

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E COMPUTAÇÃO
DEPARTAMENTO DE COMUNICAÇÕES

Aspectos Relevantes para a Definição de um Sistema de Televisão HDTV

Autor: Guillermo Leopoldo Kemper Vásquez

Orientador: Prof. Dr. Yuzo Iano

Este exemplar corresponde à redação final da tese
fornecida por Guillermo Leopoldo Kemper Vásquez
e aprovada pela Comissão
gadora em 30 / Out / 1996.
Yuzo Iano
Orientador

Dissertação submetida à Faculdade
de Engenharia Elétrica da
Universidade Estadual de Campinas,
como parte dos requisitos para a
obtenção do Título de Mestre e
Engenharia Elétrica.

Outubro 1996

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

K323a

Kemper Vásquez, Guillermo Leopoldo
Aspectos relevantes para a definição de um sistema de
televisão HDTV / Guillermo Leopoldo Kemper
Vásquez.--Campinas, SP: [s.n.], 1996.

Orientador: Yuzo Iano.

Dissertação (mestrado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica e de
Computação.

1. Televisão de alta definição. 2. Processamento de
imagens. 3. Processamento de sinais - Técnicas digitais.
4. Televisão - Transmissores e transmissão. 5.
Codificação. 6. Comunicações digitais. 7. Multiplexação.
I. Iano, Yuzo. II. Universidade Estadual de Campinas.
Faculdade de Engenharia Elétrica e de Computação. III.
Título.

*Dedico este trabalho aos meus queridos pais
Nelly e Sixtilio, a meus irmãos, a toda minha
família e a minha querida Anelli*

AGRADECIMENTOS ESPECIAIS

Ao meu Orientador Professor Dr. *Yuzo Iano* pelas sugestões e apoio para o desenvolvimento deste trabalho, assim como pela confiança e amizade que sempre demonstrou desde o primeiro dia que cheguei ao Brasil.

Ao Dr. *Victor Latorre*, ao Dr. *Cylon Gonçalves da Silva* diretor do *Laboratório Nacional de Luz Síncrotron (LNLS)* e ao Sr. *Guilherme Franco*, pela amizade e ajuda dispensada durante meu processo de admissão junto a pós-graduação da UNICAMP.

A meus queridos pais *Nelly* e *Sixtilio*, a meus irmãos e a toda minha família, por todo seu apoio, carinho e estímulo que nunca faltaram.

AGRADECIMENTOS

Agradeço a **DEUS TODO PODEROSO POR TUDO**

A meu amigo *Fernando Chung Joo* pela ajuda na reunião do material necessário para o desenvolvimento deste trabalho.

Aos meus amigos *Antônio Fançony*, *Luciane Gottardo* e *Paulo Alexandre Galasso do Amaral*, pela ajuda na correção ortográfica deste trabalho.

À *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES), ao *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq) e à *Fundação de Apoio à Pesquisa do Estado de São Paulo* (FAPESP), pela ajuda financeira durante o desenvolvimento do meu mestrado.

Ao *Centro de Pesquisa e Desenvolvimento da Telebrás* (CPqD/Telebrás) pelo fornecimento da informação necessária para este trabalho.

Ao governo da *República Federativa do Brasil* e ao *povo brasileiro* por dar-me a oportunidade de fazer estudos de pós-graduação neste belo país.

À universidade *Antenor Orrego de Trujillo - Perú* pela ajuda e apoio proporcionado à minha admissão na pós-graduação da UNICAMP.

A todos meus amigos que sempre me estimularam e apoiaram.

RESUMO

Neste trabalho realizamos um estudo das principais propostas que atualmente estão definindo os diferentes sub-sistemas de um sistema completo de televisão digital HDTV (**High Definition Television**).

Dessa forma, para o caso do sub-sistema de processamento de vídeo, é apresentado um estudo sobre os aspectos mais relevantes do sistema de compressão **MPEG-2**, assim como os formatos de vídeo que estão sendo considerados para os diferentes padrões.

No sub-sistema de processamento de áudio fazemos a análise do sistema de compressão *Dolby Surround AC-3* que foi considerado na proposta apresentada pela **GA-USA (Grande Aliança dos Estados Unidos)**.

Para o sub-sistema de transporte e multiplexagem, são apresentados os aspectos mais relevantes da recomendação ISO/IEC 13818-1 *MPEG-2 System* que foi também considerado na proposta da Grande Aliança, para o empacotamento e multiplexação dos sinais digitais de televisão, provenientes dos codificadores de áudio e vídeo respectivamente.

Para o caso do sub-sistema de transmissão são apresentados dos sistemas: o **COFDM (Coded Orthogonal Frequency Division Multiplexing)** adotado na Europa e o **8-16 VSB (Vestigial Side Band)** apresentado como proposta pela Grande Aliança. Esses dois sistemas representam as duas alternativas atuais mais importantes para a transmissão de um sinal HDTV. No final do capítulo apresentamos uma comparação entre o desempenho desses dois sistemas.

Finalmente, no último capítulo do trabalho são apresentadas as conclusões do trabalho e as perspectivas para o futuro do que será o sistema de televisão de alta definição a ser implantado em escala global.

ÍNDICE

CAPÍTULO 1 : INTRODUÇÃO GERAL

1.0.- Introdução	1
1.1.- Objetivos do trabalho	3
1.2.- HDTV e os sistemas convencionais	3
1.3.- Descrição do trabalho	5

CAPÍTULO 2 : PROCESSAMENTO DIGITAL DO SINAL DE VÍDEO

2.0.- Introdução	7
2.1.-Características e propostas para uma imagem HDTV	9
2.2.-Pré-Processamento do sinal de televisão HDTV	10
2.2.1.-Amostragem	10
2.2.2.-Quantização	12
2.2.3.-Codificação binária	13
2.2.4.-Componentes de sinais de vídeo	13
2.2.4.1.-Correção Gama	14
2.2.5.- Sinais de sincronismo	15
2.3.- Organização dos dados digitais de uma imagem de televisão HDTV	20
2.3.1.- Pixels	20
2.3.1.1.- Formatos de cor	20
2.3.2.- Blocos de pixels	22
2.3.3.- Macroblocos	23
2.3.4.- Fatias	24
2.3.5.-Seqüência de quadros e tipos de quadros	24
2.3.5.1.- I-Quadros	25
2.3.5.2.- P-Quadros	25

2.3.5.3.- B-Quadros	25
2.3.5.4.- Grupos de quadros	26
2.3.5.5.- Modos de filmes (resolução temporal)	27
2.4.- Técnicas de Predição	28
2.4.1.- Compensação de movimento	28
2.4.1.1.- Vetores de movimento numa vídeo seqüência	29
2.4.2.- Estimação de movimento	30
2.4.2.1.- Método de casamento de blocos (<i>Block Matching</i>)	31
2.4.2.1.1.- Algoritmo de procura logarítmica	32
2.4.2.1.2.- Algoritmo de procura em três passos	33
2.4.2.1.3.- Algoritmo de procura telescópica	34
2.4.3.- Interpolação temporal	35
2.5.- Sistema de codificação MPEG-2	36
2.5.1.- Transformada Discreta do Cosseno de Fourier (DCT)	40
2.5.1.1.- Transformação espacial por blocos	41
2.5.1.2.- Características dos coeficientes DCT	43
2.5.2.- Varredura Zig-Zag	44
2.5.3.- Quantização dos coeficientes DCT	44
2.5.3.1.- Intra-quantizador	44
2.5.3.2.- Não intra-quantizador	45
2.5.3.3.- Critérios para a escolha do número de níveis de quantização	46
2.5.4.- Codificação VLC (códigos de comprimento variável)	47
2.5.4.1.- Codificação de <i>Huffman</i>	48
2.5.4.2.- Codificação <i>Run-Length</i>	49
2.5.5.- Buffer de canal	50
2.6.- Atualização da Imagem	51
2.6.1.- Transmissão periódica dos I-Quadros	51
2.6.2.- Atualização progressiva	51
2.7.- Decodificador MPEG-2	52
2.8.- Comentários e conclusões finais	53

CAPÍTULO 3 : PROCESSAMENTO DIGITAL DO SINAL DE ÁUDIO

3.0.-Introdução	54
3.1.- Sistemas de Compressão de Áudio	56
3.2.- Sistema de Compressão de Áudio “Dolby Surround AC-3”	57
3.2.1.- Canais de áudio	58
3.2.2.- Conversão analógico-digital (PCM)	59
3.2.3.- Formatação da informação de áudio	60
3.2.3.1.- Blocos de Áudio	60
3.2.3.2.- <i>Frame</i> de sincronização	60
3.2.3.2.1.- Campo de sincronização (S.I.)	61
3.2.3.2.2.- Campo de informação (BSI)	61
3.2.3.2.3.- Campos de blocos de áudio codificados	62
3.2.3.2.4.- Campo de informação para a detecção de erros	62
3.2.3.- Análise espectral do sinal de áudio	64
3.2.3.1.- Fatores psico-acústicos	64
3.2.3.2.- Detecção de transitórios	67
3.2.3.3.- Transformação para o domínio da frequência (<i>Analysis Filter Bank</i>) ..	68
3.2.4.- Codificação dos coeficientes MDCT	75
3.2.4.1.- Codificação dos expoentes	76
3.2.4.2.- Processo de alocação de bits	79
3.2.4.3.- Processo de quantização de mantissas	81
3.2.5.- Acoplamento de canais	83
3.2.5.1.- Coordenadas de acoplamento	84
3.2.7.- Normalização do sistema de volume	85
3.2.8.- Compressão da faixa de reprodução de volume	85
3.2.9.- Serviços de áudio oferecido pelo sistema AC-3	88
3.3.- Conclusões e comentários finais	90

CAPÍTULO 4 : SISTEMA DE TRANSPORTE E MULTIPLEXAGEM

4.0.- Introdução	92
4.1.- Pacotes PES	94
4.1.1.- Estrutura de um pacote PES (<i>Packetized Elementary bit Streams</i>)	94
4.1.1.1.- Campo de inicialização e sincronização (<i>Packet Star Code Prefix</i>) ..	94
4.1.1.2.- Campo de identificação (<i>Stream ID</i>)	95
4.1.1.3.- Campo de indicação do comprimento do PES (<i>PES Packet Length</i>) .	96
4.1.1.4.- Campo de indicadores e sinalizadores (<i>PES Header Flags</i>)	96
4.1.1.5.- Campo indicador do comprimento do cabeçalho opcional do PES (<i>PES Header Length</i>)	98
4.1.1.6.- Campos Opcionais (<i>PES Header Fields</i>)	98
4.2.- Pacotes de Transporte	101
4.2.1 Estrutura de um pacote de transporte	101
4.2.1.1.- Campos de cabeçalho (<i>Link Header</i>)	103
4.2.1.2.- Campos de cabeçalho opcionais (<i>Adaptation Field</i>)	105
4.2.1.2.1.- Campo de indicação do comprimento do cabeçalho opcional (<i>Adaptation Field Length</i>)	105
4.2.1.2.2.- Campo de indicadores e sinalizadores	105
4.2.1.2.3.- Campos Opcionais	107
4.3.- Sistema de Multiplexagem	110
4.3.1.- Formação de programas	111
4.3.2.- Formação da seqüência de programas	112
4.4.- Sistema de Demultiplexação	113
4.5.- Comentários e Conclusões Finais	115

CAPÍTULO 5 : SISTEMA DE TRANSMISSÃO

5.0.- Introdução	116
5.1.- Sistema de Modulação OFDM/COFDM	117
5.1.2.- Geração de um sinal OFDM usando a IFFT	118
5.1.3.- Geração de um sinal COFDM	121
5.1.4.- Sistemas OFDM/COFDM propostos na Europa	123
5.1.5.- Vantagens e Desvantagens do sistema COFDM	123
5.2.- Sistema de Modulação 8 VSB - 16 VSB	124
5.2.1.- Embaralhador de dados (<i>Data Randomizer</i>)	126
5.2.2.- Codificação <i>Reed Solomon</i>	127
5.2.3.- Intercalador (<i>Interleaver</i>)	128
5.2.4.- Codificação de treliça	129
5.2.5.- Inserção da informação de sincronismo	132
5.2.5.1.- Sincronismo de Segmento (<i>Data Segment Sync</i>)	133
5.2.5.2.- Sincronismo de Campo (<i>Data Field Sync</i>)	134
5.2.6.- Modulação e inserção de piloto	135
5.3.- COFDM (Prototipo-6MHz) versus 8 VSB	135
5.4.-Conclusões e comentários finais	136

CAPÍTULO 6 : CONCLUSÕES FINAIS

Conclusões finais	138
APÊNDICE A : CODIFICAÇÃO DE <i>HUFFMAN</i>	140
APÊNDICE B	142
REFERÊNCIAS BIBLIOGRÁFICAS	154

Lista de Figuras

Figura 1.2.1. Espectro de um sinal PAL-M/NTSC	4
Figura 1.3.1. Modelo de um sistema de televisão digital de transmissão terrestre (ITU-R)	6
Figura 2.2.1. Quantização uniforme	12
Figura 2.2.2. Processo de digitalização do Sinal de Vídeo	12
Figura 2.2.3. Forma de onda do período de apagamento de linha considerado no sistema americano	17
Figura 2.2.4. Forma de onda de um pulso de sincronismo de campo considerado no sistema americano.....	17
Figura 2.2.5. Forma de onda do sinal de sincronismo de linha considerado no sistema Europeu	18
Figura 2.2.6. Detalhe do sinal de sincronismo de campo considerado no sistema Europeu	19
Figura 2.2.7. Diagrama de bloco do sistema de pré-processamento do sinal de vídeo	19
Figura 2.3.1. Representação de um pixel	20
Figura 2.3.2. Formato de cor 4:2:2	21
Figura 2.3.3. Formato de cor 4:2:0	22
Figura 2.3.4. Relação física entre o número de amostras de luminância e crominância	22
Figura 2.3.5. Representação dos blocos ..	23
Figura 2.3.6. Representação de um macrobloco	23
Figura 2.3.7. Representação de uma seqüência de quadros	26
Figura 2.3.8. Representação de uma vídeo seqüência	26
Figura 2.3.9. Hierarquia da representação digital de uma imagem	27
Figura 2.4.1. Representação de um vetor de movimento	29
Figura 2.4.2. Vetores de movimento numa vídeo seqüência	30
Figura 2.4.3. Algoritmo de procura logarítmica	33

Figura 2.4.4. Algoritmo de procura em três passos	33
Figura 2.4.5. Seqüência de quadros referencial para o algoritmo de procura telescópica	35
Figura 2.4.6. Interpolação temporal	35
Figura 2.5.1. Codificador MPEG-2	38
Figura 2.5.2. Ocupação do Buffer	39
Figura 2.5.3. Quantidade de bits por quadro	39
Figura 2.5.4. Seqüência de codificação das componentes do sinal de vídeo que formam um macrobloco	40
Figura 2.5.5. Representação de um bloco de 8x8 pixels no domíni da transformada 2-DCT	43
Figura 2.5.6. Varredura Zig-Zag	44
Figura 2.5.7. Quantizadores Intra-quadro e não intraquadro.	46
Figura 2.7.1. Decodificador MPEG-2	52
Figura 3.2.1.a. Sistema de Codificação Dolby AC-3	57
Figura 3.2.1.b. Sistema de Decodificação Dolby AC-3	58
Figura 3.2.2. Disposição dos canais de áudio de um sistema Dolby AC-3	59
Figura 3.2.3. Conversão Analógico-Digital dos sinais de áudio	60
Figura 3.2.4. Formato de uma <i>frame</i> de sincronização AC-3	61
Figura 3.2.5. Circuito gerador de códigos de detecção de erros	63
Figura 3.2.6. Percepção do ouvido humano em função da freqüência	65
Figura 3.2.7. Percepção do ouvido humano frente a dois tons que ocorrem ao mesmo tempo.	66
Figura 3.2.8. Percepção do ouvido humano frente a ocorrência de transitórios	67
Figura 3.2.9. Implementação de um sistema <i>analysis/synthesis filter bank</i>	70
Figura 3.2.10. Banco real de filtros passa-faixa	71
Figura 3.2.11. Mecanismo de cancelamento de <i>aliasing</i> no domínio do tempo	72
Figura 3.2.12. Distorção do espectro pelo efeito de bloco	74
Figura 3.2.13. Janela aplicada ao sinal de áudio pelo sistema AC-3	74
Figura 3.2.14. Processo de aplicação da transformada a cada bloco de áudio	75

Figura 3.2.15. Processo de codificação dos coeficientes MDCT	83
Figura 4.0.1 Processo de empacotamento e multiplexagem da informação de áudio e vídeo.	93
Figura 4.1.1. Estrutura de um pacote PES	94
Figura 4.1.2. Disposição dos indicadores PES <i>Header Flags</i>	96
Figura 4.1.3. Disposição dos campos opcionais no PES <i>Header Fields</i>	98
Figura 4.1.4. Organização do PES <i>Field</i>	99
Figura 4.1.5. Organização do PTS <i>Field</i> quando ambos valores PTS e DTS são codificados	100
Figura 4.1.6. Organização do PES <i>extension flags</i>	100
Figura 4.2.1. Formato de um pacote de transporte	101
Figura 4.2.2. Comparação entre a estrutura de uma célula ATM e um pacote de transporte.	102
Figura 4.2.3. Formato do <i>Link Header</i> de um pacote de transporte	103
Figura 4.2.4. Formato de <i>adaptation field</i> de um pacote de transporte	106
Figura 4.2.5. Sistema de recepção e decodificação dos pacotes de transporte ...	108
Figura 4.2.6. Circuito PLL para ajuste da frequência do relógio local	109
Figura 4.2.7. Arquitetura de um sistema de inserção de programas	110
Figura 4.3.1. Sistema de multiplexação para a formação de programas	111
Figura 4.3.2. Sistema de multiplexação de programas	113
Figura 4.3.3. Diagrama de blocos do sistema de demultiplexagem	114
Figura 4.3.4. Diagrama de blocos da seqüência de demultiplexação da informação.	114
Figura 5.1.1. Geração do espectro de um sinal OFDM com três portadoras ...	118
Figura 5.1.2. Diagrama de blocos do sistema gerador de um sinal OFDM usando IFFT/FFT.	120
Figura 5.1.3. Forma de onda de um sinal OFDM básica	121
Figura 5.1.4. Codificador de treliça para 8-PSK	122
Figura 5.1.5. Treliça para o codificador 8-PSK	122
Figura 5.1.6. Diagrama de blocos do sistema de geração de um sinal COFDM	122
Figura 5.2.1. Diagrama de blocos do sistema 8-VSB	125

Figura 5.2.2. <i>Data Frame</i> VSB	126
Figura 5.2.3. Circuito gerador polinomial de bits pseudo-aleatórios	127
Figura 5.2.4. Codificador <i>Reed Solomon</i> (207,187)	128
Figura 5.2.5. Intercalador (I=52) do sistema VSB	129
Figura 5.2.6. Codificador de treliça do sistema VSB	130
Figura 5.2.7. Multiplexador dos codificadores de treliça do sistema VSB	131
Figura 5.2.8. Mapeamento dos bits num sistema 16-VSB	132
Figura 5.2.9. Sistema 16-VSB	132
Figura 5.2.10. Formato do segmento de dados (Data Segment) para o sistema 8 VSB	134
Figura 5.2.11. Formato do segmento de sincronismo de campo para 8 VSB	134
Figura 5.2.12. Espectro do sinal VSB	135
Figura A.1. Processo de codificação <i>Huffman</i>	141

Lista de Tabelas

Tabela 2.1.1. Formatos de varredura	9
Tabela 2.1.2. Coordenadas de cromaticidade para o sistema HDTV	10
Tabela 2.2.1. Formatos de amostragem	11
Tabela 2.2.2. Formato dos níveis de tensão designados para as componentes de cor e os sinais de sincronismo no sistema americano	16
Tabela 2.2.3. Características gerais dos sinais de sincronismo e os tempos de eventos numa linha de vídeo, considerados no sistema americano	16
Tabela 2.2.4. Tempo de duração dos sinais de vídeo e sincronismo considerados no sistema americano	16
Tabela 2.2.5. Formato dos níveis de tensão designados para as componentes de cor e para os sinais de sincronismo no sistema Europeu	17
Tabela 2.2.6. Características do sinal de sincronismo considerado no sistema Europeu.	18
Tabela 2.2.7. Sinais de sincronismo de quadro do sistema Europeu	18

Tabela 2.4.1. Vetores de movimento para um B-quadro	36
Tabela 2.5.1. Exemplo de codificação <i>Huffman</i> para os pares (<i>Run- Amplitude</i>)	50
Tabela 3.1.1. Características dos sistemas de compressão de áudio	56
Tabela 3.2.1. Canais de áudio do codificador AC-3	58
Tabela 4.1.1. Mapeamento do campo de identificação <i>stream ID</i>	95
Tabela 4.1.2. Mapeamento do indicador de acesso condicional SC	97
Tabela 4.1.3. Mapeamento do indicador PD	97
Tabela 4.2.1. Mapeamento do indicador de modo de acesso	104
Tabela 4.2.2. Mapeamento do indicador <i>adaptation field control</i>	105
Tabela 4.3.1. Definição dos valores dos PID's para diferentes <i>bit streams</i>	112
Tabela 4.3.2. Exemplo de PID's designados ao programa 52	112
Tabela 5.3.1. Performance dos sistemas COFDM e 8 VSB num canal de 6MHz.	136
Tabela B.1. Mapeamento do <i>fscod</i>	142
Tabela B.2. Mapeamento do <i>frmsizecod</i>	142
Tabela B.3. Mapeamento do <i>bsmod</i>	143
Tabela B.4. Mapeamento do <i>acmod</i>	143
Tabela B.5. Tamanho em palavras (2-bytes) das primeiras 5/8 partes de uma <i>frame</i>	144
Tabela B.6. Valores da janela utilizada pelo sistema AC-3	145
Tabela B.7. Mapeamento dos expoentes diferenciais mediante a estratégia D15.	145
Tabela B.8. Mapeamento dos expoentes diferenciais mediante a estratégia D25.	145
Tabela B.9. Mapeamento dos expoentes diferenciais mediante a estratégia D45.	146
Tabela B.10. Codificação das estratégias para expoentes.	146
Tabela B.11. Codificação das estratégias para expoentes do canal LFE.	146
Tabela B.12. Mapeamento das bandas de frequências utilizadas no processo de quantização de mantissas.	147
Tabela B.13. Mapeamento do limiar de ruído permitido pelo ouvido humano para cada banda de frequências da tabela B.12	148

Tabela B.14. Mapeamento do ponteiro bap e os possíveis resultados da rotina de alocação de bits (endereço).	149
Tabela B.15. Mapeamento dos ponteiros bap e os níveis de quantização.	150
Tabela B.16. Mapeamento dos ponteiros bap e os tipos de quantização.	150
Tabela B.17. Mapeamento dos valores das mantissas (quantização simétrica) para bap=1 (3 níveis)	151
Tabela B.18. Mapeamento dos valores das mantissas (quantização simétrica) para bap=2 (3 níveis)	151
Tabela B.19. Mapeamento dos valores das mantissas (quantização simétrica) para bap=3 (7 níveis)	151
Tabela B.20. Mapeamento dos valores das mantissas (quantização simétrica) para bap=4 (11 níveis)	152
Tabela B.21. Mapeamento dos valores das mantissas (quantização simétrica) para bap=5 (15 níveis)	152
Tabela B.22. Sub-bandas de acoplamento	153
Tabela B.23. Mapeamento do sub-campo mastreplco	153
Tabela B.24. Mapeamento dos 3 bits MSB do sub-campo dynrgn.	153

Capítulo 1

Introdução Geral

1.0.- Introdução

Durante os últimos anos, os estudos sobre sistemas de HDTV (*High Definition Television*) totalmente digitais, têm aumentado notavelmente. Os diferentes grupos de pesquisa formados na Europa, nos Estados Unidos, no Japão e no Canadá têm realizado testes sobre diversos formatos cujos resultados estão sendo apresentados como propostas através de diversos documentos técnicos, normas/padrões e recomendações.

A maioria dessas propostas estão considerando principalmente uma imagem que tenha o dobro do número de linhas de varredura que a dos formatos convencionais de televisão; assim como a reprodução de um maior número de cores.

Essas melhorias originam portanto uma imagem de alta definição cuja qualidade é comparada a uma imagem típica de 35mm de cinema.

Por outro lado, junto com o sinal de vídeo deve-se considerar também a transmissão de um sinal digital de áudio através de vários canais que em conjunto reproduzem um som de alta fidelidade.

O preço a pagar por todos esses benefícios é um incremento de pelo menos 5 a 10 vezes na largura de faixa do canal de transmissão atualmente utilizado pelos sistemas convencionais (6MHz no Brasil, USA, Japão, Canadá e outros países).

As primeiras pesquisas sobre sistemas HDTV com as características anteriores, começaram no Japão no início dos anos 70. Nessa época foi desenvolvido um sistema baseado no formato de 1125 linhas de varredura, com uma frequência de campo de 60 Hz. O resultado dessas pesquisas foi o sistema **MUSE** (*Multiple Sub-Nyquist Encoding*) proposto pela NHK (*Nippon Hoso Kyokai*) onde o sinal de transmissão era codificado em forma analógica.

A partir de então foram aparecendo diversas propostas para HDTV principalmente na Europa, onde em todos os casos o tratamento do sinal de transmissão não era completamente digital. A tecnologia para HDTV mostrava-se

pronta, no entanto as tentativas de padronização universal geram uma série de impasses motivados por interesses econômicos e políticos.

Posteriormente, com o passo dos anos e com a evolução da tecnologia de processadores digitais, nasceu a bemvinda idéia de se desenvolver um sistema HDTV totalmente digital. Nesse caso, pode-se aproveitar as vantagens oferecidas pelos sinais digitais sobre os sinais analógicos.

Dessa forma, nos Estados Unidos foi criado o **ATSC** (*Advanced Television Systems Committee*) encarregado de coordenar o desenvolvimento da documentação necessária para a definição do sistema de televisão avançada **ATV- Advanced Television**. Esses documentos são constituídos por recomendações, guias técnicos, e padrões de diversos organismos e companhias com atuação nos EE.UU (Estados Unidos da América do Norte).

No ano 1987 o **FCC** (*Federal Communications Commission*) formou o **ACATS** (*Advisory Committee on Advanced Television Service*) a fim de organizar os estudos sobre os aspectos e os parâmetros técnicos para o estabelecimento da **ATV standard** nos Estados Unidos. Note-se que esta não é uma aliança econômica, mas política e técnica. A aliança somou os esforços de desenvolvimento tecnológico das companhias envolvidas, evitou a perda de pesados investimentos caso um único sistema de determinada companhia fosse padronizado e gerou um compromisso de trabalho conjunto até a padronização de um sistema totalmente digital. O ATSC deveria estabelecer também os sistemas de avaliação das diversas propostas apresentadas pelos distintos grupos de pesquisa.

A partir desse momento foram testadas mais de 20 propostas, sendo selecionados 4 sistemas apresentados por três grupos de pesquisa.

Os três grupos selecionados decidiram-se posteriormente se ajuntar para formar a aliança "**Digital HDTV Grand Alliance**" (G.A), a fim de gerar um sistema de televisão avançada de uma alta performance que constitua o padrão *standard* para os Estados Unidos. Dessa forma, a aliança ficou constituída pelas seguintes instituições:

- **AT&T and Zenith Electronics Corporation**
- **General Instrument Corporation and the Massachusetts Institute of Technology**
- **Philips Consumer Electronics, Thomson Consumer Electronics and David Sarnoff Research Center.**

Após uma série de pesquisas e testes, a Grande Aliança apresentou ao ACATS os resultados finais da proposta para que fosse feita a respectiva avaliação do sistema. Finalmente nos anos 1994 e 1995 foi emitida a documentação dos parâmetros técnicos do sistema HDTV G.A. proposto para os Estados Unidos.

O sistema está baseado nos formatos 750/60 e 1125/60, que indicam 750 e 1125 linhas de varredura para uma frequência de campo de 60 Hz.

Por outro lado, a G.A. estabeleceu a transmissão de um sinal HDTV dentro da largura de faixa de **6MHz** utilizada pelos sistemas convencionais de televisão. Isso foi estabelecido devido a vantagem de se usar o mesmo canal terrestre para a transmissão do sinal de televisão convencional sem nenhuma demanda adicional de faixa em relação a CTV (*Conventional Television*)

A G.A. propôs dessa forma o sistema “*simulcast*”, mediante o qual um mesmo programa é transmitido através do sistema NTSC convencional e também através do sistema HDTV no canal adjacente de 6 MHz. Há portanto uma compatibilidade de programação, mas não existe a compatibilidade direta e inversa a nível de transmissores e receptores.

Atualmente existe uma alta possibilidade de que o sistema proposto pela G.A. seja o padrão do sistema HDTV a se aplicar nos Estados Unidos. Grande parte deste trabalho é baseado nessa proposta cujos parâmetros técnicos foram tomados dos documentos do ATSC.

O Japão está a espera do sistema a ser implantado nos Estados Unidos. Atualmente está desenvolvendo pesquisas com respeito ao sistema de transmissão COFDM-6 Mhz , totalmente digital também.

Na Europa é definitiva a implantação do sistema COFDM-8MHz para a transmissão do sinal HDTV com o formato 1250/50 (1250 linhas de varredura, frequência de campo de 50 Hz). Não entanto, continua-se a espera de definição dos sistemas de compressão e transporte a serem implantados nesse continente.

1.1.- Objetivos do Trabalho

Este trabalho tem por objetivo, o estudo dos formatos e das técnicas de compressão, codificação e transmissão, que atualmente estão em fase de adoção e padronização pelos diferentes organismos para a definição de um sistema HDTV, principalmente nos Estados Unidos (Grande Aliança-6 MHz) e na Europa (8 MHz).

Além disso, nosso objetivo é também fornecer os aspectos mais relevantes das características dos diferentes processos de codificação, processamento de sinais, e modulação, a fim de justificar as motivações que levaram sua utilização nos diferentes sistemas propostos.

1.2.- HDTV e os Sistemas Convencionais

Atualmente, a televisão mundial está baseada nos distintos sistemas analógicos convencionais : NTSC, PAL e SECAM para definição de cor e A,B,C,E,G,H,I,K,K1,L,M e N para definição dos padrões em P&B (Preto e Branco). Cada um desses sistemas enfrenta de uma maneira distinta os mesmos inconvenientes

que, de uma forma ou outra limitam a qualidade da imagem apresentada na tela. Essas desvantagens são percebidas com maior força com a aparição das técnicas de tratamento digital de sinais bem como pelo alto desempenho alcançado por sistemas que utilizam sinais digitais na transmissão da informação.

De acordo com essas considerações podemos dizer que o sinal de televisão convencional apresenta 4 problemas principais [8]:

- Cruzamento de Luminância.
- Cruzamento de Cor.
- Susceptibilidade aos erros de transmissão.
- Ineficiência na utilização do canal.

Os dois primeiros inconvenientes resultam do fato de que a transmissão das componentes de cor e de luminância é realizada dentro de uma mesma faixa, ocupando o mesmo espaço em frequência porém com intercalamento a fim de se diminuir os efeitos do ruído de baixa visibilidade. Na Figura 1.2.1 é mostrado o espectro de um sinal PAL-M/NTSC, onde se observa esses problemas.

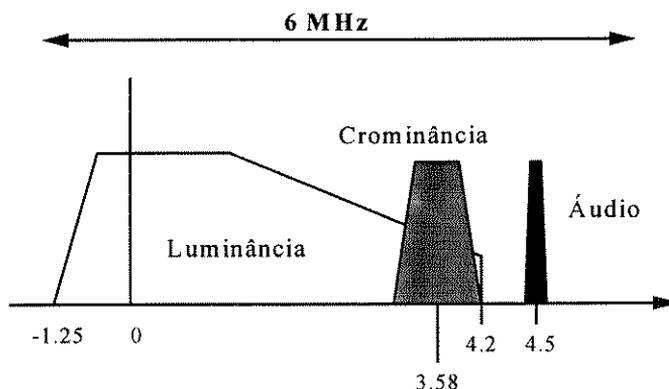


Figura 1.2.1 Espectro de um Sinal PAL-M/NTSC

No caso do sistema de televisão digital HDTV, tanto a componente de luminância como as componentes de cor (2 sinais diferença de cor) são transmitidas de forma separada através de um adequado sistema de codificação.

O outro problema apresentado é com respeito a transmissão do sinal de televisão susceptível aos erros sem contar com as ferramentas necessárias para que seja feita a respectiva correção.

A solução para esse problema é tradicional e amplamente conhecido no mundo digital. O método é contemplado dentro do sistema HDTV digital mediante a transmissão da informação com códigos de correção de erros que adicionam bits de proteção para posterior detecção e correção de erros. Dessa forma o receptor pode recuperar a informação original que foi afetada pelo ruído do canal de transmissão.

Finalmente, dadas as taxas de compressão hoje alcançadas pelos diferentes sistemas de codificação (amplamente utilizados nos canais de transmissão digitais), podemos perceber um ineficiente uso do canal de transmissão por parte dos sistemas analógicos convencionais.

A digitalização dos sinais de vídeo RGB dos sistemas PAL-M/NTSC por exemplo, implicam em uma taxa de bits de aproximadamente **216 Mbit/s** (frequência de amostragem de 13,5 MHz para luminância, 6,75 MHz para crominância e 8 bits por amostra). Aplicando o sistema de compressão MPEG-2 estudado no capítulo 2 pode-se alcançar um nível de compressão de vídeo de 50:1, o que geraria uma taxa de bits de aproximadamente 3 Mbit/s para a transmissão de sinais televisão convencional em formato digital. Dessa forma dentro do canal de **6 MHz** utilizado nesses sistemas e aplicando um adequado sistema de modulação digital pode-se transmitir ao mesmo tempo vários programas de televisão sem afetar a qualidade da imagem subjetiva pré-estimada para esses sistemas. Se isso fosse praticado, resultaria em um eficiente uso do canal de transmissão por parte dos sistemas convencionais.

O sistema digital HDTV faz um eficiente uso do canal desde que dentro da mesma largura de faixa utilizada nos sistemas convencionais, é transmitido um sinal que leva uma imagem com o dobro da resolução espacial da imagem apresentada nesses sistemas e quase 8 vezes a resolução horizontal da CTV.

Outras vantagens que apresenta um sistema digital, são a flexibilidade em termo de qualidade/taxa de bits, compatibilidade com computadores e redes de computadores, uso das técnicas de modulação digital e serviços tipo multimeios.

1.3.- Descrição do Trabalho

O desenvolvimento de um sistema de televisão totalmente digital implica na formação de vários sub-sistemas que cumprem uma função específica (geração, transmissão, armazenamento e recepção) para a formatação do sinal que será transmitido. Dessa forma, divide-se um sistema digital HDTV em 5 sub-sistemas que são os seguintes:

- Sub-sistema de Vídeo : Compressão e Codificação
- Sub-sistema de Áudio : Compressão e Codificação
- Sub-sistema de Transporte e Multiplexação
- Sub-sistema de Transmissão
- Sub-sistema de Recepção

Na figura 1.3.1 é mostrado o diagrama de bloco de todo o sistema digital HDTV.

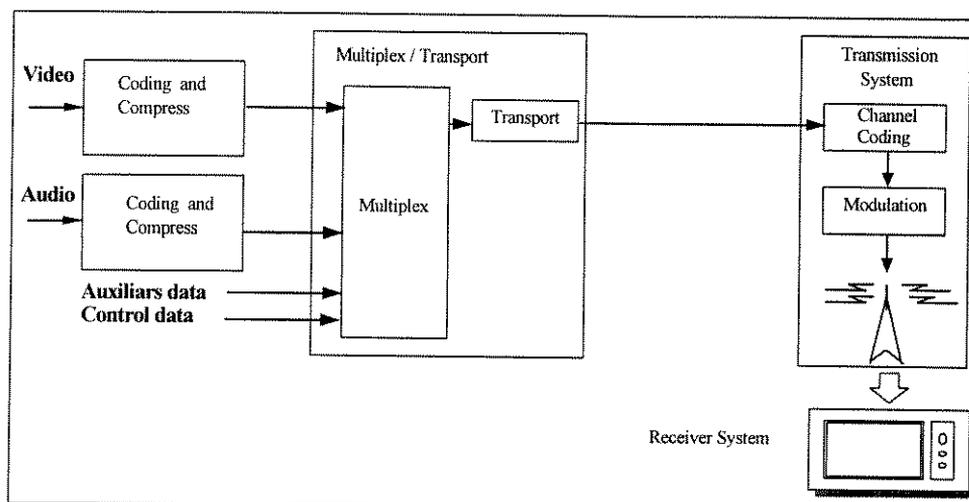


Figura 1.3.1 Modelo de um sistema de televisão digital de transmissão Terrestre (ITU-R)

Neste trabalho, estamos abordando as propostas atuais para cada um desses sub-sistemas até a parte de transmissão dentro da filosofia técnica da GA-USA, principalmente. O sub-sistema de recepção está sendo ainda definido pela GA, no entanto grande parte do mesmo esta formada por os blocos de demodulação, demultiplexação e decodificação que constituem os processos inversos de cada um dos sub-sistemas estudados nos capítulos deste trabalho.

No capítulo 2, apresentamos inicialmente as propostas para os formatos de vídeo adotados nos Estados Unidos e na Europa. A seguir, é feito a análise dos aspectos mais relevantes do sistema de compressão MPEG-2 adotado no sistema da G.A. e através do qual, pode-se chegar até um nível de compressão de 50:1. As técnicas de compressão utilizadas pelo MPEG-2 permitem que a degradação sofrida pelo sinal não seja percebida pelo nosso sistema humano de visão.

No capítulo 3, é realizado a análise do sistema de compressão de áudio “*Dolby Surround AC-3*”, que foi adotado no sistema da G.A para o tratamento digital do sinal de som. A avaliação do sistema Dolby AC-3 foi baseada na taxa de compressão alcançada pelo mesmo. Outro fator importante para a escolha do Dolby AC-3 é a alta fidelidade de som reproduzida pelos canais de áudio desse sistema.

No capítulo 4, é apresentado o sistema de transporte e multiplexação baseado na recomendação **ISO/IEC 13818-1** que propôs o sistema MPEG-2 *System* para o empacotamento, transporte e multiplexação da seqüência de bits provenientes dos codificadores de áudio e vídeo respectivamente. Dada a flexibilidade do sistema MPEG-2 *system* para aplicações em HDTV, isso foi considerado na proposta apresentada pela G.A.

No capítulo 5 finalmente é feito a análise de dois sistemas de transmissão: o COFDM adotado na Europa e o 8-16 VSB proposto pela G.A. Nesse sentido fazemos um estudo desses sistemas para finalmente apresentar uma comparação entre o desempenho dos mesmos.

Capítulo 2

Processamento Digital do Sinal de Vídeo

2.0.- Introdução

Um incremento de melhoria na definição de uma imagem de televisão implica em um aumento do número de linhas de varredura bem como uma maior variação temporal do sinal de vídeo. Como consequência disso, o sinal resultante é caracterizado por um incremento de sua largura de faixa uma vez que nesse caso é transmitida uma maior quantidade de informação por unidade de tempo.

Dessa forma para um sinal de televisão de alta definição como é o caso do sistema HDTV espera-se portanto uma largura de faixa muito maior do que a do sistema de televisão analógico convencional. No caso do sistema americano padronizou-se uma largura de **30 MHz** (luminância) e no caso do sistema europeu uma largura de faixa de **60 MHz** (luminância).

A preocupação com a compatibilidade entre o sistema de HDTV digital e o sistema convencional levou à proposta do sistema de transmissão SIMULCAST. Nessa proposta não existe compatibilidade técnica, mas apenas de programação. Assim, um sinal de alta definição passa a utilizar canais com largura de 6 MHz conforme já utilizados pelos sistemas convencionais **NTSC** e **PAL-M**. O sinal HDTV será transmitido através dessa largura de faixa, segundo a proposta do sistema americano da **Grande Aliança** dos Estados Unidos da América do Norte (USA). Assim, para assegurar a compatibilidade de largura de espectro entre esses sistemas, foi considerado pelo ATSC dos Estados Unidos a utilização de canais de **6 MHz** para a transmissão de um sinal HDTV, enquanto que na Europa foram designados canais de **8.0 MHz** para a transmissão dos sinais de televisão de alta definição

Como consequência disso o objetivo principal do processamento digital do sinal de vídeo para um sinal **HDTV** compreende a obtenção de um sinal de televisão de alta qualidade que possa ser transmitido dentro dos requisitos e limitações do canal de transmissão. A digitalização dos sinais de vídeo num sistema **HDTV** demanda taxas de transmissão de aproximadamente **1Gbps**, o que seria impossível se transmitir dentro de uma largura de faixa de 6 MHz. A solução para esse problema é a utilização de técnicas de **compressão** de faixa usando-se **codificação** do sinal original e **modulação** do sinal de linha a fim de se diminuir a taxa final de bits bem como a largura de faixa do sinal a ser transmitido, tomando-se cuidados para conservar os padrões de qualidade exigidos pelo sistema HDTV.

Um sistema de compressão que permite a obtenção de um sinal com as características citadas, é o sistema **MPEG-2** (*Motion Picture Expert Group*) que faz a compressão do sinal digital numa relação de aproximadamente **55:1** de tal forma que a taxa de bits é reduzida para pelo menos **20Mbs**. O **MPEG-2** explora a alta redundância do sinal de vídeo produzida pela alta correlação temporal e espacial que caracterizam os sinais de televisão.

O sistema **MPEG-2** é organizado dentro de um sistema de *profiles* e níveis que determinam os formatos e níveis de compressão alcançados pelo sistema para cada aplicação.

Os *profiles* (*simple, main, SNR, spatial, high*) determinam a sintaxes e os diferentes níveis de funcionalidade do sistema, enquanto que os níveis (*high, high-1440, main, low*) determinam os aspectos relacionados com a resolução da imagem, a taxa de bits, etc.

Para aplicações em HDTV considera-se o *Main Profile* e o *High Level (MP@HL)*. O *main profile* especifica tipos de quadros de predição utilizados pelo sistema como os I-quadros, P-quadros e os B-quadros. Além disso especifica também que as amostras das componentes de luminância e crominância são organizadas espacialmente usando o formato **4:2:0**, a fim de diminuir a taxa de transmissão aproveitando que a nossa visão não percebe as variações rápidas dos sinais de cor.

Por outro lado, o *High-Level* (nível alto) inclui os formatos de varredura de 1152 linhas ativas com 1920 amostras por linha ativa e uma resolução temporal de 60 quadros por segundo. O *high level* para o *main profile* é limitado a uma taxa de compressão não maior de 80 Mb/s.

Neste capítulo, vamos fazer um análise do processo de compressão baseado no formato **MPEG-2** (*main profile - high level*) que constitui uma das alternativas mais importantes na padronização do processamento digital de vídeo para aplicações em televisão de alta definição. O **MPEG-2** foi proposto pela Grande Aliança para o sistema HDTV americano e também está sendo considerado nos sistemas que estão sendo desenvolvidos na Europa.

No início do capítulo começamos apresentando algumas das propostas atuais referentes à padronização do formato de vídeo e em seguida entramos na análise da digitalização e codificação do sinal que será transmitido.

2.1.- Características e propostas para uma imagem HDTV

Atualmente, diversas propostas que foram apresentadas estão definindo as características do sistema de televisão HDTV. As resoluções poderão ser aplicadas num determinado país ou continente. Nos Estados Unidos foram propostos dois formatos de produção conforme a **SMPTE 274M** (1994) e **SMPTE S17.392** (1995) com **750** e **1125** linhas de varredura respectivamente. Na Europa, a partir do formato **1250/50** proposto ao CCIR no ano 1987 foram desenvolvidos distintos sistemas até chegar ao padrão que é apresentado neste trabalho.

- **Resolução Vertical**

Define o número de linhas de varredura do sistema e representa a quantidade de informação contida na tela. As propostas apresentadas estão dadas na Tabela 2.1.1.

<i>Característica</i>	<i>U.S.A</i>	<i>EUROPA</i>
Número de Linhas de Varredura (progressivo (1:1), entrelaçado (2:1))	750 (1:1) 1125 (1:1) 1125 (2:1)	1250 (1:1)
Número de Linhas Ativas por Quadro	720 (750) 1080 (1125)	1152 (1250)
Taxa de repetição de linha (KHz)	33,75 (2:1)	62,5 (1:1)
Taxa de Quadro/Campo (Hz)	60/59.94 30/29.97 24/23.98	50

Tabela 2.1.1 Formatos de varredura

- **Resolução Horizontal**

Define a qualidade dos detalhes reproduzidos na tela e está representada pela variação do sinal de vídeo no domínio do tempo. Dessa forma quanto maior for a variação temporal do sinal (maior largura de faixa) mais finos serão os detalhes da imagem. Nos sistemas atuais de televisão usam-se as larguras de faixas de 4.2, 5.0 e 6.0 MHz.. enquanto para o sistema HDTV são consideradas larguras de **30** e **60 Mhz** para os sistemas americano e europeu respectivamente.

- **Relação de Aspecto (RA)**

Define a relação largura/altura da tela tanto da câmera como do receptor de televisão. Nos sistemas atuais a relação de aspecto é de 4/3 enquanto que para o sistema HDTV decidiu-se que esta deveria ficar entre 5:3 e 2:1 , sendo adotado o formato **16:9** ou **1,78:1** sugerido pelos EUA uma vez que essa relação encontra-se

perto dos padrões do cinema em *Vistavision* e *Panavision* onde o formato é de 1,85:1, assim como dos formatos europeus de 1,67:1 e 1,75:1. Pelo outro lado, o formato 16:9 constitui uma média geométrica entre o formato 4/3 da televisão convencional e o formato 2,35:1 do *Cinemascope/Panavision*, permitindo centralizar a ação na menor área comum entre os diversos formatos sem perda de detalhes [19].

- **Colorimetria**

Define a qualidade da reprodução de cores na TV. A base consiste em três cores primárias linearmente independentes: vermelho (**R-Red**), verde (**G-Green**) e azul (**B-Blue**). Essas cores são definidas pelos parâmetros “x” e “y” do diagrama de cromaticidade **CIE (Comission Internationale de l’Éclairage)**.

Na Tabela 2.1.2 mostram-se as coordenadas de cromaticidade do sistema CIE para as componentes primárias R, G e B bem como as coordenadas de referência do branco para o iluminante D_{65} usado no sistema HDTV. Os valores foram escolhidos de tal forma que o sistema HDTV concentre uma maior quantidade de cores do que os sistemas de televisão atuais.

	USA/JAPÃO		EUROPA	
	x	y	x	y
R	0.6300	0.3400	0.6915	0.3083
G	0.3100	0.5950	0.0000	1.0000
B	0.1550	0.0700	0.1440	0.0297
W	0.3127	0.3290	0.3127	0.3290

Tabela 2.1.2. Coordenadas de cromaticidade para o sistema HDTV

2.2.- Pré-Processamento do Sinal de Televisão HDTV

O pré-processamento do sinal de televisão compreende a digitalização do sinal analógico de vídeo RGB e a formação das componentes de luminância e de diferença de cor de acordo com a proposta do sistema utilizado. A digitalização do sinal de vídeo no sistema HDTV faz-se mediante o processo de conversão analógico-digital baseada em fundamentos aplicados ao **PCM (Pulse Code Modulation)** que compreende as etapas que descrevemos em seguida.

2.2.1 Amostragem

O processo de amostragem consiste em se tomar amostras do sinal analógico de vídeo formado pelas três **componentes primárias RGB** que correspondem as cores **vermelho, verde e azul**. Cada componente primaria é amostrada em intervalos de tempo denominados períodos de amostragem. Geralmente a amostragem de um sinal é caracterizada pelo inverso do período chamado de **freqüência de amostragem**.

O valor da frequência de amostragem deve ser dimensionado de acordo com o teorema de **amostragem (valor de Nyquist)**, o que implica que a mesma deve ser pelo menos duas vezes maior do que a máxima frequência do sinal que está sendo amostrado. No caso da digitalização da televisão convencional a frequência de amostragem padronizada é de **13.5 MHz**, o que é razoável já que o sinal de vídeo é transmitida dentro de uma faixa máxima de 6 MHz (4,2 MHz para NTSC e PAL-M). Para os sistemas HDTV propostos, os formatos de amostragem são mostrados na Tabela 2.2.1. No caso do sistema americano da **Grande Aliança** a frequência de amostragem é gerada a partir do oscilador padrão de **27 MHz** [2] que é utilizado para sincronizar os codificadores de áudio e vídeo com o sistema de transporte.

Característica	U.S.A	Europa
Largura de Faixa Sinal de luminância Sinal de crominância	30 MHz 15 MHz	60 MHz 30 MHz
Frequência de amostragem Sinal de luminância : Sinal de crominância :	74,25 Mhz 37,125 MHz	144 MHz 72 Mhz
Número de amostras ativas por linha: Sinal de luminância : Sinal diferença de cor :	1280 (720 linhas ativas) 1920 (1080 linhas ativas) 640 (720 linhas ativas) 960 (1080 linhas ativas)	1920 (1152 linhas ativas) 960 (1152 linhas ativas)
Número total de amostras por linha: Sinal de luminância : Sinal diferença de cor:	1650 (720 linhas ativas) 2200 (1080 linhas ativas) 825 (720 linhas ativas) 1100 (1080 linhas ativas)	2304 (1152 linhas ativas) 1152 (1152 linhas ativas)
Nomenclatura das componentes de vídeo codificadas para HDTV (PCM 8 bit /amostra)	$E'_Y = Y$ $E'_Y - E'_R = C_R$ $E'_Y - E'_B = C_B$	$E'_Y = Y$ $E'_{c1} = C_1$ $E'_{c2} = C_2$
Taxa de bits (sem compressão)	1,18 Gbps	2,3 Gbps

Tabela 2.2.1 Formatos de amostragem

2.2.2.- Quantização

A quantização é um processo que consiste em se aproximar os níveis de amplitude do sinal amostrado para valores fixos os mais próximos dos mesmos e denominados níveis de quantização. Um quantizador é caracterizado pelo número de níveis de quantização aos quais se aproxima o sinal original. Quanto maior for o número de níveis de quantização, maior será a resolução do sinal amostrado (mais bits por amostra). Para o caso dos sistemas HDTV tem sido considerado a quantização uniforme com **256 níveis (8 bits por amostra)**.

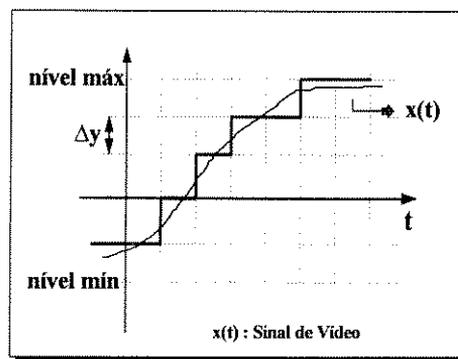


Figura. 2.2.1 Quantização Uniforme

Na Figura 2.2.2 mostra-se o processo de digitalização do sinal de vídeo.

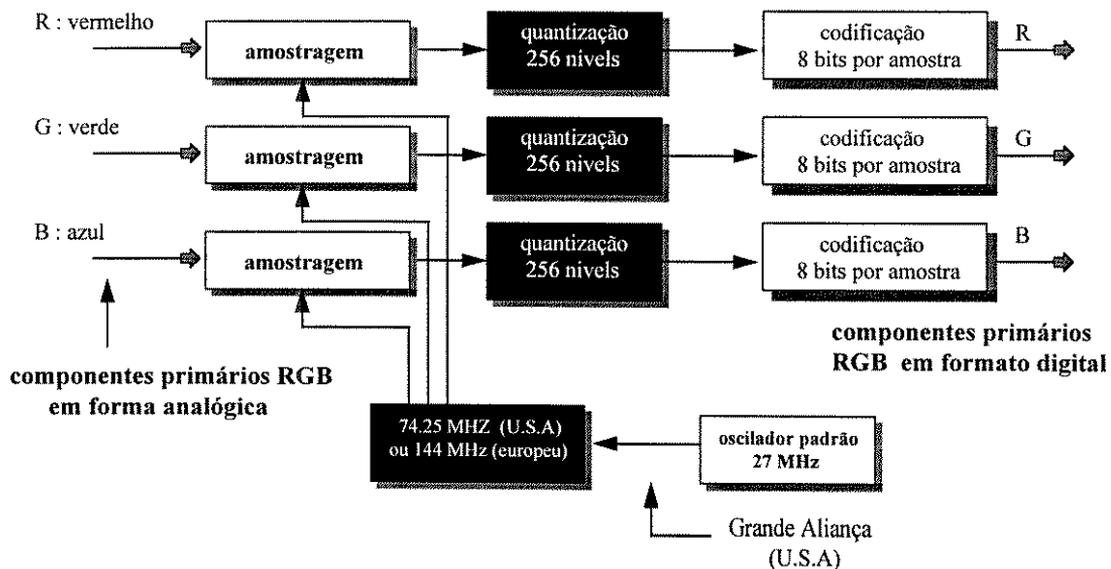


Figura. 2.2.2 Processo de Digitalização do Sinal de Vídeo

2.2.3.- Codificação Binária

Cada nível de quantização é codificado por um código digital de **8 bits**. Dessa forma, tem-se 256 combinações de bits para representar cada nível de quantização considerado na seção anterior.

2.2.4.- Componentes de Sinais de Vídeo

A partir dos três sinais digitais RGB (E_R , E_G , E_B) o sistema faz uma combinação linear dos mesmos a fim de se gerar três novos sinais de vídeo denominados componentes de **luminância** e de **crominância (2 sinais diferença de cor)**. Os objetivos de se gerar as componentes de luminância e de crominância são os seguintes:

- Compatibilizar o sinal de televisão em cores com o sinal de televisão em branco e preto (denominadas compatibilidade direta e reversa).
- Aproveitar melhor as características de percepção do olho humano mediante a diminuição da largura de faixa dos sinais de crominância.
- Melhorar a eficiência da codificação devido ao fato de que as novas componentes eliminam a alta correlação que caracterizam os sinais originais RGB.

Cada uma das componentes de vídeo tem características diferentes que se descrevem a seguir :

• **Componente de Luminância**

A componente de luminância contém a informação da intensidade das cores representadas na tela do receptor, ou seja do brilho. Os níveis de variação dessa componente vai do preto ao branco. Assim, os receptores que trabalham nesse formato unicamente, decodificam esse sinal para a formação da sua imagem em P&B (Preto e Branco). Nos sistemas americano e europeu esse sinal é representado por E_Y .

• **Componentes de Crominância**

As componentes de crominância contém informação das diferentes cores que serão representadas na tela.. A vantagem para processamento do sinal de crominância em relação ao sinal de luminância é que olho humano é pouco sensível as suas componentes de altas frequências permitindo limitar a sua largura de faixa mediante a filtragem e a sub-amostragem em relação a faixa das componentes primárias.

Para o caso do sistema americano, as componentes de crominância são limitadas a uma largura de **15 MHz**, enquanto que para o sistema europeu são limitadas a **30 MHz**.

A sub-amostragem é feita por um fator de dois, o que corresponde a uma frequência de amostragem de **37.125 MHz** no caso americano e a uma de **72 MHz** no caso europeu. O efeito da sub-amostragem na imagem implica em que dois *pixels* consecutivos tanto na direção vertical como na horizontal utilizam as mesmas amostras dos sinais de crominância.

Antes da sub-amostragem, o sinal de vídeo precisa ser passado através de um filtro passa-baixas a fim de se evitar a distorção do sinal por causa do efeito de **aliasing** na decodificação digital-analógica. Ao término da sub-amostragem obtêm-se para cada 4 amostras de luminância uma amostra de cada um dos dois sinais de crominância, gerando uma relação de amostras de 4:1 na formação de cada quadro. Esse tipo de amostragem é denominado formato de cor 4:2:0 que será estudado mais adiante.

Os sinais diferença de cor são representados no sistema americano pela nomenclatura E_{PR} e E_{PB} e no sistema europeu por E_{C1} e E_{C2} .

2.2.4.1 Correção Gama

A correção gama constitui uma correção do sinal de vídeo. Ela serve para compensar a não linearidade dos elementos transdutores (câmera, tubo de raios catódicos, etc) e evitar uma distorção da imagem reproduzida na tela do receptor. O tipo de correção é caracterizado pelo fator $\gamma=2.2$ (valor prático) e denomina-se fator **gama**. Cada componente de cor é passada por uma correção gama segundo o formato de geração de componentes. Geralmente a correção gama dos sinais RGB é feita dentro da câmera de televisão.

A correção gama para um sinal é indicada da seguinte forma : $X' = X^{1/\gamma}$

As componentes de cor para HDTV são geradas a partir do sistema de equações 2.2.1 e 2.2.2 cujos coeficientes dependem do sistema proposto:

Sistema americano :

$$\begin{aligned}
 E'_Y &= 0.212E'_R + 0.701E'_G + 0.087E'_B \\
 E'_{PR} &= \frac{(E'_R - E'_Y)}{1.576} \\
 E'_{PB} &= \frac{(E'_B - E'_Y)}{1.826}
 \end{aligned}
 \tag{2.2.1}$$

$$\begin{pmatrix} E'_Y \\ E'_{PR} \\ E'_{PB} \end{pmatrix} = \begin{pmatrix} 0.212 & 0.701 & 0.087 \\ 0.500 & -0.445 & -0.055 \\ -0.116 & -0.384 & 0.500 \end{pmatrix} \begin{pmatrix} E'_R \\ E'_G \\ E'_B \end{pmatrix} \rightarrow \begin{pmatrix} E'_R \\ E'_G \\ E'_B \end{pmatrix} = \begin{pmatrix} 1.00 & 1.5758 & -0.000657 \\ 1.00 & -0.476 & -0.2264 \\ 1.00 & -0.000377 & 1.8259 \end{pmatrix} \begin{pmatrix} E'_Y \\ E'_{PR} \\ E'_{PB} \end{pmatrix}$$

A linha especificada acima de cada componente indica a correção gama.

Sistema Europeu :

$$E'_Y = (0.3392E_R + 0.621E_G + 0.0391E_B)^{1/7}$$

$$E'_{C1} = 1.8(E'_R - E'_Y)$$

$$E'_{C2} = 0.8(E'_B - E'_Y)$$

(2.2.2)

$$\begin{pmatrix} E'_R \\ E'_G \\ E'_B \end{pmatrix} = \begin{pmatrix} 1.000 & 1.576 & 0.000 \\ 1.000 & -0.477 & -0.227 \\ 1.000 & 0.000 & 1.826 \end{pmatrix} \begin{pmatrix} E'_Y \\ E'_{c1} \\ E'_{c2} \end{pmatrix}$$

2.2.5.- Sinais de Sincronismo

Os sinais de sincronismo nos sistemas convencionais de televisão são transmitidos a fim de se indicar ao receptor o início e o término de cada linha de varredura (sincronismo horizontal) e de cada quadro ou campo (sincronismo vertical). Num sistema analógico, um sinal de sincronismo é transmitido mediante níveis de tensão dentro do sinal de vídeo composto.

No caso de um sistema completamente digital como é o caso em questão para HDTV, os sinais de sincronismo são indicados mediante a codificação de cada quadro e de cada linha de varredura. Dessa forma, os níveis de sincronismo provenientes da câmera são utilizados para sincronizar o relógio do sistema de amostragem, a fim de se indicar o início e o término do grupo de amostras ativas (amostras levando só informação de vídeo) que serão posteriormente codificadas e transmitidas. O número de amostras a ser codificadas dependerá do formato de amostragem utilizado pelo sistema.

No receptor, os quadros e as linhas são reproduzidos na tela de acordo com um formato pré-estabelecido que define os pontos de início e fim de cada um deles. Esse formato está armazenado na memória do decodificador. Isso pode ser aproveitado para a correção de alguns problemas de cintilação das grandes áreas nas cenas com muito brilho, quando são transmitidos os campos a uma frequência de 50 ou 60 Hz como acontece nos sistemas convencionais de televisão. A correção pode ser feita mediante a leitura (armazenado na memória) e reprodução de um mesmo quadro várias vezes, a fim de se alcançar frequências de reprodução maiores do que 80 Hz [19], frequência esta que constitui o limiar de detecção do olho para a cintilação em grandes áreas.

Tanto os sinais de cor como os sinais de sincronismo são representados através de níveis de tensão provenientes da câmera antes do processo de digitalização. Os níveis de tensão designados para esses sinais estão de acordo com as propostas dos diferentes sistemas e projetada sobre uma impedância de **75 ohms**. As tabelas 2.2.2, 2.2.3 e 2.2.4 mostram os níveis de tensão considerados para o sistema americano, bem como os intervalos de tempo considerados para os sinais de sincronismo vertical

e horizontal. As Figuras 2.2.3 e 2.2.4 mostram a forma de onda dos sinais de sincronismo.

No caso do sistema Europeu, as características dos sinais de vídeo são mostradas nas Tabelas 2.2.5, 2.2.6 e 2.2.7 e a forma dos pulsos de sincronismo são mostradas nas Figuras 2.2.5 e 2.2.6.

Sinal Nominal	Referência	Nível (mv)
E'_Y, E'_R, E'_G, E'_B com sincronismo	nível de preto	0
	nível de branco	700
	nível de sincronismo	± 300
E'_{PR} e E'_{PB} com sincronismo	nível zero	0
	nível de pico	± 350
	nível de sincronismo	± 300

Tabela 2.2.2 Formato dos níveis de tensão designados para as componentes de cor e os sinais de sincronismo no sistema americano

Característica	Valores Considerados
Forma do sinal de sincronismo	3 níveis bipolar
Intervalo nominal "line-blanking"	3,77 ms
Frequência do relógio de referência	74,25 MHz + 10 ppm
Período do relógio de referência	13,47 ns
Transição de subida do sincronismo	Número de períodos : 0
Transição posterior do sincronismo	Número de períodos : 44
Início do vídeo ativo	Número de períodos : 192
Fim do vídeo ativo	Número de períodos : 2112
Transição anterior do sincronismo	Número de períodos : 2156

Tabela 2.2.3 Características gerais dos sinais de sincronismo e os tempos dos eventos numa linha de vídeo, considerados no sistema americano

Intervalo (Figura 2.2.3)	Períodos de referência	Tempo (ms)
a	44	0,593
b	88	1,185
c	44	0,593
d	132	1,778
e	192	2,586
f (tempo de subida)	4	0,054
linha (total)	2200	29,630
linha (ativa)	1920	25,860

Tabela 2.2.4 Tempo de duração dos sinais de vídeo e sincronismo considerado no sistema americano (Figura 2.2.3)

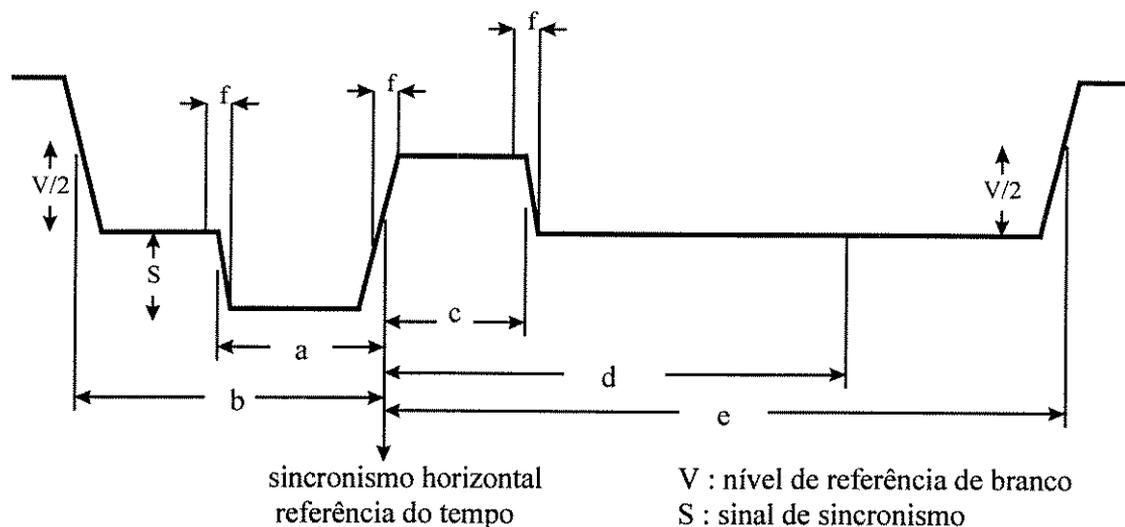


Figura 2.2.3 Forma de onda do período de apagamento de linha considerado no sistema americano

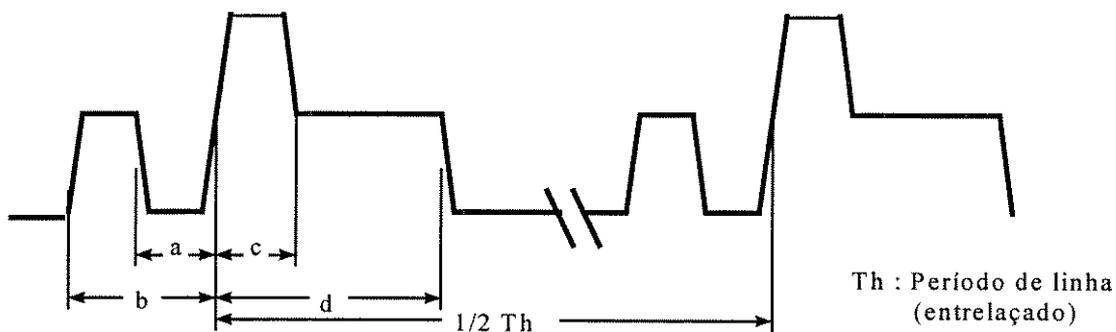


Figura 2.2.4 Forma de onda de um pulso de sincronismo de campo considerado no sistema americano

O sinal de sincronismo de campo considerado no sistema americano, está constituído por 10 pulsos do tipo mostrado na Figura 2.2.4, o espaçamento entre esses pulsos é de meio período de linha.

Sinal Nominal	Referência	Nível (mv)
$E'_Y, E'_R, E'_G, e E'_B$	nível preto nível branco	0 1000
sincronismo	nível sobre 75 ohm	± 300

Tabela 2.2.5 Formato dos níveis de tensão designadas para as componentes de cor e para o sinal de sincronismo no sistema europeu

Símbolo	Característica	ms	Relógio de Referência 2,25 MHz (número de períodos)
H	Intervalo nominal	16	36
a	Intervalo de apag.	2,667	6
b	Intervalo entre O_H e o pórstico de subida do pulso de apagamento	1,778	4
c	pórstico frontal	0,889	2
d	pulso de sincronismo	0,444	1

Tabela 2.2.6 Características do sinal de sincronismo considerado no sistema Europeu (Figura 2.2.5)

Símbolo	Característica	tempo (ms)	Número de Períodos
v	Período do quadro	20	1250
j	Int. Apag. do quadro	1,568	98
q'	Duração do pulso de sincronismo de quadro	8	1/2

Tabela 2.2.7 Sinais de sincronismo de quadro do sistema Europeu (Figura 2.2.6)

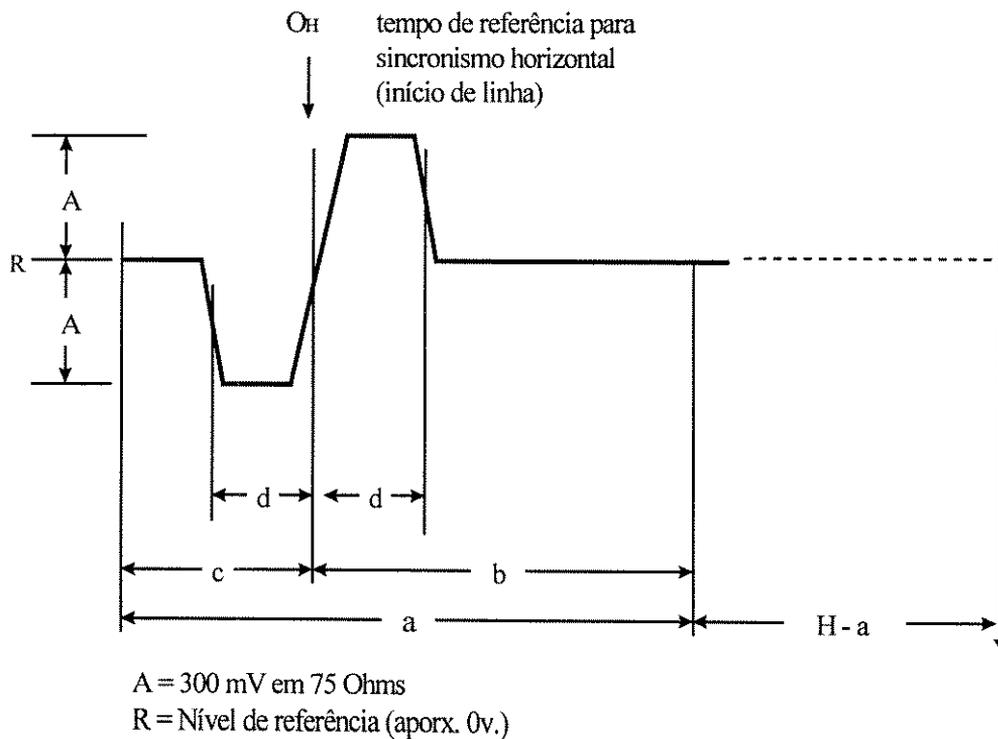


Figura 2.2.5. Forma de onda do sinal de sincronismo de linha considerado no sistema Europeu

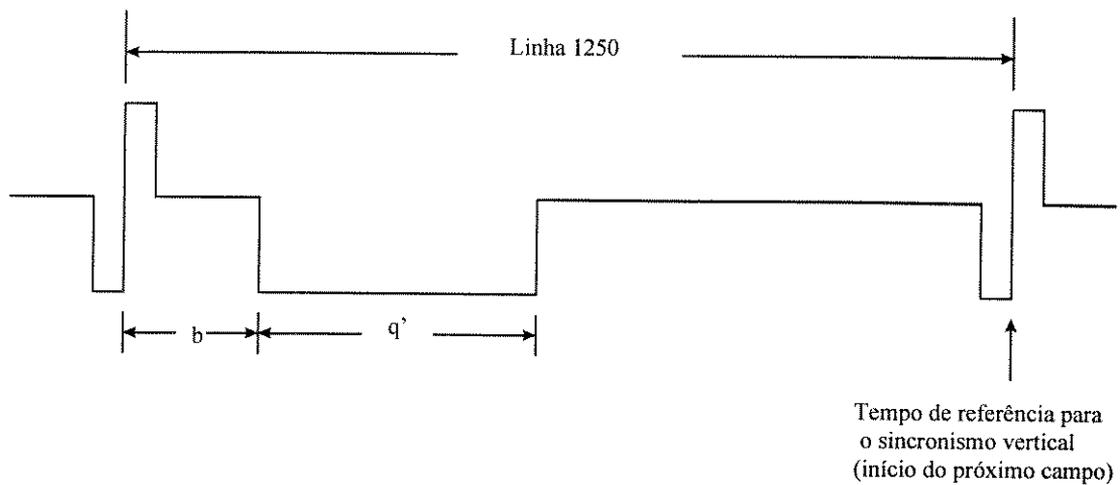


Figura 2.2.6 Detalhe do sinal de sincronismo de campo considerado no sistema Europeu

No sistema europeu, o pulso de apagamento vertical começa na linha 1241 até a linha 88 do seguinte campo. Dessa forma durante o tempo de varredura da linha 1250 e inserido o pulso de sincronismo vertical, cuja forma é mostrado na figura 2.2.6 e cujas características são mostradas na Tabela 2.2.7.

Na Figura 2.2.7. mostra-se um exemplo de um sistema de pré-processamento dos sinais de vídeo para a geração dos sinais digitais.

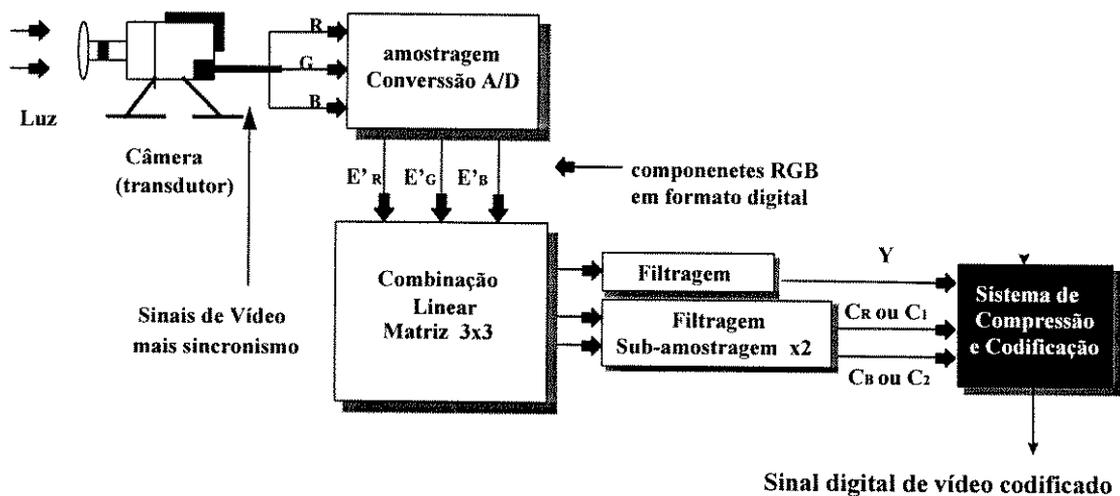


Figura. 2.2.7 Diagrama de Bloco do Sistema de Pré-Processamento do Sinal de Vídeo

2.3.- Organização dos Dados Digitais de uma Imagem de Televisão HDTV

Nos sistemas de televisão digital, uma imagem está representada por um conjunto de dados digitais cujos valores representam os níveis de amplitude ou tensão das componentes do sinal de vídeo. Dessa forma, para facilitar o processamento digital desses dados, uma imagem é dividida em regiões que são caracterizadas pelo número de amostras que levam e pela função que cumprem no processo de compressão e codificação. Uma das vantagens da divisão de uma imagem em regiões, é que muitas vezes isso permite a redução das unidades de armazenamento utilizadas no processo de compressão bem como melhora a eficiência dos algoritmos usados pelo sistema. A organização digital de uma imagem utilizada no sistema HDTV descreve-se hierarquicamente a seguir.

2.3.1.- Pixels

A forma de amostragem de uma imagem faz-se no mesmo sentido de varredura do feixe de elétrons na televisão convencional, quer dizer da esquerda para a direita e de cima para baixo. Uma amostra do sinal de luminância e uma amostra de cada um dos sinais de crominância amostradas de forma sincronizada, representam um dado digital de uma pequena parte individual da imagem que se denomina “*pixel*” ou “*pel*”. O *pixel* é a mínima representação digital de uma imagem.

Cada *pixel* contém um dado digital de 8 bits para cada componente de vídeo, quer dizer está representado por um total de 24 bits[1]. Uma linha de varredura forma uma linha horizontal de *pixels*. Quando a relação do número de *pixels* ativos por linha e o número de linhas ativas por quadro é a mesma que a relação de aspecto da tela (16/9) denomina-se formato de “*pixels quadrados*”. Na Figura 2.3.1 mostra-se a representação de um *pixel*.

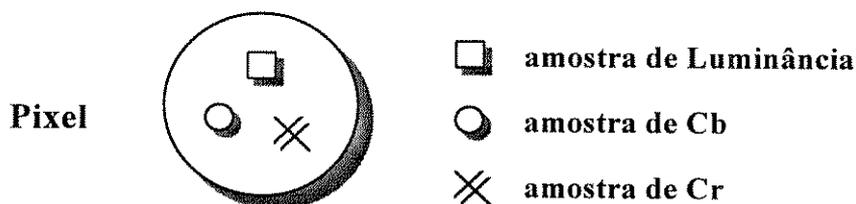


Figura. 2.3.1. Representação de um Pixel

2.3.1.1.- Formatos de Cor

Os formatos de cor especificam a forma como são organizadas as amostras de crominância dentro da formação de cada quadro. Os formatos de cor considerados pelo MPEG-2 para HDTV são descritos a seguir:

• **Formato 4:2:2**

Especifica que dois *pixels* consecutivos de luminância na direção horizontal vão compartilhar a mesma amostra do sinal de crominância. Essas amostras são obtidas diretamente do processo de sub-amostragem por um fator de dois que é feito de acordo com o formato de varredura horizontal empregado pelo sistema. Na figura 2.3.2 apresenta-se a disposição das amostras no formato 4:2:2.

• **Formato 4:2:0**

Especifica as amostras de crominância sub-amostradas tanto vertical como horizontalmente. Nesse caso , para a sub-amostragem vertical o processador faz a interpolação das amostras de crominância de *pixels* adjacentes a fim de obter finalmente o valor que será compartilhado pelos mesmos. (*pixel* da fila superior e *pixel* da fila inferior). Na Figura 2.3.3 mostra-se o formato 4:2:0, onde a sub-amostragem vertical é especificada colocando-se as amostras de crominância a meia distância das filas onde serão colocadas.

Para varredura entrelaçada as amostras de crominância não estão colocadas a igual distância das filas adjacentes de luminância , mas isso é feito a fim que a localização das amostras seja a mesma para os dois formatos de varredura. No caso do formato 4:2:2 a sub-amostragem é feita somente a nível horizontal e é empregada quando se quer obter imagens com maior qualidade de cor. Isso porém, afeta a compressão devido ao fato de que se aumenta o número de amostras a ser transmitido.

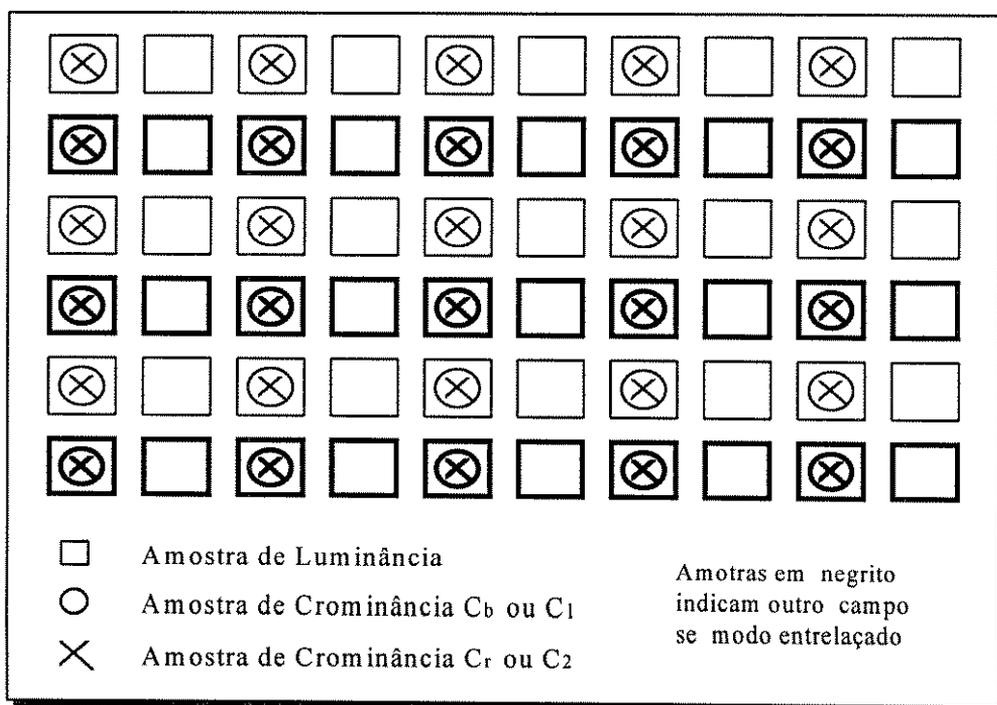


Figura. 2.3.2 Formato de cor 4:2:2

2.3.2.- Blocos de *Pixels*

Os *pixels* são organizados dentro de blocos para um melhor processamento da imagem. Cada bloco é definido por um arranjo de 8 x 8 *pixels*, como se mostra na Figura 2.3.5

Dessa forma, devido à relação espacial entre as componentes de vídeo, cada bloco contém 64 amostras de luminância, 16 de C_r ou C_1 e 16 de C_b ou C_2 , quer dizer 32 amostras de crominância.

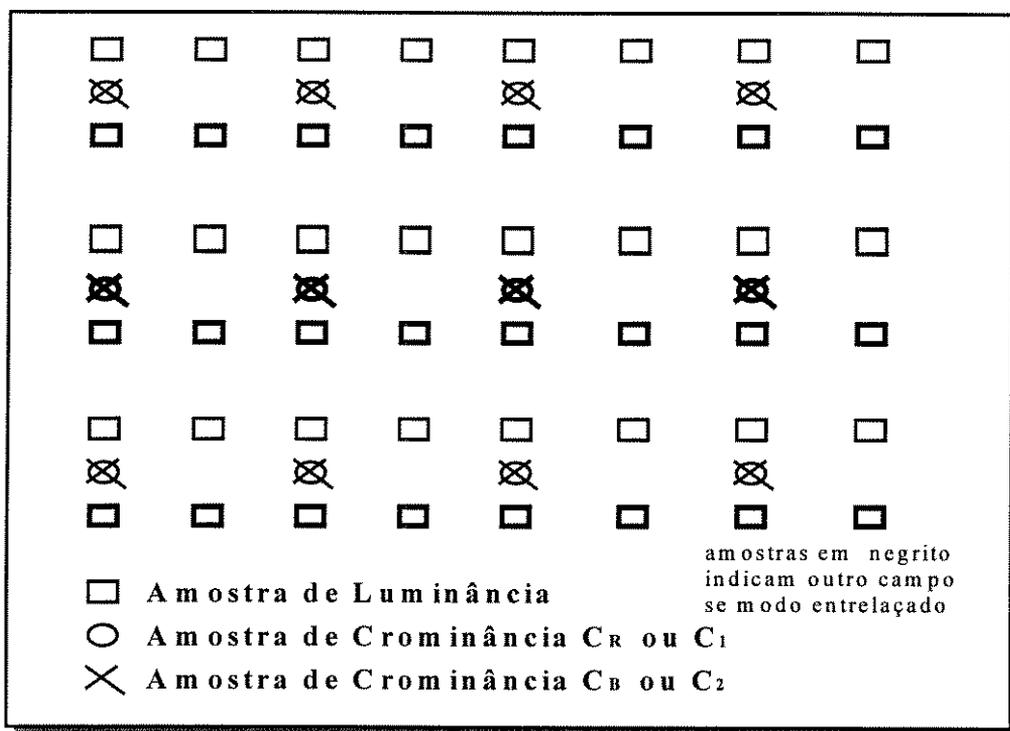


Figura. 2.3.3. Formato 4:2:0

Na Figura 2.3.4 apresentamos a relação física das informações de luminância e de crominância contidas numa imagem.

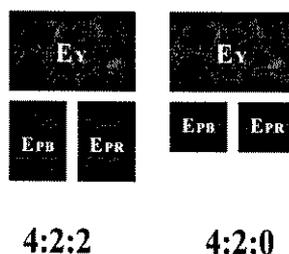


Figura. 2.3.4. Relação física entre o número de amostras de Luminância e Crominância

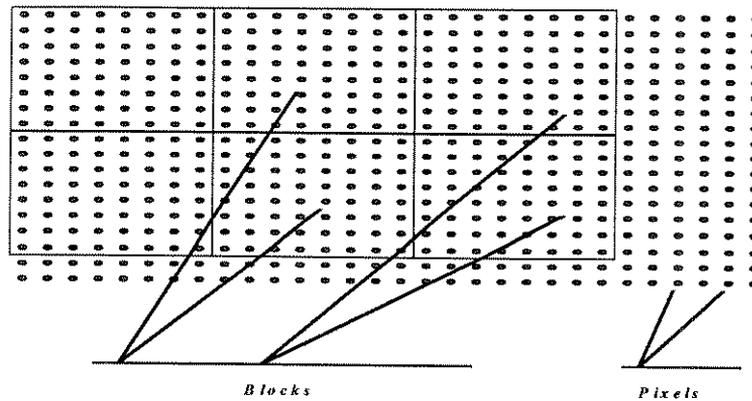


Figura 2.3.5 Representação dos Blocos

2.3.3.- Macroblocos

Um macrobloco consiste de um arranjo de 4 blocos organizados na forma que se mostra na Figura 2.3.6. Devido à relação espacial das amostras das componentes de vídeo, cada macrobloco contém 256 amostras de luminância e 128 amostras de crominância (64 de C_R e 64 de C_B).

Para o formato de varredura de 720 linhas (1280 amostras por linha) tem-se 45 linhas de macroblocos, com 80 macroblocos por linha.

Para o formato de 1080 (1125/60) linhas de varredura (1920 amostras por linha) tem-se 68 linhas de macroblocos, com 120 macroblocos por linha. Note que a existência de um número inteiro de macroblocos na imagem em direção vertical exige um número de 1088 linhas ativas. No entanto, as últimas 8 linhas são designadas pelo codificador a não levar nenhuma informação (linhas mortas). Para o caso europeu tem-se (1250/50) 72 linhas de macroblocos com 120 macroblocos por linha.

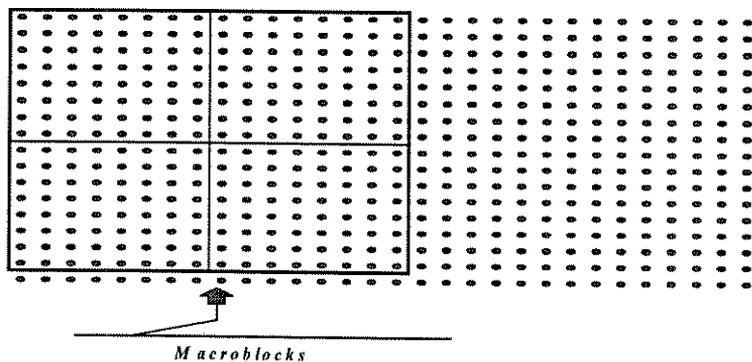


Figura. 2.3.6 Representação de um Macrobloco

2.3.4.- Fatias

Um o mais contínuos macroblocos dentro de uma fila são organizados em fatias. A organização é feita do mesmo modo que no formato de varredura horizontal da televisão convencional, quer dizer da esquerda para a direita.

A organização em fatias, permite um mecanismo que evita a propagação do erro no decodificador. O MPEG-2 gera palavras de código de comprimento variável, e qualquer erro na transmissão dos dados pode ocasionar uma perda de alinhamento na recepção da informação. Dessa forma, cada fatia começa com um “*star code*” que tem uma combinação particular de palavras de código (não considerada na codificação dos dados) e que são designadas pelo MPEG-2.

Depois da ocorrência de um erro na transmissão dos dados, o decodificador pula ao *star code* da próxima fatia para resumir corretamente a decodificação e recuperar a perda de sincronismo na recepção das palavras de código. O número de fatias afeta a eficiência de compressão, mas permite que se tenha uma menor probabilidade de erro na transmissão da informação. A fatia é a mínima unidade de sincronização depois da ocorrência de um erro.

2.3.5.- Seqüência de Quadros e Tipos de Quadros

O conjunto de fatias que formam uma imagem completa na tela denomina-se quadro, e constitui a unidade primária de codificação.

Uma seqüência de vídeo (ou vídeo seqüência) é a sucessão de consecutivos quadros que dão a sensação de movimento na imagem de televisão. Uma vídeo seqüência começa com a **seqüência de cabeçalho** e termina com a **seqüência de finalização**[1]. Qualquer vídeo seqüência pode servir como um ponto de entrada de um novo canal onde o decodificador pode inicializar e fazer a correção da sintaxes[1]. Numa seqüência de cabeçalho o codificador envia os parâmetros que caracterizam a vídeo seqüência atual ; dessa forma o decodificador configura o sistema de acordo com as características da seqüência. Alguns dos parâmetros da seqüência de cabeçalho são os seguintes:

- Resolução horizontal e vertical
- Período de transmissão dos Quadros (*Frames*)
- Relação de Aspecto
- Formato de Cor
- *Profile* e nível do MPEG-2
- Tamanho do *buffer* de vídeo VBV
- Máxima taxa de transmissão de bits

Na Figura 2.3.7 mostra-se uma seqüência de vídeo com os três tipos de quadros que são descritos a seguir.

2.3.5.1.- I-Quadros

São quadros codificados sem levar em conta as características dos quadros anteriores (predição)[7]. A codificação desse tipo de quadros é feita com informação do mesmo quadro explorando somente a redundância espacial da imagem. Esse processo é chamado de **codificação intraquadro I** (onde I denota intraquadro). Os outros processos de compressão que exploram a redundância temporal da seqüência denominam-se **codificação interquadro**.

O uso periódico de predição intra-quadros facilita para o receptor a realização de inicializações em dois momentos: quando ele é ligado e quando muda-se o canal, o que se denomina **acesso aleatório**[1]. No sistema de compressão com movimento compensado, uma predição intraquadro é usada para se inicializar o algoritmo de predição. Portanto essa predição deve estar disponível no decodificador em caso de perda de sincronismo do sistema.

2.3.5.2.- P-Quadros

São quadros onde a predição temporal é feita tomando-se como referência os anteriores I ou P quadros (predição para frente) [7]. Um P-Quadro é formado só de *pixels* da última decodificação de um I ou P quadro. Esse tipo de predição permite explorar as técnicas de codificação inter-quadro para melhorar a eficiência da compressão e a qualidade da imagem. Cada macrobloco contido num P-quadro pode ser codificado mediante predição para frente ou codificação intra-quadro.

2.3.5.3.- B-Quadros

É um tipo de quadro contido na vídeo seqüência que inclui predição do quadro futuro a partir de quadros prévios[7].

O uso de B-quadros permite explorar a alta correlação temporal entre os quadros futuros e os quadros prévios. Na Figura 2.3.7 mostra-se a seqüência de transmissão de quadros. Observa-se que os I/P quadros são transmitidas fora da ordem em que foram formados no transmissor devido ao fato de que o decodificador deve ter disponível quadros futuros para poder decodificar os B-quadros.

Um quadro futuro é formado porque a codificação da seqüência de quadros provenientes do transmissor não é feita em forma consecutiva. De fato, chega-se a um ponto onde se pula um certo grupo de quadros para se fazer a codificação de um quadro que está mais para frente. Esse quadro constitui o quadro futuro.

Depois que o quadro futuro é codificado como um P ou I quadro, os quadros que foram pulados serão codificados como B-quadros tomando como referência o quadro futuro que já foi transmitido e os anteriores P ou I quadros.

O uso de B quadros melhora a eficiência da compressão, mas aumenta o tempo de decodificação uma vez que torna-se necessário o reordenamento dos quadros no decodificador. Além disso, deve-se dispor de unidades de memória para armazenamento dos quadros futuros. Uma desvantagem de se usar quadros futuros no decodificador, é que os B-quadros não podem ser utilizados para predição.

2.3.5.4.- Grupos de Quadros

Os quadros são organizados por grupos **GoP**, onde cada grupo é caracterizado por os valores (N,M), onde N representa o número de quadros por grupo, e M representa a distância entre os I ou P quadros[8]. Para o caso da Figura 2.3.7, N=10 e M=2.

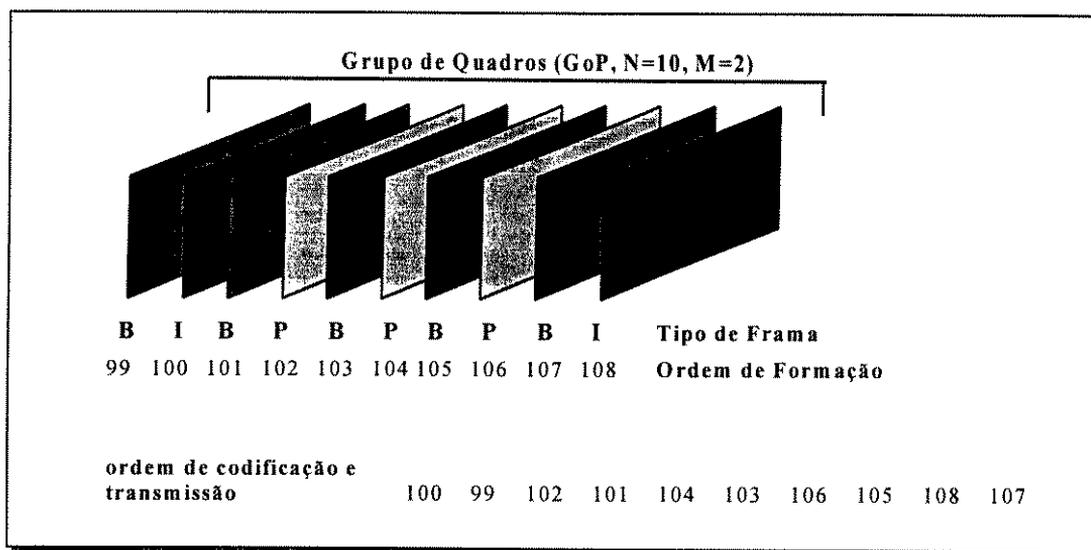


Figura. 2.3.7. Representação de uma seqüência de quadros

Na Figura 2.3.8 mostramos a representação de uma vídeo seqüência e na Figura 2.3.9. mostra-se a estrutura da hierarquia da representação digital de uma imagem

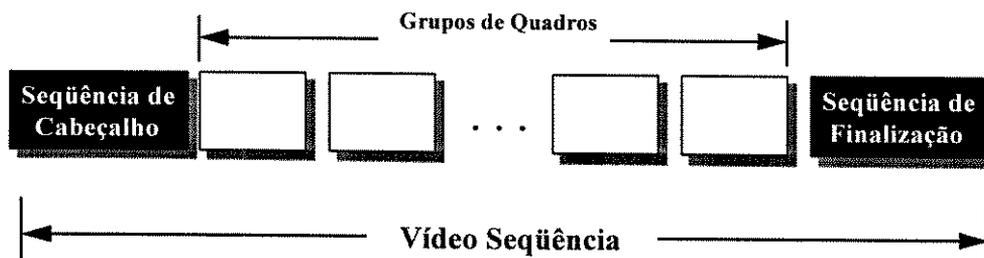


Figura 2.3.8 Representação de uma vídeo seqüência

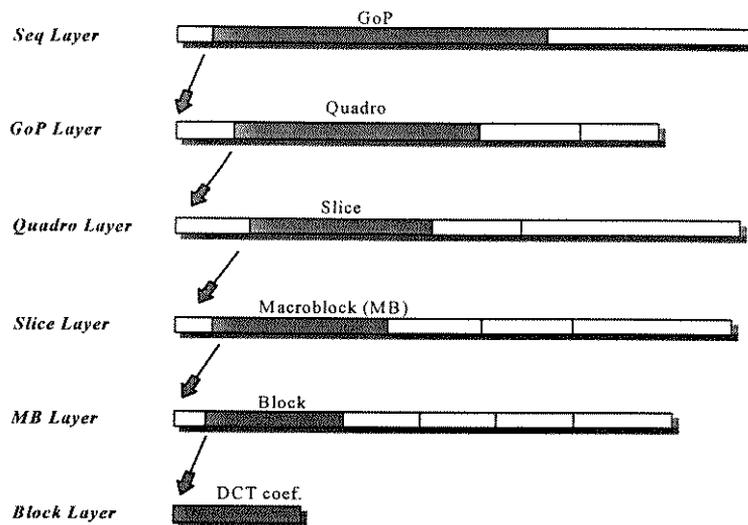


Figura. 2.3.9. Hierarquia da representação digital de uma imagem

2.3.5.5.- Modo de Filmes (Resolução Temporal)

No sistema HDTV americano as possíveis frequências de transmissão de quadros para diferentes filmes foram acordados para: 24 fps, 30 fps e 60 fps (fps significa *frames* por segundo, que equivale a quadros por segundo). Quando os *pixels* de uma vídeo seqüência tem uma variação pequena de um quadro para outro, o processador deve reconhecer automaticamente que se trata de uma transmissão menor do que 60 fps[1].

No caso de 24 fps serem transmitidos ao codificador a uma frequência de 60 Hz, utiliza-se o formato 3:2 pull-down, onde o codificador detecta consecutivamente 3 quadros muito parecidos da seqüência, seguidos por outros dois que apresentam também características similares. Dessa forma, só se codifica 24 fps, o que corresponde ao modo do filme original. O codificador indica ao decodificador o modo de filme que está sendo transmitido.

Para o caso de 30 fps serem transmitidos a uma frequência de 60 Hz, o codificador detecta de forma consecutiva dois quadros de características similares seguidos por outros dois quadros que também apresentam quase as mesmas características e dessa forma codifica somente 30fps, o que corresponde ao modo do filme original.

2.4.- Técnicas de Predição

No processo de compressão do sistema **MPEG-2**, são utilizados algoritmos e processos de tratamento de sinais que exploram tanto a redundância temporal (estimação de movimento e compensação de movimento) como a redundância espacial contida no sinal de vídeo (**DCT - Discrete Cosine Transform**)[5].

A alta correlação temporal existente entre os quadros de uma vídeo seqüência permite a construção de algoritmos que possam predizer um quadro futuro mediante a construção de quadros de predição. Dessa forma, mediante a codificação **DPCM** (*Differential Pulse Code Modulation*), o sistema só transmite as diferenças entre os *pixels* do quadro atual e do quadro de predição. A transmissão das diferenças é uma das primeiras formas que o sistema **MPEG-2** utiliza para a redução da taxa de bits a transmitir[8].

2.4.1.- Compensação de Movimento

Muitas vezes a variação de uma imagem de um quadro para outro constitui o movimento do objeto que está sendo representado na tela. Nesse sentido, o objeto tem um deslocamento tanto horizontal quanto vertical (dx, dy) com respeito a sua posição inicial (quadro passado)[5].

Os deslocamentos vertical e horizontal que formam o movimento do objeto denominam-se “**vetores de movimento**”. Mediante a técnica de movimento compensado, o sistema transmite somente os valores desses vetores e as diferenças entre o quadro de predição e o quadro que está sendo codificado. Essa forma de codificação diminui notavelmente o número de bits a se transmitir.

Os vetores de movimento são calculados a nível de macroblocos, isto é, assume-se que todos os *pixels* que estão dentro do mesmo tem igual movimento. A vantagem de se trabalhar a nível de macroblocos é o incremento da eficiência do algoritmo de “**estimação de movimento**”, tornando o sistema muito mais rápido. No entanto, o desempenho da predição é diminuído à medida que os blocos de deslocamento tornam-se cada vez maiores[1].

No processo de codificação considera-se que macroblocos no quadro atual são preditos por macroblocos no quadro de predição (**Block Matching Algorithm**) como mostrado na Figura 2.4.1.

Um quadro de predição forma-se a partir dos deslocamentos horizontal e vertical (vetores) que tornam o quadro passado mais semelhante com o quadro atual. Quanto melhor for a predição menor será a quantidade de bits a se transmitir. De acordo com isso, um quadro atual e um quadro de predição são definidos da seguinte forma[8]:

$$\text{Quadro Atual} = \text{Quadro Passado} + \text{Quadro Diferença} + \text{Vetor de Movimento}$$

$$\text{Quadro de Predição} = \text{Quadro Passado} + \text{Vetor de Movimento}$$

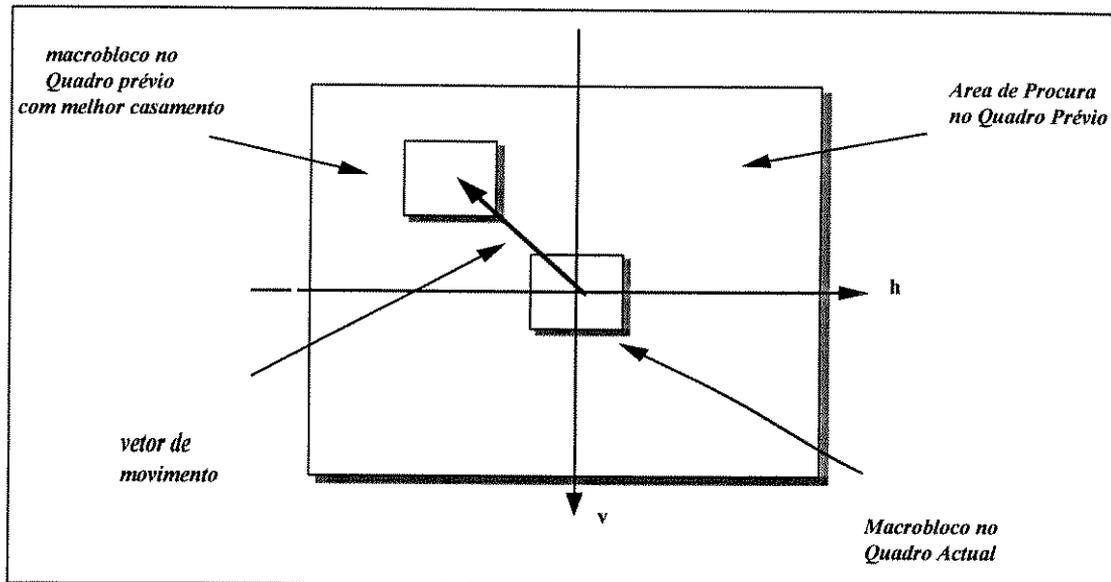


Figura. 2.4.1 Representação dos vetores de Movimento

2.4.1.1 Vetores de movimento numa vídeo seqüência

Os vetores de movimento numa vídeo seqüência são calculados de acordo com o tipo de quadro que está sendo codificado. Para o caso de um P-quadro, os vetores de movimento são calculados a partir dos I ou P quadros anteriores, chamando-se a esse tipo de codificação de **predição para frente** (*forward prediction*).

Todos os macroblocos contidos num P-quadro são calculados por predição para frente ou codificação intra-quadro. Na Figura 2.4.2 mostra-se um conjunto de quadros que formam a vídeo seqüência. Por exemplo, no caso do quadro 5 que constitui um P-quadro, os vetores de movimento são calculados com referência ao quadro 2 que corresponde a um I-quadro. Igualmente o quadro 8 é calculado com referência ao anterior P-quadro, o que seria o quadro 5. Para o caso dos intra-quadros, a codificação é feita sem nenhum quadro de referência e como conseqüência não tem vetores de movimento[7].

No caso dos vetores de movimento para os B-quadros, eles são calculados com referência aos anteriores ou futuros I / P-quadros[1]. Quando a predição é feita tomando como referência um quadro futuro é chamada de **predição para trás** (*backward prediction*) e em caso de se utilizar ao mesmo tempo os quadros anteriores e futuros, ela se denomina **predição bi-direcional**.

Um B-quadro pode apresentar também macroblocos obtidos por codificação intra-quadro. Por exemplo, na Figura 2.4.2 para a criação do quadro B-3 tem-se como referência o quadro prévio I-2 e tem-se também o quadro futuro P-5 utilizando dessa forma predição bi-direcional. Os B-quadros são criados mediante o processo de interpolação entre os quadros prévios e os quadros futuros. A seqüência de movimento entre P-quadros é feita através dos B-quadros que permite explorar a redundância temporal das imagens consecutivas de televisão[7]. O processo de interpolação será estudado na seção de estimação de movimento.

A técnica de movimento compensado aumenta a eficiência da compressão, e dá maior qualidade ao movimento do objeto na imagem.

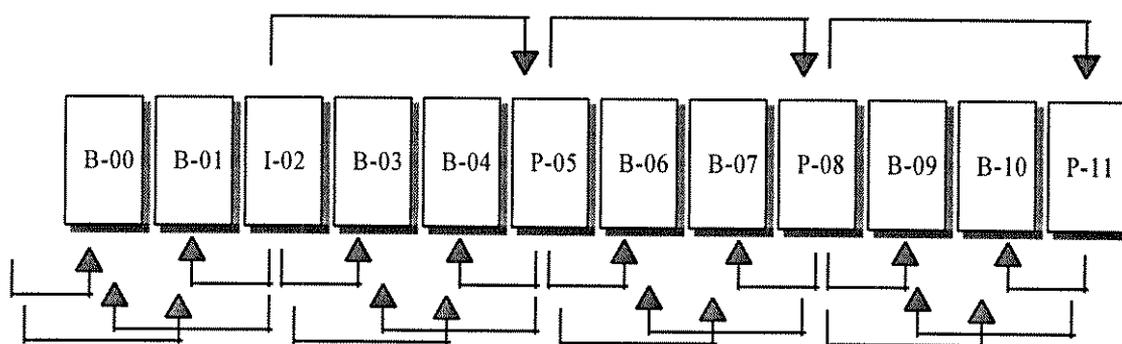


Figura. 2.4.2 Vetores de movimento numa vídeo seqüência

Uma vez obtidos os vetores de movimento, esses são codificados mediante a codificação VLC (Variable Length Code) que apresentaremos mais adiante. Os vetores são transmitidos em forma diferencial ; isso quer dizer que o primeiro vetor de uma fatia é enviado com os seus próprios valores, mas os seguintes são transmitidos mediante a diferença com o vetor anteriormente enviado.

2.4.2.- Estimação de Movimento

A estimação de movimento compreende o conjunto de técnicas utilizadas para o cálculo dos vetores de movimento[5]. Neste ponto, vamos fazer uma introdução matemática para se poder entender melhor o conceito geral da estimação :

Assume-se : $Q(x,y)$ macrobloco passado
 $P(x,y)$ macrobloco atual

Logo, empregando qualquer técnica de estimação de movimento podemos encontrar o melhor casamento entre os dois macroblocos através de um translado vertical e horizontal do macrobloco passado, tal como mostra a equação 2.4.2.1.

$$P(x,y) \approx Q(x - dx, y - dy) \quad (2.4.2.1)$$

Onde dx e dy representam os deslocamentos horizontal e vertical (vetores de movimento) do objeto entre os tempos t_{-1} (passado) e t_o (atual). Portanto, para um movimento uniforme pode-se assumir a seguinte equação[5]:

$$P(x,y) = Q(x - v_x(t - t_{-1}), y - v_y(t - t_{-1})) \quad (2.4.2.2)$$

Onde v_x e v_y são as velocidades uniformes horizontal e vertical, respetivamente.

A partir da equação 2.4.2 podemos derivar uma equação diferencial que descreve o movimento num espaço temporal com velocidade uniforme[5]:

$$v_x \frac{\partial f(x,y,t)}{\partial x} + v_y \frac{\partial f(x,y,t)}{\partial y} + \frac{\partial f(x,y,t)}{\partial t} = 0 \quad (2.4.2.3)$$

Existem vários métodos de estimação de movimento. Entre os mais importantes podemos mencionar o método de casamento de blocos (macroblocos para MPEG-2) que é considerado no sistema MPEG-2.

2.4.2.1.- Método de Casamento de Blocos (*Block-Matching*)

Esse método considera uma região de macroblocos do quadro atual e procura o deslocamento de macroblocos do quadro passado que produz o menor erro possível.

Nessa forma , no quadro atual existirá uma parte da imagem parecida com alguma região no quadro prévio e que dará o menor erro possível. O erro pode ser calculado de distintas formas tal como mostram as seguintes equações:

Correlação cruzada (MCC) :

$$MCC = \frac{\sum \sum P(m,n)Q(m+i,n+j)}{(\sum \sum P^2(m,n) \sum \sum Q^2(m+i,n+j))^{1/2}} \quad (2.4.2.4)$$

Erro quadrático médio normalizado (MMSE) :

$$MMSE = \frac{\sum \sum [P(m,n) - Q(m+i,n+j)]^2}{\sum \sum P^2(m,n)} \quad (2.4.2.5)$$

Erro médio absoluto (MAD)

$$MAD = \sum \sum [P(m,n) - Q(m+i,n+j)] \quad (2.4.2.6)$$

Geralmente, o erro médio absoluto (MAD) é mais utilizado uma vez que não apresenta multiplicações e como consequência tem-se um aumento quanto a eficiência do algoritmo.

Dentro do método de casamento de blocos existem muitas formas de calcular os vetores de movimento de uma vídeo seqüência. Entre os mais importantes temos[6] :

- Algoritmo de procura logarítmica
- Algoritmo de procura em três passos
- Algoritmo de procura telescópica.

Nesses três algoritmos, primeiramente define-se um bloco de $M \times N$ *pixels* (no caso do MPEG-2 seria um macrobloco) e seu respectivo *pixel* central z_0 [6]. Dessa forma a área de procura no quadro anterior para esse bloco é dada pela seguinte expressão:

$$z_0 \pm \begin{bmatrix} \frac{1}{2}(M-1) + d_{max} \\ \frac{1}{2}(M-1) + d_{max} \end{bmatrix} \quad (2.4.2.7)$$

onde o número de *pixels* da região, está definido como :

$$(M + 2d_{max})(N + 2d_{max}) \quad (2.4.2.8)$$

onde d_{max} é definido como o máximo deslocamento do bloco.

Em cada algoritmo são testados distintos possíveis deslocamentos do macrobloco. A eleição do melhor deslocamento será para aquele que resulte no menor erro possível[6]. Esse erro pode ser calculado através das equações 2.4.2.4, 2.4.2.5 e 2.4.2.6.

2.4.2.1.1.- Algoritmo de Procura Logarítmica

O algoritmo de procura logarítmica faz o cálculo dos vetores na direção da mínima distorção. Na Figura 2.4.3 mostra-se a seqüência de procura onde o *pixel* central z_0 é utilizado como referência para o deslocamento do macrobloco.

Em cada passo, o algoritmo faz o cálculo da função erro em 5 possíveis pontos de deslocamento[6], tal como mostra a Figura 2.4.3. Se num determinado passo o erro mínimo é encontrado sem deslocar o *pixel* z_0 , então a distância entre os pontos de procura é diminuída. Isso quer dizer que a partir desse instante são considerados deslocamentos de menor comprimento[6]. Por outro lado, quando o *pixel* z_0 chega até o limite da área de procura selecionada diminui-se a distância entre os pontos de procura.

De acordo com a Figura 2.4.3 os deslocamentos do macrobloco tomando como referência o *pixel* z_0 para 5 passos do algoritmo são : (0,2), (0,4), (2,4), (2,5), (2,6).

Finalmente, o vetor de movimento indica um deslocamento (2,6) para todos os *pixels* dentro do macrobloco.

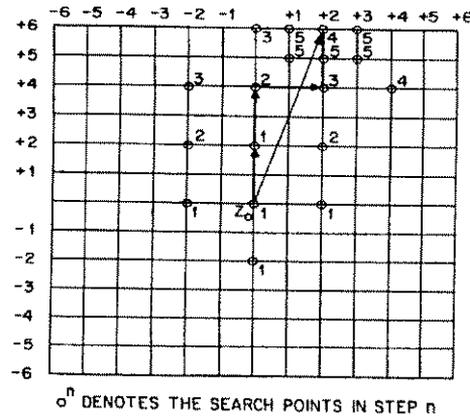


Figura. 2.4.3 Algoritmo de Procura Logaritmica

2.4.2.1.2.- Algoritmo de Procura em três Passos

Neste algoritmo, a procura do melhor casamento é feita em três passos. No primeiro passo são testados oito pontos consideravelmente espaçados em torno do *pixel* central $z_0[6]$. No segundo passo são testados oito pontos em torno do pixel calculado no passo anterior com a diferença de que o espaçamento considerado para esses pontos é menor do que a do passo anterior. No último passo continua-se com o mesmo procedimento até se chegar ao resultado final. Note que a eficiência da predição é incrementada à medida que aumenta o número de passos do algoritmo. Na Figura 2.4.4. mostra-se a seqüência desse algoritmo.

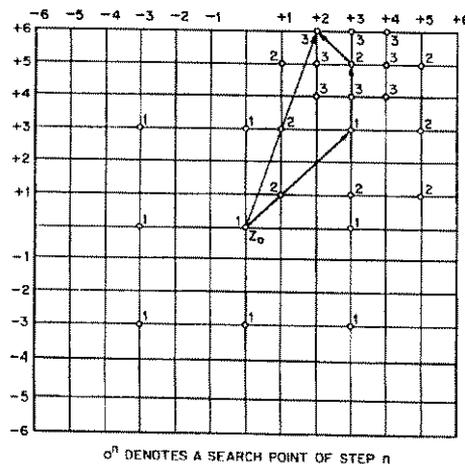


Figura. 2.4.4 Algoritmo de Procura em Três Passos

O tamanho do bloco de $M \times N$ considerado nestes algoritmos não pode ser muito grande, devido ao fato de que estamos assumindo que todos os *pixels* dentro dele tem o mesmo movimento. No entanto, isso não é necessariamente correto, pois muitos daqueles *pixels* podem ter trajetória diferente de um quadro para outro[6]. O sistema MPEG-2 considera para esse caso, os macroblocos de 16×16 *pixels*.

Por outro lado, a precisão dos vetores de movimento é incrementada no sistema MPEG-2 considerando-se no ultimo passo dos algoritmos aqueles deslocamentos de meio *pixel*. Isso é conseguido fazendo-se a interpolação dos *pixels* do macrobloco que vai ser deslocado no quadro prévio e fazendo a respectiva comparação com os valores dos *pixels* que formam o macrobloco do quadro atual. Para o caso de interpolação linear, os valores dos *pixels* podem ser calculados da seguinte forma:

Assume-se 4 valores de *pixels* (P_1, P_2, P_3, P_4) formados como mostrado a seguir. Logo, calcula-se os valores de interpolação P_h, P_v, P_c :

P_1	P_h	P_2
P_v	P_c	
P_3		P_4

onde

$$P_h = \frac{P_1 + P_2}{2}, \quad P_v = \frac{P_1 + P_3}{2},$$

$$P_c = \frac{P_1 + P_2 + P_3 + P_4}{4}$$

Disposição dos *pixels*

Dessa forma, são calculados os valores dos *pixels* de estimação para deslocamentos não inteiros.

2.4.2.1.3.- Algoritmo de Procura Telescópica

Os algoritmos estudados anteriormente apresentam o problema de que à medida que se incrementa o número de passos de procura, o comprimento dos vetores de movimento vão aumentando cada vez mais, podendo ultrapassar o comprimento máximo permitido pelo codificador. Uma solução para esse problema é empregar a procura telescópica que é descrita a seguir.

Assume-se um grupo de quadros como mostra a Figura 2.4.5. Para calcular os vetores para frente, deve seguir os seguintes passos. Considera-se por exemplo os 4 primeiros quadros do grupo. Em seguida, calcula-se os vetores de movimento para se formar o quadro B-1, a partir do quadro I-0, centralizando os valores dos vetores em zero deslocamentos. Na continuação calcula-se os vetores para o quadro B-2 a partir do quadro I-0 centralizando os valores dos vetores sobre os deslocamentos calculados para formar o quadro B-1 no primeiro passo. Igualmente o cálculo dos vetores para formar o quadro B-3 se faz a partir do quadro I-0, mas os seus valores devem estar centralizados nos valores dos vetores calculados no passo dois para formar o quadro 2. O mesmo procedimento deve ser seguido para o cálculo dos vetores para trás, mas

tomando como referência o quadro P-4 e começando o primeiro passo com o cálculo dos vetores para formar o quadro B-3 e assim sucessivamente até o quadro B1.

O cálculo dos vetores de movimento mediante a procura telescópica permite a obtenção dos valores dos mesmos na forma diferencial em relação aos vetores do quadro adjacente, diminuindo assim o número de bits para se representar os seus respectivos comprimentos.

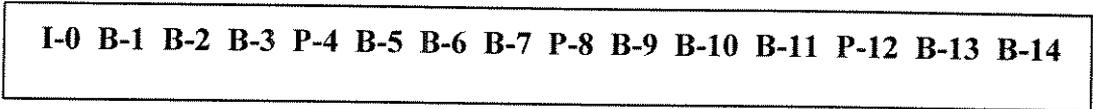


Figura 2.4.5. Seqüência de Quadros Referencial para o Algoritmo de Procura Telescópica

2.4.3. - Interpolação Temporal

Para se calcular os vetores de movimento que formam um B-quadro, o sistema utiliza a técnica de interpolação temporal que permite compensar o movimento do objeto entre o quadro prévio e o quadro futuro. Dessa forma, a técnica de interpolação temporal permite criar e prever novos quadros a partir do conhecimento das velocidades v_x e v_y nos tempos t_p e t_f (quadro prévio e o quadro futuro respetivamente). Então calcula-se os vetores de movimento ($dx=v_x t$ e $dy=v_y t$) num tempo t ($t_p \leq t \leq t_f$) relativos ao quadro passado ou ao quadro futuro. Na Figura 2.4.6 mostra-se a forma como se faz a o processo de interpolação temporal.

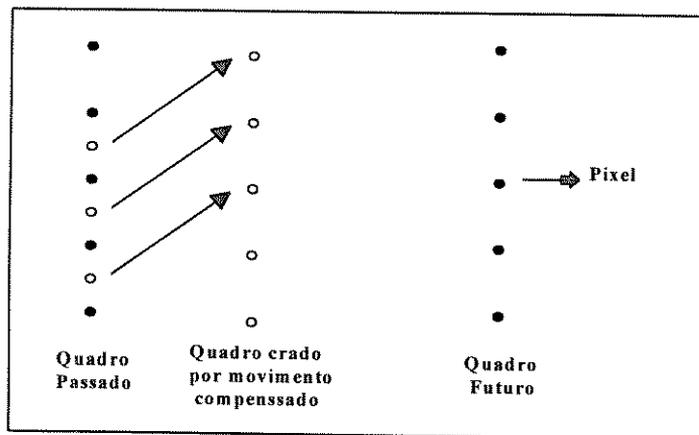


Figura. 2.4.6. Interpolação Temporal

Na Tabela 2.4.1 mostra-se o cálculo dos vetores de movimento para a criação dos B-quadros, usando interpolação. Nessa tabela, I representa o quadro atual que esta sendo codificado.

Tipo de Predição	Referência	Quadro de Predição	Error de Predição I = quadro atual
intra-quadro	valor fixo: 128	nenhum	$E = I - 128$
predição para frente: (B-quadros P-quadros)	Quadro Passado Q_p	$\hat{I} = Q_p + vt$ $\hat{I} = Q_p + \text{vetor de mov.}$	$E = I - \hat{I}$
predição para atras (B-quadros)	Quadro Futuro Q_f	$\hat{I} = Q_f + vt$	$E = I - \hat{I}$
predição bidirecional (B-quadros)	Quadro Futuro e Quadro Passado	$\hat{I} = \frac{(Q_p + vt) + (Q_f + vt)}{2}$	$E = I - \hat{I}$

Tabela 2.4.1. Vetores de movimento para um B-Quadro

2.5.- Sistema de Codificação MPEG-2

Na Figura 2.5.1 mostra-se o diagrama de blocos do codificador utilizado pelo sistema de compressão MPEG-2. A seguir vamos fazer uma breve descrição da função de cada bloco bem como das vantagens de sua utilização.

- **Memória de Reordenamento de Quadros**

Faz o armazenamento das imagens digitalizadas e o reordenamento da seqüência de quadros, de forma que sejam codificados através da codificação I-quadro, P-quadro, e B-quadro.

Como mencionou-se anteriormente para o processo de codificação dos B-quadros é necessário que se faça um reordenamento da seqüência, a fim de que primeiro sejam transmitidos os quadros futuros. Então a partir deles pode-se realizar a sua decodificação.

- **Estimador de movimento**

Encarregado do cálculo dos vetores de movimento para a geração do quadro de predição. Uma vez que os vetores são calculados, são enviados ao codificador VLC (**codificação de comprimento variável**) para sua codificação e transmissão.

O calculo dos vetores de movimento, é feito com base nos valores dos pixels que formam um macrobloco de amostras de luminância.

O primeiro vetor de uma fatia é transmitido o seu valor absoluto, enquanto que os seguintes vetores são transmitidos em forma diferencial em relação aos anteriores

- **Compensador de Movimento**

Reconstrói o quadro de predição a partir dos vetores de movimento e dos quadros prévios reconstruídos (I ou P quadros). Uma vez que o quadro de predição é gerado, passa a ser comparado com o quadro atual de forma que somente sejam transmitidas as diferenças entre os valores dos pixels que formam cada um deles. Quanto melhor for a predição, menores serão os valores dos pixels diferenciais, e menor o número de bits a transmitir.

- **Geração dos Intra-Quadros**

Como foi mencionado anteriormente, um intra-quadro é gerado sem tomar em consideração alguma referência anterior ou futura, pois nesse tipo de codificação transmite-se o valor exato dos pixels que formam o quadro atual.

A transmissão dos intra-quadros afeta a eficiência da compressão mas oferece a vantagem da inicialização do algoritmo de predição no decodificador e permite contornar mudanças rápidas de cenas onde os quadros não são correlacionados. Quando um intra-quadro é gerado, as duas chaves do codificador da Figura 2.5.1 são movidas para a outra posição a fim de que os blocos de predição não sejam utilizados.

- **DCT (Transformada Discreta do Cosseno de Fourier)**

As técnicas de predição estudadas anteriormente reduzem a taxa de bits a se transmitir devido ao fato de que diminuem a redundância temporal contida na seqüência de quadros. No entanto, o número de bits a se transmitir pode ser também reduzido através da exploração da redundância espacial contida numa imagem.

Nesse caso, aproveita-se a pouca sensibilidade da nossa visão para as altas componentes de frequência espacial responsáveis pelos detalhes finos que formam a imagem que está sendo reproduzida na tela. Dessa forma, os pixels diferenciais obtidos anteriormente, são organizados em blocos de 8 x 8 para logo serem transformados para o domínio da frequência através da DCT, cujas vantagens e características veremos mais adiante.

Os valores dos pixels diferença encontram-se no intervalo **(-255, 255)** [7] devido ao processo de quantização com **256** níveis definido na seção 2.2.2. Depois que cada bloco é transformado, os valores dos coeficientes encontram-se no intervalo **(-2048, 2047)**[7]. Muitos desses coeficientes que representam componentes de alta frequência, são de amplitude muito pequena e portanto podem ser reduzidos a

zero para sua transmissão sem que o olho possa perceber essa degeneração na imagem que será reproduzida no decodificador.

- **Quantizador**

Uma forma de se reduzir ainda mais a taxa de bits é fazer a quantização dos coeficientes DCT. Isso pode ser feito porque o olho apresenta pouca sensibilidade a precisão de seus valores e é tolerante a erros de quantização. Dessa forma, cada coeficiente DCT passa por um processo de quantização cuja descrição faremos mais adiante.

- **Codificador VLC (Códigos de comprimento variável)**

Finalmente, os resultados obtidos do processo de quantização são codificados através de códigos de comprimento variável que exploram a redundância estatística contida na informação. Dessa forma reduz-se ainda mais o número de bits a ser transmitido.

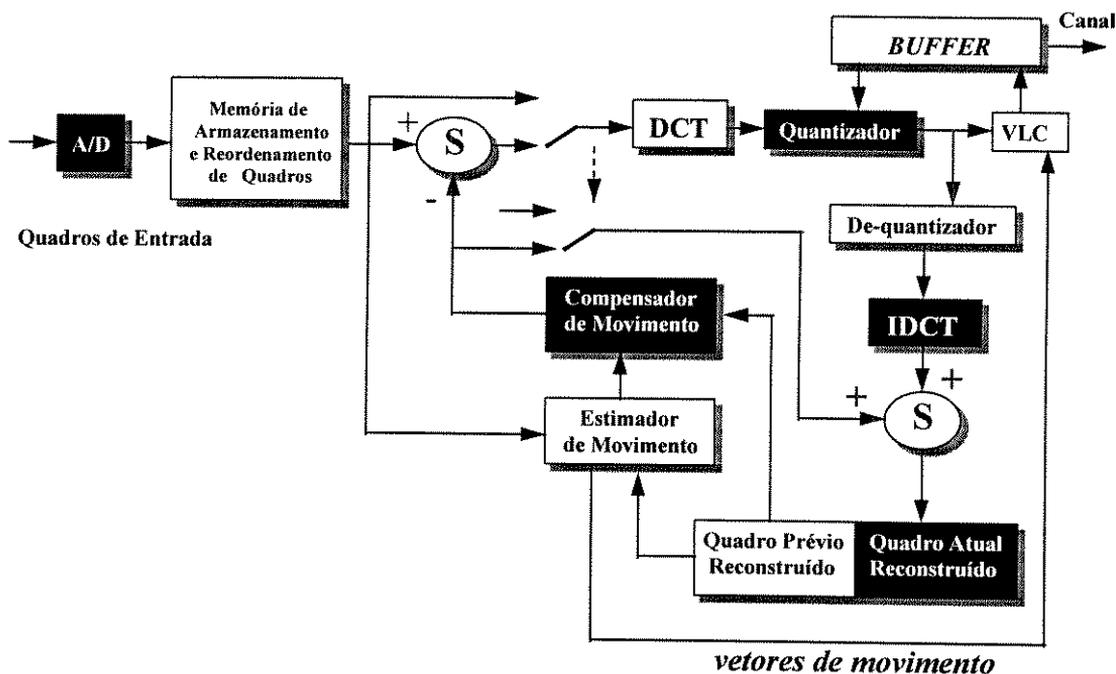


Figura. 2.5.1 Codificador MPEG-2

- **Buffer**

Armazena a informação que será transmitida e também regula a taxa de bits a transmitir mediante a variação dos níveis de quantização aplicados aos coeficientes DCT.

Os blocos **IDCT** e o **de-quantizador** são responsáveis pela reconstrução do quadro prévio que foi transmitido de forma que seja armazenado e tomado como referência para a predição do quadro consecutivo.

Na Figura 2.5.2 apresentamos a variação de ocupação do *buffer* à medida que vai aumentando o número de quadros transmitidos. Note que a quantidade de informação vai diminuindo devido ao fato de que inicialmente não se contava com quadros de predição e portanto a inicialização é feita através dos intra-quadros.

Na Figura 2.5.3 mostra-se a quantidade de informação contida nos diferentes tipos de quadros na seqüência que é transmitida. Os picos altos de informação indicam um intra-quadro, os meios um P-quadro, e os mais baixos indicam um B-quadro.

Finalmente na Figura 2.5.4 mostra a seqüência de codificação das componentes de cor para um determinado macrobloco.

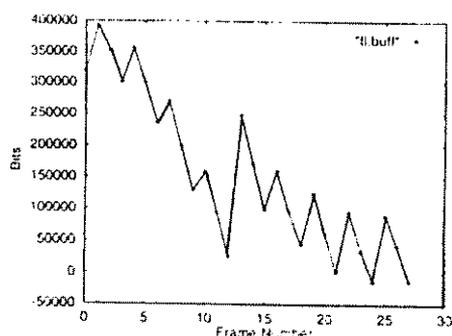


Figura 2.5.2. Ocupação do Buffer

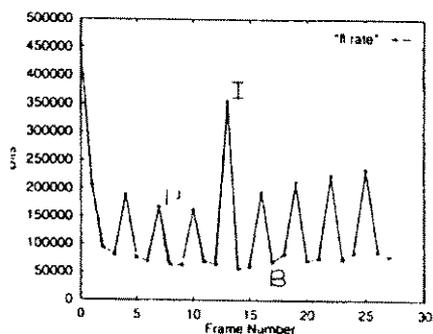


Figura 2.5.3 Quantidade de Bits por Quadro

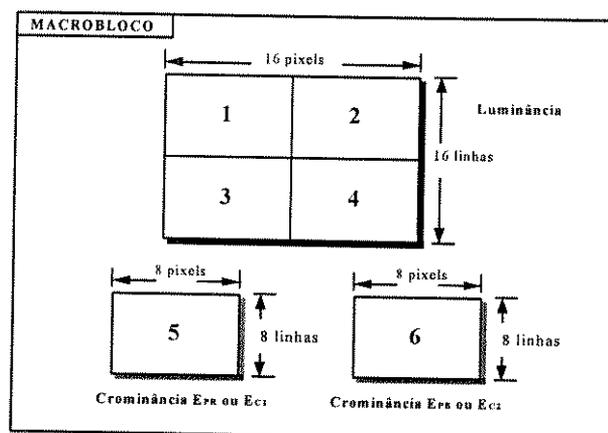


Figura 2.5.4 Seqüência de codificação das componentes do sinal de vídeo que formam um Macrobloco

2.5.1.-Transformada Discreta do Cosseno de Fourier (DCT)

A transformação de uma imagem para o domínio da frequência constitui um dos processos mais importantes para a redução da taxa de bits a ser transmitida, pois dessa forma explora-se as características da resposta do olho humano na percepção dos detalhes mais finos que conformam uma imagem.

A nossa visão apresenta pouca sensibilidade na percepção dos detalhes mínimos, os quais estão representados pelos coeficientes de alta frequência que resultam da transformação. Devido a isso, muitos desses coeficientes são reduzidos a zero ou quantizados com poucos níveis de quantização, de tal forma que a degradação sofrida pela imagem resultante não seja significativamente percebida pelo olho humano.

Por outro lado, mediante a transformação de uma imagem explora-se a concentração da energia nas componentes de baixa frequência, de forma que os outros coeficientes possam ser descartados sem que a imagem sofra uma séria degradação (compactação da energia)[1]. Isso é possível porque a DCT elimina a alta correlação existente entre as amostras de uma determinada região da imagem

Existem muitas transformadas de imagem que fazem a compactação da energia. Entre elas temos a **transformada de Karhunen-Loève (KL)** que constitui a melhor técnica de compactação, mas não é utilizada pela pouca eficiência que oferece no algoritmo de computação dos coeficientes[5]. Por outro lado, temos a **transformada discreta de Fourier (DFT)** que oferece uma maior eficiência no cálculo dos coeficientes, mas tem menos eficiência na compactação da energia.

A transformada que oferece vantagens nos dois casos é a **transformada discreta do Cosseno de Fourier (DCT)** e que é expressa pela equação 2.5.1.

$$F(u,v) = \frac{1}{4} C(u)C(v) \sum_{x=0}^7 \sum_{y=0}^7 f(x,y) \cos\left[\frac{(2x+1)u\pi}{16}\right] \cos\left[\frac{(2y+1)v\pi}{16}\right] \quad (2.5.1)$$

“x” e “y” são os índices dos *pixels* contidos no bloco de 8x8.
 u e v são os índices dos coeficientes DCT do bloco de 8x8 respectivo.
 Além disso, tem-se:

$$C(w) = \frac{1}{\sqrt{2}} \quad \text{para } w=0, \quad C(w) = 1 \quad \text{para } w=1,2,\dots,7$$

A transformada inversa é dada pela seguinte equação :

$$f(x,y) = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C(u)C(v)F(u,v) \cos\left[\frac{(2x+1)u\pi}{16}\right] \cos\left[\frac{(2y+1)v\pi}{16}\right] \quad (2.5.2)$$

2.5.1.1 Transformação espacial por blocos

Para reduzir o tamanho da memória de armazenamento e ter uma melhor eficiência no algoritmo de computação dos coeficientes, o cálculo da transformada é feito por blocos de 8x8 *pixels*, onde cada bloco é calculado independentemente baseado nas características locais. Dessa forma, depois da transformação tem-se cada bloco de *pixels* representado por um bloco de 8x8 coeficientes, como mostram as seguintes matrizes.

$$\begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & f(0,3) & f(0,4) & f(0,5) & f(0,6) & f(0,7) \\ f(1,0) & f(1,1) & f(1,2) & f(1,3) & f(1,4) & f(1,5) & f(1,6) & f(1,7) \\ f(2,0) & f(2,1) & f(2,2) & f(2,3) & f(2,4) & f(2,5) & f(2,6) & f(2,7) \\ f(3,0) & f(3,1) & f(3,2) & f(3,3) & f(3,4) & f(3,5) & f(3,6) & f(3,7) \\ f(4,0) & f(4,1) & f(4,2) & f(4,3) & f(4,4) & f(4,5) & f(4,6) & f(4,7) \\ f(5,0) & f(5,1) & f(5,2) & f(5,3) & f(5,4) & f(5,5) & f(5,6) & f(5,7) \\ f(6,0) & f(6,1) & f(6,2) & f(6,3) & f(6,4) & f(6,5) & f(6,6) & f(6,7) \\ f(7,0) & f(7,1) & f(7,2) & f(7,3) & f(7,4) & f(7,5) & f(7,6) & f(7,7) \end{bmatrix}$$

$f(x,y)$: valor diferencial do *pixel* no caso interquadro o valor do mesmo *pixel* no caso *intra-quadro*

$$\begin{bmatrix} F(0,0) & F(0,1) & F(0,2) & F(0,3) & F(0,4) & F(0,5) & F(0,6) & F(0,7) \\ F(1,0) & F(1,1) & F(1,2) & F(1,3) & F(1,4) & F(1,5) & F(1,6) & F(1,7) \\ F(2,0) & F(2,1) & F(2,2) & F(2,3) & F(2,4) & F(2,5) & F(2,6) & F(2,7) \\ F(3,0) & F(3,1) & F(3,2) & F(3,3) & F(3,4) & F(3,5) & F(3,6) & F(3,7) \\ F(4,0) & F(4,1) & F(4,2) & F(4,3) & F(4,4) & F(4,5) & F(4,6) & F(4,7) \\ F(5,0) & F(5,1) & F(5,2) & F(5,3) & F(5,4) & F(5,5) & F(5,6) & F(5,7) \\ F(6,0) & F(6,1) & F(6,2) & F(6,3) & F(6,4) & F(6,5) & F(6,6) & F(6,7) \\ F(7,0) & F(7,1) & F(7,2) & F(7,3) & F(7,4) & F(7,5) & F(7,6) & F(7,7) \end{bmatrix}$$

$F(x, y)$: valor dos coeficientes DCT correspondentes a um bloco de *pixels* diferença.

Matriz do Bloco de Coeficientes DCT

Esse tipo de processamento introduz o efeito de bloco na imagem resultante . Tal efeito manifesta-se como se a imagem estivesse dividida em pequenas regiões constituídas por blocos de 8 x 8 pixels.

Uma forma de corrigir esse defeito é codificar duas vezes os pixels localizados nas bordas de cada bloco, isso quer dizer que um mesmo pixel é considerado para cálculo em dois blocos adjacentes. Outra forma de eliminar esse efeito é dando os adequados níveis de quantização aos coeficientes DC de cada bloco.

Na escolha do tamanho do bloco para processamento através da DCT deve-se levar em conta as seguintes considerações:

- As dimensões do bloco não devem ser muito grandes, a fim de que as amostras dentro do mesmo não percam a correlação que é aproveitada pela DCT[5].
- Por outro lado, devido a que geralmente uma imagem apresenta regiões que têm uma maior concentração de detalhes que outras, a divisão da mesma em blocos permite que a DCT possa aproveitar essas características, a fim de aumentar a performance da codificação. Dessa forma os blocos de coeficientes correspondentes a regiões da imagem onde concentra-se uma grande quantidade de detalhes serão tratados distintamente a aqueles onde a imagem tem menor numero de detalhes.
- A eleição do bloco de 8x8 permite também aumentar a eficiência do algoritmo de computação da DCT assim como reduzir os requerimentos de memória do sistema.

De acordo com as considerações anteriores, são considerados blocos de 8 x 8 pixels no MPEG-2 para se realizar a DCT das imagens de televisão.

2.5.1.2.- Características dos Coeficientes DCT

O conjunto de coeficientes DCT na matriz do bloco tem características especiais que são aproveitadas pelo codificador para se diminuir a quantidade de bits a transmitir mediante a exploração da redundância espacial contida na imagem.

- Cada coeficiente contém informação segundo seu índice (u,v) sobre as componentes de frequência da imagem. Dessa forma , os índices mais baixos correspondem as componentes de frequência mais baixas e à medida que o índice aumenta os coeficientes representam componentes de frequência cada vez mais altas. Por exemplo, o índice (0,0) representa o valor médio (*DC-Direct Current*) dos coeficientes e por tanto é o valor DC das componentes de frequência.
- A maior quantidade de energia de uma imagem geralmente se concentra nas componentes de baixa frequência, por isso os valores mais altos dos coeficientes serão os que correspondem aos índices mais baixos. Dessa forma , o coeficiente DC contém geralmente o valor mais alto dentro do bloco.
- A percepção humana tem pouca sensibilidade a precisão dos valores dos coeficientes DCT e por isso esses podem ser quantizados através de um quantizador matriz que regula os níveis de quantização para cada coeficiente.
- Geralmente muitos dos coeficientes de alta frequência são quantizados com valor zero, uma vez que os seus valores são muito pequenos e não serão percebidos pelo olho humano[1].

Na Figura 2.5.5 mostramos a transformação do bloco de pixels de 8x8 para o domínio dos coeficientes DCT, onde notamos que a maior concentração da energia está nos coeficientes de baixa frequência.

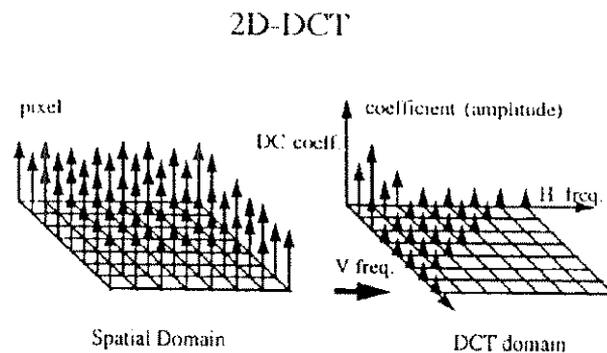


Figura 2.5.5 Representação de um bloco de 8 x8 pixels no domínio da Transformada 2-DCT

2.5.2.- Varredura Zig-Zag

Para se transmitir o arranjo de coeficientes bi-dimensional (matriz), como uma seqüência uni-dimensional utiliza-se o sistema de varredura **Zig-Zag** que define a ordem de transmissão dos coeficientes tal como mostra a Figura 2.5.6. Esse sistema permite enviar primeiro os coeficientes de baixa freqüência quantizados cujos valores são geralmente diferentes de zero para logo transmitir a longa seqüência de zeros correspondentes aos coeficientes de alta freqüência[1]. Essa seqüência de zeros apresenta propriedade de alta redundância que é explorada mediante a codificação *run lenght* que apresentaremos mais adiante.

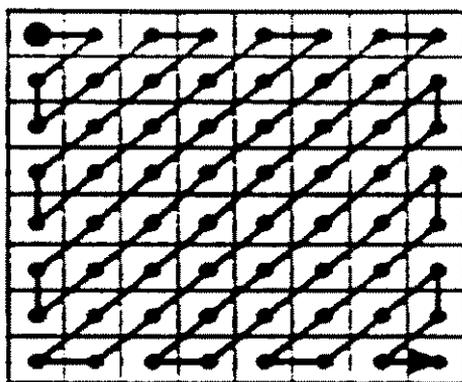


Figura. 2.5.6 Varredura Zig-Zag

2.5.3. - Quantização dos Coeficientes DCT

Aproveitando-se o fato de que a percepção humana é pouco sensível à precisão dos valores dos coeficientes DCT, então esses são quantizados através de um quantizador matriz de 8x8 que especifica o tamanho dos passos de quantização para cada coeficiente contido no bloco.

A quantização constitui uma das formas mediante a qual o codificador reduz um considerável número de bits de transmissão, pois mediante esse processo é perdida parte da informação da imagem sem que o olho possa perceber a degradação.

Dois tipos de quantização são utilizados : **intra-quantizador** (quantizador - intraquadro) e **não-intra-quantizador** .

2.5.3.1.- Intra-quantizador

É um tipo de quantizador matriz aplicado aos coeficientes de um bloco codificado através de esquema intra-quadro. A característica dos coeficientes de um intra-quadro é que eles concentram a maior quantidade de energia em quase todas as componentes de freqüência e portanto o número de níveis de quantização aplicado a

cada um deles deve levar em conta essa característica a fim de que não exista uma séria distorção da imagem que possa ser percebida pelo olho humano. Se os tamanhos dos passos do quantização são muito largos, então a imagem reproduzida pode aumentar o efeito de bloco, quer dizer, pode-se perceber uma imagem dividida em pequenas regiões ou blocos (8x8) através dos quais é calculado a DCT da imagem.

2.5.3.2.- Não Intra-quantizador

Esse tipo de quantizador é aplicado aos tipos de coeficientes que representam um bloco de *pixels* formados com base nos quadros de referência ou predição (inter-quadro).

A característica principal de um bloco DCT gerado a partir da codificação inter-quadro, é que os coeficientes do mesmo contém principalmente componentes de alta frequência, devido ao fato de que a predição é mais precisa para as componentes de baixa frequência. Isso permite que os coeficientes resultantes (alta frequência) sejam quantizados com passos de quantização um pouco mais largos (*coarse quantization*) devido a pouca sensibilidade que o olho humano tem para os seus valores. No entanto, quando esse não é o caso utiliza-se um quantizador intra-quadro.

Exemplos de matrizes de quantização aplicadas a blocos intra-quadro e inter-quadro, são mostrados a seguir.

$$\begin{bmatrix} 8 & 16 & 19 & 22 & 26 & 27 & 29 & 34 \\ 16 & 16 & 22 & 24 & 27 & 29 & 34 & 37 \\ 19 & 22 & 26 & 27 & 29 & 34 & 34 & 38 \\ 22 & 22 & 26 & 27 & 29 & 34 & 37 & 40 \\ 22 & 26 & 27 & 29 & 32 & 35 & 40 & 48 \\ 26 & 27 & 29 & 32 & 35 & 40 & 48 & 58 \\ 26 & 27 & 29 & 34 & 38 & 46 & 56 & 69 \\ 27 & 29 & 35 & 38 & 46 & 56 & 69 & 83 \end{bmatrix} \begin{bmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{bmatrix}$$

Para um bloco Intra-Quadro

Para um bloco Inter-Quadro

Matrizes de Quantização

Note que para o caso intra-quadro, a largura dos passos de quantização vai incrementando-se à medida que os coeficientes correspondentes representam componentes de alta frequência. Portanto de acordo à equação 2.5.3. pode observar-se que o valor desses coeficientes podem ser reduzidos a zero desde que os mesmos representam os detalhes finos da imagem que não são percebidos pelo olho humano. No entanto, é preciso ter um conhecimento das características da imagem, a fim de

se aplicar os critérios que veremos mais para adiante e evitar uma distorção que possa ser percebido pelo olho.

O tipo de quantizador utilizado para cada coeficiente é quase uniforme, mas o que varia é o número de níveis de quantização aplicado a cada um deles. Escolhem-se de acordo os critérios para definição dos níveis e isso será apresentado mais adiante. O tipo de quantizador a utilizar, dependera se matriz é intra-quadro ou não intra-quadro.

A diferença entre esses quantizadores radica no compartamento dos mesmos entorno do valor zero. Dessa forma os quantizadores intra-quadro não apresentam *deadzone*, o que implica que a região de quantização para o valor zero é mais pequena que os passos de quantização para os outros valores. Por outro lado, os quantizadores não intra-quadro apresentam uma grande *deadzone*, a qual é de maior comprimento que os passos de quantização utilizados para os outros valores. Observa-se portanto que dentro de um bloco de coeficientes quantizados não intra-quadro existe uma maior probabilidade que o número de coeficientes zereados seja major que para o caso intra-quadro. A Figura 2.5.7 mostra os dois tipos de quantizadores [8].

Para o caso da quantização uniforme, o valor dos coeficientes quantizados, é dado pela seguinte equação:

$$F^*(u,v) = \frac{F(u,v)}{Q(u,v)} \quad (2.5.3)$$

onde $Q(u,v)$ representa o valor correspondente da matriz de quantização, entanto que $F(u,v)$ é o valor real do coeficiente DCT.

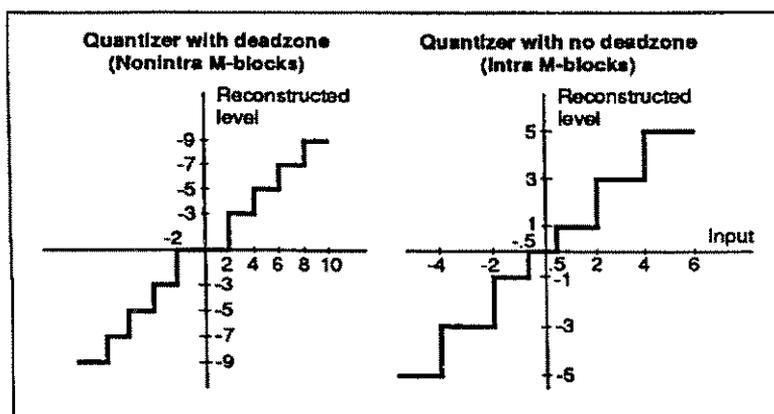


Figura 2.5.7 Quantizadores Não Intra-quadro e Intra-quadro.

2.5.3.3.- Critérios para a escolha do número de níveis de quantização

O erro nos valores dos coeficientes produzidos pelo processo de quantização pode afetar consideravelmente a qualidade da imagem. Dessa forma, o codificador

deve proporcionar a quantidade de níveis de quantização para cada coeficiente a fim de que esse efeito não aconteça.

A quantidade de distorção visível que resulta do erro de quantização, depende da ordem do coeficiente, do brilho da zona do bloco na imagem, e da duração ou da característica temporal do erro. O erro nos coeficientes DC afeta a cada bloco, e é muito mais visível que o erro nos coeficientes correspondentes as altas frequências. Outra consideração que se tem de levar em conta é que a percepção humana percebe mais os detalhes da imagem em zonas com maior brilho do que nas zonas escuras. De igual forma, o erro resultante dos coeficientes de alta frequência é menos visível nas zonas texturadas do que nas zonas planas[1]. Essas duas últimas considerações implicam na variação do número de níveis de quantização levando-se em conta o conhecimento da imagem original. Portanto, a forma como variam as componentes do sinal de vídeo dentro da imagem deve ser analisado pelo codificador a fim de ter conhecimento do tipo de quantização a aplicar num determinado bloco.

As considerações anteriores implicam que o codificador deve dar a quantidade de níveis de quantização para cada coeficiente de acordo com a importância que cada um deles tem na percepção do olho humano (*perceptual weighting*).

A resposta do olho é passa-faixa, quer dizer que cada coeficiente deve ter uma forma particular de quantização. Dessa forma, são fixados os níveis de quantização para cada coeficiente, de acordo com o nível de ruído permitido pelo olho humano.

Outro critério para se variar a escala de quantização dos coeficientes, é a conservação de uma taxa de transmissão aproximadamente constante de tal forma que o *buffer* nunca fique vazio nem tampouco saturado de informação. Para esse último caso o codificador deve variar os níveis de quantização sem afetar a qualidade da imagem. No decodificador se faz o processo inverso de quantização, obtendo-se depois dessa etapa, 64 coeficientes DCT. Então se faz o processo inverso IDCT para se obter os valores dos *pixel* diferença (no caso de inter-quadro) ou o valor absoluto dos *pixels* em caso de intra-quadro.

O processo de quantização implica que a transmissão constante de matrizes de quantização especificando os passos de quantização para cada bloco. Isso representa **0.32%** do total de informação a transmitir, no entanto essa quantidade pode ser diminuída atualizando o quantizador matriz com menor frequência ou re-utilizando anteriores quantizadores matriz quando a diferença com o atual não seja significativa.

2.5.4.- Codificação VLC (código de comprimento variável)

Através do processo da codificação VLC o sistema explora a redundância estatística contida na informação resultante. Parte dessa informação redundante é devida aos zeros representando os valores dos coeficientes depois do processo de quantização.

Dessa forma, uma codificação com palavras de código que tem igual comprimento de bits não oferece uma boa eficiência para a compressão do sinal. Assim, devido a isso considera-se a utilização de um tipo de codificação baseada na entropia ou na estatística da informação a se transmitir.

A entropia é definida como a quantidade média de informação contida numa determinada mensagem, e é expressa pela seguinte equação:

$$H = -\sum_{i=1}^L P_i \log_2 P_i \quad (2.5.4)$$

Onde H é a entropia, P_i é a probabilidade de ocorrência da mensagem. Seja agora a_i a mensagem e L o número de níveis de quantização. Portanto, tem-se :

$$\sum_{i=1}^L P_i = 1 \quad \text{e} \quad 0 \leq H \leq \log_2 L$$

A entropia é uma medida que explora a estatística do sinal a ser codificado e representa a taxa média mínima de bits a se transmitir, mas não especifica o método para se obtê-la. Os métodos estatísticos de codificação implicam que o tamanho das palavras de código variam de acordo com as probabilidades de ocorrência das mensagens a fim de se conseguir uma taxa média de bits o mais próximo da entropia. Esse tipo de codificação denomina-se codificação de comprimento variável *VLC* (*variable length coding*). Um tipo de codificação empregado para a exploração da estatística da informação é a codificação de *Huffman* que veremos na seção seguinte.

2.5.4.1.- Codificação de *Huffman*

A codificação de *Huffman* é um tipo de codificação que explora a estatística das mensagens transmitidas numa determinada seqüência de bits. Dessa forma, as mensagens que tem maior probabilidade de ocorrência serão codificadas com menos bits, e as que tem menor probabilidade de ocorrência serão codificadas com maior quantidade de bits.

À medida que cresce o número de possíveis mensagens para se transmitir, a taxa média de bits obtida por codificação empregando o método de *Huffman* aproxima-se da taxa de bits obtida pela entropia. Dessa forma, esse tipo de codificação é uma alternativa para se aumentar a eficiência de compressão.

O procedimento de codificação de mensagens através do método de *Huffman* é descrito no Apêndice A.

No caso do sistema de compressão **MPEG-2**, emprega-se a codificação de *Huffman* para a representação do par *run-amplitude* que significa corrida de zeros antes de se encontrar um coeficiente quantizado diferente de zero. Dessa forma, através da varredura *Zig-Zag* cada valor de coeficiente diferente de zero, pode estar

precedido por um conjunto de zeros que representam os valores dos outros coeficientes que pelo processo de quantização foram reduzidos a zero. Por exemplo, o par (6,2) representa que os anteriores 6 coeficientes antes do coeficiente de valor 2 são zeros, e no seu conjunto são codificados como um mensagem através da codificação de *Huffman*.

A representação do número consecutivo de zeros são codificados através da codificação *Run-Length*. O par formado pelo número de zeros codificados e pelo valor do coeficiente consecutivo diferente zero são codificados mediante a codificação de *Huffman*.

Os coeficientes DC são codificados sem considerar nenhuma corrida de zeros anterior, portanto têm sua própria distribuição de códigos VLC. Os pares correspondentes a luminância e crominância apresentam diferentes distribuições. Para o caso dos vetores de movimento, o MPEG-2 designa também uma distribuição de códigos VLC de acordo aos valores do seus comprimentos calculados no processo de estimação de movimento. As tabelas de códigos VLC designados pelo MPEG-2 são apresentadas na recomendação **ISO/IEC 13818-2 MPEG-2 - Video**.

2.5.4.2.- Codificação Run Length

Como resultado da quantização, muitos dos coeficientes são aproximados para o valor zero. Além disso, são geralmente os coeficientes das componentes de baixa frequência os que obtêm valores diferentes de zero (compactação da energia). Dessa forma, pode-se observar que muitos dos valores da matriz a se codificar são zero o que permite a utilização de esquemas que exploram a codificação *Run-Length*.

A codificação *Run-Length* explora a repetição consecutiva de um mesmo byte e codifica essa repetição com palavras de código que indicam o número de bytes consecutivos de igual valor. No caso do sistema **MPEG-2**, tem-se uma mistura da codificação de *Huffman* com a de *Run-Length* para se codificar os bits que representam as diferentes imagens.

Geralmente por causa da seqüência de varredura *Zig-Zag*, os últimos coeficientes a se codificar são zero, o que permite ser aproveitado pelo codificador para se fazer a codificação dos mesmos mediante uma palavra de código de poucos bits denominada EOB (*End of Bloco*). Mediante essa palavra de código o decodificador toma conhecimento que o resto do coeficientes do bloco representa de fato zeros e também é um sinal de que se está terminando o envio de um bloco de coeficientes.

A seguir mostra-se a geração dos coeficientes quantizados. Na Tabela 2.5.1 apresenta-se o processo de designação de palavras de código aos diferentes pares obtidos a partir da varredura *Zig-Zag*[8].

$$\begin{bmatrix} -16 & -11 & -6 & -2 & 0 & 0 & 0 & 0 \\ -11 & -4 & -2 & 1 & 4 & 1 & 1 & 1 \\ -5 & 0 & 5 & 8 & 3 & 1 & 1 & 1 \\ 4 & 6 & 7 & 5 & 5 & 4 & 4 & 4 \\ 4 & 5 & 6 & 7 & 7 & 0 & 0 & 0 \\ 6 & 6 & 6 & 6 & 5 & 2 & 2 & 2 \\ 7 & 7 & 5 & 8 & 7 & 2 & 2 & 2 \\ 7 & 7 & 5 & 5 & 8 & 3 & 3 & 3 \end{bmatrix} \begin{bmatrix} 20 & -1 & -12 & 5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & 1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

Bloco de pixels diferença obtidos a partir de um quadro de referência (inter-quadro)

Bloco de Coeficientes DCT correspondente

$$\begin{bmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Matriz de Quantização

Bloco de coeficientes quantizados e zereados

<i>Zero-Run</i>	<i>Nível de Coeficiente</i>	<i>Palavra de Código Huffman</i>
0	1	10
1	-1	0111
0	-1	111
0	-1	111
0	-1	111
2	-1	01011
EOB		10

Tabela 2.5.1 Exemplo de codificação de Huffman para os pares Run-Amplitude

2.5.5.- Buffer de Canal

Quando a codificação entrópica é utilizada, a taxa de transmissão de bits produzidos pelo codificador varia de acordo com a estatística do sinal.

Dessa forma, requer-se a colocação de um *buffer* de saída a fim de que a taxa de bits a se transmitir possa ser controlada e mantida sempre quase constante

mediante a variação dos níveis de quantização dos coeficientes DCT. Se a taxa é muito baixa, o número de níveis de quantização é aumentado e no caso contrario o número de níveis é diminuído. Um circuito de controle da taxa indicará portanto ao quantizador o número de níveis de quantização requerido num determinado momento, de tal forma que o *buffer* não fique saturado nem vazio.

No *Digital Television Standard* é especificado um *buffer* de canal de *8Mbits*[1].

2.6.- Atualização da Imagem

O processo de atualização de imagem consiste na transmissão de informação independente da predição, a fim de se evitar a propagação de erro no decodificador.

Além disso, a atualização da imagem serve para que o decodificador possa inicializar a seqüência de vídeo, quando é ligado, quando se muda de canal e depois da ocorrência de um erro. A atualização de imagem pode ser de duas formas:

2.6.1.- Transmissão Periódica de I-Quadros

Nesse caso, a atualização (*refreshing*) da imagem é feita por uma transmissão periódica de I-quadros. Esse processo permite ao decodificador fazer a inicialização do algoritmo de predição quando entra um novo canal. O período de transmissão de I-quadros pode variar, mas isso depende do codificador. Na documentação do ATSC é especificada a transmissão de um I-quadro a cada 0.5 segundos[1].

Quando o receptor entra num novo canal onde é utilizado I-quadros para a atualização de imagem, ele realiza os seguintes passos:

- Ignora todos os dados até receber a seqüência de cabeçalho.
- Decodifica a seqüência de cabeçalho e configura o sistema de acordo com os novos parâmetros.
- Ignora todos os dados até receber o primeiro I-quadro.
- Decodifica e apresenta a imagem.

Quando acontece um erro num I ou P quadro, o erro será propagado devido a utilização de quadros de predição até receber um I-quadro livre de erro.

2.6.2.- Atualização Progressiva

A atualização progressiva é um método no qual uma parte de cada imagem é codificada usando-se o modo intra-quadro. Dessa forma, as regiões são escolhidas de tal forma que depois da transmissão de um certo número de quadros, todos os macroblocos são codificados usando-se predição intra-quadro ao menos uma vez. O

decodificador somente tomará em conta os macroblocos intra-quadro para fazer a inicialização ignorando os outros.

2.7.- Decodificador MPEG-2

No processo de decodificação, o sistema faz as operações inversas do codificador a fim de se gerar os quadros que serão mostrados na tela do receptor.

Inicialmente a informação é recebida num *buffer* e logo é passada por um decodificador VLD para obter os vetores de movimento e os valores dos coeficientes quantizados. A forma como o decodificador identifica um vetor de movimento ou um coeficiente dependerá de como foram codificados e também da ordem em que foram transmitidos.

Os vetores de movimento são sempre transmitidos primeiro .de tal forma que o decodificador possa calcular o quadro de predição que será a referência para o cálculo do próximo quadro.

Depois de se obter os coeficientes quantizados, faz-se o processo inverso de quantização para se obter os valores dos coeficientes DCT através do sistema inverso de varredura *ZIG-ZAG*, o que permite formar os blocos de 8 x 8 coeficientes. Logo em seguida, faz-se o processo IDCT para formar os valores dos *pixels* diferença que compensaram o erro gerado no cálculo do quadro de predição. Uma vez formado o quadro, este será finalmente mostrado na tela.

As unidades de armazenamento realizam finalmente o reordenamento dos quadros que foram transmitidos fora de seqüência devido a codificação dos B-quadros. Na Figura 2.7.1 mostramos o diagrama de blocos do decodificador MPEG-2.

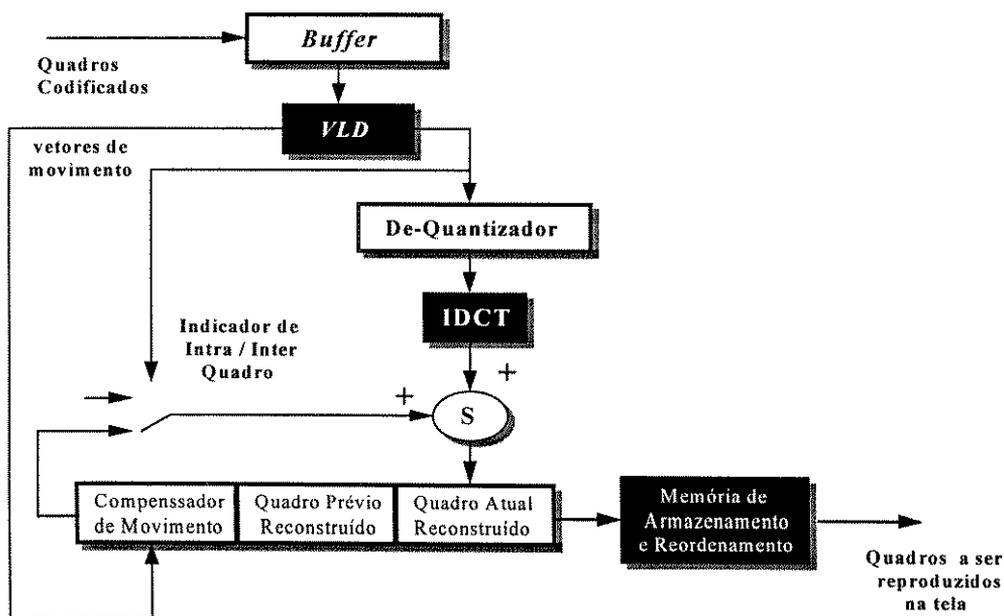


Figura. 2.7.1 Decodificador MPEG-2

Quando um intra-quadro é transmitido, a chave da Figura 2.7.1 é movida para a outra posição uma vez que não é utilizada nenhuma referência na decodificação dos mesmos. Após que os dados digitais da imagem são ordenados, faz-se a conversão D/A (conversão digital-analógico) para a geração dos componentes primários RGB.

2.8.- Conclusões e Comentários Finais

A taxa de bits resultante do processo de compressão é de **18.2 Mbps**. Como veremos mais adiante, essa taxa de bits é aceitável, desde que é utilizado o sistema de modulação **8-16 VSB (Grande Aliança)** para a transmissão do sinal de HDTV dentro do canal convencional de **6MHz**.

Dessa forma, o MPEG-2 constitui um dos sistemas de compressão que mais se ajusta aos níveis de compressão exigidos pelas limitações do canal de conservando os níveis de qualidade da imagem original, ou seja a degradação não é percebida pelo nosso sistema de visão.

Nesse capítulo temos procurado apresentar todo o processo de tratamento digital do sinal de vídeo enfatizando-se o sistema de compressão do MPEG-2 que foi adotado pela *Grand Alliance* dos Estados Unidos na padronização dos sistemas digitais de HDTV.

Esse processo estabelece a base dos níveis de compressão bem como o grau de qualidade que finalmente o telespectador deverá apreciar na imagem da tela do receptor.

Provavelmente temos deixado de lado muitos detalhes, mas a idéia principal deste trabalho é apresentar uma visão geral do que atualmente está sendo considerado para televisão de alta definição.

A qualidade da imagem obtida depois do processo de compressão dependerá também do desempenho dos algoritmos de predição e dos critérios de quantização dos coeficientes DCT.

À medida que passam os anos, muitos desses algoritmos deverão ser aperfeiçoados, explorando-se talvez outras características do sinal aumentando-se as vantagens quanto a uma maior eficiência da técnica de compressão.

Da mesma forma, simultaneamente as técnicas de codificação que estão aparecendo podem permitir um melhor aproveitamento do canal aumentando inclusive a qualidade da imagem a ser transmitida.

A formação do MPEG-2 através de níveis e *profiles* permite uma contínua evolução do sistema para diversas aplicações e logicamente também para o sistema HDTV.

Capítulo 3

Processamento Digital do Sinal de Áudio

3.0.- Introdução

Dentro de um sistema de televisão **HDTV**, deve-se considerar também a transmissão de um sinal digital de áudio, com um alto grau de fidelidade. De fato existe o som associado as imagens de alta definição , consideradas nos diferentes formatos.

Um sinal digital de áudio com um alto grau de fidelidade, é obtido a princípio, mediante a amostragem do sinal analógico original a uma frequência não menor do que **40 KHz** e com pelo menos **16** bits de precisão na representação digital do valor de cada amostra.

Por outro lado, para que a reprodução de um som seja mais agradável para nosso sistema de audição e tenha a maior naturalidade possível pode-se implementar a reprodução simultânea de vários canais de áudio localizados estrategicamente dentro do ambiente onde o sinal está sendo reproduzido. Nos sistemas de som convencionais são utilizados geralmente dois canais de áudio que constituem o formato estéreo.

Atualmente por exemplo, existem sistemas que consideram a reprodução de um sinal com as características anteriores através de 6 canais de áudio que reproduzem um som natural de alta fidelidade. Os efeitos que resultam disso são apropriados para sua aplicação em filmes de cinema.

Entre esses sistemas encontra-se o sistema de áudio **Dolby Surround AC-3** que foi lançado pelos **Laboratorios Dolby** dos Estados Unidos e cujos resultados para o processamento de áudio de diversos filmes do cinema foram muito satisfatórios.

Dessa forma, diante da iniciativa de se criar um sistema de televisão completamente digital e com um alto grau de qualidade , a **Grande Aliança** considerou o **DOLBY AC-3** como o sistema de tratamento de áudio mais adequado para o emergente sistema de televisão **HDTV** proposto para os Estados Unidos.

O **Dolby AC-3** considera a transmissão de um sinal de som através de 5 canais de áudio mais um canal de baixas frequências. Isso resulta em um melhor desempenho dos sons fortes como explosões, motores e efeitos especiais. A frequência de

amostragem considerada pelo sistema AC-3 é de **48 KHz** por canal, o que demanda uma taxa de maior do que **5 Mbps** para a transmissão dos seis canais de **áudio**.

Isso constitui um problema devido às limitações do canal de transmissão, pois de fato, no sistema completo deve-se levar junto com o sinal de áudio a informação do sinal de vídeo cuja taxa é de **18.2 Mbps**.

Para a solução desse problema é necessário realizar a **compressão** do sinal a fim de se reduzir a taxa de bits a transmitir conservando os padrões de qualidade do sinal original. O sistema **Dolby AC-3** oferece um potente sistema de compressão baseado na sensibilidade do ouvido humano que é diferente para as diversas componentes de frequência que formam o espectro da audição.

Dessa forma, o sistema AC-3 é concebido levando-se em conta nosso sistema psico-acústico, explorando as características de resposta em frequência de nosso sistema de audição. Assim as degradações que porventura o sinal venha a sofrer não serão percebidas.

Neste capítulo vamos fazer uma descrição do sistema de áudio **Dolby AC-3**. Trata-se de uma excelente alternativa de aplicação para sistemas de televisão HDTV. Cabe ressaltar, que grande parte dos formatos considerados neste capítulo foram tomados do documento "**Digital Audio Compression (AC-3) Standard**" emitido pelo ATSC.

3.1.- Sistemas de Compressão de Áudio

Antes de começar nosso estudo sobre o sistema de compressão *Dolby AC-3*, vamos apresentar um esquema comparativo desse sistema com outros sistemas de compressão que também estão disponíveis para aplicação em televisão digital. As características de cada um deles são mostradas na Tabela 3.1.1 [8].

Sistema	Características
MPEG-1-Áudio	<ul style="list-style-type: none"> • Sistemas: Mono e Estéreo • 32 sub-faixas de codificação • Frequências de Amostragem: 32, 44.1 e 48 KHz • Taxa de Bits : 32 - 448 Kbs • Qualidade em Radiodifusão : 224 Kb/s (de acordo com o <i>LayerII</i> recomendado pelo ITU-R)
MPEG-2-Áudio	<ul style="list-style-type: none"> • Multicanal (5+1Aux) canais de codificação • Compatibilidade com o MPEG-1 • Frequências de Amostragem : 32, 44.1 e 48 KHz e seus meio valores (16, 22.05 e 24 KHz). • Taxa de Bits : 32-448 Kb/s • Qualidade em Radiodifusão : <i>Layer II</i> - aprox. 224 Kb/s para sistema estéreo (2/0), ≥ 384 Kbs multicanal (3/2)
DOLBY AC-3	<ul style="list-style-type: none"> • Multicanal : (5 + 1 canal de baixas frequências) • Formato de compressão similar a MPEG-2. • Frquências de Amostragem : 32, 44.1 e 48 KHz • Taxa de Bits: 32-640 Kbs • Para ATV (<i>Advance Television System</i>) ≤ 384 Kbs (<i>Grande Aliança</i>)

Tabela 3.1.1 Características dos Sistemas de Compressão de Áudio

Os sistemas de compressão que estão sendo considerados pela maioria dos padrões são o **MPEG-2** e o **Dolby AC-3** o qual já foi adotado pelo sistema proposto pela **Grande Aliança** e será objeto de nosso estudo neste capítulo.

3.2.- Sistema de Compressão de Áudio *Dolby AC-3*

Como é feito em todo sistema de compressão, o tratamento digital é feito através de diversos algoritmos e processos que exploram a vantagem das características do sinal para se realizar a compressão e a codificação do mesmo. Dessa forma, a primeira coisa que o sistema faz é a exploração das características do sinal dentro do domínio da frequência (**Análise Espectral *Filterbank***), a fim de se fazer o tratamento digital com base na sensibilidade que cada componente de frequência tem dentro de nosso sistema de audição. A vantagem disso é que de acordo com a importância da componente, o valor do coeficiente correspondente pode ser quantizado com maior ou menor número de níveis de quantização de tal forma que o número de bits para a representação de cada amostra será variável e muito menor que aquele correspondente ao sinal original.

Por outro lado, os níveis de compressão alcançados pelo sistema AC-3 permitem que além do serviço completo de áudio, se possa oferecer alguns serviços adicionais para melhorar a qualidade do som e também para possibilitar maiores facilidades aos usuários. Cada um desses serviços será estudado ao final deste capítulo. As Figuras 3.2.1a e 3.2.1b. mostram o diagrama de blocos do sistema de codificação e decodificação AC-3.

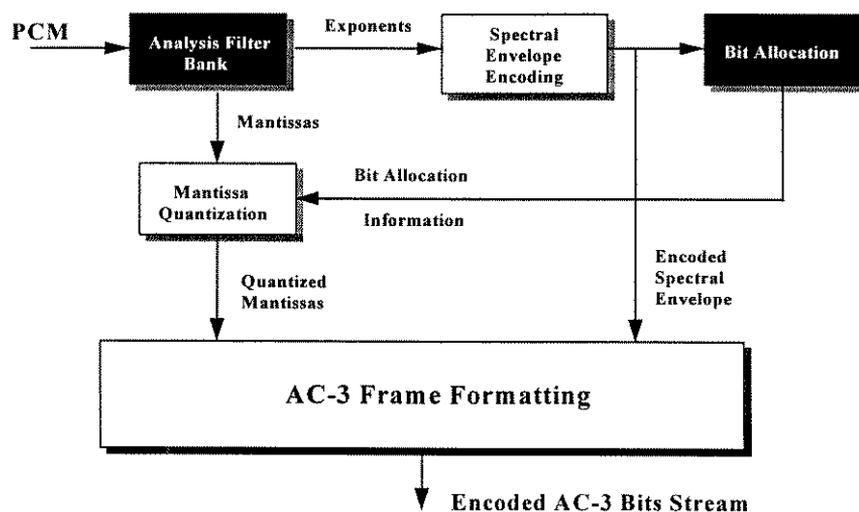


Figura 3.2.1a Sistema de Codificação Dolby AC-3

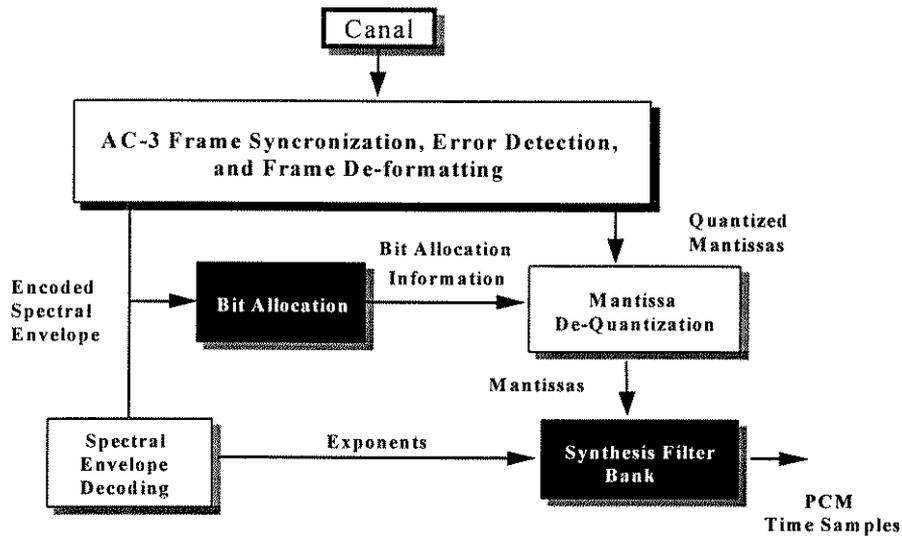


Figura 3.2.1b. Sistema de Decodificação Dolby AC-3

3.2.1- Canais de Áudio

O sistema de compressão *Dolby AC-3* trabalha com até seis canais de áudio de entrada no sistema de codificação (geralmente os seis canais são considerados como 5.1 canais de áudio, pois o canal de baixas frequências é considerado como 0.1). O sistema de canalização é feito de acordo com a recomendação ITU-R BS-775, denominado “*sistema de som estereofônico multicanal com e sem acompanhamento de vídeo*”[1]. A largura de faixa designada para cada canal é mostrada na Tabela 3.2.1.

Canal de Áudio	Largura de Faixa Faixa Base
Esquerdo	20 KHz
Direito	20 KHz
Central	20 KHz
Esquerdo Adjacente (<i>surround</i>)	20 KHz
Direito Adjacente (<i>surround</i>)	20 KHz
LFE Canal de Baixas Frequências	120 Hz

Tabela 3.2.1. Canais de áudio do codificador AC-3

A disposição de cada canal dentro do ambiente de reprodução é atribuída pelo sistema AC-3 tal como mostra a Figura 3.2.2.

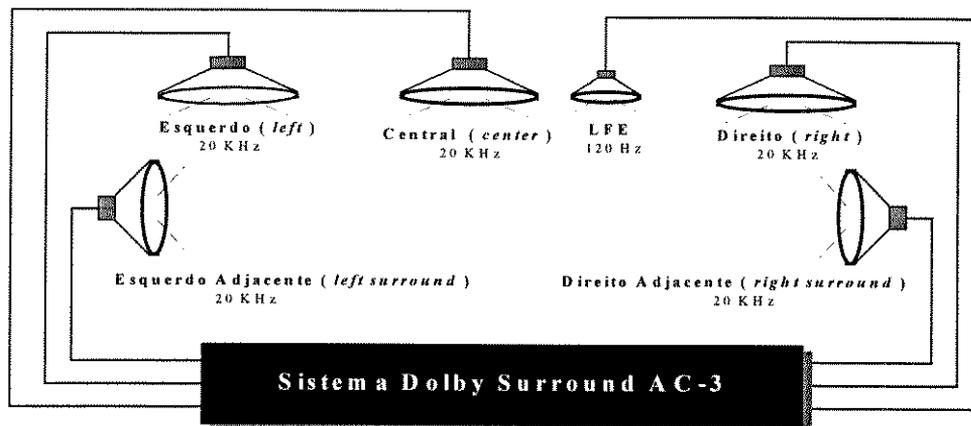


Figura 3.2.2 Disposição dos canais de áudio de um sistema Dolby Surround AC-3

Cada sinal de entrada no sistema pode estar na forma analógica ou na forma digital. No caso de estar na forma analógica, precisa-se fazer a conversão analógico-digital a fim de que se possa processar o sinal usando-se o algoritmo de compressão e codificação.

3.2.2- Conversão Analógico-Digital (PCM)

Cada canal de entrada no codificador é amostrado a uma frequência de **48 KHz** (para televisão). Essa frequência é gerada a partir do oscilador padrão de **27 MHz** [1], a fim de se sincronizar o codificador de áudio com o codificador de vídeo e com o sistema de transporte. A frequência de **48 KHz** é um compromisso entre as frequências de varredura de quadro e de linha. Com isso, assegura-se um número inteiro de amostras de áudio por quadro ou campo.

Se o sinal de entrada no codificador estivera em forma digital e a uma frequência de amostragem diferente, o codificador faz a conversão a **48 KHz** a fim de que todo sinal seja codificado a partir do mesmo formato de amostragem.

Cada amostra de áudio é quantizada e codificada com uma precisão de **16 a 24 bits** de tal forma que a saída do conversor analógico digital fica constituído por amostras **PCM (Pulse Code Modulation)** cuja precisão de representação pode variar nesse intervalo.

A filtragem dos canais de áudio e o processo de conversão **A/D** é mostrado na Figura 3.2.3. Note que no caso do canal LFE, a filtragem passa-baixas é feita por um filtro com uma frequência de corte de **120 Hz**. O nível **dc** contínuo é eliminado, mediante um filtro passa-altas com uma frequência de corte de **3Hz (-3dB)** [10].

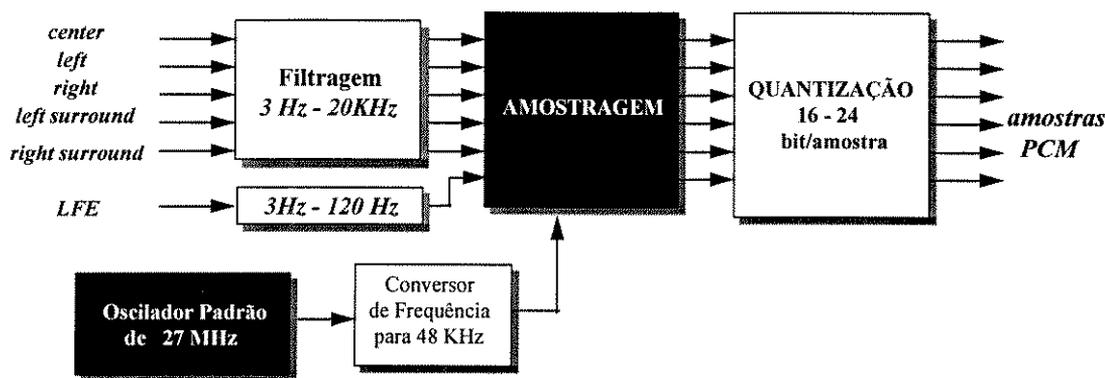


Figura. 3.2.3 Conversão Analógico-Digital dos Sinais de Áudio

3.2.3.- Formatação da Informação de Áudio

A transmissão da seqüência de bits correspondente ao sinal de áudio é feita através do formato dos *frames de sincronização* AC-3. Cada *frame* de sincronização, é formado por campos que facilitam no decodificador a recuperação da informação que está sendo transmitida [10]. A seguir vamos fazer a descrição de alguns desses campos.

3.2.3.1 .- Blocos de Áudio

Um **bloco de áudio** é formado a partir do agrupamento de 256 amostras anteriores e 256 novas amostras. Dessa forma, o número total de amostras que conformam um bloco de áudio é igual a 512 [1].

Como veremos mais adiante, cada bloco de áudio é transformado para o domínio da frequência, a fim de se ter um conhecimento do espectro do sinal num determinado instante de tempo.

Essa forma de segmentação dobra a quantidade de informação a ser processada e transmitida, mas sua vantagem está no fato de que quando o sinal é decodificado, o som reproduzido fica livre do efeito de blocos que geralmente se apresenta quando um sinal de infinitas amostras é processado com um número finito de blocos de amostras. Na Figura 3.2.11 mostra-se a formação dos blocos de áudio.

3.2.3.2 .- Frame de Sincronização

Um *frame de sincronização* é formado mediante o agrupamento de seis áudio blocos com adicionais campos de sincronização, de sinalização e de detecção de erros. Cada *frame* de sincronização representa 1536 [1] amostras de áudio codificadas e constitui a unidade de codificação e de acesso aleatório do sistema. Na Figura 3.2.4 mostra-se o formato de um *frame* de sincronização. Note que a informação de áudio formada pelos blocos de áudio AB0..AB5 vai acompanhada de informação adicional

constituída pelos campos SI, (sincronização), BSI (informação) e CRC (detecção de erro).

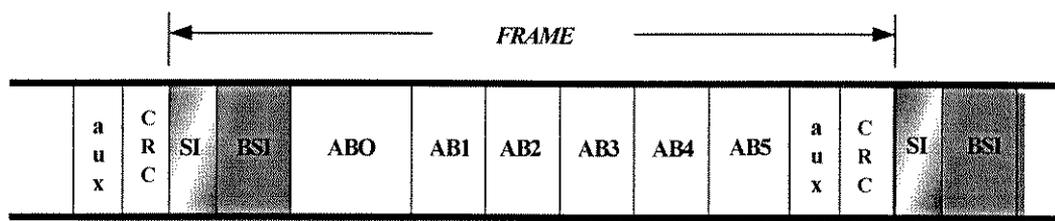


Figura. 3.2.4 Formato de uma Frame de Sincronização AC-3

3.2.3.2.1. - Campo de Sincronização (S.I.)

É o primeiro campo que se recebe no decodificador e é composto por sub-campos que contém bits de sincronização e de códigos de detecção de erros. Em seguida descreve-se cada um desses sub-campos:

- **syncword (palavra de sincronização - 16 bits)** : É uma palavra que leva sempre o valor **0B77h** ou no caso binário “0000101101110111” [10]. Quando o decodificador identifica esse código, entende que um novo *frame* está sendo recebido.
- **crc1 (16 bits)** : É um código para a detecção de erro que se aplica às primeiras 5/8 partes do *frame* sem incluir a palavra de sincronização.
- **fscod (2 bits)** : Indica a frequência de amostragem com que foi digitalizada o sinal de áudio. Na Tabela B.1 [Apêndice B] mostram-se os valores de **fscod** com a correspondente frequência de amostragem.
- **frmsizecod (6 bits)** : É um código que é usado junto com **fscod** para determinar a quantidade de palavras (2 bytes) que conformam o *frame*. Um *frame* contém sempre um número inteiro de palavras. Dessa forma, o decodificador pode saber a quantidade de informação que deve receptionar antes do início do próximo *frame*. Na Tabela B.2 [Apêndice B] mostram-se os valores **frmsizecod**, com os correspondentes tamanhos de *frame* para cada frequência de amostragem.

3.2.3.2.2. - Campo de Informação BSI (*Bit Streams Information*)

Formado por bits que indicam as características gerais da informação de áudio que está contida no *frame*. Nós vamos agora descrever os principais sub-campos de BSI à medida que avançamos em nosso estudo do processamento digital do sinal de áudio. No entanto, vamos descrever nessa parte os três primeiros sub-campos do BSI que contém a informação dos tipos de serviços que são oferecidos pelo o sistema AC-

3, bem como os canais de áudio que estão sendo codificados. Esses sub-campos são descritos em seguida:

- **bsid (5 bits)**: É um campo que tem o valor “01000” (=8) [10] e que especifica o formato do campo BSI de acordo com a versão do codificador AC-3. Dessa forma, o decodificador pode interpretar a informação segundo a versão do BSI que o codificador está transmitindo. Um decodificador AC-3 capaz de interpretar a informação na versão oito pode também decodificar versões anteriores.
- **bsmod (3 bits)** : Indicam o tipo de serviço AC-3 que está sendo codificado no *frame*. Na Tabela B.3 [Apêndice B] mostram-se os valores de bsmod para cada tipo de serviço.
- **acmod (3 bits)** : São três bits que indicam os canais de áudio que estão sendo utilizados pelo codificador. Na Tabela B.4 [Apêndice B] mostram-se os valores de acmod e os correspondentes canais de áudio. Note que quando o bit mais significativo do acmod é igual a um, indica que os canais *surround* estão sendo codificados e que o próximo campo consecutivo é o **surmixlev (2 bits)**, que especifica o nível de mistura desses canais com os outros canais do sistema. Se o bit menos significativo de acmod é igual a zero, o canal central não é utilizado, caso contrário sim é utilizado. Se o acmod tem o valor zero, então dois canais independentes Ch1 e Ch2 são codificados em formato mono [10].

A Tabela B.4 mostra também a ordem em que são codificados os respectivos canais (da esquerda para a direita). Segundo o valor acmod, o decodificador pode calcular o valor **nfchns**, que indica o número de canais que estão sendo utilizados sem considerar o canal LFE. O sub-campo **lfeon (1bit)** indica com “1” que o canal LFE está sendo utilizado e o número de canais codificados para este caso é igual a $nfchns + 1$. O sub-campo lfeon encontra-se também dentro do campo BSI.

3.2.3.2.3 Campos de Blocos de Áudio Codificados (AB0...AB5)

Conformados pela informação dos seis blocos de áudio codificados no domínio da frequência. Por exemplo o campo AB0 contém a informação dos blocos B0 de cada canal que está sendo transmitido.

Além disso, cada campo AB# está formado por bits de informação que indicam a forma como está sendo codificado o respectivo bloco. À medida que avançamos em nosso estudo do processamento do sinal de áudio, iremos fazendo a descrição dos principais sub-campos dos campos AB#.

3.2.3.2.4.- Campo de Informação para a Detecção de Erros

Constitui o último campo do *frame* e contém informação que será utilizado pelo decodificador para a detecção de erros. Dentro desse campo temos os seguintes sub-campos:

- **crersv reservado (1 bit)** : É um sub-campo opcional que é utilizado para assegurar que o **crc2** não seja igual a palavra de sincronização. Quando o **crc2** tem um valor igual a palavra de sincronização, o bit **crersv** é invertido de valor.
- **crc2 (16 bits)** : constitui a última informação contida no **frame** e que serve também para a detecção de erros dentro da informação transmitida. O **crc2** é um código de detecção de erros aplicado às 3/8 últimas partes do **frame** ou a todo o **frame** sem incluir a palavra de sincronização

Tanto o **crc1** como o **crc2** são denominados de **CRCs (Cyclic Redundancy Check)**. Para o cálculo dos CRCs é empregado o seguinte polinômio gerador [10]:

$$g(x) = x^{16} + x^{15} + x^2 + 1 \quad (3.2.3.1)$$

O circuito digital correspondente para a geração desses códigos é mostrado na Figura 3.2.5 :

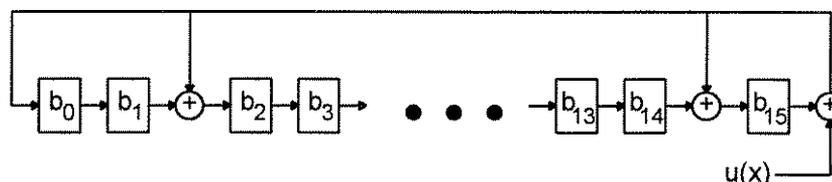


Figura. 3.2.5. Circuito Gerador de Códigos de Detecção de Erros (CRC)

Para o cálculo do **crc1**, as primeiras 5/8 partes do **frame** são passadas pelo circuito da Figura 3.2.5 através da sua entrada **u(x)**. O valor final de **crc1** é obtido através das saídas dos registradores **b#** que em seu conjunto formam os 16 bits de **crc1**.

A palavra de sincronização não é considerada para o cálculo do **crc1**, mas sim é considerada no cálculo do comprimento das primeiras 5/8 partes do **frame**. Os valores desses comprimentos são codificados de acordo com a frequência de amostragem que é utilizada e são mostrados na Tabela B.5 [Apêndice B]. O tamanho é expresso em palavras de 16 bits. O cálculo do **crc2** é similar ao de **crc1**, somente que nesse caso toda a informação contida no **frame** é passada pelo circuito da Figura 3.2.5 com exceção da palavra de sincronização.

No receptor o decodificador utiliza o mesmo circuito para a detecção de erros na seqüência de bits que é recebida. A detecção é feita ingressando as primeiras 5/8 partes do **frame** recebido e observando o valor final dos registradores. Se o valor final da cada registrador é zero, então não existiu erro na transmissão dessa parte do **frame**. Da mesma forma, se o resto do **frame** continua sendo passado através do circuito até o final, pode-se comprovar que não existiu erro se o valor final de todos os

registradores é zero. No caso em que o decodificador detecta um erro na informação recebida, o som correspondente a esse *frame* não é reproduzido [10].

Os valores de *crc1* e *crc2* calculados no codificador, devem permitir portanto que os valores finais dos registradores no decodificador sejam iguais a zero quando não existe erro na informação recebida.

3.2.3.- Análise Espectral do Sinal de Áudio

A transformação do sinal de áudio para o domínio da frequência é feita a fim de se fazer uma análise espectral do mesmo que permita a redução da taxa de bits a transmitir mediante o processo de quantização e codificação dos valores dos coeficientes. O sistema AC-3 considera as características psico-acústicas da audição humana para fixar os limites de ruído de quantização permitidos para cada componente de frequência.

3.2.3.1.- Fatores Psico-acústicos

Um das características mais importantes de nossa audição que é explorada pelo algoritmo de alocação de bits, é sua sensibilidade não uniforme às componentes de frequência que formam o espectro de audição.

Por exemplo se consideramos um oscilador que possa gerar um sinal exatamente senoidal (tom) para ser reproduzido através de um amplificador e uma caixa de som com nível constante de volume, notamos que à medida que variamos a frequência do oscilador, a percepção do ouvido aos distintos tons gerados é diferente, sendo que algumas frequências são percebidas com maior intensidade do que outras. Ainda mais, observa-se que na presença de outros sons, algumas componentes de frequências são praticamente inaudíveis.

Na Figura 3.2.6 mostramos uma família de curvas que representam os resultados do experimento anterior. Cada curva representa igual intensidade de percepção para cada componente de frequência. De acordo com isso podemos observar que para certas componentes de frequência cujo nível de percepção é baixo precisa-se aumentar o nível do amplificador, a fim de que as mesmas possam chegar com a mesma intensidade que as outras quanto à excitação de nosso ouvido. Por outro lado, em algumas componentes deve-se diminuir o nível do amplificador devido a uma maior sensibilidade do ouvido a essa componente. A intensidade do nível do amplificador em função da frequência é mostrada na Figura 3.2.6 [11]. Note que a maior sensibilidade concentra-se na faixa de 1 a 5 KHz. As curvas superiores que indicam maiores intensidades de percepção por parte do ouvido tem menor variação do nível do amplificador. Isso indica que a uniforme audibilidade das componentes dependerá também da potência que leva cada uma delas dentro do espectro. Dessa forma, é designado o adequado nível de quantização aos coeficientes dentro de uma

determinada banda utilizando-se os valores das SNR permitidas pelo ouvido para cada componente do espectro.

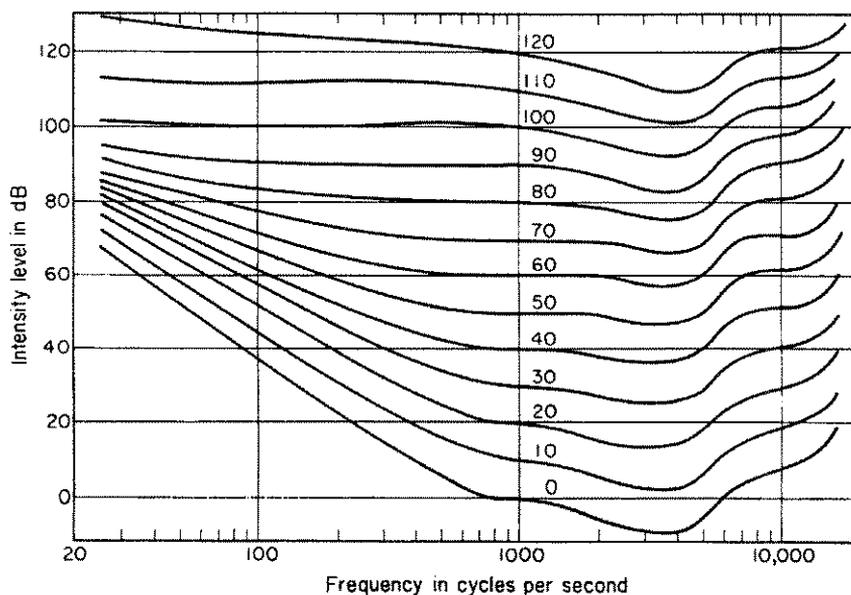


Figura 3.2.6. Percepção do ouvido humano em função da frequência.

Por outro lado, ante a presença de certas componentes de frequência existem outras de menor potência que tornam-se praticamente inaudíveis. Isso é mostrado na Figura 3.2.7, onde é feito um experimento com um tom fixo primário de 1200 Hz (80 dB acima do limiar de silêncio) e um tom secundário cuja amplitude e frequência são variáveis. Note que abaixo da curva em negrito, o tom secundário não tem efeito na percepção do tom primário, enquanto que acima da curva, os efeitos apresentam-se como uma mistura de tons, ambos tons, ou um *beat* [11].

Dessa forma, o algoritmo de alocação de bits faz a análise do espectro de um certo bloco de áudio analisando a potência de cada componente e dando maiores níveis de quantização às frequências que são audíveis e menores níveis às frequências que são praticamente inaudíveis [10].

O sistema AC-3, faz a divisão do espectro em **50 bandas** de frequências de larguras não uniforme, designando em cada processo de alocação de bits o número adequado de níveis de quantização para cada coeficiente contido em cada uma delas.

A divisão do espectro em bandas obedece à resposta de nosso sistema de audição frente à distorção que sofre o sinal devido ao processo de quantização. À medida que são consideradas bandas de maior tamanho, a distorção é mais perceptível sobretudo nas frequências menores do que 1 KHz. Por essa razão as bandas apresentadas na Tabela B.12 [Apêndice B] são mais estreitas para as componentes menores do que 1 KHz e maiores para frequências superiores a 1 KHz.

Na Tabela B.12 as bandas estão apresentadas em função dos números dos coeficientes, sendo o espaçamento entre os coeficientes consecutivos de 93.75 Hz.

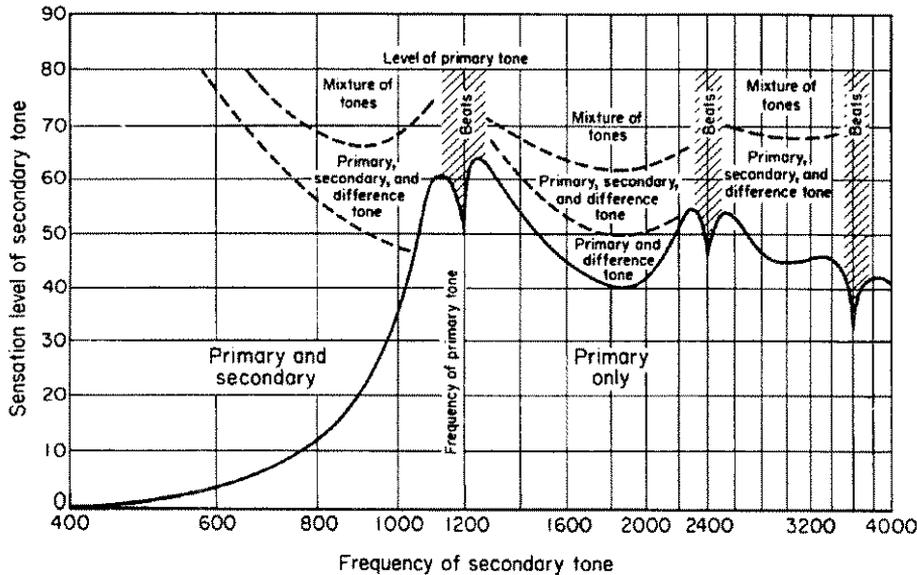


Figura 3.2.7 Percepção do ouvido humano frente a dois tons que ocorrem ao mesmo tempo.

Finalmente, outro aspecto do comportamento do ouvido que deve ser levado em conta é a resposta do mesmo ante a ocorrência de transitórios dentro do sinal de áudio. Dessa forma, pode-se inibir a percepção de outro sinal de menor intensidade que acontece antes ou depois de ocorrido o sinal primário.

Por exemplo, no caso da Figura 3.2.8, é feito um experimento com um tom de uma determinada frequência como sendo o sinal primário e um ruído branco como sendo o sinal secundário que vai ser inibido. Logo, observa-se que o sinal secundário não pode ser percebido pelo ouvido até com **20ms** antes da ocorrência do sinal primário de maior intensidade (**backward masking**).

Por outro lado, quando o tom secundário acontece depois da ocorrência do sinal primário, observa-se que este último pode inibir a percepção do secundário até depois de **250ms** (**forward masking**). A Figura 3.2.8 mostra a intensidade de inibição do sinal secundário frente ao primário versus a diferença de tempo existente entre a ocorrência dos mesmos [11].

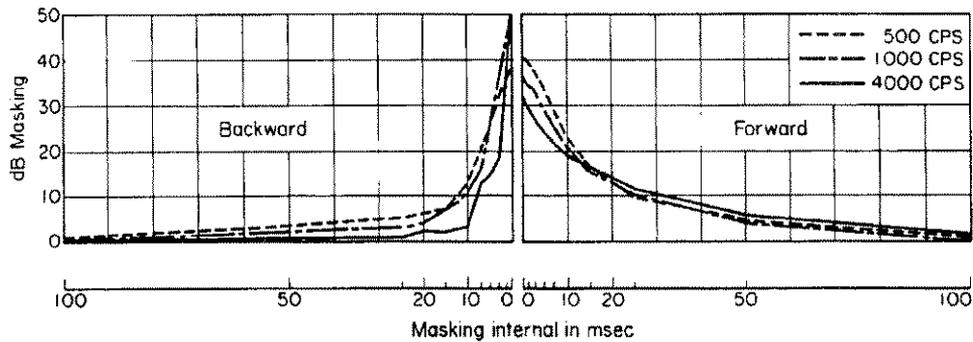


Figura 3.2.8 Percepção do Ouvido Humano frente a transitórios

3.2.3.2.- Detecção de Transitórios

Um transitório constitui um incremento da energia do sinal num curto período de tempo. Isso precisa ser considerado para a escolha do comprimento da transformada a ser aplicada a um determinado bloco de áudio.

Dessa forma, quando um transitório é detectado no segundo grupo de 256 amostras de um bloco de áudio, a transformada aplicada ao mesmo deve ser de menor tamanho do que aquela aplicada aos blocos de áudio onde não foram detectados transitórios.

Isso é feito a fim de inibir a percepção do ruído de quantização originado nas primeiras 256 amostras do bloco de áudio, mediante o fenômeno de *masking backward* explicado na seção anterior. O transitório no sinal traduzido como um incremento da energia no segundo grupo de 256 amostras permite inibir a percepção do ruído e portanto melhorar a qualidade do áudio

Por outro lado, quando um transitório não é detectado e o espectro do sinal permanece estacionário no tempo, é mais adequada a aplicação de uma transformada de comprimento longo a fim de se obter um ganho no processo de codificação e uma maior resolução do espectro no domínio da frequência. Dessa forma, o tamanho da transformada aplicada pelo sistema a um bloco de áudio de 512 amostras no domínio do tempo será de 512 amostras no domínio da frequência [10].

O sistema AC-3 possui um mecanismo de detecção de transitórios que indica o comprimento da transformada que deve ser aplicado ao sinal num determinado instante de tempo.

O processo de detecção é feito através dos seguintes passos [10] :

1. **Filtragem Passa-Altas** : O sinal original de áudio é passado através de um filtro passa-altas com frequência de corte em 8 KHz.

2. **Segmentação do Bloco** : As amostras do sinal de áudio são agrupadas de distintas formas de acordo com os níveis de hierarquia:

nível 1 : as amostras de áudio são agrupadas em blocos de 256 amostras.

nível 2 : cada bloco de 256 amostras é dividido em dois blocos de 128 amostras.

nível 3 : o bloco de 256 amostras é dividido em 4 blocos de 64 amostras.

3. **Detecção do Pico** : Nessa parte, o sistema aplica um algoritmo de detecção do nível máximo do sinal contido num determinado bloco e em seus respectivos segmentos. Para designar o valor máximo do sinal de áudio contido em cada segmento de um determinado nível é utilizado a seguinte nomenclatura :

$P[j][k] = \max(x(n))$, onde j constitui o número do nível de hierarquia, $k=1, \dots, 2j-1$, constitui o segmento k contido no nível j , e n : é a n -ésima amostra do bloco de 256 amostras

4. **Comparação com Limiar** : Aplicam-se aos distintos níveis de hierarquia através dos seguintes passos :

- Em primeiro lugar compara-se o máximo valor do sinal contido no primeiro nível com o **limiar de silêncio**. Se o valor máximo do sinal é maior que esse limiar, então indica-se que o atual bloco de áudio seja transformado em dois grupos de 256 amostras (detecção de um transitório).
- Se o valor máximo do sinal não excede esse limiar então continue-se com os seguintes níveis fazendo-se a comparação entre os níveis máximos do sinal contidos em segmentos adjacentes:

Se $(P[j][k] \times T[j]) > P[j][k-1]$ então indique-se a detecção de um transitório

O valor $T[j]$ varia de acordo com o nível j , e tem os seguintes valores:

$T[1] = 0.1$, $T[2] = 0.075$, $T[3] = 0.05$.

Note que a forma de detecção de transitórios baseia-se num incremento da energia de um segmento para outro. A indicação do transitório é feita através do sub-campo **blksw[ch]** (1 bit) localizado em cada campo **AB#** de cada *frame*. Quando **blksw** é igual a "1" indica a presença de um transitório no segundo grupo de 256 amostras de um bloco de áudio ; em caso contrário não existe transitório no atual bloco de áudio. O sub-campo **blksw[ch]** é indicado para cada canal (**ch**) que esta sendo codificado.

3.2.3.3. - Transformação para o Domínio da Freqüência (*Analysis Filter Bank*)

Como mencionamos na seção 3.2.3.1, o sistema AC-3 faz a codificação do espectro do sinal de áudio através da divisão do mesmo em sub-bandas que obedecem ao comportamento psico-acústico do nosso sistema de audição.

Esse tipo de processamento exige que a informação sobre o espectro do sinal traduzida nos valores dos coeficientes correspondentes, seja a mais precisa e seletiva possível, assegurando dessa forma a eliminação dos efeitos de *overlapping* entre sub-bandas adjacentes.

O sistema AC-3 faz a codificação do espectro do sinal aplicando-se resultados das teorias de *Short-Time Fourier Transform (STFT)* e *Time Domain Aliasing Cancellation (TDAC)* [1].

Mediante essas teorias, são originadas as amostras no domínio da frequência usando-se as transformadas do **Cosseno (DCT)** e **Seno (DST)** de **Fourier**, implementados através do sistema **Filter Bank**.

O STFT é definido pela seguinte equação:

$$X_m(e^{j\omega}) \equiv \sum_{n=-\infty}^{\infty} h(m-n)x(n)e^{-j\omega n} \quad (3.2.3.1)$$

onde $x(m)$ representa as amostras no tempo do sinal de áudio, e $h(n-m)$ representa a função de janelamento (**windowing**) que determina a porção de $x(m)$ a ser transformada.

Para o caso do *analysis filter bank* é fixada uma frequência ω_0 na equação 3.2.3.1 e o resultado é interpretado como a resposta de um filtro $h(n)$ excitado por um sinal modulado na frequência ω_0 . Na seguinte equação é dada a expressão correspondente ao *analysis filter bank* [13] :

$$X_m(e^{j\omega_0}) = h(m) * [x(m)e^{j\omega_0 m}] \quad (3.2.3.2)$$

onde $h(m)$ fixa a largura de faixa do filtro centrado na frequência ω_0 .

A geração do sinal original a partir dos coeficientes $X_m(e^{j\omega})$ da equação 3.2.3.1 é dada pela seguinte expressão [13]:

$$\hat{x}(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \sum_{r=-\infty}^{\infty} f(n-r)X_r(e^{j\omega})e^{j\omega n} d\omega \quad (3.2.3.3)$$

onde $f(n)$ é uma seqüência denominada *synthesis window* que é assumida junto com $h(n)$ como filtros de resposta impulsiva finita FIR .

Na Figura 3.2.9 mostra-se a implementação dos *analysis /synthesis filter bank* [12]. Note que a transmissão apenas da metade dos coeficientes é suficiente para a geração do sinal original no receptor. As faixas geradas a partir do sistema da Figura 3.2.9 são mostradas na Figura 3.2.10.

A sub-amostragem dos coeficientes $X_m(e^{j\omega})$ por um fator M , representa os períodos de geração de um novo bloco de áudio. Logo, tem-se : $M=256$. A totalidade

de redundância do sinal é retirada quando $M = K/2$, onde K constitui o número total de amostras no domínio da frequência. Para o caso do sistema AC-3, tem-se $K=512$.

Cada bloco de áudio está identificado por um número inteiro “m”, produto da sub-amostragem dos coeficientes $X_m(e^{j\omega})$. A janela $h(n)$ determina o tamanho do bloco de áudio, que para o caso do sistema AC-3 é de 512 amostras, das quais as 256 primeiras pertencem ao bloco de áudio anterior, e as últimas 256 constituem novas amostras. De acordo com a Figura 3.2.9, os blocos de áudio pares são transformados através da DCT, enquanto os ímpares são transformados através da DST [12]. Isso é devido aos seguintes fatores:

$$\sin\left(\frac{m\pi}{2}\right) = 0 \quad m: \text{par}$$

$$\cos\left(\frac{m\pi}{2}\right) = 0 \quad m: \text{ímpar}$$

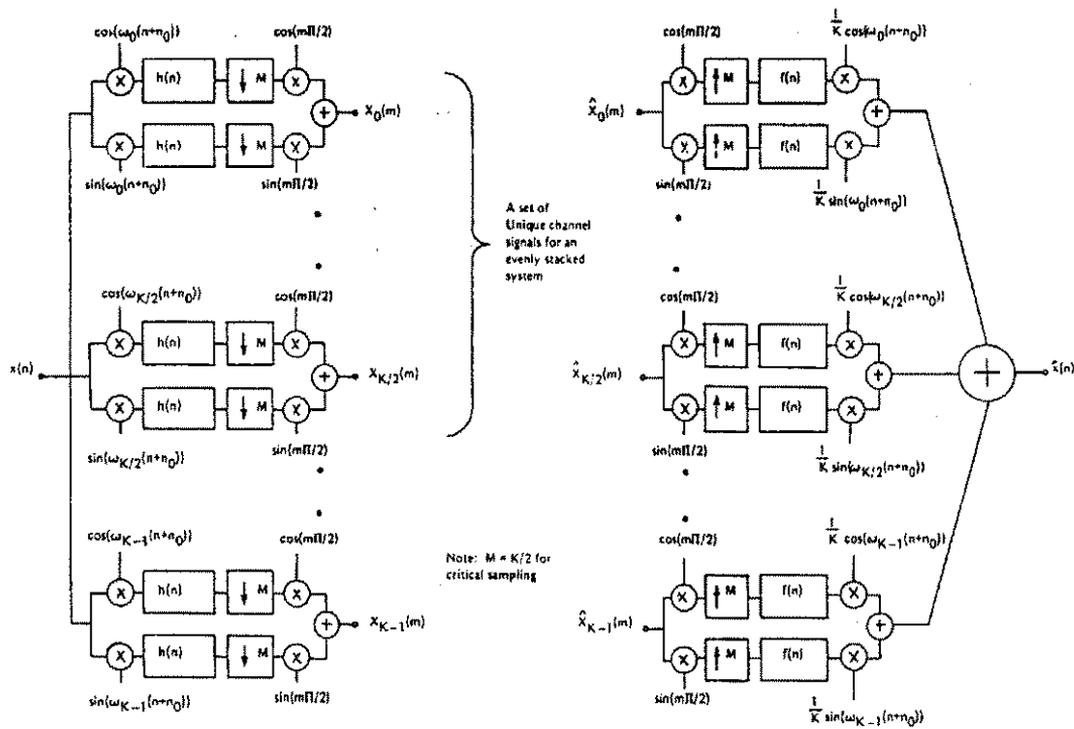


Figura 3.2.9 Implementação de um sistema analysis/synthesis Filter Bank

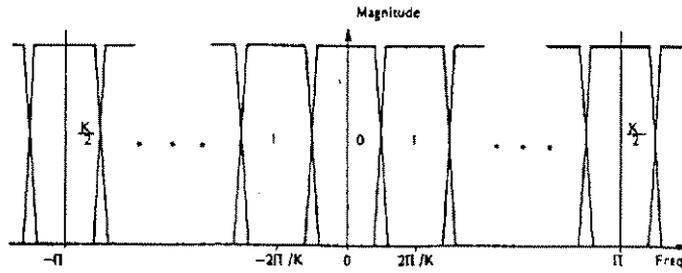


Figura 3.2.10. Banco Real de Filtros Passa-Faixa

De acordo com a Figura 3.2.9 a transformada do cosseno a se aplicar em um determinado bloco de áudio é dada pela seguinte equação:

$$X(k) = \sum_{n=0}^{K-1} x(n) \cos\left(\frac{2\pi k}{K}(n+n_0)\right) \quad (3.2.3.4)$$

$$k = 0 \dots K-1$$

A transformada inversa é dada por :

$$\overset{\Delta}{x}(n) = \sum_{k=0}^{K-1} X(k) \cos\left(\frac{2\pi k}{K}(n+n_0)\right) \quad (3.2.3.5)$$

$$k = 0 \dots K-1$$

Se substituirmos 3.2.3.4 em 3.2.3.5 e aplicamos algumas propriedades trigonométricas, obtemos a seguinte expressão :

$$\overset{\Delta}{x}(n) = \frac{x(n)}{2} + \frac{x(K-n-2n_0)}{2} \quad (3.2.3.6)$$

onde observamos que o sinal recuperado no receptor não corresponde ao sinal original, apresentando um *overlap-add* entre os dois grupos de 256 amostras que conformam neste caso um bloco de áudio ($K=512$) (assumindo o fator $n_0=0$). No entanto, se consideramos outros valores do fator de fase “ n_0 ” podemos ajustar o tipo de *overlap* da forma mais conveniente.

Para o caso da DST o sinal recuperado apresenta a seguinte expressão:

$$\overset{\Delta}{x}(n) = \frac{x(n)}{2} - \frac{x(K-n-2n_0)}{2} \quad (3.2.3.7)$$

Observando 3.2.3.6 e 3.2.3.7 notamos que o *overlap* apresentado em ambas as expressões é de sinal contrário. Esse fato é aproveitado para cancelar a distorção apresentada no sinal. Isso implica que os tipos de *overlap* apresentados nas últimas

256 amostras de um bloco de áudio e nas primeiras 256 amostras do bloco de áudio consecutivo têm que ser iguais e de sinal contrário.

A multiplexação DCT/DST entre blocos de áudio adjacentes permite que o *overlap* apresentado em cada um deles seja de sinal contrário. Por outro lado, um adequado ajuste do fator de fase n_0 permite a obtenção do mesmo tipo de *overlap* entre o segundo grupo de 256 amostras de um determinado bloco de áudio e o primeiro grupo de 256 amostras do bloco de áudio consecutivo. A eliminação ou cancelamento do *overlap* poderá ser feita fazendo-se a soma entre esses grupo de amostras. Esse processo implica no armazenamento das amostras do primeiro bloco de áudio até que o próximo bloco de áudio consecutivo seja decodificado [12]. Na Figura 3.2.11 é mostrado o mecanismo de cancelamento de *aliasing* segundo explicado anteriormente.

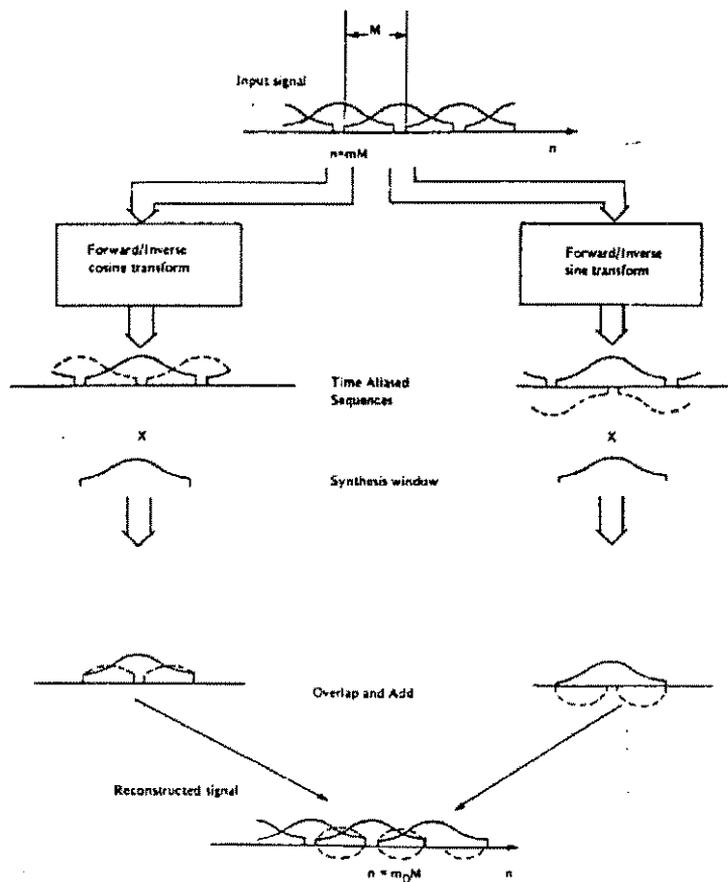


Figura 3.2.11 Mecanismo de Cancelamento de Aliasing no domínio do Tempo

O sistema Dolby AC-3 emprega as seguintes transformadas para a geração dos coeficientes [10] :

$$X(k) = \frac{-2}{K} \sum_{n=0}^{K-1} x_J(n) \cos\left(\frac{2\pi}{4K} (2n+1+2[(K/4)(\alpha+1)])(2k+1)\right) \quad (3.2.3.8)$$

$$k = 0 \dots K/2$$

$$X(k) = \frac{-2}{K} \sum_{n=0}^{K-1} x_j(n) \operatorname{sen}\left(\frac{2\pi}{4K}(2n+1+2[(K/4)(\alpha+1)])(2k+1)\right) \quad (3.2.3.9)$$

$$k = 0 \dots K/2$$

onde $x_j(n)$ representa as amostras passadas no tempo através da janela da Figura 3.2.10, $n_0 = (K/4)(1+\alpha)$, e α pode assumir os valores de -1, 0 e +1. Quando $\alpha=0$ é porque não tem sido detectado um transitório dentro do bloco de áudio que está sendo transformado. Para esse caso, $n_0=K/2$ e $K=512$, pois as 512 amostras de bloco de áudio vão ser transformadas num único conjunto de 512 amostras em frequência. Pela simetria mostrada na figura 3.2.10, o número de coeficientes ou amostras em frequência transmitido é de 256, representando frequências de até 24 KHz.

Quando um transitório tem sido detectado no atual bloco de áudio, então a transformação do mesmo é feita em dois grupos. No primeiro grupo são transformadas as primeiras 256 amostras desse bloco de áudio para o qual $\alpha=-1$, $K=256$ e $n_0=0$. Para o segundo grupo o valor de $\alpha=+1$ e $n_0=K/2$.

Essa forma de processamento é feita a fim de se aproveitar o fenômeno psicoacústico de *masking backward* explicado anteriormente. Portanto o ruído gerado na quantização dos coeficientes do primeiro grupo de amostras ($\alpha=-1$), pode ser inibido pelo transitório ocorrido no segundo grupo de amostras desse bloco de áudio.

Igualmente por simetria, são gerados para este caso 128 únicos coeficientes os quais são intercalados com os 128 coeficientes do segundo grupo para se formar um bloco de 256 coeficientes que serão codificados da mesma forma que os blocos onde não aconteceram transitórios. Os coeficientes gerados em cada grupo representam frequências de até 24 KHz [10].

Uma última consideração que tem que ser levada em conta é o tipo de janela a se utilizar. Além de gerar blocos finitos de amostras, uma janela determina a seletividade das bandas do espectro que conformam o sinal de áudio. A distorção apresentada entre faixas adjacentes apresenta-se como um *aliasing* entre as mesmas. Essa sobreposição de espectros é eliminada mediante o processo de geração dos blocos de áudio compartilhando amostras e simultaneamente usando-se o processo de janelamento. Na Figura 3.2.12 é apresentado a distorção entre as faixas do espectro.

Note que quanto maior for o *overlap* entre os blocos de áudio adjacentes consegue-se uma maior seletividade entre as faixas. Os blocos de áudio do sistema AC-3 são formados mediante o *overlap* 2:1, significando que dois blocos de áudio adjacentes compartilham a metade das amostras.

Na Figura 3.2.12 observa-se que apesar do *overlap* 2:1 entre dois áudio blocos no domínio do tempo, a distorção entre faixas persiste. Essa distorção pode ser quase eliminada empregando-se um janelamento adequado para se obter uma aceitável atenuação fora da faixa. O sistema AC-3 emprega a janela mostrada na Figura 3.2.13 cujos valores são dados no “*Digital Áudio Compression AC-3 Standard*” e que são apresentados na Tabela B.6 [Apêndice B]. Nessa Tabela relacionam-se os primeiros

256 valores da janela, sendo os 256 últimos valores iguais aos primeiros somente que com numeração invertida para se formar a janela simétrica de 512 valores mostrada na Figura 3.2.13.

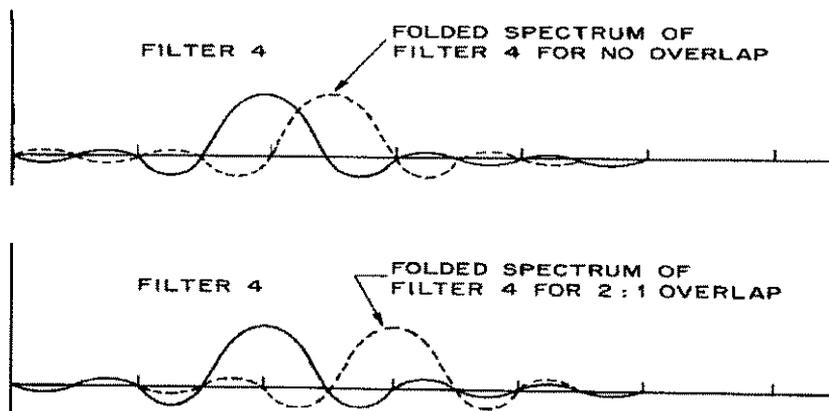


Figura. 3.2.12 Distorção do Espectro pelo efeito de Bloco

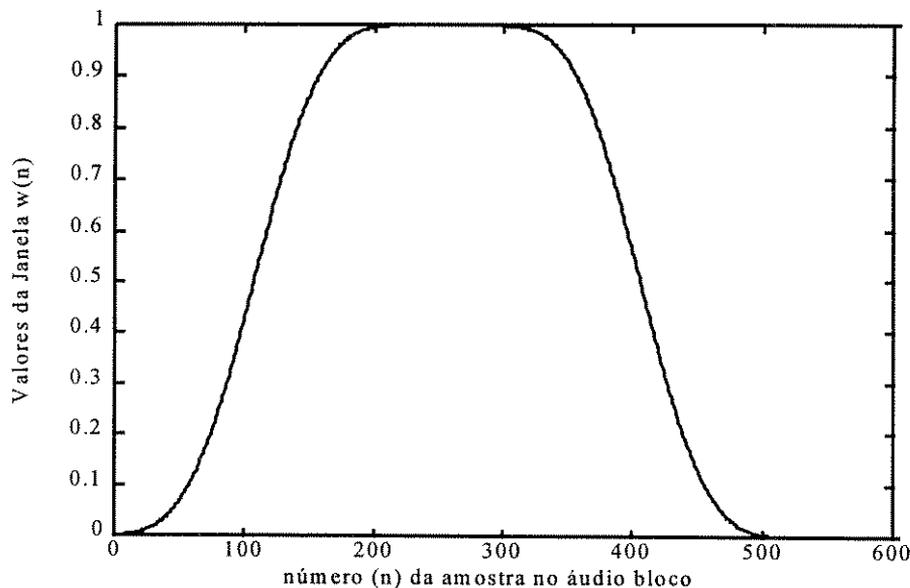


Figura. 3.2.13. Janela aplicada ao sinal de áudio pelo sistema AC-3

O processo de codificação empregado pelo sistema AC-3 é também conhecido como *codificação perceptual adaptativa* devido a que a informação é codificada no domínio da frequência baseado nas características espectrais do sinal bem como num modelo psico-acústico do ouvido humano.

Os resultados de pesquisas realizadas por **Zelinski** e **Noll** [13] sobre o tipo de transformada aplicado a sinais de voz e áudio indicam a preferência da DCT sobre a DFT. Isso é devido ao fato de que a primeira apresenta um desempenho de 4 a 5 dB de

SNR (relação sinal a ruído de quantização dos coeficientes) sobre a segunda para blocos de 128 amostras, sendo que esta diferença diminui à medida que aumenta o tamanho do bloco que vai ser transformado. Por outro lado, igual ao que acontece no processamento de imagens, a transformada ideal que apresenta muitas vantagens no tratamento de sinais é a transformada de *Karhunen-Loeve*, mas por complexidade na sua implementação dá-se uma maior preferência à DCT. De fato, a DCT é implementável na forma de circuitos integrados.

Finalmente, podemos dizer que tanto a DCT como a DST podem ser implementadas através dos algoritmos da FFT (*Fast Fourier Transformer*) permitindo dar ao sistema uma maior eficiência computacional. As variações realizadas na DCT para adequar os requisitos do sistema, geram o que se denomina a MDCT (*Modified Discrete Cosine Transform*) [10] a qual gera os coeficientes MDCT representados pelas equações 3.2.3.8 e 3.2.3.9. Na figura 3.2.14 apresenta-se o processo de geração dos coeficientes MDCT para a formação de um *frame*.

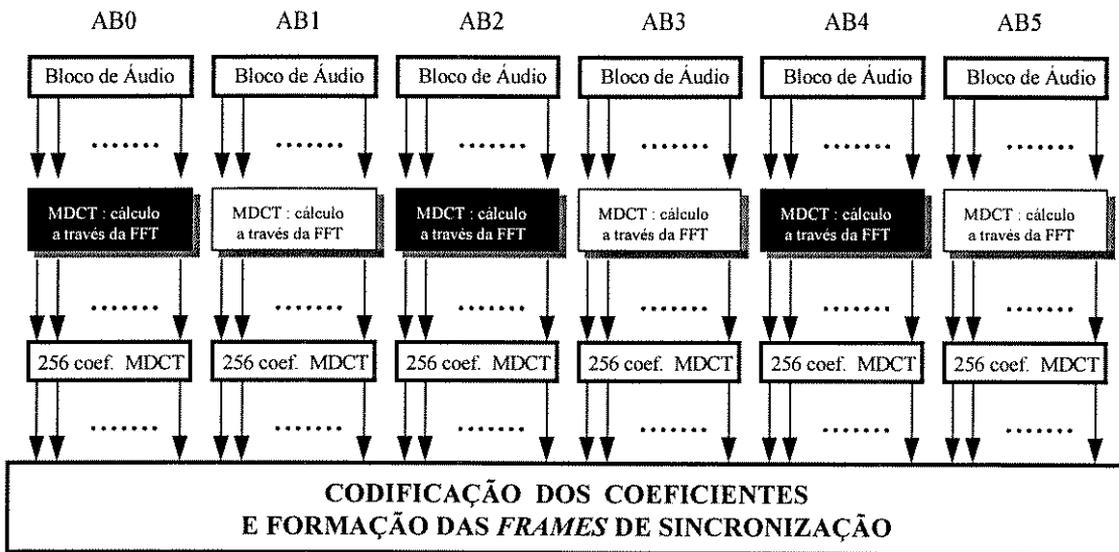


Figura. 3.2.14. Processo de aplicação da transformada a cada bloco de áudio

3.2.4.- Codificação dos Coeficientes MDCT

Depois que o sistema obtém a representação do sinal de áudio no domínio da frequência, cada valor dos coeficientes está representado por números reais cujos valores são escalonados para serem menores do que 1 [1]. O formato empregado para a representação desse tipo de números é a notação em ponto flutuante que mostramos em seguida:

Um coeficiente que tem o valor : **0.1045675** é representado no sistema binário pelo seguinte número com precisão de 16 bits:

0.0001101011000100

Portanto, a representação em ponto flutuante será feita da seguinte forma:

$$2^{-3} \times 0.1101011000100$$

O número de zeros depois do ponto decimal na primeira expressão representa o valor do expoente que neste caso é 3, e os 13 bits consecutivos (1101011000100) depois do ponto decimal na segunda expressão representam o valor da mantissa normalizada que será posteriormente quantizada e codificada junto com o expoente.

Como um exemplo, o formato em ponto flutuante para este caso é o seguinte:



Se o bit de sinal tem valor 1, o valor do coeficiente é negativo, caso contrário é positivo.

3.2.4.1 Codificação dos Expoentes

Cada bloco de áudio contido num *frame* está formado por um **conjunto de expoentes** e seu correspondente **conjunto de mantissas**. Dessa forma, um *frame de sincronização* é formado por seis conjuntos de expoentes e seis conjuntos de mantissas [1].

Para a codificação dos conjuntos de expoentes (contidos num *frame*), o sistema explora o grau de diferença que existe entre eles a fim de se aplicar os seguintes critérios [1]:

- Se a diferença entre os conjuntos é pequena então apenas um conjunto é transmitido no primeiro bloco de áudio (campo AB0), e reutilizado pelos outros blocos de áudio que pertencem a um determinado canal.
- Se a diferença é grande, então apenas um conjunto de expoentes é transmitido para blocos onde a diferença é pequena, de tal forma que grupos de blocos compartilham o mesmo conjunto de expoentes.
- Cada conjunto de expoentes é codificado explorando-se a vantagem de que a diferença entre expoentes de frequências adjacentes é de no máximo ± 2 ($\pm 12\text{dB}$). Dessa forma, o sistema transmite completamente o primeiro valor de cada conjunto e os seguintes valores consecutivos em forma diferencial. Cada expoente é codificado com valores de 5 bits, a fim de permitir que todos os possíveis valores (0...24) dos mesmos possam ser codificados. O primeiro expoente de cada conjunto é codificado com 4 bits, pois um deles é considerado com um valor fixo e não transmitido. Para o caso de canais independentes, o valor limite para a

codificação do primeiro expoente é de 15 (4 bits) considerando o bit mais significativo (mas não transmitido) igual a zero. No caso de canais acoplados (canais formados pelo acoplamento de outros canais em certas faixas de frequências) o bit menos significativo é considerado sempre zero e portanto podem ser transmitidos os valores pares do intervalo 0,2,4,.....24.

As estratégias de codificação diferencial para cada conjunto são descritas em seguida:

- **Estratégia D15**

A estratégia D15, codifica a diferença entre expoentes individuais de frequências adjacentes [10]. O mapeamento dos valores diferença é mostrado na Tabela B.7 [Apêndice B].

Para todos os casos, os valores mapeados obedecem às seguintes equações :

$$\text{Valor Mapeado} = \text{Dif Exp} + 2 \quad (3.2.4.1)$$

$$\text{Dif.Exp.} = \text{Valor mapeado} - 2 \quad (3.2.4.2)$$

$$\text{Exp}[n] = \text{Exp}[n-1] + \text{DifExp.}[n] \quad (3.2.4.3)$$

- **Estratégia D25**

A estratégia D25, codifica a diferença entre cada par de expoentes consecutivos, onde o codificador considera ao segundo expoente de cada par igual ao primeiro [10]. Na Tabela B.8 [Apêndice B] mostra-se o mapeamento dos valores diferença.

- **Estratégia D45**

A estratégia D45 codifica a diferença entre grupos consecutivos de 4 coeficientes, onde o codificador considera os 3 últimos expoentes de cada par iguais ao primeiro [10]. Na Tabela B.9 [Apêndice B] mostra-se o mapeamento dos valores diferença.

A indicação do tipo de estratégia utilizada pelo codificador, é feita através dos sub-campos **chespstr[ch]** localizados nos campos AB# de cada *frame*. Para o caso do canal LFE somente é utilizada a estratégia D15 e é indicado pelos sub-campos **lfeexpstr[ch]** localizados em cada campo AB#. O mapeamento de chespstr[ch] e lfeexpstr[ch] é mostrado nas Tabela B.10 e B.11 [Apêndice B] , respectivamente.

Os critérios para a escolha da estratégia de codificação são os seguintes [10]:

- Se um conjunto de expoentes vai ser utilizado num único bloco de áudio, então utiliza-se a estratégia D45.
- Se um conjunto de expoentes é utilizado por dois ou três blocos de áudio, então utiliza-se a estratégia D25.
- Finalmente, se um conjunto de expoentes é utilizado por cinco ou seis blocos de áudio então utiliza-se a estratégia D15.

Dentro de cada conjunto, o codificador forma **grupos** de 3 valores mapeados consecutivos (M1,M2,M3) , os quais são codificados por valores de 7 bits tal como é mostrado a seguir :

$$\text{Valor de Grupo (7 bits)} = (25 \times M1) + (5 \times M2) + M3 \quad (3.2.4.4)$$

Cada valor de grupo é representado nos sub-campos **exp[ch][grp]** (7bits) que estão contidos nos campos AB# de cada *frame*. O primeiro valor de cada conjunto constitui o valor real do expoente DC e por tanto não é mapeado nem agrupado na codificação de 7 bits [10]. A decodificação dos valores mapeados dos diferentes grupos é feita empregando-se as seguintes equações :

$$M1 = \text{int}(\text{exp}[\text{ch}][\text{grp}] / 25) \quad (3.2.4.5)$$

$$M2 = \text{int}((\text{exp}[\text{ch}][\text{grp}] - (M1 \times 25)) / 5) \quad (3.2.4.6)$$

$$M3 = \text{exp}[\text{ch}][\text{grp}] - (5 \times M2) \quad (3.2.4.7)$$

onde a função $\text{int}(x)$ devolve o valor inteiro de “x”.

Finalmente, podemos dizer que a taxa de compressão resultante devido à codificação dos expoentes é de **0.39bit/expoente** [1]. Outra coisa que tem que ser tomada em conta é que o sinal de entrada no codificador tem componentes de frequência de até 20 KHz. No entanto, somente são percebidas pelo ouvido humano as frequências até 15 KHz. Assim, os expoentes que representam componentes acima dessa frequência podem não ser codificados reduzindo a compressão a um valor aproximadamente menor do que **0.3bit/expoente** [1].

O sub-campo **chbwcod[ch]** (6 bits) localizado em cada campo AB# do *frame* indica através de um número inteiro (0-60), o limiar superior da faixa até onde são codificados os expoentes de um determinado canal. Se um valor maior que 60 é recebido então o decodificador não produz som.

3.2.4.2.- Processo de Alocação de Bits

O processo de alocação de bits consiste num algoritmo que tem como finalidade a determinação do número de níveis de quantização necessário para a codificação de cada mantissa [10].

Dentro da teoria de codificação de sinais de voz e áudio no domínio da frequência, existe um grande quantidade de algoritmos que fazem o cálculo do número de bits necessários para a representação do valor de cada coeficiente ou mantissa.

Na documentação apresentada pelo ATSC para o sistema AC-3, não se especifica exatamente o algoritmo de alocação de bits empregado, mas é definida claramente a seqüência da rotina que vamos descrever a seguir:

- **Cálculo da densidade espectral de potência (PSD)**

Nesse procedimento, o codificador calcula a densidade espectral de potência PSD com base nos valores dos expoentes de cada coeficiente. O cálculo é feito logaritmicamente. A função **PSD(k)** resultante serve como referência do nível de intensidade que apresentam as componentes do espectro de um determinado bloco de áudio [10].

- **Integração da PSD(k) por banda**

A partir dos valores PSD[k] obtidos no procedimento anterior, o sistema faz a integração dos mesmos em cada banda de frequência mostrada na Tabela B.12 [Apêndice B]. Os valores resultantes são definidos como **banpsd(k)** que dão uma referência da potência contida em cada uma dessas bandas [10].

- **Cálculo da função de Excitação**

A função de excitação estima os limiares de ruído permitidos para cada componente de frequência, segundo um modelo paramétrico que é ajustado de acordo com a forma do espectro. Os parâmetros definem a forma da função de excitação que é ajustada de acordo com a forma da função banpsd(k) calculada no procedimento anterior. Os parâmetros são ajustados continuamente até se alcançar uma estimativa ótima da função. Os valores finais dos parâmetros são transmitidos para o decodificador a fim de que a rotina seja executada apenas uma vez no receptor [10].

O nível de ruído estimado para cada componente de frequência numa mesma faixa é o mesmo. Essa forma de processamento permite obter um sinal no receptor com uma distorção mínima devido ao fato de que o ruído ajusta-se as características espectrais do sinal. Por outro lado, como se mencionou anteriormente sobre a particular divisão não uniforme do espectro em bandas, isso permite explorar a

característica do ouvido humano que recebe a maior quantidade de informação através das baixas frequências, as quais são quantizadas através de bandas de menor largura, tal como é definido na Tabela B.12. O resultado da curva de excitação é expresso no algoritmo mediante a função **excite(k)**.

- **Cálculo da *masking curve***

Depois que os valores da função de excitação são calculados, faz-se uma comparação entre os níveis de ruído especificados pela mesma com os limiares de ruído permitidos pelo ouvido humano para cada componente de frequência. Depois da comparação, é escolhido o máximo valor a fim de diminuir o número de níveis de quantização especificados para cada mantissa. Os valores finais da *masking curve* são dados para cada mantissa através do término **mask(k)** [10].

Os valores do ruído permitido pelo ouvido humano que são utilizados dentro do algoritmo de quantização são mostrados na Tabela B.13 [Apêndice B]. Esses valores são interpretados pelo algoritmo como sendo a referência para se aplicar o adequado número de níveis de quantização para cada mantissa segundo a sensibilidade do ouvido a uma determinada banda ou componente de frequência.

- **Cálculo dos ponteiros de alocação de bits (**bap**)**

Nessa parte final da rotina, o codificador calcula o tipo de quantizador designado para cada coeficiente. O cálculo é feito restando o valor correspondente da *masking curve* **mask(k)** dos valores **PSD(k)**. Logo o resultado obtido dessa diferença é deslocado para a direita em 5 bits e o valor resultante é tomado como o endereço do ponteiro **bap** tal como mostra a Tabela B.14 [Apêndice B] [10]. Note que esse procedimento equivale ao cálculo da SNR(k) em dB. Dessa forma, quanto maior for a SNR de um determinado coeficiente maior será o número de níveis de quantização designado para a mantissa correspondente.

A soma do número de bits alocados para cada mantissa deve ser **menor ou igual que o total de bits disponíveis para a codificação de um determinado frame** [10]. Dessa forma, o sistema pode fazer o ajuste da *masking curve* através dos valores **csnroffst (6 bits)** e **fsnroffst[ch] (4bits)** (disponíveis no algoritmo) que são transmitidos ao decodificador. O valor **csnroffst** indica ganhos múltiplos de ± 6 dB no nível de ruído para todas as bandas da curva de referência. O valor **fsnroffst** indica ganhos de ruído múltiplos de $\pm 3/8$ dB para todas as bandas da *masking curve* de um determinado canal (**ch**). Quando o **csnroffst** indica um ganho de 6dB, isso corresponde a um incremento de um bit na resolução dos valores de cada mantissa pois isso está indicando um menor nível de ruído para todos os coeficientes do espectro.

Os resultados finais da rotina são portanto os valores **csnroffst**, **fsnroffst** e os ponteiros **bap**. Finalmente, depois que esses resultados são obtidos, o sistema tem a possibilidade de aplicar um segundo modelo psico-acústico de maior desempenho que

pode melhorar a qualidade do som no receptor mediante a inserção de bits no *frame*. Esses bits adicionais indicam um ajuste na *masking curve* de referência quando o processo de decodificação está sendo realizado [1]. Esse processo denomina-se alocação *delta* de bits e indica ajustes de ± 6 dB na *masking curve* a fim de que o som reproduzido no receptor seja de melhor qualidade.

3.2.4.3.- Processo de Quantização de Mantissas

Uma vez calculados os ponteiros *bap* para cada componente de frequência, o sistema faz a quantização do valor de cada mantissa cujo número de níveis de quantização é especificado pelo correspondente *bap* [1].

O número de níveis de quantização pode variar de 0 até 65536 o que corresponde a uma precisão de 0 a 16 bits na representação do valor de cada mantissa [10]. Na Tabela B.15 [Apêndice B] mostra-se o número de níveis de quantização designados para cada ponteiro *bap*. A forma de quantização pode ser feita de maneira simétrica ou assimétrica de acordo com o número de níveis de quantização especificado para cada mantissa. Na Tabela B.16 [Apêndice B] mostramos o mapeamento do uso de cada tipo de quantização. A seguir faremos uma descrição dos tipos de quantização usados pelo sistema.

- **Quantização Assimétrica**

A quantização assimétrica é considerada quando os valores do ponteiro *bap* estão no intervalo $6 \leq \mathbf{bap} \leq 15$ e denomina-se assimétrica devido ao fato de que o número de níveis de quantização para valores positivos e negativos de cada mantissa não é o mesmo [10]. De acordo com isso os valores quantizados de cada mantissa vão a variar de : $(1.0 - 2^{\mathbf{qntz}z\mathbf{b}[\mathbf{bap}]})$ a -1.0 [10], sendo os valores negativos quantizados com um nível a mais do que os valores positivos.

- **Quantização Simétrica**

A quantização simétrica é considerada quando os valores do ponteiro *bap* estão no intervalo $1 \leq \mathbf{bap} \leq 5$ [10]. Esse tipo de quantização é caracterizado por considerar o mesmo número de níveis de quantização para os valores positivos e negativos das mantissas. Nas Tabelas B.17, B.18, B.20, B.21 e B.22 [Apêndice B] mostramos o mapeamento dos valores de cada mantissa e o seus correspondentes códigos para valores distintos de *bap*. A quantização simétrica é considerada devido ao fato de que para poucos níveis de quantização, a precisão dos valores da mantissa não constitui um fator que afete seriamente a qualidade de som que será reproduzido no receptor.

Finalmente , a fim de salvar alguns bits, o sistema faz a codificação dos valores designados a cada mantissa para os casos onde o *bap* é igual a 1, 2 ou 4. No caso em que o *bap* é igual a um (três níveis de quantização) três mantissas consecutivas são agrupadas em um código de 5 bits de comprimento. Para o caso

onde bap é igual a dois, três mantissas consecutivas são agrupadas em códigos com 7 bits de comprimento. Finalmente, no caso em que o $bap=4$, duas mantissas consecutivas são agrupadas num código com 7 bits de comprimento.

A codificação de grupo é mostrada em seguida [10]:

$bap = 1:$

$$\text{código-grupo} = 9x(\text{mantissa código}[a]) + 3x(\text{mantissa código}[b]) + \text{mantissa código}[c]$$

$bap = 2:$

$$\text{código-grupo} = 25 \times \text{mantissa código}[a] + 5 \times \text{mantissa código}[b] + \text{mantissa código}[c]$$

$bap = 4:$

$$\text{código-grupo} = 11 \times \text{mantissa código}[a] + \text{mantissa código}[b]$$

O processo de decodificação é o seguinte :

$bap=1$

$$\text{mantissa código}[a] = \text{int} (\text{código-grupo} / 9)$$

$$\text{mantissa código}[b] = \text{int} ((\text{código-grupo} \times (9/100)) / 3)$$

$$\text{mantissa código}[c] = (\text{código-grupo} \times (9/100)) \times (3/100)$$

$bap=2$

$$\text{mantissa código}[a] = \text{int} (\text{código-grupo} / 25)$$

$$\text{mantissa código}[b] = \text{int} ((\text{código-grupo} \times (25/100)) / 5)$$

$$\text{mantissa código}[c] = (\text{código-grupo} \times (25/100)) \times (5/100)$$

$bap=4$

$$\text{mantissa código}[a] = \text{int} (\text{código de grupo} / 11)$$

$$\text{mantissa código}[b] = \text{código de grupo} \times (11/100)$$

Finalmente, na Figura 3.2.15, apresentamos todo o processo de codificação dos coeficientes MDCT empregado pelo sistema de compressão *Dolby AC-3*.

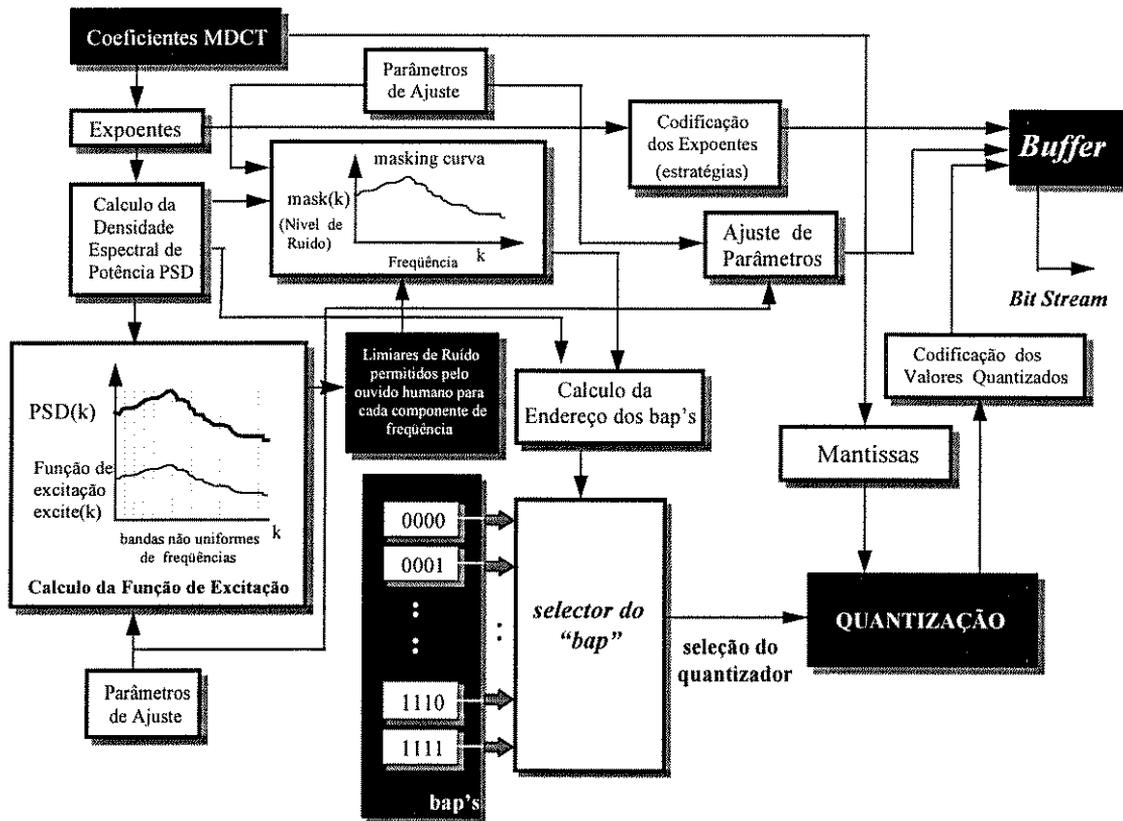


Figura. 3.2.15 Processo de Codificação dos Coeficientes MDCT

3.2.5.- Acoplamento de Canais

Muitas vezes acontece que o número de bits necessário para codificar um sinal de áudio ultrapassa o limiar permitido pelo codificador. Para dar solução a isso, o sistema AC-3 forma os canais acoplados a fim de que a informação contida em vários canais de áudio independentes possa ser transmitida através de um só canal comum [1].

Geralmente um canal acoplado pode ser formado somando-se os valores dos coeficientes de cada canal que entra no acoplamento, e dividindo por oito para assegurar que os valores do novo canal continuem sendo menores do que 1 [10].

Dessa forma, para certas faixas de frequência o sistema transmite apenas a informação do canal acoplado em lugar da informação contida em cada canal de áudio independente.

O acoplamento é feito a partir de sub-faixas que contém componentes de alta frequência. Isso é devido a que o ouvido humano detecta a localização das altas frequências em faixas estreitas a partir da envolvente do espectro onde tem pouca sensibilidade ao valor exato dos coeficientes [1].

As sub-faixas de frequências que podem entrar no acoplamento de canal (ch) de áudio são mostradas na Tabela B.22 [Apêndice B].

O intervalo das sub-faixas de acoplamento são especificadas pelos sub-campos **cplbegf (4 bits)** e **cplendf (4 bits)** localizados em cada campo AB# do *frame* de sincronização [10].

Dentro de cada sub-campo cplbegf e cplendf, especifica-se o número da sub-faixa onde começa e termina o acoplamento. Para obter o valor real da última sub-faixa soma-se dois ao valor cplendf e dessa forma o número total de sub-faixas será dado pela seguinte equação[10] :

$$\text{núm de sub-faixas de acoplam. (ncplsubnd)} = 3 + \text{cplendf} - \text{cplbegf} \quad (3.2.5.1)$$

3.2.5.1.- Coordenadas de Acoplamento

As coordenadas de acoplamento são formadas tomando-se a razão da potência contida numa faixa do canal original com a potência contida na mesma faixa no canal de acoplamento [10]. Dessa forma, o decodificador pode formar os canais que foram acoplados a partir das coordenadas e da informação contida no canal comum. As faixas de acoplamento são formadas a partir das sub-faixas mostradas na Tabela B.22..

Uma faixa de acoplamento pode estar formada por várias sub-faixas de acordo com os valores dos sub-campos **cplbndstrc[subfaixa] (1bit)** contidos em cada campo AB# do *frame*. Um sub-campo cplbndstrc[subfaixa] é transmitido para cada sub-faixa de acoplamento e obedece a seguinte regra: se cplbndstrc é “1”então a sub-faixa atual é juntada com as anteriores para formar uma faixa de acoplamento. No caso em que o valor de cplbndstrc[subfaixa] seja “0” a sub-faixa atual não é ajuntada com as anteriores e portanto constitui o início de uma nova faixa. O cplbndstrc correspondente a primeira sub-faixa de acoplamento não é transmitido pois necessariamente tem que ser “0”.

As coordenadas para cada canal são convertidas ao formato de ponto flutuante e logo em seguida quantizadas. Cada expoente e cada mantisa das coordenadas são expresos em 4 bits através dos campos **cplcoexp[canal][faixa]** e **cplcomant[canal][faixa]** . Cada canal tem suas próprias coordenadas de acoplamento, as quais podem ser ou não transmitidas, pois o decodificador faz uso das coordenadas anteriores no caso de não receber nova informação. Nos campos AB0 de cada *frame* são sempre transmitidos coordenadas de acoplamento.

Finalmente , o ganho para a reprodução de cada canal que foi codificado dentro do canal de acoplamento pode ser controlado através de um ajuste nos valores das coordenadas correspondentes . O ganho é especificado pelo valor do sub-campo **mstreplco[canal] (2 bits)** e constitui também uma coordenada de acoplamento. Na Tabela B.23 [Apêndice B] mostra-se o mapeamento de **mstreplco**. O valor das coordenadas é multiplicado por mstreplco para se obter o valor final de reprodução.

3.2.7.- Normalização do Sistema de Volume

Continuamente nos sistemas de televisão convencional, enfrentamos o problema da variação do nível (volume) de áudio quando mudamos de canal ou quando assistimos diferentes programas de um mesmo canal [1]. Isso é devido a que alguns canais transmitem o áudio em níveis mais baixos ou mais altos do que outros, de tal forma que o usuário deve variar continuamente o volume do seu receptor a fim de escutar os canais em um nível uniforme de acordo com a sua escolha. Geralmente o ajuste do nível de volume do receptor é feito tomando-se como referência o nível de diálogo do programa que se está assistindo.

O sistema AC-3 oferece uma solução para esse problema introduzindo o sub-campo *dialnorm* de 5 bits dentro do campo BSI que indica o número de dB com que foi codificado o nível de diálogo em relação ao nível máximo de codificação que é 0dB (100%) [10]. Por exemplo, se num determinado momento o dialnorm apresenta um valor igual a 19, isso indica que o nível de diálogo foi codificado com -19 dB (19dB abaixo do valor máximo que é 0dB). Os valores dialnorm podem ser de 1 a 31, sendo o valor zero reservado para o sistema. Com essa informação o decodificador pode dar um nível uniforme de áudio para todos os canais tomando como referência o nível de diálogo com que foi transmitido.

O valor dialnorm é utilizado pelo sistema de controle de volume do receptor, a fim de determinar o nível de reprodução do áudio de acordo com o nível de volume ajustado pelo ouvinte. Em seguida mostramos um exemplo para esclarecer mais o procedimento de normalização:

Suponhamos que um determinado ouvinte ajusta o volume do seu receptor no valor de 61 dB. Nesse instante ele começa a assistir um programa cujo áudio está sendo transmitido com um nível de dialogo de **-25dB** que corresponde a um valor dialnorm igual a “11001”. Dessa forma, o sistema ajusta o nível de reprodução do áudio a **86 dB (61dB + 25dB)**, de tal forma se possa compensar os **-25 dB** com que foi codificado o sinal original [10].

No caso de se ter um serviço principal de áudio com serviços associados, o receptor modifica o dialnorm de cada um deles antes de fazer uma mistura dos mesmos.

3.2.8.- Compressão da Faixa de Reprodução de Volume

Outra das facilidades que apresenta o sistema AC-3, é a compressão da faixa do nível de reprodução de volume (*Dynamic Range Compression*) do sinal que está sendo reproduzido no receptor. Mediante esse sistema são atenuados os níveis altos do sinal e amplificados os níveis baixos, a fim de se gerar um som de alta qualidade de acordo com a preferência do ouvinte.

O processo consiste na inserção de bits em cada bloco de áudio de um determinado canal, que especificam um ganho ou uma atenuação do nível do sinal contido nesse bloco de áudio. Dessa forma, é indicada uma atenuação do sinal para níveis que estão acima do nível de diálogo e uma amplificação para níveis que estão abaixo do mesmo. Isso implica portanto uma compressão da faixa de reprodução de volume do sinal de áudio no receptor [10].

O sistema AC-3 faz a indicação dessas variações através do valor **dynrgn** (8 bits) transmitido em cada campo AB#.

O nível de diálogo não é afetado pela variação da faixa pois é tomado como referência para o cálculo do valor dynrgn [1]. No receptor o usuário pode variar o valor dynrgn, a fim de aumentar a faixa de reprodução que no transmissor foi comprimida pelo *broadcaster*. No caso em que o dynrgn é selecionado a zero, o sistema reproduz o sinal em escala completa, significando que o sinal é reproduzido em 100% da faixa permitida. A alteração do dynrgn pode permitir variações do nível do sinal de até **±24dB**.

Em seguida fazemos um exemplo para ilustrar a forma como o nível de volume é alterado pela compressão da faixa.

Suponhamos que uma pessoa está assistindo um programa cujo áudio foi codificado através do sistema Dolby AC-3 e cujas características são os seguintes:

- **O nível de diálogo é codificado a -30dB**
- **Os níveis altos do sinal alcançam valores de 0dB e os níveis baixos chegam a valores de -50dB abaixo do nível de diálogo (-80dB)**

De acordo com as características do sinal, o codificador transmite os valores de dialnorm em cada bloco de áudio, segundo os níveis de sinal contido neles. Dessa forma, os valores transmitidos dentro do sub-campo dynrgn são os seguintes:

- **-Se o níveis do sinal chega até 0dB, então indica-se uma atenuação de -15dB no atual bloco de áudio.**
- **Se os níveis do sinal chegam até -80dB, então indica-se uma amplificação de 20dB para esse bloco de áudio .**

No receptor, a pessoa ajusta o volume de diálogo a 60 dB e além disso deseja escutar todos os possíveis sons que estão contidos no programa atual. Assim, indica para o decodificador a utilização dos valores dynrgn, para reprodução do nível de áudio.

A reprodução do volume de som será portanto da seguinte forma :

Nível alto (0dB) : $60 \text{ dB} + (30 \text{ dB} - 15 \text{ dB}) = 75 \text{ dB}$ (atenuação -15 dB)

Nível baixos (-80db) : $60 \text{ dB} + (-50 \text{ dB} + 20 \text{ dB}) = 30 \text{ dB}$ (amplificação 20 dB)

É importante ressaltar que no receptor os usuários podem fazer a variação da escala da faixa de compressão a fim de diminuir o efeito sobre o som do programa. Dessa forma, segundo a preferência de cada ouvinte pode-se variar a faixa de reprodução de som até a escala completa.

A codificação dos valores dynrgn é feita da seguinte forma :

Consideramos dynrgn através da seguinte expressão : $X_0X_1X_2Y_3Y_4Y_5Y_6Y_7$

Onde os valores $X=X_0X_1X_2$ são os três bits mais significativos e representam ganhos em dB com sinal, tal como mostra a Tabela B.24 [Apêndice B].

Note que a variação do ganho de um valor a outro é de 6.02 dB e é dado pela seguinte expressão [10]:

$$6.02(X + 1) \text{ dB} \quad (3.2.8.1)$$

Cada bloco de áudio pode conter seu próprio valor dynrgn, mas no caso em que não é transmitido, pode-se fazer o uso do anterior.

No bloco de áudio AB0 sempre se transmite um valor dynrgn, mas no caso que não seja transmitido assume-se o valor 00000000. Os valores Y indicam a variação linear dos valores X que são codificados em forma fracionária da seguinte forma:

$$Y=0.1 Y_3Y_4Y_5Y_6Y_7$$

Portanto, os valores Y representam valores desde 0.111111 (-0.14 dB) até 0.100000 (-6.02 dB). O valor do ganho representado pelos 8 bits de dynrgn é o seguinte [10] :

$$\text{Ganho (dB)} = 6.02(X + 1) + 20 \log (0.1 Y_3Y_4Y_5Y_6Y_7) \quad (3.2.8.2)$$

No caso em que se deseja mudar a faixa de compressão mediante uma variação fracionária, então o sistema primeiro considera o dynrgn da seguinte forma : $X_0X_1X_2Y_3Y_4Y_5Y_6Y_7$, e logo multiplica essa expressão pelo valor da fração e o resultado é o novo dynrgn expresso em forma normal. Por exemplo, diminuindo-se a faixa por um fator de 0.5, os valores dynrgn, indicaram uma variação de ± 12 dB em lugar de ± 24 dB.

Um processador colocado antes do codificador pode informar a esse o intervalo de variação do sinal com que está sendo processado no estúdio de som. Esse é o caso depois que o codificador do sistema faz a inserção de bits com os valores de dynrgn.

Se um serviço principal de áudio com serviços associados são codificados, então o intervalo de variação para o serviço associado é utilizado pelo decodificador para modificar o nível do programa completo [1].

3.2.9.- Serviços de Áudio oferecidos pelo Sistema AC-3

O sistema AC-3 classifica a reprodução de distintas formas de sons na forma de serviços. Isso tem por objetivo oferecer ao usuário com maiores facilidades e vantagens na reprodução do áudio que está sendo decodificado. Dessa forma, o conjunto de serviços oferecidos pelo sistema são classificados em dois serviços principais e 6 tipos de serviços associados [1]. Cada serviço associado é sempre reproduzido em mistura com algum serviço principal. A seguir vamos fazer uma descrição de cada um deles.

3.2.9.1.- Serviços principais.

Classificado em dois tipos de serviços :

- **Serviço principal (CM)**

É um serviço normal de áudio pois contém todos os tipos de sons que estão dentro de um programa. Esse tipo de serviço é o que normalmente se oferece na televisão analógica convencional.

O serviço completo pode utilizar qualquer número de canais disponíveis no sistema. Caso se utilize os seis canais de áudio, a taxa de transmissão de bits vai desde **320** até **384 Kb/s**.

- **Serviço de música e efeitos (ME)**

Produz todos os tipos de som do programa, com exceção do som de diálogo. Pode utilizar todos os canais disponíveis pelo sistema o que faz alcançar uma taxa de bits que vai desde **320** até **384 Kb/s** [2].

3.2.9.2.- Serviços Associados

São serviços adicionais que oferecem algumas facilidades para pessoas com problemas na vista e com problemas na audição. Além disso, existem alguns serviços de informação importantes para o ouvinte. Esses serviços são descritos a seguir:

- **Serviço para pessoas com visão diminuída (VI)**

É um serviço de áudio que contém uma narrativa das imagens que estão sendo apresentadas na tela do receptor. Dessa forma, pessoas que tem problemas na visão podem ter uma idéia mais clara do tipo de imagem do programa que estão assistindo. O serviço VI (apenas de fala) utiliza um só canal de áudio e alcança uma taxa de transmissão de bits que vai desde **48** até **128 Kb/s** [2].

- **Serviço para pessoas com audição deficiente (HI)**

É um serviço que usa um só canal de áudio e que está orientado para melhorar a integridade do diálogo a fim de oferecer a pessoa um som mais claro e entendível. A taxa de bits que alcança vai desde **48** a **96 Kb/s** [2].

- **Serviço de diálogo (D)**

Esse tipo de serviço provê o diálogo que vai ser misturado com o serviço principal CM para a formação de um serviço completo de áudio. Se o serviço principal CM utiliza mais de dois canais, então o serviço de dialogo será codificado utilizando um só canal de áudio (1/0). No caso em que o CM esteja utilizando apenas dois canais estéreo 2/0, então o dialogo pode ser codificado também através de dois canais 2/0.

É importante ressaltar que a mistura de todos os serviços é feita de forma simultânea durante a reprodução do som no receptor. Dessa forma, quando os serviço CM e D utilizam o formato 2/0, os canais direito e esquerdo de cada um deles são misturados correspondentemente. Uma das vantagens de se usar um serviço de diálogo separado é a transmissão multi-linguagem. Nesse sistema, os programas podem ser transmitidos em vários idiomas com o mesmo serviço CM. A taxa de bits alcançada pelo serviço de diálogo vai desde **64** até **128 Kbs** quando se utiliza um canal e desde **96** até **192 Kbs** quando se utilizam dois canais [2].

- **Serviço de comentário (C)**

É um tipo de serviço similar ao serviço de diálogo que oferece comentários adicionais sobre o programa atual que está sendo decodificado. O serviço C é codificado através de um só canal de áudio. Pode-se alcançar uma taxa de bits que vai desde **32** a **128 Kbs** [2].

- **Serviço de emergência (E)**

É um tipo de serviço orientado à transmissão de avisos de emergência, pois tanto o sistema de transporte quanto o receptor dão prioridade a transmissão e recepção desse tipo de serviço. Quando o decodificador recebe esse tipo de serviço,

deixa de reproduzir os serviços de áudio atuais para a reprodução do anúncio de emergência. A utilização de um só canal de áudio leva esse tipo de serviço a uma taxa que vai desde **32** até **128 Kbs** [2].

- **Serviço *voice-over* (VO)**

É um tipo de serviço similar ao serviço de emergência. A diferença está no fato de que os serviços atuais não são apagados quando a reprodução é feita no receptor. Esse serviço é misturado com a voz de aviso. De igual forma esse tipo de serviço utiliza um só canal de áudio que demanda uma taxa de bits que vai desde **64** até **128 Kbs**.

Cada serviço de áudio é codificado através de uma seqüência de frames que constitui um *elementary bit stream*, os quais posteriormente são multiplexados pelo sistema de transporte para serem transmitidos dentro de um determinado programa. O sistema de transporte será objeto de nosso estudo no próximo capítulo.

3.3.- Conclusões e Comentários Finais

Através do uso da codificação perceptual adaptativa chegou-se finalmente a resultados ótimos em relação a taxa de transmissão versus a qualidade do sinal resultante. A taxa de bits alcançada na transmissão dos seis canais de áudio num serviço completo é de **384 Kb/s**, onde o nível de compressão alcançado encontra-se na relação de **13:1**.

Durante todo o processo de codificação tem-se levado em conta a diminuição da taxa de bits conservando-se a qualidade do sinal resultante. Dessa forma, desde a codificação dos expoentes, a quantização de mantissas e a formação dos canais acoplados, as técnicas de codificação empregados no algoritmo foram definidas com base nas características de resposta em frequência de nosso sistema de audição.

Por outro lado, além de obter um sinal de alta qualidade no receptor, a utilização de um sistema de normalização e compressão da faixa do nível de volume, permite solucionar um problema apresentado na televisão convencional onde diversos programas de um mesmo canal ou de canais diferentes são reproduzidos com diferentes níveis de volume. A normalização permite a reprodução de qualquer programa de televisão codificado através do sistema AC-3 mediante um som de nível constante, inclusive quando é feita uma mudança de canal.

A compatibilidade do sistema Dolby AC-3 com sistemas de áudio estéreo convencionais é assegurada mediante a detecção de dois canais de áudio L, R, codificados em cada *frame* de sincronização.

O sistema Dolby AC-3 vem sendo utilizado para o processamento de áudio de alguns filmes de cinema desde o ano de 1992. A alta qualidade de som obtido para

esses casos bem como a taxa de compressão alcançada, permitiram a consideração do sistema AC-3 dentro do sistema HDTV proposto pela Grande Aliança Estados Unidos.

Finalmente, o formato do *frame* de sincronização apresentado neste capítulo apresenta outros campos de informação que indicam para o decodificador algumas características adicionais da informação que muitas vezes são necessárias para uma boa recepção e para uma boa reprodução de um som de alta qualidade.

Capítulo 4

Sistema de Transporte e Multiplexagem

4.0.- Introdução

Em um sistema de transmissão digital como é o caso da televisão HDTV, é necessário que a informação a se transmitir (áudio, vídeo, etc) seja agrupada em pacotes de bits com informação adicional de identificação e sinalização, de forma que se possa facilitar a formação do quadro digital mediante a multiplexação desses pacotes (programa de transporte), bem como indicar ao decodificador as características da seqüência de bits que se está recebendo em um determinado momento.

O processo de sincronização temporal entre o codificador e o decodificador, o acesso condicional à informação e a inserção de fontes com bases de tempo diferentes, constituem exemplos das facilidades obtidas através do empacotamento da informação proveniente dos codificadores.

Um sinal de televisão contém informações de áudio e vídeo que são transmitidas em forma serial ao decodificador para serem então reproduzidas através dos correspondentes elementos transdutores (tubos de raios catódicos, caixas de som, vídeo e cinescópios).

Dessa forma, a função da **camada de transporte** é fazer o empacotamento da informação (áudio, vídeo e dados) adicionando campos de sincronização, identificação e de sinalização que permitam indicar ao decodificador as características e propriedades de cada mensagem binária que é recebida. Cada um desses pacotes é chamado de **pacote de transporte**.

Por outro lado, a seqüência de bits proveniente dos codificadores de áudio e vídeo é chamada de *elementary bit streams* e pode ser empacotada inicialmente em pacotes **PES** (*packetized elementary bit streams*) a fim de dar uma maior funcionalidade ao sistema, na sincronização e identificação de cada aplicação. O empacotamento PES muitas vezes pode ser feito pelos respectivos codificadores.

O sistema de **multiplexagem** é encarregado da formação dos diferentes grupos de informação constituídos pelos **programas de transporte**, bem como o agrupamento dos mesmos para a formação da seqüência final de programas que será transmitida. Um programa de transporte é formado a partir de *elementary bit streams* gerados a partir de uma mesma base de tempo.

A multiplexagem é feita em dois níveis de hierarquia que permitem de uma forma flexível, a geração de distintos programas de televisão. No primeiro nível são multiplexados os pacotes de transporte que levam informação de áudio e vídeo. Esses são necessários para a formação de cada programa de transporte (programa de televisão). Um programa de transporte em televisão digital é o que normalmente se conhece como um “canal” em televisão convencional. No segundo nível são multiplexados os diversos programas que compõem o sistema, para dessa forma se obter finalmente a seqüência serial de programas que serão modulados e transmitidos aos diferentes receptores. De acordo com isso, observa-se que dependendo do grau de compressão do sinal de vídeo pode-se transmitir vários programas dentro de um único canal de televisão convencional.

O formato dos pacotes e a forma de multiplexação são feitos de acordo com a recomendação **ISO/IEC 13818-1**, que considera o sistema **MPEG-2**, (proposto pelo sistema americano da *Grande Aliança*), como a melhor alternativa para as aplicações requeridas nos sistemas de televisão digital HDTV.

Neste capítulo vamos descrever os aspectos mais relevantes do sistema de empacotamento de acordo com o formato MPEG-2 para em seguida fazer uma descrição do processo de multiplexagem na formação da seqüência de programas que serão transmitidos. Na figura 4.0.1 mostra-se o diagrama de blocos de processo de empacotamento e multiplexagem.

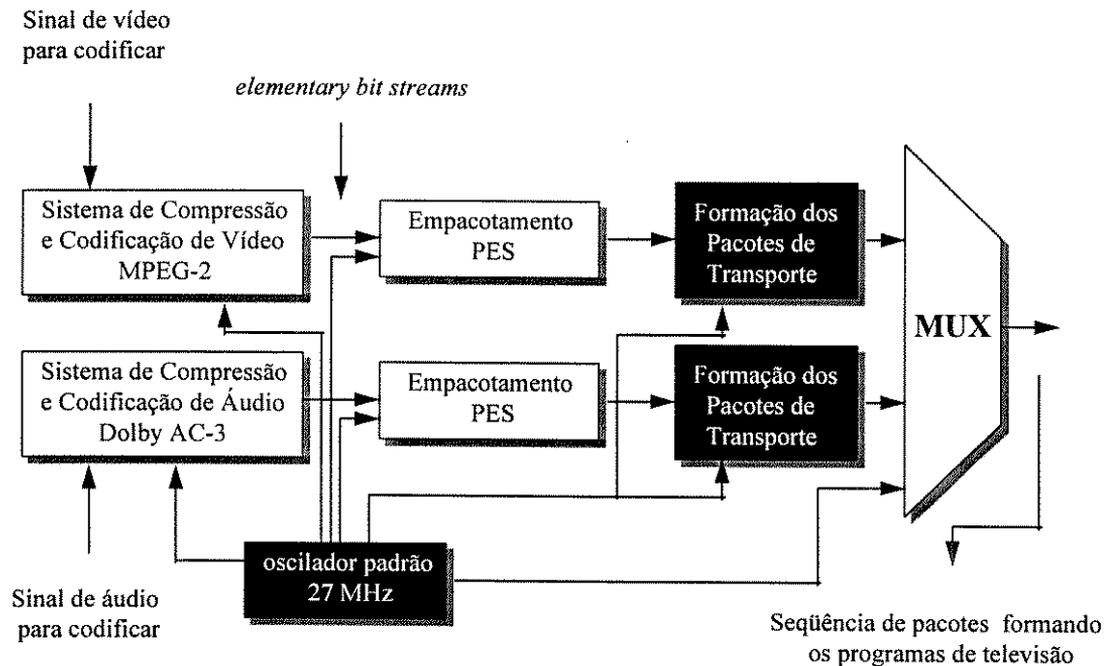


Figura 4.0.1. Processo de empacotamento e multiplexação da informação de áudio e vídeo

4.1.- Pacotes PES (*Packetized Elementary bit Streams*)

Os pacotes PES constituem uma alternativa inicial de empacotamento da seqüência de bits provenientes dos codificadores de áudio (*frames* de sincronização) e vídeo (quadros, e seqüência de quadros) denominados como *elementary bit streams*. Cada informação de entrada no sistema de transporte e multiplexagem pode estar no formato de pacotes PES ou na forma normal como *elementary bit streams*. O empacotamento da informação através de pacotes PES permite uma maior funcionalidade do sistema para processos de identificação e sincronização na etapa de recepção de cada aplicação individual do sistema de televisão.

Uma característica importante dos pacotes PES é que não têm um tamanho fixo e, portanto, contêm bits no cabeçalho que indicam ao decodificador o comprimento do pacote que está sendo recebido. A seguir vamos fazer uma descrição geral da estrutura de um pacote PES para em seguida fazer uma apresentação dos principais campos que o formam.

4.1.1.- Estrutura de um pacote PES

Um pacote PES é formado principalmente pelos campos de cabeçalho (*PES Header*) e pelo campo de dados (*Data Block or Payload*). Os campos de cabeçalho levam a informação de identificação e sinalização do pacote, enquanto que o campo de dados leva o *elementary bit streams* de uma determinada aplicação. Na Figura 4.1.1 é apresentada, a estrutura de um pacote PES.

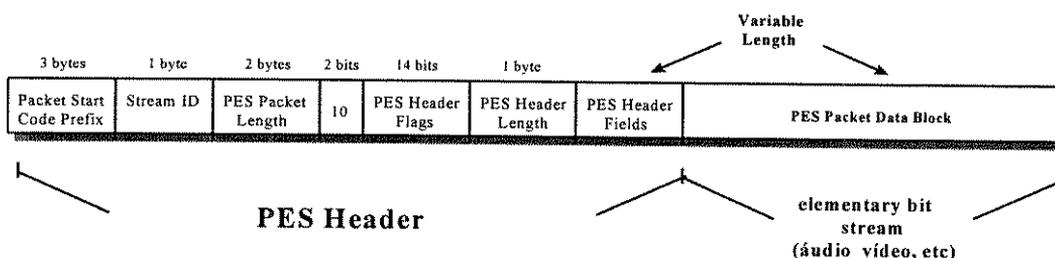


Figura. 4.1.1 Estrutura de um Pacote PES

Note que os dois últimos campos correspondentes aos campos opcionais de cabeçalho e de dados são os responsáveis pelo comprimento variável do PES. Isso é devido ao fato de que esses campos não têm um comprimento definido para cada pacote.

4.1.1.1.- Campo de Inicialização e Sincronização (*Packet Start Code Prefix*)

Trata-se de um código digital de 24 bits (3 bytes) que leva o valor “000000000000000000000001” (binário) e que indica ao decodificador o início de um novo pacote PES.

4.1.1.2.- Campo de Identificação (*Stream ID*)

É um campo de 8 bits que especifica a característica do *elementary bits stream* contido no pacote PES. O mapeamento do *stream ID* é mostrado na Tabela 4.1.1

stream id	stream codingtype
<i>1011 1100</i>	<i>program stream map</i>
<i>1011 1101</i>	<i>private stream 1</i>
<i>1011 1110</i>	<i>padding stream</i>
<i>1011 1111</i>	<i>private stream 2</i>
<i>110x xxxx</i>	<i>ISO/IEC 13018-3 audio stream or ISO/IEC 11172-3 audio stream number xxxxxx</i>
<i>1110 xxxx</i>	<i>ISO/IEC 13818-2 video stream or ISO/IEC 11172-3 video stream number xxxxxx</i>
<i>1111 0000</i>	<i>ECM</i>
<i>1111 0001</i>	<i>EMM</i>
<i>1111 0010</i>	<i>DSM CC</i>
<i>1111 0011</i>	<i>ISO/IEC 13522 stream</i>
<i>1111 xxxx</i>	<i>reserved data stream - number xxxxxx : 0100 - 1110</i>
<i>1111 1111</i>	<i>program stream directory</i>

Tabela 4.1.1 Mapeamento do Campo de Identificação Stream ID

Na Tabela anterior especifica-se os respectivos bytes para cada aplicação dos *elementary streams* contidos num PES. Note que, além de identificar a informação de áudio ou vídeo, esse campo serve para indicar outras características da informação como, por exemplo, para aplicações de armazenamento DSM (*digital storage media*) que não são considerados em transmissão terrestre de televisão.

Outro tipo de informação é aquela correspondente às mensagens de controle contidas num pacote PES que são identificadas por ECM (*entitlement control mesagge*) e EMM (*entitlement management messages*). Essas mensagens geralmente são indicadas para acessos condicionais e particulares.

Na documentação emitida pelo ATSC, especifica-se o *stream id* igual a “10111101” (*private stream1*) para a identificação do *elementary bit stream* proveniente do codificador de áudio AC-3.

Como veremos mais adiante a formação de programas através de **pacotes de transporte** consiste na multiplexação de vários desses pacotes com um pacote

adicional levando informação de identificação das características do programa. Da mesma forma, os pacotes PES podem ser multiplexados para a formação de programas adicionando-se um pacote PES denominado *program stream map* que contém as características dos *elementary bit stream* que formam o programa.

Para televisão digital a formação de programas faz-se multiplexando-se os **pacotes de transporte**, devido as vantagens do seu comprimento fixo. Cada pacote de transporte empacota os PES ou os *elementary bit stream* no caso de não existência do empacotamento PES.

Outra característica do empacotamento PES é que os primeiros bytes do *data block* sempre são alinhados com o início de uma seqüência de quadros, um grupo de quadros GOP, ou com um *star code* de um determinado quadro. Dessa forma, de acordo com a documentação emitida pelo ATSC, o primeiro byte do *payload* de um PES deverá ser sempre o primeiro byte de uma unidade de acesso de vídeo. Além disso, para transmissão terrestre de televisão, um pacote PES deverá conter um único quadro codificado.

4.1.1.3.- Campo de indicação do comprimento do PES (*PES packet length*)

É um campo de 16 bits que indica o tamanho em **bytes** do pacote PES a partir do primeiro byte do próximo campo consecutivo. O tamanho máximo de um PES que se pode indicar através do *PES packet length* é de 2^{16} bytes. No entanto, quando é especificado o valor de zero indica-se que o tamanho do pacote não é definido ou não tem limite. Isso só é utilizado quando o *elementary bit stream* corresponde à informação de vídeo. Na documentação emitida pelo ATSC especifica-se o valor de **zero** para esse campo.

4.1.1.4.- Campo de indicadores e sinalizadores (*PES Header Flags*)

Esse campo contém os indicadores das propriedades do *elementary bit streams* contido no pacote PES. Esses indicadores especificam também a existência de campos opcionais localizados no *PES Header Fields* mostrado na Figura 4.1.1. Na Figura 4.1.2 mostramos a disposição dos indicadores dentro do campo *PES Header Flags* bem como o número de bits associados a cada um deles. A existência de um sub-campo é indicado com “1” e em caso contrário com “0”.

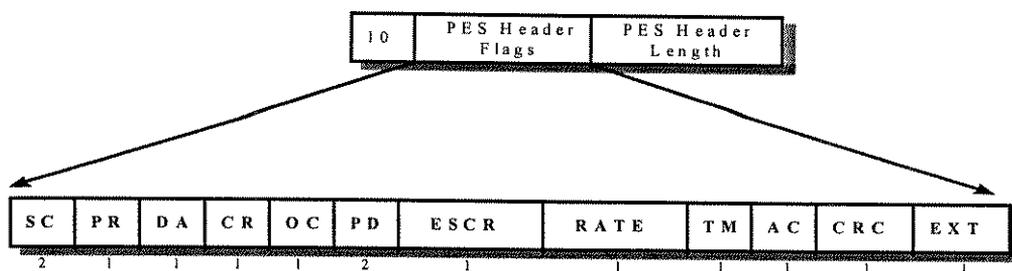


Figura 4.1.2. Disposição dos Indicadores PES Header Flags

- **SC** (*scrambling control*) : especifica o acesso condicional à informação contida no *data block* tal como é mostrado na Tabela 4.1.2. O valor considerado para aplicações em transmissão *broadcasting* de HDTV é “00” (documentação do ATSC).

Valor SC	Descrição
00	acesso livre
01	usuário definido
10	usuário definido
11	usuário definido

Tabela 4.1.2. Mapeamento do indicador de acesso condicional SC

- **PR** (*PES priority*) especifica a alta prioridade do pacote atual quando leva o valor de “1”. Dessa forma, o multiplexador do sistema toma conhecimento da prioridade do pacote para sua transmissão.
- **DA** (*data alignment indicator*) especifica que no *data block* do PES está contida uma unidade de acesso. Uma unidade de acesso no caso do vídeo, é formada pela seqüência de bits codificados que compõem um quadro, enquanto que para o sinal de áudio é formada pelo conjunto de bits que formam um áudio *frame*.

Um pacote PES é sempre alinhado para que os primeiros bytes do seu *data block* sejam os primeiros bytes de uma unidade de acesso de tal forma que o valor do DA seja sempre “1” cada vez que o PES leva esse tipo de informação. Esses bytes podem ser o início de uma seqüência de quadros, de um grupo GOP ou de um quadro (*star code*). Na documentação emitida pelo ATSC especifica-se que este indicador deverá levar sempre o valor de “1”, pois foi determinado que um PES deve levar sempre um único quadro completo codificado.

- **CR** (*copyright*) indica se a informação que leva o PES é uma cópia com (“1”) ou sem (“0”) direitos reservados.
- **OC** (*original copy*) indica se a informação que leva o PES é original ou cópia.
- **PD** (*PTS-DTS flags*) indica a presença de dois campos de sincronização no campo *PES Header Fields* mostrado na Figura 4.1.1. O mapeamento do indicador é mostrado na Tabela 4.1.3.

Valor	Presença
00	nenhum dos dois
01	não utilizado
10	PTS
11	PTS e DTS

Tabela 4.1.3. Mapeamento do indicador PD

- **ESCR** indica a presença do campo de sincronização ESCR. O valor considerado para ESCR em transmissão de televisão HDTV é “0” (documentação ATSC).
- **ES** indica a presença do campo *ES-rate (elementary stream rate)* o qual especifica, através de um número inteiro positivo, a frequência de recepção de bytes para o decodificador. O *ES-rate* é codificado em unidades de 50 byte/segundo. Para aplicações em transmissão HDTV esse indicador tem o valor “0” (documentação ATSC).
- **TM** (*trick mode*) indica a presença do campo *trick mode control* para aplicações em DSM o qual não é considerado em transmissão terrestre de televisão, mas é levado em conta para aplicações VCR (*video cassette recorder*).
- **AC** indica a presença de um campo adicional de informação para cópia com direitos reservados.
- **CRC** indica a presença do campo *previous PES packet CRC* para a detecção de erros no PES anterior. Para aplicações em transmissão de HDTV, esse indicador tem o valor de “0”.
- **EXT** indica a presença de indicadores adicionais (*PES extension flags*).

4.1.1.5.- Campo indicador do comprimento do cabeçalho do PES (*PES Header Length*)

É um campo de 8 bits que indica o tamanho em bytes dos campos opcionais mais os bytes *stuffing*. O número máximo de bytes que pode ser indicado é de 2^8 .

4.1.1.6.- Campos opcionais (*PES Header Fields*)

São campos que não são necessariamente transmitidos em cada PES e que levam informação adicional de sincronização e de descrição das características do *elementary bit streams*. A transmissão de cada um desses campos é especificada pelos indicadores relacionados anteriormente. Na Figura 4.1.3 mostra-se a disposição desses campos dentro do *PES Header Fields*.

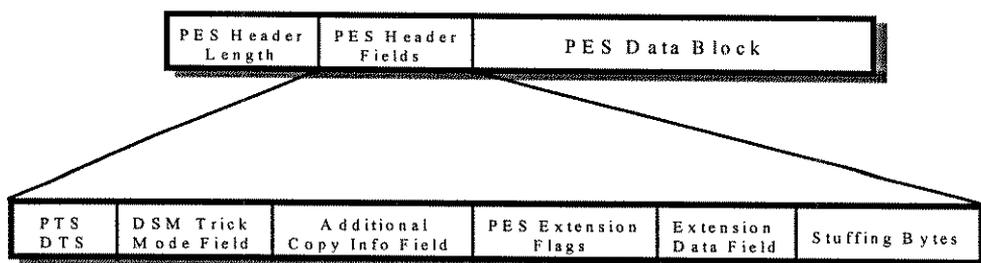


Figura. 4.1.3. Disposição dos campos opcionais no PES Header Fields

- **PTS** (*presentation time stamp*) indica ao decodificador o tempo estimado de apresentação de uma **unidade de apresentação**. Uma unidade de acesso é uma unidade de apresentação codificada. O valor de PTS refere-se ao tempo de apresentação da primeira unidade de acesso que ocorre no pacote. Dessa forma, se uma unidade de acesso não está disponível no *data block* então esse pacote não leva o campo PTS. Para aplicações em televisão digital, determinou-se que cada pacote PES levará sempre um único quadro completo codificado que constitua uma unidade de acesso ao sistema, de forma que em cada um deles exista um valor PTS codificado.

O PTS é calculado em número de períodos de um relógio de **90 KHz** (*system clock frequency*) codificados através de 33 bits, tal como mostra a seguinte equação.

$$PTS = INT(system\ clock\ frequency \times (tp_n(k))) \quad (4.1.1)$$

onde $tp_n(k)$ é o tempo de apresentação da unidade de apresentação $P_n(k)$.

O $P_n(k)$ corresponde à primeira unidade de acesso contida no *data block* do PES. A função INT retorna o valor inteiro mais próximo.

- **DTS** (*decoding time stamp*) indica ao decodificador o tempo estimado de decodificação da primeira unidade de acesso contida no PES. O valor do DTS é expresso através de um valor de 33 bits e é medido em número de períodos de um relógio de **90 KHz** (*system clock frequency*), tal como mostra a seguinte equação:

$$DTS = INT(system\ clock\ frequency \times (td_n(j))) \quad (4.1.2)$$

onde $td_n(j)$ é o tempo de decodificação da unidade de acesso $A_n(j)$.

O DTS é codificado quando o valor do mesmo não for igual ao PTS, como ocorre quando o PES leva um B-quadro ou quando leva outro tipo de quadro onde não é necessário o reordenamento. Portanto o DTS só é codificado junto a um PTS tal como é mostrado na Figura 4.1.5. Os valores de PTS ou DTS são codificados através de três campos separados por bits de marca tal como é mostrado na Figura 4.1.4. Na figura 4.2.5 mostra-se a utilização dos valores PTS e DTS no instante da decodificação da seqüência de pacotes que é recebida.

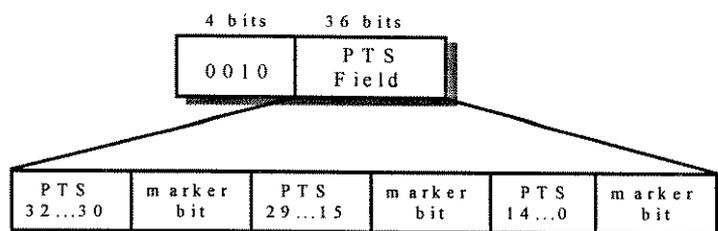


Figura. 4.1.4. Organização do PTS Field.

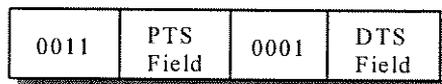


Figura.4.1.5 Organização dos PTS Field e DTS Field quando ambos valores PTS e DTS são codificados

- **PES extension flags** contém indicadores que especificam a existência de campos adicionais de informação, tal como é mostrado na Figura 4.1.6.

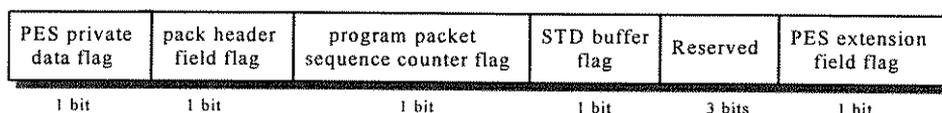


Figura. 4.1.6. Organização do PES extension flags

Cada indicador especifica a existência do campo correspondente dentro do **PES Extension Data Field** que contém todos os campos cuja presença é especificada pelos indicadores quando levam o valor de “1”. A seguir vamos descrever cada um desses campos.

- **PES private data** é um campo de 16 bits que contém informação particular que ao juntar-se com os campos anteriores e posteriores nunca deverá dar o valor 000001_b , a fim de evitar que seja confundido pelo decodificador como *packet start code prefix*. Em transmissão de HDTV esse campo não é considerado. Portanto o valor do seu indicador respectivo será igual a “0”.
- **Packet field length** é um campo de 8 bits que indica o número de bytes pelos quais é composto o *PES Header Field*.
- **Program packet sequence counter** é um campo de 87 bits que funciona como um contador cujo valor é incrementado em cada PES, o qual é multiplexado para a formação de um programa. Dessa forma, dois PES consecutivos não terão o mesmo *program packet sequence counter*. Esse campo não é considerado para transmissão de HDTV (documentação ATSC).
- **P-STD buffer scale (Program stream only)** é um campo de um bit utilizado para indicar o fator de escala na interpretação do valor binário transmitido no campo *P-STD buffer size*. Quando o *stream id* indica que o PES leva informação de áudio o *P-STD buffer scale* é transmitido com o valor de “0” e quando indica informação de vídeo o *P-STD buffer scale* é transmitido com o valor de “1”.
- **P-STD buffer size (Program stream only)** é um campo de 13 bits que serve para indicar ao decodificador o tamanho estimado do buffer de recepção de dados. Quando o *P-STD buffer scale* leva o valor de “0” o *P-STD buffer size* é calculado

em unidades de 128 bytes, e quando o *P-STD buffer scale* leva o valor de “1” então o *P-STD buffer size* é calculado em unidades de 1024 bytes.

Esses dois últimos campos não são considerados em transmissão de HDTV (documentação ATSC).

- **original stuffing length** é um campo de 7 bits que especifica o número de *stuffing bytes* para o *packet header* no ISO/IEC 11172.
- **stuffing byte** constitui um campo de bytes do tipo “11111111” que não leva nenhuma informação mas é utilizado pelo codificador quando se deseja chegar a um tamanho pré-definido do PES. Não mais de 32 desses bytes são permitidos.

Até aqui fizemos uma descrição dos principais campos que formam um pacote PES de acordo com a recomendação ISO/IEC 13818-1. Na próxima seção faremos a descrição dos principais campos que formam os **pacotes de transporte** responsáveis pelo empacotamento dos PES ou dos *elementary bits streams*.

4.2.- Pacotes de Transporte

Os pacotes de transporte são formados a partir do empacotamento dos PES ou dos *elementary bits stream* caso não exista o empacotamento PES. Cada programa de televisão é formado mediante a multiplexação dos pacotes de transporte os quais levam informação de áudio e vídeo correspondente a um determinado programa.

O formato de um pacote de transporte é determinado também de acordo com a recomendação ISO/IEC 13818-1.

4.2.1.- Estrutura de um Pacote de Transporte

A característica mais importante de um pacote de transporte é que possui um comprimento fixo de **188 bytes** onde estão incluídos os campos de cabeçalho e de informação a serem transmitidos. Na Figura 4.2.1 mostra-se o formato de um pacote de transporte.

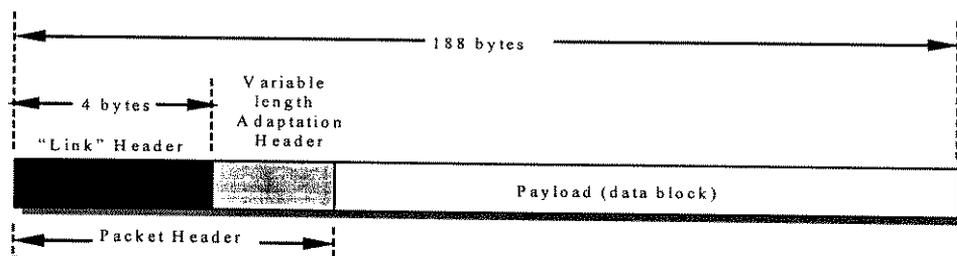


Figura 4.2.1. Formato de um Pacote de Transporte

O tamanho fixo do pacote transporte em 188 bytes, é devido às seguintes considerações :

- O tamanho do pacote deve ser de um certo comprimento de forma que o *overhead* do cabeçalho do pacote que leva informação de identificação e sinalização não represente uma porção considerável do total de dados que se deseja transmitir.
- Por outro lado o pacote não deve ser de um comprimento muito longo a fim de evitar um aumento da probabilidade de erro, devido ao fato de que se conta com um deficiente sistema de correção.
- O tamanho fixo do pacote permite dar uma forma muito flexível ao processo de multiplexação da informação de áudio e vídeo para se obter a formação dos distintos programas de televisão.
- Outra razão pela qual se escolhe o comprimento do pacote em 188 bytes é a facilidade da representação do mesmo através de 4 células ATM (*Asynchronous Transfer Mode*), permitindo dessa forma que o sinal de televisão possa ser transmitido facilmente sob esse protocolo de comunicação. A Figura 4.2.2 mostra a comparação entre a estrutura de uma célula ATM e um pacote de transporte MPEG-2.

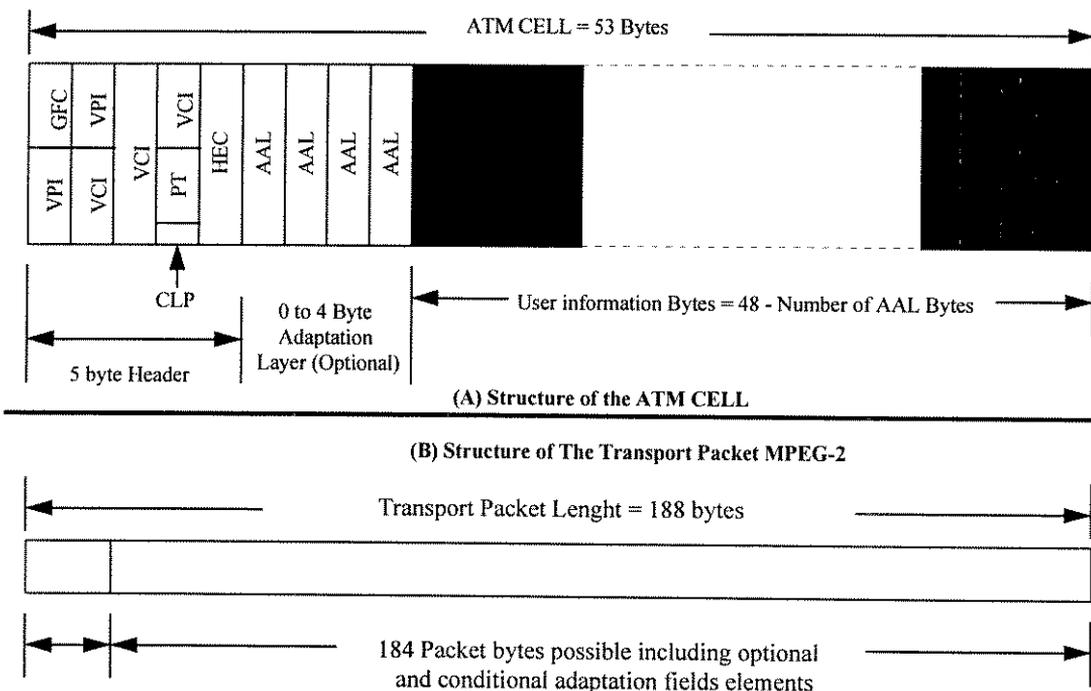


Figura 4.2.2 Comparação entre a estrutura de uma célula ATM e um pacote de transporte MPEG-2.

Cada pacote de transporte é formado principalmente pelo campo de cabeçalho (*Packet Header*) e o campo de dados (*payload - data block*). O *Packet Header* está formado pelo *link Header* de comprimento fixo e o *Adaptation Header* de comprimento variável é opcional. O *payload* leva toda a informação dos *elementary bits stream* com ou sem empacotamento PES.

O *Packet Header* é responsável por levar toda a informação de sincronização, de identificação (PID) e de sinalização necessária para assegurar uma decodificação livre de erros para cada aplicação de áudio e de vídeo.

A seguir faremos uma descrição das características dos principais campos que formam o *Packet Header* de um pacote de transporte.

4.2.1.1.- Campos de Cabeçalho (*Link Header*)

O “*link*” header é formado pelos 4 primeiros bytes do *Packet Header*. Esses bytes formam os primeiros campos de sincronização, de identificação e de sinalização do pacote de transporte. Na Figura 4.2.3 mostra-se o formato do *link header* e os respectivos campos que o formam.

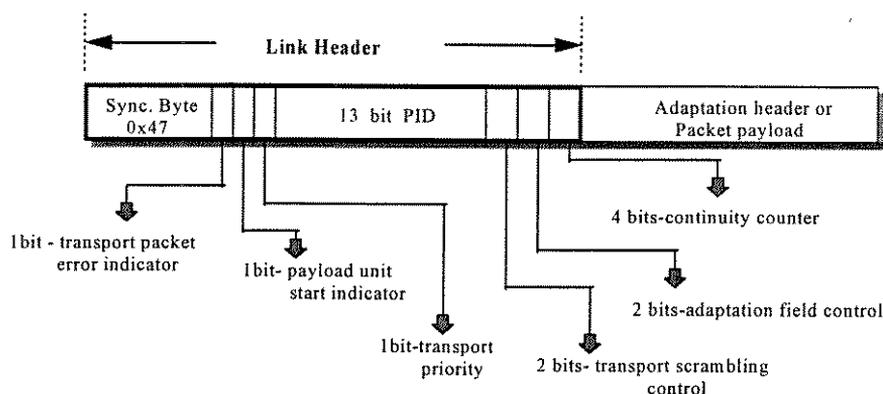


Figura. 4.2.3. Formato do Link Header de um Pacote de Transporte

- **Sync. byte** é um campo que indica o início de um novo pacote de transporte. O valor desse byte é sempre “01000111” (0x47) e serve para sincronizar a recepção dos pacotes de transporte.
- **Transport packet error indicator** é um campo de um bit utilizado pelo modem localizado no receptor para indicar a ocorrência de um erro na recepção dos dados. Quando o modem leva esse bit ao valor “1”, isso indica ao decodificador que aconteceu um erro na recepção do pacote. Caso contrário, quando o indicador leva o valor de “0”, significa que não houve nenhum erro na recepção.
- **Payload unit start indicator** é um bit utilizado quando o pacote de transporte leva (no *payload*) informação empacotada como PES ou **PSI (packet specific information)**. No caso de levar PES o indicador é levado a “1” quando os primeiros bytes do *payload* correspondem aos primeiros bytes do PES (alinhamento). Caso contrário, é “0”.

Da mesma forma, no caso em que o pacote de transporte leve um PSI, o indicador é levado a “1” quando os primeiros bytes do *payload* correspondem aos

primeiros bytes do PSI. Caso contrário é levado a “0”. Quando o pacote atual de transporte constitui um pacote nulo, o valor do indicador deve ser “0”.

- **Transport priority** é um bit que indica que o pacote atual tem prioridade sobre os outros pacotes que contém o mesmo PID e com esse bit igual a “1”. A prioridade é indicada quando esse indicador é levado ao valor de “1”.
- **PID (Packet Identification)** é um campo de 13 bits que constitui o código de identificação do *elementary bit stream* contido no pacote de transporte. O valor 0x0000 é reservado para quando a informação que está sendo transportada constitua um **program association table** que apresentaremos mais adiante. Da mesma forma o valor 0x0001 é reservado para informação com acesso condicional.

Finalmente os valores do PID no intervalo 0x0002-0x000F são reservados, e o valor 0x1FFF é reservado para pacotes nulos.

- **Transport scrambling control** é um campo de 2 bits que indica o modo de acesso à informação contida no *payload* do pacote. Na Tabela 4.2.1 mostra-se o mapeamento do indicador.

<i>transport scrambling control</i>	tipo de acesso
00	acesso livre
01	acesso livre, mas pode ser usado como indicador pelo fornecedor do serviço para uso particular.
10	acesso condicional com “chave” par
11	acesso condicional com “chave” impar

Tabela 4.2.1. Mapeamento do indicador de Modo de Acesso

No caso em que o pacote seja nulo o valor do indicador sempre deve ser levado a “00”. O acesso condicional é flexível no sentido de suportar toda transmissão onde possam ser aplicadas as chaves de controle de acesso para qualquer *elementary bit stream*.

O acesso condicional opera sobre o princípio do embaralhamento dos dados que estão chegando no decodificador, de forma a evitar que qualquer receptor não autorizado possa decodificar a informação que está sendo transmitida.

Os decodificadores autorizados recebem uma “chave” que inicializa o circuito responsável pela operação inversa de desembaralhamento, contrária àquela feita no transmissor. As “chaves” são transmitidas em forma codificada com um certo período de tempo a fim que sejam protegidas de qualquer decodificador pirata. Uma “chave” só é válida por um tempo pequeno até que o codificador transmita “chaves” novas. As chaves codificadas são transmitidas no campo de uso particular do *adaptation field* e com um PID igual a 0x0001 a fim de indicar ao decodificador que o pacote atual leva informação particular.

- **Adaptation field control** é um campo de 2 bits que indica se o pacote de transporte está levando ou não o *adaptation field* ou o *payload*. Na Tabela 4.2.2 mostramos o mapeamento desse indicador.

Valor do <i>Adaptation Field Control</i>	Descrição
00	reservado para uso futuro por ISO/IEC
01	sem <i>adpatation header</i> só <i>payload</i>
10	só <i>adptation header</i> sem <i>payload</i>
11	<i>adaptation header</i> seguido por <i>payload</i>

Tabela 4.2.2. Mapeamento do indicador *adaptation field control*

- **Continuity counter** é um campo de 4 bits que funciona como contador dos pacotes de transporte com o mesmo PID. O valor do contador retorna a zero depois do valor “1111”. Quando o *adaptation field control* tem os valores de “00” ou “01”, o contador não será incrementado. Quando se transmitem pacotes duplicados, o valor do contador será igual ao do original e o *adaptation field control* será “10” ou “01”. Um pacote duplicado tem todos os bytes iguais ao do original, com exceção do PCR. Para pacotes nulos, o valor do contador é indefinido.

4.2.1.2.- Campos de Cabeçalho Opcionais (*Adpatation Field*)

O *adaptation field* constitui o conjunto de campos opcionais que podem ser transmitidos dentro do cabeçalho do pacote de transporte. Na Figura 4.2.4 mostramos a disposição desses campos dentro do *adaptation field*.

A seguir faremos uma descrição dos principais campos e indicadores que formam o *adaptation field*.

4.2.1.2.1.- Campo indicador do comprimento do Cabeçalho Opcional (*Adaptation field length*)

É um campo de 8 bits que especifica o número de bytes do *adaptation field* a partir do primeiro byte do próximo campo consecutivo. O tamanho indicado por esse campo não deve exceder 183 bytes.

4.2.1.2.2.- Campo de Indicadores e Sinalizadores

Nesse campo, localizam-se todos os sinalizadores e indicadores do *adaptation field* que indicam certas propriedades do *elementary bit stream* contido no pacote. Além disso, especifica a presença ou não de campos opcionais no *adaptation field header*.

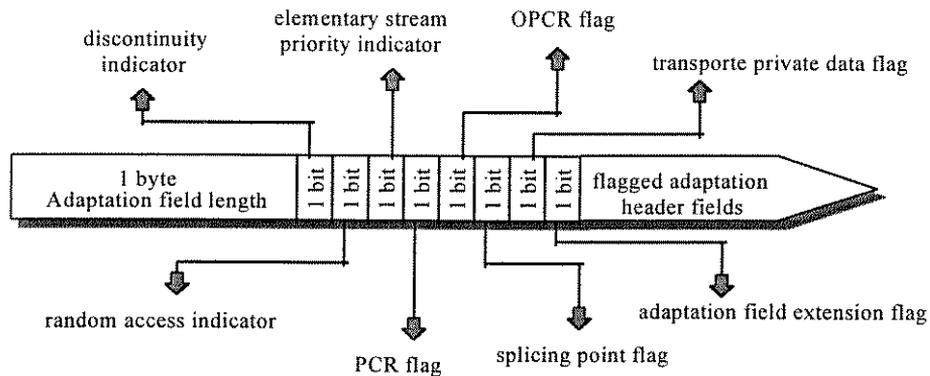


Figura. 4.2.4 Formato do adaptation field de um pacote de transporte

- **Discontinuity indicator**, é um indicador de um bit que quando leva o valor de “1” indica que o valor do próximo PCR (*program clock reference*) numa seqüência de pacotes com o mesmo PID, representa uma amostra de um novo sistema de relógio com base de tempo diferente. O *discontinuity indicator* de um pacote de transporte que leva o primeiro PCR com base de tempo diferente, é sempre levado a “1”.
- **Random access indicator**, é um indicador de um bit que quando tem o valor de “1” indica que o próximo PES dentro do atual ou no pacote de transporte seguinte está levando um PTS e um *elementary bit stream* que constituem um ponto de acesso. Um ponto de acesso em informação de vídeo é constituído pelos primeiros bytes de uma seqüência de cabeçalho e no caso de áudio é constituído pelos primeiros bytes da palavra de sincronização de um áudio *frame*. Quando o indicador é levado a “0” não define se o PES está levando ou não um *elementary bit stream* que seja ponto de acesso para alguma aplicação.
- **Elementary stream priority indicator**, é um bit que quando tem o valor de “1” indica que o *payload* do atual pacote tem maior prioridade que os *payload* de outros pacotes cujo indicador é igual a “0”. No caso de vídeo, esse indicador é levado a “1” quando o *payload* do pacote leva uma ou mais fatias codificadas como intra-quadros.
- **PCR flag**, é um bit que indica (“1”) a presença do campo PCR no *adaptation field*.
- **OPCR flag**, é um bit que indica (“1”) a presença do campo OPCR no *adaptation field*.
- **Splicing point flag**, é um bit que indica (“1”) a presença do campo *splice countdown* no *adaptation field*.
- **Transport private data flag** é um bit que indica (“1”) a presença de um ou mais campos *private data bytes*.

- **Adaptation field data flag** é um indicador de um bit que indica (“1”) a presença do campo *adaptation field extension* como último campo do *adaptation field*.

4.2.1.2.3.- Campos opcionais

São campos que não são necessariamente transmitidos em cada pacote de transporte, cuja presença é indicada pelos indicadores apresentados anteriormente. A seguir vamos descrever cada um desses campos.

- **PCR program clock reference (PCR)**, é um campo de 42 bits que, junto com o PTS e o DTS de um pacote PES, constituem as informações de sincronização de tempos transmitidas para o receptor a fim de assegurar o sincronismo entre os relógios do transmissor e do receptor.

O PCR leva as amostras do relógio de **27 MHz** utilizadas no sistema de transporte e multiplexagem do transmissor, a fim de sincronizar o relógio do decodificador e evitar defasagens que originem uma perda de sincronismo.

O PCR não muda diretamente o valor da fase do relógio local, somente ajusta a frequência do mesmo. Como se mencionou anteriormente nos capítulos 2 e 3, a partir do relógio de 27 MHz usado pelo sistema de transporte e multiplexagem obtém-se os relógios de amostragem tanto para áudio como para vídeo, pois dessa forma assegura-se o sincronismo de todo o sistema digital.

O PCR é codificado em dois valores:

Program clock reference base (PCRB), é um campo formado pelos 33 primeiros bits do campo PCR, cujos valores representam o número de períodos de um relógio de 90 KHz gerado a partir do oscilador padrão de 27 MHz. O valor do PCRB é dado por:

$$PCRB(i) = \frac{((27\text{MHz}) \cdot t_m(i))}{300} \quad (4.2.1)$$

onde o tempo (segundos) de arribo $t_m(i)$ do i -ésimo byte da seqüência de pacotes de transporte, é dado por :

$$t_m(i) = \frac{PCR(i'')}{27 \text{ MHz}} + \frac{i - i''}{\text{taxa de transporte}} \quad (4.2.2)$$

onde “ i'' ” representa o byte contendo o último bit do PCRB e o índice “ i ” representa qualquer byte da seqüência de pacotes. A taxa de transporte é definida pela seguinte equação:

$$\text{taxa de transporte}(i) = \frac{(i_b - i_a) \cdot (27 \text{ MHz})}{PCR(i_b) - PCR(i_a)} \quad (4.2.3)$$

onde “ i_a ” constitui o índice do último byte do PCRB anterior ao consecutivo PCRB contendo o byte “ i_b ”. ($i_a < i_b$).

Program clock reference extension (PCRE), é um campo formado pelos 9 últimos bits do PCR cujos valores representam o número de períodos do relógio de 27 MHz. Os valores do PCRE vão de 0 até 299. Cada vez que o PCRE completa um ciclo retornando a zero depois de atingir 299, o valor do PCRB aumenta em uma unidade. O PCRE é definido na seguinte equação:

$$PCRE(i) = INT((27 \text{ MHz}) \cdot (t_m(i))) - (300 \cdot (PCR_B(i))) \quad (4.2.4)$$

onde INT é uma função que devolve o valor inteiro mais próximo. O valor final PCR é definido pela seguinte equação:

$$PCR(i) = (300 \cdot PCR_B(i)) + PCRE(i) \quad (4.2.5)$$

A razão pela qual o PCR é codificado através do PCR_B, deve-se ao fato que esse valor constitui a referência para o decodificador dos tempos especificados nos valores PTS (33 bits) e DTS (33 bits) que estão codificados dentro dos pacotes PES. Esses três valores são codificados em base a um mesmo relógio de 90 KHz.

A transmissão contínua de PCR's assegurará por tanto um adequado sincronismo em cada nível de empacotamento para cada tipo de pacote que é decodificado. Na Figura 4.2.5 apresenta-se o processo de recepção dos pacotes e os tempos designados em cada etapa de decodificação.

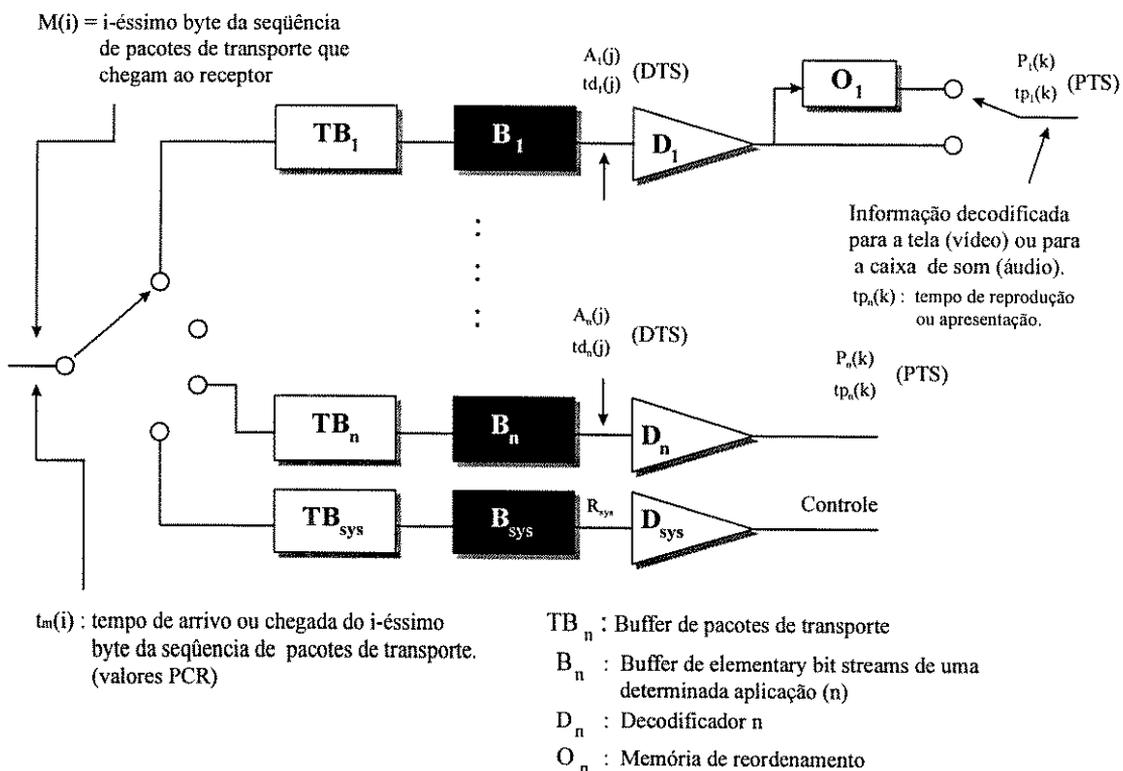


Figura 4.2.5. Sistema de recepção e decodificação dos pacotes de transporte

A Figura 4.2.6 mostra o diagrama de blocos do circuito PLL que ajusta a frequência do relógio no receptor a partir dos valores PCR que são recebidos.

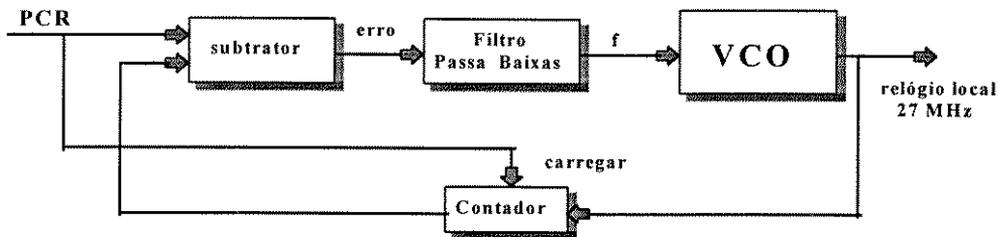


Figura. 4.2.6. Circuito PLL para ajuste da frequência do relógio local

O funcionamento do circuito da Figura 4.2.6 é descrito a seguir. Quando os valores PCR chegam ao receptor, entram num circuito subtrator o qual faz a comparação com o valor do contador local que está em unidades de 90 KHz e 27 MHz igual ao PCR. O subtrator faz a diferença entre ambos os valores para obter o valor de erro. A seqüência contínua de erros entra num filtro passa-baixas que tem um determinado ganho a fim de que o valor da tensão obtida na sua saída sirva para se fazer o controle da frequência do VCO (oscilador controlado por tensão). O sinal de saída do VCO constitui a frequência do relógio local de 27 MHz. Finalmente o valor do PCR é carregado no registro do contador local a fim de fazer a sincronização com o contador do PCR localizado no transmissor.

- **Splice countdown**, é um campo de 8 bits, que pode representar valores positivos e negativos. Um valor positivo representa o número de pacotes de transporte com o mesmo PID que precisa ser recebido antes de chegar a um **ponto de união**. Um ponto de união se define como o lugar na seqüência de pacotes onde pode ser inserida outra seqüência levando diferentes *elementary bit streams* de alguma outra fonte que represente algum programa local, uma propaganda ou dados.

O último pacote da seqüência do programa original que é transmitido antes do ponto de união leva o **splice countdown** com valor "00000000" e o último byte do *payload* do mesmo pacote corresponde ao último byte de uma unidade de acesso.

O *payload* do primeiro pacote da seqüência que é inserida, levará os primeiros bytes de um ponto de acesso. Quando o **splice countdown** leva um valor negativo (-n), indica que constitui o n-ésimo pacote da nova seqüência depois do ponto de união.

A Figura 4.2.7 mostra o diagrama de blocos de um sistema de inserção de programas, gerado a partir de uma fonte de *elementary bit streams*. A inserção de um programa é feita primeiro extraíndo-se os *elementary bit streams* do programa que vai ser modificado e a correspondente *program map table* que como veremos a seguir leva a informação dos respectivos PID de cada *elementary bit streams* que forma o programa. Em seguida, a *programa map table* do programa original é trocada por aquela do programa a ser inserido.

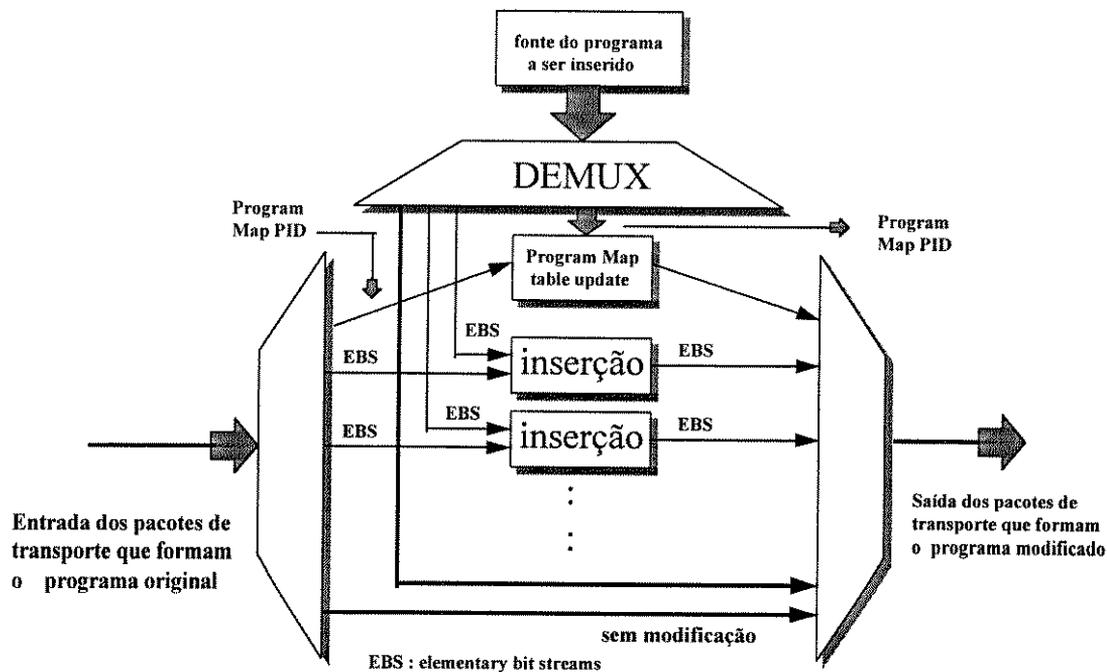


Figura. 4.2.7. Arquitetura de um sistema de inserção de programas

- **Transport private data length**, é um campo de 8 bits que especifica o número de *private data bytes* nos campos consecutivos.
- **Adaptation field extension length**, é um campo de 8 bits que indica o tamanho da extensão do *adaptation field* nos campos consecutivos.
- **Stuffing byte**, é um valor fixo de oito bits iguais a “11111111” que podem ser inseridos pelo codificador.

4.3.- Sistema de Multiplexagem

O sistema de multiplexagem é encarregado no primeiro nível, da multiplexação dos *elementary bit stream* agrupados em pacotes de transporte para a formação de um determinado programa.

No nível superior seguinte de multiplexagem, cada programa é multiplexado para a formação da seqüência final de programas, os quais serão transmitidos para cada receptor. O número de programas transmitidos dentro de um único canal de 6Mhz dependerá do grau de compressão alcançado e do modo de resolução da imagem que está sendo transmitida.

4.3.1.- Formação de Programas

Um programa é formado pela seqüência de pacotes de transporte multiplexados levando os correspondentes *elementary bit streams* que fazem parte do programa. Além disso, junto com a seqüência de pacotes que formam o programa é multiplexado um pacote de controle especificando os respectivos PID dos pacotes que formam o programa. Esses pacotes de especificação e controle são denominados como **PSI** (*program specific information*). Dentro do *payload* desses pacotes encontra-se a *program map table*.

A *program map table* leva informação dos PIDs de cada pacote de transporte que forma o programa e daqueles que levam os campos de sincronização PCR.

O primeiro campo do *elementary bit stream* que leva a *program map table* é constituído pelo *pointer field* que indica ao decodificador que o *elementary bit stream* formando o *payload* do pacote atual de transporte está levando o primeiro byte correspondente a um PSI. Na Figura 4.3.1 mostramos o diagrama de blocos de multiplexagem para a geração de um programa.

De acordo com a Figura 4.3.1 note-se que vários programas podem ser formados de diferentes combinações de *elementary bit streams* adicionando-se tão somente a respectiva *program map table* que identifique cada um deles como parte do programa.

Na documentação emitida pelo ATSC especifica-se que o intervalo máximo de transmissão de uma *program map table* deve ser de 400ms.

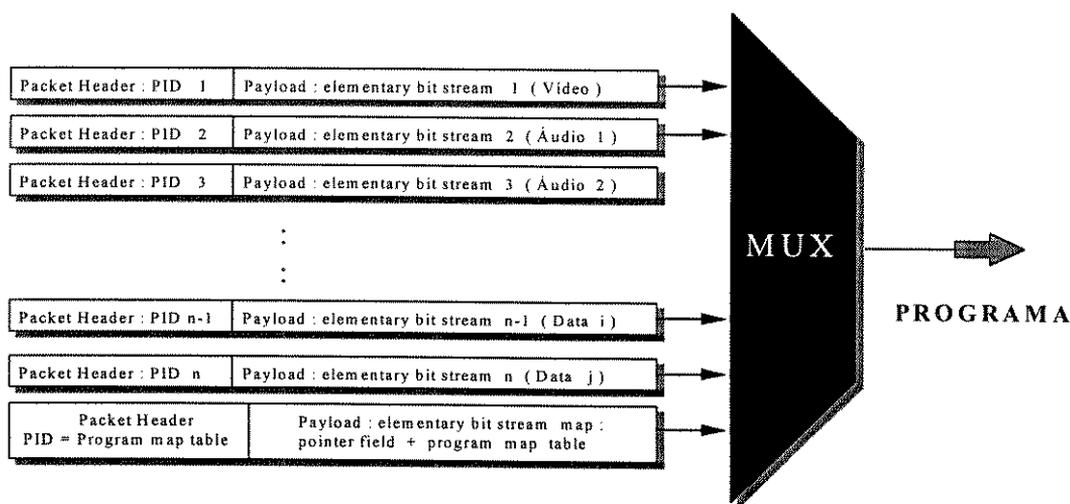


Figura. 4.3.1 Sistema de multiplexação para a formação de programas

Na Tabela 4.3.1 mostra-se a definição dos PID's designados a diferentes *bit stream*, enquanto que na Tabela 4.3.2 mostra-se um exemplo para o caso do programa 52.

Nome	Valor do PID	Descrição
PMT PID	base PID + 0x0000	PID do <i>bit stream</i> levando o <i>program map table</i> do programa
Vídeo PID	base PID + 0x0001	PID do <i>bit stream</i> levando a informação de vídeo do programa
PCR PID	base PID + 0x0001	PID do <i>bit stream</i> levando a informação de vídeo e o valor PCR
Áudio (A) PID	base PID + 0x0004	PID do <i>bit stream</i> levando a informação primária de áudio para o programa. Isto pode ser um serviço de áudio CM ou ME definidos no capítulo 2.
Áudio (B) PID	base PID + 0x0005	PID do <i>bit stream</i> levando informação de áudio que pode ser um serviço CM ou ME. No caso que o áudio (A) seja um serviço ME, então o áudio (B) leva o serviço de diálogo D para a formação do serviço completo.
Data PID	base PID + 0x000A	PID do <i>bit stream</i> que leva os dados do programa

Tabela 4.3.1 Definição dos valores dos PID's para diferentes bit stream

Nome	Valor PID
base PID	0x0340
PMT PID	0x0340
Vídeo PID	0x0341
PCR PID	0x0341
Áudio (A) PID	0x0344
Áudio (B) PID	0x0345

Tabela 4.3.2 Exemplo de PID's designados ao programa 52

4.3.2.- Formação da seqüência de programas

Cada programa que é gerado através do primeiro nível de multiplexagem entra num segundo nível de multiplexação para geração da seqüência final de programas. Essa última seqüência será transmitida.

Em cada seqüência de programas ,gerada em cada ciclo de multiplexação é transmitido um programa de controle constituído por pacotes de transporte cujos **PIDs** são iguais a **zero**, indicando que no seu respectivo *payload* levam o que se denomina como a **program association table**.

A **program association table** leva os PIDs das respectivas *program map table* de cada programa assim como o número associado a cada um deles. Cada seção **0** dentro da *program association table* constitui o início das especificações de um determinado programa. Na documentação emitida pelo ATSC especifica-se que o intervalo máximo entre duas seções 0 deve ser de 100 ms.

O programa identificado com o número "0" não leva a *program map table*, devido a ser reservado para a transmissão de uma **network information table** (NIT),

que fornece informação relativa ao presente e futuros programas que serão transmitidos. O *bit streams* de uma *network information table* constitui também um PSI.

Outro tipo de PSI é constituído pelos *elementary bit streams* cujo PID=1, o que indica que a informação contida no pacote é particular (**Conditional Access Table-CAT**). Na Figura 4.3.2 mostra-se o sistema de multiplexação para a geração da seqüência de programas a ser transmitidos.

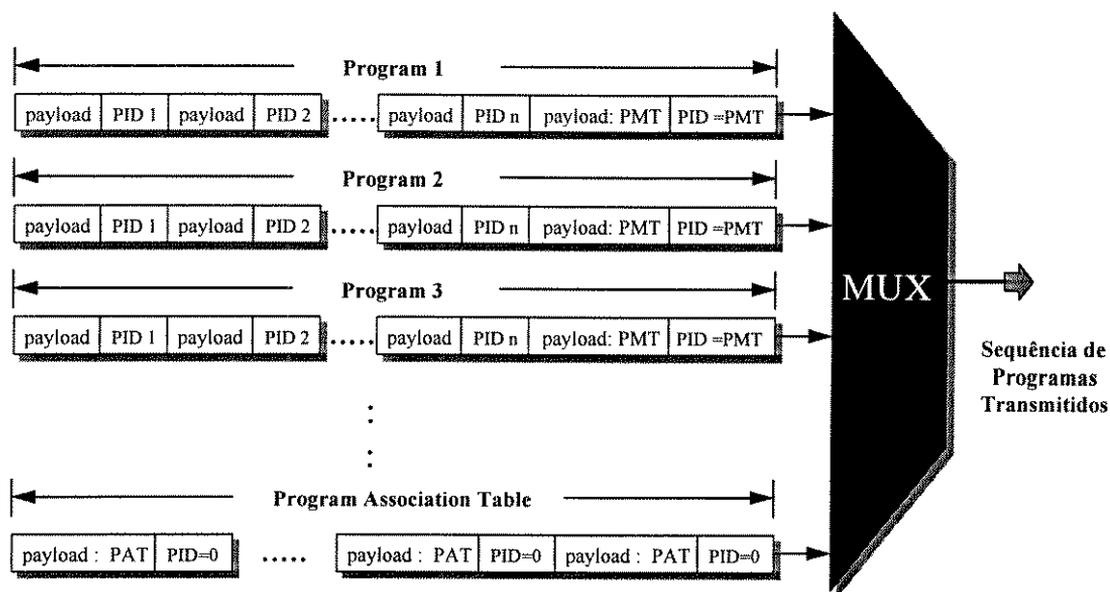


Figura. 4.3.2. Sistema de Multiplexação de Programas

4.4.- Sistema de Demultiplexação

A demultiplexação da seqüência de pacotes transmitidos, é feita em passos consecutivos de demultiplexação até se chegar aos *elementary bit streams* correspondentes às aplicações de áudio e vídeo que formam parte do programa selecionado pelo usuário.

A seguir faremos uma a descrição de cada passo realizado pelo decodificador até chegar aos diferentes *elementary bit streams* que formam o programa.

- O primeiro passo consiste em tomar da seqüência de programas transmitidos, a correspondente *program association table* identificado com PID=0. Isso é feito a fim de selecionar o PID da *program map table* que corresponda ao programa selecionado pelo usuário.
- O segundo passo consiste na obtenção da *program map table* a partir do PID obtido no primeiro passo. Com essa informação, o sistema faz a demultiplexação dos diferentes pacotes de transporte através do seus respectivos PIDs, até a obtenção dos *elementary bit streams* que pertencem ao programa selecionado.

Na Figura 4.3.3 mostra-se o diagrama de blocos do processo de demultiplexagem, e na Figura 4.3.4 mostramos a seqüência da informação que é consecutivamente demultiplexada até se chegar aos *elementary bit streams* de cada aplicação.

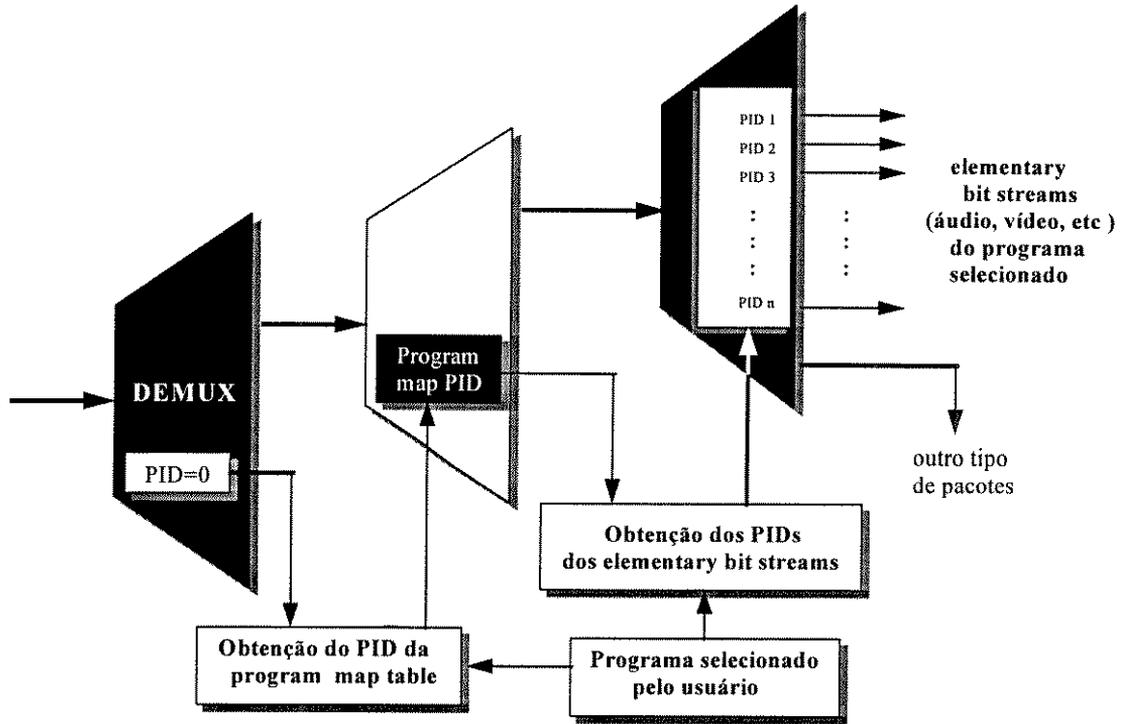


Figura. 4.3.3. Diagrama de blocos do sistema de demultiplexagem

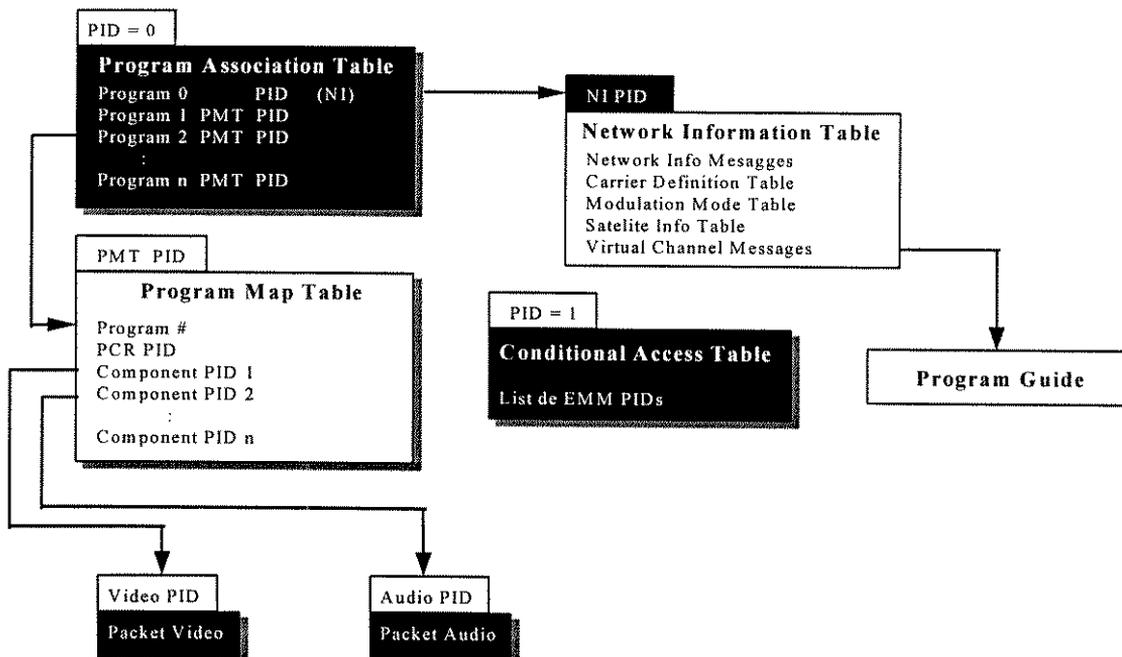


Figura. 4.3.4. Diagrama de blocos da seqüência de demultiplexação da informação

4.5.- Comentários e conclusões finais

Neste capítulo fornecemos os aspectos mais importantes da camada de transporte e de multiplexagem que, sob o formato **MPEG-2**, foi adotado pelo sistema americano da *Grande Aliança* para a transmissão dos sinais digitais de **HDTV**.

A taxa de bits obtida na saída do sistema de multiplexagem varia de acordo com meio de transmissão, considerando **19.38 Mbs** (americano) para transmissão terrestre, **27-32 Mbs** para transmissão a cabo e até **54 Mbs** para transmissão via satélite.

Por outro lado a utilização dos pacotes de transporte de um comprimento fixo, facilita em muitos aspectos a funcionalidade de todo o sistema no que se refere a sincronização, acesso condicional, inserção de programas, prioridade, proteção a informação particular e detecção de erros. Todas essas facilidades oferecidas pelo formato MPEG-2 foram levados em conta para a avaliação do mesmo como sistema de transporte e multiplexagem de informação de HDTV.

A forma de multiplexagem utilizada em escalas e níveis de multiplexação constitui uma maneira flexível de se criar e adicionar novos programas para então serem inseridos na seqüência que é finalmente transmitida. Isso melhora o aproveitamento do canal devido ao fato de que se pode aumentar facilmente o número de programas multiplexados, no caso incrementando-se a largura de faixa do canal de transmissão.

Finalmente podemos dizer que a compatibilidade com outros protocolos de comunicações como **ATM** é assegurada mediante a facilidade de partição de um pacote de transporte (188 bytes) em 4 células desse formato que se constitui num dos mais importantes protocolos da atualidade e muito provavelmente também do futuro.

Capítulo 5

Sistema de Transmissão

5.0.- Introdução

O sistema de transmissão constitui finalmente a interface entre o transmissor e o canal através do qual o sinal de HDTV será transmitido. Dessa forma, os dados digitais conformados por uns e zeros são transmitidos mediante um adequado processo de modulação, que é realizado de acordo com o tipo de canal que está sendo utilizado.

O tipo de modulação a se aplicar a um sinal digital de HDTV deve levar em conta o meio através do qual o sinal será transmitido bem como a proteção do mesmo ante a ocorrência de possíveis erros. Os erros de transmissão são produzidos geralmente devido às deficiências que o meio possa apresentar, à interferência com canais adjacentes e ante uma possível perda de sincronismo entre o transmissor e o receptor.

Dessa forma, em todos os sistemas HDTV desenvolvidos na Europa (SPECTRE, DIAMOND, STERNE, HD-DIVINE), foi considerado o sistema de modulação digital **OFDM/COFDM** para a transmissão terrestre de sinais de televisão. A característica principal do sistema **OFDM** (*Orthogonal Frequency Division Multiplexing*) consiste em que o sinal é transmitido através de um grande número de portadoras digitalmente moduladas onde cada uma delas é ortogonal a outra. As portadoras podem ser moduladas utilizando os sistemas **DQPSK, 16-QAM, 32-QAM**, etc. O sistema **COFDM** é um sistema OFDM com codificadores adicionais para a correção de possíveis erros de transmissão.

Nos Estados Unidos, as propostas apresentadas por *Digicipher of General Instruments* (1991), *Advanced Digital Television ADTV*, e *Channel Compatible Digicipher CCDC*, utilizam as modulações 16-QAM e 32-QAM, com uma codificação adicional *Reed Solomon* para a correção de erros. Por outro lado, o *Digital Spectrum Compatible DSC-HDTV* (1991) utiliza o sistema 2/4 VSB junto também com codificador *Reed Solomon*.

No sistema de modulação proposto pela *Grande Aliança* (1994) foi decidido considerar o sistema **8-VSB** (*Vestigial Side Band*) para transmissão terrestre e **16-VSB** para transmissão por cabo, em conjunção com os codificadores de treliça e *Reed Solomon*.

Atualmente, esse sistema junto com o COFDM constituem as alternativas mais importantes para a padronização do sistema de transmissão de um sinal HDTV no mundo.

Neste capítulo apresentaremos os aspectos mais relevantes desses dois sistemas, apresentando as vantagens e desvantagens que cada um deles oferece e, finalmente, comparando seus desempenhos.

5.1.- Sistema de Modulação OFDM/COFDM

O sistema de modulação **OFDM** (*Orthogonal Frequency Division Multiplexing*) foi desenvolvido na Europa para a transmissão de sinais HDTV. Esse sistema caracteriza-se pela utilização de um grande número de portadoras moduladas digitalmente onde cada uma delas é ortogonal a outra.

Os sistemas mais empregados na modulação de cada portadora são: 8-QAM, 16-QAM e DPSK. Dessa forma, os dados digitais seriais provenientes do sistema de transporte e multiplexagem são agrupados em dois, três ou quatro símbolos dependendo do tipo de modulação a ser utilizado. Por exemplo, no caso da modulação 16-QAM os dados são agrupados em grupos de quatro bits enquanto que para os sistemas 8-QAM e DPSK, são agrupados em grupos de três e dois bits respectivamente.

Cada grupo de bits dá origem a uma constelação de sinais que é modulada através de uma única portadora, para formar finalmente um sinal multiplexado por divisão de frequência que é representado pela seguinte equação:

$$y(t) = \sum_{n=0}^{N-1} (a_n \cos 2\pi f_n t + b_n \sin 2\pi f_n t) \quad 0 \leq t \leq T \quad (5.1.1)$$

onde T é a duração de cada símbolo, a_n e b_n são os componentes em fase e em quadratura do sinal modulado e f_n é a n-ésima portadora.

A ortogonalidade entre as portadoras é assegurado desde que o espectro de cada símbolo seja da forma $\frac{\text{sen}(x)}{x}$, e o espaçamento entre elas seja de $1/T$. Dessa forma, cada portadora é localizada nos pontos de cruzamentos em zero dos espectros.

Para a formação dos grupos de constelações, a seqüência serial de bits é transformada de série a paralelo (S/P), onde cada símbolo gerado ocupa uma faixa menor do espectro devido ao fato de que a duração dos mesmos torna-se maior do

que a duração de cada dígito binário. Cada símbolo é modulado paralelamente através da sua respectiva portadora para que finalmente seja gerado o sinal OFDM através da soma de cada um deles.

À medida que o número de portadoras cresce, o espectro do sinal resultante torna-se cada vez mais plano. A Figura 5.1.1 mostra o diagrama de geração do espectro de um sinal OFDM.

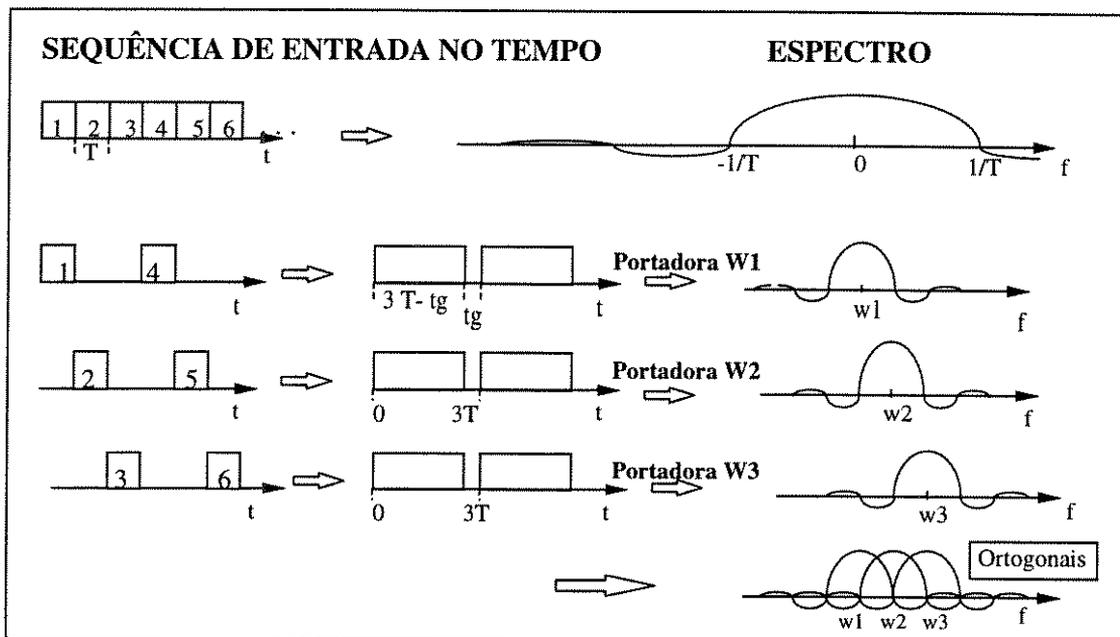


Figura 5.1.1. Geração do Espectro de um sinal OFDM com três portadoras

5.1.2 Geração de um Sinal OFDM usando a IFFT/FFT

O grande número de portadoras utilizado para a geração de um sinal OFDM torna necessária a utilização de um banco de osciladores senoidais coerentes exatamente calibrados, a fim de evitar que a ortogonalidade do sinal resultante seja afetada por um desvio em relação aos valores exatos das portadoras. O número de osciladores a se usar dependerá do número de portadoras utilizadas para a geração do sinal. Isso torna a implementação do sistema OFDM extremamente complexo, tendo sido necessário a procura de um novo método que possa diminuir tal complexidade.

Dessa forma, se observarmos a equação 5.1.1 notamos que a forma da expressão do sinal OFDM é igual à parte real da **Transformada Discreta de Fourier Inversa (IDFT)** [17] da informação original. Isso constitui uma característica importante devido à possibilidade da implementação ser feita através de um algoritmo que gere um sinal multiplexado em frequência, com as mesmas características que as da equação 5.1.1.

O procedimento para a geração do sinal resultante mediante esse método é dado a seguir:

- Inicialmente, a sequência de dados de entrada é dividida em grupos de k bits formando a sequência $(d_0, d_1, d_2, \dots, d_{N-1})$, onde N é o número de constelações ou portadoras a serem utilizadas pelo sistema. Cada valor d_n representa um número complexo da forma $d_n = a_n + jb_n$.
- Uma vez que os valores d_n são gerados, calcula-se a IFFT da informação mediante a seguinte equação:

$$D_m = \sum_{n=0}^{N-1} d_n \cdot \exp(-j2\pi nm / N) = \sum_{n=0}^{N-1} d_n \cdot \exp(-j2\pi f_n t_m / N) \quad (5.1.2)$$

onde $m=0,1,\dots,N-1$, $f_n = n/(N\Delta t)$, $t_m = m\Delta t$ e Δt é a duração do símbolos de entrada d_n .

A partir dos N números complexos gerados mediante a equação 5.1.2 obtém-se o vetor $(D_0, D_1, D_2, \dots, D_{N-1})$.

A parte real do vetor D_m é dado pela seguinte equação:

$$Y_m = \text{Re}\left\{ \sum_{n=0}^{N-1} (a_n + jb_n) \cdot \exp(-j2\pi f_n t_m) \right\} \quad (5.1.3)$$

$$Y_m = \sum_{n=0}^{N-1} (a_n \cos 2\pi f_n t_m + b_n \text{sen} 2\pi f_n t_m), \quad m=0,1,\dots,N-1$$

A implementação da transformada é feita através do algoritmo da **IFFT**, a fim de se aumentar a eficiência do sistema.

- Finalmente, as amostras Y_m são enviadas a um filtro passa-faixa em intervalos de tempo Δt , a fim de gerar o sinal OFDM por multiplexação em frequência e representado na equação 5.1.1. Para isso, é necessário que após o cálculo da IFFT, seja feita a conversão paralelo para série (P/S) das amostras Y_m . No entanto, antes que as amostras sejam enviadas para o filtro é de grande importância a inserção cíclica de um intervalo de guarda, a fim de eliminar a interferência entre símbolos, causada pela distorção devido à propagação por múltiplos caminhos. Dessa forma, cada símbolo OFDM é precedido por uma extensão periódica do próprio sinal, tal como é mostrado na Figura 5.1.3. A duração total de um símbolo seria igual a $T_s = T + T_g$, onde T é a duração efetiva do símbolo, e T_g é a duração do intervalo de guarda.

Geralmente, a duração do intervalo de guarda não ultrapassa $T/4$. Quando o intervalo de guarda é maior que a resposta de impulso do canal ou maior do que os retardos devido aos múltiplos caminhos, a interferência entre símbolos pode ser eliminada [16].

O número de portadoras a se utilizar depende da largura de faixa do canal, da taxa de dados e da duração efetiva dos símbolos. Além disso, é preferível que o número de portadoras seja sempre múltiplo de dois desde que seja utilizada a IFFT/FFT para geração e recepção do sinal OFDM. Os valores mais utilizados são 512, 1024, 2048 e assim por diante.

O tempo empregado para aplicação no filtro dos N terminais com d_n moduladores é igual a $N\Delta t$, tal como mostra a seguinte equação:

$$y(t) = \sum_{n=0}^{N-1} (a_n \cos 2\pi f_n t + b_n \text{sen } 2\pi f_n t) \quad 0 \leq t \leq N\Delta t \quad (5.1.4)$$

Na Figura 5.1.2, mostramos o diagrama de blocos de todo o processo de geração do sinal OFDM mediante a utilização da IFFT/FFT. Note que no receptor se faz exatamente as operações inversas usadas na geração do sinal. Os blocos finais *Up/Down Converter* fazem o deslocamento do espectro do sinal resultante de acordo com a faixa designada para a transmissão de um determinado sinal de televisão.

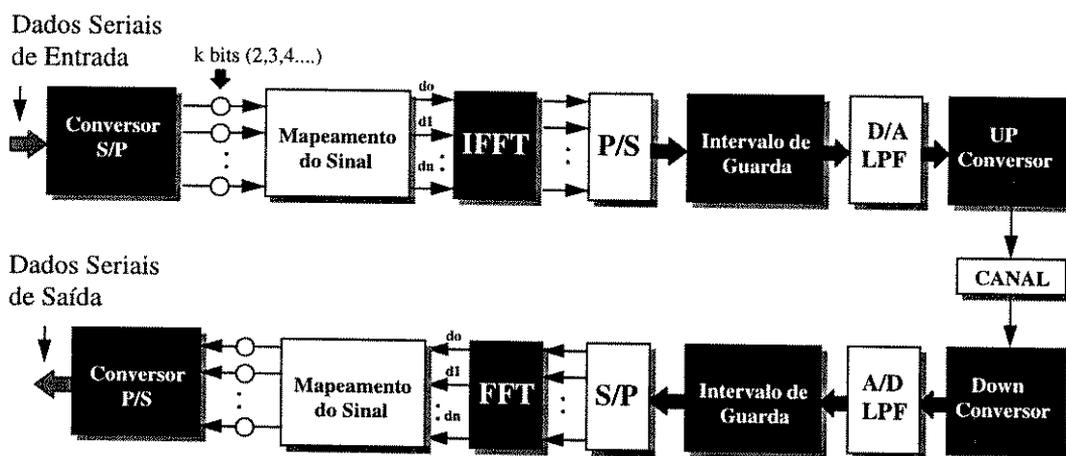


Figura 5.1.2. Diagrama de blocos do Sistema Gerador de um Sinal OFDM usando IFFT/FFT

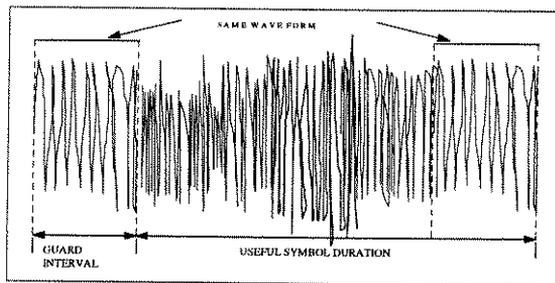


Figura 5.1.3. Forma de Onda de um Sinal OFDM básica

5.1.3 Geração de um Sinal COFDM

O uso de várias portadoras na geração de um sinal OFDM permite mudar o espectro do mesmo com desvanecimento seletivo, em pequenas sub-faixas de frequências com desvanecimento não seletivo em cada uma delas.

O desvanecimento é uma forma de distorção apresentado no sinal devido a uma mudança da função de transferência do canal que afetam certas componentes de frequência. O desvanecimento do tipo seletivo aparece quando a distorção afeta de forma desigual todo o espectro, enquanto que o desvanecimento do tipo não seletivo apresenta-se quando todas as componentes do espectro ou de uma determinada faixa são afetadas uniformemente.

Como se mencionou anteriormente a forma do sinal OFDM divide o espectro do sinal de televisão em pequenas sub-faixas de frequências que apresentam um desvanecimento não seletivo. Isso torna o sinal apropriado para aplicação da codificação em treliça [17] em cada sub-faixa.

Dessa forma, após a formação dos grupos de k bits é inserido um **codificador de treliça** que gera palavras de código de $k+1$ bits por cada grupo. O codificador de treliça permite corrigir possíveis erros na transmissão de uma determinada faixa do espectro.

Finalmente, uma vez que o codificador é inserido no sistema, o sinal resultante é denominado como sinal **COFDM** (*Coded Orthogonal Frequency Division Modulation*).

Na Figura 5.1.4 mostra-se um codificador em treliça para um sistema 8-PSK, onde cada símbolo formado por dois bits, é mapeado numa palavra código de três bits, gerados pelo codificador.

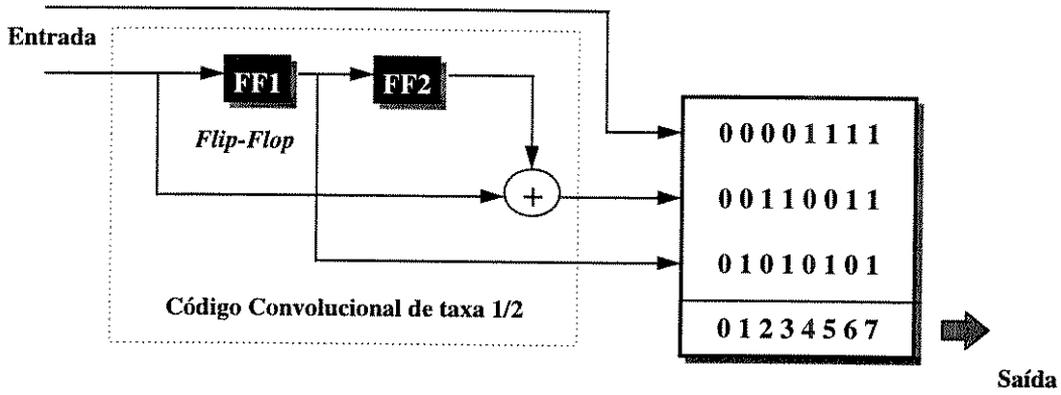


Figura 5.1.4. Codificador de Treliça para 8-PSK

Na Figura 5.1.5 mostra-se a treliça do codificador mostrado na Figura 5.1.4.

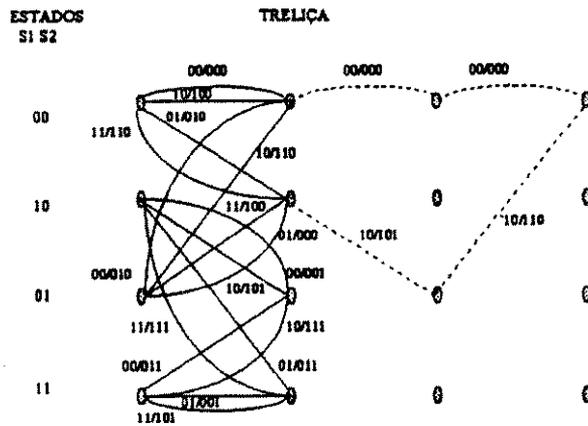


Figura 5.1.5. Treliça para o Codificador de 8-PSK

O diagrama de blocos do sistema COFDM é mostrado na Figura 5.1.6, onde é feita a inserção do codificador de treliça no sistema OFDM da Figura 5.1.2.

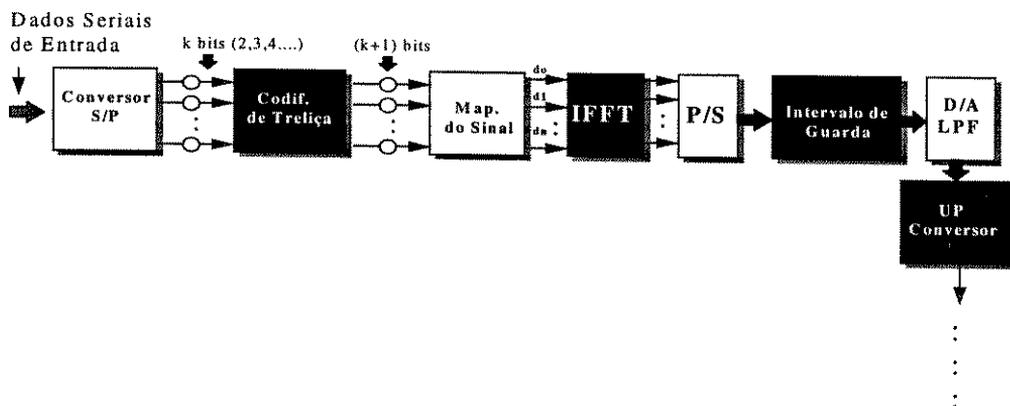


Figura 5.1.6. Diagrama de blocos do sistema de Geração de um Sinal COFDM

5.1.4. Sistemas OFDM/COFDM propostos na Europa

No continente europeu foram desenvolvidos diversos tipos de sistemas OFDM/COFDM para a transmissão de um sinal de televisão HDTV. A seguir descreveremos alguns desses sistemas com suas características principais.

- Sistema **SPECTRE HDTV** : utiliza o sistema OFDM com **400** portadoras, as quais podem ser programadas mediante os sistemas **QPSK, 8-PSK, 16-PSK e 16-QAM**, adicionando-se ao sistema um codificador de canal **Reed Solomon (255,239)** para a correção de uma longa seqüência de possíveis erros. Geralmente o codificador **Reed Solomon** é colocado antes do codificador de treliça tal como descreveremos mais adiante.
- Sistema **DIAMOND** : utiliza o sistema OFDM com a modulação **64-QAM** para uma taxa de transmissão de **30/60 Mbps** através de canais **UHF/VHF**.
- Sistema **STERNE** : utiliza uma largura de faixa de **8 MHz** para a transmissão do sinal de televisão através de uma taxa de **30 Mbps**. O sistema COFDM empregado utiliza a modulação **QAM**, codificação de treliça, um “**data interleaver**” (intercalador de dados), e um **Reed Solomon** para a correção de erros.
- Sistema **HD-DIVINE HDTV** : utiliza o sistema OFDM com a modulação **16 QAM** que inclui **512** portadoras para uma taxa de transmissão de **24 Mbps** assim como um **Reed Solomon (224,208)** para a correção de erros.

No Canadá, está sendo desenvolvido no **CRC (Communications Research Center)** um sistema **COFDM** para um canal de transmissão de **6-MHz (COFDM-6)** com a participação de *broadcasters* brasileiros e cujos parâmetros foram definidos da seguinte forma:

- Número de Portadoras : **6120**, requerendo **8Kb** de memória para a FFT.
- Modulação (constelação): **64 QAM**
- Intervalo de Guarda : **64 μs**
- Codificação de Canal : **Reed Solomon RS(255,239)** e **2/3 TCM (Trellis Coding Modulation)**.
- Taxa da transmissão : **19.12 Mbps**.

5.1.5 Vantagens e Desvantagens do Sistema COFDM

A seguir, apresentaremos as principais vantagens e desvantagens do sistema de transmissão COFDM que provam seu melhor desempenho sobre os outros sistemas de transmissão.

- O sistema COFDM é mais imune ao ruído impulsivo do que os sistemas com uma única portadora devido à longa duração dos símbolos.
- O COFDM apresenta maior robustez diante de possível distorção do sinal devido aos múltiplos caminhos.
- O conjunto de portadoras pode ser modulado de acordo com os diferentes tipos de modulação digital disponíveis, tornando o sistema muito flexível.
- A emissão de um sinal COFDM em sistemas HDTV pode ser feito de duas formas:
 1. Através de um conjunto de repetidoras sincronizadas no tempo e na frequência com um transmissor principal. Esse método é denominado SNF (*Single Network Frequency*). Através dele pode-se cobrir um extensa área utilizando-se uma única frequência. A redução do retardo de propagação pode ser feita distribuindo o sinal do transmissor às repetidoras através de enlaces por satélite, microondas ou fibras ópticas.
 2. No outro método as repetidoras recebem os sinais do transmissor principal de alta potência e retransmitem-os com a mesma frequência sem necessidade de sincronismo. No entanto, essa forma de emissão não é muito atraente para transmissão simultânea de programas de HDTV e televisão convencional devido às interferências com canais NTSC/PAL.
- Uma desvantagem do sistema COFDM constitui a alta sensibilidade do mesmo aos pequenos erros nas frequências das portadoras, comprometendo a ortogonalidade das mesmas. Por outro lado a inserção do intervalo de guarda elimina a interferência entre os símbolos, mas introduz redundância no espectro.

O desempenho do sistema COFDM será mostrado no final do capítulo em comparação com o sistema 8-VSB da Grande Aliança que estudaremos em seguida.

5.2. Sistema de Modulação 8-VSB - 16-VSB

A Grande Aliança dos Estados Unidos, propôs para a transmissão de um sinal HDTV o sistema 8-VSB (*Vestigial Sideband*) para transmissão terrestre e 16-VSB para transmissão de televisão a cabo. Além disso, a G.A. propôs a utilização do sistema “*simulcast*”, o qual considera a transmissão simultânea de dois programas de televisão, onde um deles é transmitido através do sistema convencional NTSC (canal 6 MHz) e o outro através de um sinal comprimido HDTV num adicional canal também de 6 MHz[1].

A taxa de transmissão de símbolos considerada pelo sistema é de 10.76 Msímbolo/s, os quais são modulados através das técnicas 8-VSB e 16-VSB. O desempenho do sistema é avaliado através do desempenho frente a interferência de canais NTSC adjacentes. A Grande Aliança levou em conta as seguintes considerações :

- **Reduzida taxa de erro de bits.**
- **Eficiência na utilização do canal.**
- **Redução da potência de transmissão dos sinais digitais em comparação com os sinais analógicos.**

Esses fatores levam ao uso da codificação concatenada, que utiliza dois tipos de códigos de correção de erros. O primeiro, é um codificador externo que está constituído por um *Reed Solomon* que é utilizado para a correção de uma longa seqüência de erros. O segundo é um codificador interno que é constituído por um codificador de treliça seguido por um intercalador que faz o reordenamento dos dados a fim de decorrelacionar os erros de transmissão. O codificador de treliça introduz palavras códigos para a correção de erros aleatórios de transmissão.

Na Figura 5.2.1 mostra-se o diagrama de blocos do sistema de transmissão 8-VSB. O sistema é dividido principalmente num bloco de processamento de dados e num bloco de processamento do sinal que será finalmente transmitido.

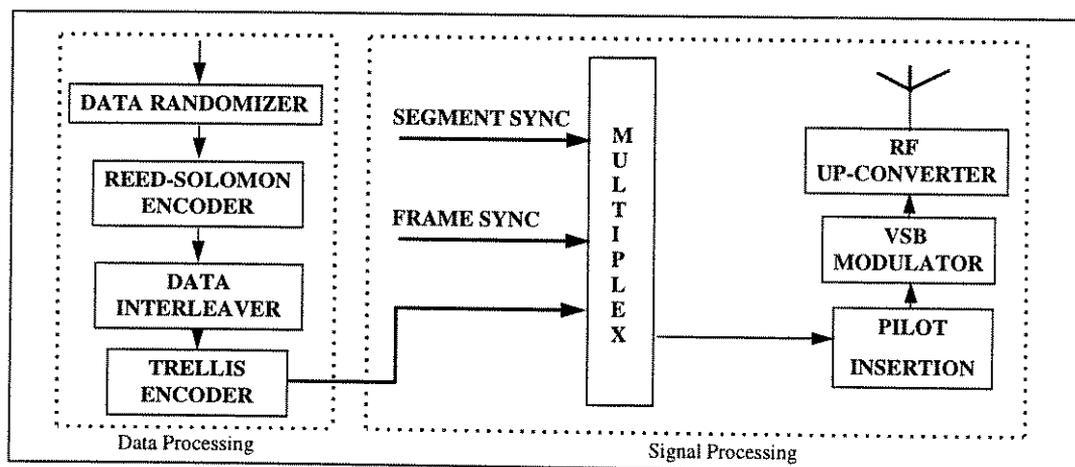


Figura 5.2.1. Diagrama de blocos do sistema 8-VSB

Os dados de entrada no sistema estão no formato dos pacotes de transporte MPEG-2 estudado no capítulo anterior. A taxa de transmissão desses pacotes representa 19.38 Mb/s.

- sistema faz a transmissão de dados através da seqüência de *Data Frames*. Cada *Data Frame* consiste de dois *Data Fields*, onde cada um deles é formado por 313 *Data Segments*. O primeiro *Data Segment* de cada *Data Field* é formado somente por informação de sincronismo (*Data Field Sync*). Cada um dos restantes

312 *Data Segments* leva a informação correspondente a um pacote MPEG-2 de 188 bytes, mais informação de correção de erro FEC (*Forward Error Correction*). Cada *Data Segment* é constituído por 832 símbolos, onde os 4 primeiros são transmitidos em forma binária e proveem a informação de sincronização do segmento. Esses primeiros símbolos correspondem ao byte de sincronismo do pacote MPEG-2. Os 187 bytes restantes do pacote representam 828 símbolos de transmissão.

Os 828 símbolos são transmitidos através de um sinal de 8 níveis, onde cada nível representa 3 bits, perfazendo um total de $828 \times 3 = 2482$ bits de transmissão por cada pacote de transporte contido em cada *Data Segment*.

A taxa de transmissão de cada *Data Segment* é de 12940 *Data Segment/s*, enquanto que a taxa de transmissão de *Data Frames* é de 20.66 *Data Frame/s*.

Na Figura 5.2.2 mostra-se o formato de transmissão da informação utilizada pelo sistema 8-16-VSB.

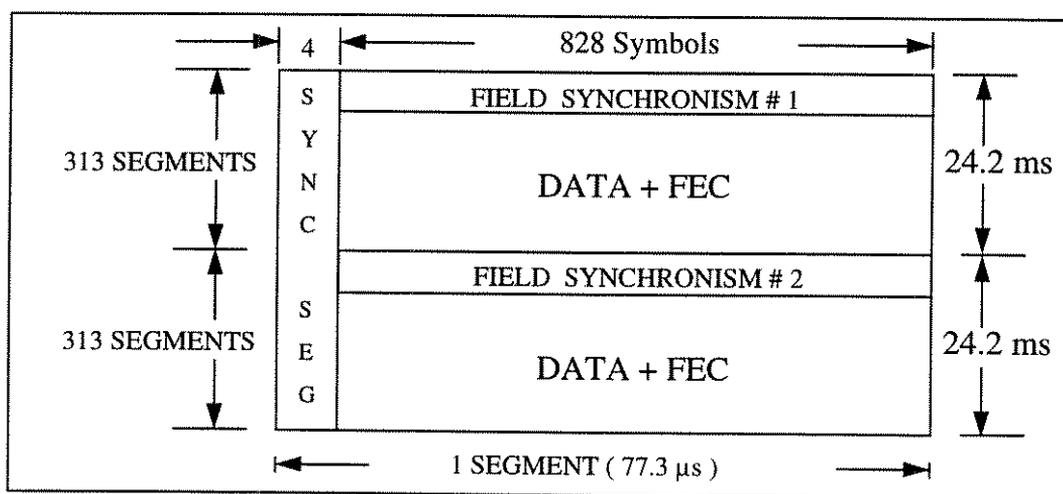


Figura 5.2.2. Data Frame VSB

A seguir vamos passar a descrever cada uma das partes que formam o sistema bem como a função que cumprem dentro do mesmo.

5.2.1 Embaralhador de Dados (*Data Randomizer*)

O primeiro bloco do sistema de transmissão 8-16-VSB está constituído pelo *Data Randomizer*. Esse sistema consiste de um circuito digital de registradores de deslocamento com 16 Flip-Flops que geram uma seqüência de bits que, mediante funções XORs fazem o embaralhamento dos dados de entrada constituídos pelos pacotes de transporte MPEG-2 (188 bytes). O embaralhamento

da informação não inclui o *Data Field Sync*, o *Data Segment Sync*, e os bytes de paridade inseridos pelo codificador *Reed Solomon*.

O polinômio PN (Pseudo Noise) gerador da seqüência de bits pseudo-aleatória é dado pela seguinte equação:

$$G_{(16)} = X^{16} + X^{13} + X^{12} + X^{11} + X^7 + X^6 + X^3 + X + 1 \quad (5.2.1)$$

O circuito de deslocamento utilizado para a geração dos bits pseudo-aleatórios é mostrado na Figura 5.2.3. O circuito faz um deslocamento de um bit em cada *byte clock* (cada 8 bits de dados) a fim de gerar o próximo byte pseudo-aleatório que será enviado a uma função XOR bit a bit com o próximo byte de dados.

A inicialização dos registros de deslocamento é feita durante o *Data Segment Sync*. A inicialização é feita carregando-se o número hexadecimal **F180_h** nos 16 Flip-Flops do circuito. Dessa forma, durante cada *Data Segment Sync* são levadas a "1" as saídas dos Flip-Flop's : X^{16} , X^{15} , X^{14} , X^{13} , X^9 e X^8 .

De acordo com a Figura 5.2.3 os bits D_x provenientes das saídas dos Flip-Flops são passados através de uma função XOR (bit a bit) junto com o byte atual de dados, para a geração do byte que será finalmente transmitido.

Os dados são embaralhados a fim de se dispersar a energia do sinal de transmissão através de todo o espectro de forma quase uniforme. Dessa maneira, reduz-se o efeito de interferência de canais adjacentes[1].

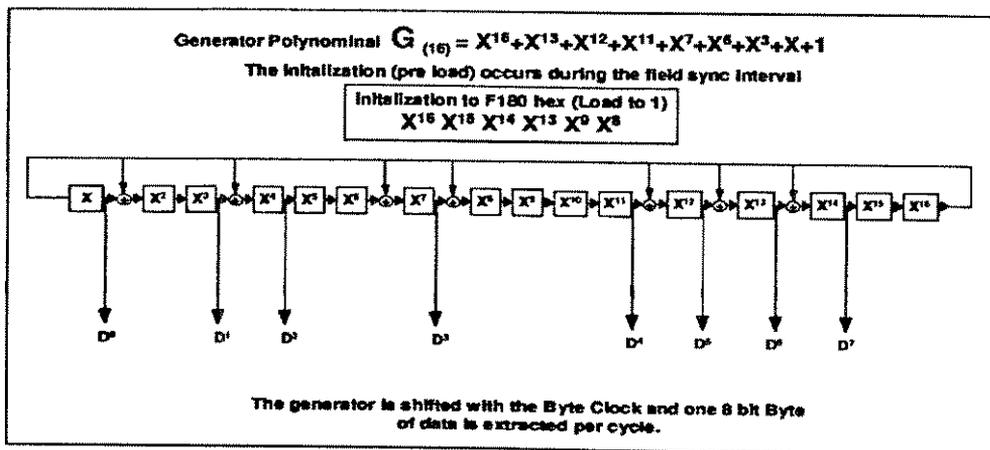


Figura 5.2.3. Circuito gerador polinomial de bits pseudo-aleatórios

5.2.2.- Codificação Reed Solomon

O codificador *Reed Solomon* utilizado no sistema VSB é do tipo $t=10$ (207,187), onde t indica o número máximo de erros que se pode corrigir através

das palavras código geradas pelo codificador. Dessa forma, são transmitidos 20 bytes adicionais aos 187 bytes de dados de cada pacote, para finalmente formar as palavras código (207,187).

O *Reed Solomon* é um codificador do tipo convolucional não binário, que é utilizado pelo sistema para a correção de uma longa seqüência de erros. Nesse caso, o código permite a correção de 10 bytes errados de informação através dos 20 bytes adicionais que são inseridos pelo codificador. Os bytes de entrada no codificador para esse caso provem do *Data Randomizer*.

O polinômio gerador dos bytes de paridade utilizado pelo *Reed Solomon* para esse caso é o seguinte:

$$G(x) = x^{20} + 152x^{19} + 185x^{18} + 240x^{17} + 5x^{16} + 111x^{15} + 99x^{14} + 5x^{13} + 220x^{12} + 111x^{11} + 150x^{10} + 69x^9 + 36x^8 + 187x^7 + 22x^6 + 226x^5 + 198x^4 + 121x^3 + 121x^2 + 165x + 174 \quad (5.2.2)$$

Na Figura 5.2.4 mostra-se o diagrama do circuito gerador dos bytes de paridade do código *Reed Solomon*. Note que primeiro são transmitidos os bytes de informação do pacote e no final transmitem-se os respectivos bytes de paridade.

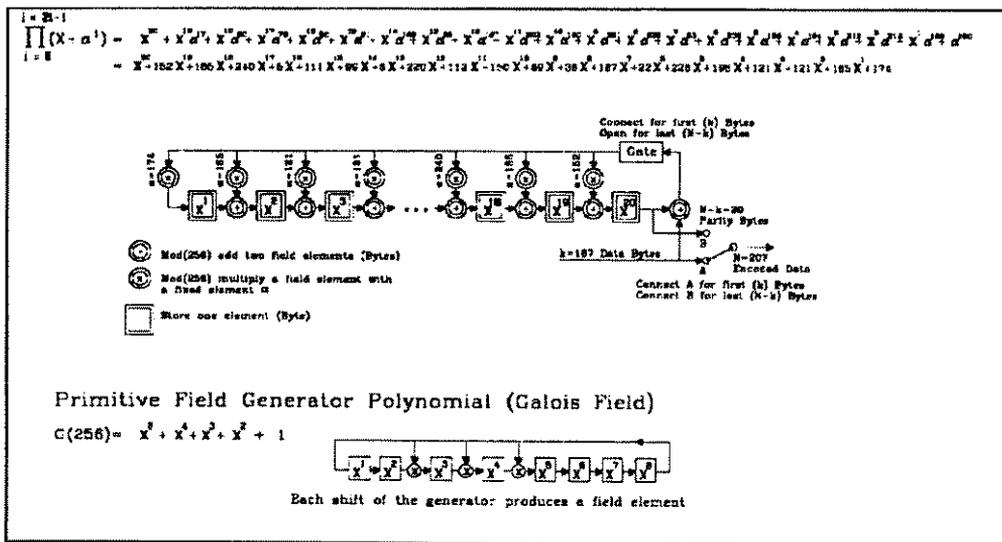


Figura 5.2.4. Codificador Reed Solomon (207,187)

5.2.3.- Intercalador (*Interleaver*)

O *intercalador* é um sistema de reordenamento de dados. Isso é realizado a fim de decorrelacionar uma longa seqüência de erros que possa acontecer durante a transmissão do sinal.

O intercalador utilizado no sistema VSB é mostrado na Figura 5.2.5, onde o grau de reordenamento dos dados é dado pelo fator $I=52$ que representa $1/6$ dos dados contidos num *Data Field* ($312/6$). O intercalador é sempre sincronizado com o primeiro byte de dados de um *Data Field*. O formato de implementação do *interleaver* mostrado na Figura 5.2.5 permite minimizar o retardo de transmissão comparado com outros sistemas de reordenamento[1].

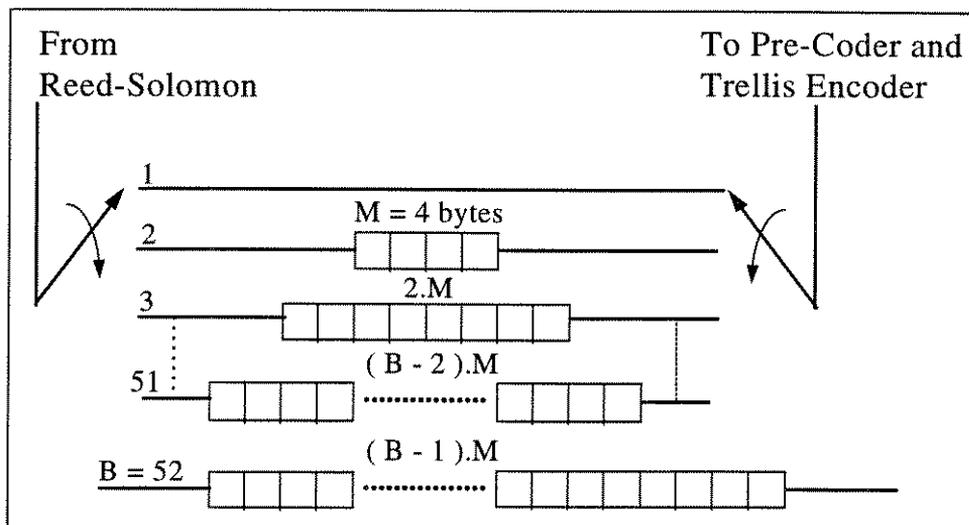


Figura 5.2.5. Intercalador ($I=52$) do Sistema VSB

Os dados provenientes do codificador *Reed Solomon* ingressam no *interleaver* a fim de mudar a ordem de transmissão dos bytes contidos na seqüência de pacotes de transporte. Dessa forma, se um byte de um determinado pacote é transmitido num instante de tempo t , o próximo byte a se transmitir no tempo $t+1$ não necessariamente pertence ao mesmo pacote de transporte. Isso permite diminuir a probabilidade de erro quando uma longa seqüência de bytes é afetada por erros de transmissão devido ao fato de que os bytes errados não necessariamente pertencem ao mesmo pacote. Dessa forma, a probabilidade de erro é diminuída desde que cada pacote tenha os seus próprios bytes de correção.

No receptor o **de-intercalador** faz a operação inversa do intercalador do transmissor. O formato do de-intercalador é o inverso do intercalador, onde a linha 52 do multiplexador da Figura 5.2.5 é a linha 1 do de-intercalador e assim sucessivamente.

5.2.4 Codificação de Treliça

A codificação em treliça é um método de codificação para a correção de possíveis erros aleatórios sem que para isso seja necessário um aumento na largura de faixa do canal. O preço pago por esse benefício é a expansão da constelação do sinal transmitido.

Em presença de ruído, essa expansão pode causar uma degradação do desempenho do sistema, porém isso é compensado em grande parte pelo ganho obtido no processo de codificação.

O codificador em treliça utilizado no sistema VSB é de taxa $2/3$ e de 4 estados, o que implica que esse codificador possui dois bits de entrada, sendo que um deles é pré-codificado (filtro dimensionado contra ruído de interferências de canais NTSC adjacentes) e o outro é codificado em dois bits através de um codificador **convolucional** de taxa $1/2$. Os três bits resultantes da codificação são mapeados em **8 símbolos** multiníveis formando a constelação 8-VSB. Dessa forma, é transmitida uma constelação unidimensional de 8 possíveis níveis. Na Figura 5.2.6 é mostrado o codificador de treliça e o respectivo mapeamento dos bits de saída nos 8 níveis do sistema[2].

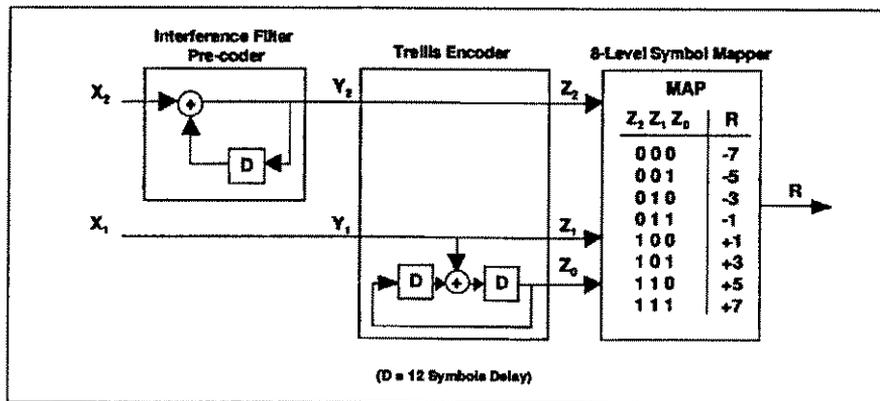


Figura 5.2.6 Codificador de Treliça do Sistema VSB

Toda codificação em treliça caracteriza-se pelo fato de que as palavras de códigos geradas num tempo “t” dependem dos bits transmitidos nos tempos anteriores t-1, t-2... , t-n, segundo a memória do codificador que é utilizado. Isso implica que qualquer erro na transmissão dos dados atuais gerará um erro na recepção dos próximos bits consecutivos.

Para proteger e ajudar o desempenho da codificação faz-se novamente o intercalamento da informação a se transmitir, a fim de que qualquer seqüência longa de erros que possa acontecer, é descorrelacionada com respeito a seqüência das palavras de código geradas.

O sistema de intercalamento utilizado no VSB é mostrado na Figura 5.2.7, onde são utilizados 12 codificadores do tipo mostrado na Figura 5.2.6. Cada codificador recebe um byte o qual é mapeado em 4 símbolos, agrupando-se os bits de dois em dois começando pelo bit mais significativo MSB (7,6,5,4,3,2,1,0). Dessa forma, primeiro são codificados os bits (7,6) onde o 7 é pré-codificado e o 6 é passado pelo codificador convolucional. Seguindo a mesma seqüência, os bits (7,5,3,1) serão sempre pré-codificados, e os bits (6,4,2,0) serão sempre passados pelo codificador convolucional [2].

Os outros bytes consecutivos ingressam nos codificadores seguintes, onde cada um deles gerará 4 novos símbolos. O seletor que multiplexa a saída dos codificadores toma somente um símbolo de cada um deles em cada ciclo para formar os grupos de símbolos correspondentes a cada codificador. Por exemplo, ao primeiro codificador correspondem o grupo de símbolos (0,12,24,36,48,...) de acordo com a ordem em que são transmitidos. Ao codificador dois correspondem os símbolos (1,13,25,37,...) e assim por diante, até formar doze grupos de símbolos.

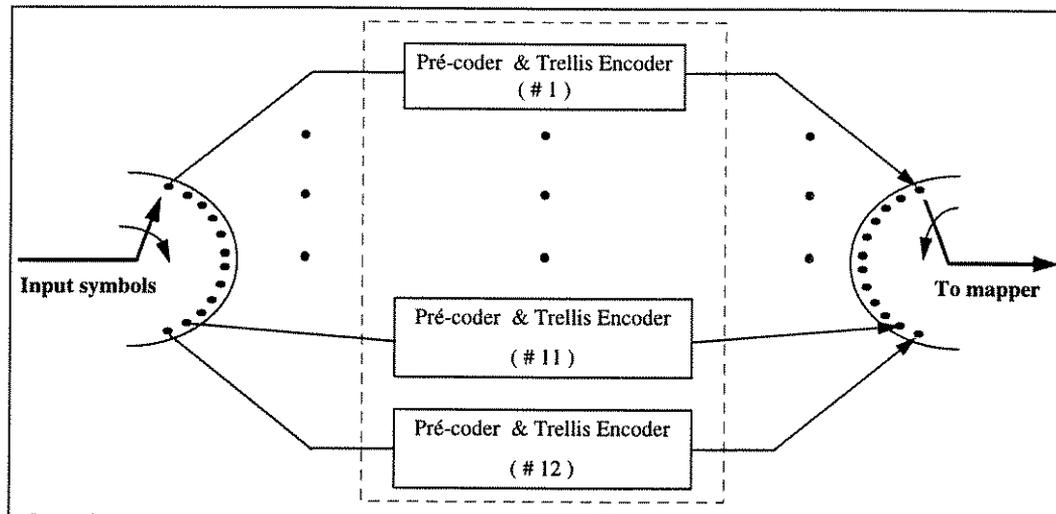


Figura 5.2.7 Multiplexador dos Codificadores de Treliça para um sistema VSB

Após o codificador transmitir o número de símbolos correspondente ao primeiro *Data Segment* de um frame, o multiplexador da Figura 5.2.7 avança 4 símbolos (4 codificadores) e a ordem de transmissão do segundo segmento começa no codificador 5 até 12 continuando do codificador 1 até o 4. Para o terceiro segmento, a ordem de transmissão de símbolos começa do codificador 9 até o 12, continuando do codificador 1 até o 8. Esse padrão de três segmentos é repetido nos 312 *Data Segments* do *Data Field*.

No sistema **16-VSB** não é utilizado o codificador de treliça, devido ao fato de que a transmissão é feita por **cabo** e também porque a probabilidade de erro do canal é muito menor do que ao do sistema de transmissão terrestre.

O sistema 16-VSB trabalha com uma constelação de 16 níveis, os quais são mapeados a partir dos bits de entrada provenientes do intercalador das palavras código *Reed Solomon*. Cada byte do código *Reed Solomon* intercalado é dividido em dois grupos de 4 bits, que então são mapeados em 16 níveis tal como mostra a Figura 5.2.8 [2].

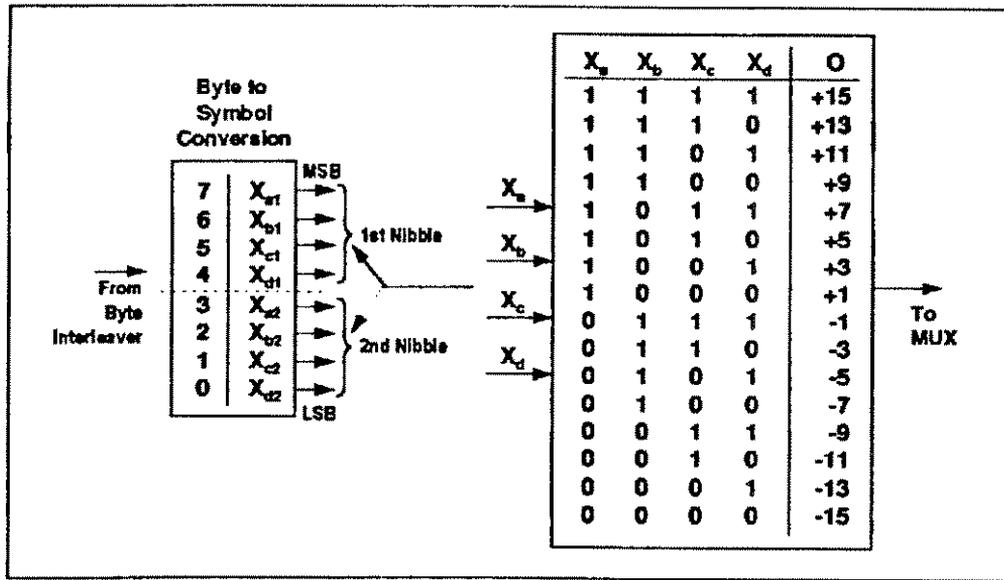


Figura 5.2.8 Mapeamento dos bits num sistema 16-VSB

Na Figura 5.2.9 mostra-se como ficaria o sistema 16-VSB ao ser excluído o codificador de treliça.

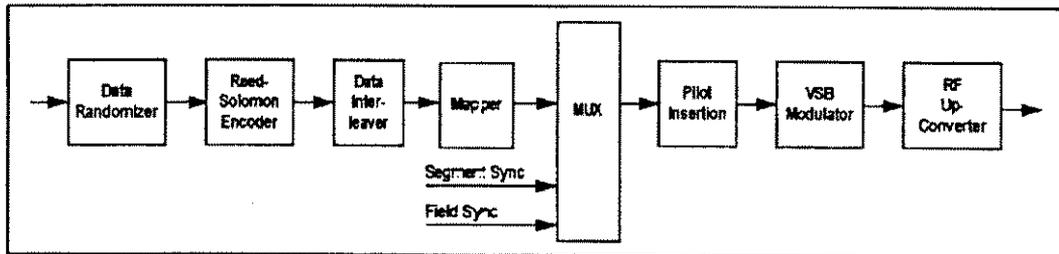


Figura 5.2.9 Sistema 16-VSB

5.2.5.- Inserção da Informação de Sincronismo

Após os dados serem processados adicionando-se informação de correção de erros com o mapeamento em 8-VSB ou 16-VSB, é feita a multiplexagem dos mesmos com a informação de sincronização a nível de segmentos (*Data Segment Sync*) e a nível de campo (*Data Field Sync*). A informação de sincronização é de grande importância, devido às seguintes considerações:

- O sincronismo de segmento de dados é usado na sincronização de fase e de frequência do *clock* de dados.
- É de grande importância a determinação do início e final de cada campo de dados.
- A informação de sincronismo pode ser utilizada no receptor como um sinal de treinamento para equalização dos dados. Além disso, pode ser utilizada como meio de obtenção de informação do sistema como por exemplo, modo de transmissão, auxílio em caso de alta interferência NTSC e outras demandas que surgirem.
- Finalmente, a informação de sincronismo pode ser utilizada a fim de decidir se no receptor é utilizado um filtro de rejeição contra interferências NTSC, cuja idéia básica é criar espaços nulos nas imediações das portadoras de imagem, sub-portadora de cor e portadora de áudio. Uma vez que as transmissões NTSC terminem, então o filtro poderá ser retirado.

5.2.5.1 Sincronismo de Segmento (*Dada Segment Sync*)

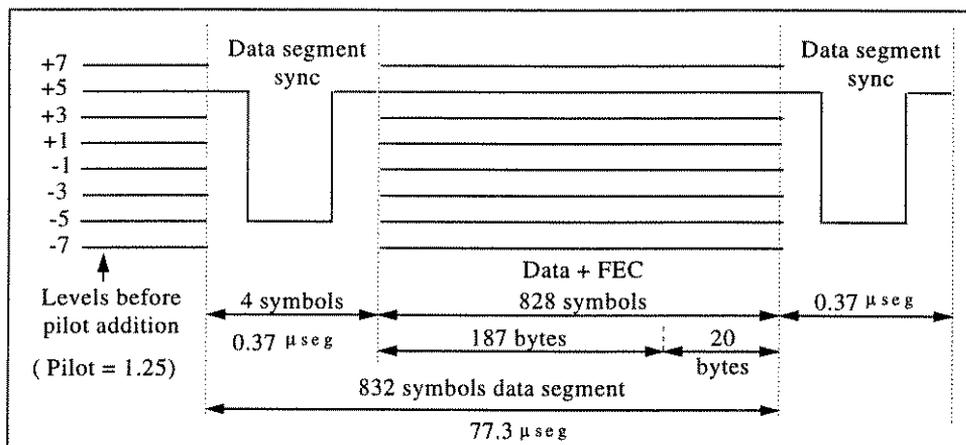
A informação de sincronismo de segmento é inserida através de 4 símbolos que representam o byte de sincronização dos pacotes de transporte MPEG-2 que não foram considerados no processo de codificação e intercalamento.

Os símbolos de sincronização são mapeados em forma binária, significando que só podem ter dois valores dos 8 ou 16 valores possíveis correspondentes ao sistema que está sendo utilizado. No caso do sistema 8-VSB um “1” lógico representa o nível +5, enquanto um “0” lógico representa o nível -5.

Para o sistema 16-VSB, um “1” lógico representa o nível +9, enquanto um “0” lógico representa o nível -9. Dessa forma, o valor binário definido para a informação de sincronismo em ambos os sistemas é **1001**.

Como mencionamos anteriormente, um segmento é composto por 828 símbolos de dados que correspondem aos 207 bytes do pacote de transporte MPEG-2 sem levar em consideração o byte de sincronismo. Dessa forma, com a inserção dos 4 símbolos de sincronismo o número total de símbolos por segmento transmitido é de 832 (208 bytes), os quais representam 77.3 μ s de transmissão. A informação de sincronismo do segmento representa 0.37 μ s[2].

Na Figura 5.2.10 mostram-se os formatos de transmissão do segmentos para os sistemas 8-VSB.



5.2.10 Formato do Segmento de Dados (Data Segment) para o sistema 8VSB

5.2.5.2 Sincronismo de Campo (Data Field Sync)

Um campo é composto por 313 segmentos, onde o primeiro deles leva a informação correspondente ao sincronismo de campo. O segmento de sincronismo de campo é inserido pelo multiplexador da Figura 5.2.1 e sua formatação é mostrada na Figura 5.2.11. Ele é constituído de 4 símbolos que representam o sincronismo de segmento de dados e de seqüências pseudo-aleatórias com extensões de 63 e 511 símbolos com a mesma amplitude que a dos símbolos de sincronismo do segmento. Além disso, existem 24 símbolos que indicam o modo de operação do sistema (VSB Mode), indicando se é 8 ou 16-VSB.

Finalmente, os últimos 12 símbolos do segmento anterior são duplicados e armazenados no final do segmento de sincronismo de campo, a fim de auxiliar o decodificador de treliça quando existe uma forte interferência do canal NTSC adjacente[2].

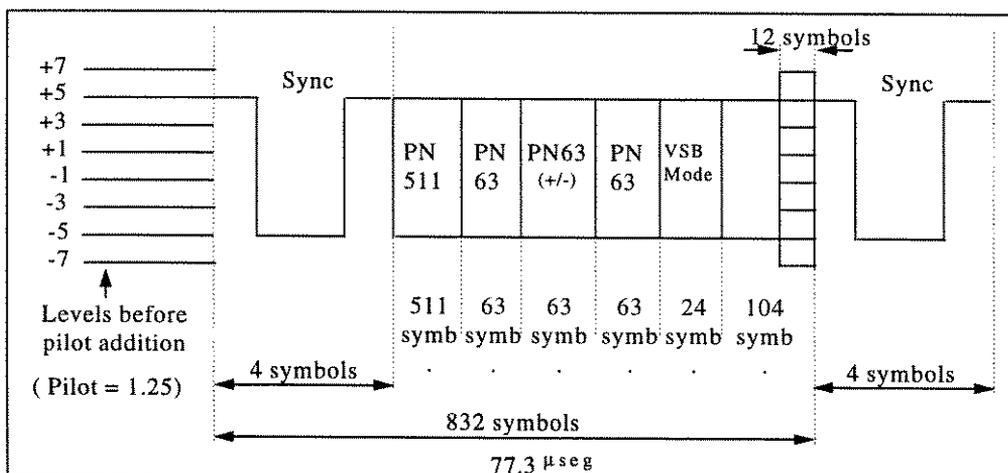


Figura 5.2.11 Formato do Segmento de Sincronismo de Campo para 8VSB

5.2.6.- Modulação e Inserção de Piloto

Após a inserção da informação de sincronismo, os dados são organizados no formato que mostra a Figura 5.2.2 para serem finalmente modulados e transmitidos. No entanto, para efeito de se ajudar a recuperação da portadora suprimida no receptor é adicionado um nível DC aos valores de todos os símbolos. Assim, quando a modulação (AM) é realizada, transmite-se o sinal mais um tom piloto com a mesma fase e frequência da portadora usada para modulação. O nível DC para o sistema 8-VSB é de 1.25 e para o caso do sistema 16-VSB é de 2.5. A potência do piloto deverá ser de 11.3dB abaixo da potência média do sinal[2].

Uma vez inserido o tom piloto, são geradas as componentes em fase e em quadratura necessárias para a geração da banda lateral vestigial. Cada componente é passada através de um conversor D/A que gera o sinal analógico que vai ser modulado em amplitude pela portadora correspondente. Finalmente, o sinal que vai ser transmitido é gerado mediante a soma das componentes moduladas que dão origem a banda VSB. A forma do espectro nas bordas da banda passante tem como resposta uma função de cosseno levantado. Essa característica permite recuperar o sinal no receptor livre de distorção.

A taxa de símbolos na entrada do modulador é de **10.76 Msymbol/s**, os quais constituem o sinal composto formado pelos dados codificados, pela informação de sincronismo e pelo tom piloto[2]. A porção da banda inferior que é transmitida é de **310 KHz**, tal como é mostrado na Figura 5.2.12.

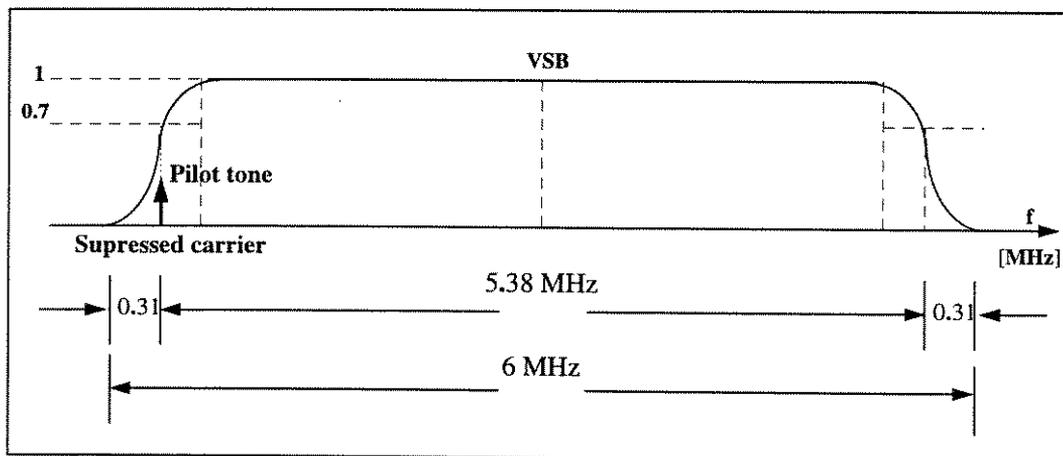


Figura 5.2.12. Espectro do Sinal VSB

5.3. COFDM (Protótipo-6MHz) versus 8-VSB

Finalmente, nessa última parte faremos uma comparação do desempenho dos dois sistemas de transmissão estudados neste capítulo: o **COFDM-6MHz** protótipo que está sendo desenvolvido no CRC do Canadá e o 8-VSB proposto

pela Grande Aliança. Na tabela 5.3.1 mostra o desempenho de cada um desses sistemas [8].

Parâmetros	COFDM-6MHz	8-VSB
Largura de Faixa (MHz)	6	6
Taxa de Bits (Mbps)	19,12	19,38
Constelação do sinais	64-QAM	8 níveis
Número de portadoras	6120	1
Codificação de canal	RS(255,239), TCM 2/3	RS(207,187), TCM 2/3
Co-canal NTSC/ATV	-2,7 dB	+1,8 dB
Canal adjacente inferior (ATV/NTSC)	-46,8 dB	-44,4 dB
Canal adjacente superior (ATV/NTSC)	-4,9 dB	-2,0 dB
<i>In band tone interference</i>	< -4,9 dB	> +9,1 dB
Eco estático simples	0 dB	5,8 dB

Tabela 5.3.1-Desempenho dos sistemas COFDM e 8-VSB num canal de 6 MHz

De acordo com a Tabela 5.3.1 observa-se um melhor desempenho do sistema COFDM frente as interferências de canais adjacentes, alocação de canais em frequência , etc. No entanto, o sistema 8-VSB proposto pela Grande Aliança tem altas possibilidades de ser tomado como sistema de transmissão para HDTV nos Estados Unidos. Por outro lado, os europeus tem já decidido o sistema COFDM como o sistema de transmissão para sinais HDTV na faixa de 8 MHz. O Japão vem investindo através do DTV Lab na implementação de sistemas utilizando COFDM em faixas de 6 MHz.

Finalmente, com respeito aos receptores, a Grande Aliança propôs também as características que devem ter os mesmos, enfatizando principalmente aquelas em filtros de recepção para a eliminação de canais adjacentes NTSC.

5.4. Conclusões e Comentários Finais

Neste capítulo tratamos de dar os aspectos mais relevantes dos principais sistemas de transmissão que estão sendo adotados para a transmissão de um sinal HDTV. Muito além das vantagens e desvantagens que possam oferecer ambos os sistemas, a escolha de um deles certamente deve-se também a considerações políticas de um determinado país ou continente.

O sistema de transmissão 8-VSB proposto pela Grande Aliança já foi inclusive mostrado em exposições, através da participação dos vários fabricantes, como *Westinhouse*, *SONY*, *Harris Sigma CD ATV*, *Thomsom* e outras companhias..

Para a demonstração, partiu-se de um sinal HDTV comprimido em áudio e vídeo através do AC-3 e MPEG-2 respectivamente. O sinal era gerado a partir de um videocassete *Panasonic D3* digital, aplicando-se um modulador 8-VSB. Obteve-se assim, um sinal modulado que foi aplicado ao *Westinghouse SiC High Power Amplifier* com um excitador ITS ligado a uma antena de transmissão com potência média ATV de 800 W. Usou-se o canal UHF 52.

Externamente ao local da exposição uma unidade móvel composta por uma antena de recepção UHF e um sistema de recepção Grande Aliança formado por um demodulador 8-VSB, um decodificador *Thomsom* acoplado ao vídeo decoder MPEG-2 HDTV *Philips* e um decoder *Dolby AC-3 surround sound*, geravam um sinal HDTV que era reproduzido e exibido para um platéia através de um telão projetado por um canhão HDTV da SONY, além de outros dois monitores HD1H-3000, também da SONY no mesmo auditório, tendo como som ambiente o Dolby Surround.

Foi uma demonstração que permitiu aos participantes, a visualização da qualidade dos sinais HDTV. Quanto ao preços dos receptores, representantes da *Westinghouse* disseram que estão procurando formas de torná-los mais competitivos em termos de preço inclusive mediante a possibilidade de possuírem display de baixa resolução, uma vez que o custo dos mesmos corresponde a mais do 70% do preço do receptor.

Dessa forma, percebeu-se um grande empenho em tornar o sistema proposto pela Grande Aliança como o padrão do sistema completo de HDTV nos Estados Unidos. A Grande Aliança foi dissolvida em 1995, deixando todas essas alternativas para a definição do sistema HDTV que foram descritos em todo este trabalho.

Com respeito ao sistema COFDM, tanto o Canadá como o Japão estão desenvolvendo trabalhos para canais de 6 MHz, enquanto a Europa está refazendo o protótipo COFDM 8 MHz para televisão digital, cuja opção de utilização como padrão *broadcast* já está tomada.

Capítulo 6

Conclusões Finais

Finalmente, pode-se concluir que a transmissão de um sinal de televisão HDTV sobre canais *broadcasting* convencionais é quase uma realidade.

Os últimos testes feitos nos Estados Unidos, demonstram que o sistema proposto pela Grande Aliança apresenta-se como um forte candidato para ser o sistema padrão de HDTV a se aplicar nesse país e em outros.

O desempenho em taxa de bits versus qualidade alcançado pelos sistemas de compressão MPEG-2 (vídeo) para uma resolução 1125/60 e AC-3 para 5.1 (0.1 canal LFE) canais de áudio, constituiu a base fundamental para o desenvolvimento de um sistema de televisão de alta definição que utilize os canais *broadcasting* convencionais. A melhoria do desempenho desses sistemas marcará portanto um avanço em diversos aspectos da televisão digital de alta definição.

Muitos dos problemas da televisão convencional foram solucionados no sistema digital HDTV.

A aplicação das técnicas de compressão estudadas neste trabalho a um sinal de televisão convencional, permitiria a transmissão de vários programas de televisão simultâneos usando-se a faixa de 6 MHz atualmente ocupada por esses sistemas, assim como uma sensível melhoria da qualidade de imagem reproduzida devido ao fato de que se conta com informação que pode fazer a correção de erros de transmissão.

A compatibilidade do sinal digital de vídeo com outros protocolos de redes de computadores é assegurada desde que um pacote de transporte MPEG-2 é facilmente representado através de 4 células ATM.

A Europa continua com as pesquisas sobre os sistemas de compressão a se implantar nesse continente. No entanto, o sistema MPEG-2 para áudio e vídeo apresenta-se também como um forte candidato. O formato de varredura a ser implantado nesse continente foi já definido em 1250/50.

Os sistemas de transmissão estão definidos, por um lado pela Grande Aliança que optou pelo sistema 8-16-VSB (6 MHz-*Simulcast*) entanto que a Europa pelo sistema COFDM para um canal de 8 MHz.

Entre esses dois sistemas percebe-se um melhor desempenho do sistema COFDM. No entanto, devido a independência com que trabalham tanto europeus como norte-americanos cada um deles desenvolve seu próprio sistema que será aplicado no seu país ou continente.

Nos próximos dois anos é muito provável a implantação do sistema de televisão HDTV nos Estados Unidos. Os sistemas convencionais não vão sofrer nenhuma alteração desde que se utilize o sistema "simulcast" considerado na proposta da Grande Aliança.

Na Europa talvez haja um tempo de espera maior do que nos EEUU antes de se implantar definitivamente o sistema HDTV. O Japão por outro lado, já tem definido o MUSE como padrão. No entanto uma vez que utiliza o mesmo sistema convencional NTSC dos Estados Unidos, está a espera do que será implantado finalmente naquele país ; provavelmente o Japão mudará seu sistema para acomodar o padrão americano totalmente digital com modulação VSB, embora esteja realizando pesquisas com o esquema OFDM.

Uma vez que o sistema HDTV tenha sido implantado no mundo inteiro, com o passar dos anos pode-se estimar que pouco a pouco os receptores analógicos de televisão convencional tenderão a desaparecer à medida que o custo dos receptores de aparelhos HDTV for diminuindo.

Neste trabalho, apresentou-se os aspectos mais relevantes dessa emergente tecnologia em ATV que está sendo definida pelos diferentes grupos de pesquisa. Na verdade, essa é uma área muito promissora para realização de futuros trabalhos. A HDTV despertou um enorme interesse no mundo inteiro por suas implicações políticas e econômicas. Afinal, a televisão continua sendo o grande meio de comunicação das massas. Ao mesmo tempo, o grande interesse pela HDTV é sem dúvida a área de cinema comercial, pela possibilidade de edição eletrônica. A viabilização da HDTV incluiu as mais recentes conquistas em processamento digital de sinais aliadas ao desenvolvimento tecnológico. De fato, falamos em novas técnicas de compressão, de codificação e de modulação enfim daquelas que sejam as mais apropriadas para fazer chegar aos receptores uma imagem-som (vídeo e áudio) de alta qualidade e de alta definição.

APÊNDICE - A

CODIFICAÇÃO de *HUFFMAN*

A codificação de *Huffman* é um processo que explora a estatística da informação a se transmitir, designando mais bits às mensagens que têm menor probabilidade de ocorrência e menos bits às mensagens que têm maior probabilidade de ocorrência.

O processo de codificação que é descrito a seguir é para o caso de 6 possíveis mensagens de transmissão ($L=6$), onde cada um deles forma um determinado nó na Figura A.1[5].

O primeiro passo de construção do código consiste em se escolher as mensagens com a mais baixa probabilidade de ocorrência. Nesse caso, serão as mensagens a_4 e a_6 . Depois soma-se as probabilidades deles para se gerar um nó a_7 , designando o valor de “1” a um ramo e um “0” ao outro. A ordem de designação afeta a codificação, mas não a taxa de bits a se transmitir. Considere-se agora as duas mensagens a_4 e a_6 como uma só mensagem a_7 com probabilidade $P_7 = \frac{1}{16}$.

O segundo passo consiste em se escolher as duas mensagens com menor probabilidade de ocorrência a partir das mensagens a_1, a_2, a_3, a_5, a_7 . Nesse caso, serão as mensagens a_3 e a_7 . Então somam-se as probabilidades deles para formar um novo nó a_8 com probabilidade $P_8 = \frac{5}{32}$ associando um “1” a um ramo e um “0” ao outro, da mesma forma que foi feito no primeiro passo. O processo continua até se obter um nó com probabilidade 1.

Finalmente, a codificação se faz seguindo-se os ramos como se estivesse voltando, isto é, partindo do nó com probabilidade 1 até chegar a mensagem desejada. Por exemplo, a mensagem a_5 tem a seguinte codificação : 101.

A comparação entre a taxa média de bits (no caso do exemplo anterior) obtida mediante a codificação com palavras código uniforme e codificação de *Huffman* é feita a seguir:

Codificação com palavras de código de igual número de bits : 3 bit/mensagem.

Codificação de Huffman :

$$\frac{5}{8} \cdot 1 + \frac{3}{32} \cdot 3 + \frac{3}{32} \cdot 3 + \frac{1}{32} \cdot 4 + \frac{1}{8} \cdot 3 + \frac{1}{32} \cdot 4 = \frac{29}{16} \frac{\text{bits}}{\text{mensagem}} = 1.813 \frac{\text{bits}}{\text{mensagem}}$$

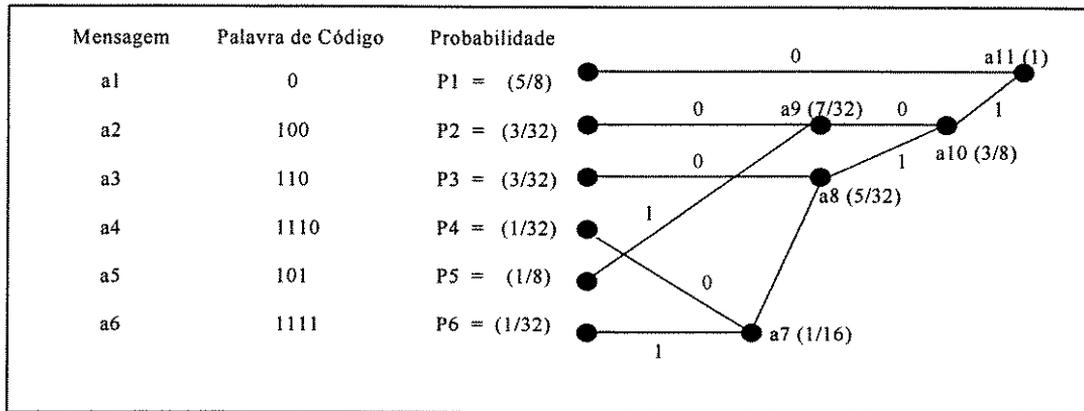


Figura. A.1 Processo de Codificação de Huffman

APÊNDICE - B

fscod	frequência de amostragem (KHz)
00	48.0
01	44.1
10	32.0
11	reservado

Tabela B.1 Mapeamento do fscod

Frmsizecod	frequência de amostragem	fs = 32 kHz palavras/frame	fs = 44.1 kHz palavras/frame	fs = 48 kHz palavras/frame
000000 (0)	32 kb/s	96	69	64
000001 (0)	32 kb/s	96	70	64
000010 (1)	40 kb/s	120	87	80
000011 (1)	40 kb/s	120	88	80
000100 (2)	48 kb/s	144	104	96
000101 (2)	48 kb/s	144	105	96
000110 (3)	56 kb/s	168	121	112
000111 (3)	56 kb/s	168	122	112
001000 (4)	64 kb/s	192	139	128
001001 (4)	64 kb/s	192	140	128
001010 (5)	80 kb/s	240	174	160
001011 (5)	80 kb/s	240	175	160
001100 (6)	96 kb/s	288	208	192
001101 (6)	96 kb/s	288	209	192
001110 (7)	112 kb/s	336	243	224
001111 (7)	112 kb/s	336	244	224
010000 (8)	128 kb/s	384	278	256
010001 (8)	128 kb/s	384	279	256
010010 (9)	160 kb/s	480	348	320
010011 (9)	160 kb/s	480	349	320
010100 (10)	192 kb/s	576	417	384
010101 (10)	192 kb/s	576	418	384
010110 (11)	224 kb/s	672	487	448
010111 (11)	224 kb/s	672	488	448
011000 (12)	256 kb/s	768	557	512
011001 (12)	256 kb/s	768	558	512
011010 (13)	320 kb/s	960	696	640
011011 (13)	320 kb/s	960	697	640
011100 (14)	384 kb/s	1152	835	768
011101 (14)	384 kb/s	1152	836	768
011110 (15)	448 kb/s	1344	975	896
011111 (15)	448 kb/s	1344	976	896
100000 (16)	512 kb/s	1536	1114	1024
100001 (16)	512 kb/s	1536	1115	1024
100010 (17)	576 kb/s	1728	1253	1152
100011 (17)	576 kb/s	1728	1254	1152
100100 (18)	640 kb/s	1920	1393	1280
100101 (18)	640 kb/s	1920	1394	1280

Tabela B.2. Mapeamento do frmsizecod

bsmod	tipo de serviço
000	serviço principal de áudio
001	serviço princip. de áudio sem dialogo
010	serviço associado: VI
011	serviço associado: HI
100	serviço associado: dialogo
101	serviço associado: comentário
110	serviço associado: emergência
111	serviço associado: voz

Tabela B.3 Mapeamento do bsmod

acmod	modo de codificação do áudio	nfchans	ordem de codificação dos canais que estão sendo utilizados
000	1+1	2	Ch1, Ch2
001	1/0	1	C
010	2/0	2	L, R
011	3/0	3	L, C, R
100	2/1	3	L, R, S
101	3/1	4	L, C, R, S(c. adjacente)
110	2/2	4	L, R, SL, SR
111	3/2	5	L, C, R, SL, SR

Tabela B.4 Mapeamento do acmod

frmsizecod	Taxa de bits nominal	fs = 32 kHz 5/8 framesize	fs = 44.1 kHz 5/8 framesize	fs = 48 kHz 5/8 framesize
000000 (0)	32 kb/s	60	42	40
000001 (0)	32 kb/s	60	43	40
000010 (1)	40 kb/s	75	53	50
000011 (1)	40 kb/s	75	55	50
000100 (2)	48 kb/s	90	65	60
000101 (2)	48 kb/s	90	65	60
000110 (3)	56 kb/s	105	75	70
000111 (3)	56 kb/s	105	76	70
001000 (4)	64 kb/s	120	86	80
001001 (4)	64 kb/s	120	87	80
001010 (5)	80 kb/s	150	108	100
001011 (5)	80 kb/s	150	108	100
001100 (6)	96 kb/s	180	130	120
001101 (6)	96 kb/s	180	130	120
001110 (7)	112 kb/s	210	151	140
001111 (7)	112 kb/s	210	152	140
010000 (8)	128 kb/s	240	173	160
010001 (8)	128 kb/s	240	173	160
010010 (9)	160 kb/s	300	217	200
010011 (9)	160 kb/s	300	217	200
010100 (10)	192 kb/s	360	260	240
010101 (10)	192 kb/s	360	261	240
010110 (11)	224 kb/s	420	303	280
010111 (11)	224 kb/s	420	305	280
011000 (12)	256 kb/s	480	347	320
011001 (12)	256 kb/s	480	348	320
011010 (13)	320 kb/s	600	435	400
011011 (13)	320 kb/s	600	435	400
011100 (14)	384 kb/s	720	521	480
011101 (14)	384 kb/s	720	522	480
011110 (15)	448 kb/s	840	608	560
011111 (15)	448 kb/s	840	610	560
100000 (16)	512 kb/s	960	696	640
100001 (16)	512 kb/s	960	696	640
100010 (17)	576 kb/s	1080	782	720
100011 (17)	576 kb/s	1080	783	720
100100 (18)	640 kb/s	1200	870	800
100101 (18)	640 kb/s	1200	871	800

Tabela B.5. Tamanho em Palavras (2-bytes) das primeiras 5/8 partes de uma Frame

	B=0	B=1	B=2	B=3	B=4	B=5	B=6	B=7	B=8	B=9
A=0	0.00014	0.00024	0.00037	0.00051	0.00067	0.00086	0.00107	0.00130	0.00157	0.00187
A=1	0.00220	0.00256	0.00297	0.00341	0.00390	0.00443	0.00501	0.00564	0.00632	0.00706
A=2	0.00785	0.00871	0.00962	0.01061	0.01166	0.01279	0.01399	0.01526	0.01662	0.01806
A=3	0.01959	0.02121	0.02292	0.02472	0.02662	0.02863	0.03073	0.03294	0.03527	0.03770
A=4	0.04025	0.04292	0.04571	0.04862	0.05165	0.05481	0.05810	0.06153	0.06508	0.06878
A=5	0.07261	0.07658	0.08069	0.08495	0.08935	0.09389	0.09859	0.10343	0.10842	0.11356
A=6	0.11885	0.12429	0.12988	0.13563	0.14152	0.14757	0.15376	0.16011	0.16661	0.17325
A=7	0.18005	0.18699	0.19407	0.20130	0.20867	0.21618	0.22382	0.23161	0.23952	0.24757
A=8	0.25574	0.26404	0.27246	0.28100	0.28965	0.29841	0.30729	0.31626	0.32533	0.33450
A=9	0.34376	0.35311	0.36253	0.37204	0.38161	0.39126	0.40096	0.41072	0.42054	0.43040
A=10	0.44030	0.45023	0.46020	0.47019	0.48020	0.49022	0.50025	0.51028	0.52031	0.53033
A=11	0.54033	0.55031	0.56026	0.57019	0.58007	0.58991	0.59970	0.60944	0.61912	0.62873
A=12	0.63827	0.64774	0.65713	0.66643	0.67564	0.68476	0.69377	0.70269	0.71150	0.72019
A=13	0.72877	0.73723	0.74557	0.75378	0.76186	0.76981	0.77762	0.78530	0.79283	0.80022
A=14	0.80747	0.81457	0.82151	0.82831	0.83496	0.84145	0.84779	0.85398	0.86001	0.86588
A=15	0.87160	0.87716	0.88257	0.88782	0.89291	0.89785	0.90264	0.90728	0.91176	0.91610
A=16	0.92028	0.92432	0.92822	0.93197	0.93558	0.93906	0.94240	0.94560	0.94867	0.95162
A=17	0.95444	0.95713	0.95971	0.96217	0.96451	0.96674	0.96887	0.97089	0.97281	0.97463
A=18	0.97635	0.97799	0.97953	0.98099	0.98236	0.98366	0.98488	0.98602	0.98710	0.98811
A=19	0.98905	0.98994	0.99076	0.99153	0.99225	0.99291	0.99353	0.99411	0.99464	0.99513
A=20	0.99558	0.99600	0.99639	0.99674	0.99706	0.99736	0.99763	0.99788	0.99811	0.99831
A=21	0.99850	0.99867	0.99882	0.99895	0.99908	0.99919	0.99929	0.99938	0.99946	0.99953
A=22	0.99959	0.99965	0.99969	0.99974	0.99978	0.99981	0.99984	0.99986	0.99988	0.99990
A=23	0.99992	0.99993	0.99994	0.99995	0.99996	0.99997	0.99998	0.99998	0.99998	0.99999
A=24	0.99999	0.99999	0.99999	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
A=25	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000				

Tabela B.6. Valores da Janela. $w(n)$ onde $n = (A \times 10) + B$

Dif Exp	valor mapeado
+ 2	4
+ 1	3
0	2
- 1	1
- 2	0

Tabela B.7 Mapeamento dos expoentes diferenças mediante a estratégia D15

Dif. Exp n	Dif. Exp n+1	valor mapeado
+2	0	4
+1	0	3
0	0	2
-1	0	1
-2	0	0

Tabela B.8 Mapeamento dos expoentes diferenças mediante a estratégia D25

Dif Exp n	Dif Exp n+1	Dif Exp n+2	Dif Exp n+3	valor mapeado
+2	0	0	0	4
+1	0	0	0	3
0	0	0	0	2
-1	0	0	0	1
-2	0	0	0	0

Tabela B.9. Mapeamento dos Expoentes Diferenciais D45

Chexpstr[ch]	Estratégia	exponentes por grupo
00	reutilizar anteriores	0
01	D15	3
10	D25	6
11	D45	12

Tabela B.10. Codificação de Estratégias para Expoentes

lfeexpstr	Estratégia	Exponentes por grupo
0	reutilizar prévios exponentes	0
1	D15	3

Tabela B.11. Codificação das Estratégias para Expoentes no canal LFE

Banda #	bndtab [banda]	bndsz [banda]
0	0	1
1	1	1
2	2	1
3	3	1
4	4	1
5	5	1
6	6	1
7	7	1
8	8	1
9	9	1
10	10	1
11	11	1
12	12	1
13	13	1
14	14	1
15	15	1
16	16	1
17	17	1
18	18	1
19	19	1
20	20	1
21	21	1
22	22	1
23	23	1
24	24	1

banda	bndtab [band]	bndsz [band]
25	25	1
26	26	1
27	27	1
28	28	3
29	31	3
30	34	3
31	37	3
32	40	3
33	43	3
34	46	3
35	49	6
36	55	6
37	61	6
38	67	6
39	73	6
40	79	6
41	85	12
42	97	12
43	109	12
44	121	12
45	133	24
46	157	24
47	181	24
48	205	24
49	229	24

Tabela B.12 Mapeamento da escala de freqüências para o processo de quantização de mantissas

banda#	hth[0][band] (fs=48 kHz)	hth[1][band] (fs=44.1 kHz)	hth[2][band] (fs=32 kHz)
0	0x04d0	0x04f0	0x0580
1	0x04d0	0x04f0	0x0580
2	0x0440	0x0460	0x04b0
3	0x0400	0x0410	0x0450
4	0x03e0	0x03e0	0x0420
5	0x03c0	0x03d0	0x03f0
6	0x03b0	0x03c0	0x03e0
7	0x03b0	0x03b0	0x03d0
8	0x03a0	0x03b0	0x03c0
9	0x03a0	0x03a0	0x03b0
10	0x03a0	0x03a0	0x03b0
11	0x03a0	0x03a0	0x03b0
12	0x03a0	0x03a0	0x03a0
13	0x0390	0x03a0	0x03a0
14	0x0390	0x0390	0x03a0
15	0x0390	0x0390	0x03a0
16	0x0380	0x0390	0x03a0
17	0x0380	0x0380	0x03a0
18	0x0370	0x0380	0x03a0
19	0x0370	0x0380	0x03a0
20	0x0360	0x0370	0x0390
21	0x0360	0x0370	0x0390
22	0x0350	0x0360	0x0390
23	0x0350	0x0360	0x0390
24	0x0340	0x0350	0x0380

banda#	hth[0][band] (fs=48 kHz)	hth[1][band] (fs=44.1 kHz)	hth[2][band] (fs=32 kHz)
25	0x0340	0x0350	0x0380
26	0x0330	0x0340	0x0380
27	0x0320	0x0340	0x0370
28	0x0310	0x0320	0x0360
29	0x0300	0x0310	0x0350
30	0x02f0	0x0300	0x0340
31	0x02f0	0x02f0	0x0330
32	0x02f0	0x02f0	0x0320
33	0x02f0	0x02f0	0x0310
34	0x0300	0x02f0	0x0300
35	0x0310	0x0300	0x02f0
36	0x0340	0x0320	0x02f0
37	0x0390	0x0350	0x02f0
38	0x03e0	0x0390	0x0300
39	0x0420	0x03e0	0x0310
40	0x0460	0x0420	0x0330
41	0x0490	0x0450	0x0350
42	0x04a0	0x04a0	0x03c0
43	0x0460	0x0490	0x0410
44	0x0440	0x0460	0x0470
45	0x0440	0x0440	0x04a0
46	0x0520	0x0480	0x0460
47	0x0800	0x0630	0x0440
48	0x0840	0x0840	0x0450
49	0x0840	0x0840	0x04e0

Tabela B.13. Mapeamento do limiar de ruído permitido pelo Ouvido Humano para cada banda de frequência da tabela B.12

Endereço	baptab [endereço]	endereço	baptab [endereço]
0	0	32	10
1	1	33	10
2	1	34	10
3	1	35	11
4	1	36	11
5	1	37	11
6	2	38	11
7	2	39	12
8	3	40	12
9	3	41	12
10	3	42	12
11	4	43	13
12	4	44	13
13	5	45	13
14	5	46	13
15	6	47	14
16	6	48	14
17	6	49	14
18	6	50	14
19	7	51	14
20	7	52	14
21	7	53	14
22	7	54	14
23	8	55	15
24	8	56	15
25	8	57	15
26	8	58	15
27	9	59	15
28	9	60	15
29	9	61	15
30	9	62	15
31	10	63	15

*Tabela B.14. Mapeamento dos ponteiros *bap* e os possíveis resultados da rotina de alocação de bits (endereço)*

bap	níveis de quantização qntzbap[bap]	bits por mantissa (grupo bits / num em grupo)
0	0	0
1	3	1.67 (5/3)
2	5	2.33 (7/3)
3	7	3
4	11	3.5 (11/2)
5	15	4
6	32	5
7	64	6
8	128	7
9	256	8
10	512	9
11	1024	10
12	2048	11
13	4096	12
14	16,384	14
15	65,536	16

Tabela B.15. Mapeamento dos ponteiros bap e os níveis de quantização

bap	níveis de quantização	tipo de quantização	mantissa bits (qntztab[bap]) (grupo bits / num. em grupo)
0	0	nehum	0
1	3	simétrica	1.67 (5/3)
2	5	simétrica	2.33 (7/3)
3	7	simétrica	3
4	11	simétrica	3.5 (11/2)
5	15	simétrica	4
6	32	assimétrica	5
7	64	assimétrica	6
8	128	assimétrica	7
9	256	assimétrica	8
10	512	assimétrica	9
11	1024	assimétrica	10
12	2048	assimétrica	11
13	4096	assimétrica	12
14	16,384	assimétrica	14
15	65,536	assimétrica	16

Tabela B.16. Mapeamento dos ponteiros bap e os tipos de quantização

Código transmitido	valor da mantissa
0	-2./3
1	0
2	2./3

Tabela B.17. Mapeamento dos valores das mantissas (quantização simétrica) para $bap=1$ (3 níveis)

código da mantissa	valor da mantissa
0	-4./5
1	-2./5
2	0
3	2./5
4	4./5

Tabela B.18. Mapeamento dos valores das mantissas (quantização simétrica) para $bap=2$ (5 níveis)

código da mantissa	valor da mantissa
0	-6./7
1	-4./7
2	-2./7
3	0
4	2./7
5	4./7
6	6./7

Tabela B.19. Mapeamento dos valores das mantissas (quantização simétrica) para $bap=3$ (7 níveis)

código da mantissa	valor da mantissa
0	-10./11
1	-8./11
2	-6./11
3	-4./11
4	-2./11
5	0
6	2./11
7	4./11
8	6./11
9	8./11
10	10./11

Tabela B.20 Mapeamento dos valores das mantissas (quantização simétrica) para $bap=4$ (11 níveis)

código da mantissa	valor da mantissa
0	-14./15
1	-12./15
2	-10./15
3	-8./15
4	-6./15
5	-4./15
6	-2./15
7	0
8	2./15
9	4./15
10	6./15
11	8./15
12	10./15
13	12./15
14	14./15

Tabela B.21. Mapeamento dos valores das antissas (quantização simétrica) para $bap=5$ (15 níveis)

número da sub-faixa de acoplam.	coefic. Inicial	coefic. Final	Frequência inicial fs=48 kHz	Frequência final fs=48 kHz	Frequência inicial fs=44.1 kHz	Frequência final fs=44.1 kHz
0	37	48	3.42	4.55	3.14	4.18
1	49	60	4.55	5.67	4.18	5.21
2	61	72	5.67	6.80	5.21	6.24
3	73	84	6.80	7.92	6.24	7.28
4	85	96	7.92	9.05	7.28	8.31
5	97	108	9.05	10.17	8.31	9.35
	109	120	10.17	11.30	9.35	10.38
7	121	132	11.30	12.42	10.38	11.41
8	133	144	12.42	13.55	11.41	12.45
9	145	156	13.55	14.67	12.45	13.48
10	157	168	14.67	15.80	13.48	14.51
11	169	180	15.80	16.92	14.51	15.55
12	181	192	16.92	18.05	15.55	16.58
13	193	204	18.05	19.17	16.58	17.61
14	205	216	19.17	20.30	17.61	18.65
15	217	228	20.30	21.42	18.65	19.68
16	229	240	21.42	22.55	19.68	20.71
17	241	252	22.55	23.67	20.71	21.75

Tabela B.22. Sub-Faixas de Acoplamento

Mstrecplco[ch]	ganho multiplicativo
00	1
01	2^{-3}
10	2^{-6}
11	2^{-9}

Tabela B.23. Mapeamento do sub-campo mstrecplco

X ₀	X ₁	X ₂	Valor Inteiro (X)	Ganho Indicado
0	1	1	3	+24.08 dB
0	1	0	2	+18.06 dB
0	0	1	1	+12.04 dB
0	0	0	0	+6.02 dB
1	1	1	-1	0 dB
1	1	0	-2	-6.02 dB
1	0	1	-3	-12.04 dB
1	0	0	-4	-18.06 dB

Tabela B.24. Mapeamento dos 3 Bits MSB do Sub-Campo dynrgn

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] United States Advanced Television Systems Committee, "**Guide to the Use of the Digital Television Standard For HDTV Transmission**", Doc. A/54 12 April 1995.
- [2] United States Advanced Television Systems Committee, "**Digital Television Standard For HDTV Transmission**". Doc A/53. 12 April 1995.
- [3] CCIR, XVIIth Plenary Assembly at Düsseldorf, 1990 on Document 11/1008-E from 15 January 1990, in Report 801-3 (MOD F). "**The Present State of High - Definition Television**".
- [4] CCIR, XVIIth Plenary at Düsseldorf, 1990). Report 801-4, "**The Present State of High - Definition Television**". (Question 27/11, study Programmes 27A/11 and 27B/11).
- [5] Jae S. Lim, "**Two-Dimensional Signal and Image Processing**", Prentice Hall, 1989.
- [6] Arun N. Netravali, Barry G. Haskell, "**Digital Pictures - Representation and Compression**", Plenum Press, 1988.
- [7] Didier Le Gall, "MPEG: A Video Compression Standard for Multimedia Applications", **Communications of the ACM**, Vol. 34, no. 4, April 1991.
- [8] Yiyang Wu, "**TV Digital: Um desafio para os Broadcasters**", Palestra Técnica realizada pela SET, S. Paulo, 06/03/96.
- [9] K.R. Rao and P. Yip, "**Discrete Cosine Transform - Algorithms, Advantages and Applications**", Academic Press, 1990.
- [10] United States Advanced Television Systems Committee, "**Digital Audio Compression (AC-3) Standard**", Doc A/52, 10 Nov. 94 - 12 Apr 95.

- [11] A. V. Oppenheim, “**Applications of Digital Signal Processing**”, Prentice Hall, 1987.
- [12] John P. Princen and Alan Bernard Bradley, “Analysis/Synthesis Filter Bank Design Based on Time Domain Aliasing Cancellation”, **IEEE Trans. Acoust., Speech and Signal Processing**, Vol. ASSP-34, No. 5, October 1986.
- [13] John M. Tribolet and Ronald E. Crochiere, “Frequency Domain Coding of Speech”, **IEEE Trans. Acoust., Speech and Signal Processing**, Vol. ASSP-27, No. 5, October 1979.
- [14] ISO/IEC 13818-1, “**MPEG-2 System**”, March 1994.
- [15] M. Sablatash.,”Transmission of All-Digital Advanced Television: State of the Art and Future Directions”, **IEEE Trans. on Broadcasting**, Vol. 40, No 2, jun/1994.
- [16] William. Y. Zou and Yiyan Wu, “COFDM: An Overview”, **IEEE Transactions on Broadcasting**, Vol. 41, No. 1, mar/1995.
- [17] Yiyan Wu and William Y. Zou, “Orthogonal Frequency Division Multiplexing: A Multi-Carrier Modulation Scheme”, **IEEE Transactions on Consumer Electronics**, Vol. 41, No. 3, ago/1995.
- [18] Breti Wayne, Gary Sgrignoli and Paul Snopko, “VSB Modem Subsystem Design for Grand Alliance Digital Television Receivers”, **IEEE Transactions on Consumer Electronics**, Vol. 41, No. 3, ago/1995.
- [19] Ademir de Jesus Lourenço, “**Sistemas HDTV: Um estudo exploratório para caracterização dos sinais e aspectos relevantes para a TV de Alta Definição**”, Campinas : FEE, UNICAMP, Julho 1994, Mestrado (Comunicações)-Faculdade de Engenharia Elétrica, Universidade Estadual de Campinas.
- [20] ISO/IEC 13818-2, “**MPEG-2 Vídeo**”, May 1996.