P_2(z) = A_1 e^{\lambda_1 y} e^{\lambda_2 z} \quad (2.8)$$

Substituindo (2.8) em (2.7), chega-se:

$$(\lambda_1^2 + \lambda_2^2 + k^2)A_1 = 0 \quad (2.9)$$

A solução não-trivial é dada por:

$$\lambda_1^2 + \lambda_2^2 = -k^2 \quad (2.10)$$

ou

$$\lambda_1 = \pm i\sqrt{\lambda_2^2 + k^2} \quad (2.11)$$

Substituindo (2.11) em (2.8) e (2.6):

$$P(y, z, t) = e^{i\omega t} \left(A e^{\lambda_2 z} e^{iy\sqrt{k^2 + \lambda_2^2}} + B e^{\lambda_2 z} e^{-iy\sqrt{k^2 + \lambda_2^2}} \right) \quad (2.12)$$

As condições de contorno requerem:

$$\frac{\partial P}{\partial y} = e^{i\omega t} \left(iA\sqrt{k^2 + \lambda_2^2} e^{iy\sqrt{k^2 + \lambda_2^2}} - iB\sqrt{k^2 + \lambda_2^2} e^{-iy\sqrt{k^2 + \lambda_2^2}} \right) \quad (2.13)$$

Aplicando a condição em $y=0$, tem-se $A=B$, então:

$$P(y, z, t) = 2A e^{i\omega t} e^{\lambda_2 z} \cos\left(y\sqrt{k^2 + \lambda_2^2}\right) \quad (2.14)$$

Aplicando a condição em $y=d$, tem-se:

$$d\sqrt{k^2 + \lambda_2^2} = n\pi \quad n=0, 1, 2, 3\dots \quad (2.15)$$

Então:

$$\lambda_1 = \pm i\left(\frac{n\pi}{d}\right) \quad (2.16)$$

e

$$\lambda_2 = \pm i\sqrt{k^2 - \left(\frac{n\pi}{d}\right)^2} \quad (2.17)$$

Assim, a solução geral fica:

$$P(y, z, t) = 2A e^{i\omega t} \cos\left[\left(\frac{n\pi}{d}\right)y\right] e^{\pm iz\sqrt{k^2 - \left(\frac{n\pi}{d}\right)^2}} \quad (2.18)$$

Cada modo se propaga na direção z com um número de onda dado por:

$$k_n = \sqrt{k^2 - \left(\frac{n\pi}{d}\right)^2} = \frac{1}{c} \sqrt{\omega^2 - \left(\frac{n\pi c}{d}\right)^2} \quad (2.19)$$

Então, o modo n pode se propagar somente se k_n for um número real, isto é:

$$f_n > \frac{nc}{2d} \text{ [Hz]} \quad (2.20)$$

A freqüência acima é chamada de freqüência de corte do modo n .

Em um duto uniforme de seção transversal circular de diâmetro d , considerando a propagação em ambientes cilíndricos, a mais baixa freqüência de corte corresponde é dada por (Samir, 2000):

$$f = \frac{1.84c}{2\pi d} \quad (2.21)$$

A implicação prática destes resultados é que em um duto uniforme de paredes rígidas, somente ondas planas podem se propagar abaixo da menor freqüência de corte. Isto é importante para o controle ativo de ruído, pois o mesmo é mais efetivo se aplicado a ondas planas (Kuo et al., 1997).

2.2 Efeitos do Ruído no Ser Humano

A exposição prolongada ao ruído pode causar deterioração do sistema auditivo, desta forma se faz necessário o conhecimento do funcionamento e do comportamento deste sistema. Também é importante conhecer os efeitos provocados pelo ruído no corpo humano.

2.2.1 O Ouvido Humano

O ouvido humano permite perceber e interpretar o som. Este processo é complexo e ainda não é completamente conhecido. O ouvido pode ser dividido em três partes: o ouvido externo, o médio e o interno, como mostra a Figura 2.2 (Marsh, 1999).

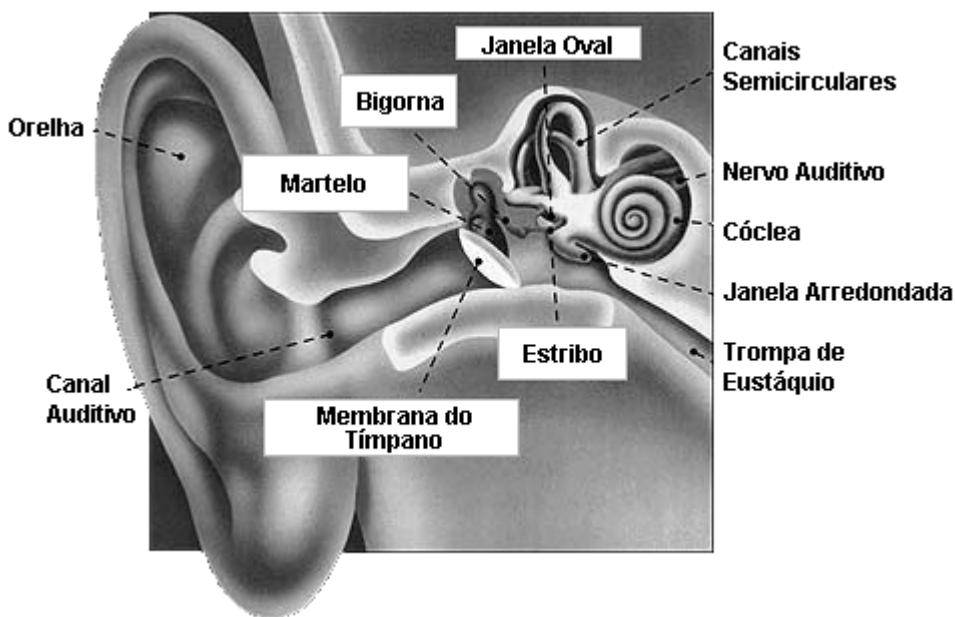


Figura 2.2 - O ouvido humano.

O ouvido externo consiste da orelha e um canal de aproximadamente 2cm. A orelha serve para proteger o ouvido médio e prevenir danos ao tímpano. A orelha também canaliza as ondas que alcançam o ouvido. Devido à geometria do canal, ele é capaz de amplificar os sons com freqüências até aproximadamente 3000Hz. À medida que o som propaga através do ouvido externo, o som ainda está na forma de uma onda de pressão acústica. Somente quando o som alcança o tímpano, na separação do ouvido externo e médio, a energia da onda é convertida em vibrações na estrutura óssea do ouvido. Uma compressão tensiona o tímpano para dentro e a rarefação o tensiona para fora. Logo, o tímpano vibra com a mesma freqüência da onda de entrada.

O ouvido médio é uma cavidade cheia de ar, consistindo de três pequenos ossos interconectados - o martelo, a bigorna e o estribo. A vibração do tímpano coloca o martelo, a bigorna, e o estribo em movimento com a mesma freqüência da onda. O estribo é conectado ao ouvido interno. Assim, as vibrações do estribo são transmitidas ao fluido do ouvido médio e criam uma onda de compressão dentro do fluido. Os três pequenos ossos do ouvido médio agem como amplificadores das vibrações da onda sonora.

Devido à vantagem mecânica, os deslocamentos da bigorna são maiores do que a do martelo. Além disso, como a onda de pressão que atinge uma grande área do tímpano é concentrada em uma área menor na bigorna, a força da bigorna vibrante é aproximadamente 15 vezes maior do que aquela do tímpano. Esta característica aumenta a possibilidade de ouvir o mais fraco dos sons. O ouvido médio é conectado à Trompa de Eustáquio e à boca. Esta conexão permite a equalização da pressão das cavidades cheias de ar do ouvido.

O ouvido interno consiste de uma cóclea, canais semicirculares, e do nervo auditivo. A cóclea e os canais semicirculares são cheios de líquido. O líquido e as células nervosas dos canais semicirculares não têm função na audição; eles simplesmente servem como sensores para detectar movimentos acelerados e na manutenção do equilíbrio do corpo.

A cóclea é um órgão em forma de espiral que possui um comprimento aproximado de 3cm. Além de estar cheio de fluído, a superfície interna da cóclea está alinhada com cerca de 20.000 células nervosas responsáveis pelas funções mais críticas na nossa capacidade de ouvir. Estas células nervosas possuem comprimentos diferentes, por diferenças minúsculas; eles também possuem diferentes graus de elasticidade no fluido que passa sobre eles. À medida que uma onda de compressão se move da interface entre o martelo do ouvido médio para a janela oval do ouvido interno através da cóclea, as células nervosas na forma de cabelos entram em movimento. Cada célula capilar possui uma sensibilidade natural a uma freqüência de vibração particular. Quando a freqüência da onda de compressão excita a freqüência natural de uma determinada célula nervosa, esta irá ressoar com uma grande amplitude de vibração. Esta vibração ressonante induz a célula a liberar um impulso elétrico que passa ao longo do nervo auditivo para o cérebro. Em um processo que ainda não é compreendido inteiramente, o cérebro é capaz de interpretar as qualidades do som pela reação dos impulsos nervosos. Maiores detalhes sobre o funcionamento do ouvido humano podem ser encontrados em Marsh, 1999.

2.2.2 Curvas de Compensação

O ouvido humano é menos sensitivo a freqüências muito baixas e muito altas. Para compensar este efeito, filtros são aplicados nas medidas sonoras. Estes filtros também são conhecidos como curvas de compensação. A mais comumente utilizada é a escala "A", que provê resultados, muito próximos ao do ouvido humano, denotados dB(A). A curva "C" também é bastante utilizada, principalmente para níveis muito altos ou para baixas freqüências.

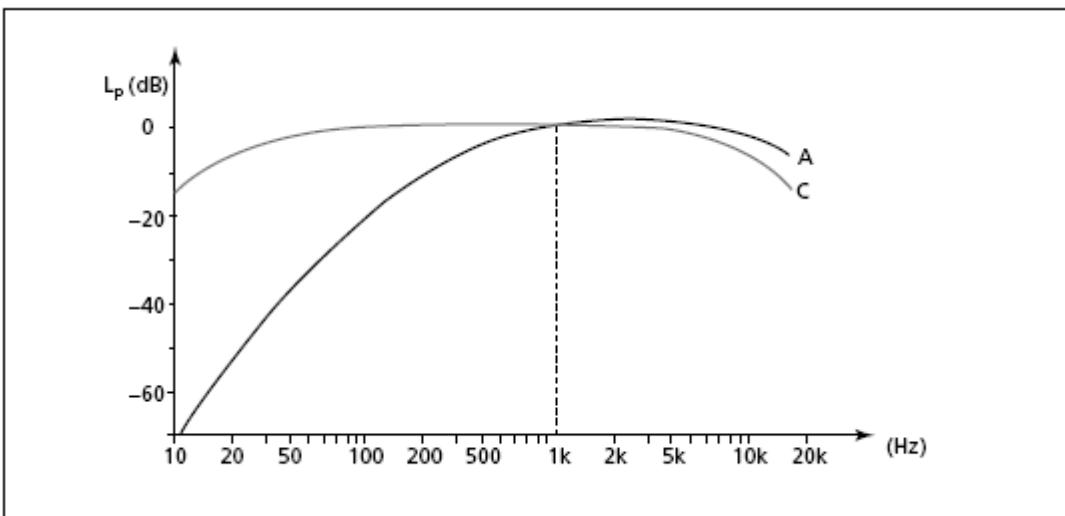


Figura 2.3 – Curvas de Compensação A e C.

2.2.3 Perda de Audição e Efeito nos Sistemas Extra-auditivos

Qualquer redução na sensibilidade de audição é considerada perda de audição. A exposição a níveis altos de ruído por um longo período danifica as células da cóclea. O tímpano, por sua vez, raramente é danificado por ruído industrial. O primeiro efeito fisiológico de exposição a níveis altos de ruído é a perda de audição na banda de freqüências de 4KHz a 6Khz. Geralmente, o efeito é acompanhado pela sensação de percepção do ruído após o afastamento do campo ruidoso. Se este efeito é temporário, o nível original do limiar de audição é recuperado. No entanto, se a exposição é repetida antes da completa recuperação, a perda temporária pode se tornar permanente. Quando as células nervosas no ouvido interno são danificadas, o processo da perda de audição é irreversível (Samir, 2000).

São conhecidos sérios problemas causados pelo ruído no corpo humano, tais como: aceleração da pulsação, aumento da pressão sanguínea e estreitamento dos vasos sanguíneos. Um longo tempo de exposição a ruído pode causar sobrecarga do coração causando secreções anormais de hormônios e tensões musculares. O efeito destas alterações aparece na forma de mudanças de comportamento, tais como: nervosismo, fadiga mental, frustração, prejuízo no desempenho do trabalho, entre outras.

O potencial de danos devido ao ruído depende não somente de seu nível, mas também de sua duração. A Portaria Brasileira 3.214 de 08/06/1978 define valores máximos ao que o ser humano pode ser exposto. Alguns destes limites podem ser vistos na Tabela 2.1.

Tabela 2.1 - Limites do NPS - Portaria 3124/1978.

NPS dB (A)	Máxima exposição diária
85	08 horas
86	07 horas
87	06 horas
88	05 horas
89	04 horas e 30 minutos
90	04 horas
91	03 horas e 30 minutos
92	03 horas
93	02 horas e 30 minutos
94	02 horas e 15 minutos
95	02 horas
96	01 hora e 45 minutos
98	01 hora e 15 minutos
100	01 hora

CAPÍTULO III

Controle Ativo de Ruído

Desde a proposta de Lueg em 1936, as metodologias de controle ativo de ruído se subdividiram em três categorias: a metodologia em malha aberta *feedforward*, a metodologia em malha fechada *feedback* e a metodologia híbrida. A escolha correta depende de vários fatores envolvidos no problema, tais como: disponibilidade de uma referência não-acústica, tamanho do sistema físico, banda de controle, dentre outros (Bai *et al.*, 2000).

Se sinais não-acústicos do ruído primário estão disponíveis, as estruturas *feedforward* e híbrida serão eficientes (Kuo *et al.*, 1996). Mesmo quando está referência não-acústica estiver indisponível, as estruturas *feedforward* e híbrida têm sido amplamente utilizadas como alternativa, onde um sensor acústico é utilizado como sinal de referência. Se o sistema é pequeno, tal como um protetor auricular ativo, a estrutura *feedback* pode ser uma alternativa viável (Bai *et al.*, 1997). Se o problema não é tão simples, uma aplicação direta do controle *feedback* pode garantir atenuação apenas em uma banda estreita de freqüências (Bai *et al.*, 1998).

Para ruídos de banda estreita, geralmente ruídos periódicos causados por máquinas rotativas, sinais não-acústicos podem ser utilizados como referência. Ao invés de se utilizar um microfone de entrada, acelerômetros ou tacômetros são utilizados para prover informações sobre a fonte geradora de ruídos. O sistema de controle pode estimar a grande quantidade de harmônicos presentes no ruído, oriundos da freqüência de rotação da máquina, e promover o cancelamento através do sinal de anti-ruído. Este tipo de controle é desejável em cabines de veículos, pois este não afetaria os alarmes do veículo, a performance do sistema de áudio ou a comunicação entre os ocupantes.

Um sistema de controle ativo de ruído é constituído dos seguintes elementos:

- **Sensores:** Microfones, acelerômetros, tacômetros ou outros dispositivos capazes de oferecer referências dos níveis de ruído a cancelar.
- **Atuadores:** Dispositivos eletroacústicos, através dos quais se modifica o campo acústico com a finalidade de obter uma diminuição da pressão acústica. Podem ser atuadores eletromecânicos.

- **Planta:** Ambiente acústico a controlar. Exemplos clássicos são ondas planas acústicas em dutos, o ruído no interior de um veículo, etc.
- **Controlador:** Sistema eletrônico que processa os sinais dos sensores e mediante algum conhecimento prévio da planta, gera um sinal que será transformado em ondas acústicas pelos atuadores.

No controle de malha aberta *feedforward*, o ruído de referência é sentido antes de se propagar através do alto-falante de cancelamento. No controle de malha fechada *feedback*, o controlador cancela o ruído sem nenhum benefício da informação anterior da entrada de referência (Widrow *et al.*, 1985).

A seguir será feita uma descrição mais detalhada dos tipos de sistemas de controle ativo de ruído, tanto para banda larga, como para banda estreita. Uma atenção especial será dada ao controle de malha aberta para banda larga e ao controle de malha fechada, já que estes esquemas serão desenvolvidos no cancelamento de ruídos em dutos mais adiante.

3.1 Sistema de Controle em Malha Aberta de Banda Larga

Uma quantidade considerável de ruído de banda larga é produzida em dutos, tais como sistemas de ventilação e exaustão. Um simples esquema de sistema de controle em malha aberta de banda larga (*Broadband Feedforward ANC*) para um duto pode ser visto na Figura 3.1. Quando se dispõe de um sinal que serve como referência do ruído que se deseja cancelar, diz-se que o sistema é de malha aberta.

Nesta configuração o sensor de referência mede o sinal indesejado em um ponto anterior e então alimenta um controlador para se obter o cancelamento do ruído em um ponto posterior. Em outras palavras, um sinal de referência $x(n)$ é medido por um microfone próximo à fonte de ruído antes que este passe pelo alto-falante. O controlador digital utiliza o sinal de referência para gerar um sinal $y(n)$ de igual amplitude e fase oposta (180°). O sinal de anti-ruído é utilizado para guiar o alto-falante para produzir um som de cancelamento que atenua o ruído acústico primário no duto.

O princípio básico do sistema de controle em malha aberta é que atraso de propagação da onda sonora entre o microfone de referência e o alto-falante ofereça tempo suficiente de se calcular o controle e ativar o anti-ruído, causando, assim, o cancelamento (Lueg, 1936).

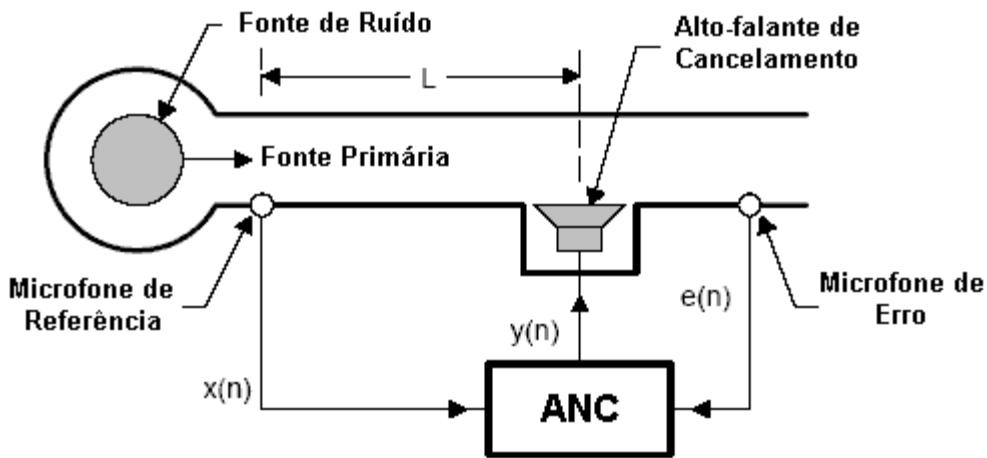


Figura 3.1 - Controle Ativo de Ruído em malha aberta de banda larga.

A distância entre o microfone de referência e o alto-falante de cancelamento deve satisfazer o princípio da causalidade. Isto significa dizer que o atraso do sinal acústico entre o microfone de referência e o alto-falante deve ser maior ou no mínimo igual ao atraso entre a fonte de ruído e o microfone de referência somado ao atraso entre o alto-falante e o microfone de erro. De outra forma, a causalidade significa que o sinal de referência deve ser medido de tal forma que exista tempo para se gerar o anti-ruído antes que o ruído chegue ao alto-falante de cancelamento. Além dos atrasos acústicos, deve-se também considerar o atraso gerado pelos cálculos realizados pelo controlador digital (Nelson, 1992).

A principal tarefa do controlador é se antecipar ao fenômeno físico, de forma preditiva, através das informações captadas pelo sensor de referência, de modo que o ruído seja cancelado no sensor de erro. Assim, o sinal de referência medido pelo sensor de entrada e o sinal proveniente do sensor de erro devem possuir alta correlação.

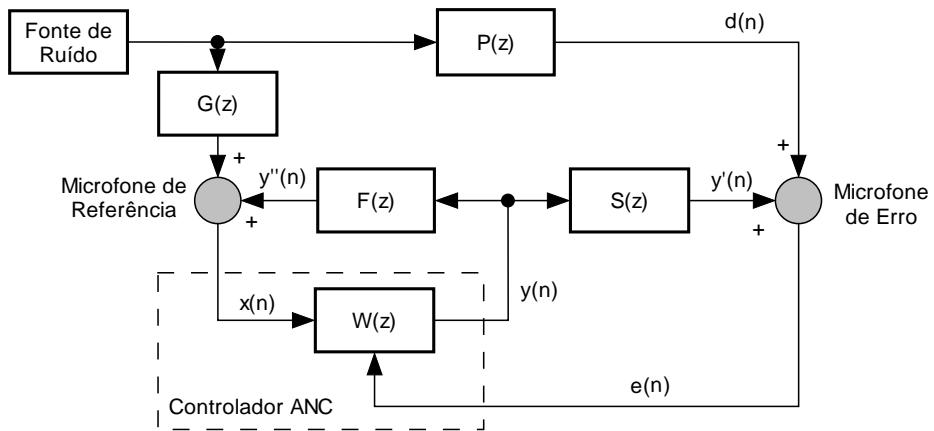


Figura 3.2 - Diagrama de blocos para um ANC malha aberta de banda larga.

Na maioria das metodologias, o controlador é um filtro digital cujos coeficientes são modificados (adaptados) segundo uma estratégia definida. O microfone de erro mede o sinal de erro (ou resíduo) $e(n)$, que é comumente utilizado como um índice de performance para adaptar os coeficientes do controlador. O uso de um sinal de erro para ajustar os coeficientes do filtro não constitui uma realimentação (malha fechada), pois o sinal de erro não é comparado com o sinal de referência. Além disto, ele não é entrada do filtro adaptativo, sendo utilizado somente na adaptação. O diagrama de blocos deste esquema pode ser visto, considerando a Transformada Z, na Figura 3.2.

Os sinais e funções de transferência são:

$d(n)$ ≡ Sinal da fonte de ruído no microfone de erro.

$e(n)$ ≡ Sinal de erro ou resíduo.

$x(n)$ ≡ Sinal de referência (entrada) do controlador.

$y(n)$ ≡ Sinal de saída do controlador.

$P(z)$ ≡ Função entre a fonte de ruído e o microfone de erro (Caminho Primário).

$G(z)$ ≡ Função de transferência entre a fonte de ruído e o microfone de referência.

$S(z)$ ≡ Função entre o atuador e o microfone de erro (Caminho Secundário).

$F(z)$ ≡ Função entre o atuador e o microfone de referência (Caminho de Realimentação).

$W(z)$ ≡ Função de transferência do sistema de controle.

Segundo à Figura 3.2, a fonte de ruído produz um sinal indesejado que sofre mudanças em suas propriedades ao longo dos caminhos $G(z)$ e $P(z)$, tornando-se $u(n)$ e $d(n)$, respectivamente. O controlador $W(z)$ recebe o sinal de referência $x(n)$, produzindo uma saída estimada $y(n)$. Esta saída também sofre mudanças ao longo dos caminhos $F(z)$ e $S(z)$, tornando-se $y''(n)$ e $y'(n)$, respectivamente. Este por sua vez, é somado acusticamente com o sinal $d(n)$, gerando o resíduo do ruído $e(n)$. A realimentação acústica indesejada $y''(n)$ pode causar problemas de instabilidade no sistema. Alguns autores desconsideram a existência da função $G(z)$, devido ao posicionamento do sensor ser muito próximo à fonte de ruído.

3.2 Sistema de Controle em Malha Aberta de Banda Estreita

Em aplicações onde o ruído primário é periódico (ou quase periódico) e é produzido por máquinas rotativas, por exemplo, o microfone de referência pode ser substituído por um sensor não-acústico, tal como acelerômetros, tacômetros, ou um sensor ótico. Esta substituição elimina os problemas de realimentação acústica que ocorre no sistema de banda larga.

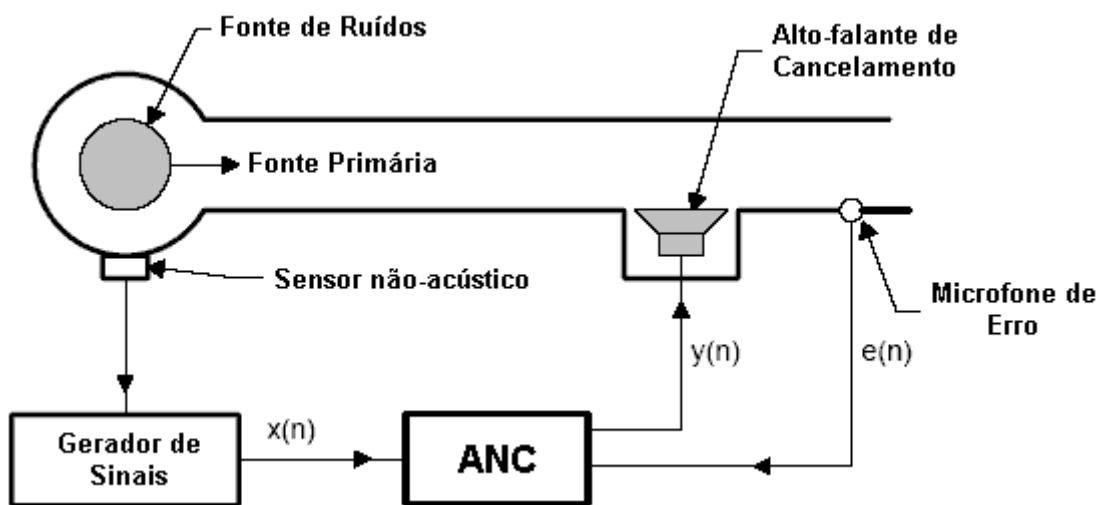


Figura 3.3 - Controle Ativo de Ruído em malha aberta de banda estreita.

O esquema para um sistema de controle em malha aberta de banda estreita (*Narrowband Feedforward ANC*) pode ser visto na Figura 3.3. O sensor não-acústico é sincronizado com a fonte de ruídos e é utilizado para sintetizar um sinal de entrada que contenha a freqüência fundamental e seus harmônicos da fonte primária. Este tipo de sistema controla os ruídos harmônicos através da filtragem adaptativa do sinal de referência sintetizado. Em veículos, as rotações por minuto (RPM) geralmente estão disponíveis e podem ser usadas como sinal de referência. Um microfone de erro ainda é necessário para medir o resíduo do ruído acústico. Este sinal de erro é então utilizado para ajustar os coeficientes do filtro adaptativo. O diagrama de blocos pode ser visto na Figura 3.4.

Segundo à Figura 3.4, a fonte de ruído produz um sinal indesejado que sofre mudanças em suas propriedades ao longo do caminho $P(z)$, tornando-se $d(n)$. O sinal de entrada do controlador $W(z)$ é sintetizado pelo mesmo ou adquirido por um sensor não-acústico. Este sinal sincronizado de referência $x(n)$ produz no controlador uma saída estimada $y(n)$. Esta saída também sofre mudanças ao longo do caminho $S(z)$, tornando-se $y'(n)$. Este por sua vez, é somado acusticamente com o sinal $d(n)$, gerando o resíduo $e(n)$. A realimentação indesejada responsável por problemas de instabilidade no sistema não ocorre neste tipo de configuração.

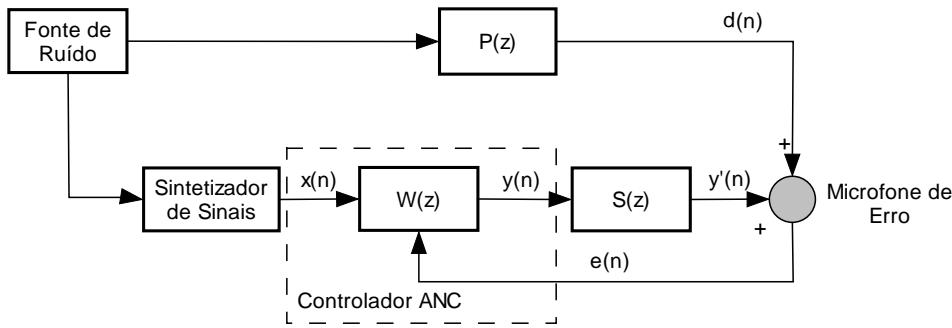


Figura 3.4 - Diagrama de blocos para um ANC malha aberta de banda estreita.

Os sinais e funções de transferência são:

$d(n)$ ≡ Sinal da fonte de ruído no microfone de erro.

$e(n)$ ≡ Sinal de erro ou resíduo.

$x(n)$ ≡ Sinal de referência (entrada) do controlador.

$y(n)$ ≡ Sinal de saída do controlador.

$P(z)$ ≡ Caminho Primário.

$S(z)$ ≡ Caminho Secundário.

$W(z)$ ≡ Função de transferência do sistema de controle.

Geralmente, esta técnica apresenta algumas vantagens como se segue (Kuo et al., 1996):

- Problemas de degradação física do microfone de referência em ambientes agressivos são eliminados. Isto é importante do ponto de vista de engenharia, por causa da dificuldade de medir o ruído de referência em altas temperaturas e dutos de gás turbulentos como em sistemas de exaustão.
- Os problemas de causalidade são eliminados devido à periodicidade do ruído. A forma de onda do ruído possui componentes em freqüência constantes. Somente ajustes em amplitude e fase são necessários. Isto permite maior flexibilidade no posicionamento do alto-falante de cancelamento.
- O controlador tem a habilidade do cancelamento seletivo, ou seja, somente alguns harmônicos podem ser controlados independentemente (filtros *notch*).
- É necessário modelar somente parte da função de transferência da planta acústica relacionada com os tons harmônicos. Um filtro FIR com poucos coeficientes pode ser usado, melhorando a eficiência computacional.

- A realimentação acústica indesejada do alto-falante de cancelamento no microfone de referência é eliminada.

3.3 Sistema de Controle em Malha Fechada

Este tipo de controle ativo ruído foi proposto por Olson em 1953. Neste esquema, um microfone é utilizado como um sensor de erro para detectar o ruído indesejado. O sinal de erro é realimentado para um filtro eletrônico com resposta em magnitude e fase desejadas para produzir o cancelamento no sensor através de um alto-falante de cancelamento localizado próximo ao microfone. Esta configuração oferece uma atenuação limitada sobre uma restrita banda de freqüências para ruídos periódicos ou de banda limitada. A Figura 3.5 mostra o esquema de sistema de controle de ruídos em malha fechada (*Feedback ANC*).

O sistema pode sofrer de instabilidade para altas freqüências, então é necessário introduzir no sistema filtros compensadores que evitem esta realimentação positiva. O sistema é mais estável quanto menor seja o atraso existente entre o microfone e o alto-falante, ou seja, situações em que a zona de silêncio esteja bem próxima ao alto-falante.

Uma das aplicações do sistema de controle em malha fechada reconhecida por Olson em 1956 foi o cancelamento do som em fones de ouvido e protetores auriculares. Nesta aplicação, o sistema reduz a pressão acústica na cavidade próxima ao ouvido. Este aparato tem sido desenvolvido e pode ser encontrado comercialmente.

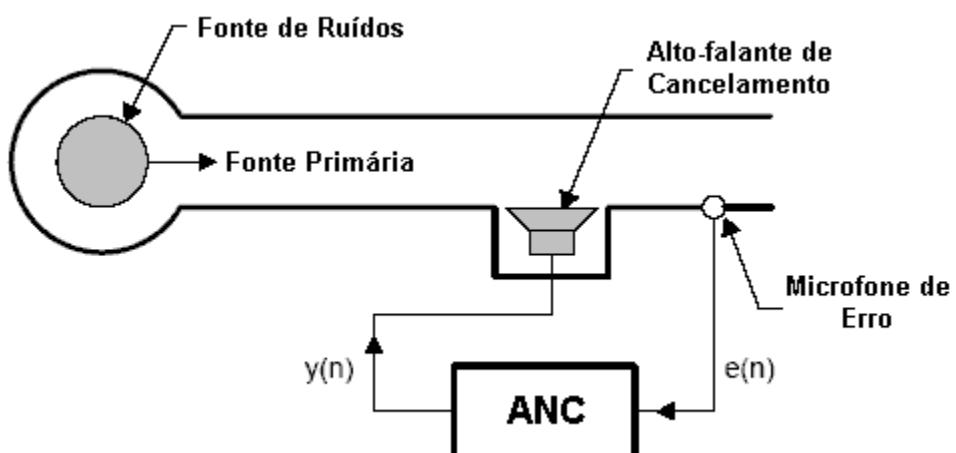


Figura 3.5 - Controle Ativo de Ruído em malha fechada.

A idéia básica deste sistema de controle consiste em se estimar o sinal do ruído indesejado, e a partir desta estimativa gerar o sinal de cancelamento (Kuo *et al.*, 1999). Neste tipo de sistema somente a informação proveniente do microfone de erro é utilizada para realizar o cancelamento. Na Figura 3.6, pode-se ver o diagrama de blocos do sistema apresentado.

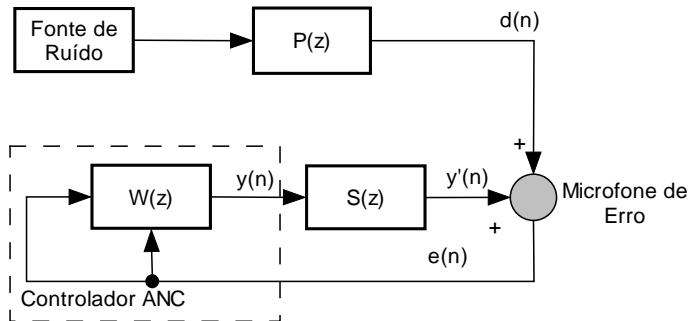


Figura 3.6 - Diagrama de blocos para um ANC em malha fechada.

Os sinais e funções de transferência são:

$d(n)$ ≡ Sinal da fonte de ruído no microfone de erro.

$e(n)$ ≡ Sinal de erro ou resíduo.

$y(n)$ ≡ Sinal de saída do controlador.

$P(z)$ ≡ Caminho Primário.

$S(z)$ ≡ Caminho Secundário.

$W(z)$ ≡ Função de transferência do sistema de controle.

Segundo à Figura 3.6, a fonte de ruído produz um sinal indesejado que sofre mudanças em suas propriedades ao longo do caminho $P(z)$, tornando-se $d(n)$. O sistema de controle recebe um sinal de erro $e(n)$, produzindo uma estimativa da entrada $x(n)$. A partir desta entrada, o controlador $W(z)$ produz uma saída $y(n)$. Esta saída sofre mudanças ao longo do caminho $S(z)$, tornando-se $y'(n)$. Este por sua vez é somado acusticamente com o sinal $d(n)$, gerando o cancelamento ou atenuação do ruído. O resíduo do cancelamento $e(n)$ é realimentado e o processo repete indefinidamente.

3.4 Sistema de Controle Híbrido

Os sistemas de malha aberta (*feedforward*) discutidos anteriormente utilizam dois sensores: um sensor de referência e outro de erro. No sistema de banda larga utiliza-se um sensor acústico como referência, já no sistema de banda estreita, utiliza-se um sensor não-acústico. Este sensor de referência capta o ruído primário a ser cancelado, enquanto o sensor de erro monitora a performance do sistema.

O sistema de malha fechada (*feedback*) utiliza somente o sensor de erro para cancelar somente as componentes estimadas do ruído primário.

A combinação das duas estruturas é denominada de sistema híbrido, como pode ser visto na Figura 3.7. A estrutura *feedforward* atenua o ruído primário que está correlacionado com o sinal de referência, enquanto que a estrutura *feedback* cancela as componentes estimadas que não são observáveis pelo sensor de referência. Sistemas híbridos são desenvolvidos para combinar as potencialidades de ambas estratégias (Yuan, 2004).

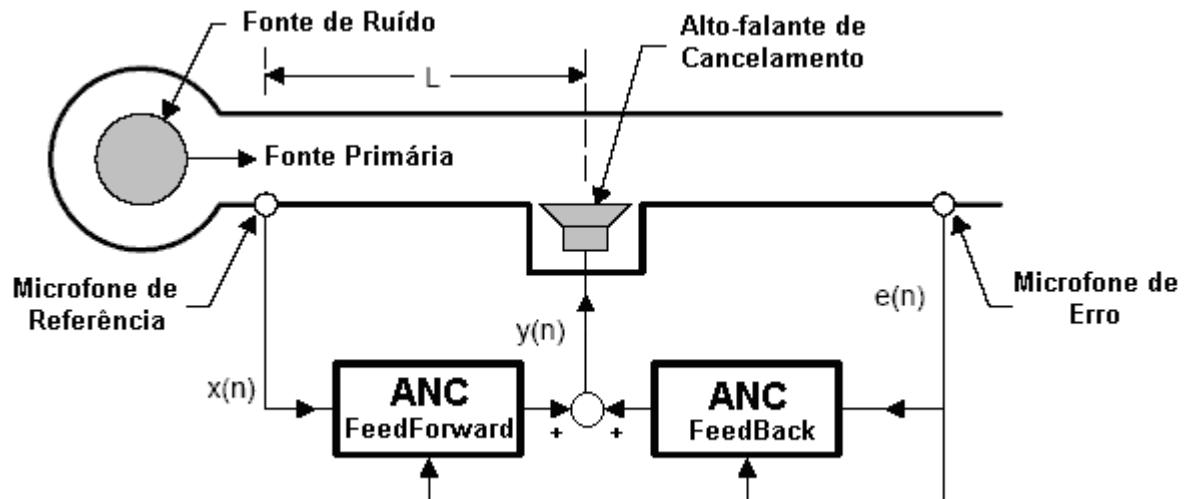


Figura 3.7 – Controle Ativo Híbrido de Ruído.

CAPÍTULO IV

Algoritmos para Controle Ativo de Ruído

Depois de realizar uma introdução sobre as metodologias de controle ativo de ruído, esta seção apresenta uma análise diferenciada, em função da configuração adotada, de diferentes algoritmos de controle para sistemas mono-canal.

O capítulo inicia discutindo a estratégia de controle para sistemas em malha aberta de banda larga, avaliando os efeitos dos caminhos secundário e de realimentação acústica.

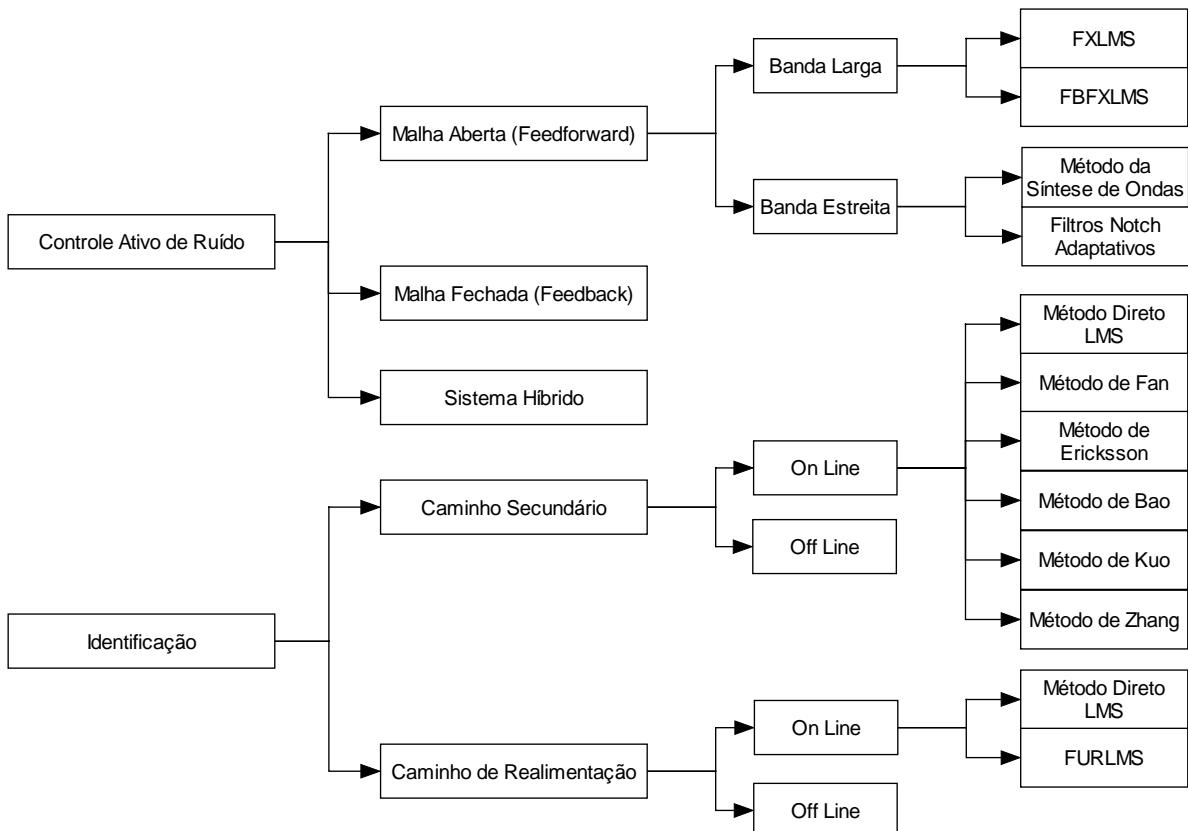


Figura 4.1 - Fluxograma da organização do capítulo.

Em seguida, trata-se dos algoritmos de malha aberta de banda estreita, apresentando duas estratégias associadas a esta configuração.

Após o estudo das estratégias em malha aberta, discute-se a estratégia em malha fechada e por fim a configuração híbrida.

A questão da identificação dos caminhos secundário e de realimentação, seja *off-line* ou *on-line*, também é objeto de estudo neste capítulo.

Um diagrama de blocos da organização do capítulo pode ser visto na Figura 4.1.

4.1 Algoritmos para Sistemas ANC em Malha Aberta de Banda Larga

Controladores ativos de ruído em banda larga podem ser descritos como um trabalho de identificação de sistemas. O diagrama esquemático deste tipo de controle pode ser visualizado na Figura 4.2.

Usando a representação no domínio discreto da freqüência, o sistema de controle ativo ideal utiliza um filtro adaptativo $W(z)$ para estimar a resposta do caminho desconhecido $P(z)$ entre o sensor de referência e o sensor de erro.

A transformada Z de $e(n)$ pode ser expressa como:

$$E(z) = D(z) + Y(z) = X(z)[P(z) + W(z)] \quad (4.1)$$

Onde $E(z)$ é o sinal de erro, $X(z)$ é o sinal de referência ou entrada e $Y(z)$ é saída do filtro adaptativo. Após o filtro adaptativo $W(z)$ convergir, deseja-se que $E(z) = 0$. A equação **Erro! A origem da referência não foi encontrada.** pode ser escrita como:

$$W(z) = -P(z) \quad (4.2)$$

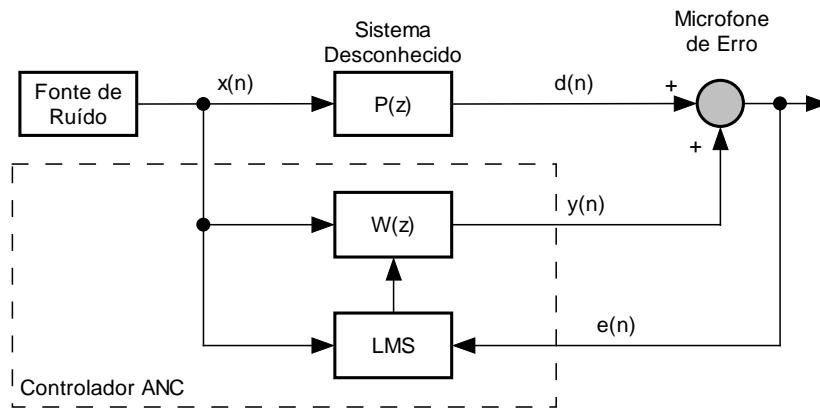


Figura 4.2 - Diagrama de um ANC de banda larga.

Isto mostra que o filtro adaptativo $W(z)$ estima a função de transferência desconhecida $P(z)$ defasada de 180° , o que implica em:

$$y(n) = -d(n)$$

Então, a saída do filtro adaptativo $y(n)$ tem a mesma amplitude, mas está em fase oposta com o ruído $d(n)$. Quando $d(n)$ e $y(n)$ são acusticamente combinados, o erro residual se torna nulo ou zero, resultando no cancelamento de ambas ondas sonoras de acordo com o princípio da superposição.

4.1.1 Algoritmo Adaptativo LMS

A maioria dos algoritmos adaptativos busca os coeficientes ótimos através do método do gradiente descendente (*Gradient Descent*) ou método da máxima descida (*Stepest Descent*) segundo a direção oposta do gradiente da superfície do erro. Quando esta superfície corresponde ao erro quadrático instantâneo e se utiliza um filtro FIR, o algoritmo em questão se denomina LMS (*Least Mean Square*). Este algoritmo é muito utilizado devido a sua eficiência e simplicidade. O desenvolvimento matemático do método é visto com mais detalhes no Anexo I.

O algoritmo LMS é apresentado a seguir:

1. Inicialização:

Coeficientes: $w_0(i) = 0$, M+1 coeficientes

Potência: $\sigma^2(0) = 1$

Parâmetro α

2. Leitura de $x(n)$ e $e(n)$.

3. Cálculo da saída do filtro FIR:

$$y(n) = \sum_{i=0}^{M} w(i)_n x(n-i) \quad (4.3)$$

4. Estimação da potência de $x(n)$:

$$\sigma^2(n) = \alpha x^2(n) + (1 - \alpha) \sigma^2(n-1) \quad (4.4)$$

5. Cálculo do passo de adaptação:

$$0 < \mu < \frac{1}{(M+1)\sigma^2(n)} \quad (4.5)$$

6. Atualização dos coeficientes:

$$w(i)_{n+1} = w(i)_n - 2\mu e(n)x(n-i) \quad (4.6)$$

7. Faça $n=n+1$ e volte ao ponto 2.

É importante notar que a equação (4.6) de atualização do filtro, em aplicações ANC, possui o sinal negativo ao invés do sinal positivo como no algoritmo convencional LMS. Isto vem do fato do sinal de erro de um sistema ANC ser $e(n) = d(n) + y(n)$, resultado da superposição (adição) acústica ao invés da subtração elétrica ou numérica.

4.1.2 Efeitos do Caminho Secundário

O sinal de erro $e(n)$ é medido no microfone de erro. A somatória na Figura 4.2 representa o ambiente acústico entre o alto-falante de cancelamento e o microfone de erro, onde o ruído primário $d(n)$ é combinado com o anti-ruído $y(n)$ calculado pelo filtro adaptativo. O anti-ruído pode ser modificado pelo caminho secundário $S(z)$ no canal acústico entre $y(n)$ e $e(n)$, tal como o ruído primário é modificado pelo caminho primário $P(z)$ entre a fonte de ruído e o sensor de erro. Um diagrama detalhado incluindo o caminho $S(z)$ pode ser visto na Figura 4.3.

Desta forma é necessário realizar uma compensação para $S(z)$ (Morgan, 1980). Em sistemas lineares, se dois blocos estão dispostos em cascata, pode-se comutar suas posições obtendo a mesma saída devido à mesma entrada. Realizando esta comutação, no ramo principal obtém-se a função $S(z)W(z)$ e no ramo inferior tem-se o filtrado FIR adaptativo. Equacionando a discussão acima:

$$E(z) = X(z)P(z) + X(z)W(z)S(z) \quad (4.7)$$

Permutando $S(z)$, obtém-se a entrada filtrada:

$$E(z) = X(z)P(z) + X(z)S(z)W(z) \quad (4.8)$$

Supondo que $\hat{S}(z)$ é uma estimativa do caminho secundário $S(z)$, a configuração obtida é semelhante à desenvolvida para o algoritmo LMS com a diferença que a entrada para a atualização dos coeficientes do filtro agora é filtrada pela função estimada $\hat{S}(z)$. O esquema pode ser visualizado na Figura 4.4.

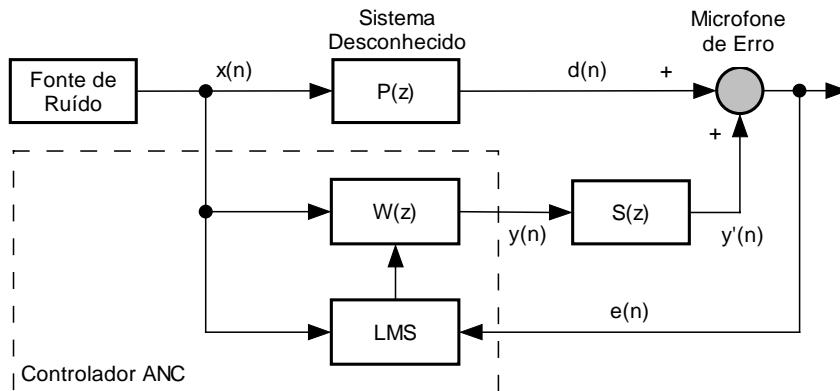


Figura 4.3 - Sistema ANC com caminho secundário $S(z)$.

Da equação (4.8), assumindo que $W(z)$ tenha convergência para os valores ótimos, supõe-se que o erro residual $E(z)$ seja nulo. Este resultado requer que:

$$W(z) = -\frac{P(z)}{S(z)} \quad (4.9)$$

Desta maneira, o filtro adaptativo $W(z)$ deve modelar o caminho primário $P(z)$ e inversamente o caminho secundário $S(z)$. A vantagem desta aproximação é que com um modelo apropriado, o sistema pode responder instantaneamente às mudanças no sinal de entrada causadas pelas mudanças nas fontes de ruído. Entretanto, a performance do ANC torna-se altamente dependente do caminho secundário. Um filtro adaptativo FIR de alta ordem deve ser utilizado para se estimar a razão $1/S(z)$. Além de ser impossível inverter o atraso causado pelo caminho secundário se o caminho primário não conter um atraso maior ou pelo menos igual de $S(z)$. Isto diz que a ordem de $S(z)$ deve ser maior que de $P(z)$. Este limite de causalidade restringe os sistemas de controle de banda larga.

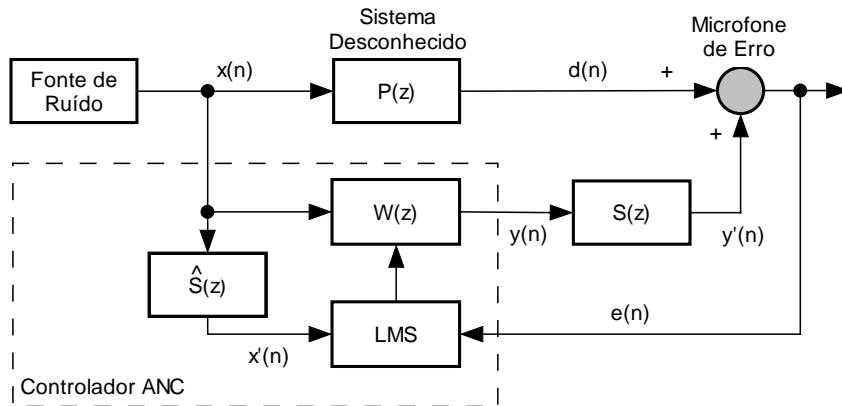


Figura 4.4 - Diagrama de blocos do algoritmo FXLMS.

Da equação (4.9), o controle será instável se em uma dada freqüência qualquer, a resposta do caminho secundário for $S(\omega) = 0$. Se o ganho do caminho secundário também for muito menor que o ganho do caminho primário para uma dada freqüência, a performance do sistema também será degradada, já que o sinal de controle não conseguirá se manifestar no microfone de erro.

O controle será inefficiente também se existir uma freqüência ω onde $P(\omega) = 0$, ou seja, um zero no caminho primário causa uma freqüência não-observável no controle. Entretanto, as características do caminho secundário possuem efeitos mais significativos na performance do sistema de controle ativo de ruído.

4.1.3 Algoritmo Adaptativo FXLMS

Para minimizar os efeitos do caminho secundário $S(z)$ e garantir a convergência do processo, a entrada para o algoritmo de adaptação deve ser filtrada pela estimativa do

caminho secundário $\hat{S}(z)$. Isto leva ao algoritmo FXLMS (*Filtered-X LMS*), que foi desenvolvido por Widrow (1981) no contexto de controle adaptativo e Burgess (1981) aplicou ao controle ativo de ruído.

As equações de atualização dos coeficientes são as mesmas do algoritmo LMS, com a diferença que os dados de entrada para a adaptação são filtrados pela estimativa do caminho secundário $\hat{S}(z)$. Assim:

$$w(i)_{n+1} = w(i)_n - 2\mu e(n)x'(n-i) \quad (4.10)$$

Onde a entrada filtrada é dada por:

$$x'(n) = \hat{s}^T x(n) = \sum_{i=0}^N \hat{s}_i x(n-i) \quad (4.11)$$

e:

$$\hat{s} = [\hat{s}_0 \ \hat{s}_1 \cdots \hat{s}_{N-1}]^T \quad (4.12)$$

é o vetor de coeficientes da estimativa do caminho secundário.

O algoritmo FXLMS é apresentado a seguir:

1. Inicialização:
Coeficientes: $w(i)_0 = 0$, M+1 coeficientes
Potência: $\sigma^2(0) = 1$
Parâmetro α
2. Leitura de $x(n)$ e $e(n)$.
3. Cálculo da saída do filtro FIR:
$$y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.13)$$
4. Cálculo da entrada filtrada:
$$x'(n) = \sum_{i=0}^{N-1} \hat{s}(i)_n x(n-i) \quad (4.14)$$
5. Estimação da potência de $x'(n)$:
$$\sigma^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma^2(n-1) \quad (4.15)$$
6. Cálculo do passo de adaptação:
$$0 < \mu < \frac{1}{(M+1+N)\sigma^2(n)} \quad (4.16)$$
7. Atualização dos coeficientes:
$$w(i)_{n+1} = w(i)_n - 2\mu e(n)x'(n-i) \quad (4.17)$$
8. Faça $n=n+1$ e volte ao ponto 2.

Neste algoritmo, o passo máximo de adaptação não deve ser dado apenas pela potência do sinal de entrada e o número de coeficientes do filtro adaptativo, conforme a equação (4.5), mas deve-se considerar o atraso em amostras do caminho secundário

(Elliott, 1987). Este atraso pode ser dado pelo número de coeficientes do filtro FIR utilizado para estimar $S(z)$, levando à nova equação dada em (4.16).

A estimativa do caminho secundário pode ser feita tanto *off-line*, como *on-line*. O item 4.5 discute o assunto mais profundamente.

4.1.4 Efeitos da Realimentação Acústica

Examinando novamente a Figura 3.1, a saída do alto-falante de cancelamento não só atenua o ruído no microfone de erro, mas, indesejavelmente, o anti-ruído irradia em direção ao microfone de referência, resultando em um sinal de entrada contaminado. Este retorno acústico introduz uma realimentação no sistema, que pode levar à instabilidade.

Este problema da realimentação indesejada tem sido intensamente estudado no controle ativo de ruídos. Algumas soluções podem ser citadas neste contexto:

- Uso de microfones e alto-falantes direcionais (Tichy *et al.*, 1983).
- Uso de sinais de compensação para cancelar os efeitos do retorno acústico. Sinais gerados através de um filtro de compensação cujos coeficientes são determinados *off-line* através de um treinamento (Kuo *et al.*, 1996).
- Uso de um filtro adaptativo em tempo real em paralelo com o caminho de retorno (Poole *et al.*, 1984).
- Uso de filtro adaptativo IIR (Eriksson *et al.*, 1987).

4.1.5 Algoritmo Adaptativo FBFXLMS

Um controle de malha aberta com compensação de realimentação pode ser visto na Figura 4.5. O filtro $\hat{F}(z)$ é um estimador do caminho de realimentação entre a saída $y(n)$ e o sinal de referência $u(n)$. O filtro $\hat{F}(z)$ remove a realimentação acústica do microfone de referência. O filtro $\hat{S}(z)$ compensa os efeitos do caminho secundário $S(z)$ no algoritmo FXLMS, como visto anteriormente. A remoção da realimentação acústica melhora de forma significativa a margem de estabilidade do sistema (Kuo *et al.*, 1996).

As expressões para o sinal de controle $y(n)$, o sinal filtrado $x'(n)$ e a equação de adaptação para o algoritmo FBFXLMS (*Feedback-Filtered-X LMS*) são as mesmas para o FXLMS, com exceção para $x(n)$ no FBFXLMS que é um sinal livre da realimentação e pode ser dado por:

$$x(n) = u(n) - \sum_{i=0}^L \hat{f}_i y(n-i) \quad (4.18)$$

Onde $u(n)$ é o sinal do microfone de referência, \hat{f}_i é o i-ésimo coeficiente de $\hat{F}(z)$ e $L+1$ é o número de coeficientes do filtro.

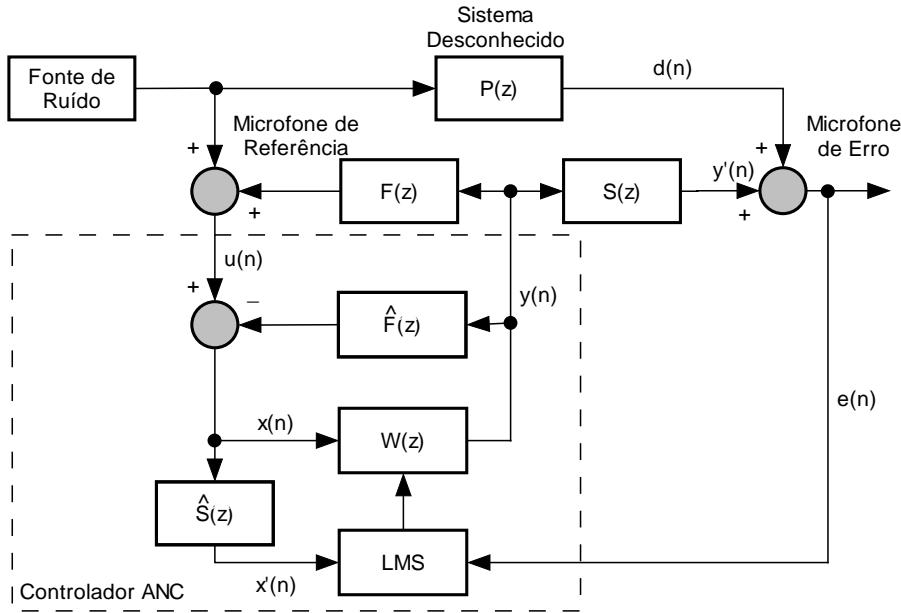


Figura 4.5 - Sistema ANC com cancelamento da realimentação acústica.

No caso da modelagem ideal do caminho de realimentação, isto é $\hat{F}(z) = F(z)$, a realimentação acústica é totalmente cancelada. O filtro adaptativo converge para a função de transferência ideal dada pela equação (4.9).

4.2 Algoritmos para Sistemas ANC em Malha Aberta de Banda Estreita

Em aplicações práticas, o ruído primário pode ser produzido por máquinas rotativas. Nestes casos, o ruído possui características periódicas e um método alternativo pode ser utilizado. Estes métodos estimam o sinal acústico utilizando uma medida indireta através de um sensor não-acústico ao invés de um microfone de referência.

Em geral, dois tipos de sinais de referência são comumente utilizados em sistemas de banda estreita. A primeira técnica, chamada de Método de Síntese de Ondas, foi proposta por Chaplin em 1983. Nela, um trem de impulsos com o período igual ao inverso da freqüência fundamental do ruído pode ser utilizado (Elliot et al., 1985).

Na segunda técnica, filtros *notch* adaptativos, são utilizados. Filtros *notch* possuem característica rejeita-faixa com banda estreita, onde somente a freqüência central é atenuada. Ondas senoidais que tenham as mesmas freqüências correspondendo aos tons

harmônicos a serem cancelados são utilizados como referência. Este método foi originalmente desenvolvido para cancelamento de interferência tonal (Widrow *et al.*, 1975) e posteriormente aplicado ao controle ativo de ruído periódico (Ziegler, 1989).

4.2.1 Método da Síntese de Ondas

O sintetizador de ondas produz uma saída $y(n)$ que excita o alto-falante de cancelamento. A onda gerada possui sincronia com o sensor não-acústico. Um microfone na zona de silêncio mede o resíduo acústico e é utilizado para a adaptação do sintetizador. O cancelamento ocorre somente nas freqüências dos harmônicos.

O sintetizador armazena amostras do ruído de cancelamento $\{w_k(n), k=0, 1, \dots, N-1\}$, onde N é o número de amostras por ciclo de onda. Estas amostras representam a onda a ser gerada para produzir o ruído de cancelamento. Isto é:

$$y(n) = w_k(n) \quad (4.19)$$

representa o k -ésimo elemento das amostras e pode ser implementado como um ponteiro de uma lista circular controlado pelas interrupções geradas pelo sinal de sincronização.

O ruído residual é amostrado pelo microfone de erro em sincronia com o sinal de referência. O erro é então utilizado para ajustar os valores da onda de cancelamento $\{w_k(n)\}$ pelo seguinte algoritmo:

$$w_k(n+1) = w_k(n) - \mu e(n) \quad (4.20)$$

Em sistemas práticos, existe um atraso de tempo entre o momento que o sinal excita o alto-falante e o momento em que ele chega ao microfone de erro. Este atraso pode ser acomodado pela subtração temporal no ponteiro k da lista circular:

$$w_{k-\Delta}(n+1) = w_{k-\Delta}(n) - \mu e(n) \quad (4.21)$$

Onde Δ é o atraso de tempo em amostras entre a saída do sinal do sintetizador e sua chegada ao microfone de erro, isto é:

$$\Delta = \frac{\tau}{T} \quad (4.22)$$

Onde τ é atraso dado pelo arranjo alto-falante e microfone (que é constante para um dado sistema) e T é o período de amostragem. Devido ao fato da taxa de amostragem ser sincronizada com a fonte de ruído, este número Δ é atualizado de acordo com as mudanças na taxa de amostragem.

O conceito da síntese de onda pode ser analisado através de um filtro FIR de order N excitado por um trem de pulsos de período T_0 . Considere a função de transferência $G(z)$ entre a entrada $X(z)$ e o erro $E(z)$ mostrada a seguir:

$$G(z) = \frac{E(z)}{X(z)} = \frac{1 - z^{-N}}{1 - (1 - \mu)z^{-N}} \quad (4.23)$$

A função transferência acima representa vários filtros *notches* centrados em cada freqüência harmônica do ruído. Os zeros têm amplitude unitária constante e são igualmente espaçados ($2\pi/T_0$) no círculo unitário no plano z. Estes zeros criam pontos de nulo na resposta em freqüência nos harmônicos. Os pólos têm a mesma freqüência dos zeros, mas são igualmente espaçados no círculo unitário à distância $(1-\mu)$ da origem. O efeito destes pólos é introduzir uma ressonância na vizinhança do nulo, isto reduz a banda do filtro *notch*.

A equação (4.23) limita o valor de μ por considerações de estabilidade, isto é, $0 < \mu < 1$. A banda de 3db para cada *notch* é aproximadamente (Elliot et al., 1985):

$$BW \approx \frac{\mu}{\pi T} (\text{Hz}) \quad (4.24)$$

Isto indica que a banda do filtro é proporcional ao passo de adaptação e à taxa de amostragem do sistema.

4.2.2 Filtros *Notch* Adaptativos

O segundo tipo de sinal de referência utilizado em controle ativo de ruído de banda estreita é um sinal senoidal com mesma freqüência que o ruído a ser cancelado. Quando um sinal senoidal é empregado como referência ao algoritmo LMS, este se torna um filtro *notch* adaptativo centrado na freqüência da referência. As vantagens deste tipo de filtro são que eles oferecem um fácil controle da banda, um nulo infinito e a capacidade de se adaptar seguindo a freqüência exata do ruído. Isto é verdade quando o ruído sofre pequenas mudanças (Kuo et al., 1996).

O diagrama de blocos para este tipo de sistema com dois coeficientes adaptativos é mostrando na Figura 4.6. O sensor não-acústico é utilizado para determinar a freqüência fundamental do ruído gerado. Por exemplo, um motor elétrico girando a 1800RPM completa 30 revoluções por segundo com uma freqüência fundamental de 30Hz.

Um gerador de onda senoidal prove uma referência na freqüência desejada. Com um defasador de 90° , a onda senoidal é dividida em duas componentes ortogonais, $x_0(n)$ e $x_1(n)$, que são utilizados como referência para o filtro adaptativo. Para o defasamento, pode ser empregada a Transformada de Hilbert, um processo de integração ou derivação, ou utilizar

um gerador de ondas senoidais e cossenoidais. Os dois sinais são filtrados e somados para produzir a saída $y(n)$:

$$y(n) = w_0(n)x_0(n) + w_1(n)x_1(n) \quad (4.25)$$

onde:

$$x_0(n) = A \sin(k\omega_0 n) \quad (4.26)$$

e

$$x_1(n) = B \cos(k\omega_0 n) \quad (4.27)$$

onde ω_0 é a freqüência fundamental, k é o índice harmônico, A e B são amplitudes e n é índice de tempo. O gerador de ondas pode ser implementado de várias maneiras em um processador DSP (Jackson, 1989).

A magnitude e fase deste sinal de referência são ajustadas no controlador. Este sinal alimenta um ou mais alto-falantes para cancelar o ruído primário. O algoritmo LMS atualiza os coeficientes do filtro para minimizar o erro $e(n)$:

$$w_i(n+1) = w_i(n) - \mu e(n)x_i(n - \Delta_k), \text{ onde } i=0 \text{ ou } 1 \quad (4.28)$$

Onde Δ_k é utilizado para compensar os efeitos do caminho secundário no harmônico k . Este fator representa o atraso introduzido entre a saída do filtro adaptativo e a entrada do erro residual.

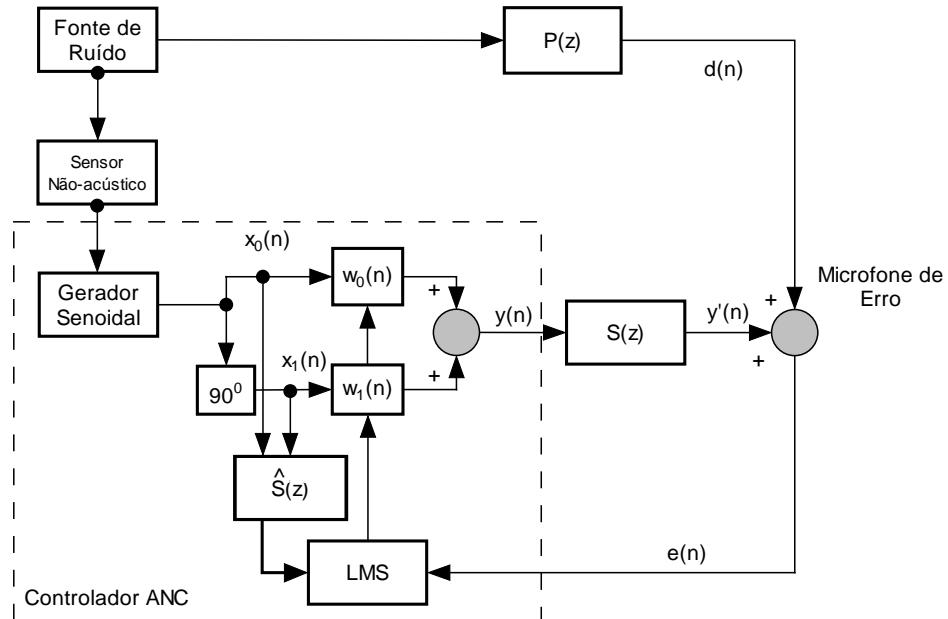


Figura 4.6 - Controle Ativo de Ruído com Filtro *Notch* Adaptativo.

No entanto, este atraso pode ser substituído pela estimativa do caminho secundário. O algoritmo *notch* adaptativo utilizando o algoritmo FXLMS pode ser expresso como:

$$w_i(n+1) = w_i(n) - \mu e(n)x_i'(n), \text{ onde } i=0 \text{ ou } 1 \quad (4.29)$$

Onde $x_i'(n)$ é versão filtrada de $x_i(n)$ pela estimativa $\hat{S}(z)$ do caminho secundário.

Em aplicações práticas, o ruído periódico usualmente contém tons na freqüência fundamental e vários harmônicos. Este tipo de ruído pode ser atenuado por um filtro com múltiplos *notches*. A realização deste tipo de estrutura requer um filtro de alta ordem, que também pode ser realizado por conexões em paralelo ou cascata de múltiplas seções de segunda ordem.

Um método para eliminar múltiplas senóides ou outras interferências periódicas foi proposta por Glover em 1977. A aplicação desta técnica é gerar a referência de entrada com uma soma de M senóides. Isto é:

$$x(n) = \sum_{m=1}^M A_m \cos(\omega_m n) \quad (4.30)$$

Quando esta soma de senóides é aplicada a um filtro adaptativo, o filtro converge para um filtro *notch* localizado em cada freqüência da referência. Este filtro *notch* adaptativo provê um simples método para ajustar e eliminar interferências harmônicas. A aplicação de Glover para atenuação ativa em máquinas foi patenteada (Pfaff *et al.*, 1992).

Uma senóide de freqüência única pode ser cancelada por um filtro adaptativo de 2 coeficientes. Para o caso onde o ruído primário contém M harmônicos, M filtros adaptativos de 2 coeficientes devem ser conectados em paralelo. Uma senóide específica é utilizada como referência para o correspondente canal $W_m(z)$, que é conectado em paralelo com outros filtros, como mostrado na Figura 4.7.

A estrutura de cada canal individual é mostrada na Figura 4.6. A função de transferência desta configuração em paralelo é:

$$W(z) = \sum_{m=1}^M W_m(z) \quad (4.31)$$

Onde $m=1, 2, \dots, M$ é o índice do canal. O sinal de cancelamento é a soma das M saídas do filtro adaptativo. Isto é:

$$y(n) = \sum_{m=1}^M w_m(n) \quad (4.32)$$

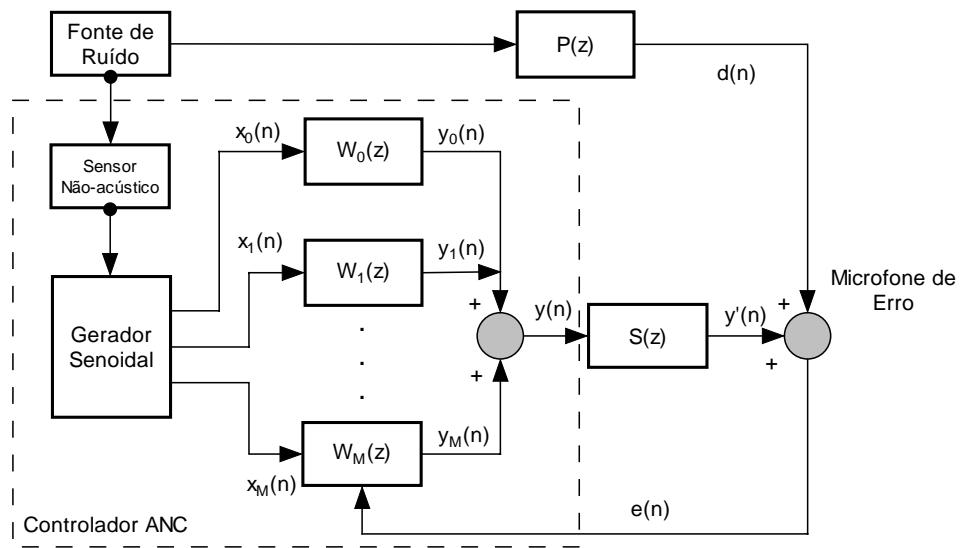


Figura 4.7 - Múltiplos Filtros Notch Adaptativos.

Cada entrada é filtrada pela estimativa do caminho secundário $\hat{S}(z)$:

$$x_m(n) = \sum_{i=0}^N \hat{s}_i x_m(n-i), \quad m=1, 2, \dots, M \quad (4.33)$$

Devido ao fato de apenas um microfone de erro ser utilizado, há somente um sinal de erro $e(n)$ utilizado para atualizar os M filtros adaptativos baseados no algoritmo FXLMS.

4.3 Algoritmos para Sistemas ANC em Malha Fechada

Ao lado dos procedimentos de controle em malha aberta, destaca-se os controladores de malha fechada (*feedback*), cujo diagrama de blocos para um sistema adaptativo é mostrado na Figura 4.8. O sinal do sensor de erro é processado pelo sistema de controle para gerar o sinal de cancelamento. Devido ao fato deste sistema não utilizar microfone de referência, não existe o problema da realimentação acústica.

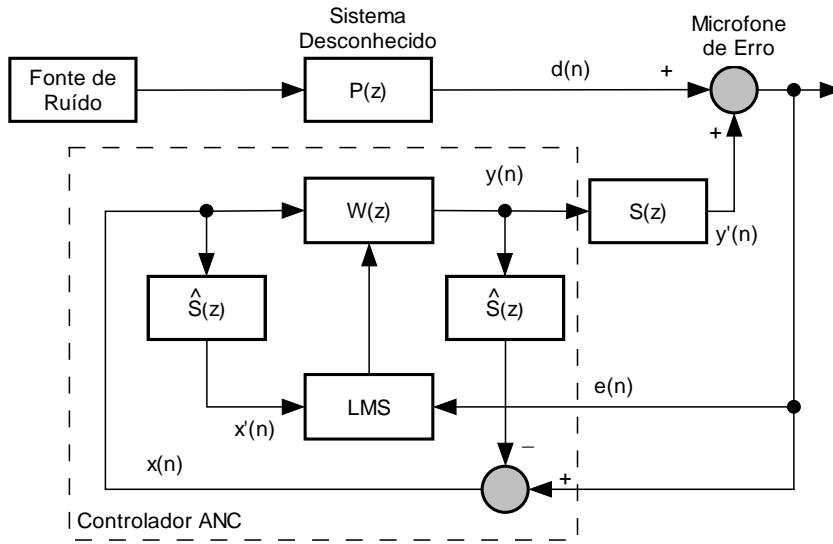


Figura 4.8 - Diagrama de Blocos do Sistema Malha Fechada.

Burgess (1981) sugeriu a configuração em malha fechada com o algoritmo FXLMS para deixar de utilizar o microfone de referência. A idéia básica deste algoritmo é estimar o ruído primário $d(n)$ e utilizá-lo como entrada no filtro adaptativo. Da Figura 4.8, o ruído primário é dado por:

$$D(z) = E(z) - S(z)Y(z) \quad (4.34)$$

Onde $E(z)$ e $Y(z)$ são disponíveis. Se a função de transferência do caminho secundário for bem modelada por $\hat{S}(z)$, pode-se escrever:

$$D(z) = E(z) - \hat{S}(z)Y(z) \approx X(z) \quad (4.35)$$

Desta forma, o ruído primário pode ser estimado por:

$$x(n) = e(n) - \sum_{i=0}^N \hat{s}_i y(n-i-1) \quad (4.36)$$

Onde $\hat{S}(z)$ é a estimativa do caminho secundário e N é a ordem do filtro. Nota-se que a entrada estimada depende da saída anterior do filtro adaptativo, ou seja, existe o atraso de uma amostra.

O algoritmo *Feedback FXLMS* pode ser visto a seguir:

1. Inicialização:
Coeficientes: $w(i)_0 = 0$, M+1 coeficientes
Potência: $\sigma^2(0) = 1$
Parâmetro α
2. Leitura de $x(n)$.
3. Estime a entrada o filtro $x(n)$:

$$x(n) = e(n) - \sum_{i=0}^N \hat{s}_i y(n-i-1) \quad (4.37)$$

4. Cálculo da saída do filtro FIR:

$$y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.38)$$

5. Cálculo da entrada filtrada:

$$x'(n) = \sum_{i=0}^N \hat{s}(i)_n x(n-i) \quad (4.39)$$

6. Estimação da potência de $x'(n)$:

$$\sigma^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma^2(n-1) \quad (4.40)$$

7. Cálculo do passo de adaptação:

$$0 < \mu < \frac{1}{(M+N+2)\sigma^2(n)} \quad (4.41)$$

8. Atualização dos coeficientes:

$$w(i)_{n+1} = w(i)_n - 2\mu e(n)x'(n-i) \quad (4.42)$$

9. Faça $n=n+1$ e volte ao ponto 2.

4.4 Algoritmos para Sistemas ANC em Proposta Híbrida

O sistema híbrido utilizando FIR feedforward e feedback adaptativo é mostrado na Figura 4.9, onde o sinal de cancelamento $y(n)$ é gerado utilizando as saídas de ambos filtros adaptativos $A(z)$ e $C(z)$, malha aberta e malha fechada, respectivamente.

O controlador combinado possui duas entradas de referência: $x(n)$ do sensor de referência e $\hat{d}(n)$ estimado do ruído primário. Versões filtradas $x'(n)$ e $\hat{d}'(n)$ são utilizadas na adaptação dos coeficientes de $A(z)$ e $C(z)$, respectivamente.

A vantagem do sistema híbrido em relação aos outros sistemas separados é que filtros de menor ordem podem ser utilizados para obter a mesma performance. Sistemas híbridos também demonstram claramente vantagem quando a planta é muito ruidosa (Kuo *et al.*, 1999).

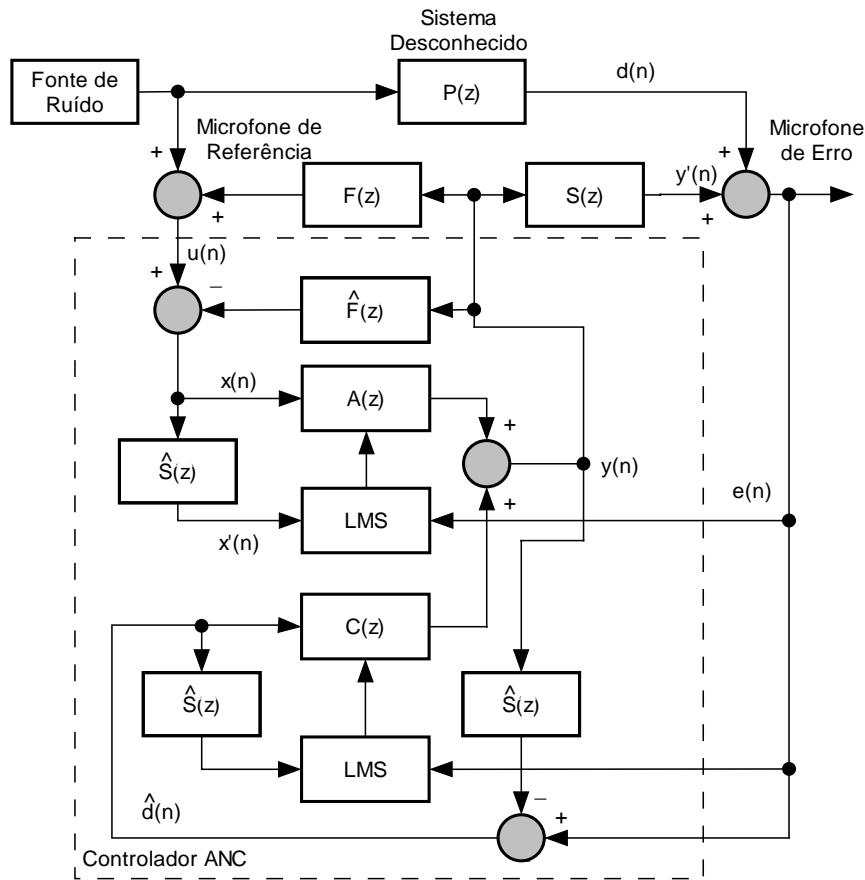


Figura 4.9 - Sistema Híbrido de Controle Ativo de Ruído.

4.5 Estimativa do Caminho Secundário

A função de transferência $S(z)$ é desconhecida e variante no tempo, devido as não linearidades dos atuadores e sensores, mudanças de temperatura, talvez turbulências de fluxo de gás no caminho secundário. Muitas vezes estas mudanças podem ser abruptas e comprometer a performance do sistema adaptativo. Em outros casos, estas mudanças podem não representar grandes problemas no cancelamento do ruído primário. Assim, várias formas de estimativa do caminho secundário já foram propostas na literatura, sendo estimativas em tempo real (*on-line*) ou através de um treinamento anterior (*off-line*).

4.5.1 Estimativa *off-line* do caminho secundário

Assumindo que as características de $S(z)$ são desconhecidas e invariantes no tempo, uma modelagem *off-line* pode ser utilizada para se estimar o caminho secundário através de um estágio de treinamento. No final deste estágio, o modelo estimado $\hat{S}(z)$ é utilizado no

controle ativo de ruído. O esquema experimental para a modelagem *off-line* é mostrada na Figura 4.10, onde um ruído branco não correlacionado é gerado internamente no DSP. Como a potência do ruído gerada é conhecida, não é necessário estimá-la recursivamente. Desta forma, o passo de adaptação é conhecido, fixo e deve satisfazer a equação (4.43).

$$0 < \mu < \frac{1}{(N+1)\sigma^2} \quad (4.43)$$

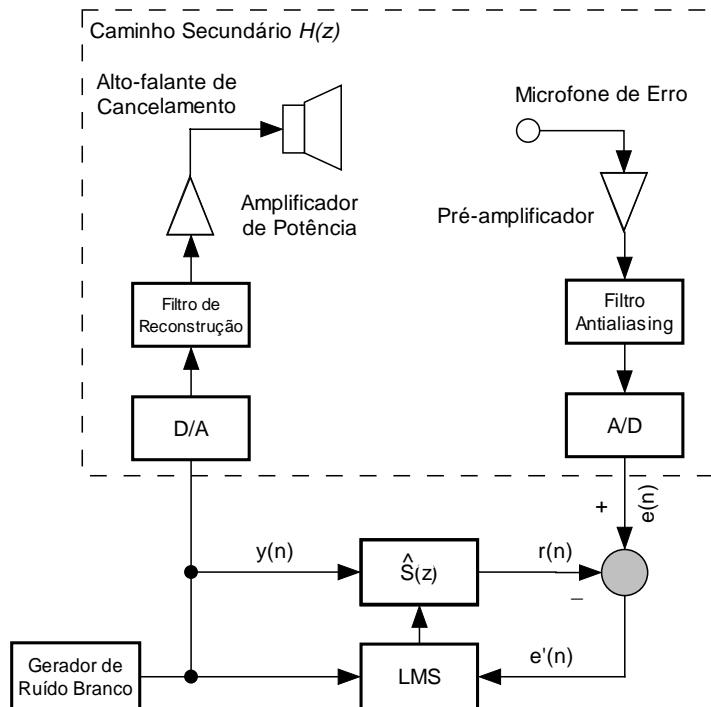


Figura 4.10 - Esquema de modelagem *off-line* do caminho secundário.

Algoritmo para a modelagem *off-line* do caminho secundário:

1. Inicialização:
Coeficientes: $\hat{S}(i)_0 = 0$, $N+1$ coeficientes
2. Gerar uma amostra do ruído branco $y(n)$. Este ruído gerado internamente é utilizado como referência para o filtro adaptativo $\hat{S}(z)$ e na adaptação de seus coeficientes.
3. Cálculo da saída do filtro adaptativo:

$$r(n) = \sum_{i=0}^N \hat{S}(i)_n y(n-i) \quad (4.44)$$
4. Compute a diferença:

$$e'(n) = e(n) - r(n) \quad (4.45)$$
5. Atualização dos coeficientes:

$$\hat{S}(i)_{n+1} = \hat{S}(i)_n + 2\mu e'(n)y(n-i) \quad (4.46)$$
6. Faça $n=n+1$ e volte ao ponto 2.
7. Repita o procedimento por cerca de 10 segundos. Salve os coeficientes do filtro adaptativo $\hat{S}(z)$ e utilize nos algoritmos dados anteriormente.

Os erros da estimativa não devem produzir diferenças de fase superiores a 90^0 no sinal do ruído acústico, pois tais desvios produzem instabilidades (Elliot *et al.*, 1987).

Esta modelagem também pode ser realizada através de um pacote de identificação de sistemas, como alguns oferecidos pelo MatLab® (Delfino *et al.*, 2005, Nuñez *et al.*, 2004). Após a identificação, os coeficientes são transferidos para o DSP e utilizados no controlador.

4.5.2 Estimativa *on-line* do caminho secundário

Em uma situação real é aconselhável se estimar a função $S(z)$ continuamente, já que este caminho pode-se modificar ao longo do tempo tal qual o caminho primário. Várias propostas já foram apresentadas para solucionar este problema. Uma revisão literária mostra que predominam duas formas de se estimar *on-line* o caminho secundário.

A primeira forma estima o caminho secundário $S(z)$ através da própria saída $y(n)$ do filtro adaptativo. Este método envolve uma modelagem dependente e muitas vezes não produz um modelo apropriado.(Bao *et al.*, 1993). A segunda forma utiliza a injeção adicional de ruído aleatório no sistema para realizar a identificação. Este método obtém uma modelagem independente e o modelo obtido é válido para toda a banda de freqüências de interesse. Esta injeção de ruído aumenta o ruído residual no sistema, no entanto, este efeito pode ser reduzido através da redução da potência de ruído injetada no sistema. Algumas destas técnicas serão tratadas a seguir.

4.5.2.1 Método Direto (Algoritmo LMS)

A solução imediata para se estimar o caminho secundário em tempo real é utilizar a configuração clássica de identificação de modelos através da modelagem direta, utilizando um filtro FIR e o algoritmo LMS proposto por Widrow em 1985, como mostra a Figura 4.11.

Pode-se dizer que o erro $h(n)$ para o algoritmo de adaptação do caminho secundário no domínio Z é dado por:

$$H(z) = E(z) - \hat{Y}(z) \quad (4.47)$$

Onde $\hat{Y}(z)$ é a saída do filtro adaptativo $\hat{S}(z)$. Substituindo $\hat{Y}(z)$ e $E(z)$, tem-se:

$$H(z) = [D(z) + Y'(z)] - \hat{S}(z)Y(z) \quad (4.48)$$

Mas $D(z)=P(z)X(z)$, $Y(z)=W(z)X(z)$ e $Y'(z)=S(z)W(z)X(z)$, assim:

$$H(z) = P(z)X(z) + S(z)W(z)X(z) - \hat{S}(z)W(z)X(z) \quad (4.49)$$

Quando o algoritmo LMS atinge a convergência, $H(z)=0$ e desde que a entrada $X(z)$ não seja nula, a função estimativa do caminho secundário:

$$\hat{S}(z) = S(z) + \frac{P(z)}{W(z)} \quad (4.50)$$

Da equação (4.50) pode-se ver que o estimador $\hat{S}(z)$ possui um erro sistemático $P(z)/W(z)$. O filtro adaptativo pode estimar corretamente o caminho secundário somente quando $P(z)$ for nulo. Esta solução não possui sentido, já que não haveria ruído a cancelar.

A equação (4.9) determinou o resultado ótimo para o filtro adaptativo. Substituindo esta equação na (4.50), tem que $\hat{S}(z)=0$. Outra solução que não possui sentido físico.

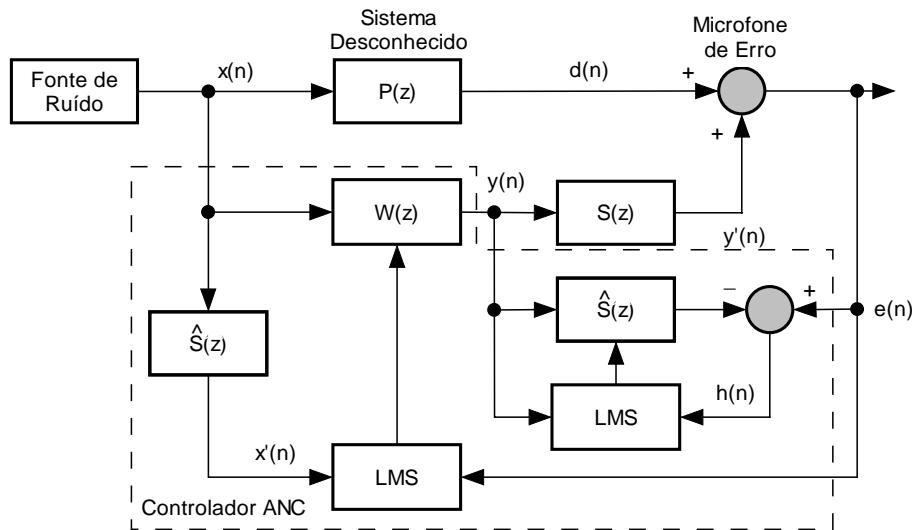


Figura 4.11 - Identificação *on-line* do caminho secundário através da modelagem direta.

Algoritmo do método direto para a estimativa *on-line* do caminho secundário:

1. Inicialização:

Coefficientes: $\hat{S}(i)_0 = 0$, N+1 coeficientes

Coefficientes: $w(i)_0 = 0$, M+1 coeficientes

Potência: $\sigma_x^2(0) = 1$ e $\sigma_y^2(0) = 1$

Parâmetro α

2. Leitura de $x(n)$ e $e(n)$.

3. Cálculo da saída do filtro adaptativo do caminho primário:

$$y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.51)$$

4. Cálculo da entrada filtrada:

$$x'(n) = \sum_{i=0}^N \hat{S}(i)_n x(n-i) \quad (4.52)$$

5. Estimação da potência de $x'(k)$:

$$\sigma_x^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma_x^2(n-1) \quad (4.53)$$

6. Cálculo do passo de adaptação:

$$0 < \mu_w < \frac{1}{(M+N+2)\sigma_x^2(n)} \quad (4.54)$$

7. Atualização dos coeficientes:

$$w(i)_{n+1} = w(i)_n - 2\mu_w e(n)x'(n-i) \quad (4.55)$$

8. Cálculo da saída do filtro adaptativo do caminho secundário:

$$y'(n) = \sum_{i=0}^N \hat{s}(i)_n y(n-i) \quad (4.56)$$

9. Compute a diferença:

$$f(n) = e(n) - y'(n) \quad (4.57)$$

10. Estimação da potência de $y(n)$:

$$\sigma_y^2(n) = \alpha y^2(n) + (1-\alpha)\sigma_y^2(n-1) \quad (4.58)$$

11. Cálculo do passo de adaptação:

$$0 < \mu_s < \frac{1}{(N+1)\sigma_y^2(n)} \quad (4.59)$$

12. Atualização dos coeficientes:

$$\hat{s}(i)_{n+1} = \hat{s}(i)_n + 2\mu_s f(n)y(n-i) \quad (4.60)$$

13. Faça $n=n+1$ e volte ao ponto 2.

4.5.2.2 Proposta de Fan

Esta modelagem proposta por Fan em 1990 é uma variante do algoritmo LMS, onde neste caso, o método converge através do próprio sinal de erro acústico $e(n)$ e não com $h(n)$. Desta forma a função de transferência só converge para os valores ótimos quando o algoritmo adaptativo de entrada atenua o ruído acústico. O esquema proposto pode ser visto na Figura 4.12.

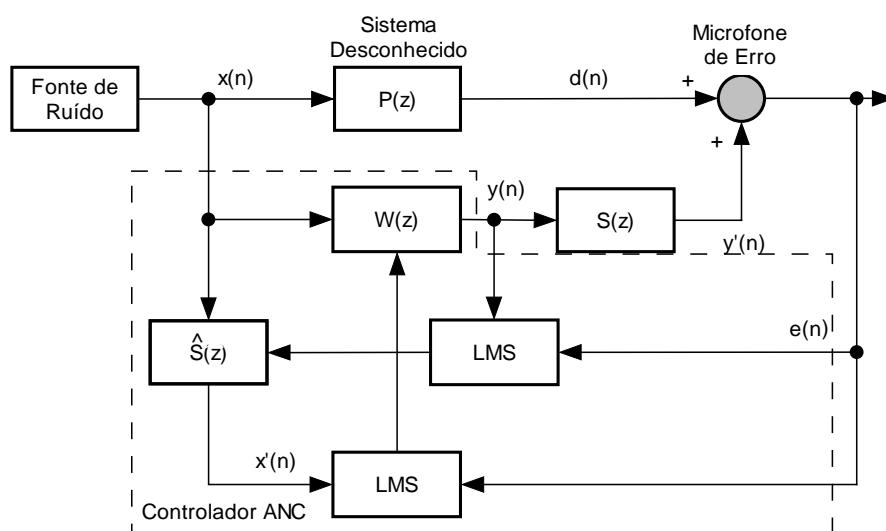


Figura 4.12 - Modelagem do caminho secundário através da proposta de H. Fan.

Algoritmo de Fan para a estimativa *on-line* do caminho secundário:

1. Inicialização:

Coefficientes: $\hat{s}(i)_0 = 0$, N+1 coeficientes

Coefficientes: $w(i)_0 = 0$, M+1 coeficientes

Potência: $\sigma_x^2(0) = 1$ e $\sigma_y^2(0) = 1$

Parâmetro α

2. Leitura de $x(n)$ e $e(n)$.

3. Cálculo da saída do filtro adaptativo do caminho primário:

$$y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.61)$$

4. Cálculo da entrada filtrada:

$$x'(n) = \sum_{i=0}^N \hat{s}(i)_n x(n-i) \quad (4.62)$$

5. Estimação da potência de $x'(k)$:

$$\sigma_x^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma_x^2(n-1) \quad (4.63)$$

6. Cálculo do passo de adaptação:

$$0 < \mu_w < \frac{1}{(M+N+2)\sigma_x^2(n)} \quad (4.64)$$

7. Atualização dos coeficientes:

$$w(i)_{n+1} = w(i)_n - 2\mu_w e(n)x'(n-i) \quad (4.65)$$

8. Estimação da potência de $y(n)$:

$$\sigma_y^2(n) = \alpha y^2(n) + (1-\alpha)\sigma_y^2(n-1) \quad (4.66)$$

9. Cálculo do passo de adaptação:

$$0 < \mu_s < \frac{1}{(N+1)\sigma_y^2(n)} \quad (4.67)$$

10. Atualização dos coeficientes:

$$\hat{s}(i)_{n+1} = \hat{s}(i)_n + 2\mu_s e(n)y(n-i) \quad (4.68)$$

11. Faça $n=n+1$ e volte ao ponto 2.

4.5.2.3 Proposta de Eriksson

Outra maneira de se estimar o caminho secundário $S(z)$ é por meio da adição de um ruído aleatório como mostra a Figura 4.13. Um ruído branco de média nula é internamente gerado e adicionado ao sinal de saída que passa pelo caminho secundário. Como ambos os ruídos são não-correlacionados, é possível utilizar o algoritmo LMS tradicional. Esta técnica foi desenvolvida por Eriksson em 1989

Os coeficientes de $\hat{S}(z)$ são otimizados para extrair de $e(n)$ a parte do sinal correlacionada com o ruído gerado pelo sistema. Desta forma $\hat{S}(z)$ tende a buscar $S(z)$ sem nenhum tipo de erro sistemático. O principal problema desta configuração é que se torna necessário introduzir uma potência de ruído similar ao nível de ruído acústico que se deseja eliminar. Como a potência do ruído aditivo é conhecida, então não é necessário estimar a mesma através do processo recursivo. Com o número de coeficientes do filtro adaptativo

$\hat{S}(z)$ é possível determinar o passo de adaptação através da equação tradicionalmente conhecida.

Se o ruído primário $x(n)$ e o anti-ruído $y(n)$ não estiverem presentes, $S(z)$ pode ser estimado off-line através do algoritmo LMS padrão, isto é:

$$\hat{s}_i(n+1) = \hat{s}_i(n) + 2\mu_s e_s(n)v(n-i) \quad (4.69)$$

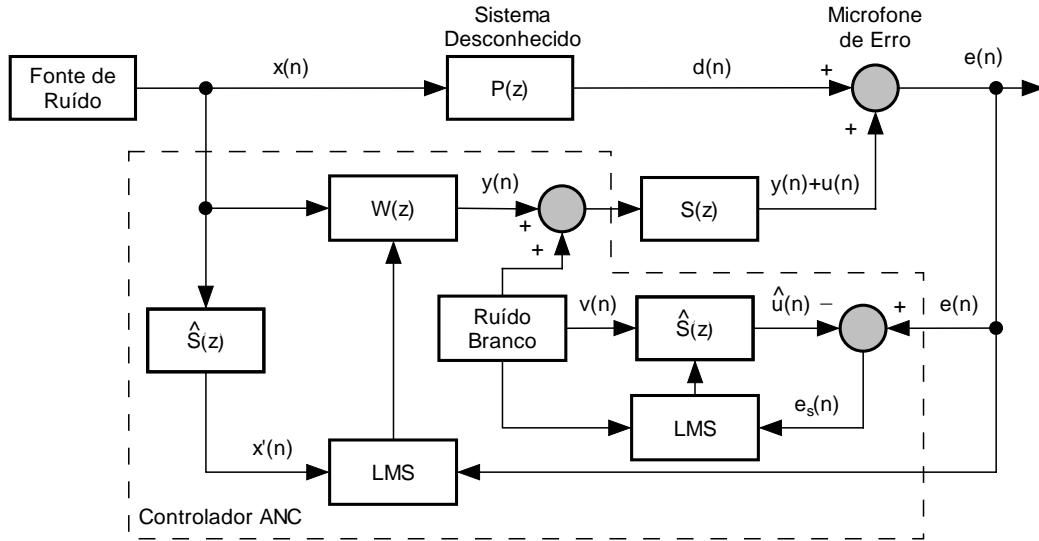


Figura 4.13 - Estimativa on-line de $S(z)$ através da técnica de Eriksson.

Onde $e_s(n) = u(n) - \hat{u}(n)$, $u(n) = s(n) * v(n)$ é a saída do caminho secundário e $\hat{u}(n) = \hat{s}(n) * v(n)$ é a saída do filtro adaptativo que estima o caminho secundário. A perfeita solução ocorre quando $e_s(n) \approx 0$.

Com a presença de $x(n)$ e $y(n)$, o sinal de $e_s(n)$ é dado por:

$$e_s(n) = e(n) - \hat{u}(n) = [d(n) + s(n) * y(n)] - \hat{u}(n) \quad (4.70)$$

Agora, a equação de atualização se torna:

$$\hat{s}_i(n+1) = \hat{s}_i(n) + 2\mu_s v(n-i)[s(n) * v(n) - \hat{s}(n) * v(n)] + 2\mu_s v(n-i)\eta(n) \quad (4.71)$$

onde $\eta(n) = d(n) + s(n) * y(n)$. Pode-se demonstrar que o valor esperado da equação acima converge para a solução ótima, visto que $v(n)$ e $\eta(n)$ são independentes e o valor esperado de $v(n)$ é igual a zero. Pode-se notar que o último termo é uma perturbação que degrada a performance de convergência da modelagem. No pior caso, este processo de estimativa pode ser divergente (Kuo et al., 1999).

Na estimativa de $W(z)$, a equação de adaptação é dada por:

$$w_i(n+1) = w_i(n) - 2\mu_w e(n)x'(n-i) \quad (4.72)$$

onde o erro, quando não há injeção de ruído branco é dado por:

$$e(n) = d(n) + s(n) * y(n) \quad (4.73)$$

No caso ideal, quando $\hat{S}(z) = S(z)$ um cancelamento perfeito é alcançado. No caso prático quando $\hat{S}(z)$ é desconhecido e deve ser estimado em tempo real enquanto o controlador estiver operando, o ruído branco $v(n)$ dever ser injetado no sistema, isto é:

$$e(n) = d(n) + s(n) * y(n) + u(n) \quad (4.74)$$

Neste caso a parcela de adaptação:

$$w_i(n+1) = w_i(n) - 2\mu_w x'(n-i)[d(n) + s(n) * y(n)] - 2\mu_w x'(n-i)u(n) \quad (4.75)$$

Como $u(n)$ é não-correlacionado com o ruído primário $x(n)$, pode-se dizer que esta parcela perturba o processo iterativo de $W(z)$.

Algoritmo de Eriksson para a estimativa *on-line* do caminho secundário:

1. Inicialização:

Coefficientes: $\hat{s}(i)_0 = 0$, N+1 coeficientes

Coefficientes: $w(i)_0 = 0$, M+1 coeficientes

Potência: $\sigma_x^2(0) = 1$

Parâmetros α

2. Leitura de $x(n)$ e $e(n)$.

3. Cálculo da saída do filtro adaptativo do caminho primário:

$$y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.76)$$

4. Cálculo da entrada filtrada:

$$x'(n) = \sum_{i=0}^N \hat{s}(i)_n x(n-i) \quad (4.77)$$

5. Estimação da potência de $x'(n)$:

$$\sigma_x^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma_x^2(n-1) \quad (4.78)$$

6. Cálculo do passo de adaptação:

$$0 < \mu_w < \frac{1}{(M+N+2)\sigma_x^2(n)} \quad (4.79)$$

7. Atualização dos coeficientes:

$$w(i)_{n+1} = w(i)_n - 2\mu_w e(n)x'(n-i) \quad (4.80)$$

8. Adição do ruído branco:

$$y(n) = y(n) + v(n) \quad (4.81)$$

9. Cálculo da saída do filtro adaptativo do caminho secundário:

$$\hat{u}(n) = \sum_{i=0}^N \hat{s}(i)_n v(n-i) \quad (4.82)$$

10. Compute a diferença:

$$e_s(n) = e(n) - \hat{u}(n) \quad (4.83)$$

11. Atualização dos coeficientes:

$$\hat{s}(i)_{n+1} = \hat{s}(i)_n + 2\mu_s e_s(n)v(n-i) \quad (4.84)$$

12. Faça $n=n+1$ e volte ao ponto 2.

4.5.2.4 Proposta de Bao

O método proposto por Bao em 1993 também trata da modelagem *on-line* do caminho secundário através da injeção de ruído branco aditivo. Esta proposta trata da redução da interferência indesejada introduzida no controlador por $\eta(n)$ segundo a equação (4.71) na estimativa de $S(z)$. O diagrama de blocos pode ser visto na Figura 4.14.

Este método utiliza um filtro adaptativo adicional, $B(z)$, para cancelar a interferência introduzida na estimativa de $S(z)$. $B(z)$ é excitado pelo ruído primário $x(n)$. A taxa de convergência da modelagem é melhorada de forma considerável. Entretanto, o efeito da perturbação causado por $v(n)$ em $W(z)$ não é considerado.

Algoritmo de Bao para a estimativa *on-line* do caminho secundário:

1. Inicialização:

Coeficientes: $\hat{s}(i)_0 = 0$, $N+1$ coeficientes

Coeficientes: $w(i)_0 = 0$, $M+1$ coeficientes

Coeficientes: $b(i)_0 = 0$, $L+1$ coeficientes

Potência: $\sigma_x^2(0) = 1$ e $\sigma_{xx}^2(0) = 1$

Parâmetros α

2. Leitura de $x(n)$ e $e(n)$.

3. Cálculo da saída do filtro adaptativo do caminho primário:

$$y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.85)$$

4. Cálculo da entrada filtrada:

$$x'(n) = \sum_{i=0}^N \hat{s}(i)_n x(n-i) \quad (4.86)$$

5. Estimação da potência de $x'(n)$:

$$\sigma_x^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma_x^2(n-1) \quad (4.87)$$

6. Cálculo do passo de adaptação:

$$0 < \mu_w < \frac{1}{(M+N+2)\sigma_x^2(n)} \quad (4.88)$$

7. Atualização dos coeficientes:

$$w(i)_{n+1} = w(i)_n - 2\mu_w e(n)x'(n-i) \quad (4.89)$$

8. Adição do ruído branco:

$$y(n) = y(n) + v(n) \quad (4.90)$$

9. Cálculo do sinal de erro compensado:

$$g(n) = e(n) - \sum_{i=0}^L b(i)_n x(n-i) \quad (4.91)$$

10. Estimação da potência de $x(n)$:

$$\sigma_{xx}^2(n) = \alpha x^2(n) + (1-\alpha)\sigma_{xx}^2(n-1) \quad (4.92)$$

11. Cálculo do passo de adaptação:

$$0 < \mu_b < \frac{1}{(L+1)\sigma_{xx}^2(n)} \quad (4.93)$$

12. Atualização dos coeficientes:

$$b(i)_{n+1} = b(i)_n + 2\mu_b g(n)x(n-i) \quad (4.94)$$

13. Cálculo da saída do filtro adaptativo do caminho secundário:

$$\hat{u}(n) = \sum_{i=0}^N \hat{s}(i)_n v(n-i) \quad (4.95)$$

14. Compute a diferença:

$$e_s(n) = g(n) - \hat{u}(n) \quad (4.96)$$

15. Atualização dos coeficientes:

$$\hat{s}(i)_{n+1} = \hat{s}(i)_n + 2\mu_s e_s(n)v(n-i) \quad (4.97)$$

16. Faça $n=n+1$ e volte ao ponto 2.

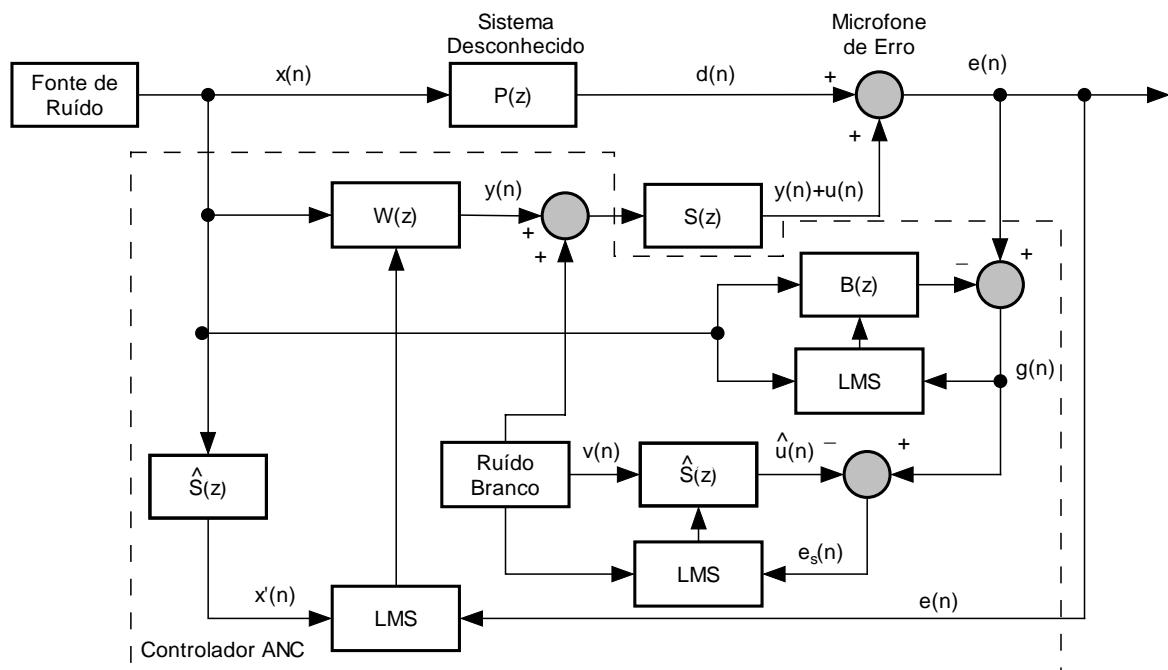


Figura 4.14 - Estimativa on-line de $S(z)$ através da técnica de Bao.

4.5.2.5 Proposta de Kuo

O conceito básico do algoritmo proposto por Kuo em 1997 é eliminar a interferência $\eta(n)$ que aparece na equação (4.71) que afeta a convergência na modelagem do filtro $\hat{S}(z)$. Devido ao fato da interferência ser variante no tempo, um preceptor adaptativo é utilizado para seguir mudanças e diminuir seus efeitos. O sistema completo de controle ativo de ruído com um preceptor adaptativo otimizado para modelagem em tempo real pode ser visto na Figura 4.15 (Kuo *et al.*, 1997).

O atraso Δ é utilizado para garantir que o ruído branco usado para a modelagem seja não-correlacionado, enquanto que as componentes do ruído primário permanecem correlacionadas com a interferência $\eta(n)$.

Assumindo que o caminho $S(z)$ seja modelado por um filtro FIR de ordem M , desde que o sinal de treino $v(n)$ possua média nula, Kuo demonstrou que atraso deve satisfazer:

$$\Delta \geq M \quad (4.98)$$

Desta forma, o sinal de treino em $e(n)$ e em $e(n-\Delta)$ serão não-correlacionados. Como consequência, o filtro preceptor $B(z)$ não será capaz de cancelar o sinal de treino. Então, a saída do filtro de erro será aproximadamente:

$$G(z) \approx S(z)V(z) \quad (4.99)$$

onde as interferências $d(n)$ e $y(n)$ estão ausentes, prevenindo a modelagem de $S(z)$ de ser degradada pelas componentes do ruído primário.

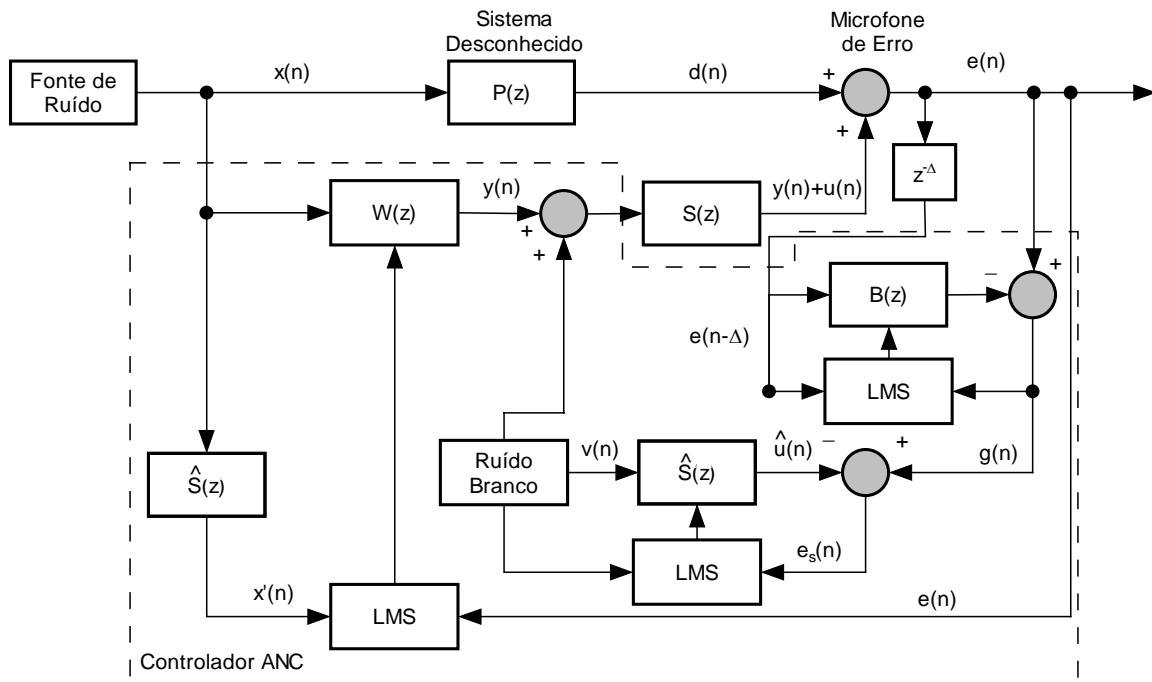


Figura 4.15 - Estimativa *on-line* de $S(z)$ através da técnica de Kuo.

Algoritmo de Kuo para a modelagem *on-line* do caminho secundário:

1. Inicialização:
 - Coefficientes: $\hat{s}(i)_0 = 0$, N+1 coeficientes
 - Coefficientes: $w(i)_0 = 0$, M+1 coeficientes
 - Coefficientes: $b(i)_0 = 0$, L+1 coeficientes
 - Potência: $\sigma_v^2(0) = 1$ e $\sigma_e^2(0) = 1$

Parâmetros α 2. Leitura de $x(n)$ e $e(n)$. 3. Cálculo da saída do filtro adaptativo do caminho primário: $y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.100)$
4. Cálculo da entrada filtrada: $x'(n) = \sum_{i=0}^N \hat{s}(i)_n x(n-i) \quad (4.101)$
5. Estimação da potência de $x'(n)$: $\sigma_x^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma_x^2(n-1) \quad (4.102)$
6. Cálculo do passo de adaptação: $0 < \mu_w < \frac{1}{(M+N+2)\sigma_x^2(n)} \quad (4.103)$
7. Atualização dos coeficientes: $w(i)_{n+1} = w(i)_n - 2\mu_w e(n)x'(n-i) \quad (4.104)$
8. Adição do ruído branco: $y(n) = y(n) + v(n) \quad (4.105)$
9. Cálculo do sinal de erro compensado: $g(n) = e(n) - \sum_{i=0}^L b(i)_n e(n-i-\Delta) \quad (4.106)$
10. Estimação da potência de $e(n-\Delta)$: $\sigma_e^2(n) = \alpha e^2(n-\Delta) + (1-\alpha)\sigma_e^2(n-1) \quad (4.107)$
11. Cálculo do passo de adaptação: $0 < \mu_b < \frac{1}{(L+1)\sigma_e^2(n)} \quad (4.108)$
12. Atualização dos coeficientes: $b(i)_{n+1} = b(i)_n + 2\mu_b g(n)e(n-i-\Delta) \quad (4.109)$
13. Cálculo da saída do filtro adaptativo do caminho secundário: $\hat{u}(n) = \sum_{i=0}^N \hat{s}(i)_n v(n-i) \quad (4.110)$
14. Compute a diferença: $e_s(n) = g(n) - \hat{u}(n) \quad (4.111)$
15. Atualização dos coeficientes: $\hat{s}(i)_{n+1} = \hat{s}(i)_n + 2\mu_s e_s(n)v(n-i) \quad (4.112)$
16. Faça $n=n+1$ e volte ao ponto 2.

4.5.2.6 Proposta de Zhang

Para o controle ativo de ruído, o perfeito cancelamento é conseguido quando a saída do filtro adaptativo é oposta ao sinal desejado. Desta forma, componentes no sinal desejado que não sejam correlacionados com o sinal de referência degradam a performance do sistema. No método de Bao existem três filtros adaptativos $W(z)$, $\hat{S}(z)$ e $B(z)$. O primeiro filtro é utilizado para cancelar o ruído primário que chega ao microfone de erro. O segundo filtro é

utilizado para modelar o caminho secundário. O terceiro é utilizado para reduzir a interferência introduzida por $\eta(n)$ na modelagem do caminho secundário.

Uma observação detalhada nos métodos de Bao e Eriksson mostra que o sinal $e(n)$ que atua como sinal de erro para o filtro $W(z)$ e simultaneamente como sinal desejado para $B(z)$, tem uma perturbação causada por $u(n)$. Esta perturbação degrada a convergência dos filtros $W(z)$ e $B(z)$. Desta forma a perturbação em $B(z)$ também perturba o sinal $g(n)$.

Zhang em 2001 propôs estágio para reduzir estas perturbações que pode ser visto na Figura 4.16. Neste sistema, o sinal de saída $\hat{u}(n) = \hat{s}(n) * v(n)$ do filtro $\hat{S}(z)$ é subtraído do sinal de erro $e(n)$ para produzir $e'(n) = e(n) - \hat{s}(n) * v(n)$, onde $e(n) = d(n) + s(n) * y(n) + s(n) * v(n)$.

Então:

$$e'(n) = d(n) + s(n) * y(n) + s(n) * v(n) - \hat{s}(n) * v(n) \quad (4.113)$$

No caso ideal, quando $\hat{S}(z) \approx S(z)$ o sinal se torna totalmente correlacionado com $x(n)$ e a perturbação devido a $v(n)$ é totalmente removida:

$$e'(n) = d(n) + s(n) * y(n) \quad (4.114)$$

Pode-se notar que o sinal $e'(n)$ é utilizado como sinal de erro para o filtro $W(z)$ e como sinal desejado para o filtro $B(z)$. O sinal $g(n)$ ainda continua sendo obtido através da diferença entre o sinal $e(n)$ e a saída o filtro auxiliar $B(z)$.

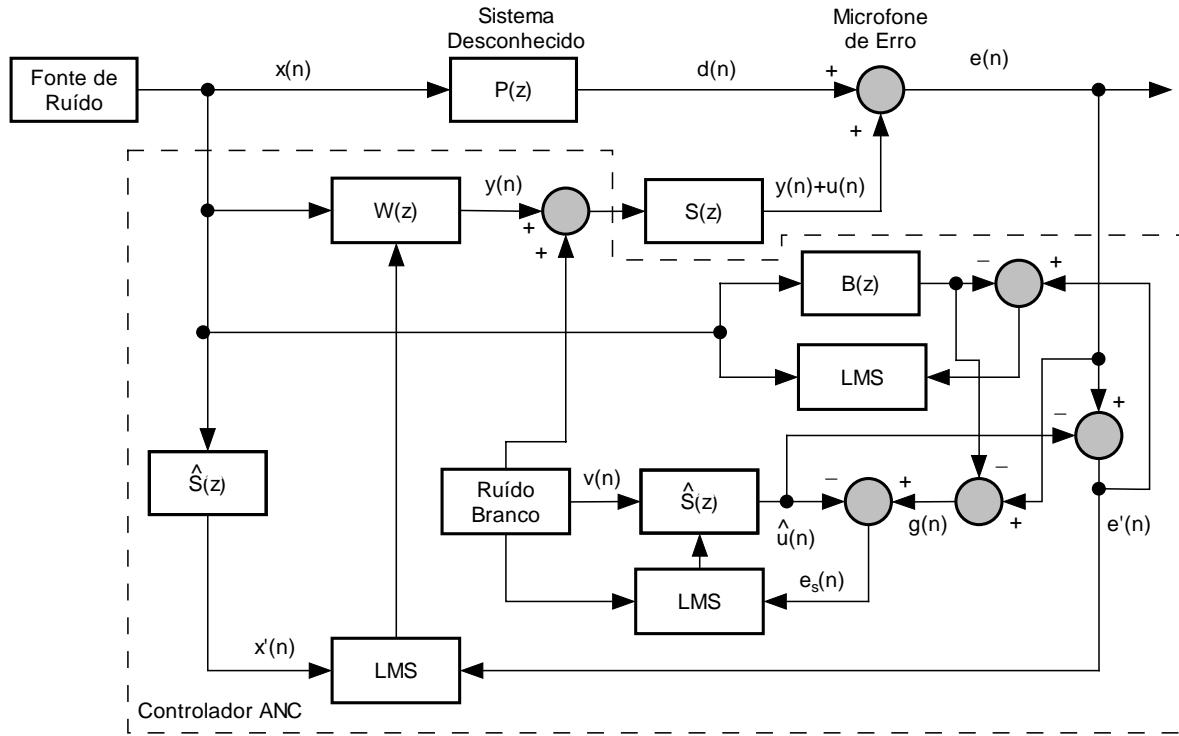


Figura 4.16 - Estimativa on-line de $S(z)$ através da técnica de Zhang.

Algoritmo de Zhang para a modelagem on-line do caminho secundário:

1. Inicialização:

Coeficientes: $\hat{S}(i)_0 = 0$, $N+1$ coeficientes

Coeficientes: $w(i)_0 = 0$, $M+1$ coeficientes

Coeficientes: $b(i)_0 = 0$, $L+1$ coeficientes

Potência: $\sigma_x^2(0) = 1$ e $\sigma_{xx}^2(0) = 1$

Parâmetros α

2. Leitura de $x(n)$ e $e(n)$.

3. Cálculo da saída do filtro adaptativo do caminho primário:

$$y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.115)$$

4. Cálculo da entrada filtrada:

$$x'(n) = \sum_{i=0}^N \hat{S}(i)_n x(n-i) \quad (4.116)$$

5. Cálculo da saída do filtro adaptativo do caminho secundário:

$$\hat{u}(n) = \sum_{i=0}^N \hat{S}(i)_n v(n-i) \quad (4.117)$$

6. Compute a diferença:

$$e'(n) = e(n) - \hat{u}(n) \quad (4.118)$$

7. Estimação da potência de $x'(n)$:

$$\sigma_x^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma_x^2(n-1) \quad (4.119)$$

8. Cálculo do passo de adaptação:

$$0 < \mu_w < \frac{1}{(M+N+2)\sigma_x^2(n)} \quad (4.120)$$

9. Atualização dos coeficientes:

$$w(i)_{n+1} = w(i)_n - 2\mu_w e'(n)x'(n-i) \quad (4.121)$$

10. Adição do ruído branco:

$$y(n) = y(n) + v(n) \quad (4.122)$$

11. Cálculo da saída do filtro auxiliar:

$$z(n) = \sum_{i=0}^L b(i)_n x(n-i) \quad (4.123)$$

12. Estimação da potência de $x(n)$:

$$\sigma_{xx}^2(n) = \alpha x^2(n) + (1-\alpha)\sigma_{xx}^2(n-1) \quad (4.124)$$

13. Cálculo do passo de adaptação:

$$0 < \mu_b < \frac{1}{(L+1)\sigma_{xx}^2(n)} \quad (4.125)$$

14. Atualização dos coeficientes:

$$b(i)_{n+1} = b(i)_n + 2\mu_b [e'(n) - z(n)]x(n-i) \quad (4.126)$$

15. Cálculo do sinal de erro compensado:

$$e_s(n) = g(n) - \hat{u}(n) = e(n) - z(n) - \hat{u}(n) \quad (4.127)$$

16. Atualização dos coeficientes:

$$\hat{s}(i)_{n+1} = \hat{s}(i)_n + 2\mu_s e_s(n)v(n-i) \quad (4.128)$$

17. Faça $n=n+1$ e volte ao ponto 2.

4.6 Modelagem do Caminho de Realimentação

4.6.1 Estimativa off-line do Caminho de Realimentação

O esquema experimental para a modelagem *off-line* do caminho de retorno é bastante similar ao mostrado na Figura 4.10, porém agora ao invés de utilizar o microfone de erro, utiliza-se o microfone de referência, já que o caminho a ser estimado está entre o alto-falante de cancelamento e o sensor de entrada. Assim, mede-se o sinal $x(n)$ e utiliza-se o filtro $\hat{F}(z)$ proposto, onde um ruído branco não correlacionado é gerado internamente no DSP. O algoritmo sugerido é bastante similar ao apresentado para a estimativa *off-line* do caminho secundário.

Algoritmo para a estimativa *off-line* do caminho de retorno:

1. Inicialização:

Coeficientes: $\hat{f}(i)_0 = 0$, $L+1$ coeficientes

2. Gerar uma amostra do ruído branco $y(n)$. Este ruído gerado internamente é utilizado como referência para o filtro adaptativo $D(z)$ e a adaptação de seus coeficientes.

3. Cálculo da saída do filtro adaptativo:

$$r(n) = \sum_{i=0}^L \hat{f}(i)_n y(n-i) \quad (4.129)$$

- | | |
|--|--------------------|
| 4. Compute a diferença:
$x''(n) = x(n) - r(n)$
5. Atualização dos coeficientes:
$\hat{f}(i)_{n+1} = \hat{f}(i)_n + 2\mu x''(n)y(n-i)$
6. Faça $n=n+1$ e volte ao ponto 2.
7. Repita o procedimento por cerca de 10 segundos. Salve os coeficientes do filtro adaptativo $\hat{F}(z)$ e utilize no algoritmo FBFXLMS dado anteriormente. | (4.130)
(4.131) |
|--|--------------------|

Como no caminho secundário, esta modelagem também pode ser realizada através de um pacote de identificação de sistemas, como alguns oferecidos pelo MatLab® (Delfino *et al.*, 2005, Nuñez *et al.*, 2004). Após a identificação, os coeficientes são transferidos para o DSP e utilizados no controlador.

4.6.2 Estimativa *on-line* do Caminho de Realimentação

Como discutido anteriormente, da mesma forma que os caminhos primário e secundário podem ser não-estacionários, o caminho de retorno também pode ter suas características físicas alteradas ao longo do tempo. Desta forma, faz-se necessária a estimação em tempo real deste caminho. A seguir dois métodos são apresentados para solucionar este problema.

4.6.2.1 Método Direto (Algoritmo LMS)

A solução mais imediata para se estimar o caminho de retorno em tempo real é utilizar a configuração clássica de identificação de modelos através da modelagem direta. Neste caso Poole em 1984 propôs o algoritmo LMS Modificado. Este esquema trata não somente dos problemas do caminho de retorno, mas também do caminho secundário. Aqui será tratada somente a modelagem *on-line* do caminho de retorno, de acordo com a Figura 4.17.

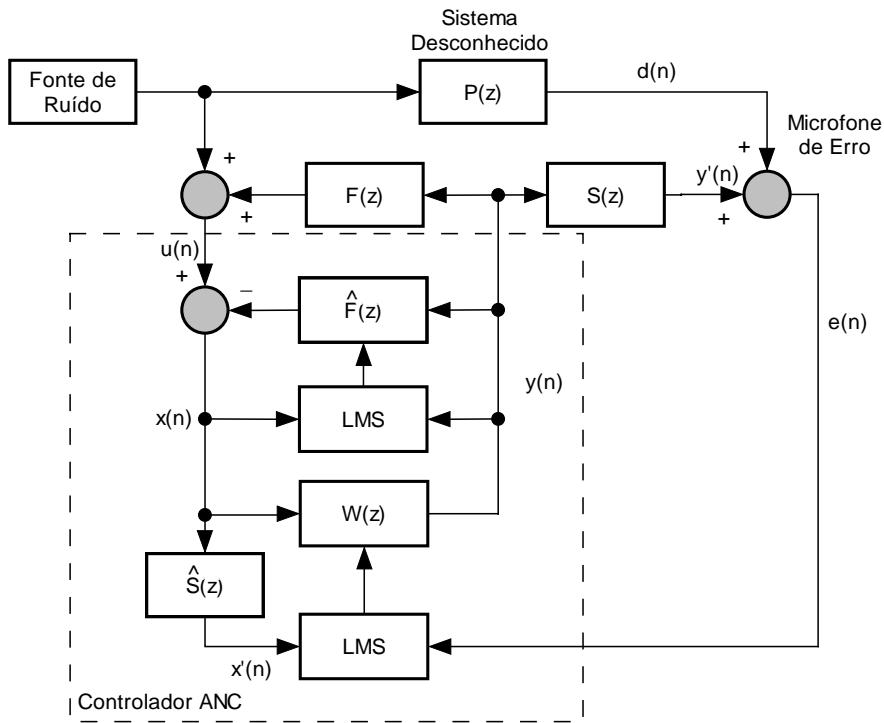


Figura 4.17 - Identificação *on-line* do caminho de retorno através da modelagem direta.

Algoritmo do método direto para a estimativa *on-line* do caminho de realimentação:

1. Inicialização:

Coeficientes: $\hat{f}(i)_0 = 0$, L+1 coeficientes

Coeficientes: $w(i)_0 = 0$, M+1 coeficientes

Potência: $\sigma_x^2(0) = 1$ e $\sigma_y^2(0) = 1$

Parâmetro α

2. Leitura de $u(n)$ e $e(n)$.

3. Cálculo da entrada livre da realimentação:

$$x(n) = u(n) - \sum_{i=0}^L \hat{f}(i)_n y(n-i-1) \quad (4.132)$$

4. Cálculo da saída do filtro adaptativo do caminho primário:

$$y(n) = \sum_{i=0}^M w(i)_n x(n-i) \quad (4.133)$$

5. Cálculo da entrada filtrada:

$$x'(n) = \sum_{i=0}^N \hat{s}(i)_n x(n-i) \quad (4.134)$$

6. Estimação da potência de $x'(k)$:

$$\sigma_x^2(n) = \alpha x'^2(n) + (1-\alpha) \sigma_x^2(n-1) \quad (4.135)$$

7. Cálculo do passo de adaptação:

$$0 < \mu_x < \frac{1}{(M+N+2)\sigma_x^2(n)} \quad (4.136)$$

8. Atualização dos coeficientes:

$$w(i)_{n+1} = w(i)_n - 2\mu_x e(n)x'(n-i) \quad (4.137)$$

9. Estimação da potência de $y(n)$:

$$\sigma_y^2(n) = \alpha y^2(n) + (1-\alpha)\sigma_y^2(n-1) \quad (4.138)$$

10. Cálculo do passo de adaptação:

$$0 < \mu_y < \frac{1}{(L+1)\sigma_y^2(n)} \quad (4.139)$$

11. Atualização dos coeficientes:

$$\hat{f}(i)_{n+1} = \hat{f}(i)_n + 2\mu_y x(n)y(n-i) \quad (4.140)$$

12. Faça $n=n+1$ e volte ao ponto 2.

4.6.2.2 Algoritmo Adaptativo FURLMS

O filtro adaptativo do tipo IIR foi proposto por Eriksson em 1987 e é conhecido como FURLMS (*Filtered-U Recursive LMS*). Este esquema considera o retorno acústico como parte da planta inteira e os pólos introduzidos pela realimentação são modelados pelos pólos do filtro IIR. O diagrama de blocos do sistema pode ser visto na Figura 4.18.

Este sistema de controle consegue atualizar dinamicamente as mudanças ocorridas nos caminhos durante o processo de cancelamento do ruído. A estrutura IIR tem a habilidade modelar funções de transferência direta com pólos e zeros.

Após a convergência de $A(z)$ e $B(z)$, deseja-se que o erro residual $e(n)$ seja nulo, assim:

$$W(z) = \frac{A(z)}{1 - B(z)} = \frac{-P(z)}{S(z) - P(z)F(z)} \quad (4.141)$$

Dada as complexidades e a estrutura de pólos e zeros de $P(z)$, $S(z)$ e $F(z)$, a convergência de $A(z)$ e $B(z)$ não leva a uma única solução, entretanto o algoritmo irá convergir para a solução que minimiza o sinal do erro residual $e(n)$. Baseada na equação (4.141), uma possível solução seria:

$$\begin{aligned} A(z) &= \frac{-P(z)}{S(z)} \\ B(z) &= \frac{P(z)F(z)}{S(z)} \end{aligned} \quad (4.142)$$

Devido ao fato das complexidades de cada solução e por questões de causalidade, é razoável usar uma ordem superior (ou no mínimo igual) para $B(z)$ que para $A(z)$. O sistema realiza a modelagem do caminho secundário segundo as propostas apontadas no item 4.5.

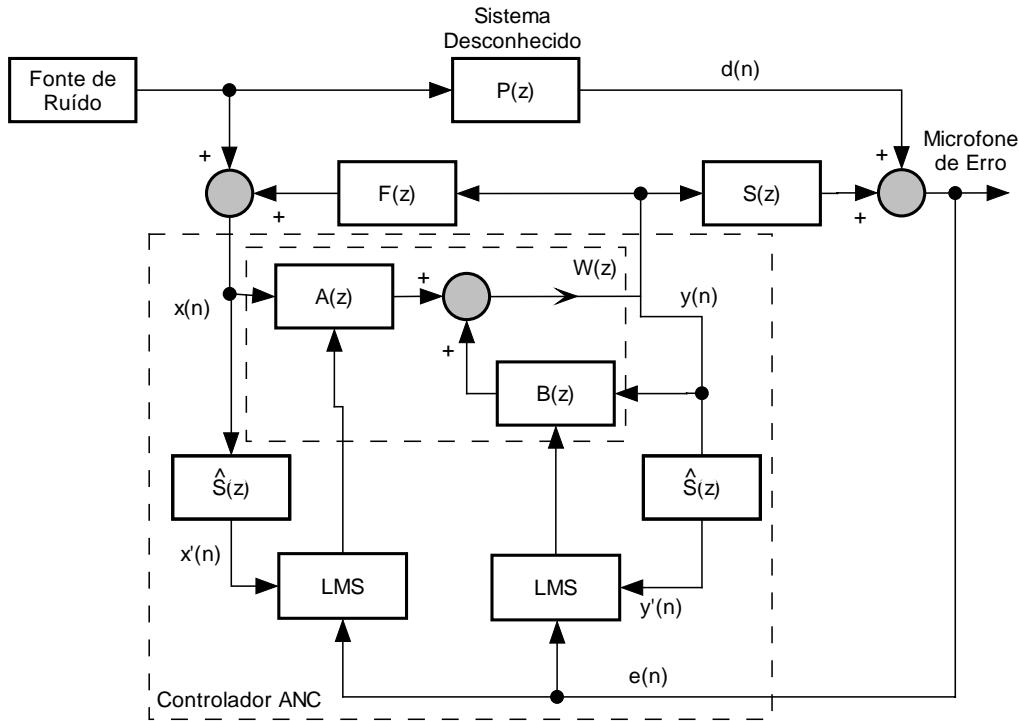


Figura 4.18 - Sistema ANC com algoritmo FURLMS.

Algoritmo FURLMS para a estimativa *on-line* do caminho de realimentação:

- Inicialização:
 - Coeficientes: $a(i)_0 = 0$, M+1 coeficientes
 - Coeficientes: $b(i)_0 = 0$, L+1 coeficientes
 - Potência: $\sigma_x^2(0) = 1$ e $\sigma_y^2(0) = 1$
 - Parâmetros: α
 - Leitura de $x(n)$ e $e(n)$.
 - Cálculo da saída do filtro adaptativo $W(z)$ do caminho primário:
$$y(n) = \sum_{i=0}^M a(i)_n x(n-i) + \sum_{j=0}^N b(j)_n y(n-j-1) \quad (4.143)$$
 - Cálculo da operação Filtrado-U:
$$y'(n) = \sum_{i=0}^N \hat{s}(i)_n y(n-i) \quad (4.144)$$

$$x'(n) = \sum_{i=0}^N \hat{s}(i)_n x(n-i) \quad (4.145)$$
 - Estimação da potência de $x'(n)$:
$$\sigma_x^2(n) = \alpha x'^2(n) + (1-\alpha)\sigma_x^2(n-1) \quad (4.146)$$
 - Cálculo do passo de adaptação:
$$0 < \mu_x < \frac{1}{(M+N+2)\sigma_x^2(n)} \quad (4.147)$$
 - Atualização dos coeficientes:
$$a(i)_{n+1} = a(i)_n - 2\mu_x e(n)x'(n-i) \quad (4.148)$$
 - Estimacão da potência de $y'(n)$:

$$\sigma_y^2(n) = \alpha y'^2(n) + (1 - \alpha)\sigma_y^2(n-1) \quad (4.149)$$

9. Cálculo do passo de adaptação:

$$0 < \mu_y < \frac{1}{(M+N+2)\sigma_y^2(n)} \quad (4.150)$$

10. Atualização dos coeficientes:

$$b(j)_{n+1} = b(j)_n - 2\mu_y e(n)y'(n-j-1) \quad (4.151)$$

11. Faça $n=n+1$ e volte ao ponto 2.

CAPÍTULO V

Projeto de Controladores Ativos de Ruídos (ANC)

O controle ativo de ruído é uma aplicação em tempo real de filtragem adaptativa que requer extensos cálculos computacionais, principalmente de convoluções de sinais. A banda de freqüência controlável varia até 500Hz (Kuo *et al.*, 1997), que pelo Teorema da Amostragem (Nyquist) leva a uma freqüência mínima de amostragem de 1000Hz, ou seja, um tempo de 1ms para realizar todos os cálculos do algoritmo. Algumas considerações na implementação eletrônica em um ANC possuem um grande impacto na performance do sistema. O sistema de *hardware* deve permitir flexibilidade de *software* assim como uma operação totalmente automática do sistema completo de controle. Os fatores físicos que limitam a performance de um ANC, tais como posicionamento dos microfones, coerência, ordem dos filtros, estabilidade e causalidade, são discutidos nesta seção.

5.1 Projeto do Sistema

5.1.1 Seleção dos Transdutores Eletroacústicos

Um controlador trabalha com sinais elétricos em sua entrada e mediante algum processo produz outros sinais elétricos em sua saída. Estes sinais representam o vínculo entre o controlador e o campo acústico. As conversões do campo acústico em sinais elétricos e vice-versa são realizados através de transdutores eletroacústicos (sensores e atuadores). Estes dispositivos devem possuir uma máxima qualidade a um mínimo custo.

A qualidade destes sensores é definida em termos da linearidade, da relação sinal/ruído e do atraso de conversão. Para o controle ativo de ruídos estes critérios devem satisfazer até a banda de 500Hz (Kuo *et al.*, 1997).

A *linearidade* é essencial para se obter um sinal que represente fielmente as ondas acústicas captadas por um microfone ou que se geram através de um alto-falante. Este tipo

de dispositivo é não-linear quanto ele apresenta distorção harmônica. Esta distorção ocorre quando um dispositivo gera freqüências harmônicas indesejadas em grande proporção em relação à freqüência fundamental. Um nível elevado de distorção nos sensores mascara o sinal que o controlador deve processar, impossibilitando a distinção entre o sinal fundamental e os seus harmônicos. No caso dos alto-falantes, estes criam novas ondas sonoras que não se cancelam, pois o controlador não possui referência das mesmas. Os valores de distorção desejáveis para dispositivos eletroacústicos devem ser inferiores a 1% (Minguez, 1998).

A relação *sinal/ruído* (*signal/noise*) dos transdutores também influencia na performance do sistema de controle. Como a distorção acústica, o ruído de fundo mascara o sinal ideal de entrada ou corrompe a conversão de saída. Relações sinal/ruído (S/N) devem ser superiores a 40dB para se garantir um bom funcionamento do controlador (Minguez, 1998). Para os sensores, à medida que a relação S/N diminui, o algoritmo adaptativo se torna incapaz de encontrar os valores ótimos dos coeficientes que garantem o cancelamento acústico, já que estes valores oscilam em torno dos valores ótimos e esta oscilação se torna maior quanto menor for a relação S/N. Da mesma forma que na distorção, os alto-falantes geram um tipo de ruído que não pode ser cancelado pelo controlador.

Grandes atrasos na conversão eletroacústica podem levar o sistema à não-causalidade. Por *causalidade* entende-se que o atraso do sinal elétrico entre o microfone de referência e o alto-falante de cancelamento deve ser menor que o atraso acústico. Atualmente, os dispositivos de conversão são bem eficientes e não representam problemas. Esta questão será abordada com maiores detalhes na próxima seção.

Para os sensores de referência e de erro podem ser utilizados simples microfones de eletreto omnidirecionais de baixo custo, já que oferecem características aceitáveis para a margem de freqüências de interesse para o controle ativo de ruído. Quando se utilizam microfones como sensores de referência, existe o problema da realimentação acústica. Para minimizar o problema, pode-se utilizar microfones direcionais.

Outra forma de eliminar o problema de realimentação seria utilizar outro tipo de transdutor. Existem muitas situações onde a fonte de ruído está disponível possibilitando a utilização de acelerômetros, por exemplo, situados diretamente sobre a fonte de ruído.

Por último, o conjunto amplificador e alto-falante de cancelamento que geram o anti-ruído acústico devem possuir potência necessária para gerar ondas de pressão acústica iguais ou maiores que as ondas do ruído primário em torno do microfone de erro.

Quando se deseja realizar algum tipo de controle de ruído acústico, deve-se conhecer bem a fonte de ruído, incluindo sua potência sonora e espectro de freqüências. É fundamental, neste caso, que o microfone de entrada não esteja localizado em um ponto

nodal (ponto de pouca ou nenhum magnitude sonora) de qualquer onda estacionária no duto antes ou durante o cancelamento. Caso isto ocorra, o controlador não irá adquirir todas informações necessárias para se realizar o controle de forma efetiva. Pontos nodais antes do cancelamento também deverão ser evitados para o microfone de erro. No caso do alto-falante de controle, este deve ser posicionado de forma que o cancelamento no microfone de erro seja alcançado com o mínimo requerimento de energia do sistema.

Os transdutores eletroacústicos utilizados em um sistema de controle ativo de ruído devem ser posicionados corretamente para que o controlador consiga a máxima atenuação possível. O posicionamento específico do alto-falante de cancelamento determina o valor máximo de atenuação. Existem procedimentos de otimização para encontrar a posição ideal (Minguez, 1998). No entanto, a função de custo para tal processo não é um problema tão trivial, já que se deve possuir uma função analítica que proporcione a impedância acústica em todos os pontos do espaço acústico considerado.

Muitas vezes, este procedimento é simplificado através da localização da fonte de controle e dos microfones em várias posições pré-determinadas. Desta forma, dados experimentais de pressão acústica são colhidos para as várias configurações. Com estes dados, determina-se a configuração ótima dentre aquelas selecionadas. Logicamente, a otimização será melhor quanto maior for o número de configurações possíveis.

5.1.2 Causalidade

Por causalidade em sistemas ANC entende-se que o atraso acústico entre o microfone de referência e o alto-falante de cancelamento deve ser maior que o atraso elétrico. Em outras palavras, pode-se dizer que a onda acústica de controle deve ser estimada antes que a onda acústica primária chegue à região de silêncio.

O atraso acústico entre o microfone de entrada e o alto-falante é dado em segundos por:

$$\partial_A = \frac{L}{C_0} \quad (5.1)$$

onde L é a distância entre o microfone e o alto-falante e C_0 é a velocidade do som, usualmente 343m/s. O atraso elétrico pode ser dado por:

$$\partial_E = \partial_W + \partial_T \quad (5.2)$$

onde ∂_T é o atraso total nos transdutores eletroacústicos, nos filtros *antialiasing* e de reconstituição, nos conversores analógico-digital (ADC) e digital-analógico (DAC) mais o

tempo de processamento (um período de amostragem). ∂_w é atraso de grupo do filtro digital $W(z)$.

Para garantir que o filtro adaptativo tenha resposta causal, então:

$$\partial_A > \partial_E \quad (5.3)$$

Então a distância mínima entre o microfone de entrada e o alto-falante de saída é:

$$L_{\min} > C_0 \partial_E \quad (5.4)$$

O atraso total é aproximadamente (Nelson *et al.*, 1992):

$$\partial_T \approx T \left(1 + \frac{3M}{8} \right) \quad (5.5)$$

onde M é número de pólos dos filtros *antialiasing* e de reconstituição e T é o período de amostragem. Por exemplo, assumindo uma freqüência de amostragem de 2KHz e que cada filtro analógico tenha 4 pólos, assim $M = 8$ e $\partial_T = 2ms$.

Para reduzir o atraso elétrico, uma alta taxa de amostragem pode ser necessária para uma dada distância. Para atenuar ruído branco em dutos, o padrão ANC é utilizar longos dutos. Para o caso de ruídos periódicos, a causalidade não é necessária, já que o sistema pode se basear em períodos anteriores e sistemas compactos são possíveis de realizar.

5.1.3 Taxa de Amostragem e Filtros Analógicos

O trabalho de um ANC é estimar precisamente as mudanças em freqüência, fase e amplitude do ruído que propaga entre o microfone de entrada e o alto-falante de cancelamento. O controlador deve completar o processamento de sinais antes que o ruído primário chegue ao alto-falante. Portanto, um sistema digital de processamento de sinais em tempo real possui um tempo de trabalho t menor que o período de amostragem T . Isto é:

$$t < T = \frac{1}{F_s} \quad (5.6)$$

Onde F_s é a freqüência de amostragem, que deve satisfazer o Critério de Nyquist dado por:

$$F_s \geq 2F_m \quad (5.7)$$

Onde F_m é a maior freqüência de interesse, no caso de sistemas ANC práticos, esta é de 500Hz. Isto leva a uma mínima freqüência de amostragem de 1000Hz e um tempo máximo de processamento de 1ms.

Como mostra a equação (5.7), para se recuperar a informação original de um sinal em sua forma amostrada, o espectro do sinal deve ser totalmente limitado em uma banda menor ou igual a metade da freqüência de amostragem. Se esta limitação em banda não é suficiente, as componentes do sinal acima deste limiar são adicionadas como freqüências fantasmas no sinal. Este fenômeno é chamado de *aliasing* e não pode ser tratado após a amostragem.

Idealmente, um filtro *antialiasing* deve possuir uma amplitude plana e fase linear na banda passante e uma atenuação infinita acima da freqüência de corte. Como em sistemas ANC a freqüência de amostragem é baixa (1KHz a 10KHz), filtros *antialiasing* de alta ordem são necessários. Porém, estes filtros de alta ordem possuem um grande atraso, conforme a equação (5.5). Isto pode criar problemas de causalidade, particularmente para sistemas de controle de banda larga em dutos pequenos.

Se um filtro de baixa ordem com uma melhor resposta em fase é utilizado, um pequeno atraso de grupo pode ser conseguido. Porém, a banda de transição do filtro terá uma atenuação menor, sendo necessária uma maior freqüência de amostragem. Este é método é conhecido como sobre-amostragem e pode ser utilizado se houver tempo suficiente para o processamento dos sinais. Lembrando que quanto maior a freqüência de amostragem, menor o tempo disponível para realizar os cálculos da lei de controle.

Um sinal contínuo pode ser recuperado sem distorção de sua versão corretamente amostrada por um filtro passa-baixa. Este tipo de dispositivo é conhecido como filtro de reconstituição, deve ter as características de resposta plana e fase linear na banda passante e atenuação infinita na banda de parada.

Apesar da utilização de filtros passa-alta com freqüência de corte próxima de 0Hz na eletrônica analógica, algum nível DC pode aparecer nos dados de entrada. O efeito deste nível no sinal é uma possível saturação, levando o sistema a instabilidade (Minguez, 1998). Outro efeito observado seria o algoritmo adaptativo tentar cancelar esta componente DC, aumentando ou diminuindo o nível DC de sua saída ao longo do tempo. No entanto, o caminho secundário geralmente não possui resposta ao nível DC, devido à resposta em freqüência dos transdutores. Este nível DC no sinal de controle pode ir aumentando até que o mesmo seja saturado.

Este nível DC pode ser eliminado através de filtro recursivo passa-alta de primeira ordem (Widrow *et al.*, 1985). De outra maneira, um filtro *notch* na freqüência 0Hz deve ser utilizado. A função de transferência deste tipo de filtro pode ser dada por:

$$G(z) = \frac{Y(z)}{X(z)} = \frac{1 - z^{-1}}{1 - (1 - \mu)z^{-1}} \quad (5.8)$$

Esta função de transferência apresenta um pólo e um zero em 0Hz. O zero sobre a circunferência de raio unitário no plano Z e o pólo muito próximo, em seu interior. A equação diferença a ser implementada no DSP é dada por:

$$y(n) = x(n) - x(n-1) + (1-\mu)y(n-1) \quad (5.9)$$

onde μ é um número menor que a unidade.

5.1.4 Limitações Práticas e Soluções

Quando o controle ativo de ruído é aplicado a sistemas de exaustão ou ventilação, há fluxo de ar ou gases no interior do duto. Esta vazão pode causar flutuações de pressão nos microfones, adicionando ruído aleatório nas medidas. Estas flutuações podem degradar a performance de cancelamento.

Este problema pode ser resolvido de duas maneiras:

- Utilizar tubos protetores que permitam a propagação do som até o microfone e evitem o fluxo de ar no interior do duto (Olson *et al.*, 1989). Entretanto, esta solução tem a desvantagem de que o microfone e seu suporte gerarem turbulências e aumentarem o ruído da vazão no microfone seguinte.
- Nishimura em 1991 mostrou que posicionar o microfone em uma pequena cavidade, fora da turbulência, ligada ao duto através de uma pequena fenda pode ser uma boa alternativa. Esta localização fora da turbulência do duto também proporciona melhor proteção e facilidade de manutenção.

As duas soluções podem ser visualizadas na Figura 5.1.

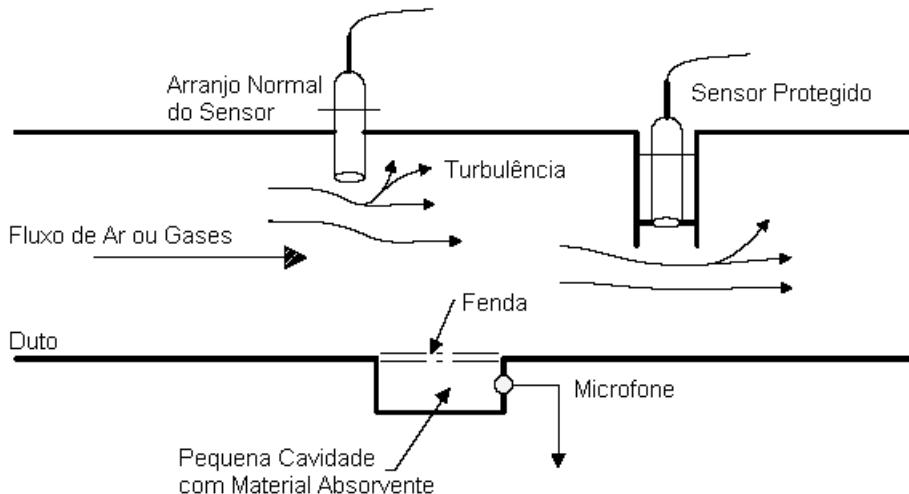


Figura 5.1 - Montagem dos microfones para redução da turbulência.

As aplicações industriais de controle ativo de ruído impõem um conjunto de limitações ao sistema. A potência do alto-falante de cancelamento devido aos altos níveis de potência sonora presentes no interior do duto está incluída nestas limitações. A proteção dos componentes devido ao ambiente agressivo também deve ser tratada. O ruído geralmente é de banda estreita ou periódico, de alta intensidade e produzido por ventiladores, sopradores, bombas de vácuo ou compressores. A taxa de vazão de gás é alta, e algumas vezes, há a presença de partículas suspensas. O ambiente apresenta altas temperaturas com fluxo de gases que muitas vezes são corrosivos e úmidos.

Estes problemas podem ser resolvidos através de uma cadeia de alto-falantes de alta potência protegidos por uma membrana que permita a transmissão de energia acústica para dentro e fora do duto. Isto mantém o ambiente limpo e seco, garantindo uma vida longa aos componentes (Burlage *et al.*, 1991). Mais de um alto-falante conectado a mesma saída do controlador pode ser utilizado para promover potência extra de saída e redundância. Sistemas industriais geralmente não são limitados em espaço físico. A proteção dos transdutores contra umidade e sujeira é essencial para garantir vida longa ao sistema.

5.2 Projeto do Software

5.2.1 Processadores Digitais de Sinais (DSPs)

Ao mesmo tempo em que as avançadas técnicas de controle estão se desenvolvendo, os projetistas possuem ferramentas efetivas para a síntese do projeto de controle digital. Desta forma, poderosas teorias de controle podem ser facilmente implementadas em aplicações práticas.

Microprocessadores, microcontroladores e processadores digitais de sinais (DSPs) têm se tornado extremamente eficientes, enquanto o custo tem se reduzido. Estes representam pequena porcentagem do custo final do projeto, quando comparado ao custo dos sensores e atuadores (Masten *et al.*, 1997).

A implementação de sistema de controle digital possui algumas vantagens em relação à implementação analógica (Masten *et al.*, 1997):

- Fácil controle de sistemas complexos, multivariáveis, lineares ou não-lineares com a leitura de vários sensores.
- Fácil ajuste e modificação dos algoritmos de controle (mudanças de *software* ao invés do *hardware*).
- Fácil incorporação de métodos avançados tais como adaptação, estimativa, otimização e identificação em tempo real.

- Formulação mais direta dos algoritmos de controle em código computacional ao invés da implementação em *hardware* analógico.
- Desenvolvimento mais rápido e barato do projeto.
- Monitoramento e manutenção aumentados de forma extraordinária.

Os microprocessadores e microcontroladores foram desenvolvidos na década de 70. Mesmo sendo bastante simples, estes promoveram muitos benefícios no controle digital. Os processadores estão divididos em duas grandes áreas: a primeira trata das manipulações de dados, tais como processadores de texto e banco de dados; a segunda trata dos cálculos matemáticos, usados em ciências, engenharia e processamento digital de sinais.

Todos microprocessadores podem realizar as duas tarefas, no entanto, se torna difícil e caro fabricar um dispositivo que seja otimizado para ambas. Há grandes diferenças no projeto do *hardware*, tal como o tamanho do conjunto de instruções. Da mesma forma que os microprocessadores, tais como o Pentium®, são construídos para grande manipulação de dados, os DSPs são projetados para realizarem cálculos matemáticos.

Estas operações aparecem com grande freqüência nos algoritmos de processamento de sinais e nos sistemas de controle moderno. O processamento digital de sinais é constituído de operações sucessivas de multiplicação e adição.

Seja, por exemplo, o cálculo de um filtro FIR.

$$y(n) = \sum_{i=0}^N h(i)x(n-i) = h(0)x(n) + h(1)x(n-1) + h(2)x(n-2) + \dots + h(N)x(n-N) \quad (5.10)$$

A expressão matemática deste tipo de filtro é uma simples convolução dos dados de entrada com os coeficientes da resposta impulsiva do filtro. Dependendo da aplicação, pode haver poucos coeficientes no filtro ou centenas deles. Apesar da transferência de dados presente na expressão, as operações matemáticas dominam o tempo de execução.

O desenvolvimento de circuitos integrados com arquitetura específica para manipulação de expressões desta natureza revolucionou, nas últimas décadas, o ramo de Processamento Digital de Sinais. Os DSPs embarcados num único chip estão mudando o processamento digital de sinais da mesma maneira que os microprocessadores mudaram o processamento de dados.

5.2.2 Benefícios dos DSPs

Atualmente, os DSPs possuem várias ferramentas e periféricos embutidos na mesma pastilha que são úteis em sistemas de controle. Com isso pode-se incorporar várias funções de controle descritas a seguir:

Processamento de Sinais: processamento adicional dos sinais provenientes dos sensores pode ser utilizado para melhorar a qualidade dos dados. Sistemas de controle com menor número de sensores também são possíveis, através da geração de estimativas sem o uso de sensores adicionais (Restle *et al.*, 1993).

Geração da entrada/trajetória: aplicações microprocessadas freqüentemente utilizam métodos de tabelas para gerar as entradas de controle. DSPs podem substituir tais tabelas por funções matemáticas específicas. Isto permite leis de controle mais complexas e diminui o uso de memória.

Leis de Controle: DSPs permitem implementações em tempo real de avançados algoritmos que resultam em melhores sistemas de controle. Sistemas com múltiplas entradas e múltiplas saídas são realizáveis por um único DSP. Controle adaptativo, controle ótimo, redes neurais, algoritmos genéticos, lógica *fuzzy*, bem como PIDs sintonizáveis podem ser realizados nestes dispositivos (Delfino *et al.*, 2005, Rajashekra *et al.*, 1996; Ahmed, 1991; Bierke, 1996; Villadiego, 2004).

Identificação de Sistemas: Em muitos sistemas de controle, é necessário estimar ou modelar alguns parâmetros do sistema antes ou durante o processo de controle. Os processadores DSP oferecem performance suficiente para tais tarefas (Villadiego, 2004).

Modulação por Largura de Pulso (PWM): Sistemas de controle, especialmente envolvendo motores elétricos são chaveados através de técnicas de PWM. Esta técnica elimina o uso de conversores D/A, reduzindo o custo do dispositivo.

Diagnósticos e proteção: Os processadores DSPs podem conter complexas rotinas de tarefas de diagnósticos e supervisão durante o processo de controle.

Controle de Ruído: Controle e redução dos níveis de ruído nos equipamentos tem se tornado importante aos projetistas. Atenuação de ruído e vibração através de técnicas ativas usando modernos algoritmos são possíveis (Delfino *et al.*, 2005).

5.2.3 Projetos Baseados em DSPs

Embora o projeto de controle digital seja de alguma forma mais complicado que o analógico, o uso de um sistema de desenvolvimento DSP acelera o tempo de projeto. Ferramentas de desenvolvimento incluindo Compiladores C, Assembler/Linker, Simuladores

e Debugadores auxiliam o processo. Muitos pacotes estão disponíveis pelos fabricantes de DSPs e companhias independentes. Ferramentas de projeto de controle tais como as presentes no MatLab® pode ser utilizados para acelerar o desenvolvimento do projeto.

O passo inicial do projeto não é diferente de um sistema de controle analógico, onde a planta é analisada e as performances desejadas do sistema são definidas. Uma estratégia de controle é escolhida com base nos custos, na performance relativa às especificações desejadas e em outros critérios. Neste estágio os detalhes de implementações não são considerados.

Tendo definida a estratégia de controle, selecionam-se os componentes para compor tal sistema. Para sistemas de controle analógico são selecionados os sensores, atuadores e os componentes analógicos para realizar o processo. Em sistemas baseados em processadores DSP, os mesmos sensores e atuadores são escolhidos, porém a plataforma DSP substitui o controle analógico. Em alguns casos, os sensores e atuadores podem ser menos precisos e de menor custo, devido à capacidade adicional de processamento digital de sinais presentes nestes sistemas.

A escolha de um DSP depende de vários fatores envolvendo o problema a ser resolvido. Dentre estes fatores pode-se citar os mais importantes na performance desejada:

- Capacidade do DSP em realizar uma ou mais operações de multiplicação acumulativa (MAC) em apenas um ciclo de instrução. Há no mercado DSPs que conseguem realizar várias destas operações simultaneamente devido à presença de várias MACs (Berkeley, 2000).
- Capacidade do processador em acessar a memória em apenas um ciclo de instrução. Isto permite que uma operação seja apontada enquanto os dados ou operandos também sejam. Por exemplo, para uma convolução, enquanto uma MAC esta sendo processada, a nova amostra do vetor de entrada está sendo adquirida (Berkeley, 2000).
- A escolha de processadores de ponto fixo ou ponto flutuante. O uso de linguagens de alto nível com ponto flutuante torna o desenvolvimento mais fácil, porém estes tipos de DSPs são mais caros, aumentando o custo final do projeto. Utilizando ponto fixo, o projetista deve desenvolver rotinas apropriadas de escalonamento para se evitar erros de truncamento e *overflow*, isto torna a implementação dos algoritmos mais complexa (Masten *et al*, 1997).
- O tamanho limitado da palavra em bits também pode causar problemas na performance do sistema. Um escalonamento deve ser realizado para se evitar os efeitos de quantização introduzidos pela resolução limitada. Os efeitos de quantização podem ocorrer tanto no processamento, quanto nas conversões A/D

e D/A. A quantização dos coeficientes armazenados na memória também pode levar algumas funções de transferência à instabilidade (Rajashekra *et al*, 1996; Ahmed, 1991).

- Por fim, um sistema eficiente de endereçamento é necessário. Alguns DSPs possuem unidade exclusiva de endereçamento para realizar tal função (Berkeley, 2000). Outros suportam a operação de reversão de bits, muito comum para implementação da Transformada Rápida de Fourier (*Fast Fourier Transform – FFT*).

Outros fatores também são importantes para a performance do sistema, tais como velocidade do processador, portais I/O, conversores A/D e D/A, dentre outros. Desta forma, pode-se ver que a escolha de um DSP requer um estudo profundo da aplicação.

5.2.4 Arquitetura dos DSPs

Para se realizar uma multiplicação em um processador, é necessário buscar três valores binário da memória: os números a serem multiplicados mais a instrução de programa descrevendo a tarefa a realizar. Esta tarefa realizada por um microprocessador tradicional é chamada de Arquitetura Von Neumann, criada pelo matemático americano John Von Neumann (1903 – 1957).

Como mostrado na Figura 5.2a, a arquitetura Von Neumann contém uma simples memória e um simples barramento para transferir dados para dentro e para fora da Unidade Central de Processamento (CPU). A multiplicação de dois números requer pelo menos três ciclos de relógio, cada um para transferir os três números no barramento da memória para a CPU. Esta arquitetura é satisfatória para realizar um processamento serial das tarefas. A maioria dos computadores atualmente possui tal arquitetura.

No entanto, quando um processamento mais veloz é necessário, deve usar outras arquiteturas. A arquitetura de Harvard, como mostrado na Figura 5.2b utiliza memórias separadas para dados e para instruções de programa com barramentos separados para cada uma. Desta maneira, com os barramentos separados, os dados e instruções podem ser acessados ao mesmo tempo, aumentando a velocidade em relação ao projeto que contenha um barramento. Atualmente, a maioria dos DSPs utilizam esta última arquitetura, incluindo os dispositivos da Texas Instruments.

5.2.5 Fator de Vazamento “Leakage”

Quando um algoritmo adaptativo é implementado em um processador de sinais com precisão limitada, principalmente naqueles de aritmética de ponto fixo, os ruídos de arredondamento e de discretização são propagados para os coeficientes do filtro. Isto pode causar um crescimento exagerado nestes coeficientes, levando o processo de controle à instabilidade (Kuo *et al.*, 1996). Uma solução comumente adotada é baseada na adição de um fator de vazamento (*leakage*) que tende a limitar o aumento dos coeficientes do filtro.

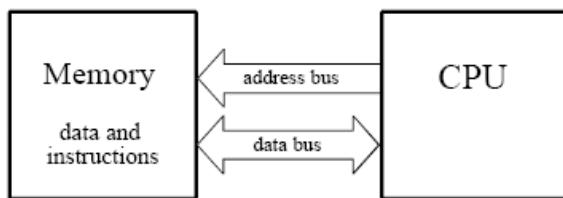
Tomando a equação do algoritmo LMS, temos que o fator é introduzido como segue:

$$w(i)_{n+1} = vw(i)_n - 2\mu e(n)x(n-i) \quad (5.11)$$

Onde v (fator de vazamento) é próximo de 1, no entanto, um pouco menor.

A única desvantagem deste fator é que muitas vezes o nível de atenuação acaba ficando comprometido, devido à redução do ganho da saída do filtro adaptativo.

a. Von Neumann Architecture (*single memory*)



b. Harvard Architecture (*dual memory*)

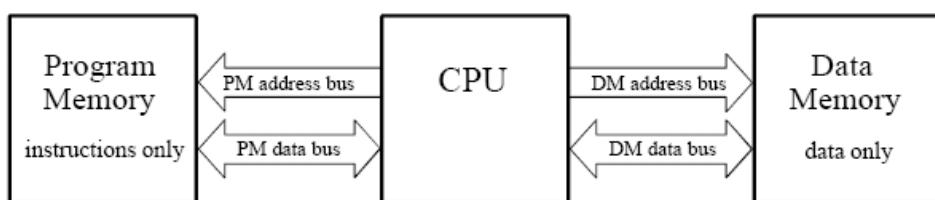


Figura 5.2 - Arquitetura de Von Neumann e Havard.

CAPÍTULO VI

Avaliação das Metodologias de Controle Ativo de Ruído

Para avaliar as metodologias estudadas no controle ativo de ruído, ensaios experimentais foram conduzidos em laboratório. Os algoritmos foram implementados na plataforma TMS320LF2407A da Texas Instruments e executados em tempo real. Estratégias de controle FXLMS, FBFXLMS, Feedback e Híbrida, com identificação *off-line* dos caminhos secundário e de realimentação foram implementadas. Estes caminhos foram identificados através do pacote de identificação do MatLab®.

Também foram implementadas três estratégias de controle com a identificação *on-line* do caminho secundário, propostas por Fan (1990), Eriksson (1989) e Zhang (2001).

6.1 A Bancada Experimental

Para testar experimentalmente os diferentes sistemas de controle apresentados, foi montada uma bancada experimental constituída por um duto de PVC que foi instrumentado com alto falantes (atuadores) e microfones (sensores) de baixo custo. O diagrama do sistema pode ser visto na Figura 6.1. A foto da bancada experimental pode ser vista no Anexo III.

O duto possui um comprimento de 3.5m e um diâmetro de 0.15m. De acordo com a equação (2.21), este duto terá ondas acústicas planas até a freqüência de 670Hz. A fonte primária de ruído foi posicionada numa das extremidades e o alto-falante de cancelamento posicionado a 2m do alto-falante de ruído. O microfone de referência foi colocado a uma distância de 0.25m e finalmente o microfone de erro a 3.4m do alto falante de ruído (fim do duto).

Os alto-falantes utilizados nesta bancada experimental foram da marca Bravox BA6SS que possuem uma potência máxima de 60 Watts RMS e resposta em freqüência de 50 a 20Khz.

Para amplificá-los, foram implementados circuitos eletrônicos baseados no amplificador de áudio TDA2030. Estes amplificadores são circuitos integrados para o uso em amplificadores do tipo AB em baixa freqüência. Tipicamente eles garantem 12W de potência de saída em cargas de 4Ω e 8W em 8Ω . O TDA2030 proporciona alta saída de corrente com pequena distorção harmônica. O dispositivo também possui proteção contra curto-circuito e um sistema de desligamento para evitar operações em altas temperaturas (TDA2030, 1993). O diagrama esquemático do circuito eletrônico completo pode ser visto no Anexo V.

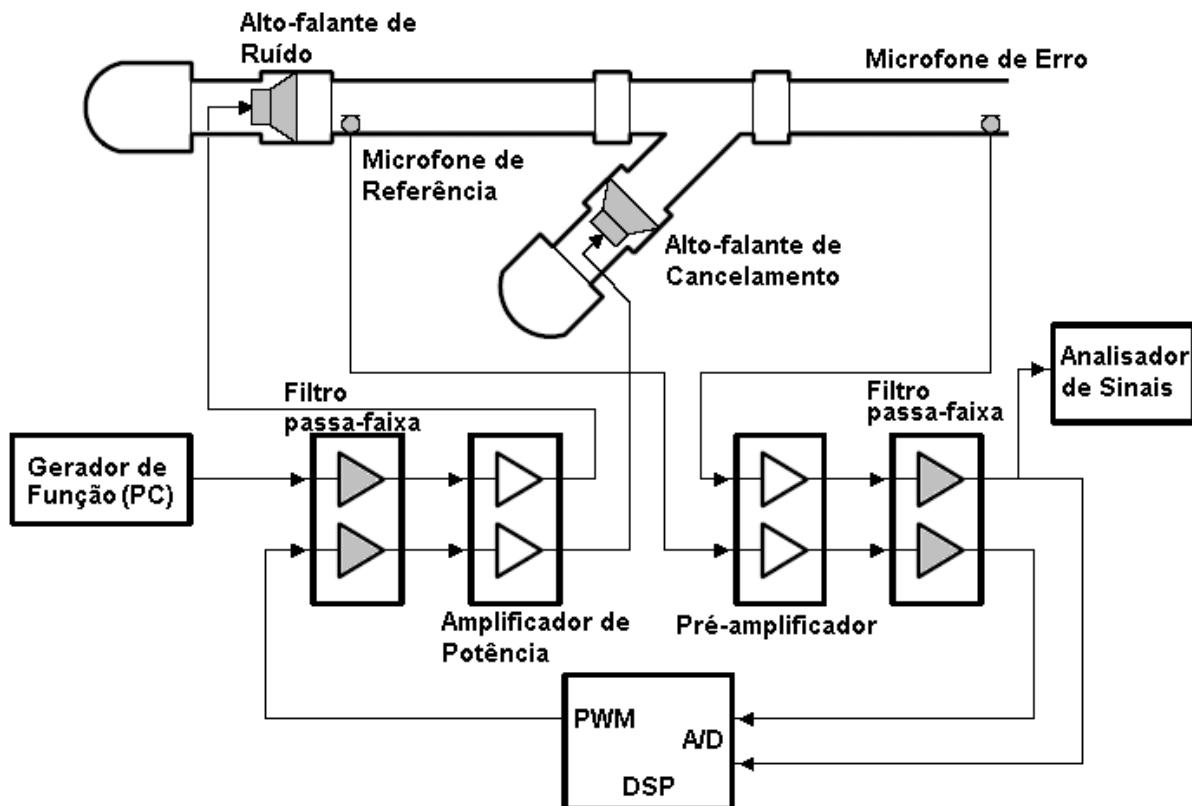


Figura 6.1 - Sistema Físico do Controle Ativo de Ruído.

Foram utilizados microfones do tipo eletreto omnidirecionais JL-061C amplamente utilizados em telefonia. Estes possuem sensibilidade de $-58\text{dB}/\mu\text{bar}$, com impedância de $2\text{K}\Omega$, faixa de freqüências de 20 a 20KHz e relação sinal/ruído superior a 55dB (Jameco Electronics, 2004).

Também foram implementados circuitos eletrônicos para a polarização e o tratamento dos sinais provenientes dos microfones. Nestes circuitos, foram utilizados amplificadores

operacionais comuns do tipo TL074. Estes amplificadores possuem entradas JFET com alta impedância de entrada, baixa corrente de *offset* e alto *slew rate*. A baixa distorção harmônica e o baixo ruído fazem esta série ideal para aplicações de alta fidelidade e pré-amplificadores de áudio (SLOS080D, 1996). O diagrama esquemático do circuito eletrônico completo também pode ser visto no Anexo V.

Em todos os circuitos, os filtros *antialiasing* e de reconstituição foram projetados para banda de freqüência de 500 Hz excluindo o nível DC.

Os algoritmos de controle foram implementados utilizando o pacote CODE COMPOSER® incluso na aquisição da plataforma DSP. A programação da mesma foi realizada em Linguagem C (SPRU024E, 1999), utilizando os pacotes de processamento digital de sinais da *Texas Instruments*®, estes pacotes podem ser encontrados na página da internet do fabricante (Filter Library, 2002).

Nesta plataforma foram utilizados 2 canais de conversão analógico digital (A/D), com 10 bits de resolução e *Sample and Hold* (S/H), para aquisição dos sinais provenientes dos microfones. Foi utilizado um sinal de saída do tipo PWM (Modulação por Largura de Pulso) para realizar a conversão digital analógica (D/A).

Para o gerador de ruído, pode-se utilizar qualquer gerador de sinais que tenha a possibilidade de gerar sinais periódicos e ruídos aleatórios. Há outras possibilidades mais acessíveis e de menor custo. Uma delas é gerar o ruído no próprio DSP e utiliza-lo através de outra saída PWM. Outra opção é utilizar um programa de processamento de áudio, que gere os sinais periódicos ou ruídos aleatórios através da placa de som do computador. Esta opção também oferece a possibilidade de se gravar um ruído real e reproduzi-lo posteriormente. Nos experimentos realizados foi utilizado o programa *Cool Edit*®.

As funções de resposta em freqüência envolvidas no duto podem ser vistas na Figura 6.2. Estas foram obtidas através da excitação do sistema com ruído branco de média nula. Cada alto-falante foi excitado individualmente e os dados foram coletados pelos microfones com a freqüência de amostragem de 2KHz

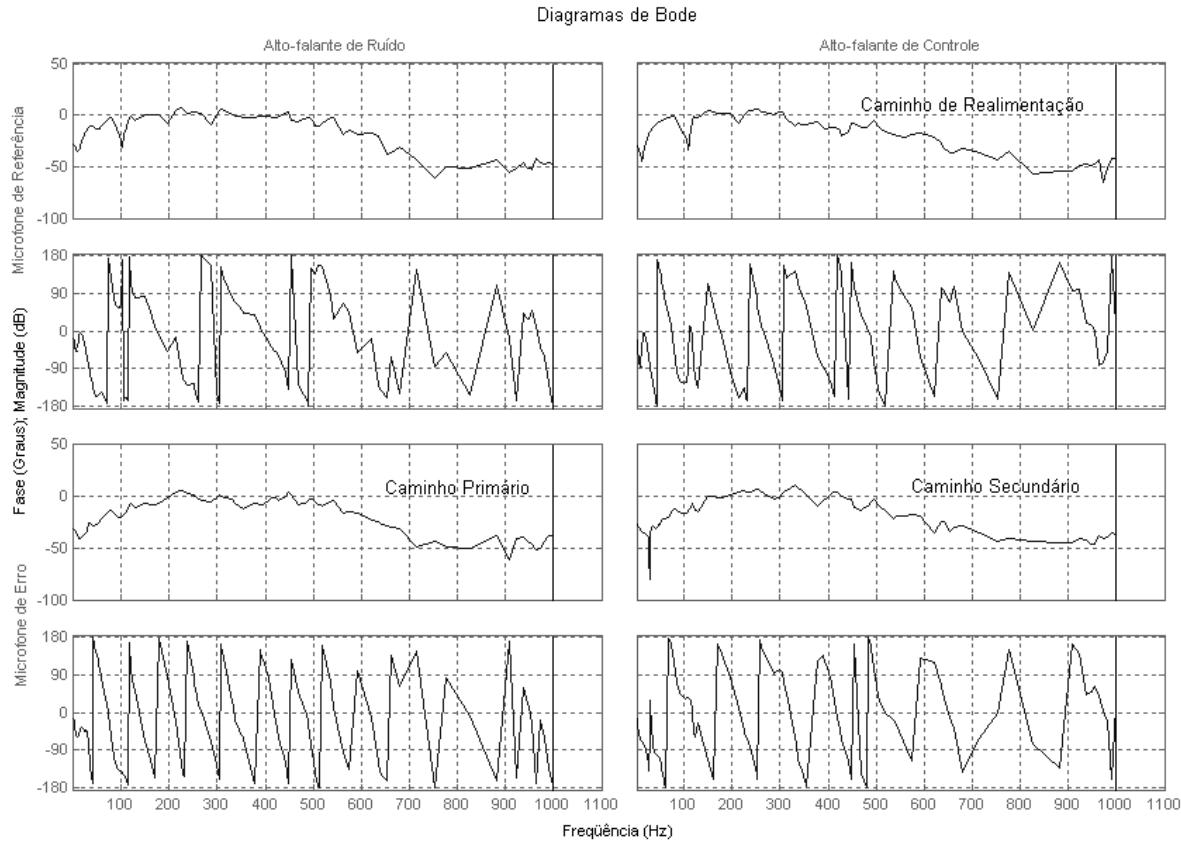


Figura 6.2 - Funções de Transferência dos Caminhos do Sistema Físico.

6.2 Recursos utilizados pelas diferentes metodologias avaliadas

Neste tópico pode-se observar os diferentes recursos utilizados pelo processador no Controle Ativo de Ruído. Os dados são baseados nos diagramas e algoritmos apresentados no Capítulo IV, no entanto, são considerados apenas os cálculos envolvendo o processo de filtragem e adaptação. Os cálculos da estimativa da potência e do passo adaptativo segundo Minguez, bem como do uso do fator de vazamento não são levados em conta.

Na Tabela 6.1, o campo “Coeficientes” está relacionado com a memória consumida para armazenar os mesmos envolvidos. O campo “Memória de Dados de Entrada” representa as linhas de atraso de entrada dos filtros. O campo “Memória de Dados de Saída” representa as linhas de atraso para a adaptação FXLMS. As multiplicações envolvidas podem também ser visualizadas. O campo “Dificuldade” é visto pelo autor como sendo a dificuldade encontrada pelo mesmo para a implementação de tais metodologias. Este último campo é subjetivo e serve como um parâmetro relativo entre os algoritmos.

Tabela 6.1 - Recursos utilizados pelo processador na implementação dos algoritmos

Algoritmo	Coeficientes	Memória de Dados de Entrada	Memória de Dados de Saída	Multiplicações	Dificuldade
FXLMS	W+S	W+S	W	2W+S	Fácil
FBXLMS	W+S+F	W+S+F	W	2W+S+F	Fácil
Feedback	W+S	W+2S	W	2W+2S	Médio
Híbrido	A+C+S+F	A+C+3S+F	A+C	2A+2C+3S+F	Difícil
Fan	W+S	W+S	W+S	2W+2S	Fácil
Eriksson	W+S	W+2S	W	2W+3S	Médio
Zhang	W+S+B	W+2S+B	W	2W+3S+2B	Difícil

W = Ordem do filtro da estimativa do caminho primário (controlador).

S = Ordem do filtro da estimativa do caminho secundário.

F = Ordem do filtro da estimativa do caminho de realimentação.

A = Ordem do filtro da estimativa do caminho primário (feedforward no híbrido).

C = Ordem do filtro da estimativa do caminho primário (feedback no híbrido).

B = Ordem do filtro auxiliar.

6.3 Testes com simples tons: Identificação off-line

De acordo com a função de transferência do caminho primário mostrada na Figura 6.2, o duto possui uma freqüência ressonante em 220Hz onde esta apresenta o maior ganho. Os algoritmos FXLMS, FBFXLMS, Feedback e Híbrido foram testados para esta freqüência.

A modelagem *off-line* dos caminhos secundário e de realimentação foi realizada através do pacote de identificação do MatLab® (Delfino *et al.*, 2005, Nuñez *et al.*, 2004). Os dados foram colhidos na plataforma DSP através da excitação por um ruído branco de média nula no alto-falante de cancelamento. A estimativa dos caminhos que utilizam filtros de ordem 120 podem ser vistas na Figura 6.3. Depois das estimativas serem realizadas, os coeficientes dos filtros foram copiados para os algoritmos de controle que as utilizam.

Destaca-se que o passo de adaptação μ utilizado nestes experimentos foi do tipo adaptativo limitado a 10% do seu valor máximo. O fator de vazamento utilizado foi de 0.9978 ($Q_{15} = 32760$) e a freqüência de amostragem utilizada pela plataforma de controle (DSP), de acordo com o critério de Nyquist, foi de 2500Hz. Os resultados no tempo e na freqüência, quando o sistema de controle está ligado e desligado, podem ser vistos na Figura 6.4, Figura 6.5, Figura 6.6 e Figura 6.7. Estes foram adquiridos através de uma placa Dspace®, utilizando também a freqüência de amostragem de 2500Hz. Foram adquiridos 30 segundos, incluindo sinal do erro sem controle e com controle, no entanto serão mostrados apenas intervalos de tempo próximo à ativação do controlador.

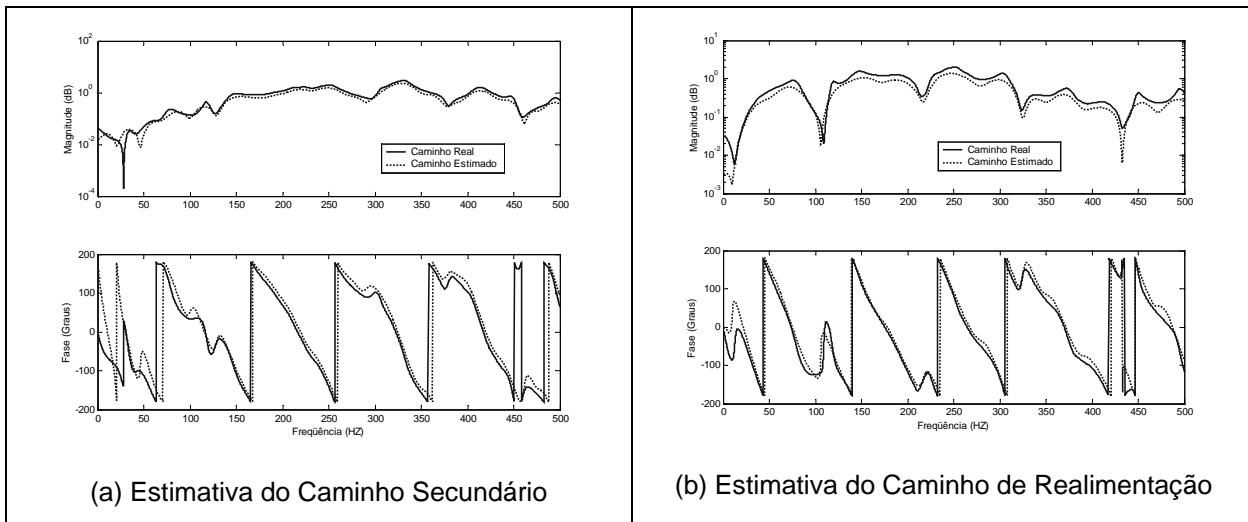


Figura 6.3 – Estimativas do Caminhos através do MatLab®.

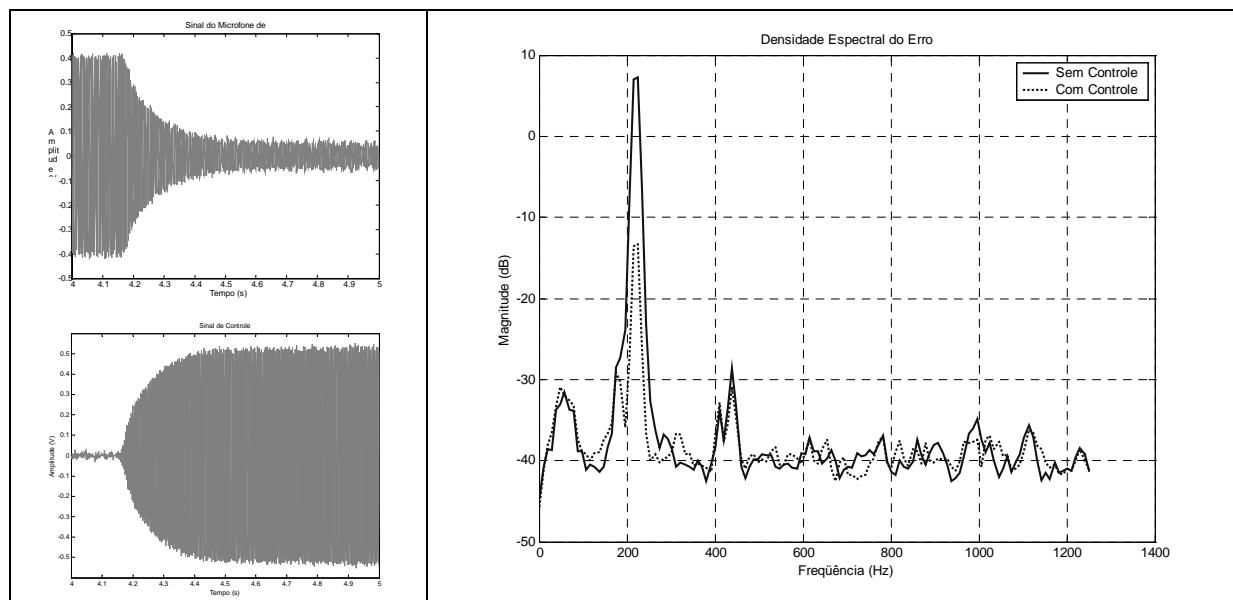


Figura 6.4 – Controle FXLMS para a freqüência de 220Hz. (a) Sinal do Microfone de Erro.
(b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

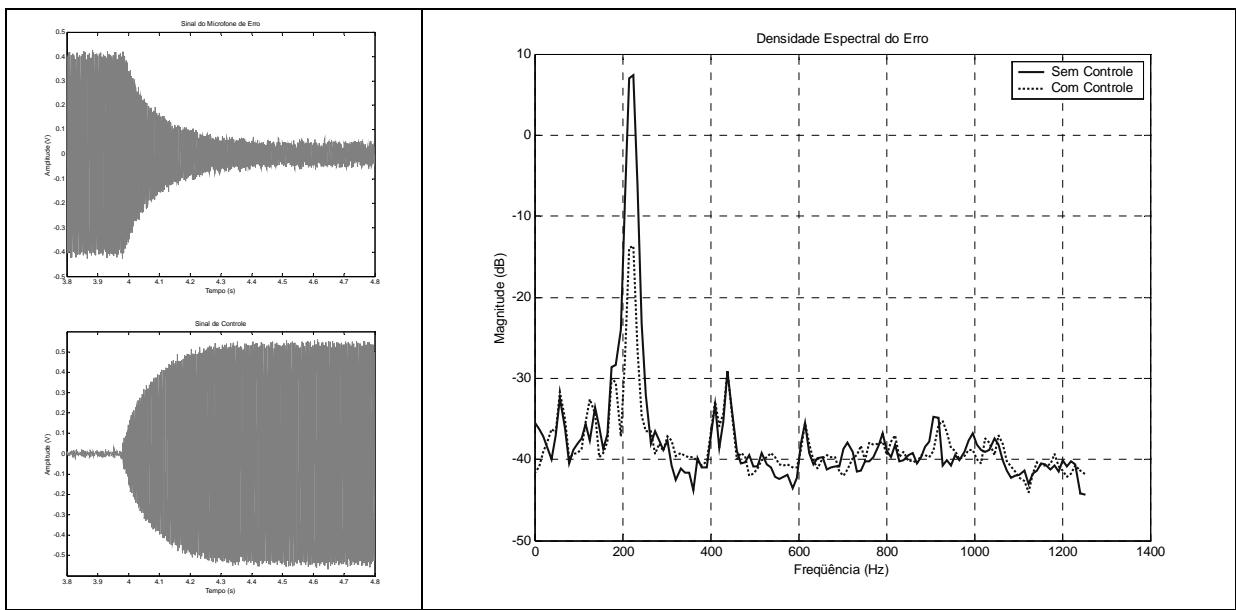


Figura 6.5 – Controle FBFXLMS para a freqüência de 220Hz. (a) Sinal do Microfone de Erro.
(b) Sinal de Cancelamento. (c) Densidade Espectral de Potênciade erro.

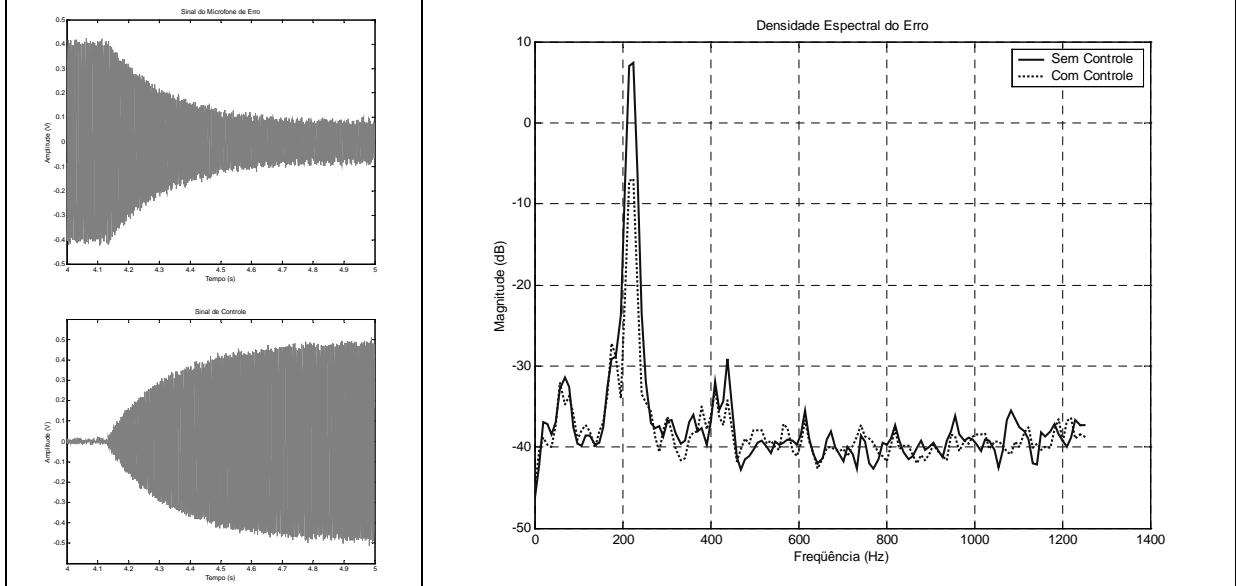


Figura 6.6 – Controle Feedback para a freqüência de 220Hz. (a) Sinal do Microfone de Erro.
(b) Sinal de Cancelamento. (c) Densidade Espectral de Potênciade erro.

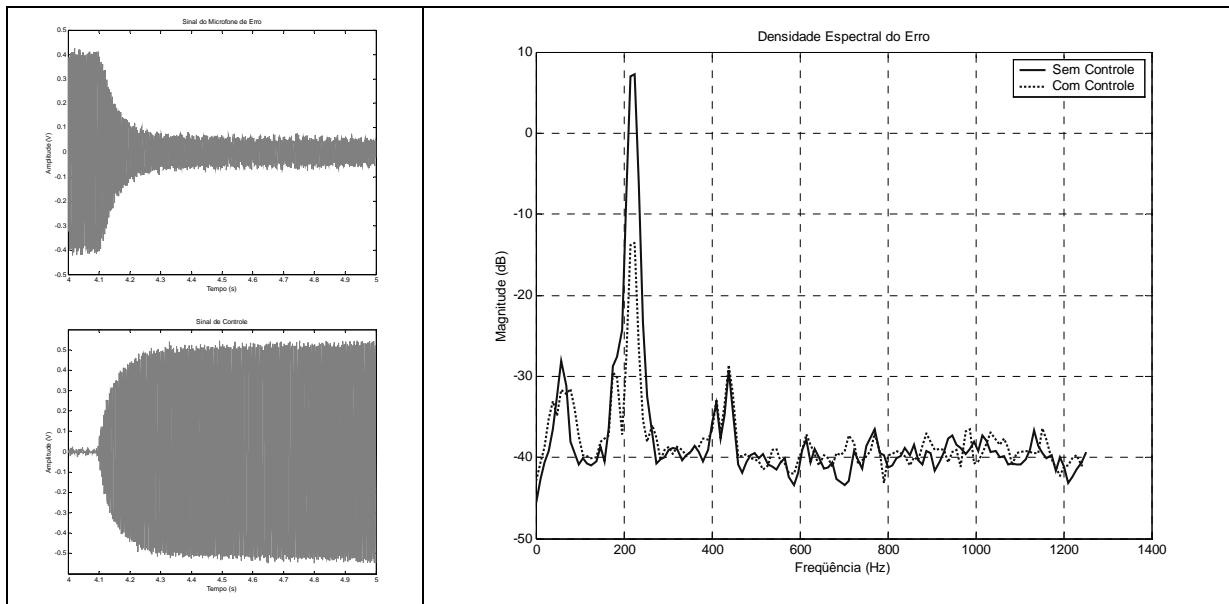


Figura 6.7 – Controle Híbrido para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

Os resultados anteriores mostram que atenuações de 20dB podem ser conseguidas quando o sistema de controle é ligado. Para avaliar a capacidade dos algoritmos, o sistema físico foi excitado por uma função senoidal em freqüências que variam entre 150 e 450 Hz em passos de 50 Hz. Foram mantidos os mesmos parâmetros utilizados no controlador do experimento anterior. No entanto, para a análise dos procedimentos, foi utilizado o analisador SD380 da Spectrum Analyzer, com a freqüência de amostragem de 500Hz, resolução em freqüência de 0.6249Hz, janela Hanning e 10 médias. Utilizou-se o Trigger Internal Repetitive com Thresholding de 1%.

Inicialmente, o sistema foi excitado e o valor em dBV da freqüência de excitação anotado depois de completada as 10 médias. Posteriormente, o sistema de controle foi ligado, a memória do analisador limpada e após outras 10 médias um novo valor foi colhido. No algoritmo híbrido, utilizou-se a estrutura feedforward que apresentou melhores resultados, tanto na atenuação, quanto no tempo de acomodação e estabilidade.

Na Tabela 6.2, pode-se observar um desempenho significativo, atenuações acima de 25 dB, para os diferentes algoritmos de controle ativo de ruído. Sendo que na área de acústica, atenuações acima de 6 dB são sensíveis ao ouvido do ser humano.

Tabela 6.2 - Ensaios experimentais dos algoritmos com identificação *off-line*.

FREQÜÊNCIA (Hz)	SEM ANC (dBV)	FXLMS (dBV)	FBFXLMS (dBV)	FEEDBACK (dBV)	HÍBRIDO (dBV)
150	-12.8	-35.1	-33.5	-34.0	-35.1
200	-11.3	-38.1	-22.9(*)	-31.3	-38.5
250	-11.3	-29.7	-29.3	-24.6	-30.3
300	-11.8	-30.9	-31.1	-28.6	-31.3
350	-17.7	-28.7	-28.8	-27.8(*)	-28.8
400	-13.4	-44.0	-43.0	-41.3	-45.2
450	-10.9	-47.2	-49.5	-18.6(*)	-51.2
Ordem dos Filtros		W(z) : 32 S(z) : 120	W(z) : 32 S(z) : 120 D(z) : 120	W(z) : 32 S(z) : 120	A(z) : 32 C(z) : 32 S(z) : 120 D(z) : 120

(*) Sistema apresentou instabilidade.

A proposta *feedforward* se mostrou eficiente para as diferentes freqüências e a inclusão da estimativa do caminho de realimentação (FBFXLMS) para se obter um sinal de referência limpa melhorou a performance de forma pouco significativa em algumas freqüências. Particularmente para a freqüência de 200Hz, esta inclusão trouxe instabilidade ao sistema.

No controlador *feedback* observou-se também uma boa performance. Contudo, o controle se mostrou instável nas freqüências onde não foram obtidas boas estimativas da referência. Neste caso o fator de vazamento evitou que o sistema divergisse.

Com relação ao sistema Híbrido, observaram-se resultados similares ao sistema *feedforward*. No entanto, a convergência do algoritmo se mostrou mais rápida e eficiente. Quanto à estabilidade e a banda de controle, o algoritmo apresentou melhores resultados, não se mostrando tão dependente da freqüência de excitação.

6.4 Testes com simples tons: Identificação *on-line*

Como visto no Capítulo IV, nem sempre é possível realizar a estimativa *off-line* do caminho secundário, ou este, apresenta características não-estacionárias, sendo necessária a estimativa *on-line*. Desta maneira, alguns algoritmos de tal natureza foram testados para avaliar a capacidade de atenuação.

Manteve-se a configuração dos experimentos, bem como os parâmetros utilizados na análise realizada pelo SD380. Para esta abordagem, utilizou-se a freqüência de amostragem de 2500Hz, tanto para o DSP quanto para a placa Dspace.

Os algoritmos propostos por Fan (Fan *et al.*, 1990), Eriksson (Eriksson *et al.*, 1989), Zhang (Zhang *et al.*, 2001) foram testados para freqüência de 220Hz. Os resultados no

tempo e na freqüência, quando o sistema de controle está ligado e desligado, podem ser vistos na Figura 6.8, Figura 6.9 e Figura 6.10.

Sabe-se que a inclusão do ruído aditivo para a modelagem do caminho secundário resulta em uma degradação na performance geral do sistema. Para resolver tal problema, propôs-se realizar a estimativa do caminho secundário somente no início do processo de controle, desta forma o ruído aditivo é aplicado ao sistema no intervalo de tempo da estimação. Optou-se também por não realizar a adaptação do caminho primário enquanto o caminho secundário é estimativo evitando, assim, a adaptação cruzada.

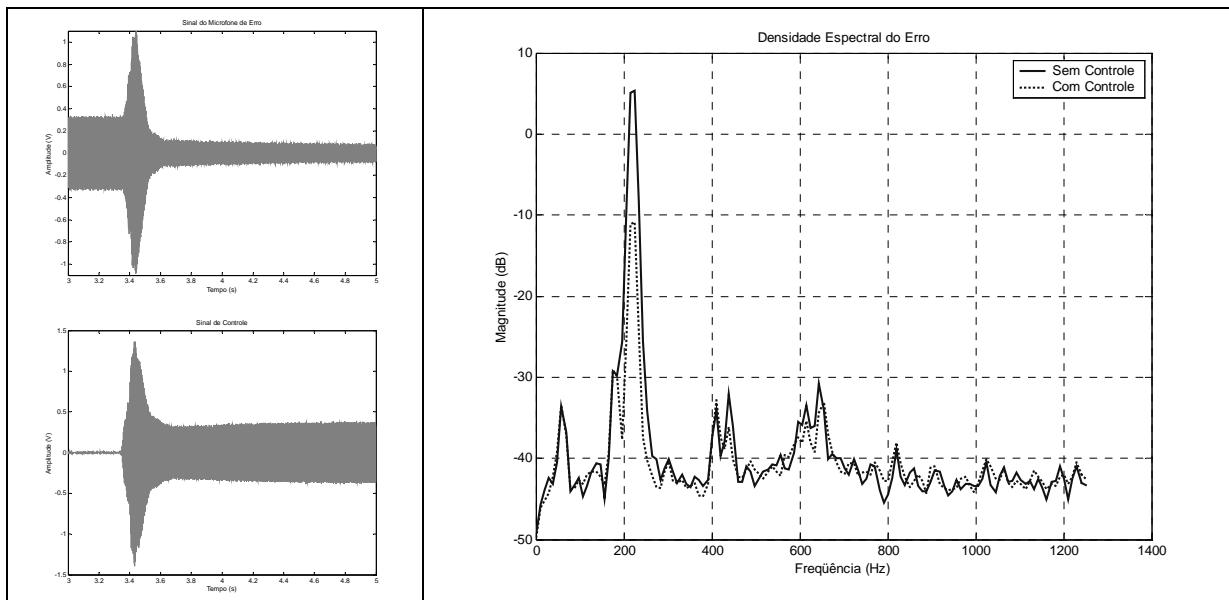


Figura 6.8 – Controle Fan para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

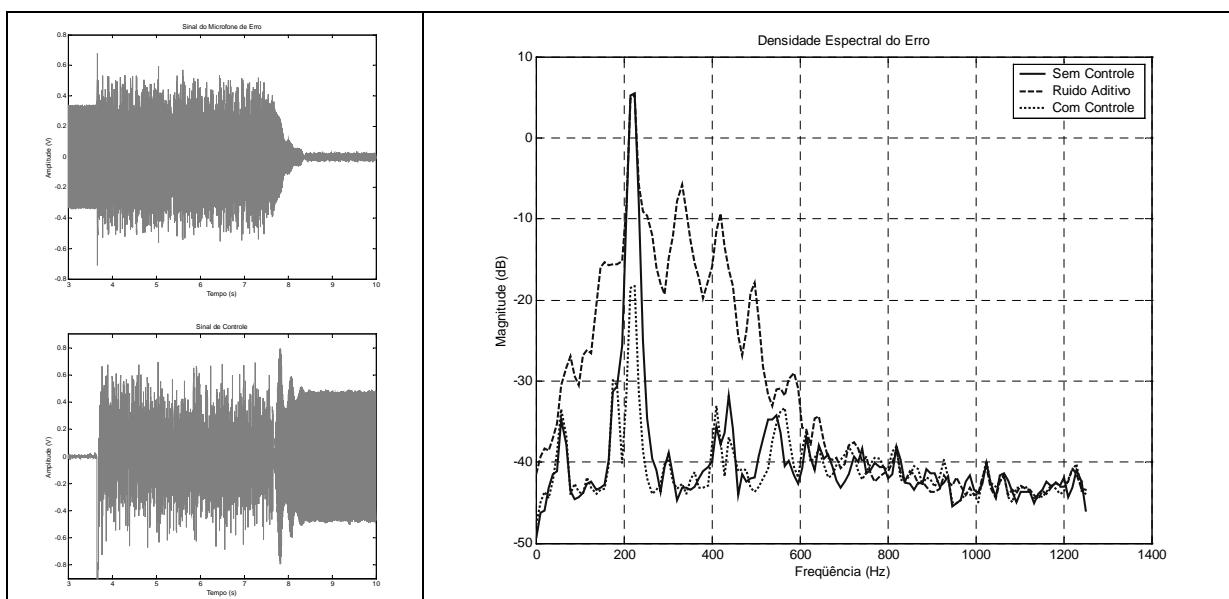


Figura 6.9 – Controle Eriksson para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

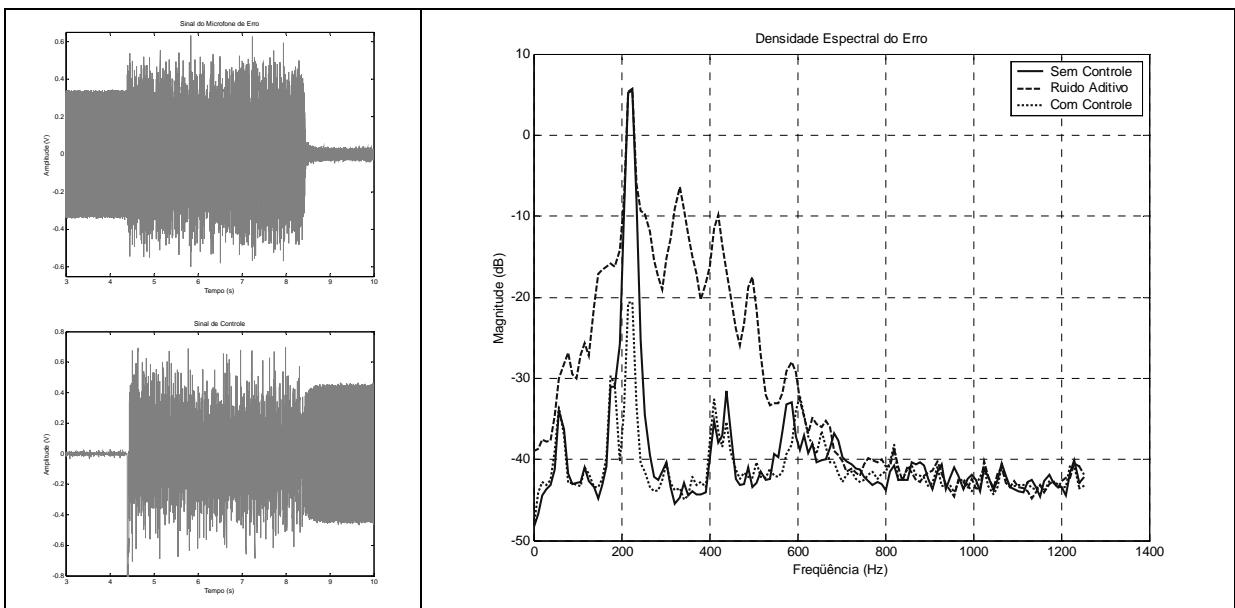


Figura 6.10 – Controle Zhang para a freqüência de 220Hz. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

Os resultados anteriores mostram que atenuações de 20dB podem ser conseguidas quando o sistema de controle é ligado. No entanto, os sistemas que utilizam ruído auxiliar apresentaram um aumento significativo do ruído de fundo. Para avaliar a capacidade dos algoritmos, o sistema físico foi novamente excitado por uma função senoidal em freqüências que variam entre 150 e 500 Hz em passos de 50 Hz.

Tabela 6.3 - Ensaios experimentais dos algoritmos com identificação *on-line*.

FREQÜÊNCIA (Hz)	SEM ANC (dBV)	Fan (dBV)	Eriksson (dBV)	Zhang (dBV)
150	-17.2	-38.7	-34.3	-46.3
200	-14.0	-35.2	-35.0	-35.9
250	-13.0	-35.4	-32.0	-34.5
300	-15.6	-30.7	-30.4	-32.4
350	-18.2	-28.7	-28.4	-30.6
400	-14.9	-40.2	-43.9	-49.6
450	-14.8	-34.0	-29.3	-32.5*
Ordem dos Filtros		W(z) : 32 S(z) : 76	W(z) : 32 S(z) : 76	W(z) : 32 B(z) : 32 S(z) : 76

(*) Sistema apresentou ligeira instabilidade.

Na Tabela 6.3 pode-se observar um desempenho significativo, atenuações acima de 30 dB, para os diferentes algoritmos de controle ativo de ruído. Nota-se que em relação à Tabela 6.2, a amplitude da excitação teve que ser menor, devido ao fato de algumas

propostas em certas freqüências modelarem o caminho secundário com alto ganho, levando o sistema à saturação e instabilidade.

Na proposta de Fan, que realiza a modelagem *on-line* do caminho secundário sem a necessidade de ruído branco adicional, verificou-se grande desempenho na banda. No entanto, este algoritmo não aceitou o fator de vazamento (*leakage*) na adaptação dos coeficientes do caminho secundário, com a inclusão do mesmo, o algoritmo não aumentava o ganho do caminho secundário e por consequência o caminho primário não se adaptava, anulando a saída do sistema. O fator de vazamento continuou sendo utilizado na adaptação dos coeficientes do caminho primário.

Neste caso, foi realizado um monitoramento da potência do sinal de erro do sistema, quando este se mostrava abaixo de um limiar ($Q_{15} = 300$), o sistema não realizava mais a adaptação do filtro que estimava o caminho secundário, continuando a adaptar somente o filtro primário. A adaptação do caminho secundário poderia voltar a ocorrer quando o erro fosse maior que o limiar. Esta proposta também apresentou dificuldade em relação à convergência, onde muitas vezes o controlador divergiu logo na inicialização, necessitando ser reiniciado até atingir a convergência. Isto ocorreu devido à adaptação cruzada dos filtros $W(z)$ e $\hat{S}(z)$ e também ao fato dos dois filtros utilizarem o mesmo sinal de erro.

A proposta de Eriksson, que utiliza ruído aditivo para auxiliar na modelagem do caminho secundário, apresentou resultados de elevada performance, com atenuações em toda a banda excitada. Neste caso, utilizou-se o fator de vazamento também somente na adaptação do caminho primário e o algoritmo se mostrou estável, sem a necessidade de monitoramento do sinal de resíduo.

A proposta de Zhang, que tem por definição corrigir algumas falhas na teoria da proposta de Eriksson, apresentou melhores performances de atenuação, bem como uma melhor estimativa do caminho secundário. É de grande dificuldade a escolha do passo de adaptação do preditor $B(z)$, neste caso utilizou o passo do tipo adaptativo baseado na potência do sinal de entrada. Este filtro muitas vezes também aumentava seus coeficientes, o que acarretava em estimativas não satisfatórias do caminho secundário, para resolver tal problema, utilizou-se também o fator de vazamento ($Q=32760$).

De todas as propostas validadas experimentalmente, incluindo as de modelagem *off-line* e *on-line* do caminho secundário, pode-se observar que não existe um algoritmo de melhor performance geral. Ela está intimamente ligada ao sistema físico envolvido e com a natureza do ruído primário. Cabe ao projetista realizar uma análise qualitativa e quantitativa para escolher qual a estratégia ideal para um determinado cenário de cancelamento.

6.5 Testes com múltiplos tons

Neste trabalho, os algoritmos testados com simples tons também foram testados com múltiplos tons. Neste caso, foram escolhidas as freqüências de 150, 250 e 350Hz. Os resultados no tempo e na freqüência, quando o sistema de controle está ligado e desligado são mostrados a seguir. As configurações de controle no DSP e de análise por parte do Dspace e do SD380 foram mantidas iguais às utilizadas nos testes com simples tons.

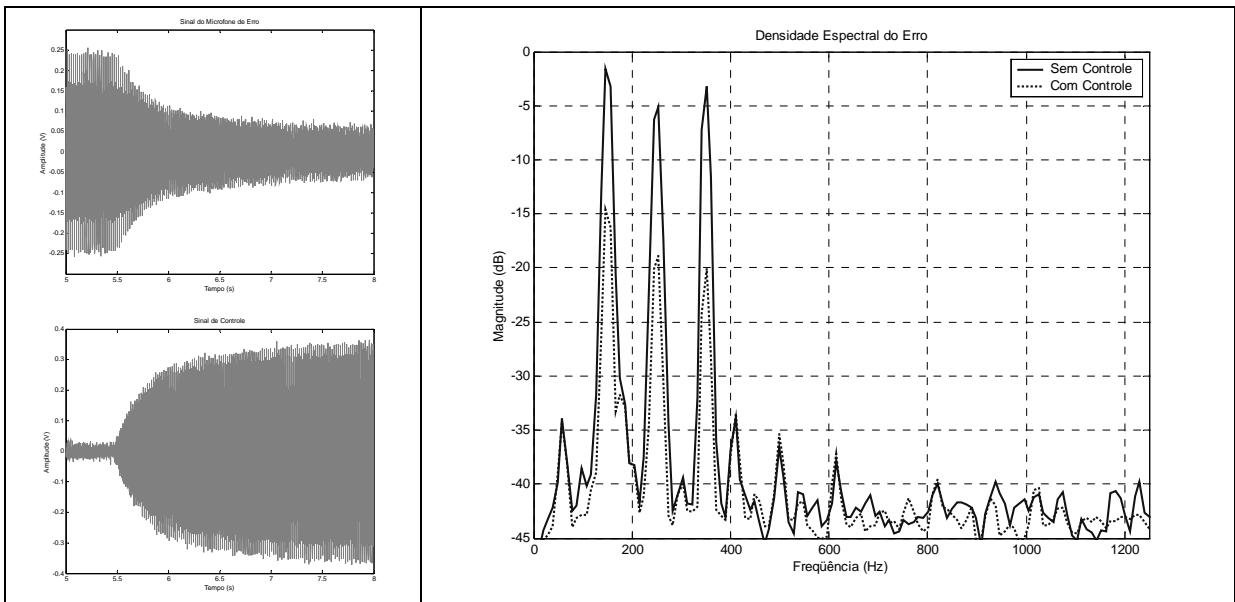


Figura 6.11 – Controle FXLMS multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

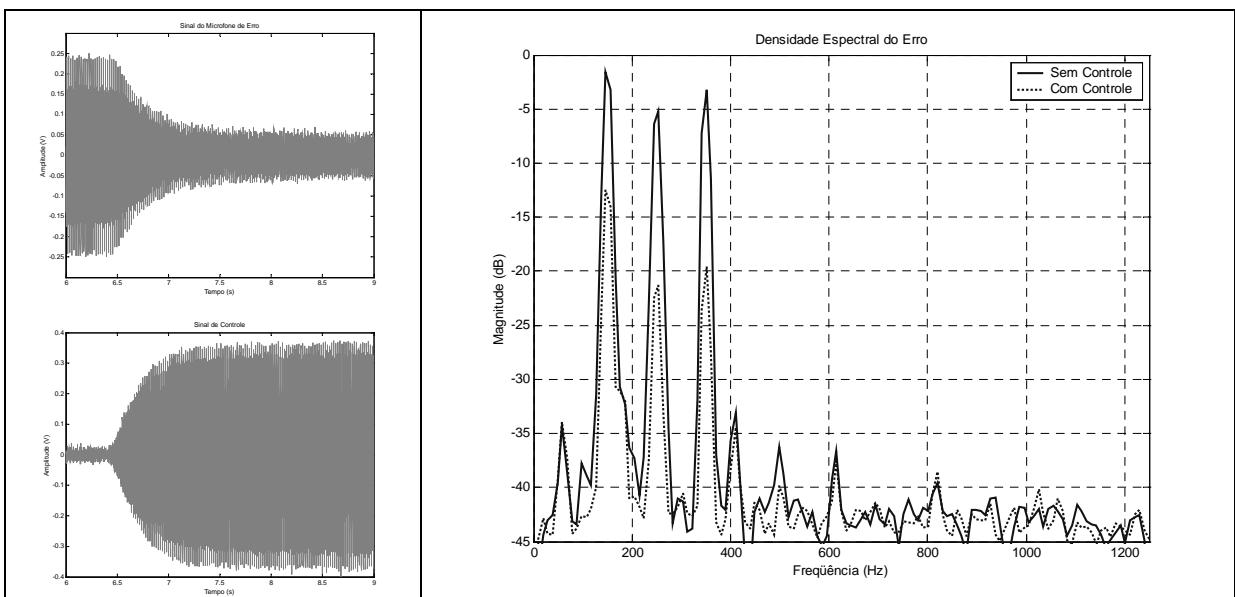


Figura 6.12 – Controle FBFXLMS multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

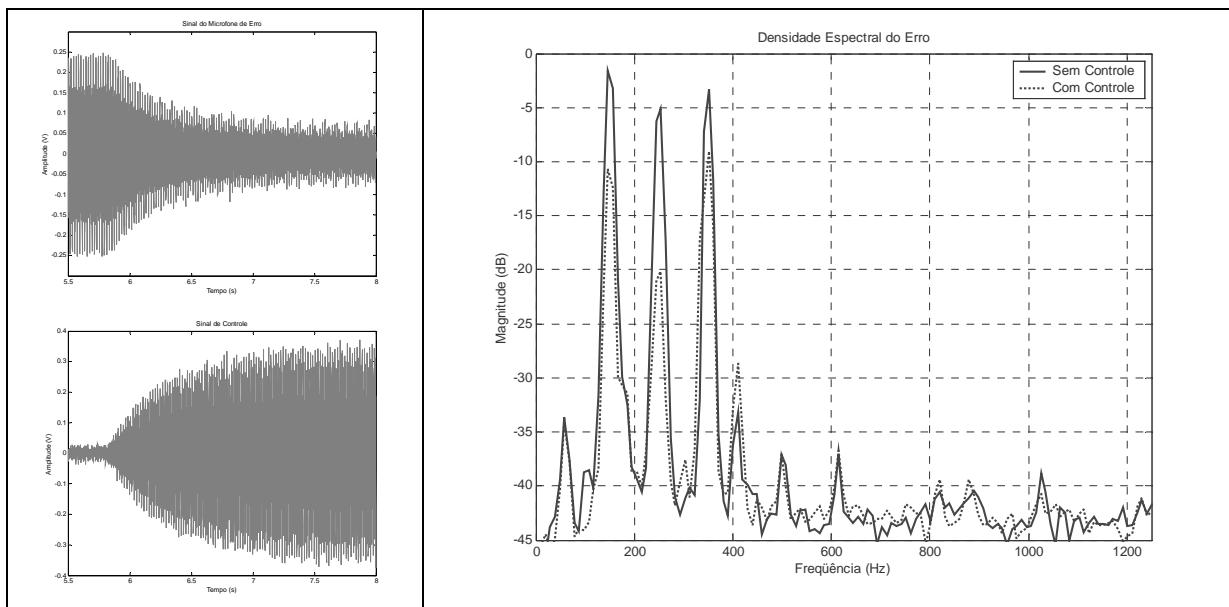


Figura 6.13 – Controle Feedback multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

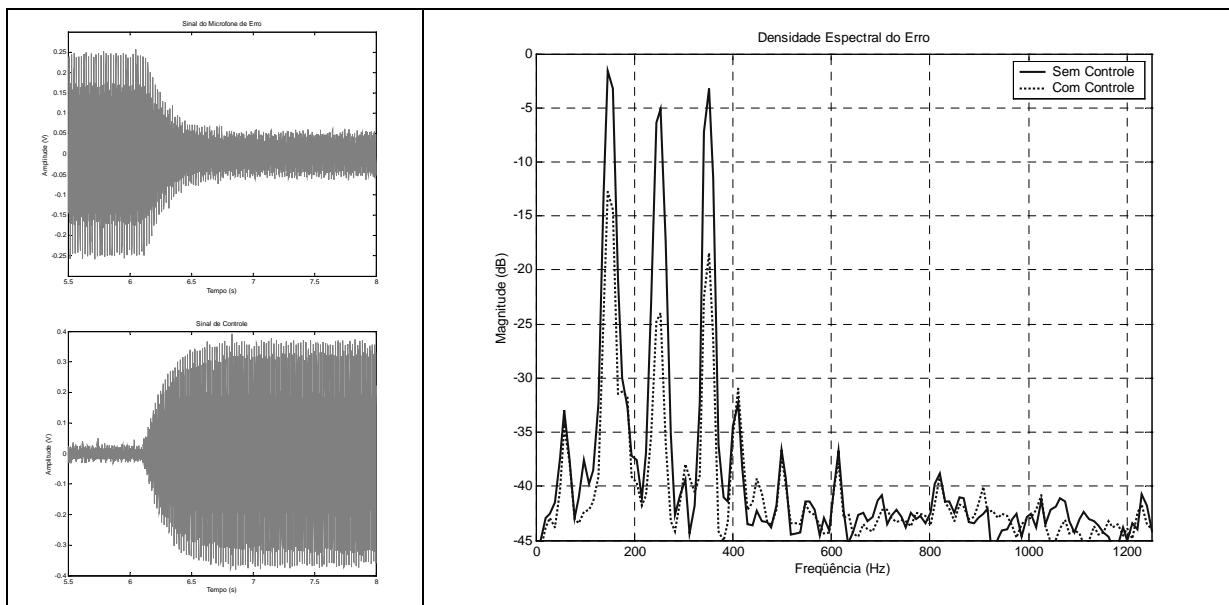


Figura 6.14 – Controle Híbrido multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

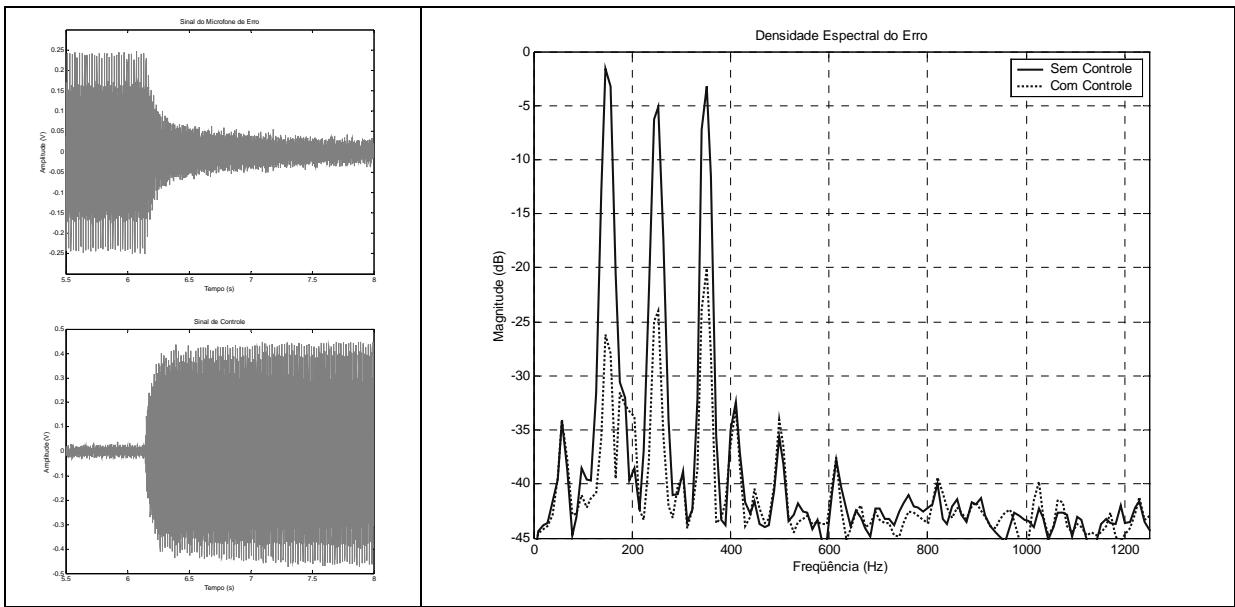


Figura 6.15 – Controle Fan multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

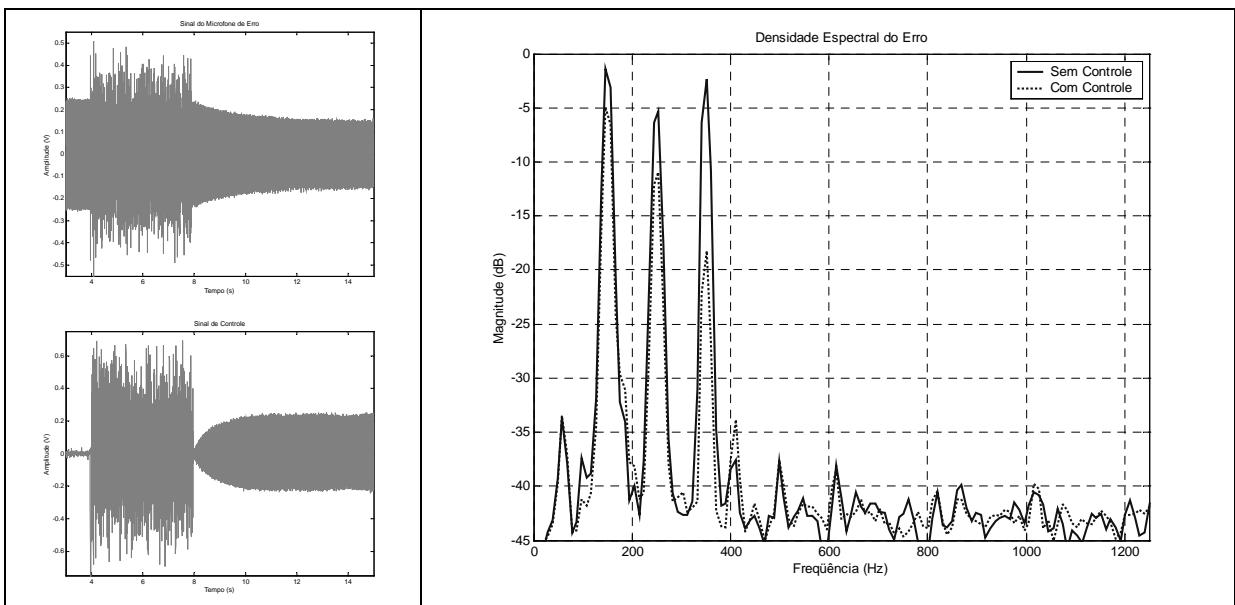


Figura 6.16 – Controle Eriksson multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

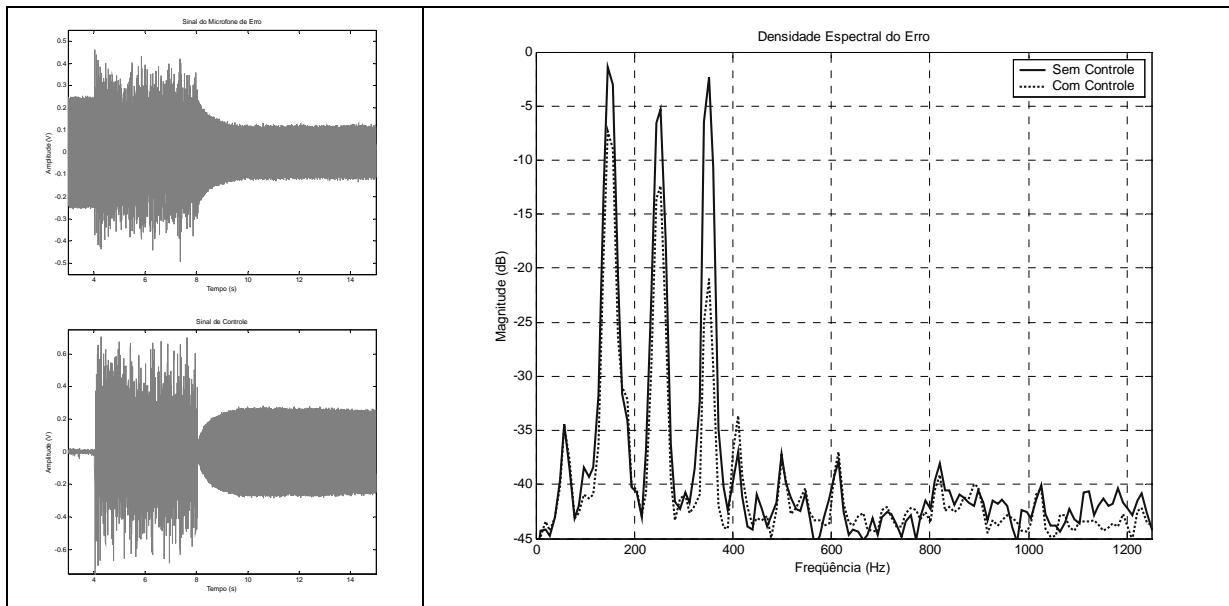


Figura 6.17 – Controle Zhang multi-tom. (a) Sinal do Microfone de Erro. (b) Sinal de Cancelamento. (c) Densidade Espectral de Potência do erro.

Com base nos dados colhidos, pode-se observar que os algoritmos com modelagem *on-line* que utilizam ruído aditivo não obtiveram grandes performances nos testes multi-tom. Já o método de Fan apresentou os melhores resultados entre as metodologias avaliadas, sendo o mais indicado para tratar este tipo de problema. A boa performance desta proposta vem do fato dela modelar o caminho secundário somente para as freqüências contidas no ruído primário. Isto melhora bastante a estimativa do caminho secundário, já que nos outros métodos o mesmo número de coeficientes deve modelar toda a banda de controle, tornando a estimativa ruim para algumas freqüências.

Apesar das atenuações conseguidas, a estrutura *Feedback* foi a de pior performance, não sendo indicada para este tipo de problema. Esta apresentou certa instabilidade durante o processo de cancelamento, criando em alguns momentos outras freqüências no sinal de controle. Como o método utiliza somente o microfone de erro, estimando assim a entrada, pode-se concluir que faltou informação para que o algoritmo calculasse a saída ótima do filtro adaptativo. Esta falta de informação degradou a performance do sistema para esta configuração adotada. Este tipo de estrutura é aconselhada para pequenos sistemas físicos ou com fontes de ruído com pouco espalhamento espectral.

De forma geral as metodologias avaliadas se comportaram bem diante da entrada com três tons harmônicos.

CAPÍTULO VII

Conclusões e Sugestões

Neste trabalho foram apresentadas e analisadas algumas estratégias de controle ativo de ruído em dutos. Focou-se essencialmente em metodologias mono-canais, de natureza adaptativa e capazes de serem implementadas em processadores digitais de sinais. Dos estudos realizados há algumas conclusões de ordem geral e outras particulares associadas a cada metodologia,

De forma geral, comprovou-se que o controle ativo de ruído utilizando filtros adaptativos é uma excelente estratégia, que devidamente utilizada gera controladores eficientes e estáveis. O projeto destes controladores é simplificado pela proposta adaptativa LMS, já que esta dispensa o conhecimento prévio de algumas funções de resposta em freqüência envolvidas. Fato muito importante pois, em acústica, a modelagem matemática dos fenômenos físicos é freqüentemente um problema de natureza complexa e, em muitas circunstâncias, com características não-estacionárias.

Através do trabalho, comprovou-se as potencialidades de um processador DSP de baixo custo na implementação dos algoritmos de controle. Estes, devido à evolução tecnológica experimentada, destacam-se como instrumentos valiosos na implementação de metodologias de controle avançadas na indústria.

Na realização experimental dos algoritmos utilizou-se uma plataforma DSP de aritmética de precisão finita. Verificou-se que, no algoritmo LMS, o uso do *fator de vazamento* – que busca limitar o efeito da “contaminação” dos coeficientes do filtro pelos ruídos de arredondamento e de discretização do algoritmo - é muito importante para a estabilidade do sistema.

Devido à aritmética de ponto fixo do DSP em alguns momentos a convergência do algoritmo foi degradada. Particularmente, em situações onde os caminhos acústicos apresentaram um ganho maior que o unitário em certas freqüências, ocorreram problemas de saturação (*overflow*) no cálculo da saída do filtro, o que restringiu a amplitude dos sinais de entrada, prejudicando a convergência do controlador.

Na escolha da ordem dos filtros, verificou-se que para os casos de estimativas *on-line*, acima de 41 coeficientes o controle não apresentou melhora significativa, consumindo apenas mais memória e maior tempo de processamento. Somente foram utilizados filtros de ordem superior (120) nas modelagens *off-line*, pois para estas estimativas verificaram-se melhoras acentuadas com o aumento do número de coeficientes.

Nos ensaios experimentais conforme o esperado, verificou-se que a performance do sistema é altamente dependente do posicionamento dos sensores e atuadores, já que a capacidade de atenuação do sistema de controle depende dos caminhos acústicos envolvidos. No caso do duto avaliado observou-se que nas freqüências de ressonância de maiores ganhos do duto (220Hz e 450Hz), o sistema apresentou os melhores resultados.

De forma mais específica as conclusões sobre as diferentes estratégias de controle podem ser divididas nos casos mono-tonal com identificação *off-line*, mono-tonal com identificação *on-line* e multi-tonal.

Nos caso mono-tonal off-line os resultados experimentais obtidos demonstraram que a proposta *feedforward* revelou-se bastante eficiente e a inclusão da estimativa do caminho de realimentação (FBFXLMS) melhorou a performance de forma pouco significativa para a configuração estudada.

O controlador *feedback* apresentou também boa performance, contudo o controle revelou-se instável naquelas freqüências onde não foram obtidas boas estimativas da referência. Neste caso o fator de vazamento mostrou-se muito importante, evitando a instabilidade do controle. No caso do sistema Híbrido, observaram-se resultados similares ao sistema *feedforward*, com uma convergência rápida e eficiente, boa estabilidade e maior banda de controle.

Nas metodologias que envolviam estimativas *on-line* dos caminhos acústicos, observou-se que a estratégia de identificação tem que ser conduzida com muito cuidado. Os estudos recomendam evitar identificações simultâneas dos caminhos primários e secundários e que o monitoramento da potência do sinal de erro do sistema pode orientar nas decisões em se estimar ou não este ou aquele caminho ao longo do processo de controle. Neste trabalho, para os algoritmos que utilizam ruído aditivo, decidiu-se estimar o caminho secundário por durante um período de tempo antes de se iniciar a adaptação do caminho primário.

A proposta de Eriksson, que utiliza ruído aditivo para auxiliar na modelagem do caminho secundário, apresentou resultados de elevada performance, com atenuações em toda a banda excitada. A proposta de Zhang, que procura corrigir algumas falhas na teoria da proposta de Eriksson, apresentou melhores performances de atenuação, bem como uma

melhor estimativa do caminho secundário. A dificuldade na escolha do passo de adaptação do preditor $B(z)$ requereu, nos casos estudados, a utilização do fator de vazamento.

Nos estudos multitonais pode-se observar que os algoritmos com modelagem *on-line* que utilizam ruído aditivo não obtiveram grande desempenho. A proposta de Fan – que modela o caminho secundário apenas nas freqüências contidas no ruído primário – apresentou os melhores resultados entre as metodologias avaliadas, sendo o mais indicado para tratar este tipo de problema. A estrutura *Feedback* foi a de pior performance, não sendo indicada para este tipo de problema, pois esta metodologia utiliza apenas um sensor e este não foi capaz de gerar informação necessária para o controlador adaptativo. Esta estrutura é aconselhada para pequenos sistemas físicos ou em fontes de ruído com pouco espalhamento espectral.

De forma geral, a partir das propostas avaliadas experimentalmente pode-se observar que não existe um algoritmo de melhor performance absoluta. A melhor estratégia está intimamente ligada ao sistema físico envolvido e depende fortemente da natureza da excitação. Cabe ao projetista realizar uma análise qualitativa e quantitativa para escolher qual a metodologia ideal para um determinado cenário de cancelamento.

Dos estudos conduzidos alguns desafios apontam para o futuro, onde o campo é bastante vasto. Há inúmeros desafios a serem vencidos no controle ativo de ruído em dutos ou em ambientes tridimensionais. No caso das ondas planas em dutos, a extensão das análises para o caso de múltiplas entradas e múltiplas saídas é um caminho natural.

O emprego de uma controlador supervisor de alto nível que determine quais os melhores algoritmos de controle as serem utilizados, quais caminhos que devem ser (re)estimados, escalonamento do sinal de entrada, dentre outros parâmetros parece algo bastante interessante.

Um outro desafio que se coloca para o problema aqui estudado é de como otimizar a distribuição de sensores e atuadores na planta para uma dada situação de excitação primária. Uma vez que o sistema dinâmico é de modelagem complexa e, em muitas circunstâncias, não estacionário, as implementações usuais mostram-se limitadas.

A despeito de todos estes desafios futuros, naturais e próprios da investigação científica, entende-se que os objetivos postos para este trabalho e formulados na sua introdução, quais sejam: “...estudar as diferentes técnicas de controle ativo de ruído em dutos, discutir suas vantagens e desvantagens e, avaliar experimentalmente algumas delas quando implementadas em um processador digital de sinais” foram alcançados.

REFERÊNCIA BIBLIOGRAFICA

- Ahmed, I., "Digital control applications with the TMS320 family – selected applications notes", Digital Signal Products, Texas Instruments, Houston, USA, 1991.
- Bai, M. R. and Lee, D. J., "Implementation of an active headset by using the H_{∞} robust control theory", Journal of the Acoustical Society of America 102, 2184-2190, 1997.
- Bai, M. R. and LIN, Z., "Active noise cancellation for a three-dimensional enclosure by using multiple-channel adaptive control and H_{∞} control", ASME, Journal of Vibration Acoustics 120, 958-964, 1998.
- Bai, M. R. and Chen, H., "Active cancellation of noise in a car cabin using the zero spillover controller", Journal of Sound and Vibration, 235, (5), 787-800, 2000.
- Bao, C., Sas, P. and Van Brussel, P., "Comparison of two online identification algorithms for active noise control," in Proc. Recent Advances in Active Control of Sound Vibration, pp. 38–51, 1993.
- Berkeley, "Choosing a DSP Processor", Berkeley Design Technology, Inc. (BDTI), 2000.
- Bierke, S., "Enhanced control of an AC motor using fuzzy logic and a TMS320 digital processor", Application Note, Texas Instruments, Houston, USA, 1996.
- Burgess, J. C., "Active Adaptive Sound Control in a Duct: A Computer Simulation", Journal of the Acoustical Society of America, 70, (3), pp. 715–726, September 1981.
- Burlage, K., et al., "An Update of Commercial Experience in Silencing Air Moving Devices with Active Noise Control," Proc. Noise-Con., pp. 253–258, 1991.
- Chaplin, G. B. B., Smith, R. A. and Bramer, T. P. C., "Methods and apparatus for reducing repetitive noise entering the ear", US Patent No. 4654871, 1987.
- Chaplin, G. G. B., and Smith, R. A., "Waveform Synthesis - The Essex Solution to Repetitive Noise and Vibration," Proc. Inter-noise 83, pp. 399–402, 1983.

Chassaing, R. and Horning, D. W., "Digital Signal Processing with the TMS320C25", John Wiley & Sons Inc., 1990.

Clarkson P. M. "Optimal and Adaptive Signal Processing". Boca Raton, FL: CRC Press, 1993.

Conover, W. B., "Noise reducing system for transformers", US Patent No. 2776020, 1957.

Elliott, S. J., and Darlington, P., "Adaptive Cancellation of Periodic, Synchronously Sampled Interference," IEEE Trans. on ASSP, Vol. ASSP-33, No. 3, pp. 715–717, June 1985.

Delfino, L. C., Nuñez, I. J. C., Cardoso, P. M., Ribeiro, J. F., "Active Noise Control Implemented in a DSP Platform", Internoise 2005, Rio de Janeiro, Brazil, 2005.

Elliott, S. J., Stothers, I. M. and Nelson, P. A., "A Multiple Error LMS Algorithm and Its Application to the Active Control of Sound and Vibration", IEEE Trans. on ASSP, Vol. ASSP-35, No. 10, pp. 1423–1434, October 1987.

Eriksson, L. J. and Allie, M. A., "Use of random noise for online transducer estimate in an adaptive active attenuation system," J. Acoust. Soc. Amer., vol. 85, pp. 797–802, Feb. 1989.

Eriksson, L. J., Allie, M. C. and Greiner, R. A., "The Selection and Application of an IIR Adaptive Filter for Use in Active Sound Attenuation", IEEE Trans. on ASSP, Vol. ASSP-35, No. 4, pp. 433–437, April 1987.

Eyre, J. and Bier, J., "The Evolution of DSP Processors", Berkeley Design Technology, Inc. (BDT), 2000.

Fan, H. and Vemuri, R., "Robust adaptive algorithms for active noise and vibration control", IEEE, 1990.

Filter Library Documentation, Texas Instruments, Houston, May 2002.

Gerges, S. N. Y., "Ruído: Fundamentos e Controle", Copyflo, 2^a edição, 2000.

Glover, J. R., "Adaptive Noise Canceling Applied to Sinusoidal Interferences", IEEE Trans. on ASSP, Vol. ASSP-25, No. 6, pp. 484–491, December 1977.

Goodwin G. C. and Sin K. S., "Adaptive Filtering Prediction and Control", Englewood Cliffs, NJ: Prentice-Hall, 1984.

Jackson, L. B., "Digital Filters and Signal Processing", 2nd Edition, Chapter 13, Kluwer Academic Publishers, Norwell, MA. 1989.

Jameco Electronics, "Omni-directional Electret Condenser Microphone Cartridges", Catalog 242, pp. 102, May 2004.

Jessel, M. and Magiante, G. A., "Active sound absorbers en an air duct". Journal of Sound and Vibration, 23, (3), 383-90, 1972.

Kido, K., Reduction of noise by use of additional sound sources". Proceeding of Inter-noise 75: International Conference on Noise Control Engineering, Sendai, Japan, pp. 647-50, 1975.

Kuo S., Panhai I., Chung K., Horner T., Nadeski M., Chyan J., "Design of Active Noise Control Systems With the TMS320 Family", Texas Instruments (SPRA042), 1996.

Kuo, S. M. and Morgan, D.R., "Active Noise Control: A Tutorial Review", in Proceedings of. IEEE, Vol. 87, No. 6, pp. 943–973, June 1999.

Kuo, S. M. and Vijayan, D., "A secondary path estimate techniques for active noise control systems," IEEE Trans. Speech Audio Processing, vol. 5, pp. 374–377, July 1997.

Lueg, P., "Process of silencing sound oscillations", US Patent No. 2043416, 1936.

Marsh, A., "Human Ear and Hearing", School of Architecture and Fine Arts, The University of Western Australia, Online Information and Course Notes, 1999.

Masten, M. K. and Panahi, I., "Digital Signal Processors for Modern Control Systems", Control Eng. Practice, Vol. 5, No. 4, pp. 449-458, 1997.

Minguez, A. O., "Ingeniería Avanzada para Sistemas de Control de Ruido Acústico mediante Técnicas Adaptativas", Universidad Politécnica de Madrid, Tesis Doctoral, 1998.

Morgan, D. R., "An analysis of multiple correlation cancellation loops with a filter in the auxiliary path", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-28, pp. 454–467, Aug. 1980.

Nelson, P. A. and Elliott, S. J., "Active Control of Sound", San Diego, CA: Academic, 1992.

Nishimura, M., "Some Problems of Active Noise Control for Practical Use," Proc. Int. Symp. Active Control of Sound and Vibration, Tokyo, pp. 157–164, 1991.
No. 4,878,188. Oct. 1989.

Nuñez, I. J. C., Cardoso, P. M., Ribeiro, J. F., "Controlador FXLMS Feedback Adaptativo Aplicado a Vibrações de Sistemas Mecânicos", Anais do 3º Congresso Temático de Dinâmica e Controle da SBMAC, UNESP, Ilha Solteira, Junho 2004.

Olson, D. A., Hallstrom, A. D. and Wise, S. S., "Active Noise Control Systems and Air Moving Devices," Proc. Inter-noise 89. pp. 475–478, 1989.

Olson, H. F. and May, E. G., "Electronic sound absorbers", Journal of the Acoustical Society of America, 25, (6), 1130-6, 1953.

Olson, H. F., "Electronic control of noise, vibration and reverberation", Journal of the Acoustical Society of America, 28, (5), 966-72, 1956.

Pfaff, D. D., Kapsokavathis, N. S. and Parks, N. A., "Methods for Actively Attenuating Engine Generated Noise", US Patent 5,146,505. September 1992.

Poole, L. A., Warnaka, G. E. and Cutter, R. C., "The Implementation of Digital Filter Using a Modified Widrow-Hoff Algorithm for the Adaptive Cancellation of Acoustic Noise," Proc. ICASSP, pp. 21.7.1–21.7.4, San Diego, CA, 1984.

Potta, H. R. and Kelkar, A. G., "Modeling and Control of Acoustic Ducts", Journal of Vibration and Acoustics, ASME, vol. 122, 1-9, October 2000.

Rajashekara, K., et al., "Sensorless control of AC motor drives, speed and position sensorless operation", IEEE Press, Piscataway, New Jersey, USA, 1996.

Restle, R.C and Panahi, I., "DSPs redefine motion control", Motion Control Magazine, December 1993.

Simshauser, E. D. and Hawley, M. E., "The noise canceling headset – an active ear defender", Journal of the Acoustical Society of America, 27, (1), 207, 1955.

SPRA366, "Generating Efficient Code with TMS320 DSPs: Style Guidelines", Texas Instruments, Houston, 1997.

SPRA366, "TMS320C2x/C2xx/C5x Optimizing C Compiler User's Guide", Texas Instruments, Houston, 1999.

SPRS145G, "TMS320LF2407A, TMS320LF2406A, TMS320LF2403A, TMS320LF2402A, TMS320LC2406A, TMS320LC2404A, TMS320LC2402A DSP Controllers", Texas Instruments, Houston, 2002.

SPRU357B, "TMS320LF/LC240xA DSP Controllers Reference Guide", Texas Instruments, Houston, 2001.

Tichy, J., and Warnaka, G. E., "Effect of Evanescent Waves on the Active Attenuation of Sound in Ducts", Proceedings of Inter-Noise, pp. 435–438, 1983.

Tokhi, M. O. and Leitch, R. R., "Active Noise Control", Oxford University Press, New York, 1992.

Villadiego, J. C. T, "Sintonia de controladores PID a partir do método do relé", Dissertação de Mestrado, Universidade Federal de Uberlândia, Minas Gerais, Brasil, 2004.

Widrow, B. and Stearns, S. D., "Adaptive Signal Processing", Englewood Cliffs, NJ: Prentice-Hall, 1985.

Widrow, B., Glover, J. R., McCool, J. M., Kaunitz, J., Williams, C. S., Hern, R. H., Zeidler, J. R., Dong, E. and Goodlin, R. C., "Adaptive Noise Canceling: Principles and Applications", Proc. of IEEE, Vol. 63, No. 12, pp. 1692–1716, December 1975.

Widrow, B., Shur D. and Shaffer, S., "On adaptive inverse control", in Proc. 15th Asilomar Conf., pp. 185-189, 1981.

Yuan, J., "A hybrid active noise controller for finite ducts", Applied Acoustics 65, 45-57, 2004.

Zhang, M., Lan, H. and Ser, W., "Cross-Updated Active Noise Control System with Online Secondary Path Modeling", IEEE Transactions on Speech and Audio Processing, vol. 9, No. 5, July 2001.

Ziegler, E. W., "Selective Active Cancellation System for Repetitive Phenomena", U.S. Patent,

ANEXO I

Método do Gradiente Descendente

I.1 Regra Delta de Widrow-Hoff

A regra de aprendizagem de Widrow-Hoff pode ser derivada considerando o nó da Figura I.1, onde D é um vetor de valores desejados e Y é definido pela equação (I.1) como o produto escalar dos pesos e o vetor das entradas.

$$Y = \sum_{i=1}^n w_i x_i \quad (I.1)$$

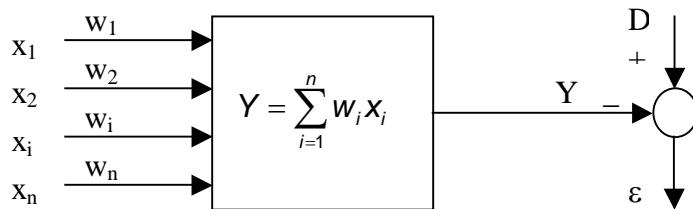


Figura I.1 - Rede com valor desejado D e erro ε .

Da Figura I.1, pode-se ver o erro ε como função de todos os pesos w_i , conforme na equação (I.2).

$$\varepsilon = (D - Y) = (D - \sum_{i=1}^n w_i x_i) \quad (I.2)$$

O erro quadrático ε^2 pode ser visto como:

$$\varepsilon^2 = (D - Y)^2 = (D - \sum_{i=1}^n w_i x_i)^2 \quad (I.3)$$

O gradiente do erro quadrático pode ser calculado pelas derivadas parciais com respeito a cada um dos pesos i .

$$\frac{\partial \varepsilon^2}{\partial w_i} = -2(D - Y) \frac{\partial Y}{\partial w_i} = -2(D - Y)x_i \quad (I.4)$$

Desde que o gradiente envolve somente o i-ésimo peso, a somatória da equação (I.1) desaparece. Para proposta de demonstração, considere uma rede com somente dois pesos w_1 e w_2 . Desta forma, o erro quadrático agora é dado por:

$$\varepsilon^2 = (D - w_1x_1 - w_2x_2)^2 \quad (I.5)$$

O mínimo do erro quadrático ocorre quando as derivadas parciais em relação aos pesos w_1 e w_2 são nulas:

$$\frac{\partial \varepsilon^2}{\partial w_1} = -2(D - w_1x_1 - w_2x_2)x_1 = 0 \quad (I.6)$$

$$\frac{\partial \varepsilon^2}{\partial w_2} = -2(D - w_1x_1 - w_2x_2)x_2 = 0 \quad (I.7)$$

Desde que x_1 e x_2 não podem ser nulos, as quantidades em parênteses, que são idênticas em ambas equações, devem ser nulas. Isto leva a:

$$D - w_1x_1 - w_2x_2 = 0 \quad (I.8)$$

Desta forma, os valores de w_1 e w_2 podem ser dados por:

$$w_1 = \frac{D - w_2x_2}{x_1} \quad (I.9)$$

$$w_2 = \frac{D - w_1x_1}{x_2} \quad (I.10)$$

Substituindo cada um destes valores na equação (I.5) leva o erro quadrático para zero. Tecnicamente isto é correto, porém no mundo real o erro mínimo quadrático nunca é zero, devido às não-linearidades, ruídos indesejáveis e imperfeições dos dados. Este erro que nunca se anula é designado de erro quadrático mínimo ε_{\min}^2 .

A equação (I.5) mostra que o gráfico de ε^2 em relação aos pesos w_1 e w_2 terá a forma parabólica. A curva parabólica do erro quadrático ε^2 em relação ao peso w_1 é mostrada na

Figura I.2 para os dois casos de erro quadrático: zero e ε_{\min}^2 .

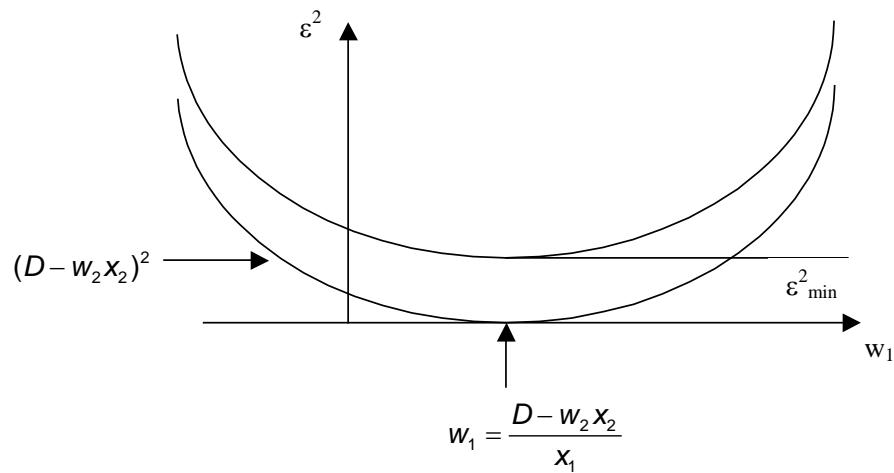


Figura I.2 - Minimização do erro quadrático.

Em ambos os casos, o mínimo ocorre no valor de w_1 dado pela equação (I.9). Um resultado idêntico pode ser obtido para o gráfico do erro quadrático em relação ao peso w_2 , onde o mínimo ocorre no valor dado pela equação (I.10). Quando o erro quadrático é visto em três dimensões (w_1, w_2, ϵ^2), a superfície do erro é um parabolóide de revolução com o vetor de pesos descendo em direção ao valor mínimo através de um vetor gradiente. A projeção deste vetor gradiente no plano w_1, w_2 é um vetor delta como mostra a Figura I.3. O vetor delta move o vetor de pesos na direção do gradiente negativo da superfície para a posição do vetor ideal de pesos. Por causa do método seguir o gradiente, é chamado de *algoritmo do gradiente descendente* ou *algoritmo da máxima descida*.

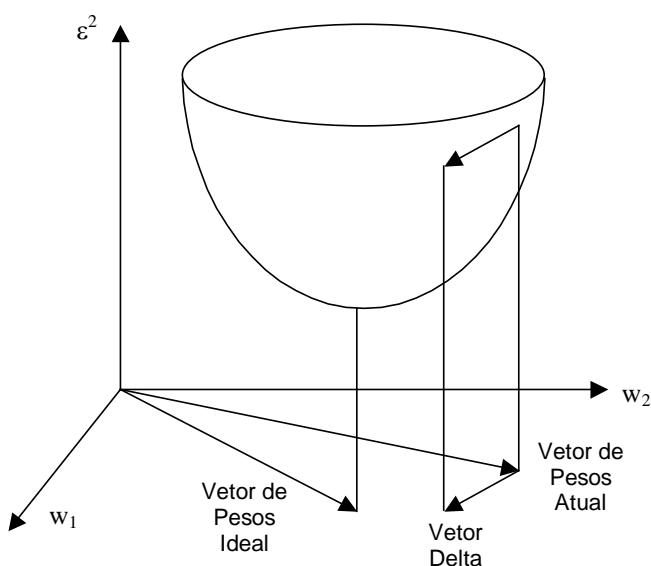


Figura I.3 - Interpretação geométrica do método do gradiente.

Desde que o gradiente é o caminho mais eficiente para a região inferior da superfície, o passo delta é o modo mais eficiente para minimizar o erro quadrático. Isto é verdadeiro se o vetor de pesos é descendente em direção ao mínimo global. Se há mínimos locais, que são comuns em problemas multidimensionais, outras técnicas devem ser utilizadas para garantir que a solução não estacione em um destes mínimos locais.

O método modifica cada componente do vetor de pesos proporcionalmente ao negativo de seu gradiente:

$$\Delta w_i = -\mu \frac{\partial \varepsilon^2}{\partial w_i} = 2\mu \varepsilon x_i \quad (\text{I.11})$$

Onde μ é uma constante de proporcionalidade. O sinal negativo é introduzido por causa do processo de minimização. Supondo um procedimento recursivo, onde as informações anteriores são utilizadas na próxima iteração, pode-se dizer que:

$$w_i(k) = w_i(k-1) + \Delta w_i(k) = w_i(k-1) + 2\mu \varepsilon(k) x_i(k) \quad (\text{I.12})$$

I.2 Passo Adaptativo

O parâmetro μ é passo da minimização que governa a velocidade de convergência do algoritmo, da forma que quando se escolhe um fator μ grande se chega rapidamente ao erro quadrático mínimo, oscilando ao redor com uma determinada magnitude. No entanto se um fator pequeno for escolhido, a velocidade de convergência será lenta e as oscilações sobre o mínimo também serão pequenas.

Supondo que cada entrada da Figura I.1 seja em diferentes tempos amostrais e que o vetor de pesos seja um filtro linear do tipo FIR, de forma que a equação (I.1) agora seja dada por:

$$y(k) = \sum_{i=0}^M w(i)x(k-i) \quad (\text{I.13})$$

Neste caso a equação de adaptação (I.12) pode ser escrita como:

$$w_i(k) = w_i(k-1) + 2\mu \varepsilon(k) x_i(k-i) \quad (\text{I.14})$$

Na equação (I.13) $M+1$ é número de coeficientes do filtro. Os limites do passo de minimização são dados por (Widrow *et al.*, 1985):

$$0 < \mu < \frac{1}{(M+1)\sigma^2} \quad (\text{I.15})$$

Onde σ^2 é a potência do sinal de entrada $x(k)$ do filtro. Se μ estiver fora dos limites dados pela equação (I.15), o algoritmo diverge e não é capaz de encontrar os valores dos coeficientes que minimizam o erro quadrático. O valor habitual escolhido é 10% do valor máximo, ou seja:

$$0 < \mu < \frac{0.1}{(M+1)\sigma^2} \quad (\text{I.16})$$

Para se determinar o passo de minimização é necessário conhecer a potência do sinal de entrada σ^2 . Em um algoritmo recursivo, esta pode ser estimada por (Minguez, 1998):

$$\sigma^2(k) = \alpha x^2(k) + (1-\alpha)\sigma^2(k-1) \quad (\text{I.17})$$

Desta forma, a potência é estimada com base na potência instantânea e no histórico das potências. O parâmetro α depende do número de amostras de $x(k)$ que se utilizam para a estimação, então:

$$\alpha = \frac{1}{N} \quad 0 < \alpha < 1 \quad (\text{I.18})$$

Onde N é o número de amostras. Porém, em estimativas em tempo real, não é possível se determinar quantas amostras serão utilizadas. Neste caso, valores para α abaixo de 0,1 (10%) são bem aceitos para se ter uma boa estimativa.

O erro só se minimiza quando existe alguma correlação entre o sinal desejado D e o sinal de entrada X . Quando o sistema convergir, o erro tenderá para as componentes não correlacionadas com X . Neste caso, se existir algum ruído aditivo não correlacionado, ele será retirado da saída Y . Se não existir correlação alguma entre o sinal desejado D e o sinal da entrada X , os coeficientes do filtro FIR serão praticamente nulos e a saída Y será desprezível. O sinal de entrada X também é conhecido como sinal de referência.

I.3 Algoritmo do Método da Máxima Descida

1. Inicialização:

Coeficientes: $w(i)_0 = 0$, M+1 coeficientes

Potência: $\sigma^2(0) = 1$

Parâmetro α

2. Cálculo da saída do filtro FIR:

$$y(k) = \sum_{i=0}^{M-1} w(i)_k x(k-i)$$

3. Cálculo do erro:

$$\varepsilon(k) = d(k) - y(k)$$

4. Estimação da potência de $x(k)$:

$$\sigma^2(k) = \alpha x^2(k) + (1 - \alpha)\sigma^2(k - 1)$$

5. Cálculo do passo de adaptação:

$$0 < \mu < \frac{0.1}{(M + 1)\sigma^2}$$

6. Atualização dos coeficientes:

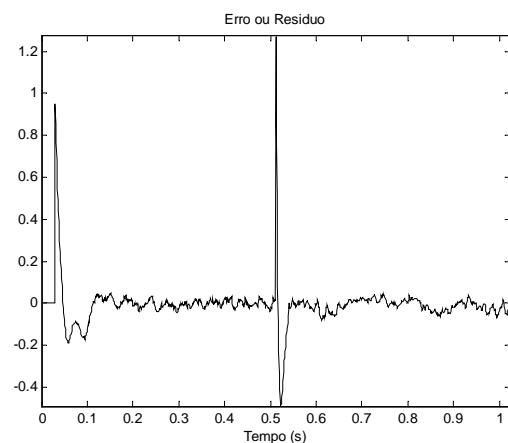
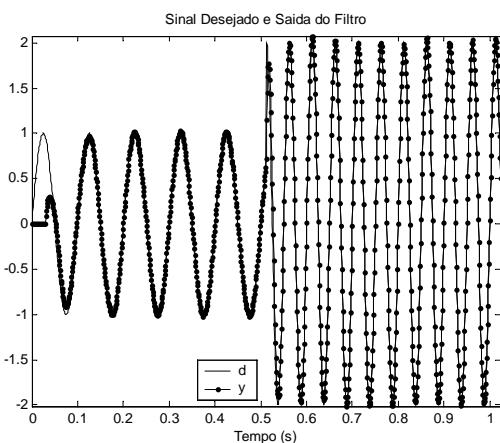
$$w(i)_{k+1} = w(i)_k + 2\mu\varepsilon(k)x(k - i)$$

7. Faça $k=k+1$ e volte ao ponto 2.

I.4 Exemplo

Neste exemplo, deseja-se que o sinal $d(k)$ seja inicialmente um harmônico de 10Hz com amplitude unitária de pico. Depois, simula-se mudanças nos parâmetros de tal forma que o harmônico passe a ter freqüência de 20Hz e amplitude de pico de 2V. Este sinal foi gerado no MATLAB com 1024 pontos e resolução temporal de 1ms. Foi escolhido um filtro adaptativo FIR com 30 coeficientes. O valor de α escolhido foi de 1%.

O sinal desejado $d(k)$ foi gerado utilizando a função *seno*, enquanto que o sinal de entrada $x(k)$ foi gerado utilizando a função *co-seno*. Desta maneira, os dois sinais estão defasados de 90° , porém apesar disto, eles são correlacionados. A correlação entre os dois sinais é uma exigência para a convergência do método. Ao sinal de entrada foi adicionado um ruído branco com $0.1 \text{ V}_{\text{RMS}}$ para se verificar se o sistema será robusto ao ruído aditivo.



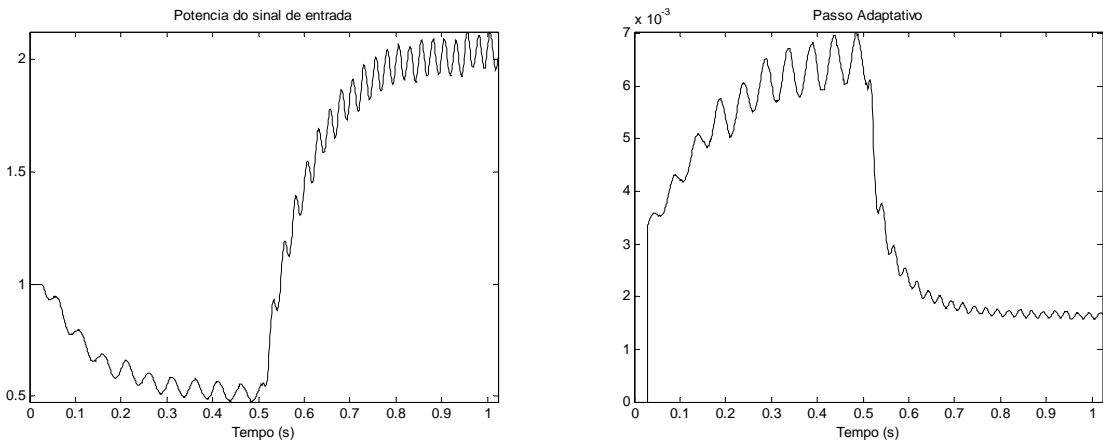


Figura I.4 - Resultados do exemplo. (a) Sinal desejado e saída do filtro FIR.
(b) Erro ou resíduo. (c) Potência estimada da entrada. (d) Passo adaptativo.

Os resultados podem ser visualizados na Figura I.4. Pode-se observar que a saída do filtro FIR conseguiu seguir o sinal desejado, mesmo sendo um sinal não-estacionário. No início e nas mudanças dos parâmetros do processo, pode-se notar um erro grande na estimativa, porém o sistema convergiu rapidamente para o sinal desejado. É importante salientar que o erro contém a parcela do ruído branco aditivo, como era esperado, já que este não possui correlação com o sinal desejado.

A potência do sinal de entrada foi bem estimada pelo método. As oscilações no valor estimado se deve ao valor de α . Quanto maior o valor deste parâmetro, maiores serão as oscilações em torno do valor verdadeiro e mais rápida será a adaptação. Enquanto que valores menores apresentam menos oscilações, no entanto uma adaptação mais lenta. O valor do passo adaptativo μ está relacionado com a estimativa da potência.

ANEXO II

DSP TMS320LF2407A

II.1 Arquitetura do TMS320LF2407A

Serão apresentadas algumas características da plataforma *Starter Kit* TMS320LF2407A da Texas Instruments utilizada nos ensaios experimentais. Estes dispositivos são membros da família TMS320™ de processadores digitais de sinais projetados para uma ampla faixa de sistemas de controle digital. Esta família consiste de DSPs de ponto fixo e ponto flutuante. Ela possui uma arquitetura projetada especialmente para processamento de sinais em tempo real. A série 240xA de DSP combina o processamento em tempo real com periféricos de microcontroladores para criar uma solução ideal nas aplicações de sistemas de controle (SPRU357B, 2001).

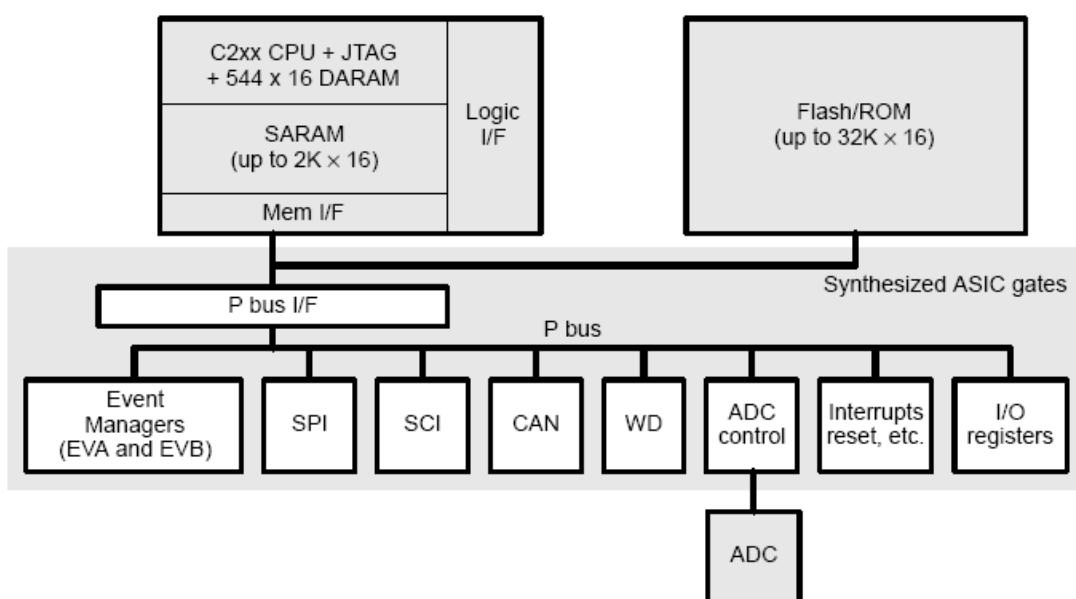


Figura II.1 - Arquitetura do Dispositivo 240xA (SPRU357B, 2001).

A arquitetura da série 240xA utiliza palavras de 16 bits com registradores de 32 bits para armazenar os resultados intermediários e ainda possui dois deslocadores de bits para escalar valores sem o uso da CPU. Esta combinação minimiza os erros de truncamento e quantização.

Estes dispositivos são implementados como processadores digitais de sinais ASIC (cDSPs™). Na CPU, o programa na ROM/FLASH é implementado como uma seqüência de comandos como mostrado na Figura II.1. A CPU utiliza um LP256 que consiste do núcleo TMS320C2xx DSP CPU, uma RAM de duplo acesso (DARAM) de 16 bits de palavra, a interface lógica JTAG, a interface da memória interna (SARAM) e a interface lógica. A interface lógica, entretanto, não é utilizada na série 240xA.

A interface entre periféricos e memória interna da CPU ocorre através da interface PBUS. Todos os periféricos embutidos são acessados através do barramento de periféricos, PBUS. Todos os periféricos, excluindo o *watchdog*, são mantidos pelo *clock* da CPU. O terceiro módulo ASIC é o conversor A/D.

II.1.1 Organização da Memória

Além das memórias RAM de simples acesso (SARAM) e de duplo acesso (DARAM), que faz parte do núcleo da CPU, os dispositivos 240xA incluem memória FLASH EEPROM para adicional memória de programa embutida.

Estes dispositivos tem um barramento de endereço de 16 bits que podem acessar três seções individuais de memória:

- 64k – memória de programa
- 64k – memória de dados
- 64k – memória I/O

Todos 240xA têm 544 palavras de 16 bits de memória DARAM, que podem ser acessadas duas vezes no mesmo ciclo de máquina. As 544 palavras são divididas em três blocos: B0, B1 e B2. Esta memória é geralmente utilizada para dados, no entanto, a B0 pode ser utilizada para programas.

Em operação normal, a CPU lê dados no terceiro ciclo e escreve no quarto. Entretanto, a memória de duplo acesso permite à CPU escrever e ler em um ciclo. A CPU escreve na DARAM na fase mestra do ciclo e lê na fase escrava.

Alguns dos dispositivos 240xA possuem 2K de 16 bits de memória de simples acesso (SARAM). Os endereços associados com esta memória podem ser utilizados tanto para

memória de dados como de programa. A memória SARAM é acessada somente uma vez por ciclo. Quando a CPU necessita de múltiplos acessos, a SARAM programa estes acesso de forma seqüencial, executando somente um acesso por ciclo. Em um acesso duplo, enquanto a DARAM toma um ciclo de máquina, a SARAM utiliza dois ciclos.

A *memória de programa* armazena códigos do usuário, operando e tabelas. O endereçamento de memória de programa permite 64K de palavras de 16 bits. Este número inclui parte da memória DARAM e a memória FLASH EEPROM/ROM.

Dois fatores determinam a configuração da memória de programa:

- Bit CNF: determina onde a DARAM B0 é mapeada:
 - CNF = 0, as 256 palavras são mapeadas como memória externa.
 - CNF = 1, as 256 palavras são mapeadas como memória de programa.
- Pino MP/ \overline{MC} : determina se as instruções de programa são lidas da memória FLASH ou da memória externa:
 - MP/ \overline{MC} = 0, o dispositivo é configurado no modo microcontrolador. A memória FLASH é acessível. O dispositivo aponta o vetor de inicialização para esta memória. O acesso à memória de programa será feito nos endereços de 0000h – 7FFFh no caso do 2407A.
 - MP/ \overline{MC} = 1, o dispositivo é configurado no modo microprocessador. O dispositivo aponta o vetor de inicialização para a memória externa. Neste caso uma imagem da memória externa é criada nos mesmos endereços de 0000h – 7FFFh no caso do 2407A.

A *memória de dados* endereça 64K de 16 bits, onde 32K (0000h – 7FFFh) são de memória interna. Esta inclui registradores, DARAM e registradores dos periféricos. Os outros 32K (8000h – FFFFh) são de memória externa. A memória de dados pode ser acessada tanto por endereçamento direto ou indireto.

A organização da memória de programa, dados e I/O pode ser vista na Figura II.2.

II.1.2 Osciladores

Os dispositivos 240xA utilizam um circuito embutido PLL (*Phase-Locked Loop*) na CPU para sintetizar o *clock* interno a partir de um *clock* externo. O circuito PLL é um circuito oscilador onde a freqüência de saída é múltipla da freqüência de referência (entrada). A série 240xA utiliza um PLL que suporta fatores multiplicativos de 0.5 até 4 vezes a freqüência do *clock* de entrada.

O propósito do PLL é evitar o uso externo à pastilha de sinais de alta freqüência, que podem conter excessivo ruído eletro-magnético causado por rádio-freqüência. A configuração do mesmo é feita através de registradores internos.

II.1.3 Entradas e Saídas Digitais

As portas digitais de I/O (*Input/Output*) proporcionam um método flexível de utilização das funções dedicadas ou de propósito gerais dos pinos. Todas as funções dos pinos são controladas por nove registradores de 16 bits.

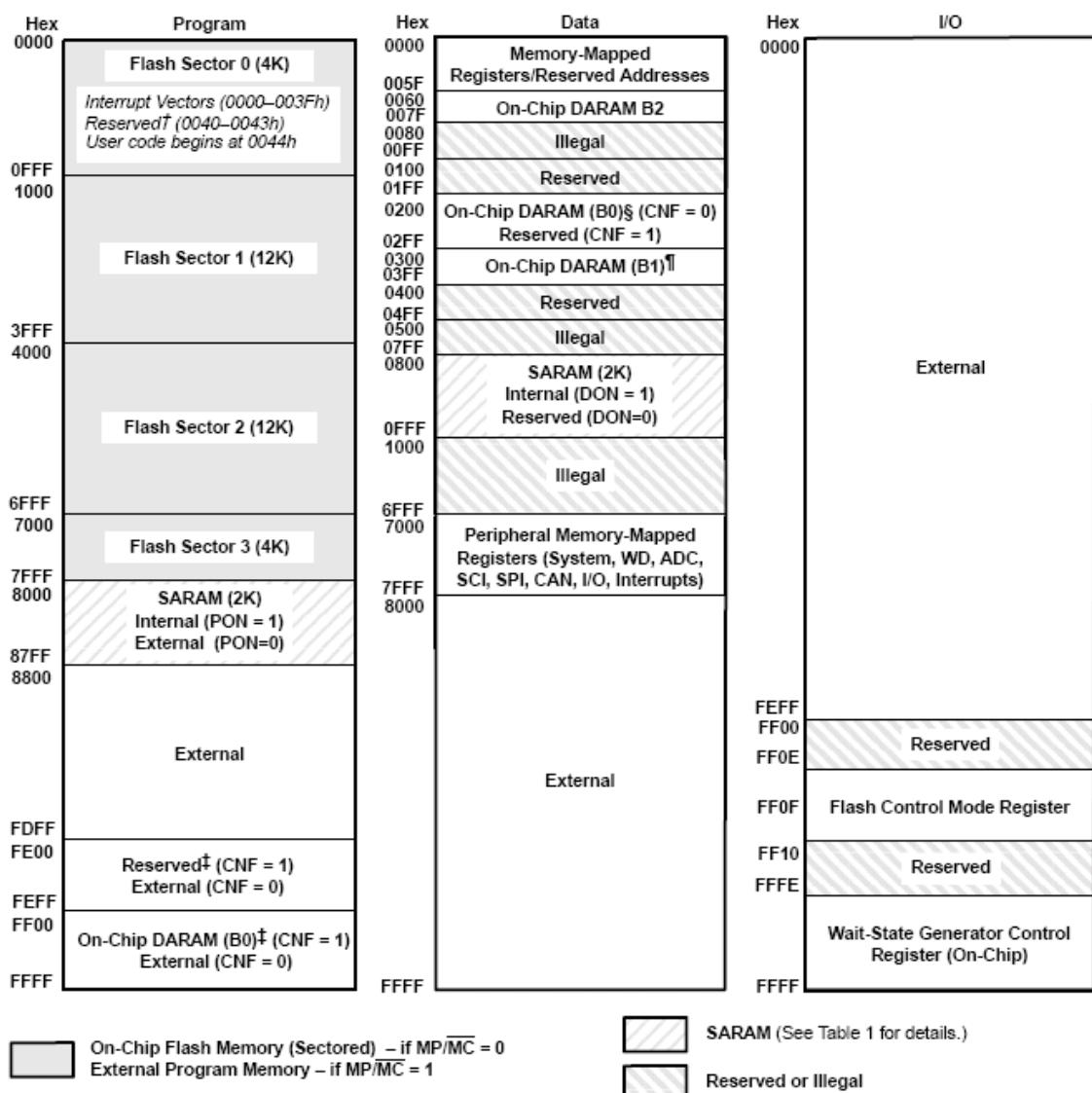


Figura II.2 - Mapa de memória do TMS320LF2407A (SPRS145G, 2002)

Os registradores são divididos em dois tipos:

- *Registradores I/O MUX Control (MCRx)* – Utilizados para o controle entre a função primária do pino ou a função de propósito geral de I/O.
- *Registradores Data and Direction Control (PxDATDIR)* – Utilizados para o controle dos dados e da direção dos pinos bidirecionais.

A estrutura de controle para os pinos compartilhados é mostrada na Figura II.3, onde cada pino possui três bits que definem sua operação:

- *Mux control bit* – este bit seleciona entre a função primária (1) ou função I/O (0).
- *I/O direction bit* – se a função I/O for selecionada para o pino, este bit determina se o pino é uma entrada (0) ou uma saída (1).
- *I/O data bit* - se a função I/O for selecionada para o pino, se a direção selecionada for como entrada, o dado é lido do bit, se a direção escolhida for saída, o dado é escrito neste bit. O estado do bit representa o estado do pino.

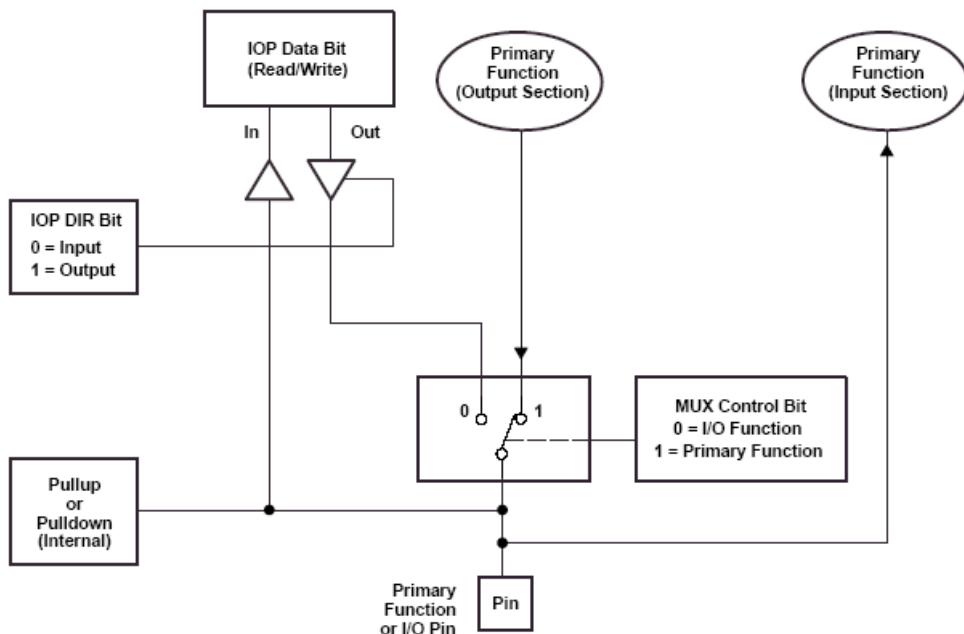


Figura II.3 - Configuração dos pinos compartilhados (SPRS145G, 2002).

II.1.4 Módulos administradores de eventos (EVA, EVB)

Os módulos administradores de eventos (*Event Manager*) possuem temporizadores de propósito geral, unidades de comparação e PWM (*Pulse Width Modulation*), unidades de captura e circuitos encoder de pulsos em quadratura (QEP). O diagrama de blocos pode ser visto na Figura II.4.

Um sinal modulado por largura de pulso (PWM) é uma seqüência de pulsos com larguras variáveis. Os pulsos são propagados sobre um número de períodos de tamanho fixo. Este período fixo é chamado de período do PWM e seu inverso é chamado de freqüência do PWM. As larguras dos pulsos são determinadas, ou moduladas, por uma seqüência de valores desejados, o sinal modulante. A freqüência do sinal modulante é tipicamente muito menor (pelo menos 10 vezes) que a freqüência do PWM.

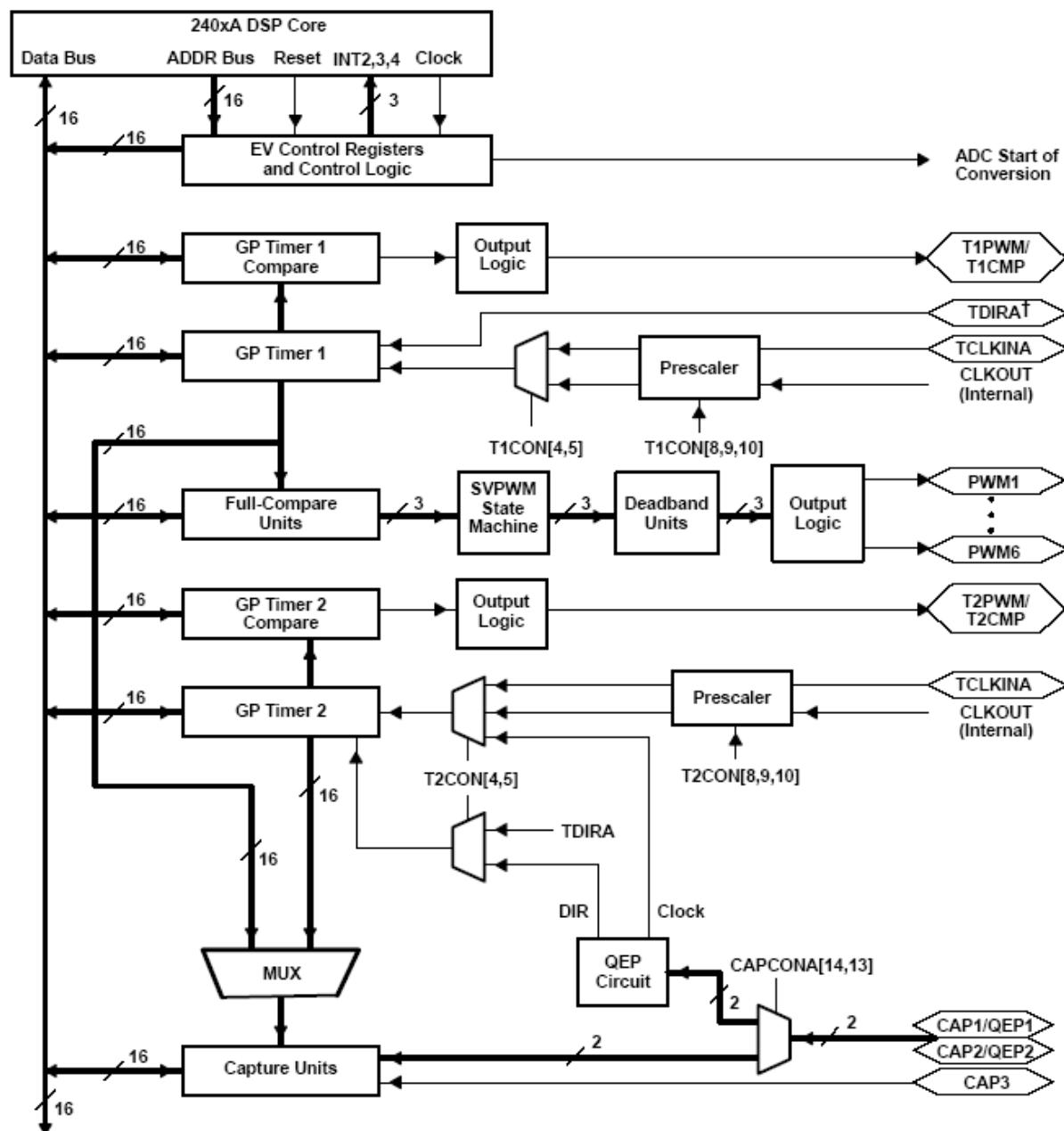


Figura II.4 - Diagrama de Blocos do EVA (SPRS145G, 2002).

Há dois temporizadores de propósito geral GP em cada módulo. Estes podem ser utilizados em diversas funções, dentre elas:

- Geração do período de amostragem em um sistema de controle.
- Prover a base de tempo para a operação de QEP e unidades de captura.
- Prover a base temporal para a geração das saídas PWM.

Os temporizadores podem operar de modos independentes ou sincronizados entre eles. O registrador de comparação associado com cada GP pode ser utilizado para funções de comparação e geração de PWM. Há três modos contínuos de operação para cada temporizador. *Clocks* de entrada interno ou externo com *prescaler* programável para cada GP podem ser utilizados.

Mais de oito sinais de PWM podem ser geradas simultaneamente por cada administrador de eventos: três pares independentes gerados por três unidades de comparação com *deadbands* (utilizado em controle de motores) e duas ondas independentes geradas pelos comparadores *GP-timer*.

II.1.5 Conversores A/D

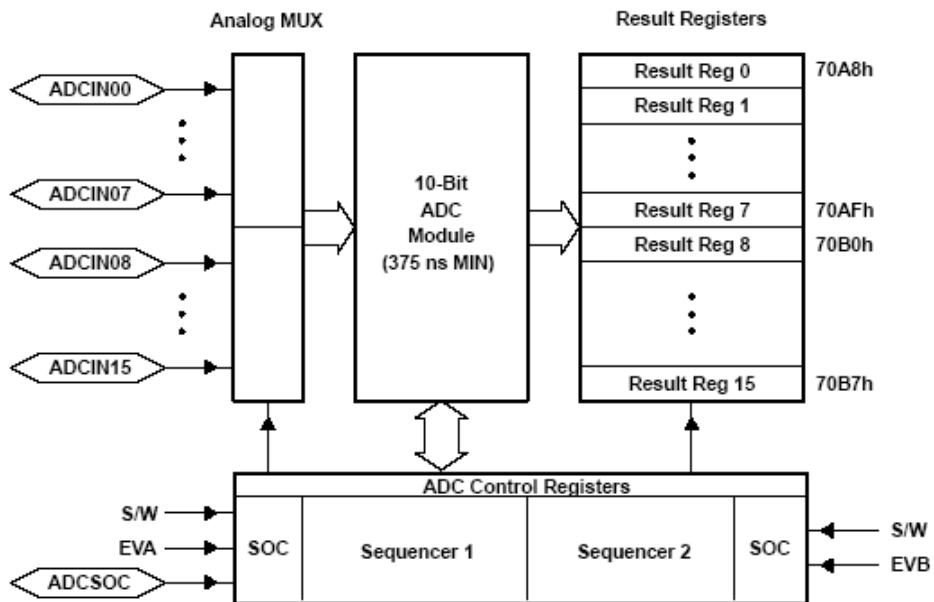


Figura II.5 - Diagrama de blocos do módulo ADC (SPRS145G, 2002).

Um diagrama de blocos funcional do módulo ADC pode ser visto na Figura II.5. Este módulo consiste de somente um conversor A/D de 10 bits com circuito S/H (*Sample-Hold*). Este conversor é compartilhado pelos seqüenciadores. As funções do módulo ADC incluem:

- 16 canais, entradas multiplexadas.
- Seqüenciador automático com capacidade de 16 conversões em uma simples sessão. Cada conversão pode ser programada em uma das 16 entradas.
- Dezesseis registradores de resultados individualmente endereçados.
- Múltiplos *triggers* para o início da conversão:
 - S/W – Início imediato por *software*.
 - EVA – Administrador de eventos A.
 - EVB – Administrador de eventos B.
 - Ext – Pino externo (ADCSOC).
- Controle flexível da interrupção de fim da seqüência de conversão.
- EVA e EVB podem operar de forma independente.
- Tempo de aquisição do S/H com controle de *prescaler* separado.

II.2 Aritmética de Ponto Fixo

Como a família TMS320TM suporta a aritmética de complemento de dois, uma revisão do sistema de complemento de dois deve ser vista. Para facilitar o entendimento, um sistema de 4 bits será utilizado ao invés do sistema de 16 bits utilizado pelo DSP. Uma palavra de 4 bits pode representar números sem sinais (*unsigned*) de 0 a 15. Se uma unidade (1) é adicionada ao maior número (15), o sistema estoura o limite, volta ao primeiro valor e apresenta zero (0) como resultado. Sistemas de bits limitados têm as mesmas propriedades que uma roda numérica, como mostrado na Figura II.6(a). Para a adição de dois números x e y , a operação deve ser feita da seguinte maneira:

- Encontre o número x na roda.
- Contando y unidades na direção do relógio encontra-se o resultado.

Por exemplo, a adição (5+7) traz como resultado o número 12, enquanto que (12+10) nos dá como resultado o número 6.

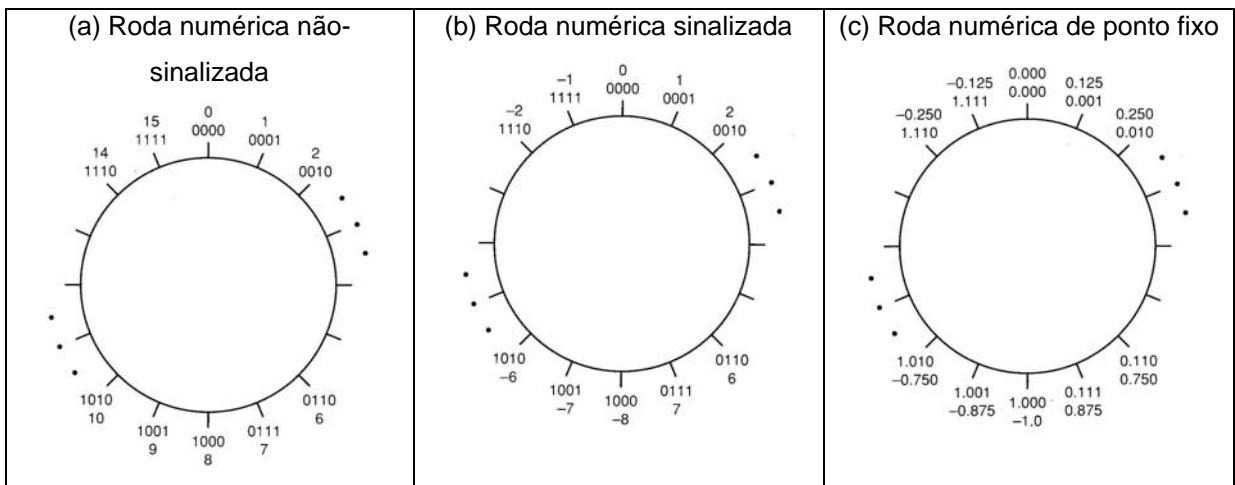


Figura II.6 - Rodas Numéricas (Chassaing et al., 1990).

No entanto, números negativos necessitam de uma interpretação diferente dos números na roda. Imagine agora que a metade do lado direito da roda representa os números positivos e a outra metade, os negativos, como mostra a Figura II.6(b). Esta representação é conhecida como complemento de dois. Os números negativos são complementos de dois dos números positivos, e vice-versa. Um número binário no sistema complemento de dois é dado por:

$$B = b_{n-1} \cdots b_1 b_0 \quad (2.1)$$

O seu inteiro decimal é dado por:

$$D(B) = -b_{n-1} \times 2^{n-1} + \cdots + b_1 \times 2^1 + b_0 \times 2^0 \quad (2.2)$$

Onde b_i são os dígitos binários. Note que o bit de sinal possui um peso negativo, enquanto que todos os outros possuem pesos positivos. Por exemplo, considere o número -2 :

$$1110 = -1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = -8 + 4 + 2 + 0 = -2 \quad (2.3)$$

Para aplicar a técnica gráfica para a operação $6+(-2)$, localize o número 6 na roda, depois conte 1110 ou 14 unidades no sentido do relógio para encontrar a resposta 4 .

Quando -7 é adicionado a -8 no sistema de 4 bits, a resposta será $+1$ ao invés do esperado -15 , que está fora da faixa. Quando dois números de sinais iguais produzem uma resposta com sinal oposto, um *overflow* ocorre no sistema.

Ao invés de se utilizar valores inteiros, como discutido anteriormente, a maioria das aplicações em DSPs utilizam números fracionários de ponto fixo que possuem valores entre $+0.99$ e -1 . Para se obter o número fracionário de n bits, a expressão a seguir pode ser usada:

$$F(B) = -b_0 \times 2^0 + b_1 \times 2^{-1} + \cdots + b_{n-1} \times 2^{-(n-1)} \quad (2.4)$$

Note novamente que o bit de sinal tem um peso negativo 1 e os outros pesos são números positivos de potência de 1/2. A roda representativa pode ser vista na Figura II.6(c). Os números fracionários são obtidos dos números inteiros escalonados (divididos) por 2^3 . Devido ao fato do número de bits no sistema de 4 bits ser pequeno, a faixa vai de -1 a 0.875.

Já a família TMS320TM utiliza palavras de 16 bits, onde os números inteiros estão na faixa entre -32768 a 32767. Para se obter os números fracionários, deve-se escalar os números inteiros por 2^{15} ou 32768, que resulta em uma faixa de -1 a 0.999969 (usualmente dito +1). Este é o formato conhecido como Q15 e a roda numérica é vista na Figura II.6(c).

A multiplicação de dois números sinalizados (*signed*) possui alguns problemas interessantes. A dedução comum é que a multiplicação de dois números de n bits produza um resultado de $2n$ bits. No entanto, isto é verdade somente para números sem sinais (*unsigned*). Lembre-se que os números sinalizados necessitam de um bit de sinal seguido pelos outros pesos positivos.

Considere a multiplicação de dois números fracionais de 4 bits, onde cada número é representado por 3 bits fracionais e um bit de sinal. O produto será representado por 8 bits.

$$\begin{array}{r}
 -0.50 = 1.100 \\
 \times 0.75 = \times 0.110 \\
 \hline
 & 11111000 \\
 & 111000 \\
 \hline
 11.101000 = -2^1 + 2^0 + 2^{-1} + 2^{-3} = -0.375
 \end{array}$$

O bit mais significativo do resultado acima é denominado de bit de sinal estendido e pode ser eliminado. Desta maneira, o produto pode ser representado por 7 bits em um sistema de 4 bit. De forma mais genérica, pode ser representado por $(2n-1)$ bits em um sistema de n bits. Em um sistema de 16 bits, no resultado, o bit 31 é um sinal estendido do bit 30.

Nas operações de um processador de 16 bits, o resultado de 32 bits deve ser truncado ou arredondado para 16 bits. Os bits mais significativos, com o bit de sinal e sua duplicação, estão na parte superior do acumulador. Este resultado deve ser deslocado de um bit para a esquerda para eliminar o bit de sinal extra e obter um bit a mais na resolução, quando movido para a locação de memória de 16 bits.

Na família TMS320TM existe o Modo de Deslocamento do Produto (*Product Shift Mode – PM*). Este é composto de dois bits no Registrador de Status ST1. Quando estes dois bits são 00, o resultado de 32 bits é carregado na Unidade Lógica e Aritmética (ALU) sem deslocamentos. Quando PM=01, o resultado é carregado com o deslocamento de um bit para a esquerda, com o LSB zerado. Se PM=10, o resultado é carregado com o

deslocamento de 4 bits para a esquerda, com o LSB zerado. Se PM=11, há um deslocamento de 6 bits para a direita com a produção de bits de sinal estendido. Os deslocamentos ocorrem na transferência do PREG para a ALU. O registrador PM é carregado pelas instruções SPM e LST #1.

Já que os números na representação Q15 estão entre +1 e -1, não há *overflow* nos resultados da multiplicação. No entanto, nas adições e subtrações, isto ainda é um problema. Para manipular este efeito, a família TMS320TM possui algumas ferramentas:

- *Bit de Overflow (OV)*: Este bit vale 1 quando um *overflow* ocorre na unidade aritmética.
- *Bit de Modo Overflow (OVM)*: Quando este bit é 0, resultados de *overflow* podem ocorrer naturalmente no acumulador. No entanto, quando este vale 1, o acumulador é saturado, como em um sistema analógico, em seu maior ou menor valor, dependendo de qual direção ocorre. As instruções SETC e CLRC são utilizadas para escrever 1 ou 0 no bit, respectivamente.

No entanto, quando se utiliza linguagem C para se implementar os algoritmos de controle no DSP, muitas vezes os bits de *overflow* não são acessíveis. O modo PM também pode ser usado de modo indesejado. Desta forma é melhor utilizá-lo sem deslocamentos (PM=00). Uma forma de se multiplicar dois números no formato Q15 seria (SPRA366, 1997):

```
x = ((long)y * (long)z)>>15;
```

Esta linha de comando permite a obtenção do resultado sem acessar as rotinas de multiplicação de dados de 32 bits, que são mais lentas. A implementação do algoritmo LMS pode ser vista como:

```
uen = ((long)u * (long)en)>>15;
for (i=0; i<N;i++)
    W[i] += ((long)uen * (long)xn[i])>>15;
```

Estas rotinas de multiplicação são mais eficientes quando se faz o uso de ponteiros, como permitido pela estrutura da linguagem C.

II.3 Biblioteca de Processamento de Sinais da Texas Instruments

Na implementação dos algoritmos, o projetista deverá escolher não somente o DSP, mas também o pacote de ferramentas. A escolha de linguagens de alto nível tais como C, Pascal e outras pode tornar o desenvolvimento mais agradável devido ao fato destas linguagens serem mais familiares e de fácil uso.

No entanto, as linguagens de alto nível não são específicas para um tipo de DSP e como resultado, podem não ser otimizadas para aplicações em tempo real. As linguagens de alto nível são tipicamente utilizadas para inicialização e outras tarefas pouco importantes, enquanto que a linguagem *assembly* é utilizada para tarefas críticas em tempo real (Masten *et al.*, 1997).

A maior desvantagem de linguagens *assembly* é sua dependência com o *hardware*, e o fato do usuário necessitar conhecer intimamente o conjunto de instruções para cada DSP. Por outro lado, o conhecimento do *hardware* e da arquitetura proporciona o uso da linguagem *assembly* eficientemente de modo a minimizar o tamanho do código (diminuindo o uso da memória) e minimizar o tempo de execução (otimizar o atraso computacional).

De conhecimento dos fabricantes, estes oferecem bibliotecas (*libraries*) otimizadas para o processamento de sinais, onde rotinas em *assembly* podem ser utilizadas tanto em programas de linguagem de alto nível como em *assembly*, reduzindo assim o tempo de processamento das tarefas críticas.

A *Texas Instruments* oferece bibliotecas de processamento digital de sinais para a série C2000 que incluem geradores de sinais, funções trigonométricas, filtros digitais, transformadas e outras ferramentas.

Para a elaboração desta dissertação foi utilizada a biblioteca de filtros digitais. Esta apresenta várias estruturas para filtros digitais FIR e IIR, tanto de ordem fixa, como de ordem genérica. Dentre os filtros FIR, tem-se a estrutura com o *buffer* linear ou circular de entrada.

II.3.1 Filtro FIR genérico utilizando *buffer* linear

O filtro FIR é essencialmente uma somatória de produtos utilizando um vetor de valores guardados em uma linha de atraso (*buffer* dos valores de entrada), como pode ser visto Figura II.7. A linha de atraso para este módulo é implementada como um *buffer* linear, que necessita de uma movimentação manual dos dados através da linha de atraso. Um dado novo é escrito na posição vazia no topo do *buffer*. A série C24x de DSPs pode realizar esta movimentação de dados em paralelo com o processamento aritmético (instrução DMOV), o uso desta ferramenta está limitada à memória interna. Devido a esta restrição é necessário alojar o *buffer* de entrada do filtro nos blocos de memória interna DARAM B0, B1 e B2 (Filter Library, 2002).

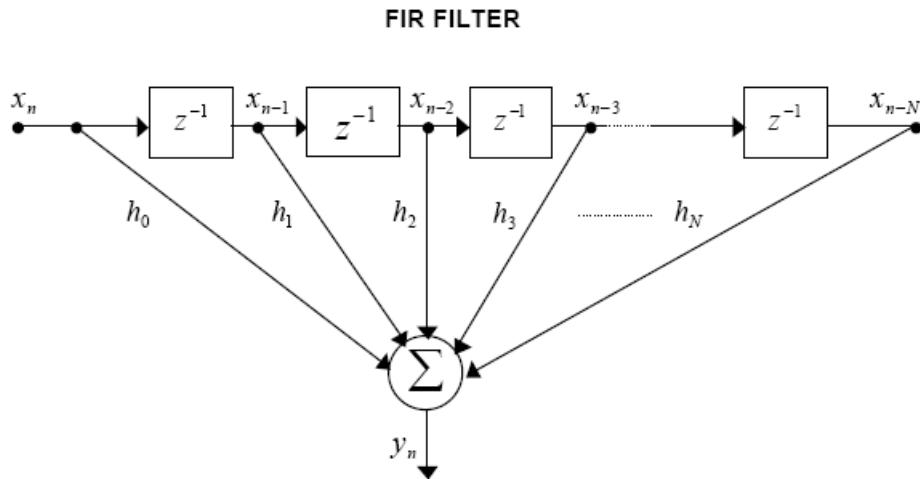


Figura II.7 - Estrutura do filtro FIR.

A estrutura definida pelo objeto FIRFILT_GEN da biblioteca é dada como:

```
typedef struct {
    int *coeff_ptr;
    int *dbuffer_ptr;
    int order;
    int input;
    int output;
    void (*init)(void *);
    void (*calc)(void *);
} FIRFILT_GEN;
```

Variáveis

Tabela II.1 - Tabela das variáveis do objeto FIRFILT_GEN.

Item	Nome	Descrição	Formato	Faixa (Hex)
Entrada	coeff_ptr	Ponteiro para o vetor de coeficientes do filtro. O tamanho do vetor deve ser igual a ordem+1.	N/A	N/A
	dbuffer_ptr	Ponteiro para a linha de atraso. O tamanho deve ser igual a ordem do filtro.	N/A	N/A
	order	Ordem do filtro.	Q0	0002 – 7FFF
	input	Dado de entrada do filtro	Q15	8000 – 7FFF
Saída	output	Saída do filtro digital	Q15	8000 – 7FFF

Constantes especiais e tipos de dados

FIRFILT_GEN: A definição do módulo é criada como um tipo de dado. Para criar vários filtros, simplesmente deve-se declarar variáveis do tipo FIRFILT_GEN.

FIRFILT_GEN_handle: Dado definido pelo usuário para apontar para o módulo.

FIRFILT_GEN_DEFAULTS: Estrutura utilizada para inicializar o módulo.

Funções

void init(FIRFILT_GEN_handle);

```
void calc(FIRFILT_GEN_handle);
```

Estas duas funções são utilizadas para a inicialização e o cálculo da convolução do filtro FIR, respectivamente.

O seguinte pseudocódigo é utilizado como exemplo para o uso do módulo. Os coeficientes do filtro estão disponíveis no arquivo FIRCOEFF.H para um filtro passa baixa com freqüência de corte em 3000Hz.

```
#define FIR_ORDER 50      /* Ordem do filtro FIR */

#pragma DATA_SECTION(fir, "firfilt");      /* Local onde será alocado o objeto filtro */
FIRFILT_GEN lpf = FIRFILT_GEN_DEFAULTS;

#pragma DATA_SECTION(dbuffer,"firldb");    /* Local onde será alocada a linha de atraso */
int dbuffer[FIR_ORDER];

#pragma DATA_SECTION(coeff,"fircoeff");    /* Local onde serão alocados os coeficientes */
int coeff[FIR_ORDER+1]= FIR_LPF50;

main()
{
    lpf.dbuffer_ptr=dbuffer;           /* Ponteiro da linha de atraso */
    lpf.coeff_ptr=coeff;              /* Ponteiro dos coeficientes */
    lpf.order=FIR_ORDER;             /* Ordem do filtro */
    lpf.init(&lpf);                 /* Inicialização do filtro */
}

void interrupt isr20khz()
{
    lpf.input=xn;                   /* Dado de entrada */
    lpf.calc(&lpf);                /* Cálculo da convolução */
    yn=lpf.output;                  /* Saída do filtro */
}
```

Algumas observações devem ser realizadas:

- A seção “firldb” deve ser alocada no bloco DARAM
- O objeto filtro e os coeficientes são alocados nas seções “firfilt” e “fircoeff” respectivamente para se manter a generalidade e organização. Estas duas seções podem ser alocadas em qualquer lugar da memória de dados.
- O código é compacto devido ao fato de ser implementado em laços.
- Os resultados intermediário e final da computação da somatória de produtos são saturados, e a saída do filtro é dada no formato Q15. Desta forma, se o ganho do filtro é maior que o unitário ao longo da banda passante, o usuário necessitará restringir a amplitude da entrada de tal forma que a saída não seja saturada.

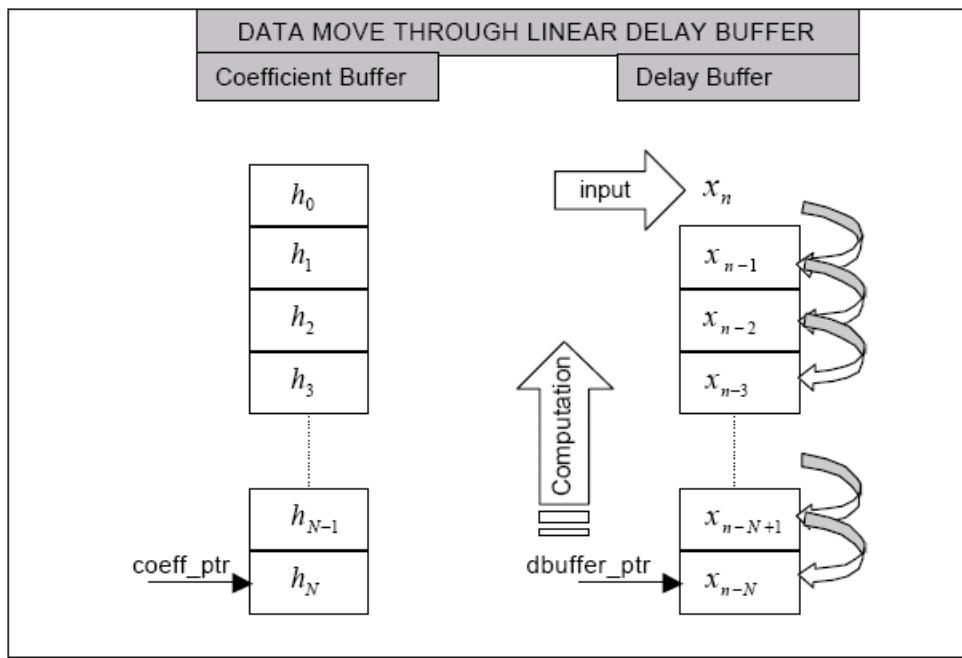


Figura II.8 - Computação do filtro FIR com *buffer* linear.

O diagrama da Figura II.8 mostra a computação do filtro FIR, o procedimento ocorre dos valores mais antigos para os mais novos. O coeff_ptr e dbuffer_ptr são manipulados para o fim do *buffer* na rotina de inicialização.

II.3.2 Filtro FIR genérico utilizando *buffer* circular

Esta rotina utiliza a implementação de bit-reverso de *buffer* circular para a linha de atraso. O *buffer* circular é implementado pela movimentação do ponteiro através dos dados, ao invés da própria movimentação dos mesmos. O único requerimento para a implementação do *buffer* circular é manter o ponteiro na faixa dos dados.

A estrutura definida pelo objeto FIRFILT_GEN da biblioteca é dada como:

```
typedef struct {
    int *coeff_ptr;
    int brindex;
    int order;
    int *dbuffer_ptr;
    int input;
    int output;
    void (*init)(void *);
    void (*calc)(void *);
} FIRFILT_CGEN;
```

Variáveis

Tabela II.2 - Tabela das variáveis do objeto FIRFILT_CGEN.

Item	Nome	Descrição	Formato	Faixa (Hex)
Entrada	coeff_ptr	Ponteiro para o vetor de coeficientes do filtro. O tamanho do vetor deve ser igual a ordem+1.	N/A	N/A
	dbuffer_ptr	Ponteiro para a linha de atraso. O tamanho deve ser igual ao menor numero de potência de 2 que seja maior ou igual a ordem.	N/A	N/A
	order	Ordem do filtro.	Q0	0002 – 7FFF
	input	Dado de entrada do filtro	Q15	8000 – 7FFF
Saída	output	Saída do filtro digital	Q15	8000 – 7FFF
Config	brindex	Valor utilizaddo para implementar o <i>buffer</i> circular utilizando o endereçamento BR.	Q0	0001 – 4000

Constantes especiais e tipos de dados

FIRFILT_CGEN: A definição do módulo é criada como um tipo de dado. Para criar vários filtros, simplesmente deve-se declarar variáveis do tipo FIRFILT_CGEN.

FIRFILT_CGEN_handle: Dado definido pelo usuário para apontar para o módulo.

FIRFILT_CGEN_DEFAULTS: Estrutura utilizada para inicializar o módulo.

Funções

```
void init(FIRFILT_CGEN_handle);
void calc(FIRFILT_CGEN_handle);
```

Estas duas funções são utilizadas para a inicialização e o cálculo da convolução do filtro FIR, respectivamente.

O seguinte pseudocódigo é utilizado como exemplo para o uso do módulo. Os coeficientes do filtro estão disponíveis no arquivo FIRCOEFF.H para um filtro passa baixa com freqüência de corte em 3000Hz.

```
#define FIR_ORDER 50      /* Ordem do filtro FIR */
#define DBUFF_SIZE 64      /* Tamanho do buffer */

#pragma DATA_SECTION(fir, "firfilt");           /* Local onde será alocado o objeto filtro */
FIRFILT_GEN lpf = FIRFILT_CGEN_DEFAULTS;

#pragma DATA_SECTION(dbuffer,"fircdb");        /* Local onde será alocada a linha de atraso */
int dbuffer [DBUFF_SIZE];

#pragma DATA_SECTION(coeff,"fircoeff");        /* Local onde serão alocados os coeficientes */
int coeff[FIR_ORDER+1]= FIR_LPF50;

main()
{
    lpf.dbuffer_ptr=dbuffer;                  /* Ponteiro da linha de atraso */
    lpf.coeff_ptr=coeff;                     /* Ponteiro dos coeficientes */
    lpf.order=FIR_ORDER;                    /* Ordem do filtro */
    lpf.init(&lpf);                        /* Inicialização do filtro */
```

```

}

void interrupt isr20khz()
{
    lpf.input=xn;          /* Dado de entrada */
    lpf.calc(&lpf);        /* Cálculo da convolução */
    yn=lpf.output;         /* Saída do filtro */
}

```

Algumas observações devem ser realizadas:

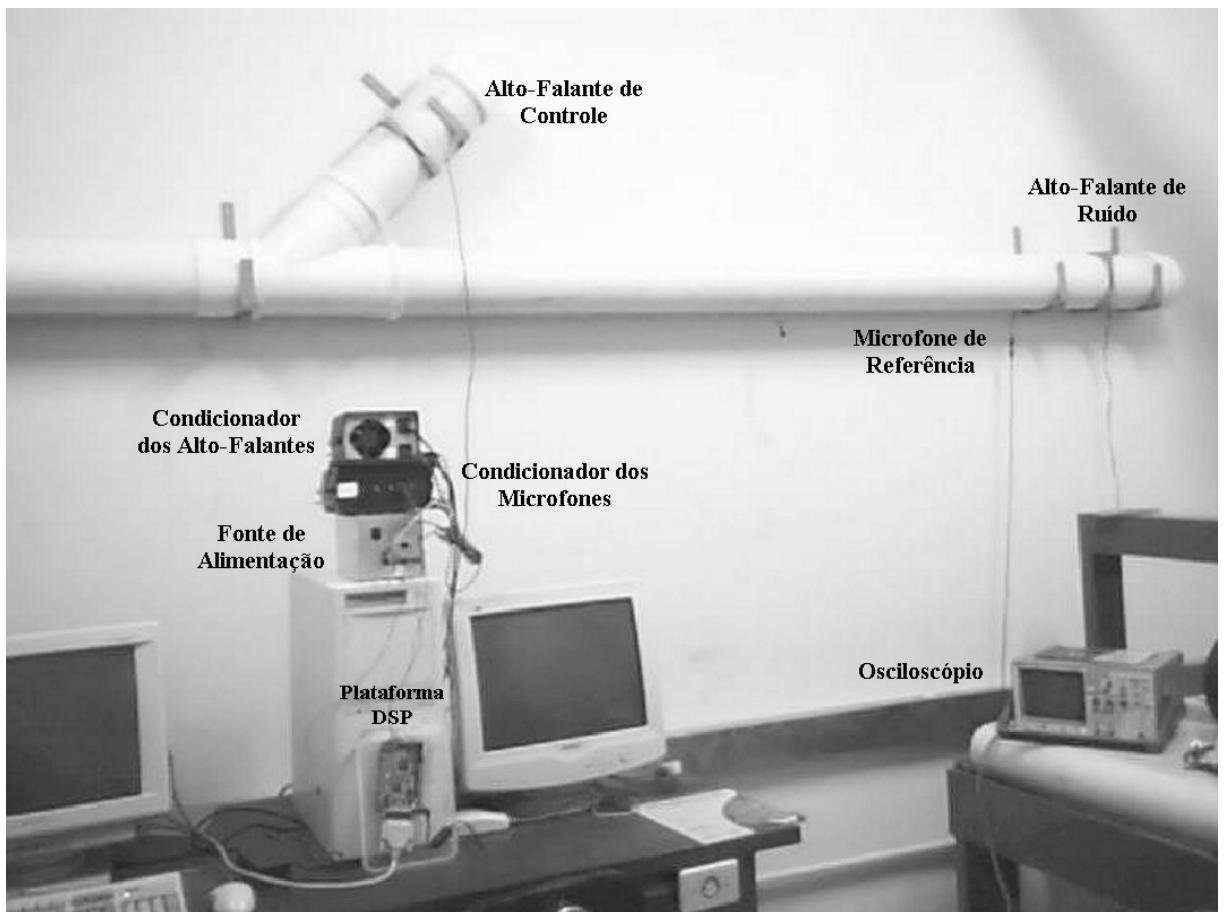
- A seção “fircoeff” pode ser alocada em qualquer local da memória de dados.
- O objeto filtro e os coeficientes são alocados nas seções “firfilt” e “fircoeff” respectivamente para se manter a generalidade e organização. Estas duas seções podem ser alocadas em qualquer lugar da memória de dados.
- O código é compacto devido ao fato de ser implementado em laços.
- Os resultados intermediário e final da computação da somatória de produtos são saturados, e a saída do filtro é dada no formato Q15. Desta forma, se o ganho do filtro é maior que o unitário ao longo da banda passante, o usuário necessitará restringir a amplitude da entrada de tal forma que a saída não seja saturada.

Algumas restrições devem ser colocadas para a utilização deste método:

- O tamanho do *buffer* deve ser um número de potência de 2 maior ou igual a ordem do filtro. Por exemplo, um filtro de ordem 48 deve ter um *buffer* de tamanho 64.
- O *buffer* deve ser alinhado com a borda 2^n .
- Inicialize o *dbuffer_ptr* em qualquer posição do *buffer*, geralmente é utilizado o início do mesmo.
- O valor do índice necessário para o endereçamento de bit-reverso é calculado na inicialização baseado na ordem do filtro.

ANEXO III

Foto da Bancada Experimental



ANEXO IV

Algoritmo ANC – Controle Híbrido

```
#include "stdlib.h"
#include "regs2407.h"
#include "tms320lf2407.h"
#include "regs_fun.h"
#include "firfilt.h"
#include "Myfilter.h"
#include "qmath.h"

/*Define o Periodo de Amostragem                                */
/*T1PERIOD = dt*clk_cpu/prescaler = (1/2000)*40e6/1 */
/*F = 2000Hz -> 20000 */
/*F = 2500Hz -> 16000 */
/*F = 3200Hz -> 12500 */
#define PERIOD      16000 /*16 bits max value           */

/* Ordem de inicializacao dos Filtros                            */
#define FIR_ORDER    32
#define FIR_ORDER2   120
#define BUFFER       128

/****** FEEDFORWARD *****/
=====
/* Filtro adaptativo W com buffer linear que estima o caminho primario */
#pragma DATA_SECTION(W, "firfilt");
    FIRFILT_CGEN_W = FIRFILT_CGEN_DEFAULTS;

/* Buffer de entrada do Filtro W (deve ser localizado na memoria B0) */
#pragma DATA_SECTION(Wdbuffer,"firldb");
int Wdbuffer[FIR_ORDER];

/* Vetor dos coeficientes do Filtro W */
#pragma DATA_SECTION(Wcoeff, "fircoeff");
int Wcoeff[FIR_ORDER+1];

=====
/* Filtro S com buffer circular que estima o caminho secundario */
#pragma DATA_SECTION(S, "firfilt");
```

```

FIRFILT_CGEN S= FIRFILT_CGEN_DEFAULTS;

/* Define o Buffer de entrada do filtro S, deve estar alinhado com os
limites
da seção e deve ter o tamanho multiplo de potencias de 2(2n) que seja maior
ou
igual a ordem do filtro */
#pragma DATA_SECTION(Sdbuffer,"fircdb");
int Sdbuffer[BUFFER];

/* Vetor dos coeficientes do Filtro S      */
#pragma DATA_SECTION(Scoeff, "fircoeff");
int Scoeff[FIR_ORDER2+1]= FIR_120_SECONDARY_2500Hz;

/*
=====
=
/* Filtro D com buffer circular que estima o caminho de realimentação  */
#pragma DATA_SECTION(D, "firfilt");
    FIRFILT_CGEN D= FIRFILT_CGEN_DEFAULTS;

/* Define o Buffer de entrada do filtro D, deve estar alinhado com os
limites
da seção e deve ter o tamanho multiplo de potencias de 2(2n) que seja maior
ou
igual a ordem do filtro */
#pragma DATA_SECTION(Ddbuffer,"fircdb");
int Ddbuffer[BUFFER];

/* Vetor dos coeficientes do Filtro D      */
#pragma DATA_SECTION(Dcoeff, "fircoeff");
int Dcoeff[FIR_ORDER2+1]= FIR_120_RETURN_2500Hz;

***** FEEDBACK *****/
/*
=====
=
/* Filtro adaptativo A com buffer linear que estima o caminho primário  */
#pragma DATA_SECTION(A, "firfilt");
    FIRFILT_CGEN A = FIRFILT_CGEN_DEFAULTS;

/* Buffer de entrada do Filtro W (deve ser localizado na memoria B0) */
#pragma DATA_SECTION(Adbuffer,"firldb");
int Adbuffer[FIR_ORDER];

/* Vetor dos coeficientes do Filtro W      */
#pragma DATA_SECTION(Acoeff, "fircoeff");
int Acoeff[FIR_ORDER+1];

/*
=====
=
/* Filtro B com buffer circular que estima o caminho secundário  */
#pragma DATA_SECTION(B, "firfilt");
    FIRFILT_CGEN B= FIRFILT_CGEN_DEFAULTS;

/* Define o Buffer de entrada do filtro S, deve estar alinhado com os
limites

```

```

da seção e deve ter o tamanho multiplo de potencias de 2(2n) que seja maior
ou
igual a ordem do filtro */
#pragma DATA_SECTION(Bdbuffer,"fircdB");
int Bdbuffer[BUFFER];

/*
=====
=
/* Filtro C com buffer circular que estima o caminho secundario */
#pragma DATA_SECTION(C, "firfilt");
    FIRFILT_CGEN  C= FIRFILT_CGEN_DEFAULTS;

/* Define o Buffer de entrada do filtro S, deve estar alinhado com os
limites
da seção e deve ter o tamanho multiplo de potencias de 2(2n) que seja maior
ou
igual a ordem do filtro */
#pragma DATA_SECTION(Cdbuffer,"fircdB");
int Cdbuffer[BUFFER];

/*
=====
=
/* Filtro xf com buffer linear que estima o caminho secundario */
#pragma DATA_SECTION(xf, "firfilt");
    FIRFILT_GEN  xf= FIRFILT_GEN_DEFAULTS;

/* Buffer de saída do sinal filtro FXLMS, esta estrutura é utilizada
para rotacionar o buffer de forma mais rápida */
#pragma DATA_SECTION(xfbuffer,"firldb");
int xfbuffer[FIR_ORDER+1];

/*
=====
=
/* Filtro df com buffer circular que estima o caminho secundario */
#pragma DATA_SECTION(df, "firfilt");
    FIRFILT_GEN  df= FIRFILT_GEN_DEFAULTS;

/* Buffer de saída do sinal filtro FXLMS, esta estrutura é utilizada
para rotacionar o buffer de forma mais rápida */
#pragma DATA_SECTION(dfbuffer,"firldb");
int dfbuffer[FIR_ORDER+1];

/*Variáveis utilizadas no controle */
int *xn,      /* Sinal de entrada do FeedForward */
     *dn,      /* Sinal de entrada do FeedBack */
     *yn,      /* Sinal de saída do controle */
     *en,      /* Sinal de erro do AD */
     *un,      /* Sinal do mic de referencia */
     *pot,     /* Potencia estimada do sinal filtrado FF */
     *p,       /* Potencia estimada do sinal filtrado FB */
     *mi,      /* Passo adaptativo FF */
     *m,       /* Passo adaptativo FF */
     *alfal,   /* Fator de esquecimento da estimativa da potencia */
     *alfa2,   /* Fator de esquecimento da estimativa da potencia */
     *uen,
     *aux;

```

```

/*Variáveis utilizadas para selecionar a estrutura */
int    feedback, feedforward, FB;

int ad1=0, ad1_ant=0, ad2=0, ad2_ant=0, un_ant=0, en_ant=0;
int n = 0;

/*********************************************************/
/* Função vazia para se detectar interrupções indesejadas */
void interrupt phantom(void)
{
    static int phantom_count;
    phantom_count++;
}

/*********************************************************/
/* Laço do controle que ocorre a cada período de amostragem */
interrupt void control(void)
{
    unsigned int out, i;

    if (PIVR == ADCINT)      /* Testa a interrupção ADC */
    {
        /* Leitura do AD */
        ad1=(RESULT0-32768); /* Bipolar Q-15 */
        /*Filtro notch para 0Hz*/
        *un = ad1 - ad1_ant;
        *un += ((long)29000*(long)un_ant)>>15;
        ad1_ant = ad1;
        un_ant = *un;

        /* Leitura do AD */
        ad2=(RESULT1-32768); /* Bipolar Q-15 */
        /*Filtro notch para 0Hz*/
        *en = ad2 - ad2_ant;
        *en += ((long)29000*(long)en_ant)>>15;
        ad2_ant = ad2;
        en_ant = *en;

        if (n>20000)
        {
            /***** FEEDFORWARD *****/
            if (feedforward)
            {
                if (FB)
                {
                    /* Computa a referência livre da realimentação
 */
                    D.input = *yn;
                    D.calc(&D);
                    *xn = *un - D.output;
                }
                else *xn = *un;
                /* Estimativa do anti-ruido pelo sinal */
                W.input = *xn;
                W.calc(&W);

                /* Calculo da entrada filtrada */
                S.input = *xn;
                S.calc(&S);
            }
        }
    }
}

```

```

/*Rotacao do buffer dos dados da entrada filtrada */
xf.input = S.output;
    xf.calc(&xf);

        /* Estimacao da potencia da entrada filtrada xf */
        *uen = xfbuffer[0];
*aux = ((long)*uen*(long)xfbuffer[0])>>15;
*uen = ((long)*alfa1*(long)*aux)>>15;
*aux = ((long)*alfa2*(long)*pot)>>15;
*pot = *uen + *aux +1;

/* Calculo do passo de adaptacao mi                         */
*mi = ((int)32767/(*pot));
*mi *= 10;

/* Atualizacao dos coeficientes      */
/*W(i)=W(i-1) -2*u*e(n)*xf(n-i) */
*uen = ((long)*mi*(long)*en)>>15;
*aux = 32766; /*Fator de Vazamento*/
for (i=0;i<=W.order;i++)
{
    Wcoeff[i] = ((long)*aux*(long)Wcoeff[i])>>15;
    Wcoeff[i] -= ((long)*uen*(long)xfbuffer[i])>>15;
}

}
***** FEEDBACK *****
if (feedback)
{
    /* Calculo da referencia estimada dn(n)   */
    C.input = *yn;
    C.calc(&C);
    *dn = *en - C.output;

/* Calculo do anti-ruido */
A.input = *dn;
A.calc(&A);

/* Calculo da entrada filtrada      */
B.input = *dn;
B.calc(&B);

/*Rotacao do buffer dos dados da entrada filtrada */
df.input = B.output;
    df.calc(&df);

        /* Estimacao da potencia da entrada filtrada xf */
        *uen = dfbuffer[0];
*aux = ((long)*uen*(long)dfbuffer[0])>>15;
*uen = ((long)*alfa1*(long)*aux)>>15;
*aux = ((long)*alfa2*(long)*p)>>15;
*p = *uen + *aux + 1;

/* Calculo do passo de adaptacao mi                         */
*m = ((int)32767/(*p));
*m *= 5;

/* Atualizacao dos coeficientes      */
/*A(i)=A(i-1) -2*m*e(n)*df(n-i) */

```

```

        *uen = ((long)*m*(long)*en)>>15;
        *aux = 32760;
        for (i=0;i<=A.order;i++)
        {
            Acoeff[i] = ((long)*aux*(long)Acoeff[i])>>15;
            Acoeff[i] -= (((long)*uen*(long)dfbuffer[i])>>15);
        }
    }

/* Compoa a saída baseada nas duas estruturas */
if (feedforward && feedback)      *yn = A.output + W.output;
else if (feedforward)              *yn = W.output;
else if (feedback)                *yn = A.output;
else                                *yn = 0;

/*Saida na resoluçao do PWM      */
out = *yn + 32768; /* Unsigned */
T1CMPR = ((long)out*(long)PERIOD)/65536;
}
if (n < 20000) n++; else n = 20001;
/* Reset SEQ1 and clear ADC SEQ1 int flag */
setup_adctrl2(ADCTRL2|ADC_RST_SEQ1|ADC_INT_FLAG_SEQ1);

}

void main()
{
    unsigned int i;
    /*O espaço é importante */
    asm (" setc INTM"); /* Desabilita todas interrupções */ *
    asm (" clrc SXM"); /* Limpa o bit de sinal */
    /*
    asm (" setc OVM"); /* Modo de saturação de Overflow */
    /*
    asm (" clrc CNF"); /* Configura o bloco BO como dados */
    /*
    asm (" spm 0 "); /* Multiplicação no modo normal */
    */

    /*WaitStates - 0 */
    setup_wsg(BVISOFF|ISWS0|DSWS0|PSWS0);

    /*SCSR1
    CLK=40MHz
    Habilita clock do ADC
    Habilita clock do EVA
    Limpa o bit de endereçamento ilegal */
    setup_scsr1(CPUCLK|LPM0|CLK_PS4|ADC_CLK_ENA|EVA_CLK_ENA|ILLADR);

    /*Desabilita o Watchdog */
    setup_wdcr(WD_WDDIS|WD_NORMAL|WD_PRE64);

    /*MCRA - Saída T1PWM      */
    setup_mcra(T1PWM);

    /*MCRB - Pinos de I/O */
    setup_mcrb(W_R|BIO|IOPC2|IOPC3|IOPC4|IOPC5|IOPC6|IOPC7);
}

```

```

/*MCRC - Pinos de I/O */
setup_mcrc(IOPE0|IOPE1|IOPE2|IOPE3|IOPE4|IOPE5|IOPE6|IOPE7);

/*T1CON
Emulaçao termina ao final do período
Modo de contagem: continuous-up
Timer Clock Prescaler: 1
Desabilita o Temporizador
Clock do Temporizador na fonte interna
Recarrega quando o contador=0|per
Utiliza o periodo do Temporizador 1
Habilita a operaçao de comparaçao */

setup_t1con(TIMER_SOFT_STOP|TIMER_CONT_UP|TIMER_CLK_PS1|TIMER_DISABLE|TIMER
_CLK_SRC_INT|TIMER_LD_ON_ZERO_OR_PRD|TIMER_OWN_PERIOD|TIMER_CMP_ENA);

/*GPTCONA
Começa o ADC pela interrupcao do periodo T1
Polarizaçao da saída do comparador GPT1 = ativa alta
Habilita todas as saídas do GPT */

setup_gptcona(GPT_T2TOADC_DIS|GPT_T1TOADC_PER|GPT_T1PIN_AHIGH|GPT_TCOMPOE_E
NA);

/*IMR - ADC prioridade alta -> habilita INT1 ADC
   ADC prioridade baixa -> habilita INT6 ADC
*/
setup_imr(INT1_ENA);

/*ADCTRL1
Completa a conversao antes de parar
Timer Clock Prescaler: 16xTclk
ADC logic Clock = CLK/2
Start-stop mode
Prioridade alta na interrupcao do ADC
Modo sequencial duplo
Modo de calibraçao desativado
*/
setup_adctrl1(ADC_SOFT_STOP|ADC_CLK_PS8|ADC_CPS2|ADC_START_STOP|ADC_INT_HIG
H|ADC_SEQ_DUAL|ADC_CAL_DIS|ADC_BRG_FULL);

/*ADCTRL2
Reseta o sequenciador 1 (CAL ENA = 0)
Interrupcao imediata
EVA inicia o sequenciador 1
*/
setup_adctrl2(ADC_RST_SEQ1|ADC_INT_SEQ1_MODE1|ADC_EVA_SOC_SEQ1);

/*Os Registradores para EVAIMRx não precisam ser ajustados,
devido ao fato de não existir interrupções de temporizadores */

T1PR = PERIOD;      /* Inicializa o período do T1 */
T1CMPR = T1PR/2; /* Ciclo do PWM */                      */
T1CNT = 0;           /* Valor inicial para o contador T1 */

MAXCONV = 1;        /*2 Conversões (N-1) */

/*CHSELSEQ1 - Sequenciador dos canais AD */

```

```

setup_chselseq1((CH0)|(CH1<<4));

EVAIFRA=0xFFFF;      /* clear EV Interrupt Flag Register Group A */
EVAIFRB=0xFFFF;      /* clear EV Interrupt Flag Register Group B
*/
EVAIFRC=0xFFFF;      /* clear EV Interrupt Flag Register Group C
*/
IFR = 0xFFFF; /* Reseta todas as interrupções pendentes */

/* Inicializações das variáveis de controle */
xn    = malloc(1);
dn    = malloc(1);
yn    = malloc(1);
en    = malloc(1);
un    = malloc(1);
pot   = malloc(1);
p     = malloc(1);
mi    = malloc(1);
m     = malloc(1);
alfal= malloc(1);
alfa2= malloc(1);
uen   = malloc(1);
aux   = malloc(1);

*pot = *p = 1000;
*alfal = 3250;
*alfa2 = (32768-3250);

/* Inicialização dos Filtros */
W.dbuffer_ptr = Wdbuffer;
W.coeff_ptr = Wcoeff;
W.order = FIR_ORDER;
W.init(&W);

S.dbuffer_ptr = Sdbuffer;
S.coeff_ptr = Scoeff;
S.order = FIR_ORDER2;
S.init(&S);

D.dbuffer_ptr = Ddbuffer;
D.coeff_ptr = Dcoeff;
D.order = FIR_ORDER2;
D.init(&D);

A.dbuffer_ptr = Adbuffer;
A.coeff_ptr = Acoeff;
A.order = FIR_ORDER;
A.init(&A);

B.dbuffer_ptr = Bdbuffer;
B.coeff_ptr = Scoeff;
B.order = FIR_ORDER2;
B.init(&B);

C.dbuffer_ptr = Cdbuffer;
C.coeff_ptr = Scoeff;
C.order = FIR_ORDER2;
C.init(&C);

```

```
xf.dbuffer_ptr = xfbuffer;
xf.coeff_ptr = Wcoeff;
xf.order = FIR_ORDER+1;
xf.init(&xf);

df.dbuffer_ptr = dfbuffer;
df.coeff_ptr = Acoeff;
df.order = FIR_ORDER+1;
df.init(&df);

for (i=0;i<=W.order;i++)
    Wcoeff[i]=0;
    for (i=0;i<=A.order;i++)
        Acoeff[i]=0;

/* Sinal de Trigger */
PEDATDIR = 0xFF01;

/* Habilita a chave geral das interrupções      */
asm(" clrc INTM");

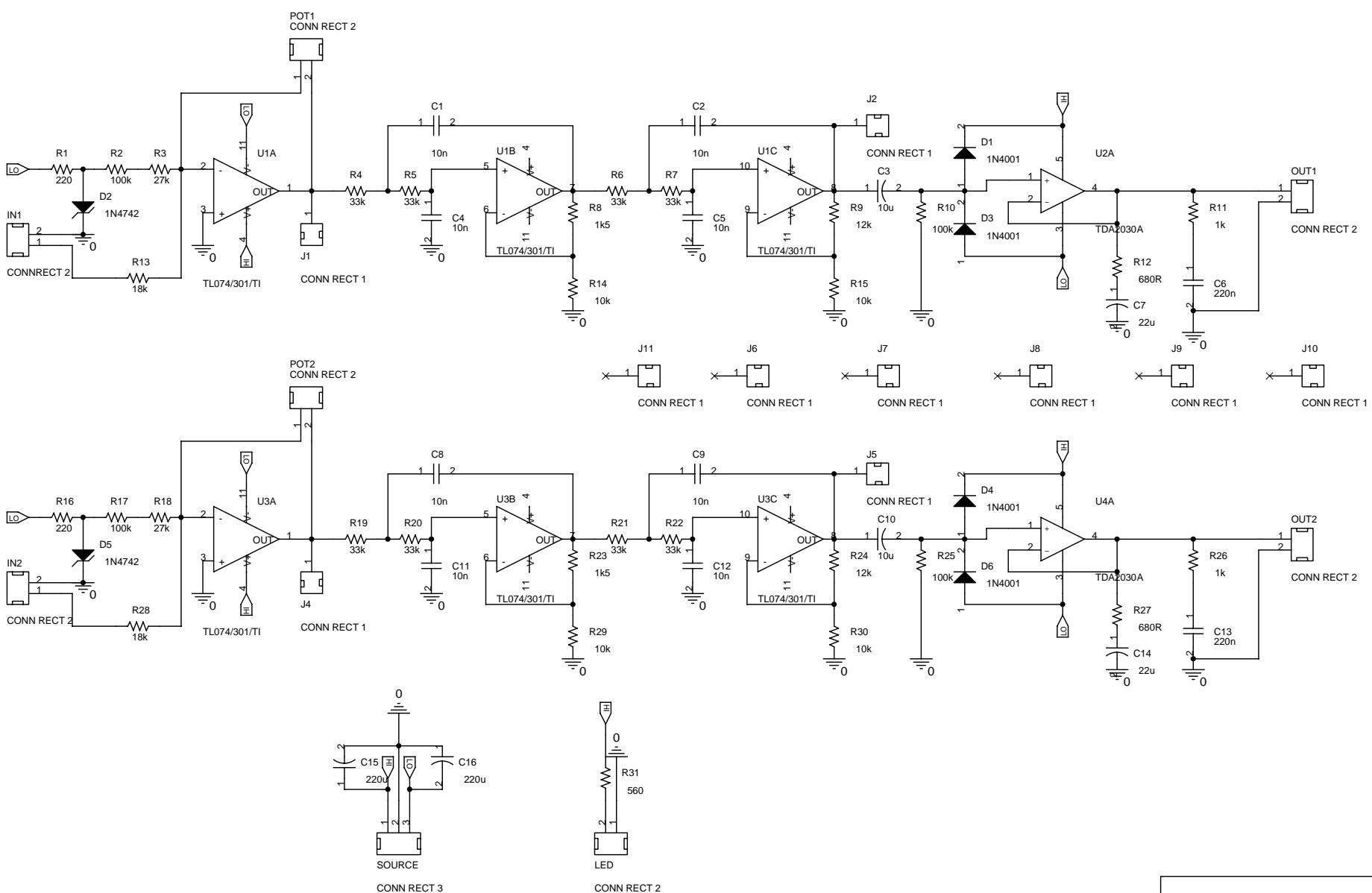
/* Habilita o T1      */
setup_t1con(T1CON|GPT_TCOMPOE_ENA);

/* LOOP PRINCIPAL */
while(1);

    exit(1);
}
```

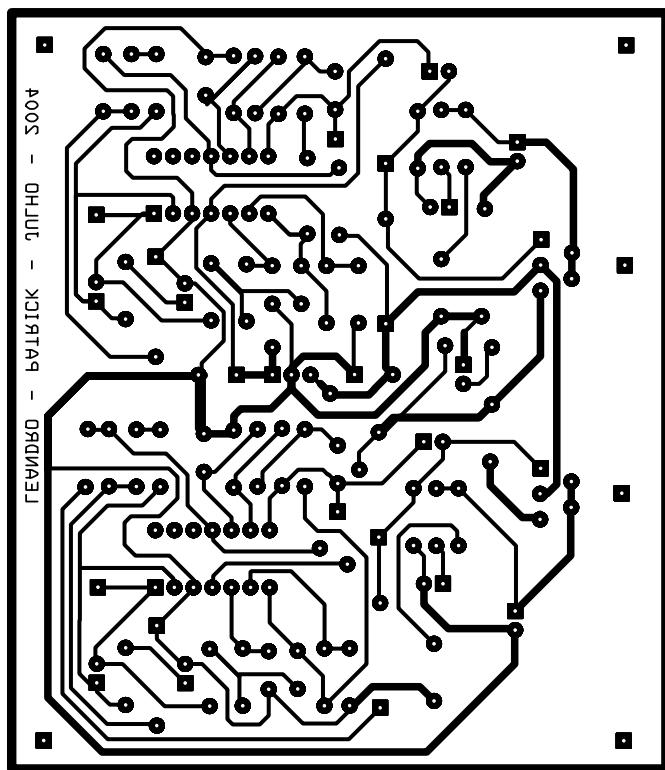

ANEXO V

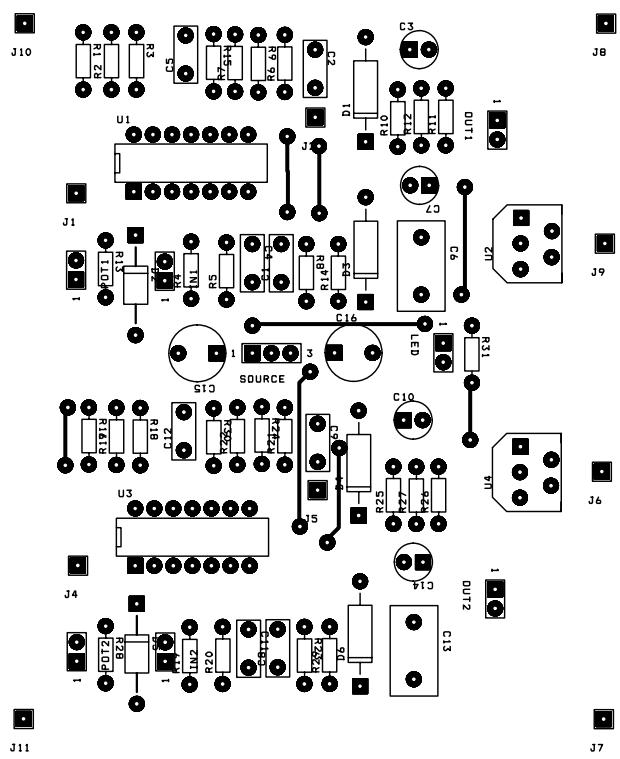
Diagrama dos Circuitos Eletrônicos

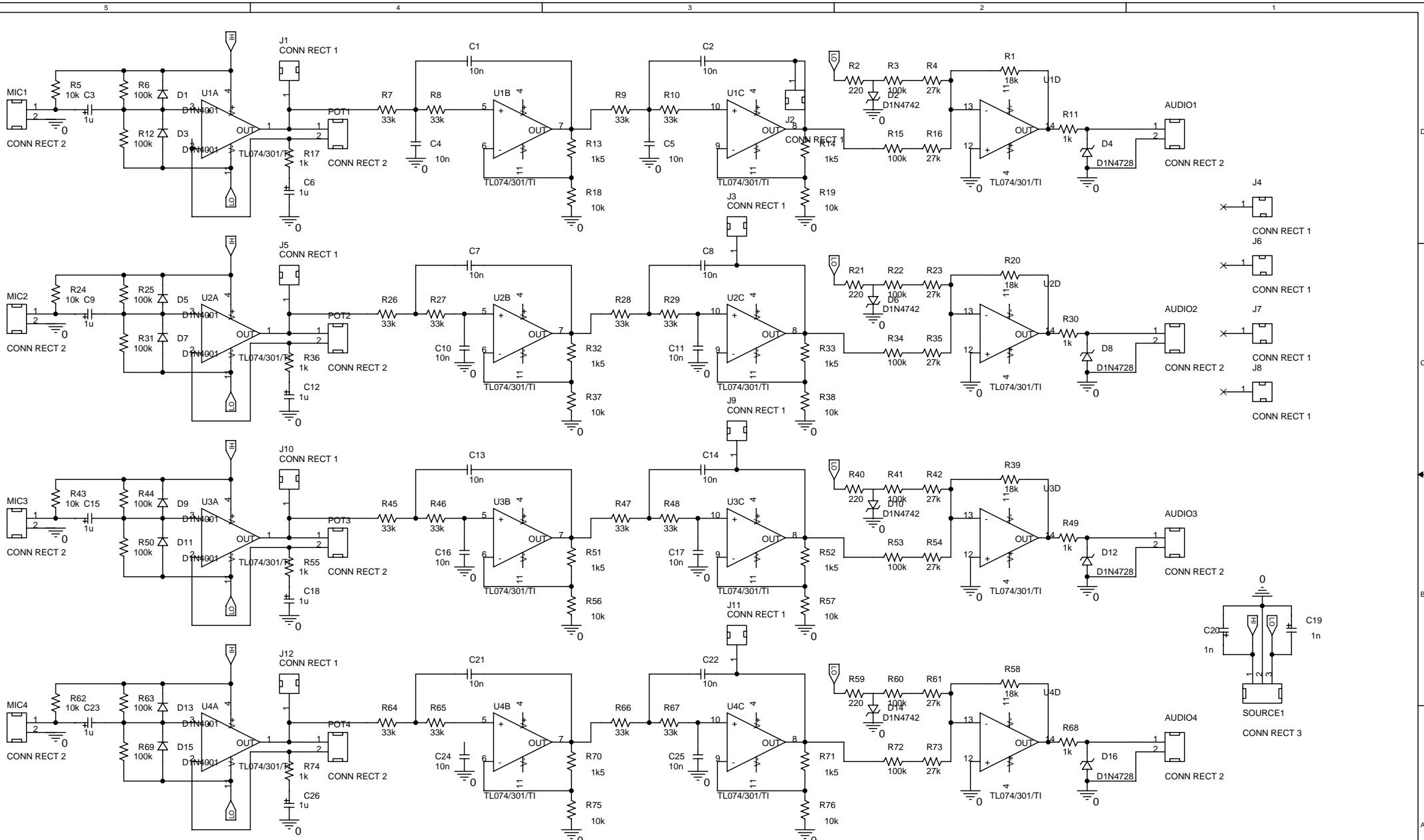


Title	
Condicionador dos Alto-falantes	
Size	Document Number
B	Controle Ativo de Ruído

Date: Friday, February 11, 2005 Sheet 1 of 1 Rev 2







Title	
Condicionador dos Microfones	
Size	Document Number
B	Controle Ativo de Ruído
Date:	Wednesday, February 16, 2005
Rev	3

