

Sumário

Resumo	xv
Abstract	xvi
1. Introdução	1
2. Fundamentos de Modulação Spread Spectrum	9
2.1. Introdução	9
2.2. Revisão histórica	9
2.3. Fundamentos de um sistema SS	9
2.3.1. Imunidade à Interferência	10
2.3.2. Imunidade a reflexões sucessivas	12
2.3.3. Capacidade de Múltiplo Acesso utilizando CDMA (<i>Code Division Multiplex Access</i>)	13
2.4. Seqüências Pseudo-Aleatórias	14
2.5. Modulação DS-SS	17
2.5.1. Aspectos teóricos	17
2.5.2. Modulação DS/BPSK (<i>Direct Sequence Binary Phase Shift Keying</i>)	18
2.6. Conclusões	21
3. Técnicas de Transmissão de Dados em Redes Elétricas	23
3.1. Introdução	23
3.2. Tecnologias Existentes	23
3.2.1. X-10	23
3.2.2. LONworks	24
3.2.3. CEBus	25
3.3. Comparação entre as tecnologias	26
3.4. Conclusões	27
4. Transmissão de Dados em Redes de Energia Elétrica Utilizando Técnicas de Spread Spectrum Carrier	28
4.1. Introdução	28
4.2. Concepção Básica	28
4.3. Estrutura Básica do Protocolo CEBus	28
4.3.1. Implementação da comunicação em pacotes de dados	29

4.3.2. CRC (<i>Cyclical Redundancy Check</i>)	33
4.3.3. CSMA/CD (<i>Carrier Sense Multiple Access / Collision Detection</i>)	34
4.4. Codificação dos bits no protocolo CEBus	35
4.4.1. <i>Unit Symbol Time</i> (UST)	35
4.4.2. Codificação de Símbolos no Protocolo CEBus	37
4.4.3. Função e Codificação do Preâmbulo no Protocolo CEBus	38
4.4.4. Codificação do Corpo do Pacote no Protocolo CEBus	39
4.5. Características de Transmissão do Protocolo CEBus	41
4.5.1. Overhead	41
4.5.2. Taxa de Transmissão Efetiva	42
4.6. Acoplamento Eletro-Magnético entre Fases	43
4.7. Conclusões	48
<u>5. Projeto do modem DLC–P300 (<i>Digital Line Carrier</i>)</u>	<u>49</u>
5.1. Introdução	49
5.2. Visão Geral do DLC P–300	49
5.2.1. Função do DLC–P300	49
5.2.2. Componentes do DLC – P300	50
5.2.3. Estrutura do DLC–P300	51
5.3. Controlador de Rede Intellon® SSC P300 PL.	52
5.4. Amplificador com controle de ganho Intellon® SSC P111.	59
5.5. <i>Driver</i> de Tensão, Filtro Harmônico, Filtro de Entrada e Acoplador.	61
5.5.1. Introdução	61
5.5.2. Filtro Harmônico / <i>Driver</i> de Tensão	61
5.5.3. Filtro de Entrada	62
5.5.4. Acoplador da Rede Elétrica	64
5.6. Resultados	66
5.6.1. Simulação no PSPICE®	66
5.6.2. Medição de sinais no protótipo do DLC–P300	71
5.7. Conclusões	72
<u>6. Proposição de Algoritmos de Transmissão e Recepção de <i>Spread Spectrum</i> Carrier Baseados em Técnicas DSP.</u>	<u>73</u>
6.1. Introdução	73
6.2. Vantagens da substituição	73
6.3. Princípios básicos dos algoritmos de recepção de sinais digitais	74

6.3.1. Transmissão de símbolos	74
6.3.2. <i>Matched Filter</i> e Filtro de Correlação	75
6.3.3. Transmissão de Sinais Bipolares	78
6.4. Modelagem do canal de comunicação em sistemas SSC	79
6.4.1. Atenuação em redes elétricas	79
6.4.2. Canal AWGN	80
6.4.3. Atenuação em frequência	81
6.4.4. Modelagem da atenuação em redes elétricas	88
6.5. Análises iniciais de um algoritmo de recepção SSC	96
6.5.1. Considerações iniciais	96
6.5.2. Análise do <i>Buffer</i> de Entrada	97
6.5.3. Análise da função de autocorrelação do sinal SSC	99
6.6. Proposição de um Algoritmo para Recepção de Sinais SSC no Protocolo CEBus.	102
6.6.1. Considerações iniciais	102
6.6.2. Estágio 1: Detecção de Preâmbulo	104
6.6.3. Estágio 2: Sincronismo	109
6.6.4. Estágio 3: Detector de Símbolo e Conversor Símbolo-Bit	114
6.7. Simulação do Algoritmo Proposto	122
6.8. Otimização do “<i>Matched Filter</i>”	128
6.8.1. Simplificação por blocos	128
6.8.2. Operações MACD	129
6.8.3. Aproximação por operações XOR	130
6.9. Conclusões	133
7. Conclusões	134
Apêndice A – Análise e Projeto de Um Acoplador de Fases para Sistemas SSC em Redes Trifásicas de Baixa Tensão.	137
Apêndice B – Coeficientes dos filtros digitais dos canais de comunicação	152
Referências Bibliográficas	159

Índice de Figuras

Figura 1.1 – Principais obstáculos enfrentados na transmissão de dados: ruído, atenuação e distorção. _____	2
Figura 1.2 – Sistema distribuído. _____	3
Figura 1.3 – Características dos proprietários de bens eletrônicos nos Estados Unidos (*2010: projeção). Fonte: Park's Associates, Outubro de 2000. _____	5
Figura 1.4 – Evolução dos serviços prestados nos Estados Unidos (*2010: projeção). Fonte: Park's Associates, Outubro de 2000. _____	5
Figura 1.5 – Evolução dos serviços de informação no mundo (Fonte: 3R Corp). _____	6
Figura 2.1 - Conceito de Comunicação Spread Spectrum _____	10
Figura 2.2 - Recuperação do sinal com interferência: a) Sinal recebido, b) Resultado do operador $\varepsilon^{-1}(\cdot)$ (despreading). c) Resultado do operador $F(\cdot)$. _____	11
Figura 2.3 – Registrador de deslocamento para geração de seqüências pseudo-aleatórias, com $k=4$. _____	15
Figura 2.4 – Seqüência pseudo-aleatória obtida com o registrador de deslocamento mostrado na Figura 2.3, com as seguintes condições iniciais: $X_1X_2X_3X_4=1000$. _____	16
Figura 2.5 – Gráfico da função de auto-correlação ($R_c(n)$) da seqüência pseudo-aleatória mostrada na Figura 2.4. _____	16
Figura 2.6 - Algoritmo de recepção DS-SS _____	17
Figura 2.7 - Diagrama básico da modulação DS/BPSK: a) Concepção do transmissor, com modulação BPSK, b) Receptor, com filtro de entrada e demodulador BPSK. _____	20
Figura 2.8 – a) Modulação BPSK de um bitstream ($b=11010010$) b) Código pseudo-aleatório com $k=8$, $N=127$. _____	20
Figura 2.9 – a) Sinal transmitido ($SNR=17dB$) b) Sinal recuperado e filtrado _____	21
Figura 2.10 - Comparação entre o sinal enviado e o sinal recebido _____	21
Figura 3.1 – Temporização no protocolo X-10. _____	23
Figura 3.2 – Propagação dos símbolos nas três fases (V_A , V_B e V_C) no protocolo X-10. _____	24
Figura 3.3 – Freqüências utilizadas pela tecnologia DCF. a) Freqüências utilizadas pela banda C; b) freqüências utilizadas pela banda A. _____	25
Figura 3.4 – Gráfico do espalhamento do sinal SSC. _____	26
Figura 4.1 – Implementação das estruturas de dados em um pacote CEBus. Esta estrutura de pacote só é válida para os meios PowerLine e RF [1]. _____	29
Figura 4.2 – Exemplo da estrutura do pacote como uma carta mandada por um serviço postal. _____	29
Figura 4.3 – Descrição bit a bit do Campo de Controle _____	30

Figura 4.4 – Estrutura de endereçamento no pacote CEBus, válidas para o endereço de origem e para o endereço de destino.	31
Figura 4.5 – Transmissão de uma string de dados genérica	32
Figura 4.6 – Transmissão dos parâmetros da linguagem CAL	32
Figura 4.7 – Algoritmo de detecção de erro CRC – 16. O resto da divisão, R_e , é calculado e incluído ao final da mensagem transmitida. No receptor, é novamente calculado o resto, R_r , avaliando possíveis erros de transmissão.	34
Figura 4.8 – UST do protocolo CEBus (Chirp)	36
Figura 4.9 – Gráfico da STFT do chirp da Figura 4.8.	37
Figura 4.10 – Fast Fourier Transform do chirp ($F_{amostragem} = 2 \text{ MHz}$)	37
Figura 4.11 – Transmissão dos símbolos do preâmbulo (“10010010”) e EOF. Notar o período de $14 \mu\text{s}$ de ausência de sinal após o chirp ser transmitido.	39
Figura 4.12 – Codificação do preâmbulo em função dos estados SUPERIOR (S_1) e INFERIOR (S_2). Notar o símbolo de EOF ao final do preâmbulo.	39
Figura 4.13 – a) Transmissão dos bits “1011” utilizando a codificação PRK. b) Mudança de estado na modulação PRK. Notar que um chirp é exatamente o inverso do outro.	40
Figura 4.14 – Codificação do início do corpo do pacote CEBus. Notar que o primeiro estado do corpo do pacote é codificado com estado diferente do estado do preâmbulo.	40
Figura 4.15 – Codificação do final do pacote CEBus. O CRC é calculado e incluído ao final do símbolo EOP.	41
Figura 4.16 – Diagrama esquemático típico de um transformador de distribuição, em configuração Δ -Y. O circuito magnético que enlaça os enrolamentos (primários e secundários) tem elevada relutância em altas frequências.	44
Figura 4.17 – Dispositivos situados na mesma fase do secundário de um transformador de distribuição.	44
Figura 4.18 – Dispositivos situados em fases diferentes. O único caminho disponível para o estabelecimento do link de comunicação é através dos enrolamentos secundários do transformador de distribuição.	45
Figura 4.19 – Dispositivos situados em fases diferentes, mas estabelecendo um link de comunicação, através do acoplador trifásico conectado às fases do transformador.	45
Figura 4.20 – a) Dispositivo de acoplamento de alta frequência entre as fases. b) Detalhe construtivo do transformador de acoplamento.	46
Figura 4.21 – Arranjo de transformadores para isolar eletricamente a rede.	47
Figura 4.22 – Ligação dos dispositivos CEBus na rede elétrica isolada.	47
Figura 5.1 – Algumas configurações para a ligação do modem DLC–P300. Todos os dispositivos conectados à rede elétrica podem partilhar os dados.	49
Figura 5.2 – Vista externa e dimensões do DLC–P300.	50

Figura 5.3 – Montagem interna do DLC–P300 (vista superior).	51
Figura 5.4 – Diagrama esquemático principal do modem DLC–P300.	52
Figura 5.5 – Pinagem do circuito integrado (IC) SSC P300.	53
Figura 5.6 – Estrutura interna do circuito integrado SSC P300.	53
Figura 5.7 – Conexão Host – P300.	54
Figura 5.8 – Protocolo dos comandos de escrita (write-only).	54
Figura 5.9 – Protocolo dos comandos de escrita (read-only).	55
Figura 5.10 – Fluxograma da transferência de comandos write-only.	56
Figura 5.11 – Fluxograma da transferência de comandos read-only.	57
Figura 5.12 – Estrutura Interface Flags do P300.	57
Figura 5.13 – Fluxograma de uma rotina de atenção.	58
Figura 5.14 – a) Diagrama de blocos do IC SSC P111 b) Pinagem do IC.	59
Figura 5.15 – Características de tensão e corrente do IC P111.	60
Figura 5.16 – Conexão P111 – P300.	60
Figura 5.17 – Filtro harmônico e driver de tensão.	61
Figura 5.18 – a) Magnitude do filtro harmônico (em dB); b) Fase do filtro harmônico (em graus). Gráfico para valores nominais de projeto: $L_1 = 180\mu\text{H}$, $C_4 = 3300\text{ pF}$ e $R_5 = 510\ \Omega$.	62
Figura 5.19 – Diagrama do filtro de entrada e da etapa amplificadora.	62
Figura 5.20 – a) Magnitude do filtro de entrada (em dB); b) Fase do filtro de entrada (em graus). Gráfico para valores nominais de projeto: $L_2 = 120\mu\text{H}$, $L_3 = 180\mu\text{H}$, $C_5 = 10\text{ nF}$, $C_6 = 3300\text{ pF}$, $C_7 = 1500\text{ pF}$ e $R_6 = 180\ \Omega$.	63
Figura 5.21 – Diagrama esquemático do acoplador, com as conexões aos sistemas de entrada e saída do sinal do pacote CEBus.	64
Figura 5.22 – a) Magnitude da relação V_{C7} / V_{AC} . b) Fase	65
Figura 5.23 – Acoplador	65
Figura 5.24 – Característica V_o/V_i do acoplador da Figura 5.23.	66
Figura 5.25 – Simulação de dois dispositivos DLC–P300 (TX e RX)	66
Figura 5.26 – Diagrama de simulação do circuito transmissor (TX)	67
Figura 5.27 – Diagrama de simulação do circuito receptor (RX)	68
Figura 5.28 – Rede elétrica: a) sem sinal do DLC–P300 b) com sinal do DLC–P300	68
Figura 5.29 – Detalhe da Figura 5.28.	69
Figura 5.30 – Sinais no bloco TX: a) saída do P300 (pino SO); b) saída do P111 (pino TXO); c) entrada do transformador de acoplamento.	69
Figura 5.31 – Sinais do bloco RX: a) sinal na entrada do filtro de condicionamento de entrada; b) sinal na saída do filtro de entrada; c) entrada do P300 (pino SI).	70
Figura 5.32 – Sinais no buffer de entrada: a) Sinal na base do transistor Q_4 ; b) Sinal na entrada do P300 (pino SI).	70

Figura 5.33 – Sinal em um dispositivo DLC–P300 no modo de transmissão (em volts): a) saída do P300 (pino SO); b) saída do P111 (pino TXO); c) entrada do transformador de acoplamento. Taxa de aquisição: 10 MHz.	71
Figura 5.34 – Sinal em um dispositivo DLC–P300 no modo de recepção (em volts): a) sinal no entrada do filtro de condicionamento de entrada; b) sinal na saída do filtro de entrada; c) entrada do P300 (pino SI). Taxa de aquisição: 10 MHz.	71
Figura 5.35 – Sinais no buffer de entrada (em volts): a) Sinal na base do transistor Q ₄ ; b) Sinal na entrada do P300 (pino SI). Taxa de aquisição: 25 MHz.	72
Figura 6.1 – Substituição do SSC P300 e do host por um processador DSP.	73
Figura 6.2 – Integração entre o dispositivo CEBus e outras tecnologias de comunicação de dados.	74
Figura 6.3 – Matched filter	76
Figura 6.4 – Características do “matched filter” a) símbolo a ser transmitido b) resposta ao impulso do “matched filter”.	76
Figura 6.5 – Filtro de Correlação para $N_s=2$.	77
Figura 6.6 – Matched filter considerando (6.11) e (6.12).	79
Figura 6.7 – Filtro de correlação considerando (6.11) e (6.12).	79
Figura 6.8 – Representação de um canal AWGN	81
Figura 6.9 – a) Função de autocorrelação de um processo AWGN; b) Espectro de um processo AWGN.	81
Figura 6.10 – Operador C() representando as alterações sofridas pelo sinal em um canal de comunicação.	82
Figura 6.11 – Experimento para obtenção experimental da atenuação em redes elétricas.	82
Figura 6.12 – Gráfico de atenuação x frequência, complexo industrial, período diurno.	83
Figura 6.13 – Gráfico de atenuação x frequência, complexo industrial, período noturno.	84
Figura 6.14 – Gráfico de atenuação em frequência, complexo hospitalar.	85
Figura 6.15 – Gráfico de atenuação em frequência em prédios residenciais.	86
Figura 6.16 – Gráfico de atenuação em frequência de uma residência em bairro de classe média, sem cargas específicas.	87
Figura 6.17 – Gráfico de atenuação em frequência de uma residência em bairro de classe média, com cargas específicas.	87
Figura 6.18 – Tela do MATLAB mostrando a estrutura do comando yulewalk.	89
Figura 6.19 – Estimação do canal de comunicação “tirando um retrato” do canal de comunicação.	90
Figura 6.20 – Aproximação em frequência. Caso: Complexo industrial, período diurno, curva de atenuação I.	91

Figura 6.21 – Aproximação em frequência. Caso: Complexo industrial, período diurno, curva de atenuação IS.	91
Figura 6.22 – Aproximação em frequência. Caso: Prédio residencial, curva de atenuação IH.	92
Figura 6.23 – Aproximação em frequência. Caso: Complexo hospitalar, curva de atenuação A2.	92
Figura 6.24 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: complexo industrial, período diurno. As convenções são as mesmas utilizadas no gráfico da Figura 6.12. Ordem dos filtros encontrados para os casos IS, I, A1, A2: 20ª ordem.	93
Figura 6.25 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: complexo industrial, período noturno. As convenções são as mesmas utilizadas no gráfico da Figura 6.13. Ordem dos filtros encontrados para os casos A1e A2: 20ª ordem; caso I: 19ª ordem.	93
Figura 6.26 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: complexo hospitalar. As convenções são as mesmas utilizadas no gráfico da Figura 6.14. Ordem dos filtros encontrados para os casos I, A1e A2: 20ª ordem.	94
Figura 6.27 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: prédios residenciais. As convenções são as mesmas utilizadas no gráfico da Figura 6.15. Ordem dos filtros encontrados para os casos ISH, IL, OH e OL: 20ª ordem; caso IH: 18ª ordem.	94
Figura 6.28 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: residência em bairro de classe média. As convenções são as mesmas utilizadas no gráfico da Figura 6.16. Ordem dos filtros encontrados para os casos I, O, OCD e OER: 20ª ordem.	95
Figura 6.29 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: residência em bairro de classe média. As convenções são as mesmas utilizadas no gráfico da Figura 6.17. Ordem dos filtros encontrados para os casos ITV, IER, OCD e OER: 20ª ordem.	95
Figura 6.30 – Aproximação da operação de amplificação e quadratura de um sinal aplicado ao ponto X_{in} .	98
Figura 6.31 – a) Aplicação do amplificador e dos operadores não-lineares em um sinal SSC. b) No detalhe, notar a relação de oposição entre a saturação e o corte do transistor e a polaridade do sinal SSC.	98
Figura 6.32 – Seqüência de autocorrelação de $s(n)$.	99
Figura 6.33 – Conversor TTL/PN	99
Figura 6.34 – a) Entrada (X_{in}) e saída (X_{pn}) do arranjo “quadrador SSC” + conversor TTL/PN; b) detalhe das formas de onda.	100

Figura 6.35 – Seqüência de autocorrelação de X_{pn} .	100
Figura 6.36 – Comparação dos gráficos das funções de autocorrelação de X_{in} e de X_{pn} .	101
Figura 6.37 – a) Seqüência de correlação cruzada entre X_{in} e X_{in} ; b) Seqüência de correlação cruzada entre X_{in} e $0.5 \cdot X_{in}$.	102
Figura 6.38 – a) Seqüência de correlação cruzada entre X_{pn} e X_{pn} ; b) seqüência de correlação cruzada entre X_{pn} e X_{pn}' , que são sinais derivados de X_{in} e $0.5X_{in}$.	102
Figura 6.39 – Estrutura do algoritmo de recepção de pacotes CEBus proposta.	103
Figura 6.40 – Estrutura do bloco “Detector de Preâmbulo”	104
Figura 6.41 – Preâmbulo de um pacote CEBus. As letras A, B _i e C demarcam o início do preâmbulo, a ausência de sinal (devido à modulação ASK) e o início do PRE_EOF, respectivamente.	105
Figura 6.42 – a) Cópia local do símbolo a ser transmitido; b) Coeficientes do “matched filter”.	105
Figura 6.43 – Realização do “matched filter”.	106
Figura 6.44 – Algoritmo do bloco “Detector de Preâmbulo”	107
Figura 6.45 – Preâmbulo transmitido (“10011010b”), mostrando o contador de símbolos detectados, multiplicado por um fator de escala de 50, para efeito de visualização.	108
Figura 6.46 – Saída do “matched filter” ($x_{out}(n)$),.	108
Figura 6.47 – Estrutura do bloco “Sincronismo”.	109
Figura 6.48 – Estrutura dos filtros $H_{Acci}(z)$ ($i=1,2,3$).	110
Figura 6.49 – Estrutura geral de $H_{Acci}(z)$.	112
Figura 6.50 – Exemplo da operação módulo N em um vetor $s(n)$, para $p=0$, $p=-1$ e $p=-2$.	112
Figura 6.51 – Gráfico das saídas $ACC_i(n)$, para $N_{ramos} = 3$.	113
Figura 6.52 – Níveis de decisão do algoritmo encontrados (variáveis K_{sinc} e K_{track}).	113
Figura 6.53 – Algoritmo do bloco de sincronismo	114
Figura 6.54 – Estrutura do bloco “Detector de Símbolo”	114
Figura 6.55 – a) seqüência pseudo-aleatória $c_1(n)$; b) seqüência pseudo-aleatória $c_2(n)$; c) multiplicação de $c_1(n)$ e $c_2(n)$;	115
Figura 6.56 – Estrutura de correlação para seqüências pseudo-aleatórias	116
Figura 6.57 – Estrutura alternativa para o cálculo de ACC_1 e ACC_2 .	118
Figura 6.58 – Vetor de símbolos SSC	118
Figura 6.59 – Algoritmo do bloco “Detector de Símbolos”	119
Figura 6.60 – Gráfico das funções $ACC_1(n)$ e $ACC_2(n)$.	120
Figura 6.61 – Gráfico das funções $ACC_1(n)$ e $ACC_2(n)$, enfatizando o recebimento do símbolo EOF.	120
Figura 6.62 – Gráfico das funções $ACC_1(n)$ e $ACC_2(n)$, enfatizando o recebimento do símbolo EOP.	121

Figura 6.63 – Gráfico de correlação, mostrando o padrão ideal (0% de erros), as curvas $ACC_1(n)$ e $ACC_2(n)$ e o nível de decisão (K_{track}) para a validação do símbolo SSC. ____	121
Figura 6.64 – Simulação do algoritmo proposto na seção 6.6. _____	122
Figura 6.65 – Símbolos SSC atenuados pelos seguintes canais: 1-IS, 1-I, 1-A1, 1-A2, 2-A1 e 2-A2. _____	125
Figura 6.66– Símbolos SSC atenuados pelos seguintes canais: 2-I, 3-I, 3-A1, 3-A2, 4-IH e 4-IL. _____	125
Figura 6.67– Símbolos SSC atenuados pelos seguintes canais: 4-ISH, 4-OH, 4-OL, 5-I, 5-O e 5-OCD. _____	126
Figura 6.68 – Símbolos SSC atenuados pelos seguintes canais: 5-OER, 6-IER, 6-ITV, 6-OCD e 6-OER. _____	126
Figura 6.69 – Coeficientes do “matched filter”, com as marcações dos blocos. _____	128
Figura 6.70 – Rearranjo dos coeficientes para otimizar a estrutura do cálculo de $H_{rc}(z)$. _	129
Figura 6.71 – Estrutura para o cálculo do “matched filter” utilizando a instrução MACD (somente processadores da Texas®). _____	129
Figura 6.72 – Realização do “matched filter” $H_{rc}(Z)$ a partir de operações XOR _____	130
Figura 6.73 – Tabela da soma dos bits de uma operação XOR. _____	131
Figura 6.74 – Quebra de X e s em palavras menores (16 bits), denominadas wd_i e wc_i , respectivamente. Cada palavra é formada por 2 bytes (MSB e LSB). _____	131
Figura 6.75 – a) Algoritmo para atualização da memória do filtro; b) Algoritmo para o cálculo de $x_{out}(n)$. _____	132
Figura 6.76 – Simulação no cálculo da função de correlação utilizando a Figura 6.72 ____	133

Índice de Tabelas

Tabela 1.1 – Comportamento do mercado em sistemas “networking” _____	6
Tabela 2.1 – Algumas propriedades do operador linear $\varepsilon ()$ _____	10
Tabela 2.2 - Propriedades de uma sequência pseudo-aleatória _____	15
Tabela 3.1 – Comparação entre X-10, CEBus e LONworks _____	26
Tabela 4.1 – Parâmetros do Chirp _____	36
Tabela 4.2 – Codificação dos símbolos no protocolo CEBus _____	38
Tabela 4.3 – Características nominais do transformador trifásico da Figura 4.21 _____	47
Tabela 4.4 – Atenuação do Sinal no Experimento da Figura 4.22 _____	48
Tabela 5.1 – Principais componentes do modem DLC–P300 _____	51
Tabela 6.1 – Preços (em US\$, sem carga tributária) dos componentes base. _____	74
Tabela 6.2 – Atraso T de s(n-T) calculado a partir do parâmetro p. _____	110
Tabela 6.3 – Tabela de detecção de símbolos CEBus _____	119
Tabela 6.4 – Parâmetros de simulação do algoritmo de recepção SSC _____	122
Tabela 6.5 – Atenuação média do símbolo SSC ao passar pelo canal H(z) _____	123
Tabela 6.6 – Resultados da simulação _____	124
Tabela 6.7 – Análise dos erros da simulação _____	127

Resumo

Este trabalho tem por objetivo abordar um novo tipo de método de comunicação que vem emergindo nos últimos anos: o uso das linhas elétricas como meio para comunicação entre equipamentos. As técnicas pesquisadas nesta dissertação utilizam o conceito de “espalhamento espectral”, ou Spread Spectrum (SS). Este tipo de técnica é superior a outras técnicas utilizadas (FSK, QFSK, Carrier, etc.) por utilizar não uma portadora, mas um conjunto de portadoras de frequências diferentes. A característica de espalhamento confere um importante conjunto de vantagens em relação a outras técnicas: (a) Grande imunidade a ruídos (Podendo o ruído ter até mesmo a mesma ordem de grandeza do sinal). (b) Imunidade à interferência intencional (“Jam”). (c) Atenuação da sensibilidade à variação da impedância da linha. (d) Alta relação sinal-ruído.

A dissertação aborda duas linhas: a primeira linha de pesquisa visa a construção de um equipamento para a transmissão de dados utilizando a rede elétrica implementado com circuitos integrados especiais dedicados a gerenciar todo o processo de transmissão e recepção de dados em redes elétricas. Tais circuitos, fabricados pela Intellon®, oferecem um padrão estabelecido pela EIA (Electronics Industries Association), conhecido como protocolo CEBus®, que tem a tendência de se tornar padrão para a comunicação em sistemas de automação residencial e predial.

A segunda linha de pesquisa visa a proposição e implementação de algoritmos baseados em técnicas de processamento digital de sinais para efetuar a transmissão e a recepção dos pacotes CEBus pela rede elétrica, através da implementação dos algoritmos propostos em processadores digitais de sinais (DSP).

Na dissertação, também se abordará as características dos meios de transmissão e seu fator de degradação, uma vez que estes têm importância fundamental na recuperação do sinal. Circuitos condicionadores e filtros também serão abordados nesta etapa.

Com este trabalho, espera-se consolidar tecnologias de forma a implantar sistemas eficientes de comunicação com tecnologia Spread Spectrum, para utilização em uma grande gama de aplicações, tais como em automação predial e residencial, automação do sistema de distribuição (sistemas de medição on-line e tarifas horo-sazonais), automação industrial (sistemas de coleta de dados e envio de comandos remotos, etc.), sistemas hospitalares, hotelarias, etc.

Abstract

This work has for objective to approach a new type of communication method that is emerging in the last decades: the use of the electric power lines as medium for communication among equipment's. The techniques researched in this dissertation they use the concept of spread spectrum (SS). This technique is superior to other used (FSK, QFSK, Carrier, etc.), because the use of multiple tones of many different frequencies. The spread characteristic checks an important group of advantages in relation to other techniques: (a) Great immunity to noises (Being able to the noise to have even the same order of greatness of the sign). (b) Immunity to the intentional interference (JAM). (c) Attenuation of the sensibility to the variation of the impedance of the line. (d) High signal-to-noise rate.

The dissertation approaches two lines: the first research line seeks the construction of an equipment for the transmission of data using the electric power line implemented with special integrated circuits, dedicated to the administration of whole the transmission process and reception of data in electric power lines. Such IC's, manufactured by Intellon[®], offer an established pattern for EIA (Electronics Industries Association), known as *CEBus* protocol, that has the tendency of becoming standard for the communication in residential and/or building automation systems.

The second research line seeks the proposition and implementation of algorithms based on digital signal processing techniques to make the transmission and the reception of *CEBus* packages for the electrical networking, using implementations of the proposed algorithms in digital signal processors (DSP's).

In this dissertation, it will also be approached the characteristics of the transmission medium and your degradation factor, once these have fundamental importance in the recovery of the sign. Conditioning circuits and filters will also be approached in this stage.

With this work, it hopes to consolidate the technologies to implant efficient communications systems with spread spectrum technology, for use in a great range of applications, such as in building and residential automation, automation of the distribution system (systems of on-line measurement and seasonal tariffs), industrial automation (systems of data collection and remote commands, etc.), hospital systems, hostelryes, etc.

1. Introdução

Enquanto que a idéia de enviar sinais elétricos para a comunicação no mesmo cabeamento da rede de energia elétrica é tão velha quanto o próprio telégrafo, o número de dispositivos de comunicação instalados com cabeamentos dedicados excede em muito o número de dispositivos instalados que utilizam a rede elétrica AC como meio de transmissão [41]. A razão para isto não é o resultado da negligência das possibilidades da comunicação utilizando a rede elétrica nas últimas décadas, como se poderia pensar, pois já em meados de 1920 foram enviados dois pedidos de patentes ao “*American Telephone and Telegraph Company*”, na área de “*Carrier Transmission Over Power Circuits*”. As patentes, nºs 1.607.668 e 1.672.940, apresentadas em 1924, mostraram sistemas para a transmissão e recepção de sinais de comunicação em redes AC trifásicas. As principais razões para que, até 20 anos atrás, não houvesse o desenvolvimento em larga escala de dispositivos que usassem a rede elétrica AC (*powerline*) para o transporte de informações foram [36]:

- a) A necessidade do projeto de uma tecnologia que fosse eficiente em superar as dificuldades impostas pela rede elétrica;
- b) A falta de um protocolo de comunicação que pudesse ser considerado padrão;
- c) O alto custo do projeto e implantação de novas tecnologias no setor;

O primeiro tópico (superação das dificuldades impostas pela rede elétrica) é derivado das características da rede elétrica, que são resumidas a seguir [39]:

- a) Grandes variações da impedância da rede elétrica com o tempo;
- b) Grandes variações da sensibilidade em frequência;
- c) Várias cargas geradoras de ruídos (motores universais, chaveamento de cargas, sistemas chaveados, etc).

O segundo tópico (falta de um protocolo padrão) é derivado da resistência das indústrias em promover um modo único de compartilhamento de dados entre seus equipamentos. Isto foi em parte resolvido pela decisão, na década de 90, pelas grandes empresas do setor eletro-eletrônico, de adotar o padrão *CEBus* da EIA (*Electronics Industries Association*) [36];

O terceiro tópico (alto custo de projeto) vem sendo aos poucos minimizado, em função do uso de sistemas VLSI e da adoção de tecnologias em DSP.

Um dos grandes problemas na transmissão de dados em redes de energia elétrica é a injeção de portadoras de alta frequência, necessárias para se alcançar taxas de comunicação adequadas à automação residencial, comercial e industrial. A rede de energia elétrica, bem como seus componentes (transformadores, capacitores, linhas de transmissão, motores, etc.) tem seu projeto fixado para que haja uma maior eficiência na frequência de operação de 60 Hz, ou em outras palavras, a rede elétrica é projetada para fornecer potência, e não para transmitir dados. Os principais efeitos na transmissão de dados em redes elétricas podem ser vistos na Figura 1.1 [42].

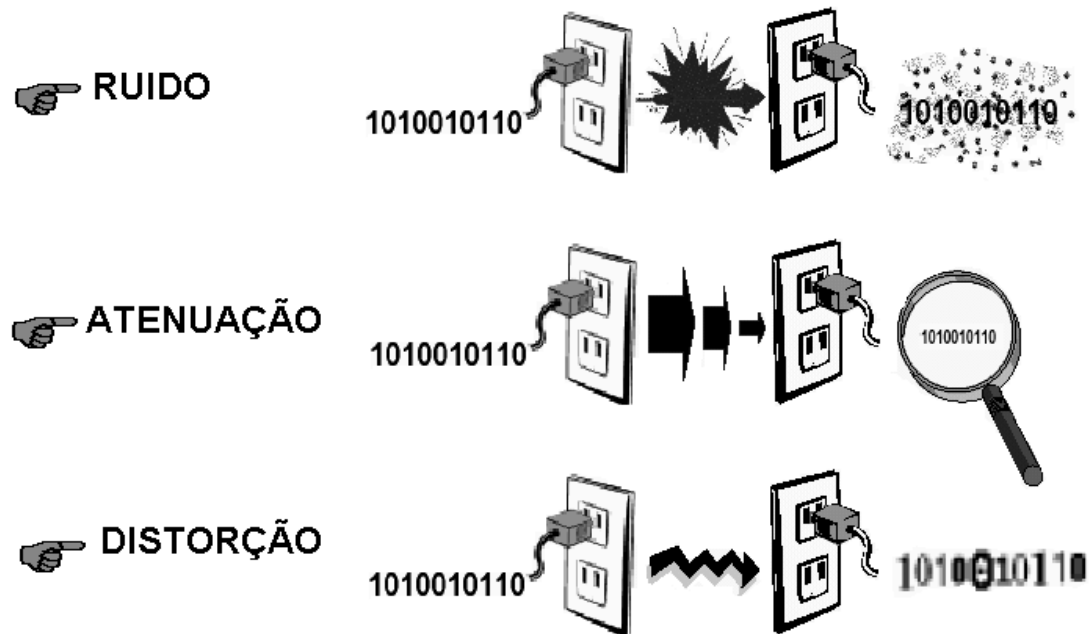


Figura 1.1 – Principais obstáculos enfrentados na transmissão de dados: ruído, atenuação e distorção.

Uma vez superadas estas dificuldades, uma série de vantagens pode ser observada, como por exemplo, no controle de processos distribuídos. Sistemas de controle distribuídos são hoje a principal aplicação de sistemas *powerline*, como são chamados os sistemas que utilizam a rede elétrica para comunicar dados [36]. As aplicações vão desde o controle de ambientes em edifícios até, passando até pela administração e operação do sistema elétrico de distribuição. Um exemplo seria o controle de aquecimento, da ventilação e do sistema de ar-condicionado em um edifício comercial. Um sistema de controle deste tipo tem muitos sensores, extensamente espalhados, que medem tanto a umidade do ar quanto a temperatura. Sensores adicionais monitoram a operação destes equipamentos. Em resposta às condições determinadas pelos sensores, o controle do sistema pode tomar certas atitudes, como, por exemplo, fechar ou abrir um duto de ventilação.

Nos sistemas comuns (i.e, que não utilizam a tecnologia *powerline*), cada um dos sensores (ou nós), é ligado através de um cabo diretamente ao controle central. Em geral, tal cabeamento (com raras exceções) é extremamente dispendioso, pois implica em duas situações específicas:

- a) O projeto do cabeamento estruturado no ato da construção do edifício pressupõe quais os sensores que serão utilizados, a quantidade e a localização de cada um deles, o que é praticamente impossível de se obter;
- b) A construção do cabeamento estruturado depois da construção da estrutura que irá abrigar o sistema de controle nem sempre é possível, em virtude de não haver rotas para os cabos, ou quando houver, se tornar dispendioso demais.

Em geral, o uso de sistemas baseado em tecnologia *powerline* pode ser empregado nas seguintes situações [38]:

- a) é impraticável ou com custo proibitivo a instalação de cabeamento exclusivo;
- b) é preciso transferir baixas ou médias quantidades de dados entre múltiplos nós;
- c) a rede AC (ou DC) está disponível para a comunicação;

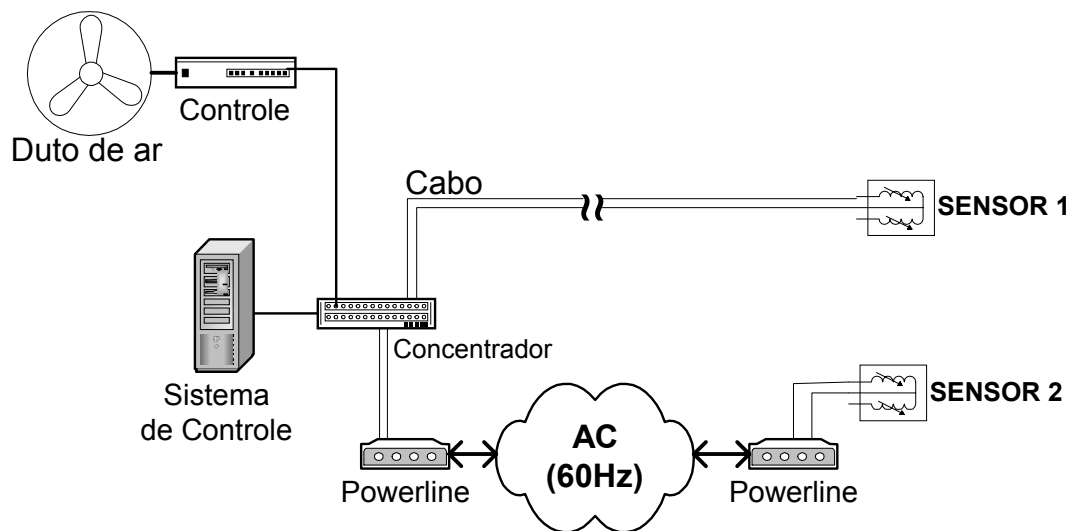


Figura 1.2 – Sistema distribuído.

Seja o exemplo anteriormente citado, representado na Figura 1.2. O sensor 1 é ligado ao sistema de controle por meio de um cabo. Já o sensor 2, é ligado ao sistema de controle por meio de um link *powerline*, que dispensa o uso de cabos para a conexão sensor-controlador. Neste caso, é necessária a conexão de um equipamento modulador ao

sensor e a conexão de um equipamento demodulador ao concentrador da rede. A vantagem do uso de um link *powerline* é que novos sensores podem ser instalados sempre que houver o aumento da necessidade de uma melhor monitoração.

Mais exemplos do uso de sistemas *powerline* são listados a seguir [38]:

- a) Sistemas de iluminação (controle de luminosidade);
- b) Sistemas eletrônicos de segurança (sensores de presença, alarmes de incêndio, etc.);
- c) Módulos aplicativos: sistema de automação residencial/comercial, para permitir a comunicação entre dispositivos eletro-eletrônicos, tais como refrigeradores, fornos de micro-ondas, máquinas de lavar, secadoras, etc.);
- d) Controle de ambientes (sensores de temperatura e qualidade do ar, controle de temperatura e umidade – HVAC);
- e) Gerenciamento de energia (medidores, termostatos, controle de aquecimento, etc.);
- f) Gerenciamento de tráfego (controle de semáforos, controle de poluição em túneis, etc);

Uma pergunta deve ser formulada neste ponto: qual o maior motivo para o aumento da demanda de produtos com tecnologia *powerline*? Algumas esclarecimentos podem ser dados [39]:

- a) O uso “explosivo” da Internet na comunicação de dados abriu as portas para a revolução da informação. A distribuição da informação passa ser ferramenta vital no cotidiano. Por exemplo, a música que foi retirada da Internet (*download*) pelo PC pode ser tocada no aparelho de som, sem que haja nenhuma intervenção ativa na troca de informações entre os dois dispositivos;
- b) O florescimento de novos serviços baseados em Internet, e a possibilidade de integração ilimitada atraem cada vez mais consumidores. Por exemplo, o celular que controla o forno de micro-ondas à distância e comanda a abertura do portão da garagem se torna ferramenta de integração ao cotidiano;
- c) A facilidade do uso pelos consumidores e a facilidade de instalação dos dispositivos *powerline* (“compre, plugue, use”) atrai novos consumidores;

Isto faz com que o mercado seja altamente promissor, estando em plena expansão e gerando uma expectativa das indústrias que chegam a bilhões de dólares. A Figura 1.3 mostra a porcentagem de proprietários de produtos eletrônicos, desde 1990, e fazendo uma projeção baseada no panorama de consumo atual nos Estados Unidos [37].

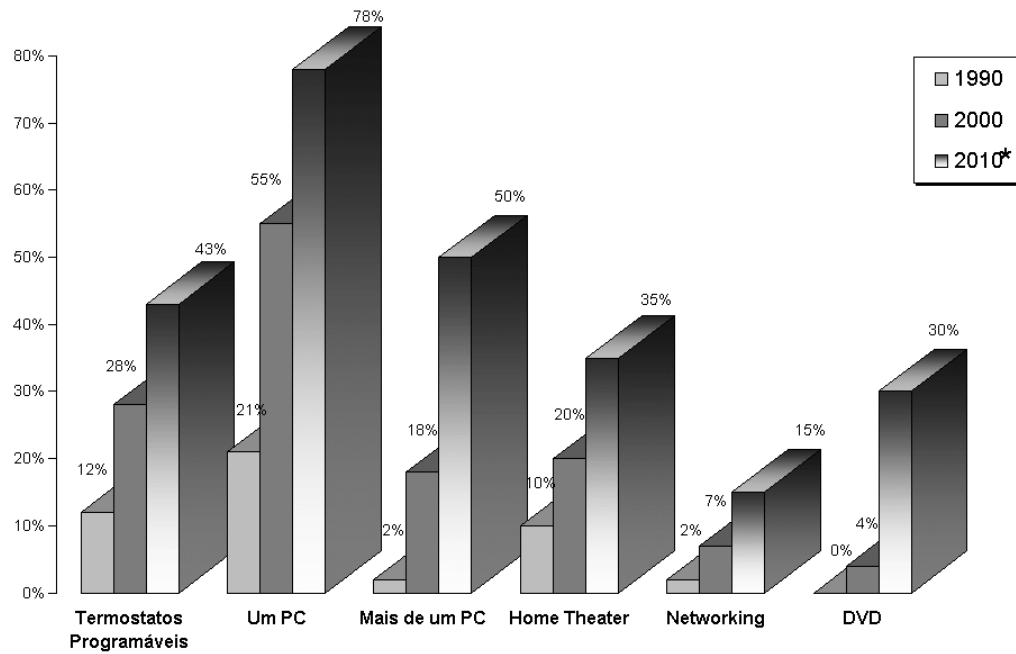


Figura 1.3 – Características dos proprietários de bens eletrônicos nos Estados Unidos (*2010: projeção). Fonte: *Park's Associates*, Outubro de 2000.

Não só o panorama de consumo está em franca expansão: a prestação de serviços de tecnologia tem crescido a passos largos, como pode ser visto na Figura 1.4, que mostra a evolução dos serviços prestados na área de tecnologia nos EUA [37].

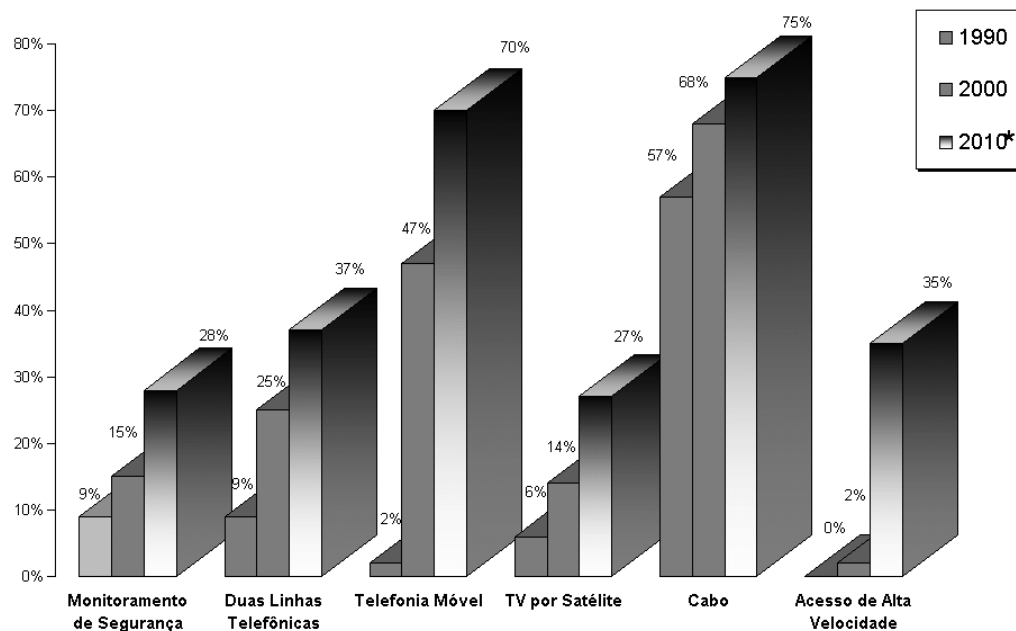


Figura 1.4 – Evolução dos serviços prestados nos Estados Unidos (*2010: projeção). Fonte: *Park's Associates*, Outubro de 2000.

Como pode ser visto na Figura 1.3 e na Figura 1.4, o aumento da demanda na área de comunicação é evidente. A Tabela 1.1 mostra o comportamento do mercado em função do aumento da demanda de novos produtos e serviços, segundo os especialistas da área.

Tabela 1.1 – Comportamento do mercado em sistemas “networking”

Categoria	Valor
<i>Home Networking</i>	EUA: US\$ 6 bi em 2000, US\$ 14 bi em 2002. Mercado mundial: US\$ 42 bi em 2002.
<i>SOHO Networking</i>	EUA: US\$ 60 bi em 2000 Alemanha: US\$ 40 bi
Expectativa de serviços com sistemas <i>powerline</i>	Mercado Mundial: US\$ 1 bi

O gráfico da Figura 1.5 mostra o crescimento do número de usuários que usufruem os serviços de informação e informática, tanto na área de computação como na área de “home networking” [39].

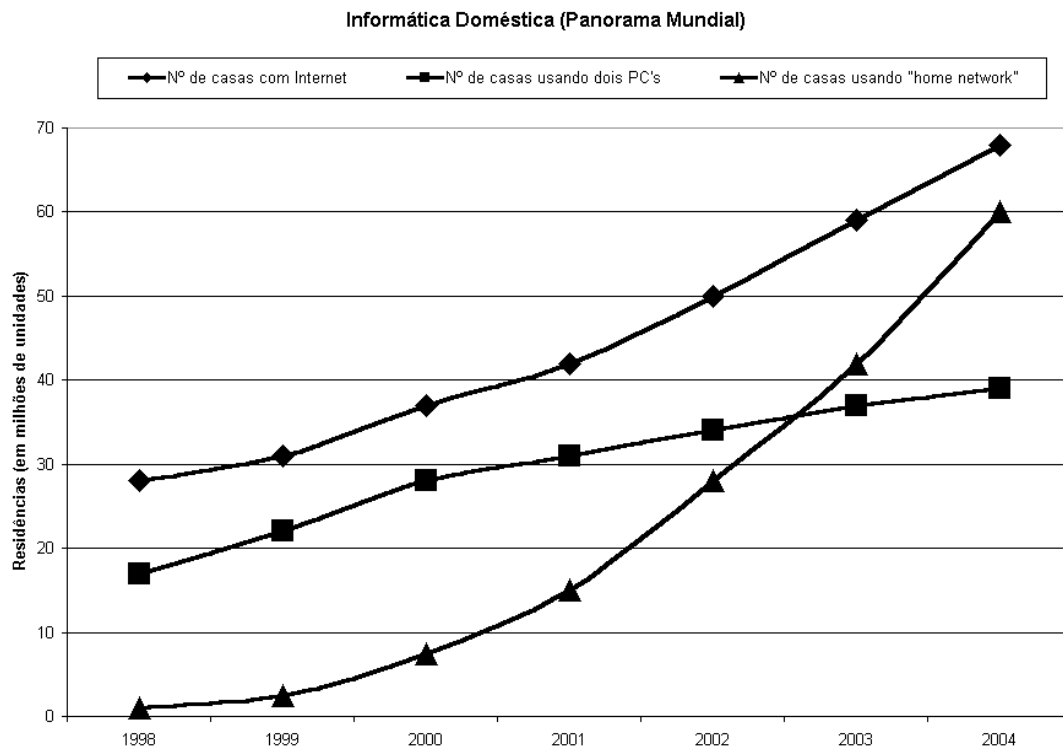


Figura 1.5 – Evolução dos serviços de informação no mundo (Fonte: 3R Corp).

Este trabalho nasceu de uma parceria entre a Faculdade de Engenharia da Universidade Federal de Juiz de Fora, na pessoa do Prof. Carlos Augusto Duque, D.S.c, e a ATI (Automação e Telecomunicação Industrial), empresa situada no município de Belo Horizonte, no estado de Minas Gerais. Tal parceria tem o objetivo de desenvolver a tecnologia de sistemas *powerline* e gerar produtos nas áreas de automação industrial e telemetria, bem como no desenvolvimento de sistemas distribuídos inteligentes. Neste trabalho, serão apresentados os resultados das pesquisas na área de sistemas *powerline*. Este trabalho envolve três grandes áreas de concentração:

- 1º. Verificação do estado da arte do desenvolvimento de sistemas *powerline*;
- 2º. Projeto e desenvolvimento de um equipamento de baixo custo (intitulado “*Digital Line Carrier*”) para a comunicação de dados em ambientes residenciais, comerciais e industriais;
- 3º. Pesquisa e desenvolvimento de algoritmos para a comunicação de dados com sistemas *powerline* que envolvam técnicas de processamento digital de sinais aplicados a processadores digitais de sinais (DSP).

Na primeira área (Estado da arte) foram efetuadas pesquisas sobre os seguintes tópicos:

- a) Revisão bibliográfica das tecnologias de comunicação em banda estreita (*narrow band*) e em banda larga (*spread spectrum*);
- b) Estudo dos protocolos atualmente utilizados para a comunicação em redes elétricas;
- c) Estudo da evolução da comunicação em redes de energia elétrica;
- d) Estudo dos canais de comunicação.

Na segunda área (projeto do *Digital Line Carrier*), os seguintes tópicos foram abordados:

- a) Estudo de condicionamento de sinais (filtros passivos e ativos);
- b) Estudo dos circuitos integrados necessários ao projeto;
- c) Projeto de software de baixo nível (*Assembly*) para o *kernel* do DLC;
- d) Projeto de software de alto nível (C++) para o controle do DLC;
- e) Estudo do protocolo serial RS-232.

É necessário destacar que a empresa ATI, nesta parte, teve participação crucial no desenvolvimento do DLC, tanto pela confecção do hardware, como pelo ajuste necessário dos componentes ao projeto, e também pelo ajuste do projeto aos componentes disponíveis.

Na terceira parte (Pesquisa e desenvolvimento de algoritmos), foram abordados os seguintes temas:

- a) Estudo de técnicas de processamento digital de sinais aplicados à comunicação de dados;
- b) Estudo da implementação dos algoritmos de *spread spectrum*;
- c) Estudo da modelagem de canais de comunicação para o teste de algoritmos;
- d) Estudo de algoritmos aplicados a processadores digitais de sinais (DSP).

Para englobar as três partes da pesquisa, os capítulos foram organizados da seguinte forma:

- a) Capítulo 2: Fundamentos de Modulação Spread Spectrum. Este capítulo introduz o conceito de espalhamento de espectro em sistemas de comunicação digital;
- b) Capítulo 3: Técnicas de Transmissão de Dados em Redes Elétricas. Visa referenciar as técnicas de comunicação em redes elétricas mais utilizadas;
- c) Capítulo 4: Transmissão de Dados em Redes de Energia Elétrica Utilizando Técnicas de *Spread Spectrum Carrier*. Este capítulo visa estudar o protocolo *CEBus*, bem como o funcionamento das técnicas de Spread Spectrum Carrier (SSC) para a transmissão de dados em redes elétricas, bem como o estudo de propagação dos sinais em sistemas com transformadores de distribuição;
- d) Capítulo 5: Projeto do modem DLC–P300 (*Digital Line Carrier*). Este capítulo visa especificar o projeto do DLC – P300, os componentes utilizados e suas conexões, os circuitos de condicionamento utilizados, o projeto do software de baixo nível (*kernel*) do DLC, bem como simulações em PSPICE e testes reais no protótipo construído;
- e) Capítulo 6: Proposição de Algoritmos de Transmissão e Recepção de *Spread Spectrum Carrier* Baseados em Técnicas DSP. Este capítulo aborda todas as questões referentes ao projeto de algoritmos para transmissão e recepção de pacotes CEBus para serem implantados em processadores digitais de sinais. A modelagem dos canais de comunicação também será abordada.
- f) Capítulo 7: Conclusões. Este capítulo encerra a dissertação com as conclusões do trabalho.

2. Fundamentos de Modulação Spread Spectrum

2.1. Introdução

Neste capítulo serão abordados os fundamentos dos sistemas de comunicação que envolva modulação *spread spectrum*. Serão analisados os fundamentos matemáticos, bem como um breve estudo das seqüências pseudo-aleatórias. Serão apresentados resultados de simulação para que haja um melhor entendimento dos sistemas *spread spectrum*. Não serão apresentados os fundamentos de sistemas SS que operem com mudança de frequência (*Frequency Hopping*), que podem ser encontrados em [9] e [18].

2.2. Revisão histórica

As aplicações iniciais das técnicas de *spread spectrum* (SS) foram no desenvolvimento de sistemas de comunicação militares. No fim da II Guerra Mundial, a utilização de sistemas SS com o propósito de evitar interferências intencionais era familiar aos engenheiros que desenvolviam sistemas de radar. Durante os anos subseqüentes à guerra, a motivação para a pesquisa de sistemas que utilizassem a tecnologia SS era primariamente com vistas ao desenvolvimento de sistemas de comunicação com alta rejeição a sinais de interferência intencional. Porém, o resultado destas pesquisas trouxe à tona diversas soluções para as mais diversas áreas, como sistemas de comunicação de múltiplo acesso e sistemas de alta resolução, dentre outras. O nome *spread spectrum* deriva do fato de que a largura de banda utilizada para a transmissão do sinal é muito maior do que a largura de banda mínima necessária se fosse utilizado somente uma portadora de frequência única [9].

A comunicação de dados usando técnicas de SS ficou restrita ao meio militar até meados da década de 80, quando foi introduzida nos sistemas de comunicação comercial, como, por exemplo, em telefonia celular CDMA (*Code Division Multiplex Access*) [10].

2.3. Fundamentos de um sistema SS

Seja um sistema de comunicação representado pelo diagrama da Figura 2.1 e composto de três partes básicas: o transmissor, o meio em que o sinal se propaga e o receptor [10]:

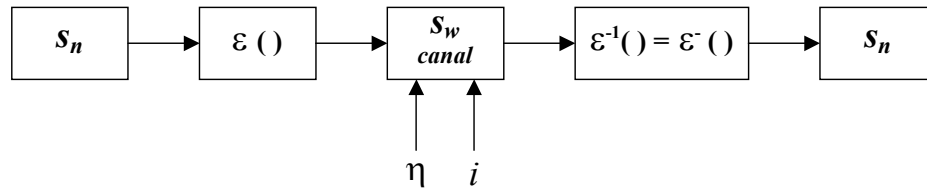


Figura 2.1 - Conceito de Comunicação Spread Spectrum

O sistema proposto na Figura 2.1 é definido como sendo *spread spectrum* (SS) se e somente se atender a três regras básicas [9]:

- O sinal ocupa uma largura de banda (W_{ss}) muito maior do que o mínimo necessário para a transmissão do sinal (W).
- O espalhamento do sinal (*spreading*) é efetuado pelo operador linear $\varepsilon()$ [10], também chamado de código de espalhamento, que independe do sinal a ser enviado.
- A recuperação do sinal (*despreading*) é efetuada pela correlação do sinal espalhado com a cópia do operador $\varepsilon^{-1}()$ utilizada para espalhar o sinal, localizada no receptor.

O operador $\varepsilon()$ tem algumas características especiais, e algumas delas são mostradas na Tabela 2.1.

Tabela 2.1 – Algumas propriedades do operador linear $\varepsilon()$

$\varepsilon^{-1}() = \varepsilon^{\sim}()$	(2.1)
$\varepsilon^{-1}(\varepsilon(s_n)) = \varepsilon(\varepsilon^{\sim}(s_n)) = s_n$	(2.2)
$\varepsilon(k_1sa_w + k_2sb_w) = k_1 \cdot \varepsilon(sa_w) + k_2 \cdot \varepsilon(sb_w)$	(2.3)

Como visto em (2.1) e (2.2), as operações de *spreading* e *despreading* são as mesmas, sendo $\varepsilon()$ um operador linear. Outras propriedades do operador de espalhamento são mostradas no restante deste capítulo.

Os sistemas SS possuem algumas características especiais, que são relacionadas nos tópicos a seguir:

2.3.1. Imunidade à Interferência

O sistema *spread spectrum* tem alta capacidade de rejeição de interferências (intencionais ou não). Para a análise, considere o sistema da Figura 2.1. Aplicando-se o

operador $\varepsilon^{-1}(\cdot)$ a um sinal recebido, s_w , que foi corrompido por um sinal de banda estreita, i_n , como mostrado na Figura 2.2, tem-se:

$$\varepsilon^{-1}(s_w + i_n) = \varepsilon^{-1}(\varepsilon(s_n)) + \varepsilon^{-1}(i_n) = s_n + i_w \quad (2.4)$$

Como mostrado em (2.4) o operador $\varepsilon^{-1}(\cdot)$ restaura o sinal original, porém, espalha o espectro da interferência por toda a largura de banda W_{ss} . Aplicando um filtro de largura de banda W (igual à largura de banda de s_n), tem-se:

$$F(s_n + i_w) = s_n + i_{wr} \quad (2.5)$$

Observa-se, de acordo com (2.5), que a energia remanescente do sinal i_n após a operação de filtragem, denotada por $F(\cdot)$, é muito menor que a energia do sinal s_n . Ou seja, os operadores $\varepsilon^{-1}(\cdot)$ e $F(\cdot)$, aplicados em cascata, aumentam a robustez do sistema à interferência intencional.

O processo de *despreading* pode ser observado na Figura 2.2. A Figura 2.2a mostra o sinal espalhado $s_w = \varepsilon^{-1}(s_n)$ contaminado pela interferência i_n . A Figura 2.2b mostra o resultado ao se passar o sinal $s_w + i_n$ pelo operador de *despreading*, $\varepsilon^{-1}(\cdot)$. Conforme pode ser observado, a energia da interferência é espalhada pela banda W_{ss} . Finalmente a Figura 2.2c mostra o resultado da operação de filtragem no sinal $s_n + i_w$. Esta operação elimina grande parte da energia do sinal de interferência.

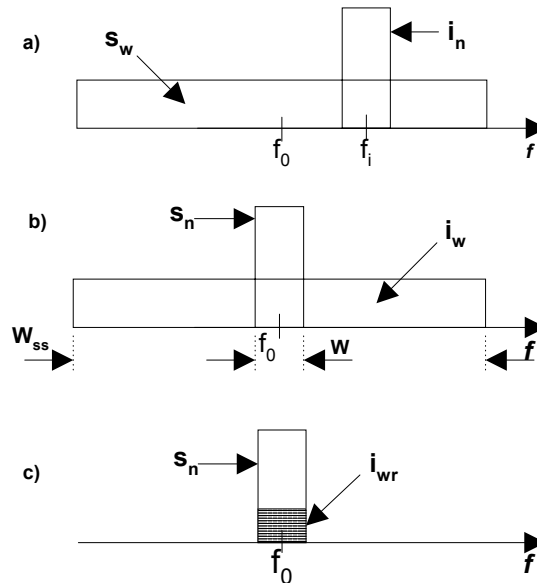


Figura 2.2 - Recuperação do sinal com interferência: a) Sinal recebido, b) Resultado do operador $\varepsilon^{-1}(\cdot)$ (*despreading*). c) Resultado do operador $F(\cdot)$.

Calculando a relação de potências entre i_w e i_{wr} , pode-se afirmar que:

$$\begin{aligned} P(i_w) &= G \cdot P(i_{wr}) \\ G &= \frac{W_{ss}}{W} \end{aligned} \quad (2.6)$$

O parâmetro G em (2.6) define qual a quantidade de energia do sinal i_n que é suprimida, sendo chamado de ganho de processo, ou seja, G define proporcionalmente a invulnerabilidade do sistema à interferência [10].

2.3.2. Imunidade a reflexões sucessivas

Em um ambiente que provoca a reflexão do sinal transmitido, o sinal recebido será uma combinação do sinal transmitido com múltiplas cópias atrasadas. Na maioria das vezes, as componentes refletidas do sinal são tidas como interferências, agindo destrutivamente. Seria desejável que os sinais pudessem ser separados e combinados construtivamente, anulando o efeito da multi-propagação. Esse efeito pode ser conseguido utilizando o conceito de *spread spectrum*. Seja o sinal recebido representado por:

$$r(t) = \sum_{l=0}^{N_\beta-1} \beta_l \cdot s_w(\tau_l) = \sum_{l=0}^{N_\beta-1} \beta_l \cdot \varepsilon_{\tau_l}(s_n(\tau_l)) \quad (2.7)$$

Aplicando o operador $\varepsilon^{-1}()$ (*despreading*) com um atraso de τ_s ao sinal recebido, tem-se:

$$\varepsilon_{\tau_s}(r(t)) = \sum_{l=0}^{L-1} \varepsilon_{\tau_s}(\beta_l \cdot s_w(\tau_l)) = \sum_{l=0}^{L-1} \varepsilon_{\tau_s}(\beta_l \cdot \varepsilon_{\tau_l}(s_n(\tau_l))) \quad (2.8)$$

As componentes do sinal dependem somente da diferença entre τ_s e τ_l , ou seja, o operador $\varepsilon()$ pode ser descrito como $\varepsilon_{s-l}()$, e a operação de *despreading* resulta em:

$$\varepsilon_{s-l}(s_n(\tau_l)) \Rightarrow \begin{cases} s_n(\tau_l) & \text{se } \tau_s - \tau_l = 0 \\ s_w(\tau_l) & \text{se } \tau_s - \tau_l > 0 \end{cases} \quad (2.9)$$

ou seja, o operador $\varepsilon()$ é sincronizado com uma componente do sinal com atraso $\tau_s = \tau_m$. Após a filtragem do sinal por um filtro com largura de banda W , o sinal resultante é:

$$\begin{aligned}
 F(\varepsilon_{rm}(r)) &= \beta_m \cdot s_n(\tau_m) + \sum_{\substack{l \neq m \\ l=0}}^{L-1} \beta_l \cdot s_{wr}(\tau_l) \\
 &= \beta_m \cdot s_n(\tau_m) + R_m
 \end{aligned} \tag{2.10}$$

O termo R_m é caracterizado por ter baixa energia, deste modo o sinal é completamente recuperado, mesmo na presença de reflexões.

2.3.3. Capacidade de Múltiplo Acesso utilizando CDMA (*Code Division Multiplex Acess*)

Para aumentar a capacidade de transmissão utilizando a mesma largura de banda, vários sinais espalhados podem coexistir na mesma banda de transmissão. Seja o sinal proveniente do receptor, $r(t)$:

$$r(t) = \sum_k s_{wk} = \sum_k \varepsilon_k(s_{nk}) \tag{2.11}$$

A equação (2.11) define uma soma de k sinais com largura de banda W_{ss} , onde $\varepsilon_k(\)$ define o operador de espalhamento do sinal k . Para que os sinais possam coexistir e ser restaurados, é necessário uma propriedade adicional do operador $\varepsilon(\)$:

$$\varepsilon_i^{-1}(s_{wj}) = \varepsilon_i(\varepsilon_j(s_{nj})) = \begin{cases} s_{ni}, & i = j \\ s_{wij}, & i \neq j \end{cases} \tag{2.12}$$

Logo, a aplicação do operador $\varepsilon_k(\)$, $k=i$, ao sinal da equação (2.11), e utilizando a propriedade da equação (2.12), tem-se:

$$\varepsilon_i^{-1}\left(\sum_k s_{wk}\right) = s_{ni} + \sum_{\substack{k=1 \\ k \neq i}}^K s_{wik} \tag{2.13}$$

Aplicando o operador $F(\)$ em (2.13), o sinal resultante fica:

$$F\left(\varepsilon_i^{-1}\left(\sum_k s_{wk}\right)\right) = s_{ni} + \sum_{\substack{k=1 \\ k \neq i}}^K s_{rik} \tag{2.14}$$

Em outras palavras, o processo de *despreading* resulta em um sinal de alta energia, s_{ni} , e um somatório de sinais de baixa energia (s_{rik}), pois a separação dos sinais é baseada na baixa correlação entre ε_i e ε_j , $\forall i \text{ e } j, i \neq j$. Para que a propriedade descrita em (2.12) seja alcançada, os operadores ε_k são gerado por diferentes códigos pseudo-aleatórios, daí o nome CDMA (*Code Division Multiplex Access* – Múltiplo Acesso por Divisão de Código).

2.4. Seqüências Pseudo-Aleatórias

O estudo de seqüências pseudo-aleatórias é vital para a compreensão de como ocorre a implementação dos operadores ε . Uma seqüência pseudo-aleatória é definida como sendo uma seqüência determinística, mas que exhibe propriedades estatísticas de um ruído branco amostrado [9]. Seja uma seqüência $c(n)$, definida em (2.15) [18]:

$$\{ c(n) \}_{n=0}^{N-1} = \pm 1 \quad (2.15)$$

A equação (2.15) exhibe propriedades de uma seqüência pseudo-aleatória se apresentar três propriedades básicas: *Balanço*, *Corrida* e *Correlação*.

- a) *Balanço*: O número de dígitos igual a 1 de $c(n)$ difere do número de dígitos igual a -1 por no máximo uma unidade;
- b) *Corrida*: Por corrida, se define como uma seqüência contínua de +1 ou -1. A mudança de estado de +1 para -1 (ou vice-versa) define uma nova corrida. O comprimento da corrida é o número de dígitos da corrida. Para uma seqüência pseudo-aleatória, é desejável que metade do número de corridas de mesmo sinal (+1 ou -1) seja de comprimento 1, das corridas tenha comprimento 2, tenha comprimento 3, e assim por diante;
- c) *Correlação*: Se a seqüência $c(n)$ é comparada dígito a dígito com uma cópia deslocada $c(n-k)$, a diferença do número de ocorrências iguais ($c(n)=c(n-k)$) para o número de ocorrências diferentes ($c(n) \neq c(n-k)$) não deve ser maior que uma unidade;
- d) *Periodicidade*: Assumindo $c(n)$ periódica, com tamanho N, podemos dizer que, para qualquer k inteiro, $c(n)=c(n+kN)$.

As propriedades que traduzem as características pseudo-aleatórias são mostradas na Tabela 2.2.

Tabela 2.2 - Propriedades de uma seqüência pseudo-aleatória

Propriedade 1: <i>Balanço</i>	$\sum_{n=0}^{N-1} c(n) \approx \pm 1 \quad (2.16)$
Propriedade 2: <i>Corrida</i>	$\frac{1}{2^p} = p, \text{ onde } \frac{1}{2^p} \text{ é o tamanho da corrida} \quad (2.17)$
Propriedade 3: <i>Periodicidade</i>	$\sum_{n=0}^{N-1} c(n) \cdot c(n+k \cdot N) = N, \forall k = 0,1,2,\dots \quad (2.18)$
Propriedade 4: <i>Correlação</i>	$\frac{1}{N} \sum_{n=0}^{N-1} c(n) \cdot c(n+i) = \begin{cases} 1, & i = 0 \\ 1/N, & 0 < i < N \end{cases} \quad (2.19)$

Uma seqüência pseudo-aleatória pode ser gerada usando um registrador de deslocamento, visualizado na Figura 2.3 (com 4 estágios). Como o registrador opera apenas valores com valores binários (“1” e “0”), existe uma relação entre os valores binários e os valores da seqüência pseudo-aleatória, definido em (2.20).

$$\begin{cases} \text{Saída} = 1 \therefore c(n) = 1 \\ \text{Saída} = 0 \therefore c(n) = -1 \end{cases} \quad (2.20)$$

O registrador é iniciado com um valor diferente de zero. No nosso exemplo, a palavra binária $X_1X_2X_3X_4 \neq 0$. O comprimento máximo da seqüência pseudo-aleatória N (e conseqüentemente o número de deslocamentos do registrador) é definido em função do número de estágios do registrador (k), de acordo com (2.21).

$$N \leq 2^k - 1 \quad (2.21)$$

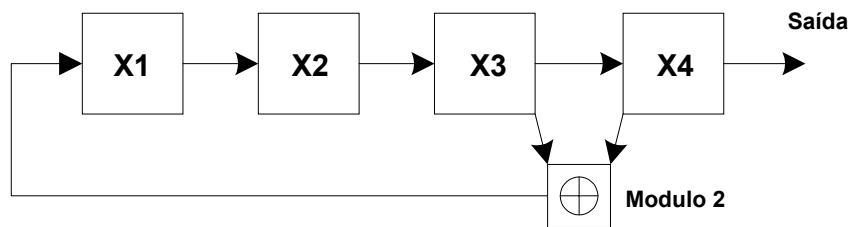


Figura 2.3 – Registrador de deslocamento para geração de seqüências pseudo-aleatórias, com $k=4$.

Se N é máximo, $c(n)$ é definida como seqüência de comprimento máximo. Se não, $c(n)$ é definida como seqüência de comprimento não-máximo. Vale dizer que a configuração mostrada na Figura 2.3 não é única: a saída pode ser em qualquer estágio do deslocador, bem como o operador módulo-2 pode ser utilizado para somar quaisquer estágios do

registrador. Para cada configuração, a sequência é diferente, e pode ser máxima ou não-máxima. Um exemplo de sequência é mostrado na Figura 2.4 , para $k=4$ e $X_1X_2X_3X_4=1000$. Neste caso, temos $N=15$, e por (2.21), a sequência gerada é máxima.

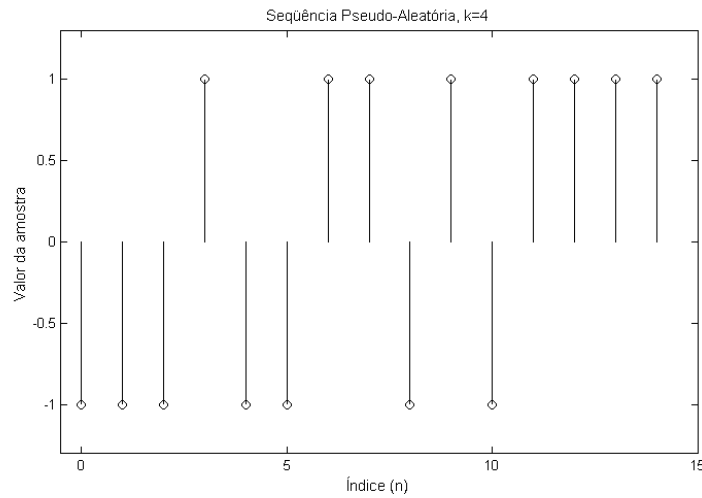


Figura 2.4 – Sequência pseudo-aleatória obtida com o registrador de deslocamento mostrado na Figura 2.3, com as seguintes condições iniciais: $X_1X_2X_3X_4=1000$.

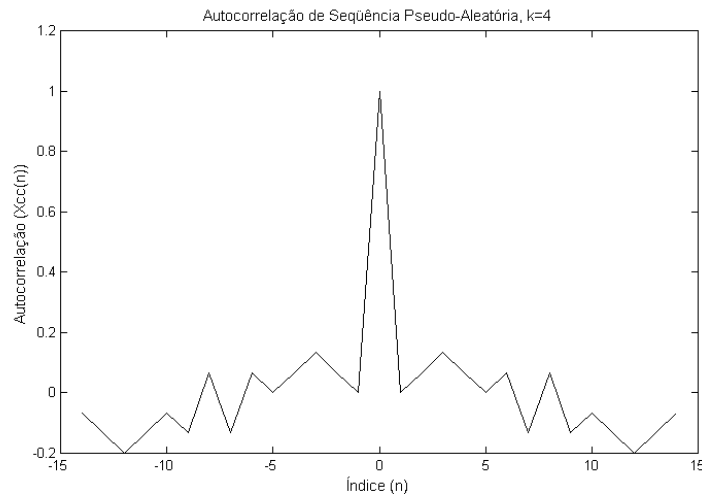


Figura 2.5 – Gráfico da função de auto-correlação ($R_c(n)$) da sequência pseudo-aleatória mostrada na Figura 2.4.

A função de auto-correlação de $c(n)$, $R_c(n)$, mostrada em (2.22) [20], é mostrada na Figura 2.5. A função de correlação cruzada, $R_{xy}(n)$, que está definida em [20], é mostrada em (2.23).

$$R_x(n) = \sum_{k=-(N-1)}^{N-1} x(k) \cdot x(n-k) \quad (2.22)$$

$$R_{xy}(n) = \sum_{k=-(N-1)}^{N-1} x(k) \cdot y(n-k) \quad (2.23)$$

2.5. Modulação DS-SS

2.5.1. Aspectos teóricos

Seja um sinal $s(n) = Eb$ a ser transmitido sobre um canal AWGN (*additive white gaussian noise*), ou seja, um canal que adiciona um ruído branco com média zero ao sinal transmitido. O sinal recebido, $r(n)$, pode ser escrito como [19]:

$$r(n) = E \cdot b + \omega(n) \quad (2.24)$$

aonde $E > 0$ é a energia do pulso associado a cada bit, $b = \pm 1$ é o bit que se quer transmitir e $\omega(n)$ é um ruído gaussiano branco com média zero. Para um sistema receptor que detecte apenas o nível do sinal, tem-se:

$$\begin{cases} r(n) > 0 : +1 \text{ foi enviado} \\ r(n) < 0 : -1 \text{ foi enviado} \end{cases} \quad (2.25)$$

Fica claro que se $\omega(n)$ for suficiente grande, o detector poderá avaliar a resposta erradamente. Considere agora que o sinal $s(n)$ seja transmitido sobre um canal AWGN, utilizando para isso uma seqüência pseudo-aleatória para “modular” o bit a ser transmitido. O sinal recebido pode ser descrito em (2.26):

$$r(n) = E \cdot b \cdot c(n) + \omega(n) \quad (2.26)$$

aonde $c(n)$ é uma seqüência pseudo-aleatória, também chamada de código pseudo-aleatório. Logo, o algoritmo de detecção, levando em conta a equação (2.26), é mostrado na Figura 2.6.

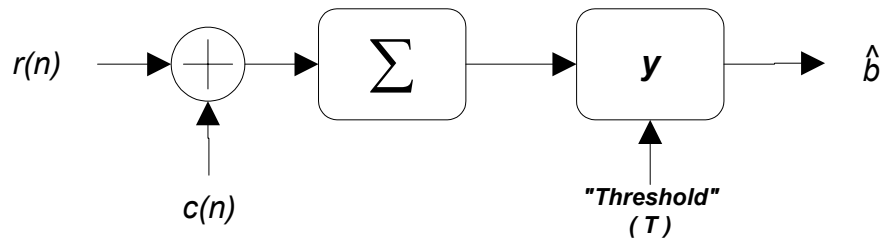


Figura 2.6 - Algoritmo de recepção DS-SS

O sinal recebido é multiplicado pela mesma seqüência pseudo-aleatória que foi utilizada para a transmissão. O valor de y é calculado em (2.27), de acordo com (2.19).

$$y = \sum_{k=0}^{N-1} c(k) \cdot r(k) = \sum_{k=0}^{N-1} E \cdot b \cdot c(k) \cdot c(k) + \sum_{k=0}^{N-1} c(k) \cdot \omega(k)$$

$$\therefore$$

$$y = N \cdot E \cdot b + \sum_{k=0}^{N-1} c(k) \cdot \omega(k) = E \cdot b \cdot R_{cc}(0) + R_{\omega c}(0)$$
(2.27)

O resíduo da equação (2.27), denotado por $R_{\omega c}$, é a correlação cruzada de $c(n)$ e $\omega(n)$, sendo numericamente bem menor que o valor da auto-correlação de $c(n)$, tornando o sistema robusto contra interferências. A saída y é comparada com um valor de threshold (T) para a decisão de detecção. O valor de threshold deve ser escolhido adequadamente, e levam em conta fatores como robustez na detecção, confiabilidade do sistema, dentre outros, e é uma fração do valor máximo da auto-correlação de $c(n)$, como mostrado na equação (2.28).

$$T = K_1 \cdot N \cdot E$$

$$0 \leq K_1 \leq 1$$
(2.28)

A detecção ocorre quando o valor de saída do detector (y) é maior ou igual ao threshold estabelecido. Se isto não ocorre, ou seja, se o valor de y for abaixo do threshold, considera-se que não houve a detecção de bit transmitido, como mostrado em (2.29).

$$y = R_{rc} \geq T : \text{bit detectado}$$

$$y = R_{rc} < T : \text{sem detecção de bit}$$
(2.29)

2.5.2. Modulação DS/BPSK (*Direct Sequence Binary Phase Shift Keying*)

Na prática, os sistemas SS de seqüência direta são implementados para garantir a segurança dos dados e a imunidade contra interferências, intencionais ou não. O sinal a ser transmitido é modulado com técnicas convencionais (BPSK, QAM, etc.) e em seguida é espalhado por modulação *spread spectrum*. O sinal recebido é então multiplicado pela cópia local do código pseudo-aleatório, e em seguida, utiliza-se técnicas convencionais de demodulação. Na modulação BPSK (*Binary Phase Shift Keying* – Codificação binária por

mudança de fase), a fase do sinal a ser transmitido é função do bit a ser transmitido, de acordo com a equação (2.30).

$$s(t) = \cos\left(2 \cdot \pi \cdot f_0 \cdot t + \frac{(b-1) \cdot \pi}{2}\right) \quad (2.30)$$

Maiores detalhes sobre a modulação DS/BPSK podem ser encontrados em [9]. O diagrama básico da modulação DS/BPSK pode ser visto na Figura 2.7. Um exemplo utilizando o sistema mostrado na Figura 2.7 é mostrado abaixo. Uma seqüência de dados ($b=11010010$) é modulado por BPSK, e em seguida é multiplicado por um código pseudo-aleatório (Figura 2.8), que foi gerado utilizando-se um registrador de deslocamento com oito estágios ($k=8$); o somador módulo-2 está conectado aos estágios X_4 e X_7 , a saída do registrador está conectada ao estágio X_1 , e a condição inicial dos estágios são $X_1X_2X_3X_4X_5X_6X_7X_8=10010101$. Foi utilizada nesta simulação uma relação sinal-ruído ($SNR=17$ dB), simulando a transmissão do sinal por um canal de comunicação AWGN. O sinal recebido é então multiplicado pelo código pseudo-aleatório, e então filtrado por um filtro digital passa-baixa Butterworth de 5ª ordem, $f_c=5f_0$, aonde f_c é a freqüência de corte do filtro. O sinal resultante é mostrado na Figura 2.9. Uma comparação entre o sinal transmitido e o sinal recebido é mostrado na Figura 2.10.

Como pode ser observado na Figura 2.10, o sinal é recuperado, e está pronto para ser demodulado. O atraso entre o sinal enviado e o sinal recebido é decorrência do processo de filtragem. Uma observação importante é que a cópia local do código pseudo-aleatório localizado no receptor deve estar sincronizada com o código pseudo-aleatório enviado pelo transmissor. Isso decorre em função da propriedade descrita em (2.19), pois se os sinais não estiverem sincronizados, a correlação entre eles será baixa, e o sinal recebido será perdido. No diagrama mostrado na Figura 2.7, o processo de sincronização ocorre pela estimação do atraso T_d e da fase denotada por ϕ . Portanto, em um sistema de recepção *spread spectrum* há dois algoritmos-chave: a sincronização, quando o sinal recebido é sincronizado com a cópia local do código pseudo-aleatório, e o *tracking*, que garante que a sincronização permaneça durante todo o processo de comunicação. Alguns algoritmos básicos de sincronismo e de *tracking* podem ser encontrados em [9], [10].

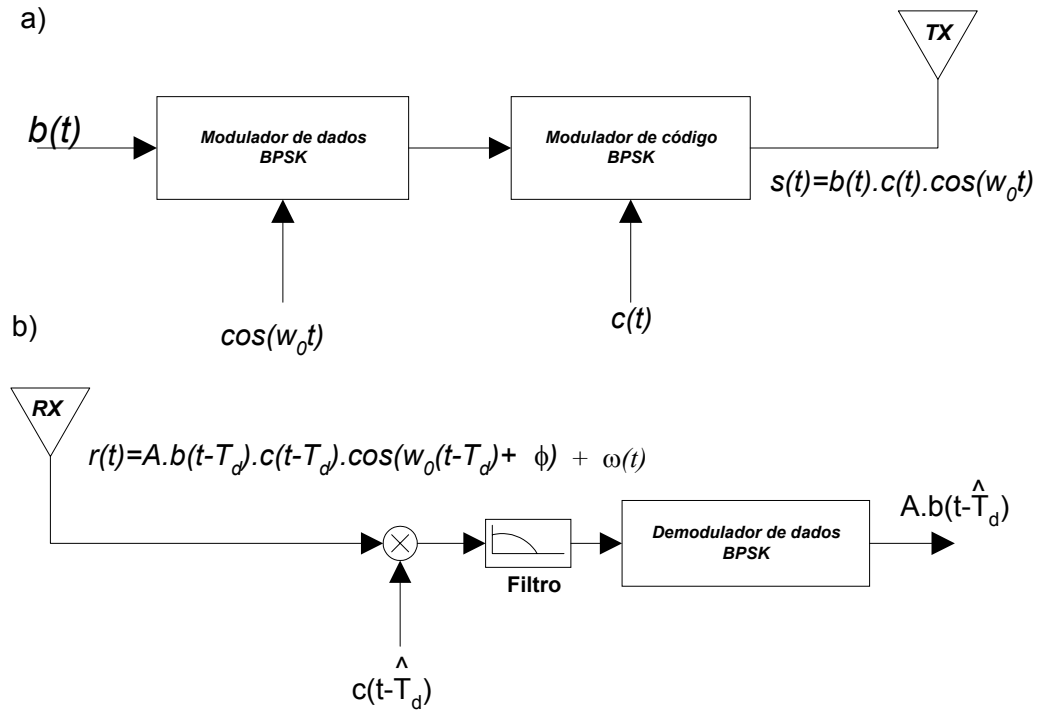


Figura 2.7 - Diagrama básico da modulação DS/BPSK: a) Concepção do transmissor, com modulação BPSK, b) Receptor, com filtro de entrada e demodulador BPSK.

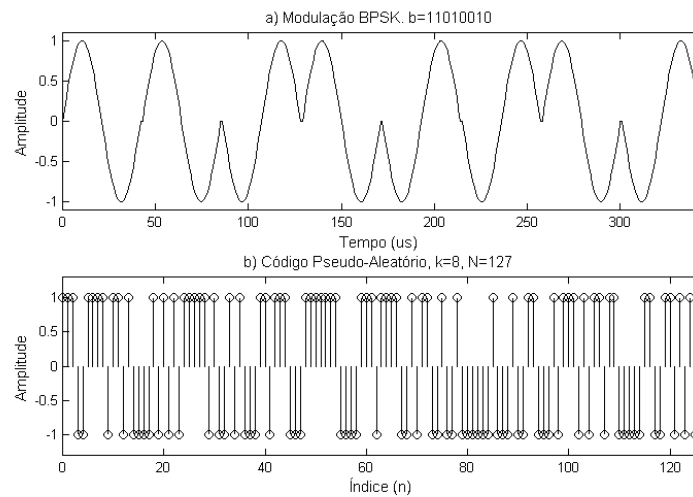


Figura 2.8 – a) Modulação BPSK de um *bitstream* ($b=11010010$) b) Código pseudo-aleatório com $k=8$, $N=127$.

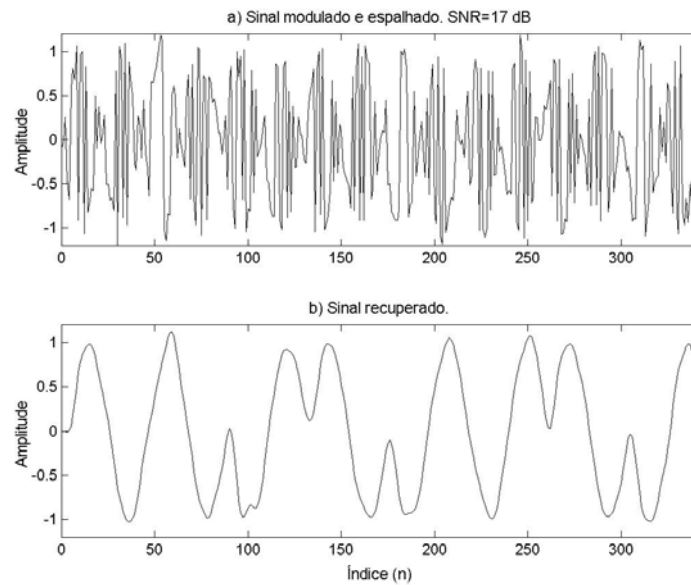


Figura 2.9 – a) Sinal transmitido (SNR=17dB) b) Sinal recuperado e filtrado

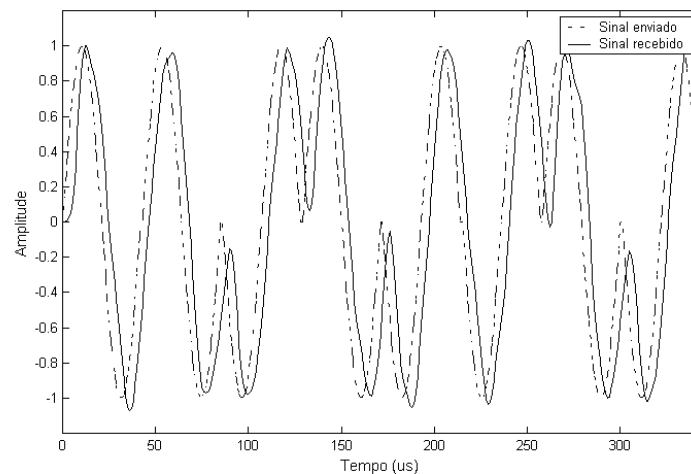


Figura 2.10 - Comparação entre o sinal enviado e o sinal recebido

2.6. Conclusões

Neste capítulo, foram apresentados os fundamentos dos sistemas *spread spectrum*. Mostrou-se que é possível, a partir de códigos pseudo-aleatórios, gerar sistemas de comunicação que apresentem imunidade ao ruído, bem como sejam protegidos de interferências intencionais ou não. Foram apresentados os fundamentos da geração de códigos pseudo-aleatórios e suas propriedades principais. Foram apresentadas simulações utilizando a técnica de DS-SS, mostrando a característica de espalhamento (*spreading*) e a

recuperação do sinal recebido (*despreading*). As conclusões obtidas a partir deste capítulo é que:

- a) O processo de espalhamento do sinal aumenta a confiabilidade do processo de comunicação como um todo;
- b) A geração das seqüências pseudo-aleatórias, sendo um processo com extremas variações, tanto de condições iniciais como de saídas possíveis, produz um código eficiente para a segurança dos dados transmitidos;

Desta forma, já é possível obter uma comparação dos métodos empregados em transmissão digital de sinais pela rede elétrica, que serão vistos no próximo capítulo.

3. Técnicas de Transmissão de Dados em Redes Elétricas

3.1. Introdução

A necessidade da comunicação de dados em redes elétricas tem gerado nas últimas décadas uma demanda no desenvolvimento de tecnologias capazes para tal feito. As tecnologias existentes podem ser agrupadas em dois grandes ramos: portadora única e portadora múltipla. Neste capítulo, será dada uma revisão das técnicas existentes mais difundidas, bem como suas características básicas.

3.2. Tecnologias Existentes

3.2.1.X-10

A tecnologia de transmissão de dados utilizando o protocolo X-10 é, dentre todas que aqui serão apresentadas, a mais antiga e a mais difundida, devido à simplicidade de operação na detecção dos sinais na rede elétrica. O protocolo de transmissão X-10 se baseia no ponto de passagem da onda fundamental de 60 Hz pelo zero para que o sistema de geração de dados possa disparar o gerador de símbolos que serão propagados na rede elétrica [33], [37]. No protocolo X-10, os símbolos são representados por “carries” de 120 KHz, sobrepostos à fundamental (60 Hz), como mostrado na Figura 3.1.

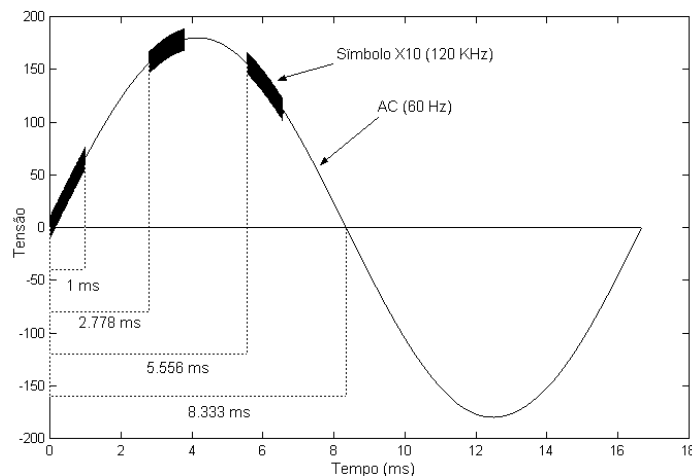


Figura 3.1 – Temporização no protocolo X-10.

O protocolo X-10 representa um bit na rede elétrica pela presença, por aproximadamente 1ms, do *carrier* de 120 KHz. O bit “1” é identificado com a presença do *carrier* na rede, e o bit “0” é identificado pela ausência do *carrier*. Uma sequência de Start é utilizada para indicar o início da transmissão, sendo composta por 3 bits “1” e um bit “0”. Para garantir que o sinal se propague nas três fases da rede elétrica, o pulso de 120 KHz é repetido atrasado de 60° elétricos, o que equivale a um atraso de 2.778 ms para o início da transmissão do próximo pulso (Figura 3.1). A Figura 3.2 mostra as três fases da rede elétrica, demarcando o início dos pulsos de 120 KHz no cruzamento por zero da rede elétrica.

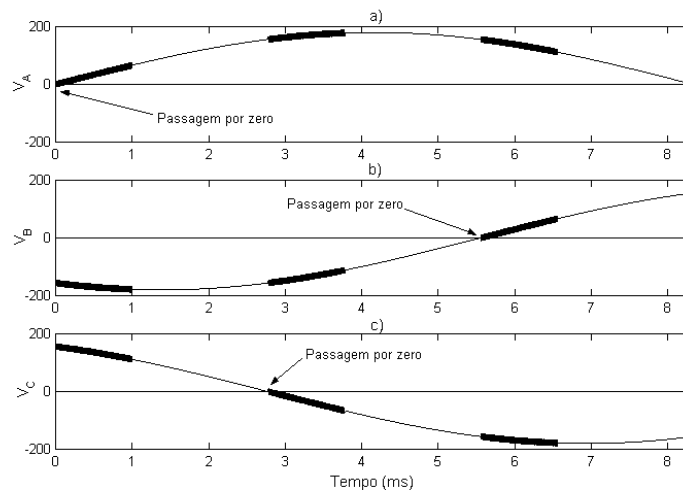


Figura 3.2 – Propagação dos símbolos nas três fases (V_A , V_B e V_C) no protocolo X-10.

Um grande limitador deste tipo de tecnologia é a baixa velocidade de transmissão de dados, que no protocolo X-10 corresponde a 120 bps, o que limita sua aplicação a sistemas com pouco fluxo de informação. Maiores detalhes sobre o protocolo de transmissão X-10 podem ser encontrados em [33].

3.2.2. LONworks

O padrão *LONworks* se baseia no uso de modulação por frequência única para a transmissão de pacotes pela rede elétrica, e utiliza a tecnologia de *Dual Carrier Frequency* (DCF) [43], que permite o “chaveamento” entre duas frequências, escolhidas de acordo com a menor atenuação provocada pela rede elétrica. Os protocolos que a tecnologia DCF utiliza são o CENELEC EN50065-1 e o EIA-709.2, que permitem a comunicação de dados a uma taxa de 5400 bps. As frequências utilizadas pela tecnologia *Dual Carrier Frequency* são definidas em função da banda do protocolo utilizado, como mostra a Figura 3.3 [42].

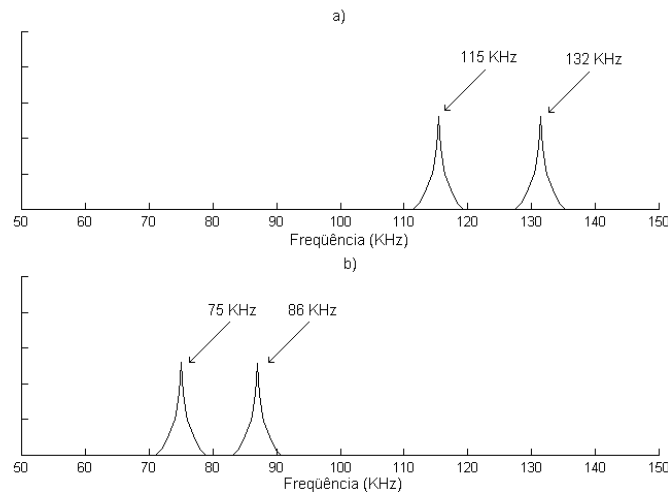


Figura 3.3 – Frequências utilizadas pela tecnologia DCF. a) Frequências utilizadas pela banda C; b) frequências utilizadas pela banda A.

Na banda A do protocolo CENELET, as frequências utilizadas são de 75 KHz e 86 KHz, enquanto que na banda C são de 115 KHz e 132 KHz.

O processo de transmissão se baseia na transmissão de um pacote inicial (START) com a frequência de 86 KHz (Banda A) ou 132 KHz (Banda C). Se não há resposta do nó receptor, então o transmissor muda a frequência da portadora para 75 KHz (Banda A) ou 115 KHz (Banda C). Maiores detalhes sobre o padrão *LONworks* podem ser encontrados em [34] e [43].

3.2.3.CEBus

O protocolo *CEBus* (*Consumer Electronics Bus*), também conhecido pela sigla EIA-600, foi desenvolvido para a comunicação de produtos eletro-eletrônicos em redes elétricas residências e comerciais, e utiliza a tecnologia *Spread Spectrum Carrier* (SSC), baseada na transmissão de símbolos SSC (*chirps*) com frequência variável (*spread spectrum*). A tecnologia SSC confere uma maior imunidade ao ruído e à distorção da rede elétrica ao se efetuar a transmissão dos pacotes na rede elétrica, comparada a tecnologias com frequência única. O espalhamento dos símbolos é de 100 KHz a 400 KHz, e cada símbolo tem duração de 100 μ s, o que fornece uma taxa de transmissão de 10.000 bauds. A Figura 3.4 mostra como o sinal é espalhado pela faixa de frequência especificada (100 KHz a 400 KHz). Maiores detalhes sobre o protocolo *CEBus* e das características técnicas da modulação SSC são mostrados nos próximos capítulos.

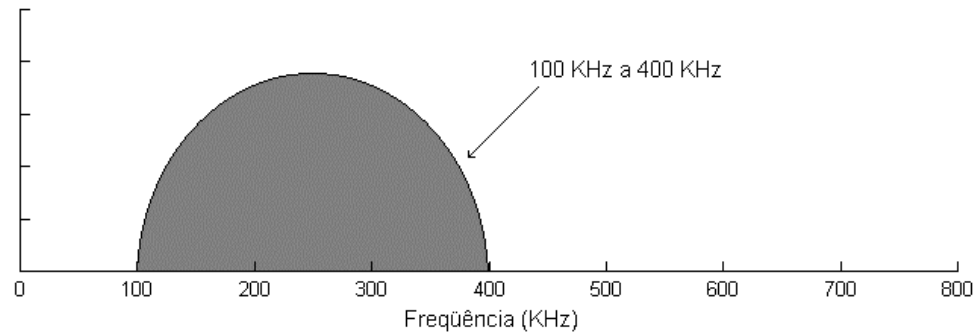


Figura 3.4 – Gráfico do espalhamento do sinal SSC.

3.3. Comparação entre as tecnologias

A Tabela 3.1 mostra o quadro comparativo entre as tecnologias existentes citadas na seção anterior, com as vantagens e desvantagens de cada uma delas.

Tabela 3.1 – Comparação entre X-10, *CEBus* e LONworks

Características	X-10	<i>CEBus</i>	LONworks
Simplicidade	☺	☹	☹
Custo de implantação	☺	☹	☹
Controle de Acesso	☹	☺	☺
Imunidade a interferências	☹	☺	☺
Difusão no mercado	☺	☺	☹
Taxa de comunicação	☹	☺	☺

Embora não seja possível dar um veredicto sobre qual o melhor sistema a ser implantado, é necessário fazer algumas observações a respeito de cada um deles [35]:

- X-10: barato, disponível e fácil de instalar, oferece um rico universo de produtos compatíveis. Os principais contra-pontos são a baixa velocidade de transmissão e, por não prover um controle de acesso eficiente, é considerado não-confiável por muitos especialistas.
- O protocolo *CEBus* é um padrão industrial para a transmissão de sinais de controle pelas redes elétricas. O padrão *CEBus* utiliza a tecnologia Spread Spectrum que permite a imunidade a ruídos e a outras fontes de interferências

em redes elétricas. Dispositivos *CEBus*, como sistemas de controle de iluminação, são altamente confiáveis, mas permanecem caros comparados aos dispositivos “comuns”.

- c) O padrão LONworks é outro padrão de baixa tensão que parece estar ganhando terreno principalmente em sistemas comerciais e mercados multi-usuários (como por exemplo, apartamentos, hotéis, centros de convenção, etc.).

3.4. Conclusões

Neste capítulo foram examinadas as principais tecnologias existentes para possibilitar a comunicação de dados em redes elétricas. Foram levantados os aspectos teóricos e práticos das tecnologias, ressaltando os pontos a favor e contra. Neste trabalho, decidiu-se pelo estudo e projeto de dispositivos baseados no protocolo *CEBus*, em razão das seguintes características:

- a) oferecer uma tecnologia mais confiável e robusta;
- b) ter a tendência de se tornar um padrão para a comunicação de dados em redes elétricas *intrabuilding*, através da padronização do protocolo *CEBus*;
- c) ter uma taxa de comunicação de dados que atende as necessidades de automação residencial e industrial.

4. Transmissão de Dados em Redes de Energia Elétrica Utilizando Técnicas de *Spread Spectrum Carrier*

4.1. Introdução

No capítulo anterior foram revisadas várias técnicas de comunicação de dados que utilizam as redes elétricas como meio de transmissão. Este capítulo se dedica a uma técnica chamada *Spread Spectrum Carrier* [4], abordando as questões técnicas relativas à implementação destes sistemas. O protocolo de comunicação EIA-600 (*CEBus*) também será abordado.

4.2. Concepção Básica

A idéia básica da técnica *Spread Spectrum Carrier* é utilizar uma forma de onda conhecida que tem a capacidade de representar um determinado bit a ser transmitido. Esta forma de onda, também chamada de símbolo ou UST (*Unit Symbol Time* – Símbolo Unitário) é a menor unidade de informação do sistema de comunicação. A combinação destes símbolos em uma seqüência lógica e pré-definida define os parâmetros necessários para que dois ou mais dispositivos em um nó da rede elétrica possam ser capazes de trocar informações entre si. Esta combinação é definida basicamente por dois parâmetros:

- Codificação do bit: como combinar e rearranjar uma ou mais UST's para representar um determinado bit. Existem várias técnicas para viabilizar [9] este rearranjo, como por exemplo, ASK, PSK, PRK, DPSK, etc.
- Codificação dos bytes: a organização dos bytes em estruturas lógicas, que formem um conjunto de instruções compatíveis que todos os dispositivos em um nó da rede possam ler e interpretar corretamente. Basicamente, é o protocolo de comunicação que define como os bytes se agrupam nas estruturas lógicas, como endereços, máscaras, etc.

O protocolo EIA – 600, também chamado de ***CEBus*** (*Consumer Electronics Bus*), define a codificação dos bytes nos sistemas *Spread Spectrum Carrier*.

4.3. Estrutura Básica do Protocolo *CEBus*

O protocolo *CEBus* descreve um método de comunicação entre dispositivos usando seis diferentes tipos de meio: *PowerLine*, Par Trançado, Cabo Coaxial, RF, Infravermelho e Fibra Ótica [1]. O protocolo *CEBus* possui varias características, a seguir:

4.3.1. Implementação da comunicação em pacotes de dados

O protocolo implementa as várias estruturas lógicas pré-definidas em um pacote de dados, que é transmitido para todos os nós da rede. O pacote de dados tem tamanho variável, limitado em 41 bytes. Na estrutura do pacote, estão definidos campos lógicos, mostrado na Figura 4.1. Para ilustrar como estes campos lógicos são utilizados, pode-se utilizar uma analogia entre o pacote *CEBus* e uma carta enviada por um serviço postal, mostrada na Figura 4.2.

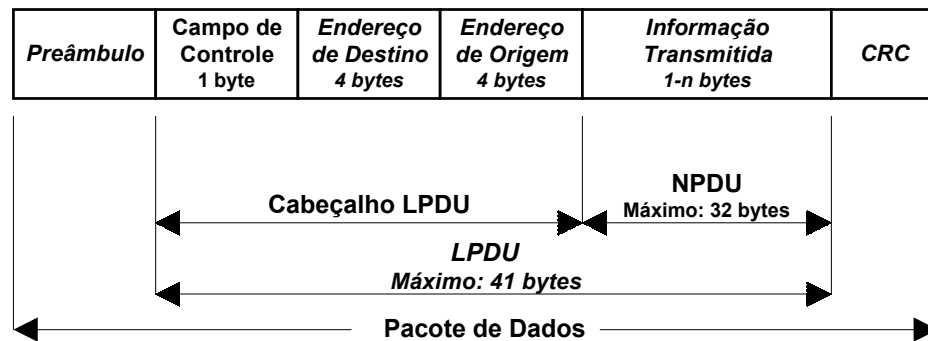


Figura 4.1 – Implementação das estruturas de dados em um pacote *CEBus*. Esta estrutura de pacote só é válida para os meios PowerLine e RF [1].

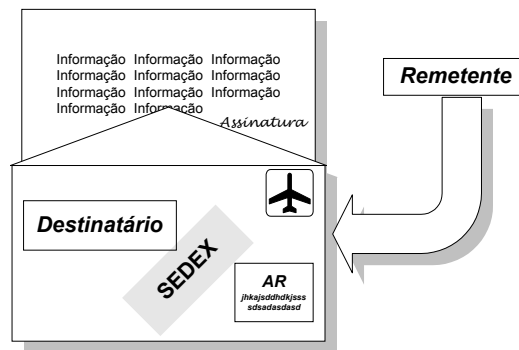


Figura 4.2 – Exemplo da estrutura do pacote como uma carta mandada por um serviço postal.

As estruturas mostradas na Figura 4.1 são descritas a seguir. Mais detalhes podem ser encontrados em [1].

4.3.1.1. Campo de Controle

Componente do *LPDU Header* (*Link Protocol Data Unit*). Especifica a classe de serviço do pacote (Básico ou Avançado), a prioridade (Alta ou Padrão) e o tipo de pacote a

ser enviado (lack, Ack_Data, Unack_Data, Failure, Addr_Ack_Data, Addr_lack, Addr_Unack_Data). A estrutura é mostrada na Figura 4.3. Fazendo-se uma analogia com a carta, o Campo de Controle representa se a carta será mandada por SEDEX ou envio normal (Prioridade e Classe de Serviço) e se será mandada ou não com AR – Aviso de recebimento – (Tipo de pacote).

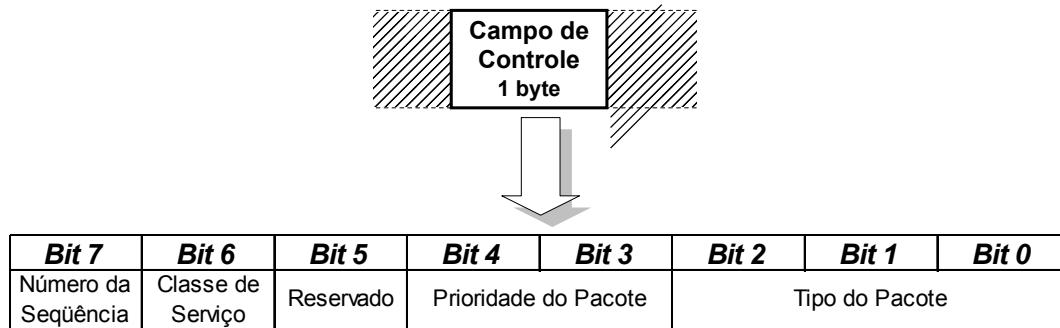


Figura 4.3 – Descrição bit a bit do Campo de Controle

4.3.1.2. Endereços de Destino e de Envio

Componente do *LPDU Header (Link Protocol Data Unit)*. O protocolo **CEBus** gerencia o endereçamento de cada dispositivo conectado na rede através de uma palavra binária de 32 bits, o que confere uma capacidade da ordem de 10^9 endereços, mostrado em (4.1). Tanto a estrutura Endereço de Destino como a estrutura Endereço de Origem tem o mesmo formato, mostrado na Figura 4.4.

$$C = 2^n = 2^{32} = 4.2949 \cdot 10^9 \quad (4.1)$$

Cada dispositivo tem um endereço programável, e é subdividido em duas partes lógicas: Endereço de Sistema (SA – *System Address*) e Endereço de Controle de Acesso ao Dispositivo (MAC – *Media Access Control*), também conhecido como Endereço de Dispositivo (DA – *Device Address*). Cada parte lógica do endereço (SA e DA) é formada por uma palavra binária de 16 bits, subdivididas em dois bytes: o mais significativo e o menos significativo (MSB e LSB, respectivamente). Os bytes dos campos de endereço são transmitidos em ordem reversa, com um marcador de campo (EOF) marcando o final do campo de endereço, evitando que se transmitam zeros desnecessários em endereços com menos de 32 bits. Mais informações sobre marcadores de campo (ou *tokens*) podem ser encontradas em [4].

Em geral, os dispositivos que formam um grupo bem definido de nós têm o mesmo endereço de sistema (SA), facilitando a organização lógica dos dispositivos. Por exemplo, os

dispositivos situados em uma residência “A”, com endereço de sistema $SA=XA_h$, não irão interferir com os dispositivos situados na residência vizinha “B”, que tem o endereço de sistema $SA=XB_h$, com $XA \neq Xb$. Por outro lado, dispositivos com endereços de sistema (SA) diferentes podem ter, sem problema, endereços de dispositivos (DA) iguais, pois fazem parte de redes lógicas diferentes.

Cada dispositivo situado na rede só irá receber os dados apropriadamente quando o endereço de destino do nó transmissor (SA e DA) for igual ao endereço de origem do nó receptor (SA e DA), ou ainda, quando o endereço de destino do nó transmissor tiver valor igual a zero ($DA=00_h$ e $SA=00_h$), configurando desta forma um endereçamento de *broadcasting* (transmissão para que todos os dispositivos tenham acesso aos dados).

Na analogia com a carta, as estruturas de endereçamento seriam os endereços de destinatário e remetente da carta.

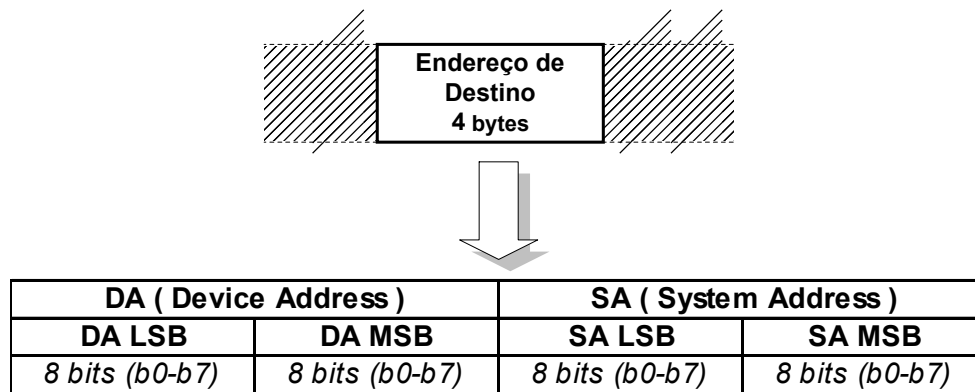


Figura 4.4 – Estrutura de endereçamento no pacote *CEBus*, válidas para o endereço de origem e para o endereço de destino.

4.3.1.3. Informações transmitidas

Componente do NPDU (*Network Protocol Data Unit*). O protocolo *CEBus* permite que dois dispositivos situados na rede se comuniquem de duas formas distintas:

- Transmitindo uma string genérica (por exemplo, uma parte de um texto em ASCII ou uma sequência de dados coletados por um PLC). Neste caso, a interpretação dos dados que chegam ao receptor é intimamente dependente da função de cada dispositivo, ou seja, é extremamente dependente das características técnicas das aplicações gerenciadas pelos dispositivos *CEBus*. A transmissão de uma string genérica é mostrada na Figura 4.5.
- Transmitindo uma série de constantes e funções pré-definidas pelo protocolo *CEBus*. Essas funções fazem parte da linguagem **CAL** (*CEBus Application*

Language – Linguagem de Aplicação do Protocolo *CEBus*), um conjunto de instruções e estruturas criadas para habilitar a operação comum entre produtos usados em automação residencial. A razão para a implementação da linguagem CAL é permitir a interoperabilidade entre produtos de diferentes fabricantes sem que seja necessário o conhecimento de todas as características técnicas de cada produto [1]. Por exemplo, uma lâmpada de um determinado fabricante, controlada por uma interface *CEBus*, pode ter a sua luminosidade controlada, sem que seja necessário saber que o controle da corrente se dá por um amplificador da classe “B”. Cada dispositivo *CEBus* é controlado através de IV’s (*Instant Variable*), que são funções com variáveis que podem ser programadas através da linguagem CAL. Maiores informações podem ser encontradas em [1]. A configuração da linguagem CAL no pacote *CEBus* é mostrado na Figura 4.6, aonde os campos *NPDU Header* e *APDU Header* são os campos de configuração das chamadas CAL.

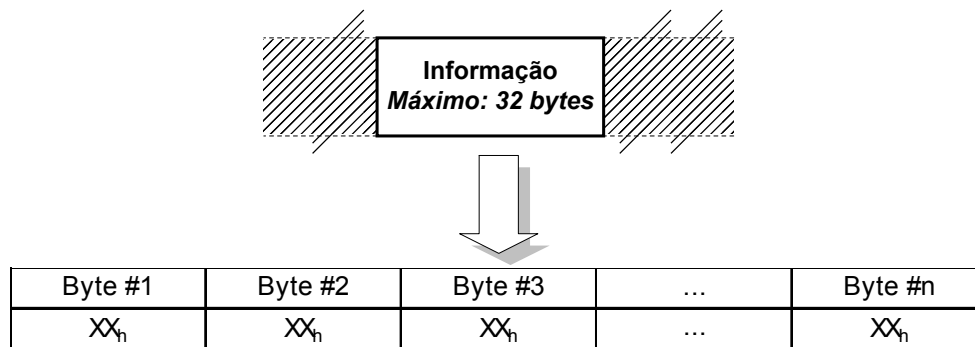


Figura 4.5 – Transmissão de uma *string* de dados genérica

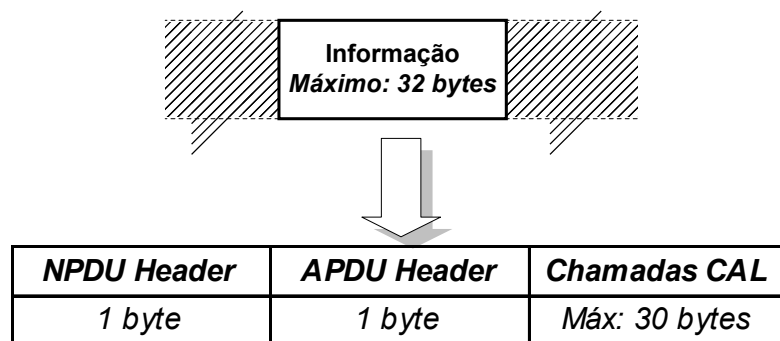


Figura 4.6 – Transmissão dos parâmetros da linguagem CAL

Utilizando a analogia da carta, o campo de informações corresponde ao conteúdo da carta enviada. No caso em que se transmite um pacote com chamadas CAL, o conteúdo da carta pode ser entendido como sendo um formulário a ser preenchido, onde cada campo do formulário é uma IV a ser programada.

4.3.2. CRC (*Cyclical Redundancy Check*)

Até mesmo se as metodologias de comunicação de dados são bem projetadas, erros ainda podem acontecer durante a transmissão de dados. Então, algoritmos que detectam o erro são muito importantes em qualquer protocolo de transmissão, pois mantêm a integridade de dados e a confiabilidade do sistema de transmissão [21]. Um algoritmo bastante usado é o CRC (*Cyclical Redundancy Check*), que adiciona ao final do pacote uma série de bits, baseado em algum polinômio gerador. O algoritmo usado pelo protocolo *CEBus* é um algoritmo cíclico de 16 bits (CRC–16), baseado em um polinômio gerador G , de ordem 17, mostrado em (4.2). O algoritmo de CRC trata o blocos de bits como coeficientes de um polinômio, ou seja, se $G=x^{16}+x^5+x^2+1$, a seqüência de bits equivalente é, de acordo com (4.2), $G_b=10000000000100101_b$.

$$G = \sum_{i=0}^{17} b_i \cdot x^i \quad (4.2)$$

O cálculo do CRC–16 trata os dados a serem transmitidos como um polinômio, P , que é dividido pelo polinômio gerador G , como mostrado em (4.3). O polinômio Q é o quociente da divisão de P por G , e R é o resto da divisão.

$$P = Q \cdot G + R \quad \therefore \quad \frac{P}{G} = Q + \frac{R}{G} \quad (4.3)$$

Após o cálculo da divisão de P por G , o resto R é adicionado ao final dos bits a serem transmitidos. No receptor, os dados recebidos são separados, e a operação é novamente repetida. Se o resto transmitido for igual ao resto calculado localmente, a probabilidade que os dados foram corretamente transmitidos é 99,99%. Caso os restos forem diferentes, presume-se que houve um erro de transmissão, e o pacote deverá ser retransmitido. O diagrama da Figura 4.7 explica o processo de detecção de erro CRC – 16. O algoritmo de CRC é redundante, pois adiciona uma seqüência de dados que não contém nenhuma informação. O polinômio usado afeta o resultado do resto, logo, os polinômios geradores G são iguais, tanto no receptor quanto no transmissor.

Para o cálculo do resto, algoritmos recursivos são largamente utilizados, uma vez que têm baixo índice de complexidade computacional. Os algoritmos para divisão de polinômios da forma mostrada em (4.2) envolvem o uso de registradores de deslocamento. Maiores detalhes podem ser encontrados em [21] e [22]. O protocolo *CEBus* calcula e incorpora os bits do CRC no final do LPDU (Figura 4.1).

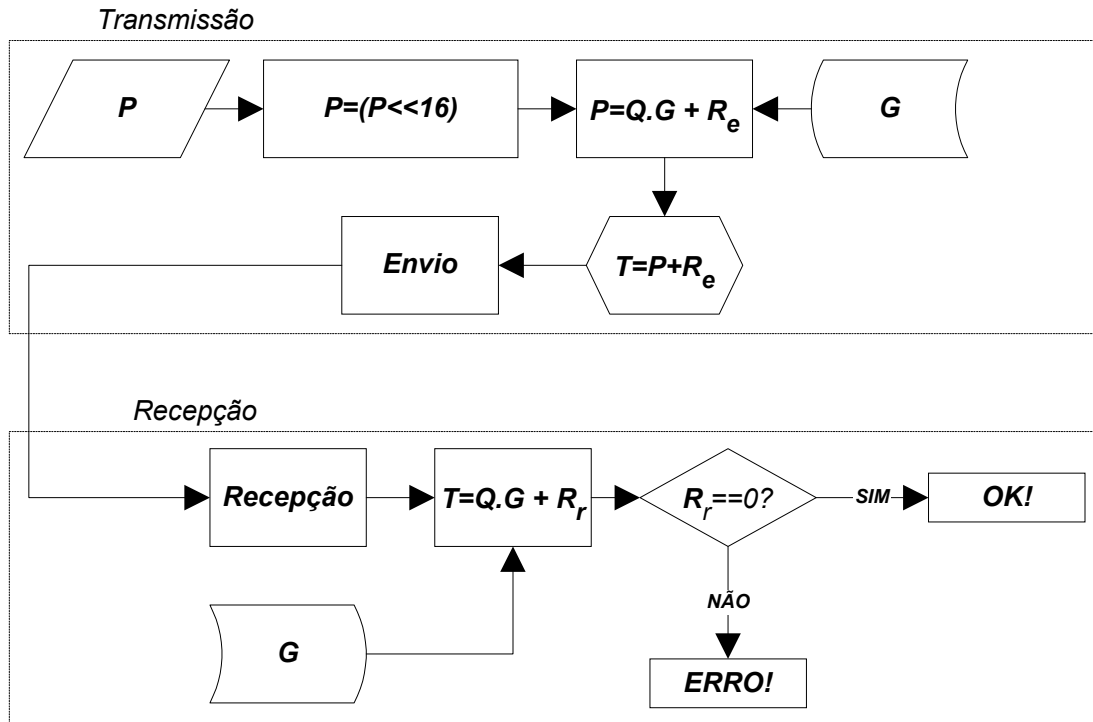


Figura 4.7 – Algoritmo de detecção de erro CRC – 16. O resto da divisão, R_e , é calculado e incluído ao final da mensagem transmitida. No receptor, é novamente calculado o resto, R_r , avaliando possíveis erros de transmissão.

4.3.3. CSMA/CD (Carrier Sense Multiple Access / Collision Detection)

O protocolo *CEBus* implementa o múltiplo acesso através de CSMA (*Carrier Sense Multiple Access* – Múltiplo Acesso por Detecção de Carrier). Isso dá à rede a capacidade de evitar colisão de dados quando dois dispositivos tentam acessar a rede ao mesmo tempo. Basicamente, o processo de CSMA opera da seguinte forma: o dispositivo *CEBus* que deseja transmitir aguarda até que a linha não esteja sendo usada. Uma vez liberada a linha, o dispositivo *CEBus* começa a transmitir um pacote de dados denominado preâmbulo, formado por uma seqüência de dados aleatórios. A transmissão do preâmbulo é vista pelos outros dispositivos *CEBus* conectados à rede elétrica, e se qualquer deles desejar transmitir, é necessário aguardar até que a linha esteja liberada. Se dois nós transmitem ao mesmo tempo, os dispositivos conectados à linha, através de um *transceiver*, detectam que o padrão transmitido é diferente do padrão recebido, indicando que houve uma colisão. A habilidade de um dispositivo de detectar colisões é chamada de *Collision Detection* (Detecção de Colisão). Depois de detectada a colisão, o dispositivo *CEBus* cessa a transmissão do preâmbulo, e deve aguardar um período de tempo até que possa retransmitir, minimizando a chance de outra colisão [24]. O dispositivo irá tentar retransmitir

um número pré-determinado de vezes (programável), de acordo com o volume de dados que trafegam na linha [24]. Se o dispositivo não obtiver sucesso na transmissão, será gerada uma mensagem de erro.

4.4. Codificação dos bits no protocolo *CEBus*

4.4.1. *Unit Symbol Time (UST)*

Como dito anteriormente, a combinação de várias *UST*'s em uma sequência lógica e pré-definida define os parâmetros necessários para que dois ou mais dispositivos em um nó da rede elétrica possam ser capazes de trocar informações entre si. A codificação dos bytes é efetuada pelo protocolo, enquanto que a codificação (ou rearranjo) dos bits é efetuada pela sinalização do protocolo. Por sinalização compreende-se o ato de combinar várias *UST*'s para codificar um bit. No caso da modulação *Spread Spectrum Carrier*, cada *UST* (ou símbolo) é formado por uma forma de onda denominada *chirp* [25], que é definida em termos de uma função seno com frequência angular variável, crescente ou decrescente. Um *chirp* pode ser logarítmico ou polinomial, e se for polinomial de 1º grau, é definido como linear. Um *chirp* polinomial é definido em (4.4), e um *chirp* logarítmico é definido em (4.5).

$$y_p = \sin \left(2 \cdot \pi \cdot \left(\frac{\beta}{p+1} \cdot t^{1+p} + f_0 \cdot t \right) + \phi_0 \right) \quad (4.4)$$

$$\beta = \frac{f_1 - f_0}{t_1^p}$$

$$y_l = \sin \left(2 \cdot \pi \cdot \left(\frac{10^{(\beta \cdot t)} - 1}{\beta \cdot \ln(10)} + f_0 \cdot t \right) + \phi_0 \right) \quad (4.5)$$

$$\beta = \frac{\log_{10}(f_1 - f_0)}{t_1}$$

aonde:

f_1 : Frequência de cruzamento em t_1 .

f_0 : Frequência inicial do *chirp*.

p : Ordem do *chirp*.

t_1 : Momento do cruzamento da frequência f_1 .

ϕ_0 : Fase inicial.

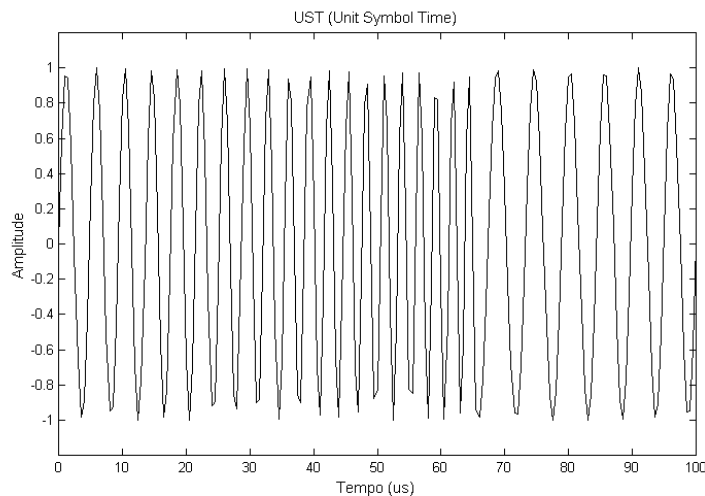
Na modulação *Spread Spectrum Carrier*, a UST é uma combinação de dois *chirp*'s lineares, com duração total de 100 μ s. O primeiro *chirp*, y_{p1} , tem frequência inicial de 200 KHz e frequência final de 400 KHz em 65 μ s, e o segundo *chirp*, y_{p2} , tem frequência inicial de 100 KHz e frequência final de 200 KHz em 100 μ s. O fato da frequência inicial de y_{p1} ser igual à frequência final de y_{p2} permite uma transição suave entre os símbolos. A Tabela 4.1 fornece os parâmetros de cada *chirp* para a composição da UST.

Tabela 4.1 – Parâmetros do *Chirp*

<i>Chirp</i> 1 (y_{p1})	<i>Chirp</i> 2 (y_{p2})
$p = 1$ $f_0 = 200\text{KHz}$ $f_1 = 400\text{KHz}$ $t_1 = 65\mu\text{s}$ $\phi_0 = 0^\circ$	$p = 1$ $f_0 = 100\text{KHz}$ $f_1 = 200\text{KHz}$ $t_1 = 100\mu\text{s}$ $\phi_0 = 0^\circ$

A UST é definida em (4.6) e é mostrada na Figura 4.8. A Figura 4.9 mostra o mapeamento do *chirp* em um gráfico de tempo x frequência, obtido por STFT (*Short Time Fourier Transform*), com janela de Hanning de 32 pontos, para verificar a variação de frequência do *chirp* em função do tempo [26].

$$UST = y = \begin{cases} yp_1, & 0 \leq t \leq 65\mu\text{s} \\ yp_2, & 65\mu < t \leq 100\mu\text{s} \end{cases} \quad (4.6)$$

Figura 4.8 – UST do protocolo CEBus (*Chirp*)

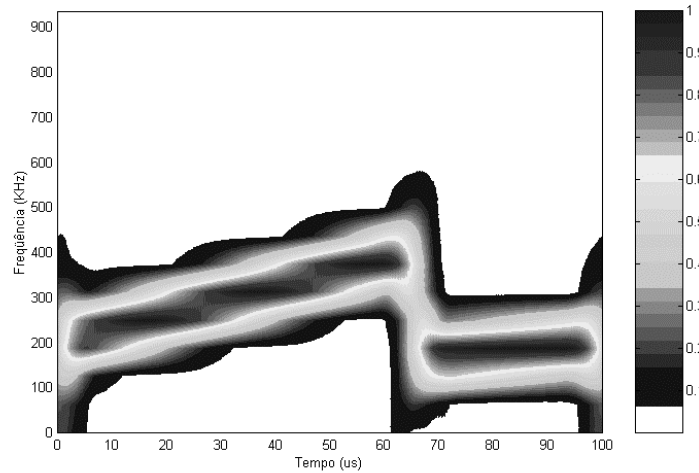


Figura 4.9 – Gráfico da STFT do *chirp* da Figura 4.8.

A FFT (*Fast Fourier Transform*) do *chirp* é mostrada na Figura 4.10, mostrando a característica de espalhamento do espectro na banda definida pela composição do *chirp*, de 100 KHz a 400 KHz.

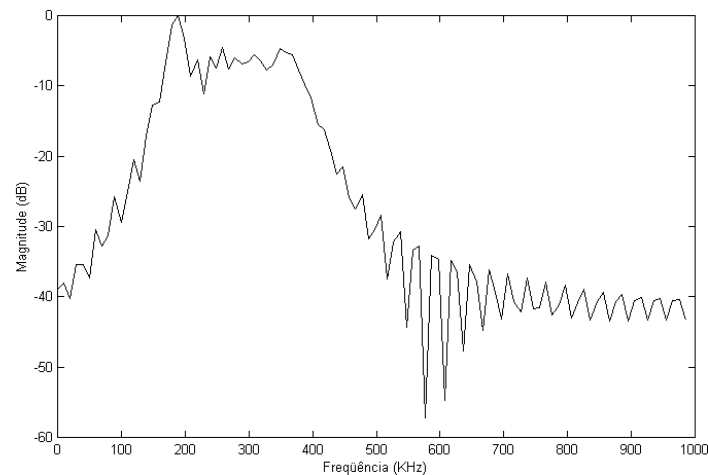


Figura 4.10 – Fast Fourier Transform do *chirp* ($F_{\text{amostragem}} = 2 \text{ MHz}$)

4.4.2. Codificação de Símbolos no Protocolo *CEBus*

Uma vez definido o menor símbolo do protocolo *CEBus*, outros símbolos são obtidos a partir de combinações do *UST*. Cada símbolo tem uma duração pré-determinada, e depende se o símbolo pertence ao preâmbulo ou ao corpo do pacote *CEBus*, como mostrado na Tabela 4.2.

Tabela 4.2 – Codificação dos símbolos no protocolo *CEBus*

Símbolo	Preâmbulo		Corpo do Pacote	
	UST's	Duração	UST's	Duração
"1"	1	114 μ s	1	100 μ s
"0"	2	228 μ s	2	200 μ s
EOF	8	800 μ s	3	300 μ s
EOP	N/A	N/A	4	400 μ s

O símbolo EOF (*End of Frame* – Final de Frame) serve a dois propósitos: marcar o final do preâmbulo e marcar o final de cada campo de endereço do pacote *CEBus*, otimizando o envio de endereços menores que 32 bits, pois não é preciso transmitir os zeros não significativos dos campos de endereço.

O símbolo EOP (*End of Packet* – Final de Pacote) serve para marcar o final do pacote transmitido e o início dos bits para a checagem de CRC.

4.4.3. Função e Codificação do Preâmbulo no Protocolo *CEBus*

A função do preâmbulo é prover o dispositivo *CEBus* da capacidade de múltiplo acesso pr CSMA. O preâmbulo é formado por uma seqüência pseudo-aleatória de 8 bits, seguido por um símbolo de EOF. Para codificar o preâmbulo, é usada a modulação ASK (*Amplitude Shift Keying* – Codificação por Mudança de Amplitude), que utiliza dois estados para a codificação dos símbolos, chamados de estado SUPERIOR e estado INFERIOR, ou S_1 e S_2 , respectivamente. O estado S_1 representa a presença do *chirp*, e o estado S_2 representa a sua ausência. Pelo fato do transmissor não enviar nenhum sinal no estado S_2 , os estados S_1 enviados por outros dispositivos que querem acessar a linha podem ser detectados durante o preâmbulo do pacote *CEBus*. Isso habilita aos dispositivos *CEBus* que queiram acessar a linha a capacidade de múltiplo acesso por CSMA.

Um exemplo pode ser visto na Figura 4.11. O preâmbulo, no caso a palavra binária 10010010, gerada por uma seqüência pseudo-aleatória, é transmitido pelo dispositivo *CEBus*. Notar que os símbolos transmitidos no preâmbulo têm 100 μ s de duração, seguidos por 14 μ s de ausência de sinal. Quando o preâmbulo é codificado, o primeiro símbolo, seja "1" ou "0", é codificado como sendo um estado SUPERIOR (S_1). Após a transmissão do preâmbulo, o símbolo de EOF (PRE_EOF) é codificado com oito estados S_1 , sem os 14 μ s de ausência de sinal. Uma vez transmitido o preâmbulo, e não havendo nenhuma detecção

de colisão (o que significa que o acesso ao canal está livre), o dispositivo *CEBus* está pronto para codificar e enviar o corpo do pacote. A Figura 4.12 ilustra a codificação do preâmbulo em função dos estados S_1 e S_2 .

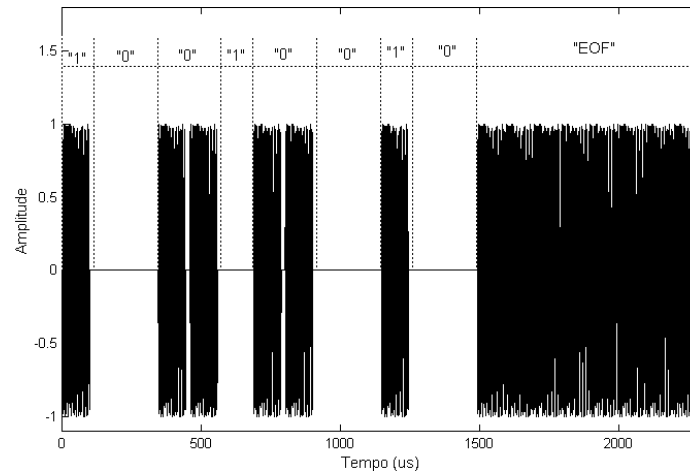


Figura 4.11 – Transmissão dos símbolos do preâmbulo (“10010010”) e EOF. Notar o período de 14 μ s de ausência de sinal após o *chirp* ser transmitido.

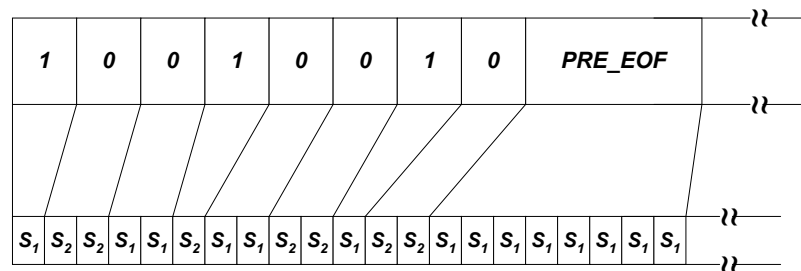


Figura 4.12 – Codificação do preâmbulo em função dos estados SUPERIOR (S_1) e INFERIOR (S_2). Notar o símbolo de EOF ao final do preâmbulo.

4.4.4. Codificação do Corpo do Pacote no Protocolo *CEBus*

No corpo do pacote *CEBus*, um segundo tipo de modulação é utilizado para codificar os bits a serem transmitidos: PRK (*Phase Reversal Keying* – Codificação por Fase Reversa), que utiliza novamente dois estados para codificar os dados. O estado SUPERIOR (S_1) representa a presença do *chirp* na linha, e o estado INFERIOR (S_2) representa a presença do *chirp* na linha, só que defasado de 180°. Esta técnica, mais robusta que a

modulação ASK, permite que a linha só seja liberada após o pacote ser totalmente transmitido, pois não há ausência de sinal durante a transmissão. Após o EOF do preâmbulo ser transmitido, o primeiro *chirp* do corpo do pacote é codificado como estado INFERIOR (S_2), oposto ao último estado do preâmbulo (S_1).

O final da transmissão do pacote é marcado com o símbolo de EOP, que é codificado com 4 estados opostos ao último estado do pacote. Se o último estado do pacote for S_2 , o símbolo de EOP é codificado com 4 estados S_1 . A Figura 4.13a mostra a codificação PRK sendo transmitida, com o detalhe da mudança de estado na Figura 4.13b. A Figura 4.14 mostra a codificação do início do corpo do pacote, enquanto que a Figura 4.15 mostra a codificação do final do corpo do pacote.

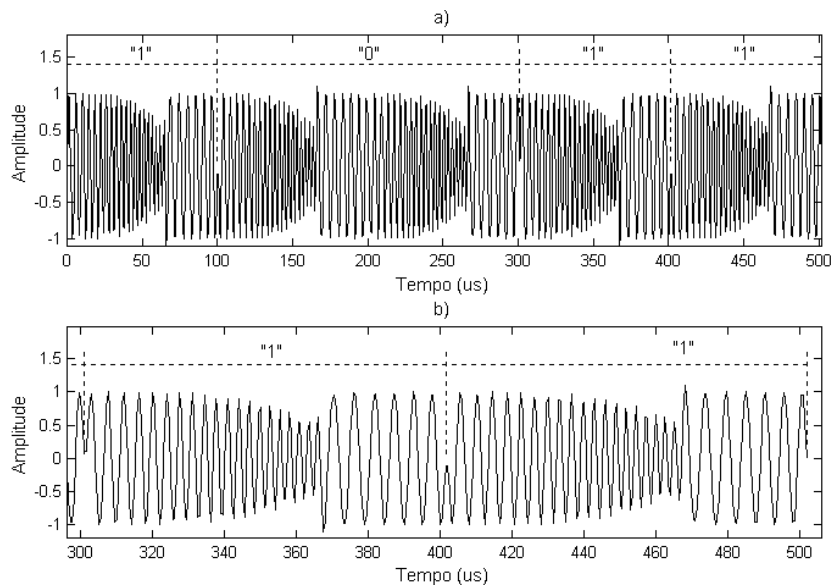


Figura 4.13 – a) Transmissão dos bits “1011” utilizando a codificação PRK. b) Mudança de estado na modulação PRK. Notar que um chirp é exatamente o inverso do outro.

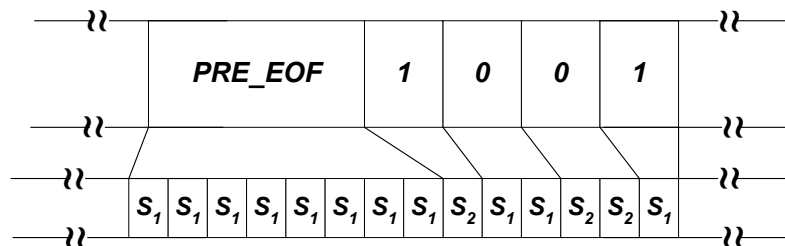


Figura 4.14 – Codificação do início do corpo do pacote *CEBus*. Notar que o primeiro estado do corpo do pacote é codificado com estado diferente do estado do preâmbulo.

$$Overhead_{\text{mínimo}} = \frac{B_A}{B_A + B_I} \cdot 100 = \frac{11}{11 + 32} \cdot 100 = 25.58\% \quad (4.8)$$

$$Overhead_{\text{máximo}} = \frac{B_A}{B_A + B_I} \cdot 100 = \frac{11}{11 + 1} \cdot 100 = 91.7\% \quad (4.9)$$

Portanto, é necessário que se transmita vários bytes em um mesmo pacote, reduzindo assim o overhead e maximizando o uso da linha para transmitir informações.

4.5.2. Taxa de Transmissão Efetiva

Uma medida de eficiência de um protocolo é a quantidade de bytes que podem ser transmitidos em segundo. A essa taxa se dá o nome de bits por segundo (bps), ou ainda, caracteres por segundo (cps). Uma segunda medida de velocidade do protocolo é a quantidade de símbolos que podem ser transmitidos em um segundo, e é conhecida por símbolos por segundo, ou ainda, bauds. Para o protocolo CEBus, cada símbolo tem a duração de 100 μ s, e o cálculo da taxa em bauds é mostrado em (4.10).

$$V_{\text{bauds}} = \frac{1 \text{ símbolo}}{100 \mu\text{s}} = 10000 \text{ símbolos/s} = 10000 \text{ bauds} \quad (4.10)$$

A taxa de transmissão em bps é variável, pois depende da quantidade de bits "0s" e "1s" transmitidos. A maior e a menor taxa de transmissão se dá quando todos os bits a serem transmitidos são "1" e "0", respectivamente (equações (4.11) e (4.12)).

$$V_{\text{max bps}} = \frac{1 \text{ bit}}{100 \mu\text{s}} = 10 \text{ kbps} \quad (4.11)$$

$$V_{\text{min bps}} = \frac{1 \text{ bit}}{200 \mu\text{s}} = 5 \text{ kbps} \quad (4.12)$$

Assumindo que em cada pacote haja uma distribuição uniforme de bits "0" e "1", pode-se separar cada par de bits "1" e "0", e desta forma, calcular a taxa efetiva do protocolo CEBus. Para cada par de bits "1" e "0" transmitido, é necessário um período de tempo de 300 μ s para transmitir dois bits, logo, a taxa efetiva de transmissão é calculada em (4.13):

$$V_{\text{efet}}_{bps} = \frac{2 \text{ bits}}{300 \mu s} = 6.666 \text{ kbps} \quad (4.13)$$

Deve-se utilizar a taxa efetiva de transmissão para avaliar se a velocidade de transmissão dos dados é suficiente para a aplicação proposta.

4.6. Acoplamento Eletro-Magnético entre Fases

O sistema de distribuição de energia elétrica se baseia no uso de transformadores trifásicos (Figura 4.16) para suprir os consumidores com tensão adequada ao uso residencial e industrial (127 V/ 220V). A construção do transformador de distribuição é efetuada em um único circuito magnético que enlaça os enrolamentos das três fases (ao contrário de um banco trifásico de transformadores monofásicos, aonde cada trafo monofásico tem seu próprio circuito magnético). O transformador de distribuição é uma máquina elétrica estática extremamente eficiente, alcançando rendimentos da ordem de 97%. Isso ocorre em função do projeto eficiente do transformador, otimizado para operar na frequência de 60 Hz. Porém, as mesmas características construtivas do transformador que garante sua eficiência na frequência de 60 Hz o faz ser um equipamento extremamente ineficiente quando operando em altas frequências. Isso acontece por vários motivos [11]:

- a) A construção do núcleo em ferro-silício, que é um material que tem um rendimento extremamente pobre em altas frequências;
- b) As perdas por histerese, que são proporcionais à frequência de operação do transformador;
- c) As perdas por correntes parasitas, que são proporcionais ao quadrado da frequência de operação do transformador;
- d) As indutâncias dos enrolamentos do transformador, que criam grandes reatâncias indutivas série em altas frequências.

A utilização do transformador de distribuição trifásico como caminho de propagação na comunicação de dados em altas frequências implica na formação de um canal de comunicação muito pobre em se tratando da resposta em frequência e atenuação desejadas para a maximização do enlace de comunicação [3], pois o sinal que se propaga em uma fase sofre grande atenuação ao percorrer o circuito magnético do transformador trifásico. Outro aspecto que deve ser considerado é a atenuação do sinal de alta frequência provocada pela impedância de terra, em virtude de sua distribuição não-uniforme entre o nó transmissor e o nó receptor. Neste caso, uma abordagem mais específica deve ser feita, levando-se em conta todos os parâmetros que são necessários para a análise de

impedância de terra em sistemas de distribuição trifásicos. Maiores detalhes sobre essa abordagem podem ser encontrados em [14].

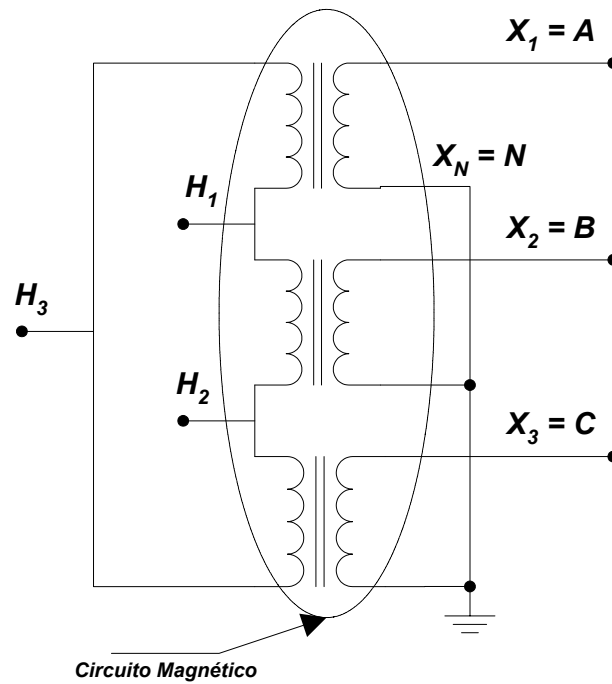


Figura 4.16 – Diagrama esquemático típico de um transformador de distribuição, em configuração Δ -Y. O circuito magnético que enlaça os enrolamentos (primários e secundários) tem elevada relutância em altas frequências.

Considere o diagrama da Figura 4.17, que mostra dois dispositivos *CEBus* conectados na mesma fase (Fase A). O caminho de menor atenuação é entre os dois dispositivos, e o sinal de alta frequência – o *chirp* – circula do transmissor para o receptor sem que haja a necessidade de nenhum tipo de acoplamento entre fases. Neste caso, a atenuação depende somente da impedância dos cabos de energia que conectam os dispositivos e da impedância de aterramento [14].

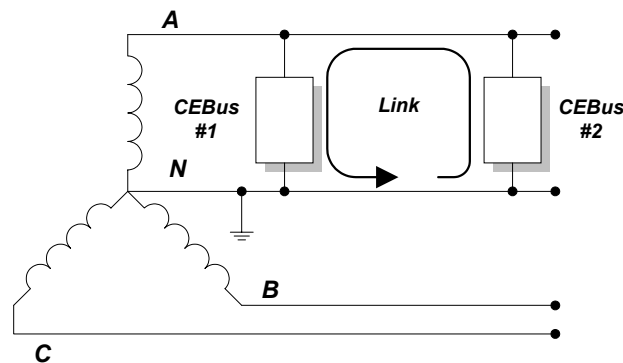


Figura 4.17 – Dispositivos situados na mesma fase do secundário de um transformador de distribuição.

Considere agora a hipótese mostrada na Figura 4.18. O único caminho disponível para transmitir o sinal, entre o transmissor e o receptor, é através dos enrolamentos secundários do transformador de distribuição. Porém, os enrolamentos representam um caminho de alta impedância para o sinal, o que resulta em um canal de comunicação muito pobre, com níveis de atenuação de 30 dB ou mais. A atenuação, combinada com outras fontes de ruído, pode resultar em queda de performance do sistema de comunicação [3].

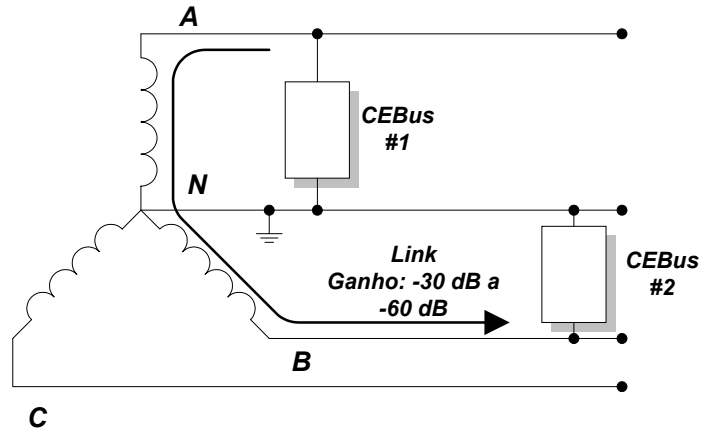


Figura 4.18 – Dispositivos situados em fases diferentes. O único caminho disponível para o estabelecimento do *link* de comunicação é através dos enrolamentos secundários do transformador de distribuição.

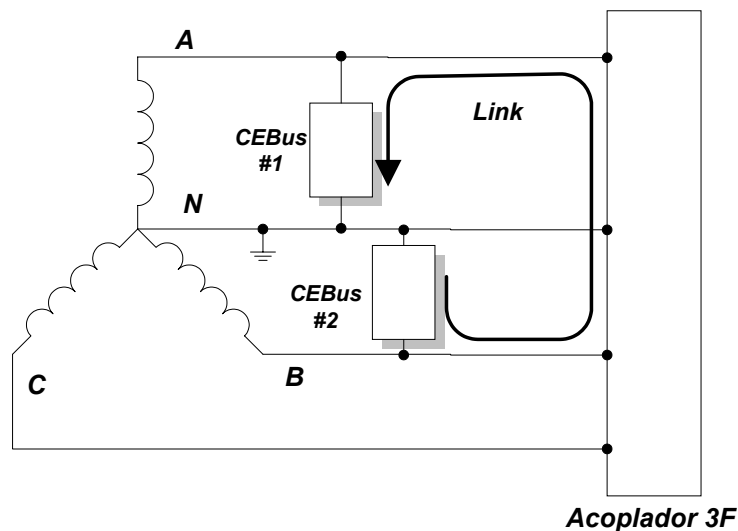


Figura 4.19 – Dispositivos situados em fases diferentes, mas estabelecendo um *link* de comunicação, através do acoplador trifásico conectado às fases do transformador.

Para garantir que dispositivos *CEBus* localizados em fases diferentes possam estabelecer entre si um link de comunicação é necessária a instalação de um dispositivo que permita que os sinais de alta frequência do transmissor possam encontrar um caminho de mais baixa impedância até o receptor, como mostrado na Figura 4.19. O dispositivo de acoplamento assegura um caminho de baixa impedância entre as fases para o sinal de alta frequência. O dispositivo de acoplamento, mostrado na Figura 4.20a é composto por filtros capacitivos e acopla magneticamente as fases através de um transformador de acoplamento, enrolado em um núcleo toroidal de ferrite (para maximizar o fluxo em altas frequências), mostrado em detalhes na Figura 4.20b. Os resistores montados em paralelo com os capacitores têm a função de descarrega-los quando o acoplador é desconectado da rede elétrica, evitando choques elétricos.

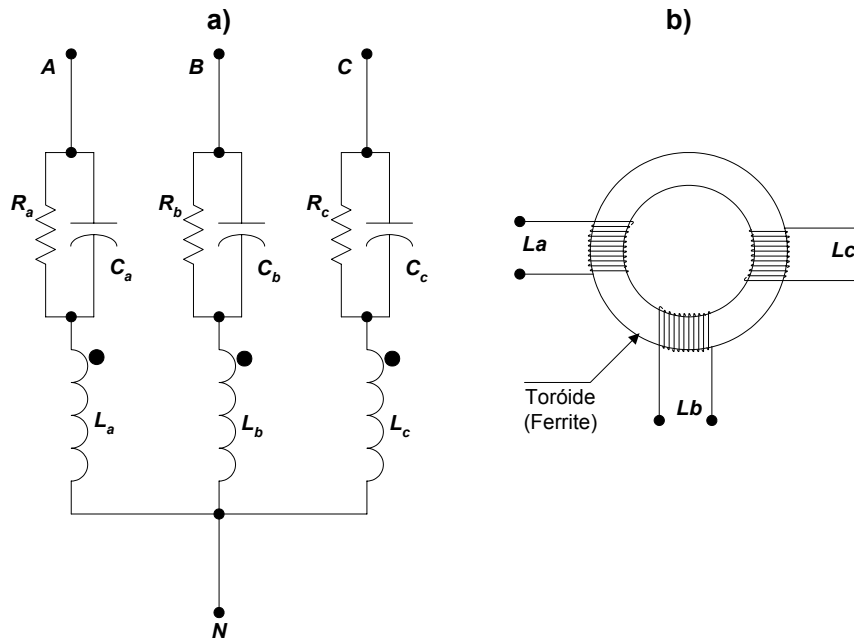


Figura 4.20 – a) Dispositivo de acoplamento de alta frequência entre as fases. b) Detalhe construtivo do transformador de acoplamento.

Para comprovar a necessidade do dispositivo de acoplamento, considere o arranjo de transformadores visualizado na Figura 4.21. Os transformadores trifásicos T1 e T2 têm a finalidade de isolar eletricamente a rede de energia, e garantir que não há cargas conectadas entre fases, o que poderia provocar o acoplamento entre as fases de modo ineficiente e indesejado para o teste proposto [15]. O arranjo é ligado nas fases A, B e C da rede de energia elétrica da concessionária de energia, e as saídas A_i , B_i , C_i e N_i são utilizadas para energizar e isolar os dispositivos *CEBus*, garantindo o isolamento da rede elétrica, mostrado na Figura 4.22. As características nominais (de placa) de cada transformador trifásico são dadas na Tabela 4.3.

Tabela 4.3 – Características nominais do transformador trifásico da Figura 4.21

Fabricante:	ITAVOLT S.A.
Potência Nominal:	1000 VA
Tensão nominal (BT/AT):	110 V / 220V
Frequência Nominal:	50 / 60 Hz
Configuração (BT/AT):	Y – Y

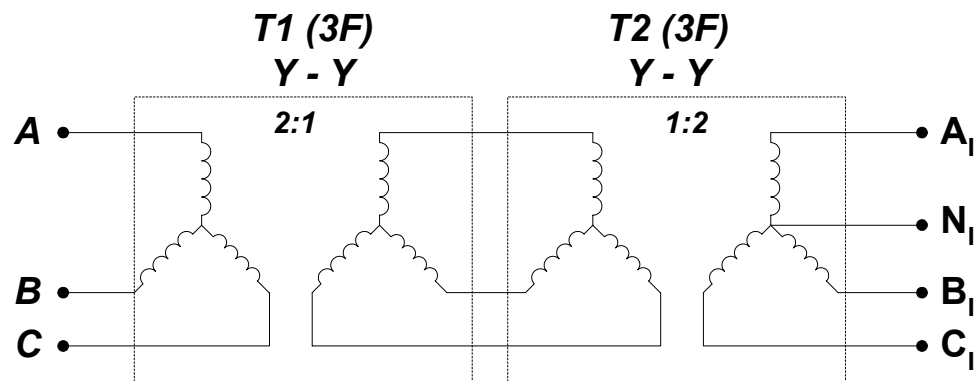


Figura 4.21 – Arranjo de transformadores para isolar eletricamente a rede.

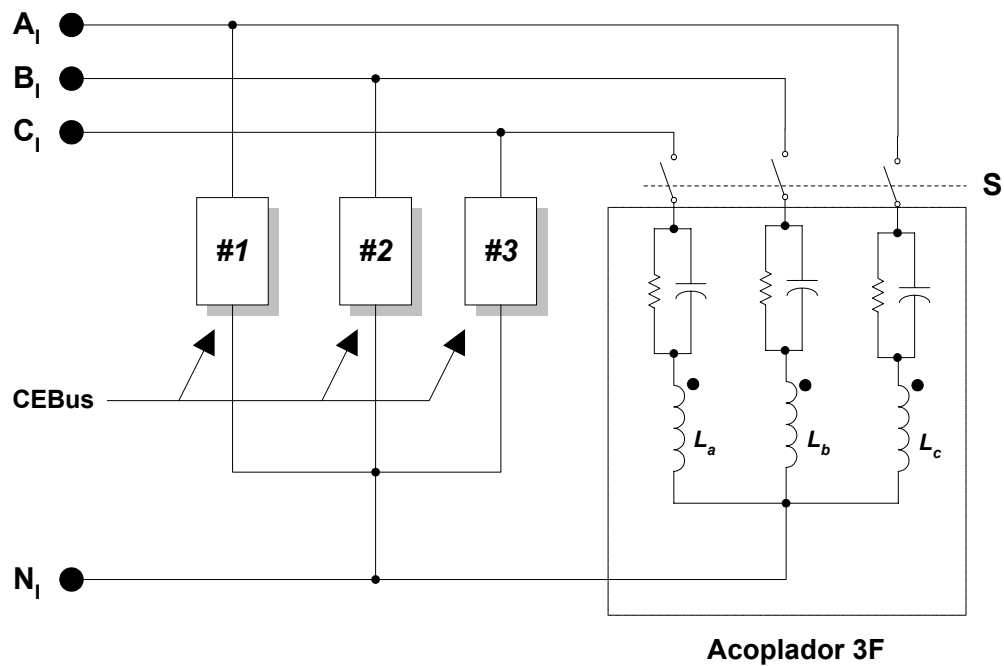


Figura 4.22 – Ligação dos dispositivos CEBus na rede elétrica isolada.

O dispositivo *CEBus* #1 é configurado para enviar um pacote de dados aos dispositivos *CEBus* #2 e #3. A chave **S** permite que o dispositivo de acoplamento seja retirado da linha. O experimento consiste em calcular a energia do sinal de saída do dispositivo *CEBus* #1 e a energia do sinal que é recebida pelos dispositivos *CEBus* #2 e #3. A Tabela 4.4 mostra os resultados obtidos. A energia do sinal foi calculada digitalizando a forma de onda do osciloscópio e obtendo-se o valor numérico com o *MATLAB*.

Tabela 4.4 – Atenuação do Sinal no Experimento da Figura 4.22

Chave S	$\sigma^2_{\#1}$	$\sigma^2_{\#2}$	$\sigma^2_{\#3}$	Atenuação (dB)	
				$10 \cdot \log\left(\frac{\sigma^2_{\#2}}{\sigma^2_{\#1}}\right)$	$10 \cdot \log\left(\frac{\sigma^2_{\#3}}{\sigma^2_{\#1}}\right)$
Aberta	5.445	0.05445	0.0512	-20	-20.26
Fechada	5.445	5.12	4.805	-0.26	-0.54

Como pode ser observado na Tabela 4.4, a colocação do dispositivo de acoplamento melhora a performance do enlace, permitindo um ganho de 20 dB no processo de transmissão. Maiores detalhes sobre a construção e projeto de um acoplador trifásico podem ser encontrados no Apêndice A.

4.7. Conclusões

Neste capítulo foram apresentados os aspectos técnicos do protocolo *CEBus*, que possibilita que dados trafeguem entre dois ou mais nós da rede elétrica utilizando a tecnologia *Spread Spectrum Carrier*, que também foi apresentada neste capítulo. Pôde ser visto que o protocolo *CEBus* aumenta a confiabilidade na transmissão, por prover o sistema de comunicação de ferramentas bastante úteis, como o múltiplo acesso (CSMA) e o controle de erros (CRC). O conhecimento da operação do protocolo *CEBus* na comunicação entre dois nós da rede elétrica é essencial no projeto do modem DLC-P300, que será apresentado no próximo capítulo. Também foram efetuados testes de acoplamento, aonde se verificou a necessidade dos acopladores para garantir um *link* de comunicação confiável em um sistema trifásico.

5. Projeto do modem DLC–P300 (*Digital Line Carrier*)

5.1. Introdução

Como visto no Capítulo 4, a tecnologia *Spread Spectrum Carrier* (SSC) proporciona as condições para a implementação de um dispositivo para comunicação que utiliza a rede elétrica como meio para o tráfego de dados. Neste capítulo, serão mostrados os resultados inerentes ao projeto e construção de um dispositivo de comunicação baseado na tecnologia SSC, denominado DLC–P300. Serão ainda apresentados resultados dos testes efetuados em redes residenciais e comerciais.

5.2. Visão Geral do DLC P–300

5.2.1. Função do DLC–P300

O dispositivo DLC–P300 (abreviatura de *Digital Line Carrier*) é um protótipo de um modem que permite que dois ou mais dispositivos (ligados à rede elétrica), possam transmitir e receber os pacotes de dados do protocolo *CEBus*, através da tecnologia SSC, que trafegam em uma rede elétrica convencional, como mostrado na Figura 5.1.

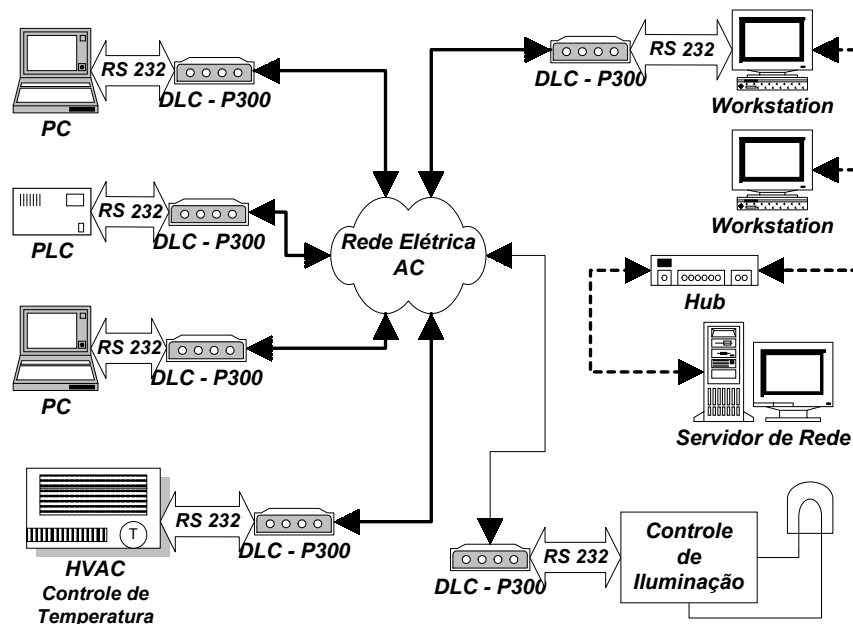


Figura 5.1 – Algumas configurações para a ligação do modem DLC–P300. Todos os dispositivos conectados à rede elétrica podem partilhar os dados.

As principais funções e características do DLC–P300 são:

- a) Codificar e decodificar os símbolos (*chirps*) na transmissão e na recepção dos dados;
- b) Possibilitar o controle e o gerenciamento do protocolo *CEBus*;
- c) Permitir a comunicação com periféricos externos (tipicamente um PC ou um PLC), através de interface RS–232;
- d) Operar remotamente dispositivos de temperatura (HVAC), iluminação e sistemas de segurança (alarmes, sensores de presença, etc.) em ambientes residenciais e industriais;
- e) Possibilitar o compartilhamento de dados com outras redes, como Internet e redes locais;

5.2.2. Componentes do DLC – P300

O protótipo do modem DLC–P300 foi construído em uma caixa de metal com dimensões de 58mmx125mmx225mm (Figura 5.2), contendo uma placa de circuito impresso e um transformador abaixador (Figura 5.3). A placa de circuito impresso (dupla face) contém todos os componentes, mostrados na Tabela 5.1.

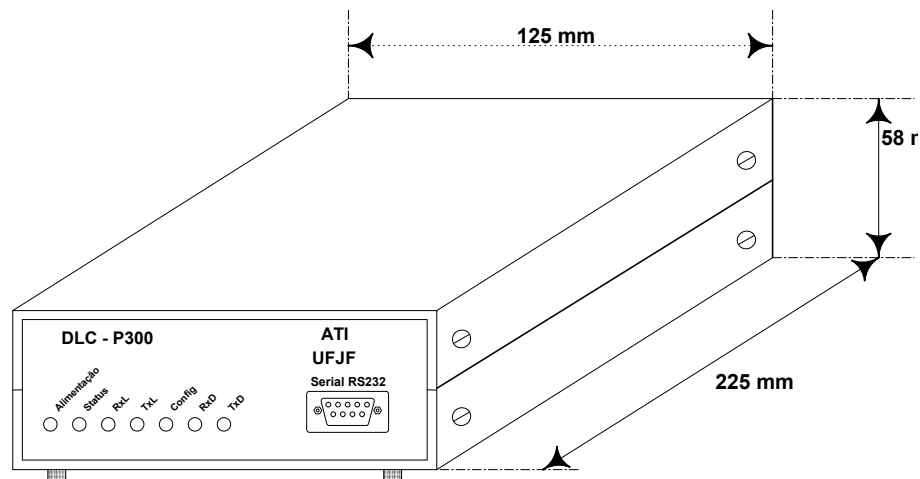


Figura 5.2 – Vista externa e dimensões do DLC–P300.

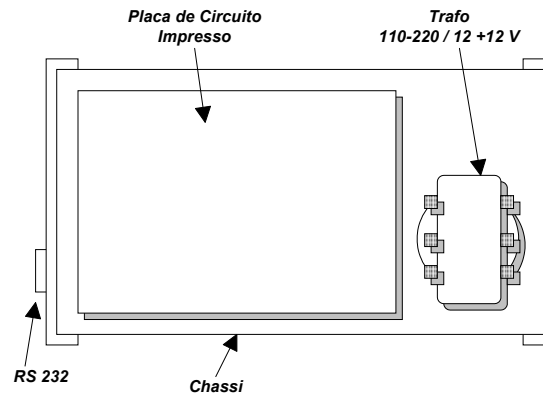


Figura 5.3 – Montagem interna do DLC–P300 (vista superior).

Tabela 5.1 – Principais componentes do modem DLC–P300

Quantidade	Código	Descrição	Fabricante
1	AT89S8252	Microcontrolador de 8 bits	ATMEL
1	P300	Controlador de rede <i>CEBus</i>	INTELLON
1	P111	Amplificador com controle de ganho	INTELLON
1	7805	Regulador de tensão +5 V	MOTOROLA
1	7810	Regulador de tensão +10V	MOTOROLA
1	MAX232	Conversor padrão RS–232 para TTL	ST
1	WB 141	Transformador 110–220 / 12 V	W.B. Ltda
2	AS1RC2	Relé 12 Vcc	METALTEX
1	RCB 0998	Cristal 18.43200 MHz	-
1	RCB 0599	Cristal 12.000 MHz	-
1	TAI–001	Transformador toroidal de ferrite 1:1	ATI

5.2.3. Estrutura do DLC–P300

A Figura 5.4 mostra o diagrama (somente os componentes principais) do modem DLC–P300. A comunicação entre o modem e outros dispositivos externos, tais como PC, PLC, dentre outros, é efetuada utilizando-se uma porta serial RS–232. A própria conexão ao sistema AC permite a alimentação do circuito, através de um transformador abaixador, um retificador de onda completa e circuitos reguladores de tensão (por conveniência, os circuitos de alimentação não são mostrados na Figura 5.4).

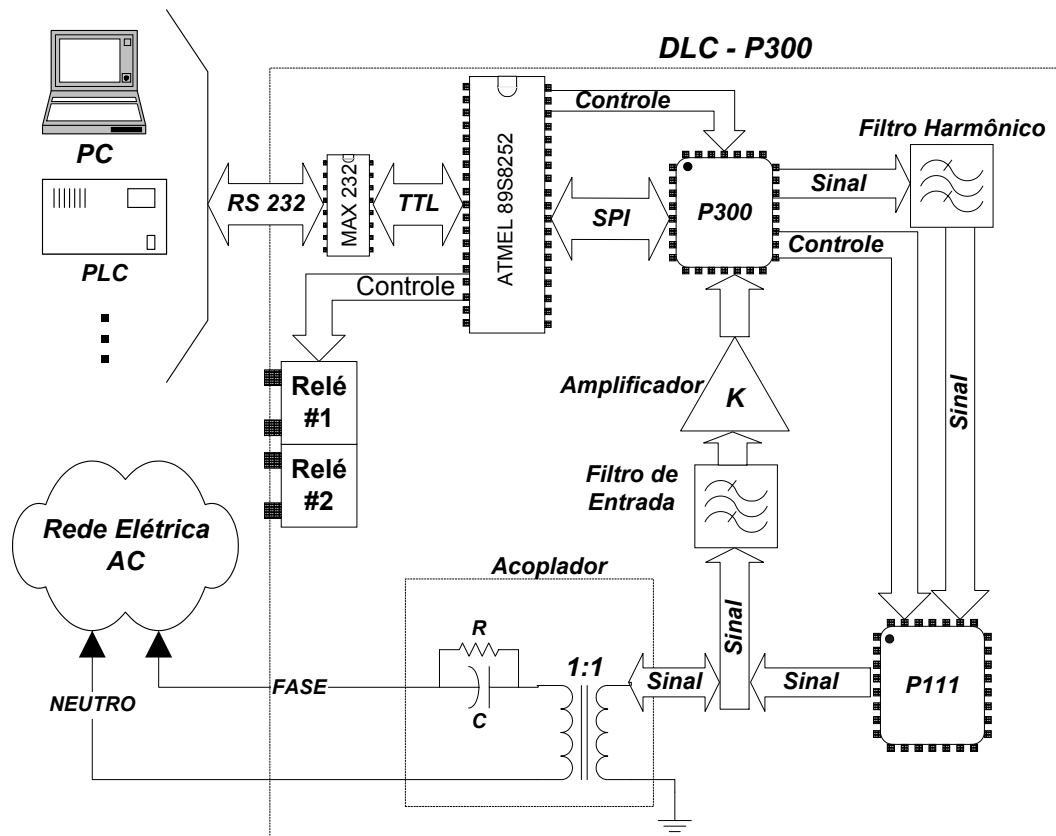


Figura 5.4 – Diagrama esquemático principal do modem DLC-P300.

5.3. Controlador de Rede Intellon® SSC P300 PL.

O principal componente do modem DLC-P300 é o circuito integrado SSC P300 PL, fabricado pela Intellon®, que implementa diversas funções, tais como [8]:

- a) Implementa o protocolo EIA-600 (*CEBus*) para controle de acesso e serviços de comunicação;
- b) Especifica serviços de DLL (Data Link Layer) do protocolo *CEBus*, como transmissão e recepção de pacotes, conversão byte-símbolo para pacotes a serem transmitidos, conversão símbolo-byte para pacotes recebidos, acesso ao canal (de acordo com as prioridades do pacote e das regras de acesso do protocolo *CEBus*), inclusão e cálculo de CRC;
- c) Implementa um *transceiver* com tecnologia Spread Spectrum Carrier, permitindo o múltiplo acesso por CSMA;
- d) Comunicação *Host*-P300 de alta velocidade (SPI – *Serial Peripheral Interface*) de até 1 Mbps;

e) Necessidade de poucos componentes externos.

A Figura 5.5 mostra a pinagem do SSC P300, e a Figura 5.6 mostra sua estrutura interna, em diagrama de blocos.

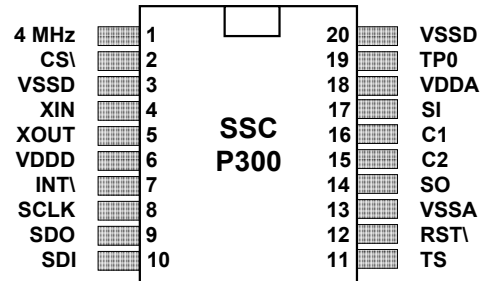


Figura 5.5 – Pinagem do circuito integrado (IC) SSC P300.

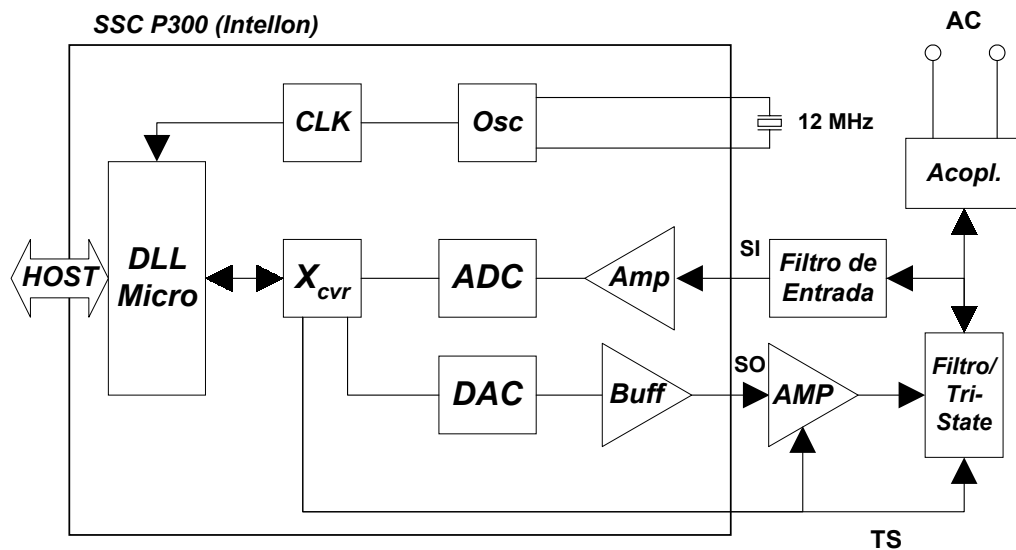


Figura 5.6 – Estrutura interna do circuito integrado SSC P300.

O controle do SSC P300 é efetuado por um microcontrolador (*host*), que possuem entre si uma relação mestre-escravo, ou seja, os dados armazenados no P300 só podem ser enviados quando o *host* ordenar tal ação. O P300 somente pode requisitar serviços do *host* via um sinal de interrupção. O *host* responde ao pedido de interrupção enviando comandos, que permite que os dados armazenados nos registros internos do P300 sejam transferidos para o *host*. Estes registros internos são organizados em estruturas lógicas, que são usadas para configurar o P300, ajustar os parâmetros do link de dados, retornar o status do P300, transferir de e para o *host* os pacotes recebidos e enviados, respectivamente.

Toda a comunicação *host* – P300 é efetuada através de comandos específicos, cada um permitindo o acesso a uma estrutura lógica específica. Cada estrutura admite a

leitura ou a escrita dos seus dados em caráter mutuamente exclusivo, ou seja, as estruturas podem ser do tipo *ready-only* ou *write-only*. Não é possível escrever em uma estrutura *ready-only*, nem acessar os dados de uma estrutura *write-only*. Logo, os comandos são agrupados em comandos de leitura e de escrita. Uma descrição detalhada dos comandos e suas respectivas estruturas podem ser encontrados em [8]. O protocolo de transmissão dos comandos de escrita (*write-only*) entre o *host* e o P300 é mostrado na Figura 5.8. A conexão *host*-P300 pode ser observada na Figura 5.7.

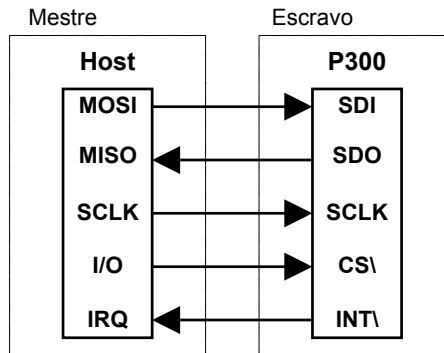


Figura 5.7 – Conexão Host – P300.

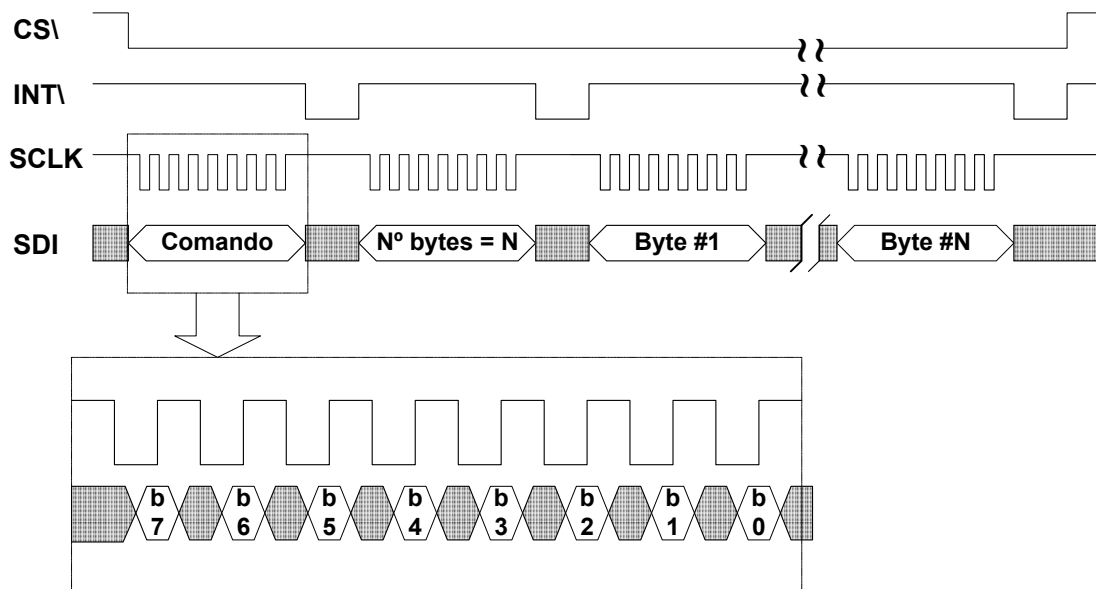


Figura 5.8 – Protocolo dos comandos de escrita (*write-only*).

O protocolo de transmissão dos comandos de escrita (*write-only*) entre o *host* e o P300 é mostrado na Figura 5.9.

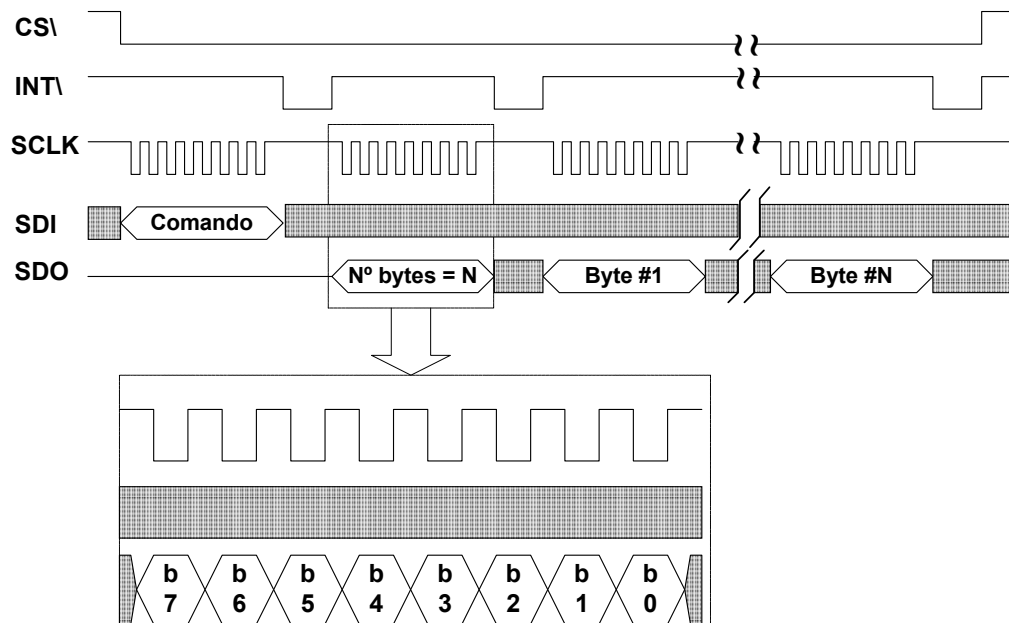


Figura 5.9 – Protocolo dos comandos de escrita (*read-only*).

Como pode ser observada, a transmissão de dados entre o *host* e o P300 é serial e síncrona, e é conhecida pela sigla SPI (SPI – *Serial Peripheral Interface* – Interface Serial Periférica). A transmissão de dados por uma porta SPI tem a vantagem de poder transmitir uma palavra de qualquer comprimento (8 bits, 16 bits, 32 bits, etc) com somente três fios. A desvantagem é que o tempo de transmissão cresce com o número de bits da palavra, resultando em uma transmissão mais lenta, desvantagem essa que pode ser minimizada utilizando-se altas taxas de transmissão. Tipicamente, o P300 opera com taxas de transmissão na porta SPI de aproximadamente 1 Mbps, tendo a frequência máxima de operação de 2 Mbps [27].

Deve-se observar alguns detalhes na transferência dos bytes pela SPI:

- A cada byte transmitido, o P300 sinaliza com um pulso em nível baixo no pino INT depois da transmissão, que sinaliza que o byte foi corretamente transmitido e que o próximo byte pode ser enviado. Esse pulso (chamado de *handshake*) deve vir, no máximo, em até 10 ms após a transmissão do byte (tipicamente, este valor é ajustado em 1 ms). A não ocorrência do sinal neste intervalo de tempo significa que houve erros na transmissão, e deve-se abortar a operação de escrita (ou leitura) dos dados.
- Nas operações de leitura, os dados são colocados pelo P300 na via MISO do *host* (SDO do P300). Como a porta SPI só é acionada quando se escreve um byte na via MOSI do *host* (SDI do P300), deve-se colocar um byte aleatório (chamado de “*dummy*” byte) para iniciar a transferência de dados. Após a

transferência do *dummy* byte, os dados provenientes da via SDO do P300 são colocados na porta SPI.

O fluxograma da Figura 5.10, como parte do programa monitor que gerencia o *host*, exemplifica todos os passos necessários para que comandos do tipo write-only possam ser transferidos entre o *host* e o P300 (onde COMM é o comando a ser transferido). Já o fluxograma da Figura 5.11 mostra os passos necessários para a transferência dos comandos *read-only*.

Como dito anteriormente, o P300 deve sinalizar com um sinal de interrupção quando ocorre algum evento (tipicamente, um pacote CEBus foi recebido ou enviado com sucesso). Este sinal é ligado a um pino de interrupção externa do *host*, que deve executar uma rotina de verificação de eventos para determinar o motivo da requisição do serviço, chamada de rotina de atenção. Tal rotina consiste da transferência de uma determinada estrutura do P300, chamada de Interface Flags, que contém a identificação do evento ocorrido. Tal estrutura contém um byte, e cada um de seus bits contém um flag para um determinado evento. A Figura 5.12 mostra a estrutura Interface Flags e a descrição dos seus bits.

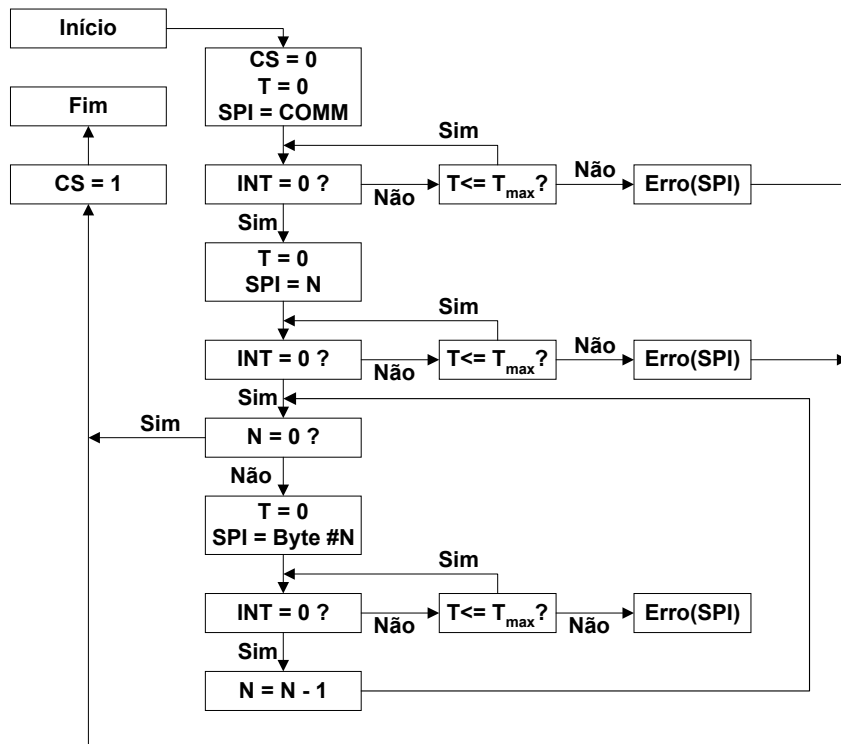


Figura 5.10 – Fluxograma da transferência de comandos *write-only*.

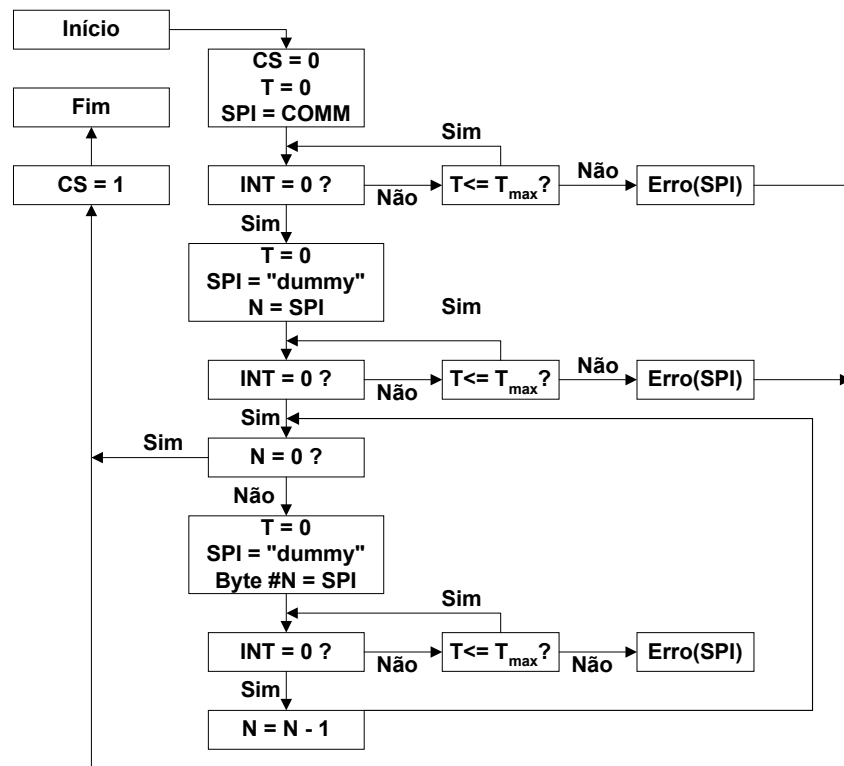


Figura 5.11 – Fluxograma da transferência de comandos *read-only*.

Interface Flags

b7	b6	b5	b4	b3	b2	b1	b0
RA	TA	TEX	REX	PLF	IE	LSC	RES

Figura 5.12 – Estrutura Interface Flags do P300.

A descrição dos bits da estrutura Interface Flags é mostrada a seguir:

Bit 7: *Reception Attention (RA)*: Se igual a 1, indica que um pacote válido (sem errors) foi recebido.

Bit 6: *Transmittion Attention* (TA): Se igual a 1, indica que a transmissão do pacote foi efetuada.

Bit 5: *Transmission Exception (TEX)*: Se igual a 1, indica que houve alguma ocorrência no envio do pacote (pacote ainda sendo transmitido ou erro na transmissão).

Bit 4: *Reception Exception (REX)*: Se igual a 1, indica que houve alguma ocorrência no pacote recebido.

Bit 3: *Physical Failure (PLF)*: Se igual a 1, indica que houve algum erro no layer físico.

Bit 2: *Interface Error (IE)*: Se igual a 1, indica que houve erro na transmissão dos dados na porta SPI.

Bit 1: *Link Status Condition (LSC)*: Se igual a 1, indica que houve algum *overflow* de algum contador interno.

Bit 0: *IC Reset (RES)*: Se igual a 1, indica que houve algum *overflow* de algum contador interno.

Em cada flag da estrutura Interface Flags, a ocorrência de algum bit igual a 0 indica que não houve nenhuma exceção relacionada àquele flag. No caso de alguma exceção encontrada, pode ser necessário ler uma ou mais estruturas do P300. O fluxograma da Figura 5.13 mostra como a rotina de interrupção externa deve proceder em uma rotina de atenção.

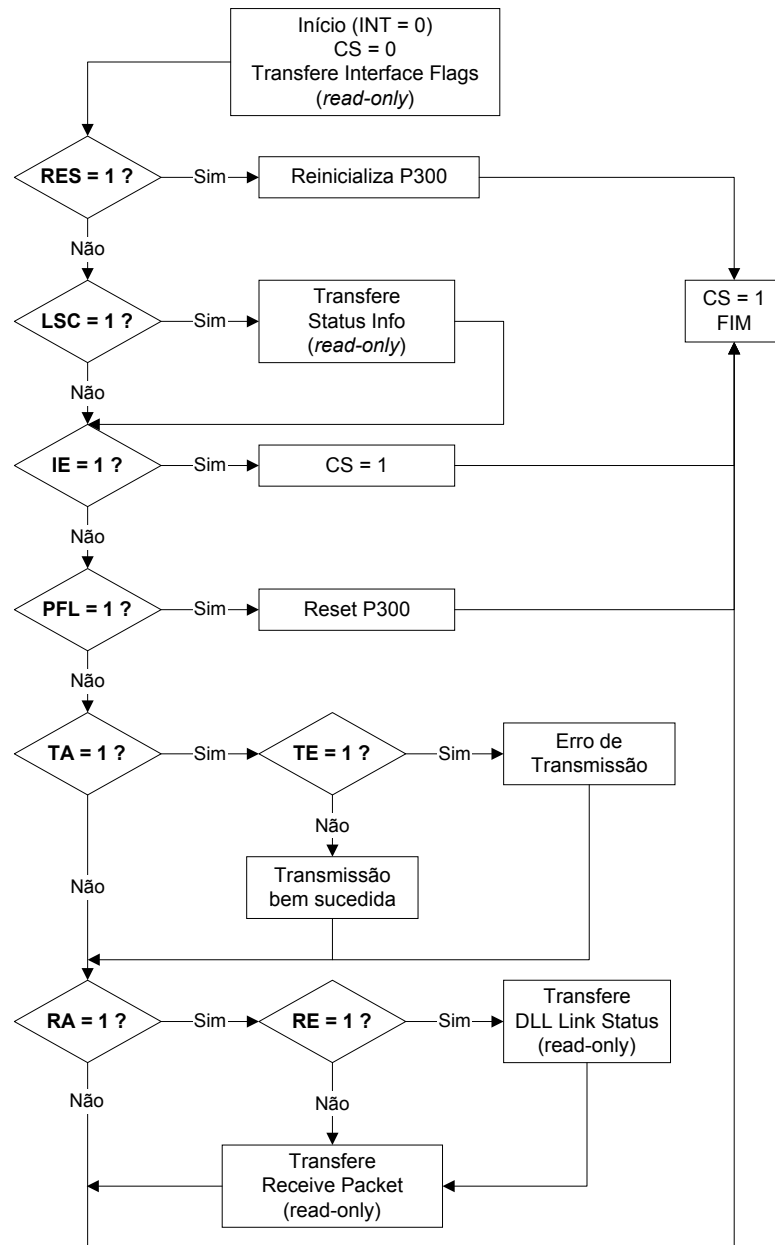


Figura 5.13 – Fluxograma de uma rotina de atenção.

Como observado no fluxograma da Figura 5.13, a rotina de atenção pode requisitar do P300 outras estruturas (tais como Status Info, Receive Packet e DLL Link Status). Dependendo do estado de cada uma destas estruturas, outras devem ser transferidas [8]. Maiores referências da elaboração e projeto do software monitor do *host* podem ser encontrados em [28].

5.4. Amplificador com controle de ganho Intellon® SSC P111.

A próxima etapa na transmissão do pacote *CEBus* é garantir a amplificação do sinal transmitido e promover a isolamento do controlador de rede (P300) do restante dos circuitos analógicos. Isto pode ser obtido com o IC SSC P111, um amplificador com controle de ganho que é compatível com o controlador de rede SSC P300. Algumas características do P111 são mostradas a seguir [7]:

- a) Amplificador classe AB, com baixo consumo de potência (máximo de 2.5 W na transmissão e 200 mW na recepção)
- b) Chave de três estados
- c) Controle da tensão do sinal de saída de acordo com a impedância da rede (tipicamente 6 Vp-p para carga de 10 Ω).
- d) Circuito de proteção contra altas temperaturas (atua no ganho do amplificador)

O diagrama da Figura 5.14 mostra a arquitetura interna do SSC P111 e sua respectiva pinagem, e a Figura 5.15 mostra a característica da tensão de saída e a corrente de alimentação de acordo com a carga.

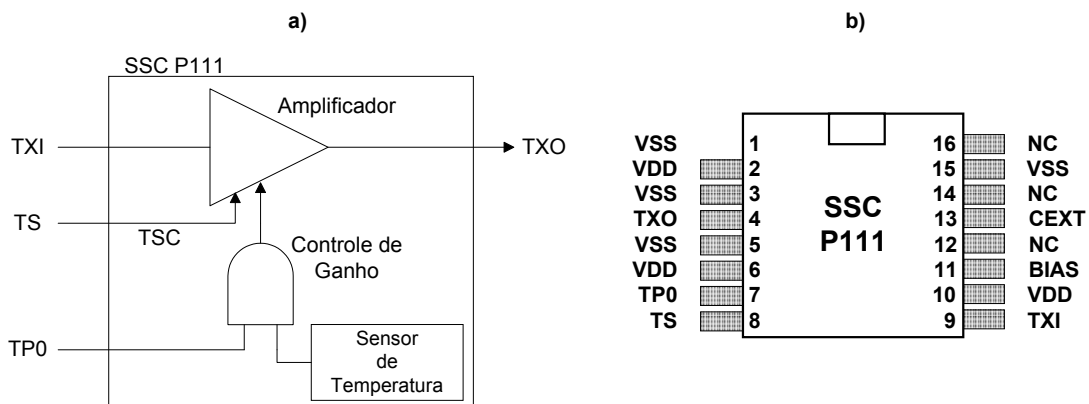


Figura 5.14 – a) Diagrama de blocos do IC SSC P111 b) Pinagem do IC.

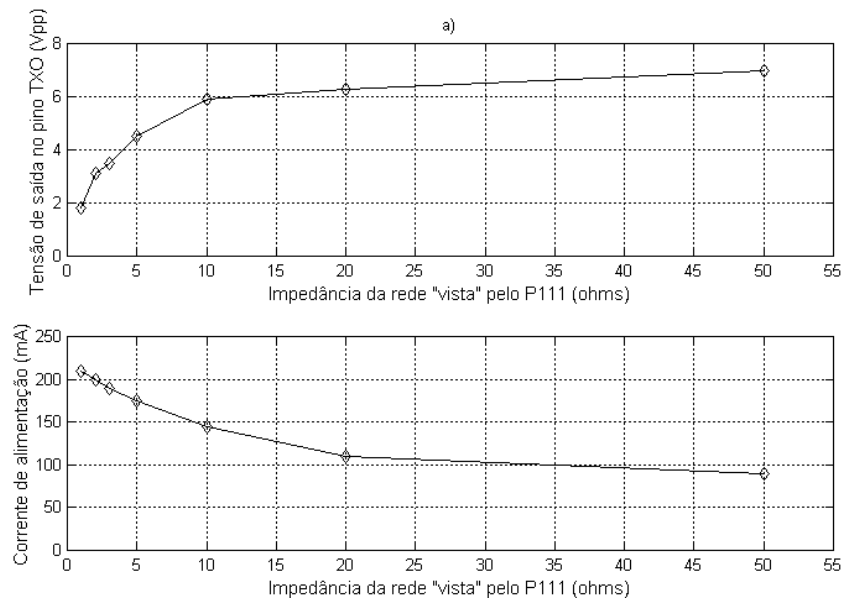


Figura 5.15 – Características de tensão e corrente do IC P111.

A conexão do SSC P111 com o SSC P300 é mostrada na Figura 5.16, com os filtros necessários para a operação do conjunto P111 – P300.

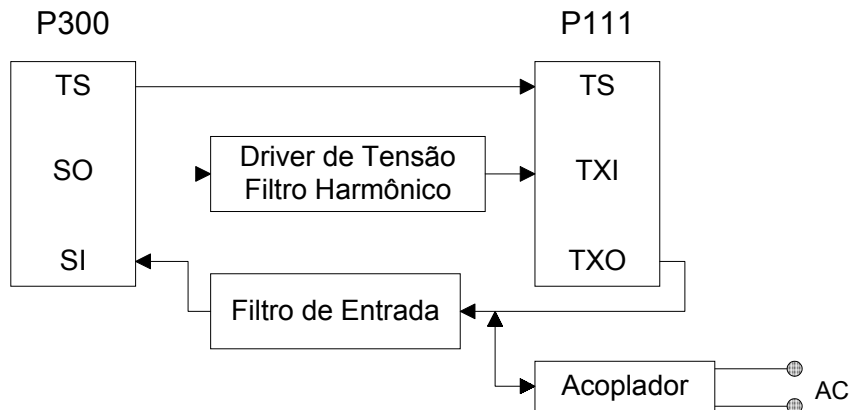


Figura 5.16 – Conexão P111 – P300.

O controle de temperatura do P111 garante que o ganho do amplificador seja ajustado para o valor unitário, caso a temperatura no chip subir além de um determinado limite.

Quando no estágio de transmissão, o controlador de rede P300 ajusta o nível lógico no pino TS para “1”, habilitando o circuito de amplificação do IC P111. Quando no estágio de recepção, o nível lógico do pino TS é colocado em “0”, e o amplificador passa ter uma alta impedância de saída, e o sinal recebido é totalmente desviado para o filtro de entrada.

5.5. Driver de Tensão, Filtro Harmônico, Filtro de Entrada e Acoplador.

5.5.1.Introdução

No DLC–P300, os sinais de entrada e saída são filtrados por filtros analógicos, formados por redes RLC passivas, necessários para condicionar o sinal a padrões aceitáveis [6]. A função de cada filtro é mostrada a seguir.

5.5.2.Filtro Harmônico / Driver de Tensão

O filtro harmônico tem a função de limitar o espectro harmônico máximo do sinal a ser transmitido. Isso é necessário porque o sinal a ser transmitido é gerado (internamente ao P300) pela leitura de uma tabela em uma ROM interna, e desta forma, o sinal tem componentes harmônicas fora do intervalo de frequência especificado no Capítulo 3. O filtro empregado é um filtro passa-baixa de 3 pólos, com um transistor atuando como seguidor de tensão, necessário para limitar a corrente drenada do P300. A função de transferência do filtro empregado é mostrada em (5.1). O filtro de entrada pode ser observado na Figura 5.17, e a característica de ganho de tensão em função da frequência pode ser observada na Figura 5.18, para valores nominais de projeto. Após o sinal passar pelo filtro, ele é conduzido ao pino TXI do IC P111.

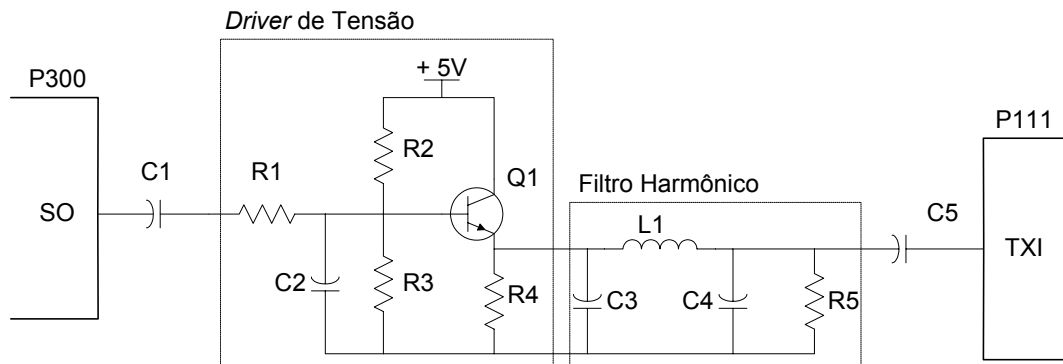


Figura 5.17 – Filtro harmônico e *driver* de tensão.

$$H_{saída} = \frac{R_5}{(L_1 \cdot C_4 \cdot R_5) \cdot s^2 + L_1 \cdot s + R_5} \quad (5.1)$$

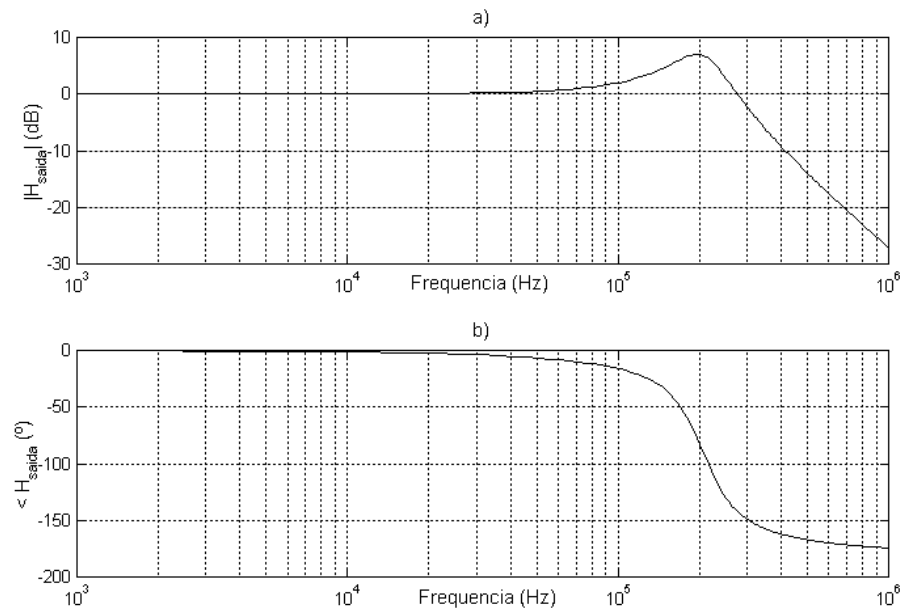


Figura 5.18 – a) Magnitude do filtro harmônico (em dB); b) Fase do filtro harmônico (em graus). Gráfico para valores nominais de projeto: $L_1 = 180\mu\text{H}$, $C_4 = 3300\text{ pF}$ e $R_5 = 510\ \Omega$.

5.5.3. Filtro de Entrada

O pacote *CEBus* proveniente da rede é desviado pelo P111 para o filtro de entrada. Porém, o sinal pode estar contaminado por ruídos de alta frequência, harmônicos da rede elétrica, dentre outros, que podem atrapalhar o processo de detecção do sinal no IC P300. Desta forma, o sinal a ser enviado para ser decodificado no IC P300 deve ser condicionado para atenuar interferências com grande energia que estão fora do intervalo especificado do sinal.

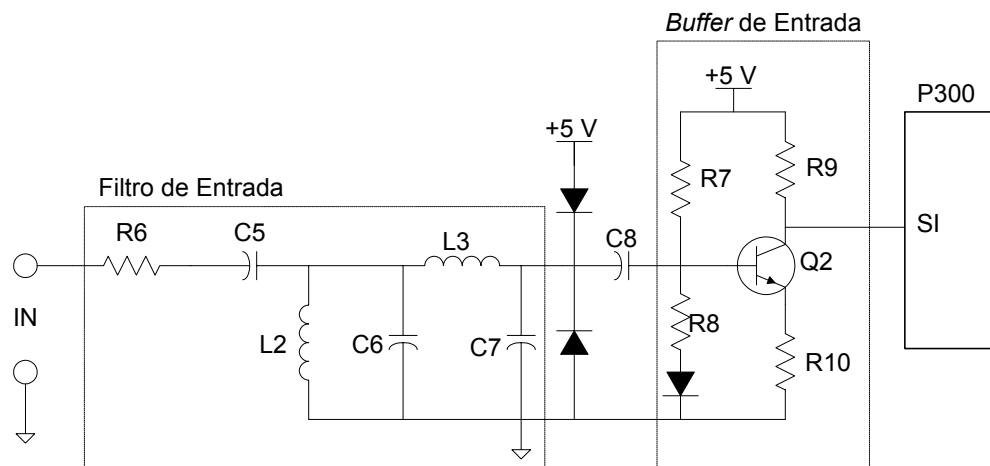


Figura 5.19 – Diagrama do filtro de entrada e da etapa amplificadora.

A função de transferência do filtro de entrada pode ser vista em (5.2), e os gráficos da Figura 5.20 mostram a característica de ganho de tensão em função da frequência e fase.

$$\left\{ \begin{array}{l} H_{saida} = \frac{L_2 \cdot C_5 \cdot s^2}{a_5 \cdot s^5 + a_4 \cdot s^4 + a_3 \cdot s^3 + a_2 \cdot s^2 + a_1 \cdot s + a_0} \\ a_5 = R_6 \cdot L_2 \cdot L_3 \cdot C_5 \cdot C_6 \cdot C_7 \\ a_4 = (C_5 + C_6) \cdot L_2 \cdot L_3 \cdot C_7 \\ a_3 = (L_2 \cdot C_5 + L_2 \cdot C_7 + L_3 \cdot C_7) \cdot R_6 \cdot C_5 \\ a_2 = (L_2 \cdot (C_5 + C_6 + C_7) + L_3 \cdot C_7) \\ a_1 = R_6 \cdot C_5 \\ a_0 = 1 \end{array} \right. \quad (5.2)$$

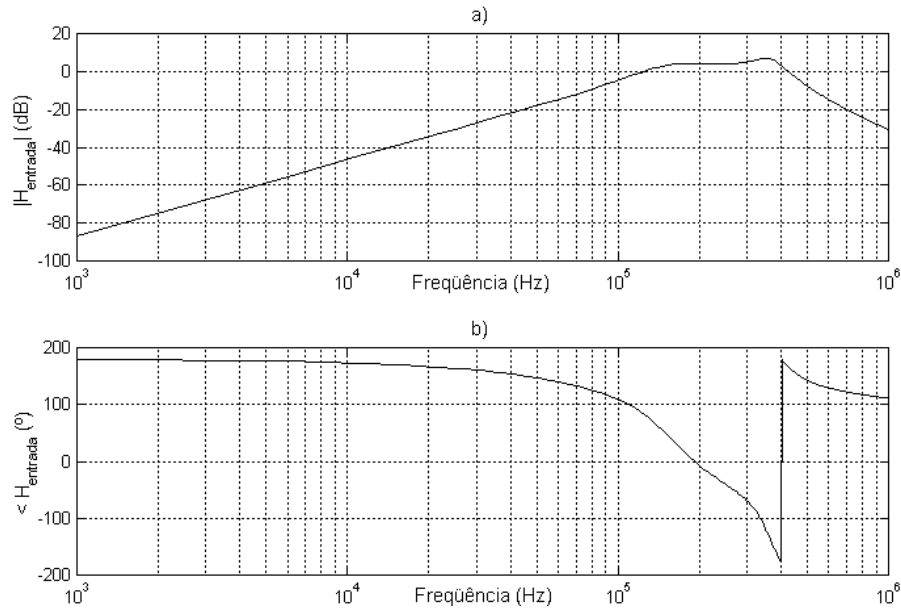


Figura 5.20 – a) Magnitude do filtro de entrada (em dB); b) Fase do filtro de entrada (em graus). Gráfico para valores nominais de projeto: $L_2 = 120\mu\text{H}$, $L_3 = 180\mu\text{H}$, $C_5 = 10\text{ nF}$, $C_6 = 3300\text{ pF}$, $C_7 = 1500\text{ pF}$ e $R_6 = 180\ \Omega$.

Uma vez filtrado, o sinal é amplificado por um transistor, com um ganho aproximado de 20 dB, para compensar eventuais perdas no processo de transmissão. O ganho é calculado em (5.3). O valor de 20 dB é escolhido de forma que o dispositivo *CEBus* seja capaz de receber sinais de até 1mVp-p. Ganhos com valores diferentes podem ser

utilizados para maximizar a capacidade de recepção, mas em contra-partida reduzem a imunidade ao ruído do dispositivo.

$$G_{Q2}(dB) \approx 20 \cdot \log_{10} \left(\frac{R_9}{R_{10}} \right) \quad (5.3)$$

5.5.4. Acoplador da Rede Elétrica

O acoplador tem a função de induzir o sinal do pacote *CEBus* proveniente do IC P111 na rede elétrica, e extrair o sinal da rede elétrica e direcioná-lo ao filtro de entrada. Outra função do capacitor é bloquear o sinal de 60 Hz da rede elétrica para evitar que altas tensões possam chegar aos circuitos integrados do DLC-P300. A Figura 5.21 mostra o diagrama esquemático do acoplador, e os valores nominais para o acoplador são: $R_{ac}=470k\Omega$, $C_{ac}=470$ nF, $L_{ac}=125$ μ H, $C_8=1\mu$ F e $L_4=1.8\mu$ H. As indutâncias são montadas em núcleo toroidal de ferrite.

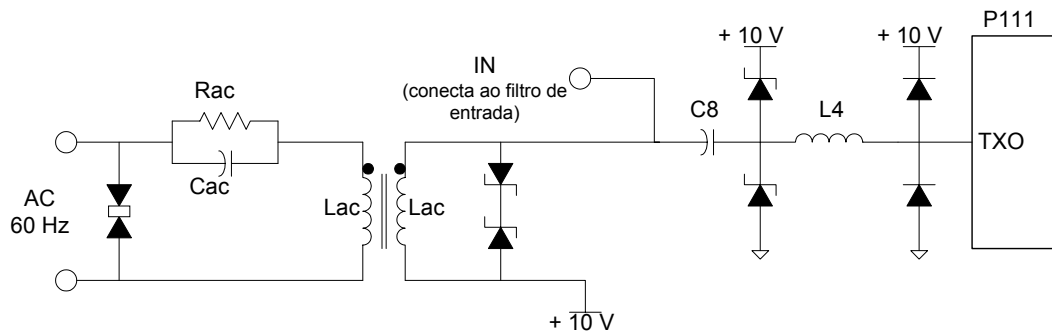


Figura 5.21 – Diagrama esquemático do acoplador, com as conexões aos sistemas de entrada e saída do sinal do pacote *CEBus*.

As indutâncias L_{ac} são montadas em um núcleo de ferrite, semelhante ao mostrado na Figura 4.20 b. O capacitor R_{ac} tem a função principal de bloquear os sinais de 60 Hz da rede AC, logo, tem um alto valor de tensão de isolamento (tipicamente 400 V). Os diodos zener têm a função de proteção do circuito e limitação do sinal de entrada. O dispositivo MOV (varistor de óxido metálico) na entrada do acoplador tem a função de proteger o circuito de entrada de transitórios na rede elétrica, que podem causar danos aos circuitos integrados. O diagrama de Bode da Figura 5.22 mostra, para os valores nominais, a relação entre a saída do filtro de entrada e a entrada AC (sem os dispositivos de proteção).

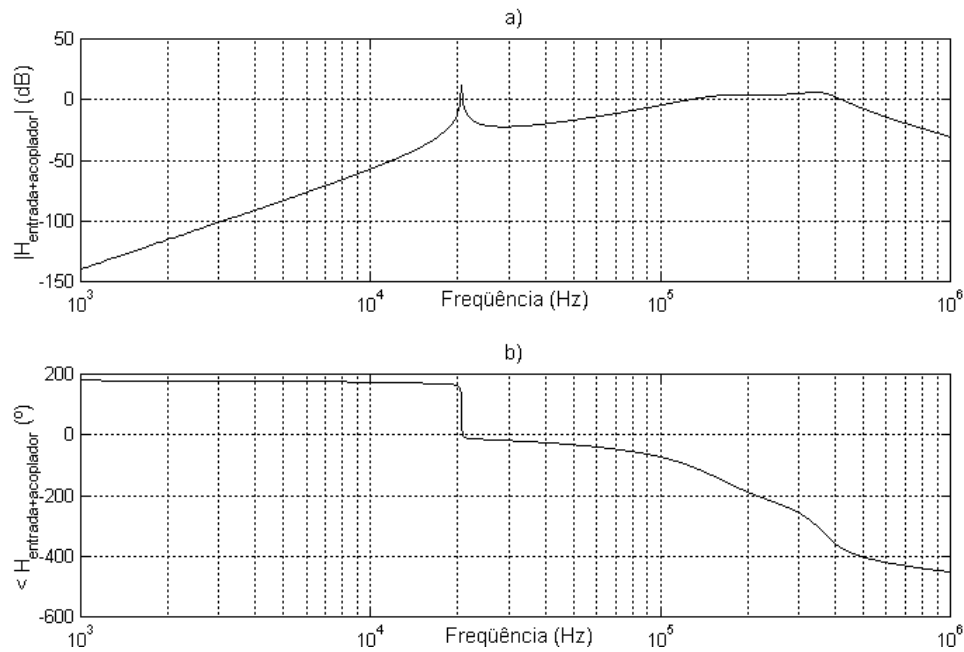


Figura 5.22 – a) Magnitude da relação V_{C7} / V_{AC} . b) Fase

A análise do função de transferência do acoplador da Figura 5.23 mostra a relação entre a tensão de entrada e a tensão de saída, dada em (5.4).

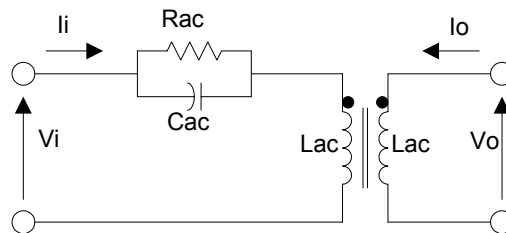


Figura 5.23 – Acoplador

$$\frac{V_o}{V_i} = \frac{k \cdot (L_a \cdot R_a \cdot C_a \cdot s^2 + L_a \cdot s)}{L_a \cdot R_a \cdot C_a \cdot s^2 + L_a \cdot s + R_a}, I_o = 0 \quad (5.4)$$

A constante k da expressão (5.4) é o fator de acoplamento entre as bobinas do transformador de acoplamento, que por estarem em um circuito magnético de alto rendimento, têm um elevado valor de k (superior a 0.95).

O diagrama de Bode da Figura 5.24 mostra a característica V_o/V_i para os valores nominais dos componentes. Notar que na banda de passagem o ganho é 0 dB, enquanto que para a frequência de 60 Hz o ganho é de aproximadamente -105 dB.

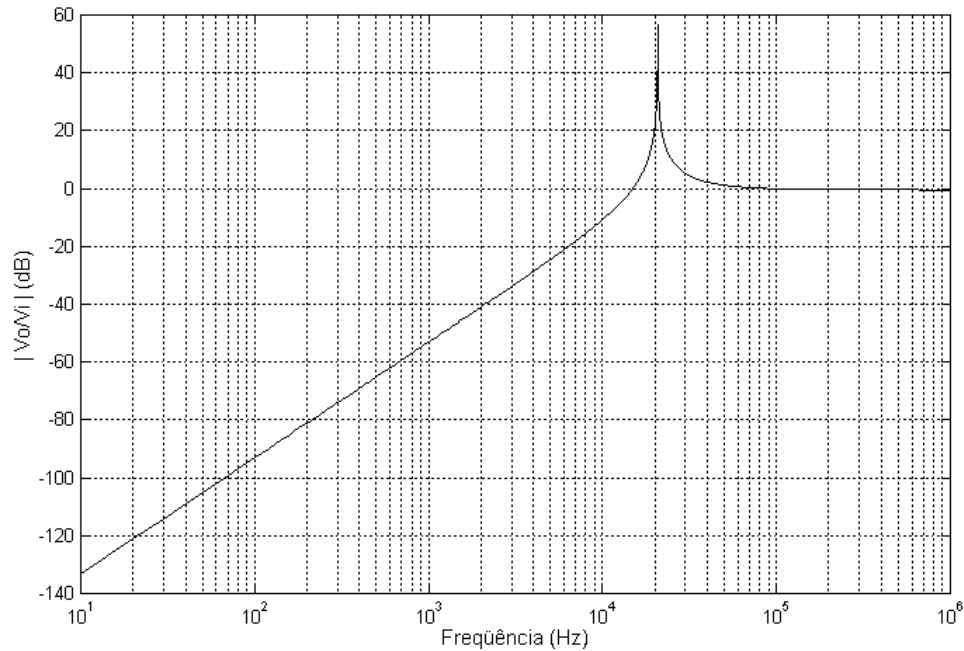


Figura 5.24 – Característica V_o/V_i do acoplador da Figura 5.23.

5.6. Resultados

5.6.1. Simulação no PSPICE®

Para a simulação do DLC-P300, foi construído um modelo equivalente no simulador PSPICE, mostrado na Figura 5.25, aonde dois dispositivos DLC-P300 são colocados em uma rede elétrica monofásica, representada pelas resistência R_{Linha} .

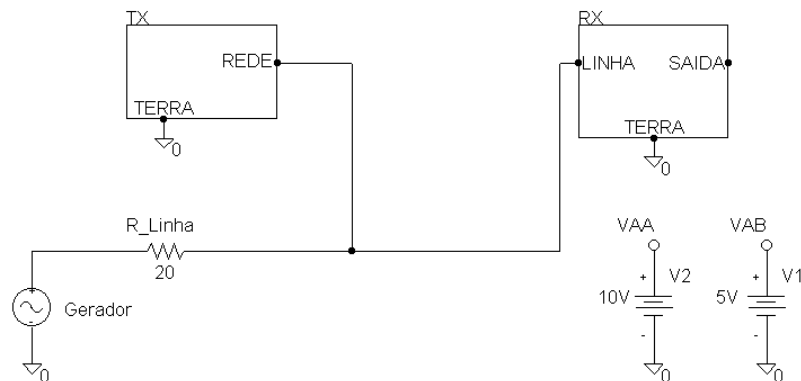


Figura 5.25 – Simulação de dois dispositivos DLC-P300 (TX e RX)

O bloco TX (transmissor) é mostrado na Figura 5.26, enquanto que o bloco RX (receptor) é mostrado na Figura 5.27. O objetivo desta simulação é verificar se o sinal de alta frequência é injetado na rede pelo módulo transmissor e se o módulo receptor consegue captar o sinal e amplificá-lo. Para que a simulação do DLC-P300 pudesse ser realizada, alguns componentes do circuito foram simplificados, como os IC's P111 e P300. As modificações efetuadas para a simulação são:

- Em função do simulador PSPICE (Versão 8.0 *Evaluation*) ter limitações no número de nós que se pretende simular, o filtro e o buffer de entrada do bloco TX foram suprimidos, bem como o filtro de saída do bloco (RX);
- O IC P111 foi substituído por uma fonte controlada com ganho de 1.5 e uma grande impedância de entrada ($10^4 \Omega$);
- O IC P300 foi substituído por uma série de *chirps* gerados de acordo com a equação (4.4), gravados em um arquivo de dados (*spread.plc*);
- Foram introduzidos resistores pequenos (0.1Ω) para solucionar problemas de oscilação numérica no ramo do acoplador;
- A impedância da linha foi substituída por uma carga resistiva;
- O varistor (MOV) de proteção foi retirado da simulação;
- Os transformadores de acoplamento foram substituídos por modelos lineares, com o fator de acoplamento k igual a 0,95.

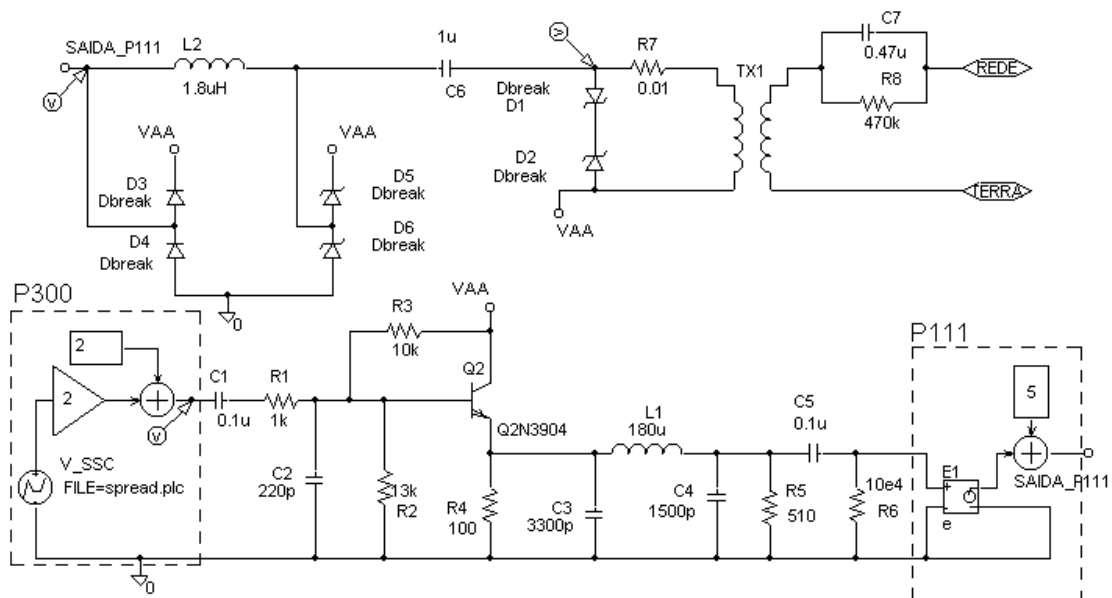


Figura 5.26 – Diagrama de simulação do circuito transmissor (TX)

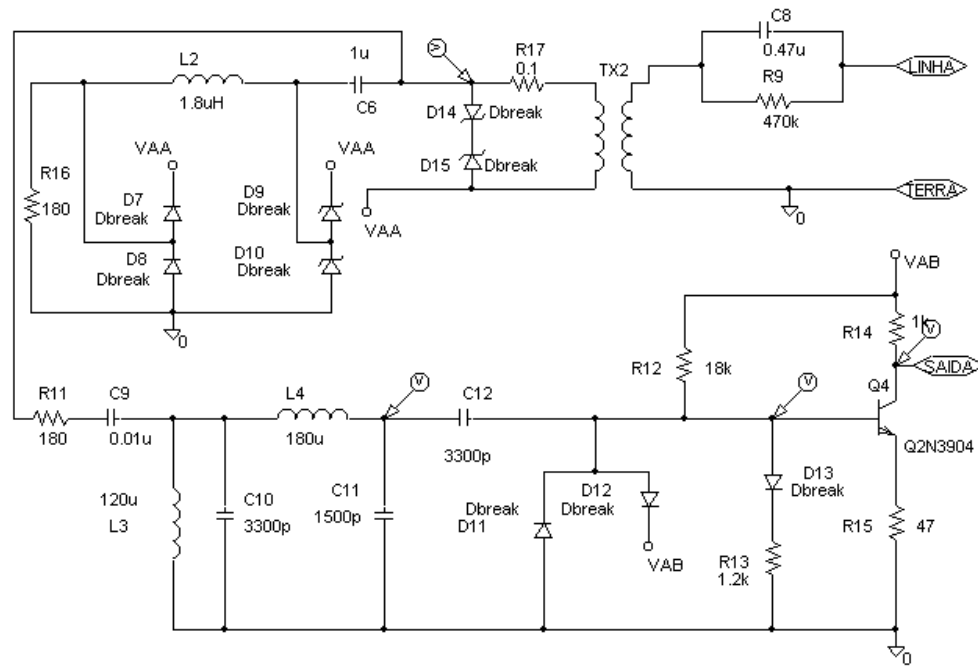


Figura 5.27 – Diagrama de simulação do circuito receptor (RX)

Os resultados da simulação podem ser vistos a seguir. A Figura 5.28 mostra a propagação do *chirp* na rede elétrica, e a Figura 5.29 mostra um detalhe da Figura 5.28.

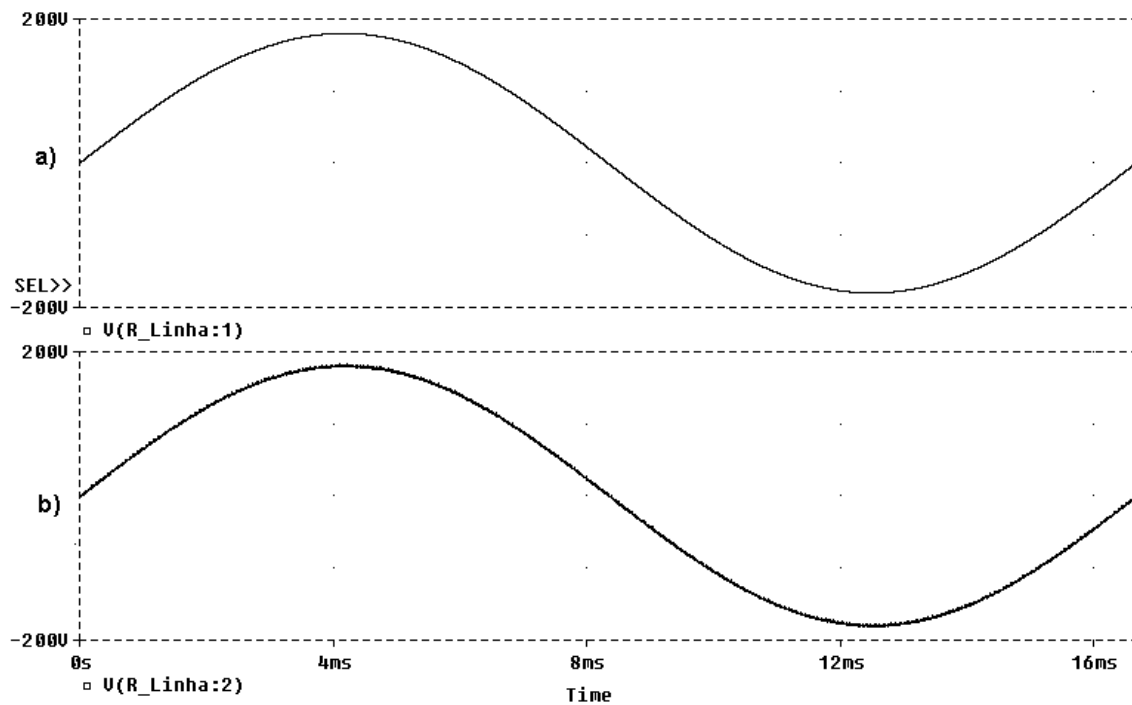


Figura 5.28 – Rede elétrica: a) sem sinal do DLC-P300 b) com sinal do DLC-P300

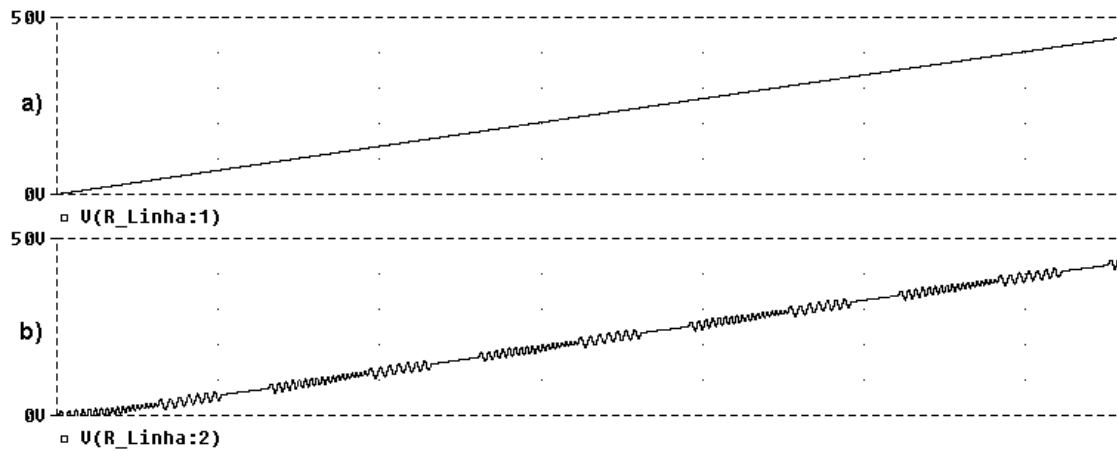


Figura 5.29 – Detalhe da Figura 5.28.

Como pôde ser visto na Figura 5.29, o DLC-P300 injeta harmônicas de alta frequência, ou seja, os sinais do transmissor são somados à rede de energia de 60 Hz.

A Figura 5.30 mostra os sinais obtidos no bloco TX (transmissor).

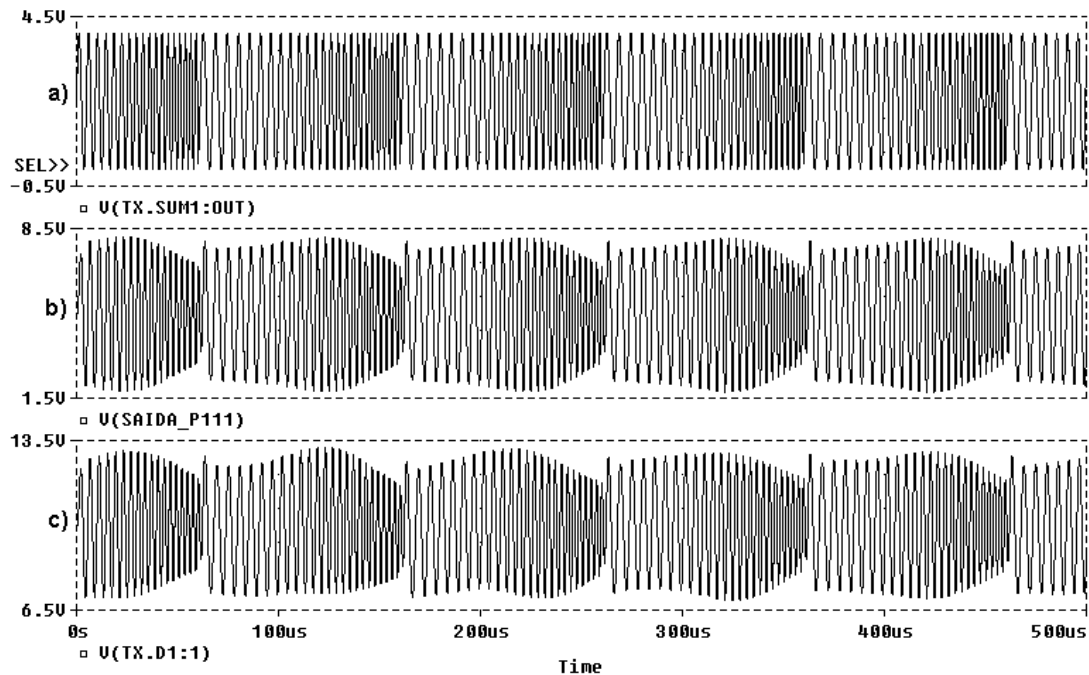


Figura 5.30 – Sinais no bloco TX: a) saída do P300 (pino SO); b) saída do P111 (pino TXO); c) entrada do transformador de acoplamento.

A Figura 5.31 mostra os sinais no bloco RX (receptor). Um detalhe é que o SSP P300 decodifica o sinal não como um *chirp*, mas sim como uma série de pulsos gerados a partir da saturação e corte do transistor que finaliza o circuito de entrada, como mostrado na Figura 5.32. A saturação e o corte acontecem em função da alta impedância de descarga do

capacitor C_{17} da Figura 5.27, provocada pelo diodo D_{13} . Isso faz com que o transistor esteja sempre em saturação (tensão da base negativa) ou corte (tensão da base positiva).

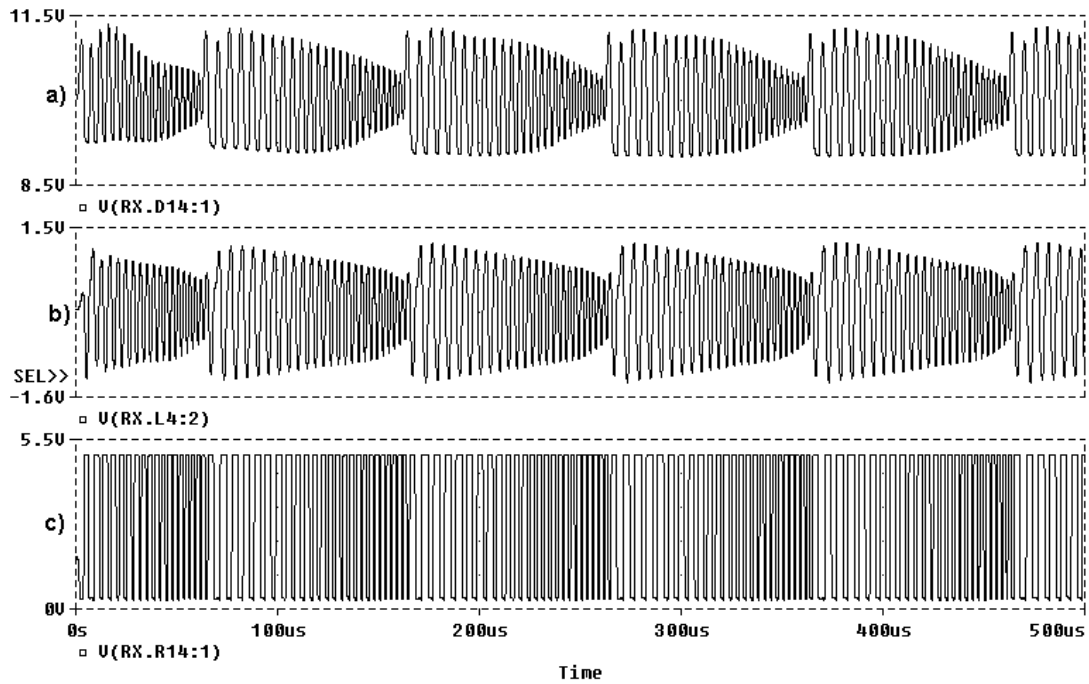


Figura 5.31 – Sinais do bloco RX: a) sinal na entrada do filtro de condicionamento de entrada; b) sinal na saída do filtro de entrada; c) entrada do P300 (pino SI).

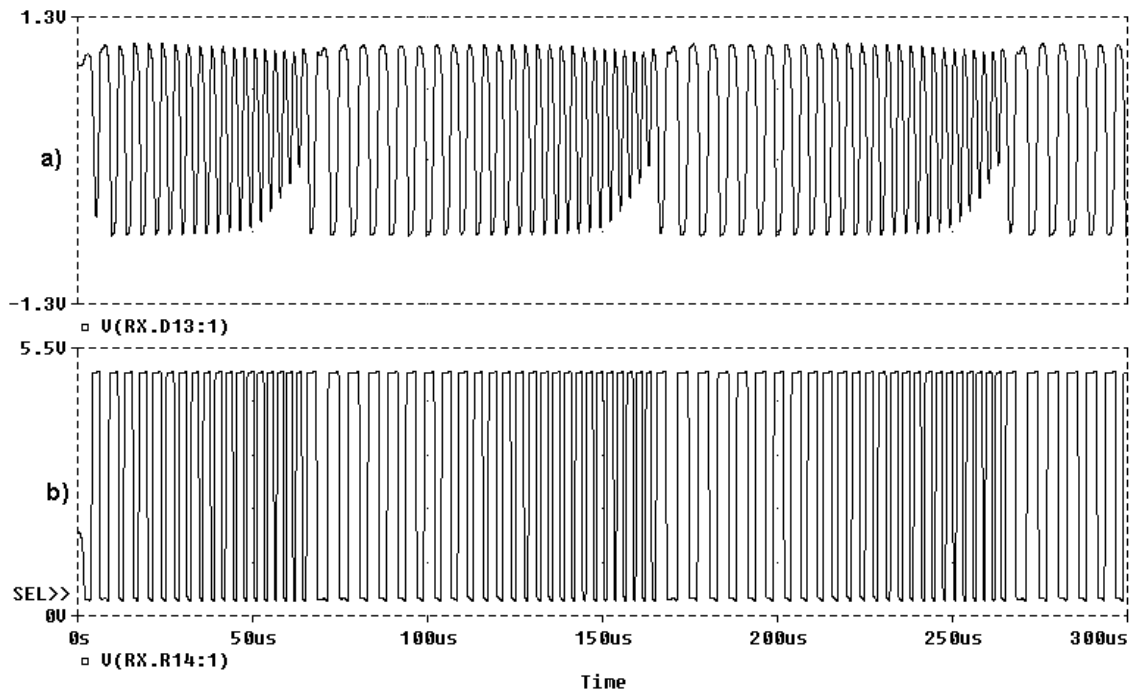


Figura 5.32 – Sinais no *buffer* de entrada: a) Sinal na base do transistor Q_4 ; b) Sinal na entrada do P300 (pino SI).

5.6.2. Medição de sinais no protótipo do DLC-P300

A seguir serão mostrados os sinais do protótipo do DLC-P300, tanto em dispositivos transmissores quanto receptores.

Os mesmos sinais que foram mostrados na simulação serão mostrados a seguir. As formas de onda foram gravadas por um osciloscópio digital, e exportadas para o programa MATLAB.

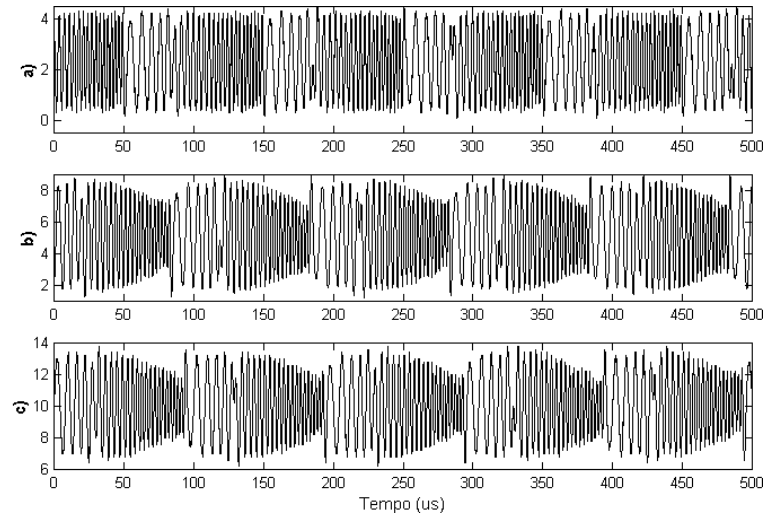


Figura 5.33 – Sinal em um dispositivo DLC-P300 no modo de transmissão (em volts): a) saída do P300 (pino SO); b) saída do P111 (pino TXO); c) entrada do transformador de acoplamento. Taxa de aquisição: 10 MHz.

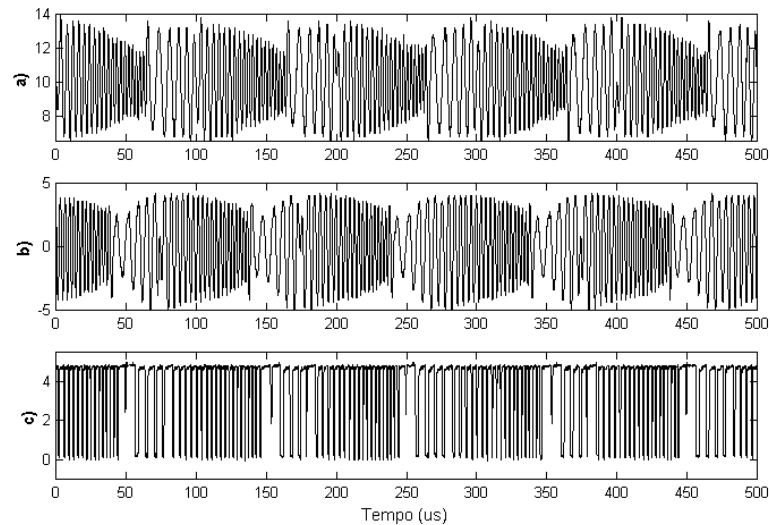


Figura 5.34 – Sinal em um dispositivo DLC-P300 no modo de recepção (em volts): a) sinal na entrada do filtro de condicionamento de entrada; b) sinal na saída do filtro de entrada; c) entrada do P300 (pino SI). Taxa de aquisição: 10 MHz.

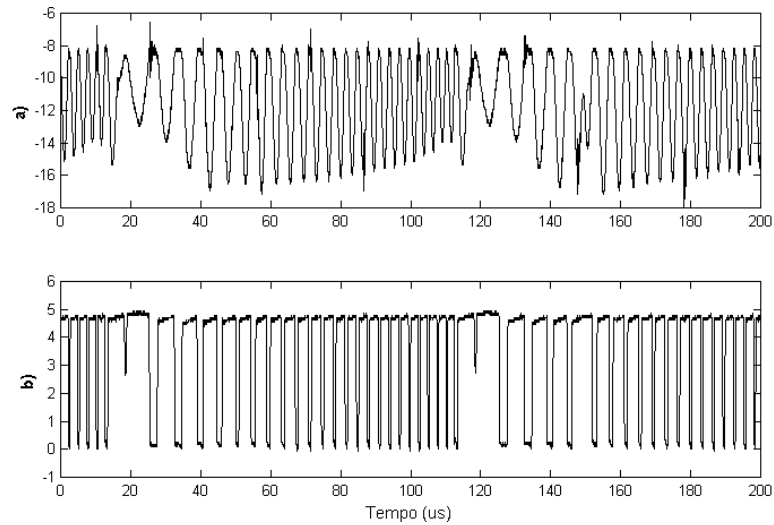


Figura 5.35 – Sinais no *buffer* de entrada (em volts): a) Sinal na base do transistor Q₄; b) Sinal na entrada do P300 (pino SI). Taxa de aquisição: 25 MHz.

5.7. Conclusões

Os objetivos deste capítulo foram mostrar as características construtivas do modem DLC-P300. Através das simulações e análises dos circuitos foi possível adaptar os parâmetros para o funcionamento ótimo do projeto, bem como dominar a tecnologia já existente. Conseguiu-se também o projeto eficiente de um software para o DLC-P300 (*kernel*) que pode ser executado em micro-controladores de baixo custo. A partir da construção do protótipo deste modem adquiriu-se conhecimento suficiente para a realização da etapa seguinte: identificar e propor algoritmos de processamento digital de sinais para a demodulação dos sinais SSC em um pacote *CEBus*.

6. Proposição de Algoritmos de Transmissão e Recepção de *Spread Spectrum Carrier* Baseados em Técnicas DSP.

6.1. Introdução

Como pôde ser visto no Capítulo 5, o integrado SSC P300 desempenha, dentre outras funções, a codificação e decodificação dos símbolos do pacote *CEBus* que trafegam na rede elétrica. Neste capítulo, serão apresentadas propostas de algoritmos (baseados em técnicas de processamento digital de sinais – DSP) que permitam a substituição do SSC P300 por um processador DSP. Como os algoritmos de codificação e decodificação do integrado SSC P300 não são conhecidos, serão examinadas as características intrínsecas ao *chirp* que serve de símbolo-base para o pacote *CEBus*, bem como as informações técnicas procedentes dos manuais técnicos do integrado P300, para coletar informações suficientes para a implementação dos algoritmos em processadores DSP. Resultados de simulação dos algoritmos também serão mostrados.

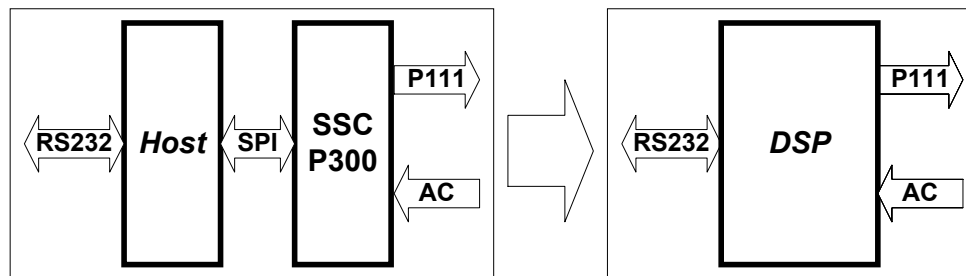


Figura 6.1 – Substituição do SSC P300 e do *host* por um processador DSP.

6.2. Vantagens da substituição

As vantagens da implementação de algoritmos baseados em técnicas de DSP em processadores DSP com o objetivo de substituir o IC SSC P300 são citadas abaixo:

- Permite maior flexibilidade no controle e gerenciamento do protocolo *CEBus*;
- Controle sobre os parâmetros dos algoritmos de codificação e decodificação;
- A substituição do P300 por um processador DSP implica na eliminação do microcontrolador (*host*) do sistema e da necessidade da comunicação entre P300 e *host* por SPI (Figura 6.1);
- Possibilidade de *upgrade* dos algoritmos;

- e) Redução do custo no desenvolvimento de sistemas baseados no protocolo *CEBus* em até 50% (componentes básicos), como mostrado na Tabela 6.1;
- f) Facilidade na integração de novas tecnologias de comunicação (USB, ISDN, Ethernet) entre os dispositivos *CEBus* e os sistemas de controle (PC, PLC, etc.), como mostrado na Figura 6.2.

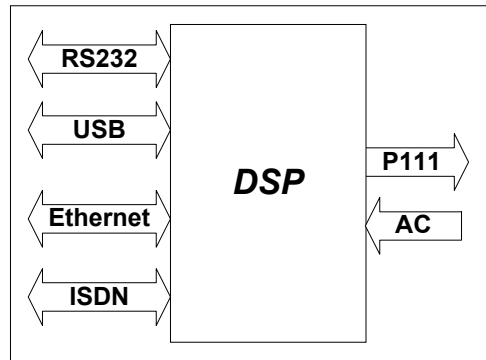


Figura 6.2 – Integração entre o dispositivo *CEBus* e outras tecnologias de comunicação de dados.

Tabela 6.1 – Preços (em US\$, sem carga tributária) dos componentes base.

IC	Preço (US\$)
SSC P300 (Intellon)	20.00
AT89S8252 (Atmel)	10.00
DSP (Texas - TMS320C2407)	15.00

6.3. Princípios básicos dos algoritmos de recepção de sinais digitais

6.3.1. Transmissão de símbolos

Um sistema de transmissão de dados digital utiliza símbolos para representar as informações a serem transmitidas [9]. Uma vez que os símbolos estejam transformados em ondas elétricas, o sinal já pode ser transmitido pelo canal de comunicação (par trançado, ar, etc.). Um sistema de comunicação de dados digital pode usar tantos símbolos quanto necessários, logo, o sinal a ser transmitido, $s_i(t)$ ($0 \leq t \leq T$, $i=1,2,...,N_s$), pode ser escrito como uma seleção dos símbolos a serem transmitidos, como mostra a equação (6.1):

$$s_i(t) = \begin{cases} s_1(t), & 0 \leq t \leq T & \text{para o símbolo } S_1 \\ s_2(t), & 0 \leq t \leq T & \text{para o símbolo } S_2 \\ \vdots & \\ s_{N_s}(t), & 0 \leq t \leq T & \text{para o símbolo } S_{N_s} \end{cases} \quad (6.1)$$

onde N_s é o número de símbolos a serem usados, e T é o tempo de duração de um símbolo. Para um sistema de comunicação digital que utilize dois símbolos ($N_s=2$), o sinal a ser transmitido é:

$$s_i(t) = \begin{cases} s_1(t), & 0 \leq t \leq T & \text{para o símbolo } S_1 \\ s_2(t), & 0 \leq t \leq T & \text{para o símbolo } S_2 \end{cases} \quad (6.2)$$

O sinal recebido por um receptor pode ser escrito como:

$$r(t) = s_i(t) + \omega(t) \quad i = 1, 2; \quad 0 \leq t \leq T \quad (6.3)$$

onde $\omega(t)$ é um ruído aditivo branco com média zero (AWGN) [9]. Se o sinal $r(t)$ é amostrado com uma taxa F_s , pode-se dizer que:

$$r(n) = s_i(n) + \omega(n) \quad i = 1, 2; \quad 0 \leq n \leq N-1 \quad (6.4)$$

O processo de detecção pode, então, ser dividido em duas partes: a primeira parte consiste na redução da forma de onda, $r(n)$, a um único número, $z(N-1)$. A segunda é, a partir do número obtido, obter uma estimativa confiável para estabelecer qual símbolo foi transmitido.

6.3.2. Matched Filter e Filtro de Correlação

O cálculo do parâmetro $z(N)$ pode ser feito de forma eficiente com a utilização de dois métodos:

- Utilizando um filtro digital chamado “*matched filter*” (do inglês *matched*, que significa combinado);
- Utilizando uma estrutura de cálculo chamada de filtro de correlação, que é a realização de um caso específico do “*matched filter*”.

Seja o parâmetro $z(N-1)$ composto pela adição de duas componentes, $a_i(N-1)$ e $\sigma(N-1)$, sendo $a_i(N-1)$ a contribuição do símbolo recebido para o fator $z_i(N-1)$, e $\sigma(N-1)$ a

contribuição (destrutiva) do ruído ao parâmetro $z_i(N-1)$, como mostra a equação (6.5). Considerando a estrutura da Figura 6.3, que mostra o cálculo de $z_i(N-1)$ utilizando-se o “*matched filter*”, pode-se provar que a maximização da relação das contribuições de $z_i(N-1)$, mostrada em (6.6), ocorre quando $H_i(z)$ apresentar a resposta ao impulso apresentada na equação (6.7) [9]. A Figura 6.4 mostra as características do “*matched filter*”.

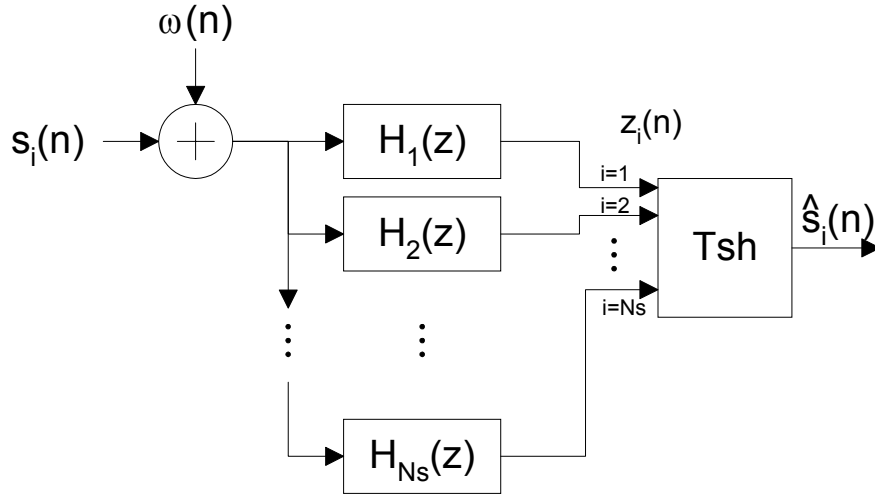


Figura 6.3 – *Matched filter*

$$z_i(N-1) = a_i(N-1) + \sigma(N-1) \quad (6.5)$$

$$R\left(\frac{a}{\sigma}\right) = \frac{a_i(N-1)}{\sigma(N-1)} \quad (6.6)$$

$$h_i(n) = s_i(N-1-n) \quad (6.7)$$

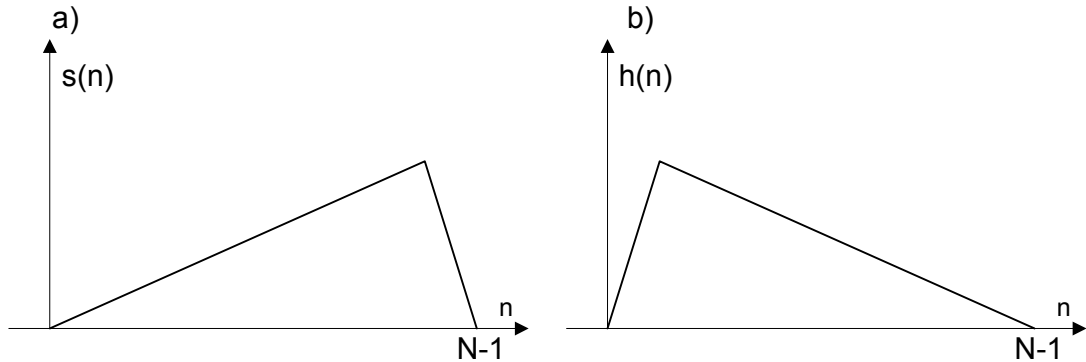


Figura 6.4 – Características do “*matched filter*” a) símbolo a ser transmitido b) resposta ao impulso do “*matched filter*”.

Uma das formas para a realização da função de transferência $H_i(z)$ é a utilização de um filtro digital FIR (*Finite Impulse Response* – Resposta ao Impulso Finita), cuja equação pode ser escrita em função da resposta ao impulso $h_i(n)$:

$$H_i(z) = \sum_{n=0}^{N-1} h_i(n) \cdot z^{-n} \quad (6.8)$$

A utilização do “*matched filter*” para a detecção de N_s símbolos implica na utilização de N_s filtros $H_i(z)$. Isso faz com que o cálculo de todos os parâmetros $z_i(n)$ tenha um esforço computacional excessivo, como mostra a equação (6.9), onde EC_{MF} é o esforço computacional de N_s filtros FIR com N *taps* (ou ordem $N-1$).

$$EC_{MF} = N_s \cdot (N \text{ multiplicações} + (N-1) \text{ adições}) \quad (6.9)$$

Uma forma de evitar o número excessivo de operações é a utilização do filtro de correlação. Para isso, seja o esquema da Figura 6.5, mostrando um filtro de correlação para $N_s=2$. O símbolo recebido é multiplicado amostra a amostra por cada símbolo que faz parte do sistema de comunicação, que dá origem a N_s parâmetros $z_i(N-1)$. A comparação e avaliação destes parâmetros é que decide qual o símbolo foi transmitido.

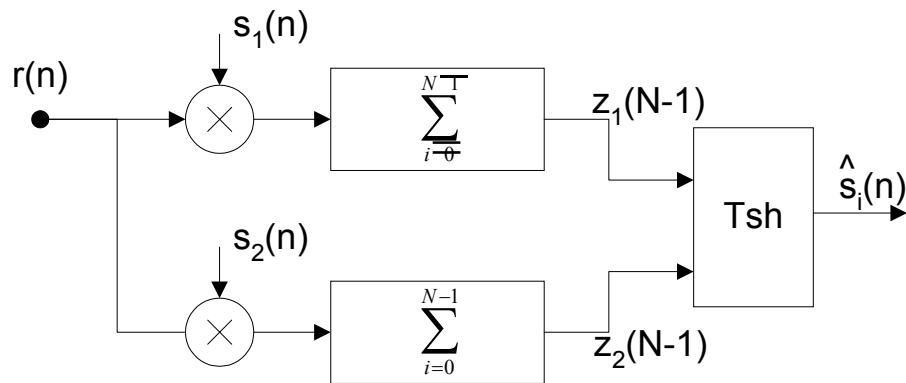


Figura 6.5 – Filtro de Correlação para $N_s=2$.

Porém, para que $z_i(N-1)$ seja corretamente calculado, $r(n)$ e $s_i(n)$ devem estar sincronizados, ou seja, é necessários algoritmos que estimem o atraso sofrido pelo símbolo $r(n)$ na recepção, que depende intimamente da constituição de cada símbolo.

Depois do cálculo de $z(N-1)$, é necessário estimar qual símbolo foi recebido. Para isso, realiza-se um teste, chamado de teste estatístico, aonde se compara o valor de $z_i(n)$ com os valores calculados com as equações (2.28) e (2.29) para cada símbolo do sistema

de comunicação. A equação (2.29) define o limite de *threshold* que serve para avaliar se o símbolo foi corretamente detectado.

O esforço computacional do filtro de correlação para N_s símbolos é:

$$EC_{FC} = N_s \cdot (1 \text{ multiplicação} + 1 \text{ adição}) \quad (6.10)$$

Uma distinção importante a ser feita é que o filtro de correlação e o “*matched filter*” só produzem a mesma resposta para o valor de $n=N-1$. A saída do “*matched filter*” pode ser vista como um conjunto das saídas do filtro de correlação, cada uma delas calculada a cada amostragem do sinal. Essa similaridade de funções faz com que muitas vezes o “*matched filter*” seja chamado às vezes de filtro de correlação.

6.3.3. Transmissão de Sinais Bipolares

Seja um sistema de comunicação digital que utilize dois símbolos $s_1(t)$ e $s_2(t)$ ($N_s=2$), tal que a relação entre os símbolos transmitidos seja:

$$s_2(t) = -s_1(t) \quad (6.11)$$

A equação (6.11) representa um sistema de transmissão com dois símbolos que estão defasados entre si de 180° , ou seja, os sinais tem polaridades opostas, chamados de sinais bipolares. Logo, algumas simplificações podem ser efetuadas nos algoritmos de recepção que usam os filtros de correlação e o “*matched filter*”:

- a) Permite a utilização de apenas um “*matched filter*”;
- b) Permite que seja utilizado somente um multiplicador no filtro de correlação;

Isso vem do fato de que a função de correlação cruzada descrita em (2.19) ter a seguinte propriedade, quando usada a equação (6.11):

$$Xs_1s_2 = \sum_{n=0}^{N-1} s_1(n) \cdot s_2(n) = \sum_{n=0}^{N-1} s_1(n) \cdot -s_1(n) = -\sum_{n=0}^{N-1} s_1(n) \cdot s_1(n) = -Xs_1s_1 \quad (6.12)$$

onde Xs_1s_2 é a correlação cruzada entre os sinais $s_1(n)$ e $s_2(n)$, e Xs_1s_1 é a autocorrelação de $s_1(n)$.

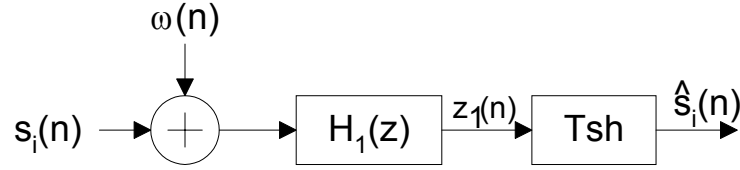


Figura 6.6 – *Matched filter* considerando (6.11) e (6.12).

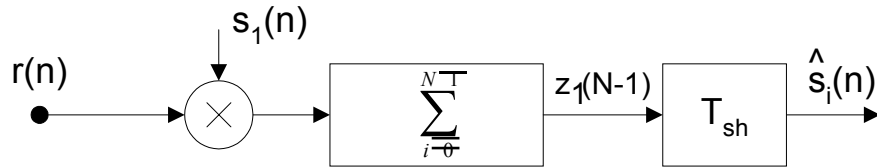


Figura 6.7 – Filtro de correlação considerando (6.11) e (6.12).

Logo, o processo de detecção consiste comparar o valor de saída $z_1(N-1)$ com o valor de Xs_1s_1 definido em (6.12). Valores negativos com módulo superior ao expresso em (2.28) indicam que o símbolo S_2 foi detectado, enquanto que valores positivos indicam a detecção do símbolo S_1 . Com as simplificações nos processos de detecção, a complexidade computacional é reduzida, e podem ser expressas como:

$$EC_{MF} = N \text{ multiplicações} + (N - 1) \text{ adições} \quad (6.13)$$

$$EC_{FC} = 1 \text{ multiplicação} + 1 \text{ adição} \quad (6.14)$$

Nos dois casos, o esforço computacional é reduzido à metade.

6.4. Modelagem do canal de comunicação em sistemas SSC

6.4.1. Atenuação em redes elétricas

Seja o sistema de comunicação mostrado na Figura 2.1, que representa a transformação de um sinal de comunicação devido à sua propagação no meio, chamado de canal de comunicação. Tal transformação advém de inúmeros fatores, cada qual transformando o sinal de uma determinada maneira. Alguns desses fatores são: Atenuação; Atraso; Múltiplas reflexões; Ruídos; Interferências (intencionais ou não), etc.

Um mesmo fator pode modificar um sinal de maneiras distintas, devido às variações (às vezes aleatórias) das contribuições dos agentes causadores do fator. Logo, modelos

matemáticos se fazem necessários para expressar os fatores e os agentes responsáveis pelas respectivas contribuições. Porém, nem sempre o canal de comunicação pode ser perfeitamente modelado. Modelos mais simples devem então ser considerados para que possa haver a avaliação da performance do sistema de comunicação, como os níveis de ruído admissíveis no receptor, o alcance máximo proporcionado, a atenuação provocada em diferentes frequências, a robustez do sistema a interferências, etc.

A utilização de redes elétricas como canal de comunicação implica no conhecimento de como os sinais de alta frequência são afetados pelas características da rede elétrica. Características como ruído, atenuação e impedância da rede elétrica são os fatores mais importantes a serem considerados na análise da distorção dos sinais no trajeto entre o transmissor e o receptor [15].

Vários autores já se propuseram a modelar redes elétricas *intrabuilding* para a avaliação da distorção dos sinais de comunicação [15], [29]. As conclusões a que chegaram foram [15]:

- a) As cargas localizadas entre fases diferentes fornecem um caminho de condução do sinal, porém resultam em uma atenuação entre dois pontos;
- b) A atenuação dos sinais de comunicação é particularmente alta quando o caminho de condução do sinal entre o transmissor e o receptor é longo e contém diversas cargas espalhadas;
- c) A variação da carga causa mudanças na característica em frequência entre dois pontos;
- d) Para altas frequências, podem ocorrer efeitos de linha de transmissão quando o caminho entre o transmissor e o receptor é longo (Um sinal de 100 KHz tem 3000 m de comprimento de onda, logo, distâncias superiores a 500 m podem ocasionar os efeitos de linha de transmissão);

Devido aos fatores citados acima, a complexidade e a indisponibilidade dos diagramas completos da rede elétrica e de todas as suas cargas instaladas torna virtualmente impossível a obtenção de um modelo generalizado do canal de comunicação [15]. Porém, alguns modelos mais simplificados podem ser obtidos para a avaliação do *link* de comunicação. Neste trabalho serão abordados dois modelos: o canal de transmissão AWGN e o efeito de atenuação em frequência.

6.4.2. Canal AWGN

Um modelo bastante difundido para testar a confiabilidade e a robustez de algum algoritmo de recepção é o canal AWGN (*Additive White Gaussian Noise*), mostrado na Figura 6.8 [9].

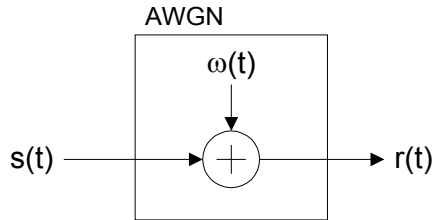


Figura 6.8 – Representação de um canal AWGN

As características de um canal (ou processo) AWGN são mostradas na Figura 6.9. Um canal AWGN tem a característica de “adicionar” um sinal com energia constante que contém todas as frequências do espectro ao sinal a ser transmitido [25].

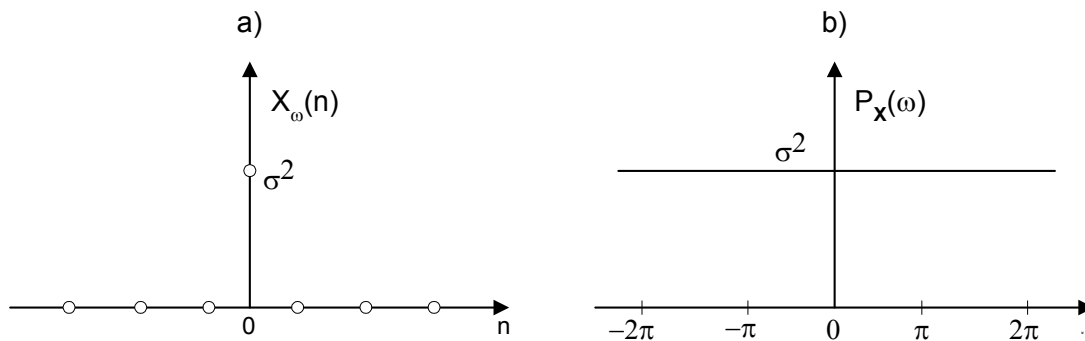


Figura 6.9 – a) Função de autocorrelação de um processo AWGN; b) Espectro de um processo AWGN.

Na Figura 6.9a, temos a função de autocorrelação de $\omega(n)$, onde pode-se observar que o processo AWGN só tem correlação não-nula para valores da sequência de correlação iguais a zero. A energia do processo AWGN (Figura 6.9b) é dada pela variância (quadrado do desvio padrão de $\omega(n)$, $0 \leq n \leq N-1$) de $\omega(n)$, denotada por σ_{ω}^2 .

6.4.3. Atenuação em frequência

Ao propagar-se pelo canal, o sinal sofre mudanças tanto de amplitude como de fase, ocasionadas pela absorção da energia do sinal pelo canal. Tais mudanças podem ser representadas por um operador $\mathbf{C}(\)$, aplicado ao sinal de entrada (Figura 6.10). Em uma rede elétrica do tipo *intrabuilding*, o operador é função da quantidade de cargas ligadas na linha, da geometria e disposição dos condutores, do tipo das cargas instaladas, do tipo de

ligação da carga na rede elétrica, etc. Como tais fatores não são constantes nem concomitantes, nem sempre uma relação matemática pode ser encontrada entre os fatores e o operador $C(\cdot)$.

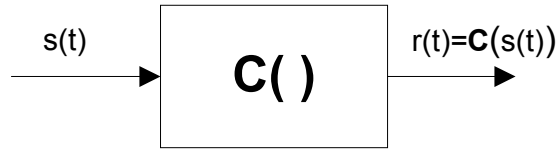


Figura 6.10 – Operador $C(\cdot)$ representando as alterações sofridas pelo sinal em um canal de comunicação.

Uma solução é a obtenção de parâmetros da atenuação em sistemas elétricos que possam ser considerados representativos. Isto pode ser feito através de medidas de atenuação de sinais com frequência variável, injetados em pontos estratégicos da rede elétrica, como mostrado na Figura 6.11.

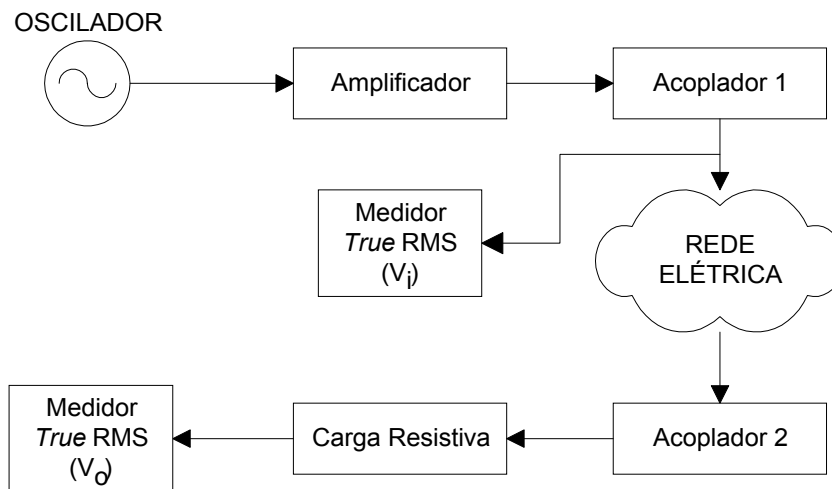


Figura 6.11 – Experimento para obtenção experimental da atenuação em redes elétricas.

Logo, o valor em módulo da atenuação da rede elétrica para uma frequência de entrada f_i , em função de V_i e V_o (valor RMS) é:

$$A(f_i)_{dB} = 20 \cdot \log_{10} \left(\frac{\overline{V_o}}{\overline{V_i}} \right) \quad (6.15)$$

aonde a barra indica que é utilizada uma média dos sinais injetados na rede, procedimento este necessário para anular o ruído produzido pela rede. Para a avaliação da rede elétrica como um canal de comunicação, Chen *et al* [15] produziram várias curvas de atenuação em frequência de cinco tipos de redes distintas:

- a) Complexo industrial com alimentação trifásica e cargas indutivas e não-lineares (motores, microcomputadores, equipamentos eletrônicos, etc);
- b) Complexo hospitalar (540 leitos);
- c) Prédio residencial com 4 andares;
- d) Prédio residencial com 12 andares;
- e) Residência em bairro de classe média (essa e outras 20 residências são atendidas por um transformador).

Vale salientar que essas medições foram efetuadas nos EUA, logo, os casos c), d) e e) citados anteriormente são alimentados com um transformador “*split phase*”, muito comum em sistemas de distribuição residencial americanos [3]. Os dados foram coletados em 16 frequências distintas, entre 20 KHz e 240 KHz. Os dados obtidos são mostrados nas figuras a seguir:

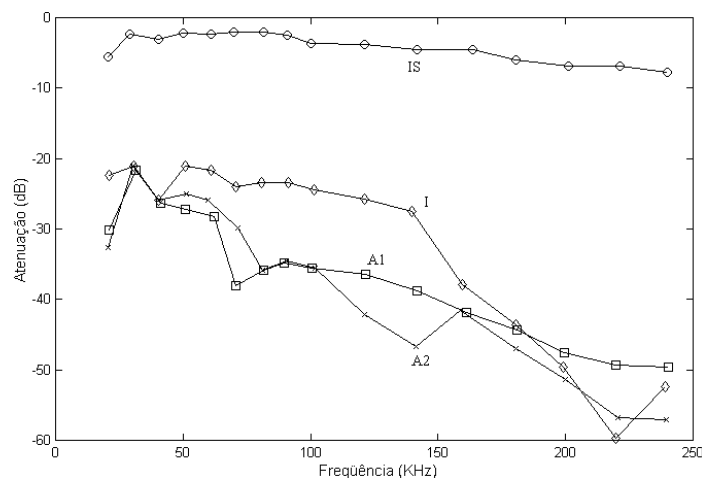


Figura 6.12 – Gráfico de atenuação x frequência, complexo industrial, período diurno.

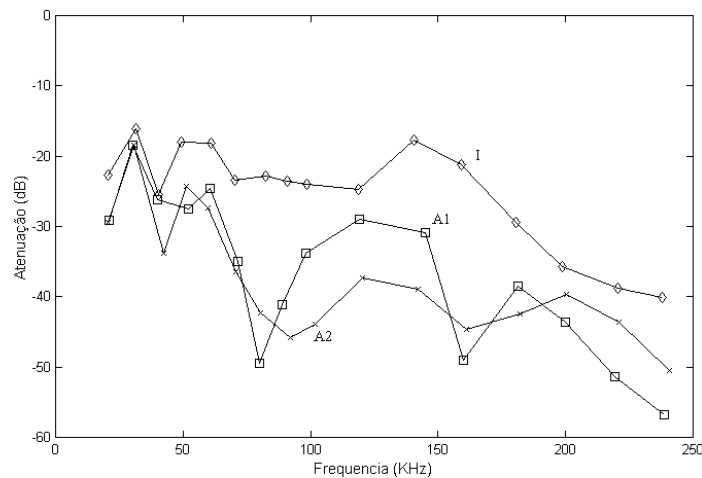


Figura 6.13 – Gráfico de atenuação x frequência, complexo industrial, período noturno.

O gráfico da Figura 6.12 indica a atenuação do sinal entre dois pontos com distâncias distintas, no período diurno, que são marcadas nos gráficos como IS, I, A1 e A2. Os termos significam o seguinte:

- a) IS: mesma fase, pequenas distâncias (aproximadamente 10 m);
- b) I: mesma fase, grandes distâncias (tipicamente acima de 20m, podendo chegar a 300 m);
- c) A1: fases diferentes (entre fase “1” e fase “3”) e grandes distâncias;
- d) A2: fases diferentes (entre fase “1” e fase “3”) e grandes distâncias

O gráfico da Figura 6.13 indica a atenuação do sinal dos mesmos pontos do gráfico da Figura 6.12, só que no período noturno. Nota-se que há uma flutuação maior no nível de atenuação nas medidas de A1 e A2. Tal diferença entre as curvas se deve ao fato que à noite há uma diminuição das cargas ligadas entre fases.

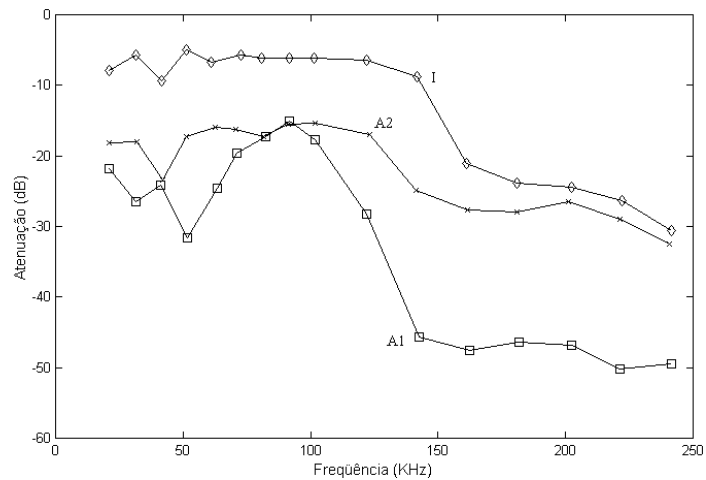


Figura 6.14 – Gráfico de atenuação em frequência, complexo hospitalar.

O gráfico da Figura 6.14 mostra a atenuação em frequência no período diurno de um complexo hospitalar. De um modo geral, as cargas relacionadas a tal complexo são praticamente constantes durante as 24 horas do dia, logo, as redes elétricas são menos sujeitas às variações de impedância.

O gráfico da Figura 6.15 mostra a medida da atenuação em prédios residenciais, efetuadas entre o período de 18:00 h e 20:00 h, e com as seguintes denominações:

- a) IL: mesma fase, prédio de 4 andares, grandes distâncias;
- b) IH: mesma fase, prédio de 12 andares, grandes distâncias;
- c) ISH: mesma fase, prédio de 12 andares, pequenas distâncias;
- d) OL: fases opostas, prédio de 4 andares, grandes distâncias;
- e) OH: fases opostas, prédio de 12 andares, grandes distâncias.

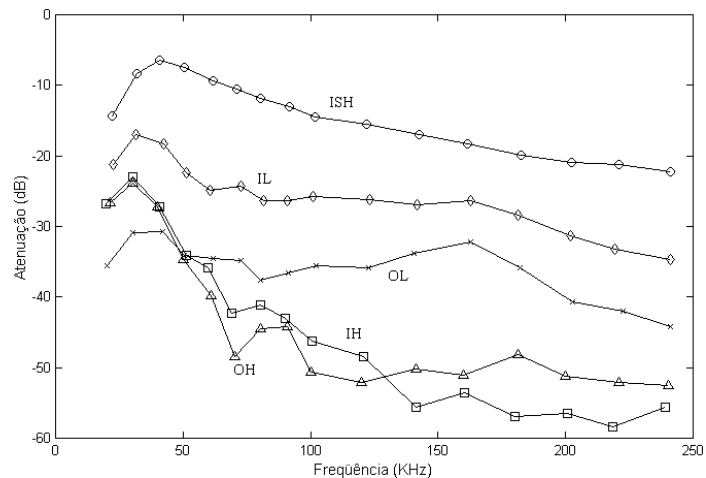


Figura 6.15 – Gráfico de atenuação em frequência em prédios residenciais.

Como pode ser observado na Figura 6.15, o prédio de 4 andares mostra uma menor atenuação em frequência do que o prédio de 12 andares, fato que se deve ao menor número de cargas instaladas no prédio de 4 andares. Porém, nem sempre a atenuação entre fases opostas é menor que a atenuação na mesma fase, como mostra as curvas IH e OH da Figura 6.15.

Já a Figura 6.16 mostra a atenuação em uma residência situada em um bairro de classe média, sem cargas, ao contrário da Figura 6.17, que mostra a atenuação do sinal da residência com cargas específicas. As denominações para os gráficos da Figura 6.16 e da Figura 6.17 são mostradas a seguir:

- a) I: mesma fase, sem carga;
- b) O: fases opostas, sem carga;
- c) ITV: mesma fase, com uma televisão como carga;
- d) IEK: mesma fase, com uma panela elétrica como carga;
- e) OCD: fases opostas, com uma secadora de roupas como carga;
- f) OER: fases opostas, com um aquecedor elétrico como carga.

As curvas I e O da Figura 6.16 foram medidas às 4:00h, quando virtualmente não existem cargas ligadas na residência. As curvas OER e OCD são as mesmas da Figura 6.17, colocadas para comparação.

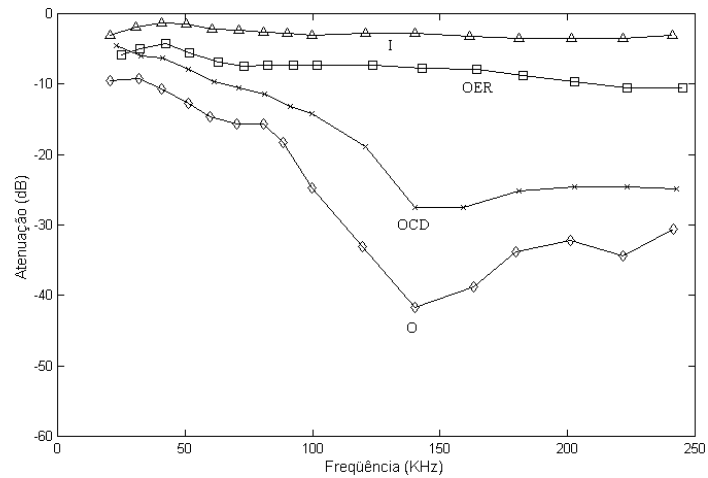


Figura 6.16 – Gráfico de atenuação em frequência de uma residência em bairro de classe média, sem cargas específicas.

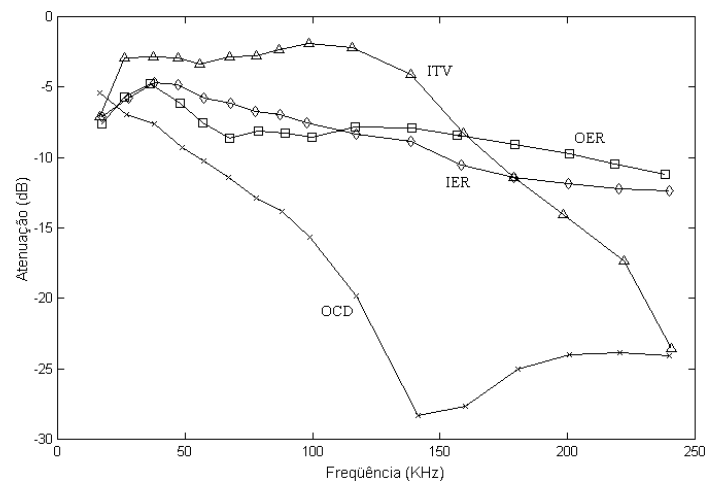


Figura 6.17 – Gráfico de atenuação em frequência de uma residência em bairro de classe média, com cargas específicas.

Para a mensuração do efeito que as cargas domésticas causam na atenuação da rede elétrica residencial, o seguinte procedimento foi utilizado: na situação da curva I (sem cargas ligadas), foram introduzidas na rede as cargas denotadas por TV, CD e ER, separadamente, e em cada carga, foram medidos os níveis de atenuação da rede. Como pode ser observado, a inserção das cargas OER e OCD reduziu o nível de atenuação da rede, por fornecer um caminho de baixa impedância para o sinal. Já a inserção das cargas ITV e IER provocaram o aumento da atenuação, pois são cargas significativas (a carga ER é uma resistência pura de 10Ω , enquanto que a carga TV tem um comportamento puramente capacitivo).

As conclusões que podem ser tomadas levando em conta os gráficos anteriores são [15]:

- a) Exceto por casos aonde o caminho em que o sinal percorre é curto, a atenuação em redes elétricas *intrabuilding* é tipicamente maior que 20 dB, a não ser que o transmissor e o receptor se situem na mesma fase;
- b) Quando o transmissor e o receptor estão na mesma fase, a atenuação tende a ser menor do que se estivessem em fases diferentes. Entretanto, as diferenças entre as atenuações na mesma fase e de fases diferentes não são tão significativas. Podem ocorrer casos em que a atenuação na mesma fase seja maior que em fases diferentes;
- c) A atenuação do sinal tende a crescer com a frequência, embora tal crescimento não seja sempre monotônico;
- d) A variação das cargas em uma rede elétrica afeta a atenuação da rede elétrica. Como o perfil das cargas varia com o tempo, a atenuação pode ser considerada variante no tempo sobre qualquer frequência. Variações de 20 dB não são difíceis de ocorrer;
- e) Não é possível encontrar um modelo matemático geral para a modelagem do canal da rede elétrica *intrabuilding*.

6.4.4. Modelagem da atenuação em redes elétricas

Como visto no item anterior, as redes elétricas do tipo *intrabuilding* têm grandes variações nas curvas de atenuação em função da frequência, o que inviabiliza a obtenção de um modelo generalizado. Porém, para testar algoritmos de recepção, é necessária uma idéia de como o canal de comunicação se comporta. Uma aproximação particularmente interessante é a obtenção de filtros digitais que possam representar o comportamento do canal de comunicação durante a transmissão de um pacote de dados. Neste caso, o filtro é uma aproximação estacionária do canal de comunicação, ou seja, um retrato do canal em um determinado momento. Para aproximar os filtros desejados, considere a seguinte função discreta $C(z)$, mostrada a seguir:

$$C(z) = \frac{\sum_{i=0}^{L-1} b_i \cdot z^{-i}}{\sum_{i=0}^{M-1} a_i \cdot z^{-i}} \quad (6.16)$$

Os termos a_i e b_i de (6.16) são os coeficientes que definem as características do filtro digital IIR (*Infinite Impulse Response*) [25]. A aproximação consiste em, dados os valores da atenuação do canal e as respectivas frequências, determinar os coeficientes que minimizem o erro entre a resposta em frequência obtida no filtro $C(z)$ e a resposta em frequência desejada do canal. As aproximações foram calculadas utilizando o método de Yule-Walker modificado com estimadores em frequência ARMA (*Auto Regressive Mean Average*) [30], [31]. A implementação da aproximação foi feita no software MATLAB, utilizando a função *yulewalk* (Figura 6.18).

```

>> help yulewalk

YULEWALK Recursive filter design using a least-squares method.
[B,A] = YULEWALK(N,F,M) Finds the N-th order recursive filter
coefficients B and A such that the filter:

      -1      -(n-1)
      b(1) + b(2)z + .... + b(n)z
B(z)  -----
      -1      -(n-1)
      1 + a(1)z + .... + a(n)z
A(z)  -----

matches the magnitude frequency response given by vectors F and M.
Vectors F and M specify the frequency and magnitude breakpoints for
the filter such that PLOT(F,M) would show a plot of the desired
frequency response. The frequencies in F must be between 0.0 and 1.0,
with 1.0 corresponding to half the sample rate. They must be in
increasing order and start with 0.0 and end with 1.0.

See also FIR1, BUTTER, CHEBY1, CHEBY2, ELLIP, FREQZ and FILTER.

>>
  
```

Figura 6.18 – Tela do MATLAB mostrando a estrutura do comando *yulewalk*.

A função *yulewalk* faz a aproximação do filtro no domínio do tempo pelo método dos mínimos quadrados. Os coeficientes do denominador a_i são computados pelas equações de Yule-Walker modificadas, usando os coeficientes de correlação computados pela transformada inversa de Fourier da resposta em frequência desejada. O numerador (e seus coeficientes b_i) é computado por um procedimento de 4 passos. Primeiramente, um polinômio que corresponde a uma decomposição aditiva do quadrado da resposta em frequência é computado. Depois, a resposta em frequência completa correspondendo ao numerador e ao denominador é calculada. Então uma técnica de fatorização espectral é usada para obter a resposta ao impulso do filtro. Finalmente, o numerador é obtido pelo

método de mínimos quadrados que ajusta a curva à resposta ao impulso desejada. Maiores detalhes da aproximação podem ser obtidos em [31]. A Figura 6.19 mostra o procedimento necessário para a estimação de um canal de comunicação.

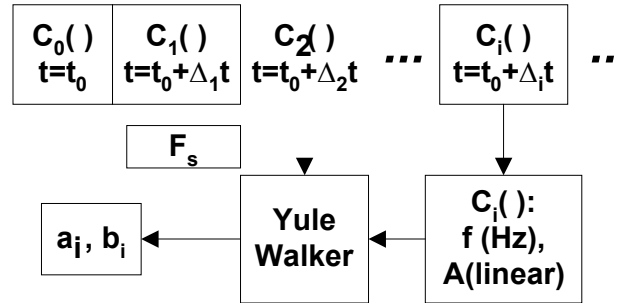


Figura 6.19 – Estimação do canal de comunicação “tirando um retrato” do canal de comunicação.

A aproximação de um canal de comunicação pela função yulewalk exige que seja definida uma frequência de amostragem específica. Isso decorre em função do canal de comunicação, que é analógico em essência. As modelagens aqui obtidas são utilizadas nas simulações dos algoritmos de transmissão e recepção, que são projetados para operar com uma frequência de amostragem de 2 MHz. Logo, as aproximações dos canais de comunicação medidos por Chen [15] não cobrem todas as frequências do sinal SSC (100 KHz a 400 KHz), pois vão de 20 KHz a 240 KHz. Porém, Chen afirma que as variações nos canais de comunicação não podem ser previstas, e que grandes diferenças nas medidas de atenuação podem ser encontradas para uma mesma frequência em momentos diferentes. Logo, as medidas dos canais de Chen servem mais de caráter qualitativo do que quantitativo, e podem ser modificadas no aspecto quantitativo para se obter uma melhor adequação às frequências a serem estudadas. Para isso, considere as seguintes modificações para os canais de Chen:

- a) Sejam A_i e f_i as medidas de atenuação e frequência de Chen, e A_{Ni} e f_{Ni} as novas medidas de atenuação a serem obtidas e posteriormente aproximadas. A relação entre os parâmetros é:

$$\begin{aligned} A_{Ni} &= A_i \\ f_{Ni} &= 2 \cdot f_i \end{aligned} \quad (6.17)$$

Logo, houve uma alteração no eixo das frequências, multiplicando-o por um fator de escala igual a 2. Desta forma, o novo canal cobre todas as frequências do sinal SSC (100 KHz a 400 KHz), pois as medidas são de 40 KHz a 480 KHz.

- b) Sejam A_{Ni} e f_{Ni} as medidas de atenuação de Chen. Um ponto adicional será introduzido, em $f_{Ni} = 1$ MHz com medida de atenuação $A_{Ni} = A_{Ni-1} - 6$ dB. Desta forma, o canal agora pode ser aproximado para uma taxa de amostragem igual à 2MHz, igual à utilizada pelos algoritmos de recepção e transmissão. O fator de subtração de 6 dB foi escolhido para não provocar mudanças bruscas no novo perfil de atenuação do canal.

Os resultados das aproximações podem ser vistos nos gráficos a seguir, aonde são comparados com os resultados desejados:

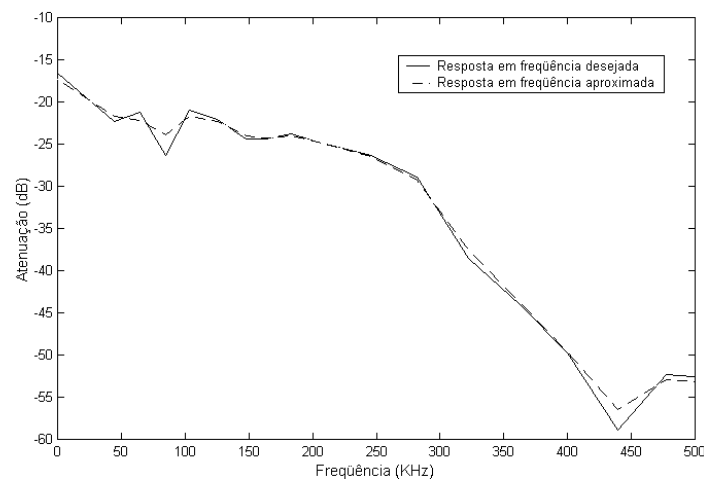


Figura 6.20 – Aproximação em frequência. Caso: Complexo industrial, período diurno, curva de atenuação I.

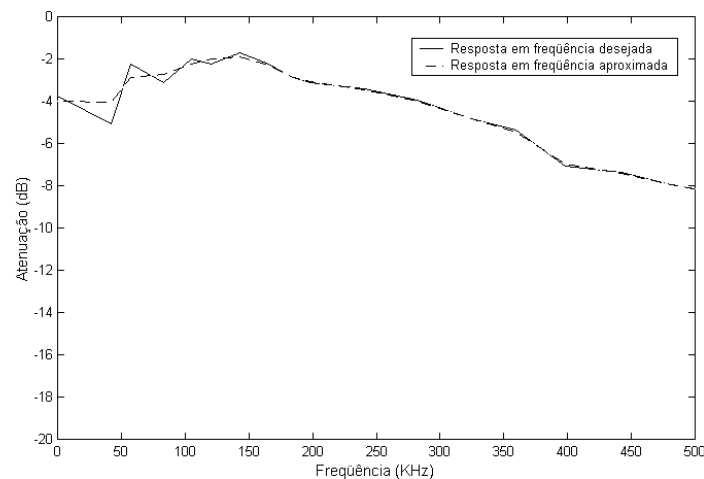


Figura 6.21 – Aproximação em frequência. Caso: Complexo industrial, período diurno, curva de atenuação IS.

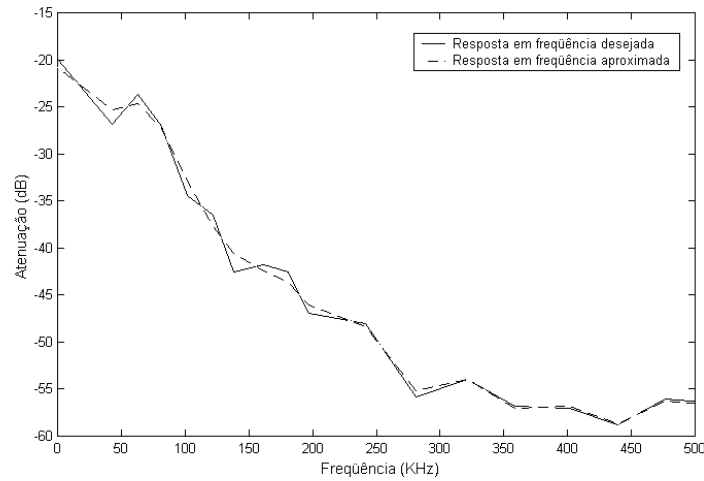


Figura 6.22 – Aproximação em frequência. Caso: Prédio residencial, curva de atenuação IH.

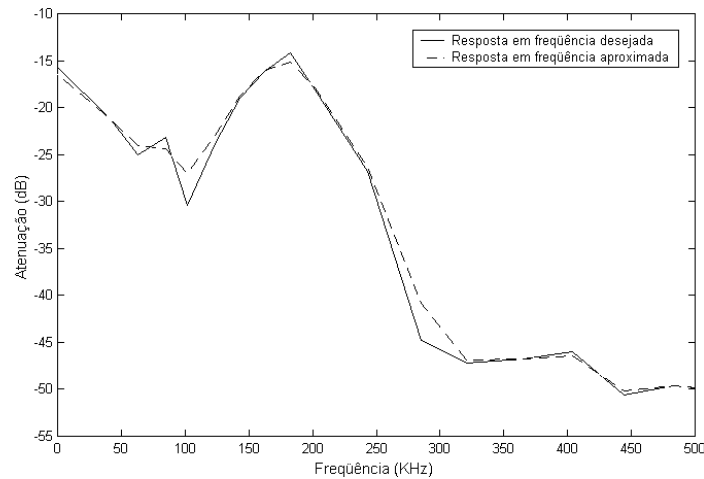


Figura 6.23 – Aproximação em frequência. Caso: Complexo hospitalar, curva de atenuação A2.

Como pode ser visto, as aproximações efetuadas produziram erros, embora o comportamento qualitativo das curvas foi mantido. Como o canal é extremamente variante no tempo, os filtros digitais produzidos pelas aproximações das curvas de atenuação podem ser considerados aproximações bastante exatas dos canais de comunicação aqui apresentados. As curvas de atenuação em frequência dos modelos encontrados são mostradas a seguir. Por dificuldade de representação (os gráficos iriam ficar muito confusos), somente serão apresentados os modelos obtidos, sendo omitida a comparação com as curvas desejadas. Porém, como parâmetro quantitativo, o gráfico da Figura 6.23 mostra a pior aproximação obtida, enquanto que o gráfico da Figura 6.21 mostra a melhor aproximação obtida.

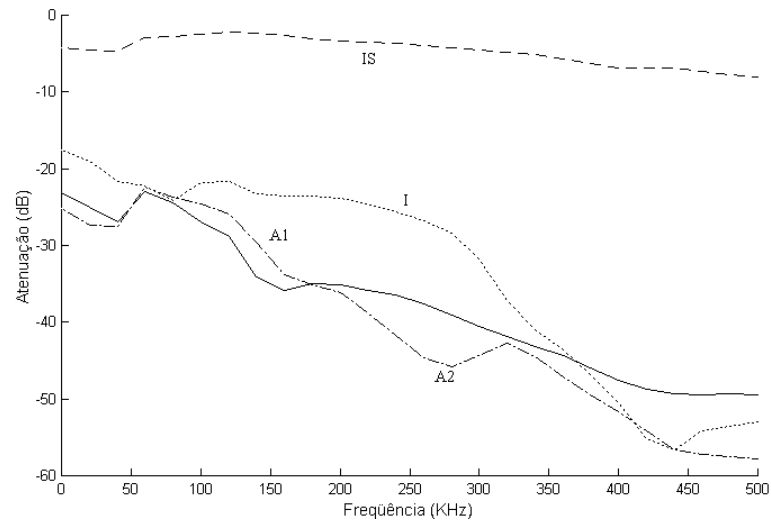


Figura 6.24 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: complexo industrial, período diurno. As convenções são as mesmas utilizadas no gráfico da Figura 6.12. Ordem dos filtros encontrados para os casos IS, I, A1, A2: 20ª ordem.

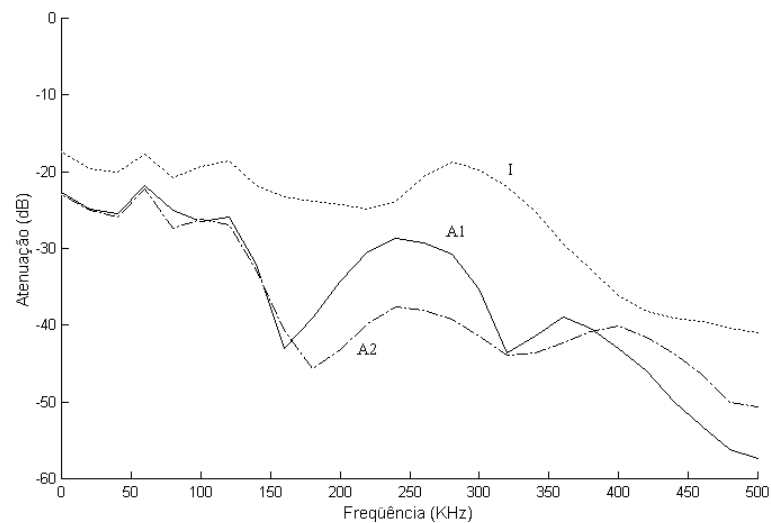


Figura 6.25 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: complexo industrial, período noturno. As convenções são as mesmas utilizadas no gráfico da Figura 6.13. Ordem dos filtros encontrados para os casos A1e A2: 20ª ordem; caso I: 19ª ordem.

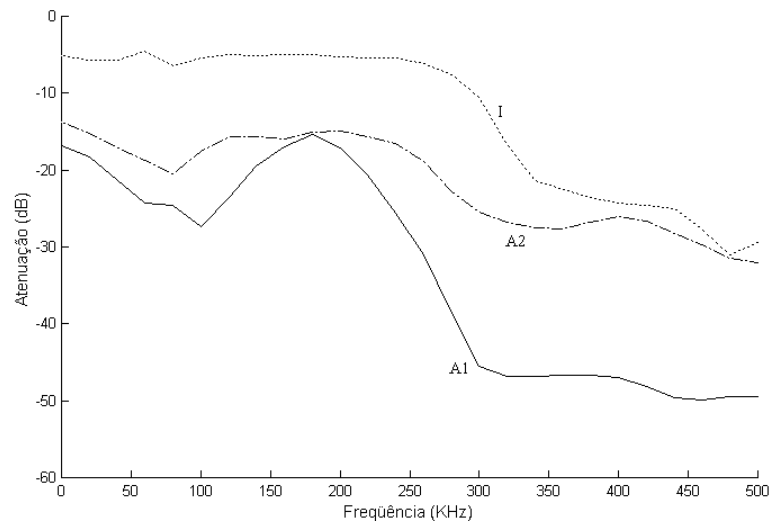


Figura 6.26 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: complexo hospitalar. As convenções são as mesmas utilizadas no gráfico da Figura 6.14. Ordem dos filtros encontrados para os casos I, A1e A2: 20^a ordem.

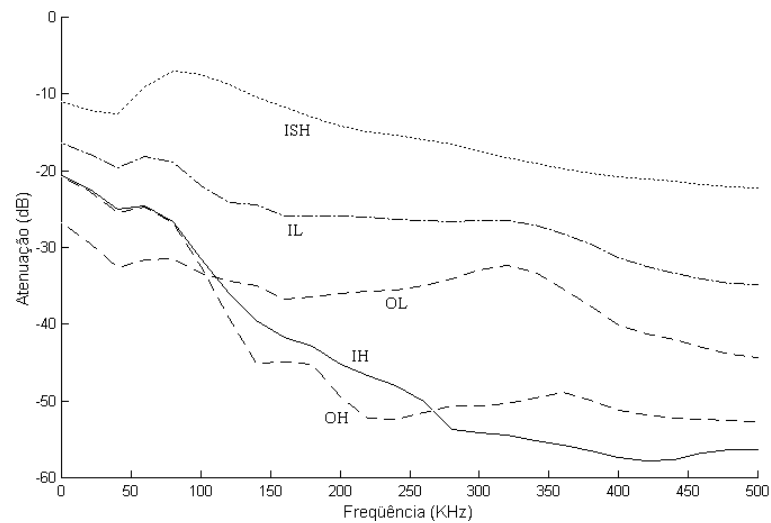


Figura 6.27 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: prédios residenciais. As convenções são as mesmas utilizadas no gráfico da Figura 6.15. Ordem dos filtros encontrados para os casos ISH, IL, OH e OL: 20^a ordem; caso IH: 18^a ordem.

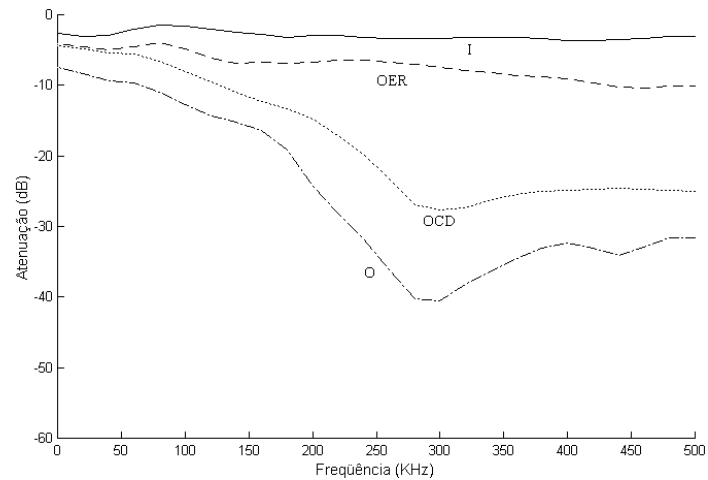


Figura 6.28 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: residência em bairro de classe média. As convenções são as mesmas utilizadas no gráfico da Figura 6.16. Ordem dos filtros encontrados para os casos I, O, OCD e OER: 20^a ordem.

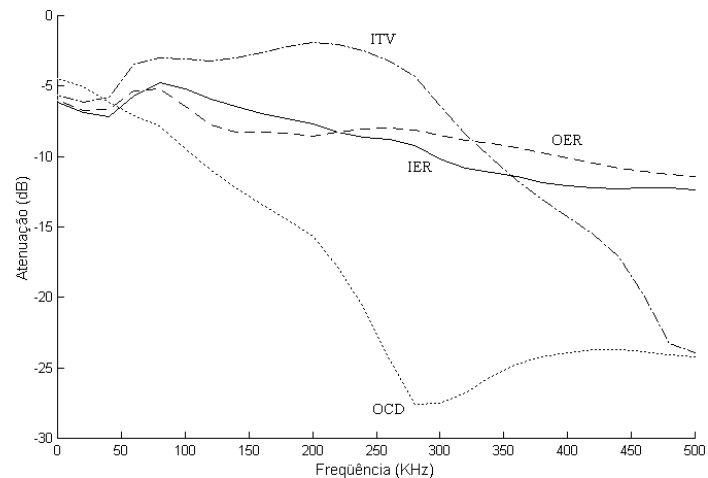


Figura 6.29 – Gráfico de atenuação x frequência dos modelo encontrados, para o seguinte caso: residência em bairro de classe média. As convenções são as mesmas utilizadas no gráfico da Figura 6.17. Ordem dos filtros encontrados para os casos ITV, IER, OCD e OER: 20^a ordem.

A quase totalidade dos filtros obtidos são de 20^a ordem. Os casos em que os filtros têm 18^a ordem e 19^a ordem são devidos a um menor erro quadrático da aproximação. Os coeficientes dos filtros encontrados são apresentados no Apêndice B.

A partir deste momento, todos os casos aproximados terão uma terminologia: terá um número indicando o grupo a qual pertence, seguido da denominação de cada curva. Logo, os nomes dos canais são:

- a) Complexo industrial, período diurno: Caso 1. Nomes dos canais: Caso 1IS, Caso 1I, Caso 1A1 e Caso 1A2;
- b) Complexo industrial, período noturno: Caso 2. Nome dos canais: Caso 2I, Caso 2A1 e Caso 2A2;
- c) Complexo hospitalar: Caso 3. Nome dos canais: Caso 3I, Caso 3A1 e Caso 3A2;
- d) Prédios residenciais: Caso 4. Nome dos canais: Caso 4IL, Caso 4IH, Caso 4ISH, Caso 4OL e Caso 4 OH;
- e) Residência em bairro de classe média, sem cargas específicas: Caso 5. Nome dos canais: Caso 5I, Caso 5O, Caso 5OCD e Caso 5OER;
- f) Residência em bairro de classe média, com cargas específicas: Caso 6. Nome dos canais: Caso 6ITV, Caso 6IER, Caso 6OCD e Caso 6OER.

Algumas observações sobre os filtros encontrados devem ser consideradas:

- a) Todos os filtros digitais encontrados são estáveis (pólos dentro do círculo unitário);
- b) Todos os filtros obtidos pelo método de Yule-Walker são de fase mínima, ou seja, os zeros estão localizados dentro do círculo unitário;

6.5. Análises iniciais de um algoritmo de recepção SSC

6.5.1. Considerações iniciais

Os requisitos para a implantação de um algoritmo para detecção de símbolos SSC no protocolo CEBus são:

- a) Detectar o início do pacote (preâmbulo) na rede elétrica;
- b) Verificar se há indícios de colisão;
- c) Sincronizar os símbolos recebidos com suas cópias locais;
- d) Detectar os símbolos do corpo do pacote;
- e) Decodificar os símbolos do pacote *CEBus* em bits;
- f) Calcular o CRC para verificar erros;
- g) Detectar erros de transmissão;

O núcleo do algoritmo de recepção é a detecção do preâmbulo na rede elétrica e a sincronização e decodificação dos símbolos do pacote CEBus. Uma vez que estes módulos sejam implementados, os demais podem ser adicionados *a posteriori*.

6.5.2. Análise do *Buffer* de Entrada

O sinal analógico SSC, trafegando pela rede, é convertido em um sinal digital pelo dispositivo *CEBus*. O processo de conversão analógico para digital utiliza um conversor AD de 1 bit, ou seja, o sinal analógico é convertido em uma sequência de números ± 1 . Esta etapa de conversão tem início no circuito do buffer de entrada da Figura 5.19, e pode ser explicada como segue:

- a) Quando a tensão da base do transistor Q2 for “levemente” negativa, o transistor Q2 entra em corte, e a saída do sinal (coletor de Q2) é aproximadamente +5V;
- b) Quando a tensão da base for “levemente” positiva, o transistor Q2 entra em saturação, e a saída do sinal (coletor de Q2) é aproximadamente +0 V.
- c) Quando a tensão de base se situar entre as situações anteriores, o circuito opera em modo linear com um ganho de 20 dB.

Logo, o sinal a ser processado é uma sequência de pulsos com valor entre 0 e 5V. As vantagens da operação do sinal SSC em somente dois níveis lógicos são:

- a) a não utilização de uma etapa conversora AD (tipicamente um conversor AD), o que reduz o custo de implementação;
- b) a simplificação dos algoritmos de recepção;
- c) o alto ganho introduzido no processo (20 dB) permite que sinais de baixa amplitude sejam detectados;

As desvantagens são:

- a) o ruído é também amplificado pelo mesmo fator (20 dB);
- b) o sinal da base do transistor Q2 é composto por duas componentes, o sinal SSC mais o ruído. Após passarem pelo transistor Q2, o ruído não pode mais ser separado do sinal, pois a operação efetuada pelo transistor no sinal de entrada é tipicamente não-linear, o que limita a robustez do algoritmo de recepção;

A Figura 6.30 mostra uma aproximação da operação do buffer de entrada, implementada no Matlab para gerar um sinal com os níveis 0 e 5V. Este bloco é aqui denominado “quadrador SSC”.

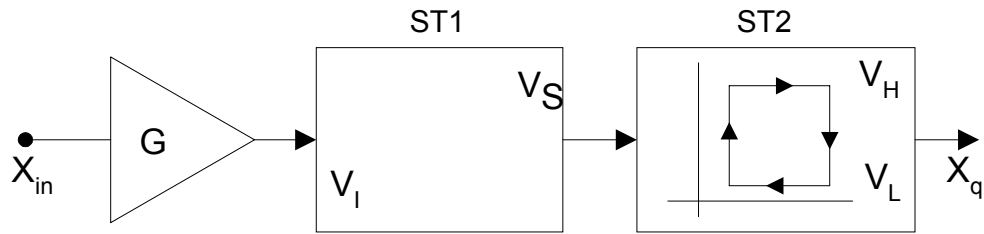


Figura 6.30 – Aproximação da operação de amplificação e quadratura de um sinal aplicado ao ponto X_{in} .

A operação descrita pelo “quadrador SSC” é não-linear, e aproxima o comportamento do circuito de amplificação e a operação em modo chaveado do transistor Q2. O gráfico da Figura 6.31 mostra uma simulação de um sinal SSC (um período de 100μ) após passar pelo “quadrador SSC”:

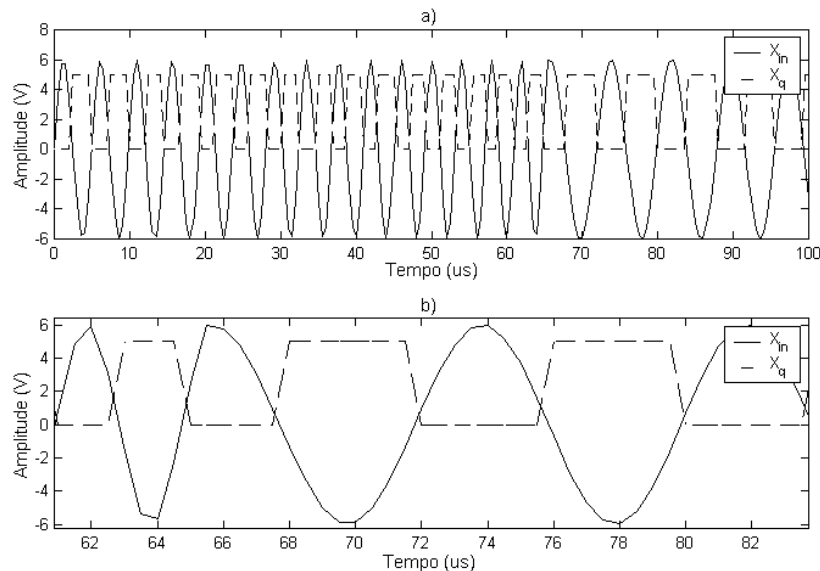


Figura 6.31 – a) Aplicação do amplificador e dos operadores não-lineares em um sinal SSC.
b) No detalhe, notar a relação de oposição entre a saturação e o corte do transistor e a polaridade do sinal SSC.

No “quadrador SSC”, o bloco de ganho G e o bloco de saturação (ST1) são os principais componentes do dispositivo. O bloco ST2, um Schmidt Trigger, é usado somente para conformar a onda a um nível de pulsos TTL e simular a entrada de um dispositivo lógico (como uma porta de IO de um DSP).

Os limites escolhidos para os parâmetros do “quadrador SSC” são $G = 10$ (20 dB de ganho), $V_S = 4.7$, $V_I = 0.7$, $V_H = 3.7$ e $V_L = 1.7$. Os parâmetros V_S e V_I são os valores de saída do coletor de Q2, e V_H e V_L foram escolhidos para proporcionar uma excursão

linear entre VS e VI. O parâmetro G foi escolhido para proporcionar o mesmo ganho do *buffer* de entrada.

6.5.3. Análise da função de autocorrelação do sinal SSC

Seja $s(n)$ a amostragem do sinal SSC $s(t)$ a ser propagado em uma rede elétrica. O gráfico da função de autocorrelação (normalizada) de $s(n)$ é mostrado na Figura 6.32.

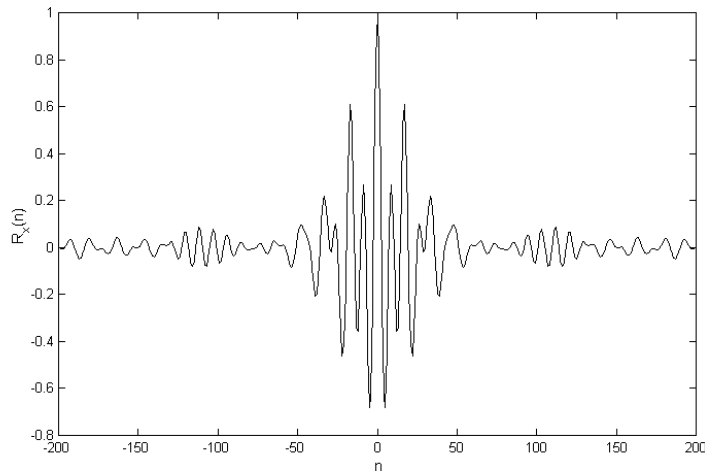


Figura 6.32 – Seqüência de autocorrelação de $s(n)$.

O valor máximo de $R_x(n)$ ocorre para $n=0$, e indica o instante de tempo em que existe a sincronização entre as duas cópias de $s(n)$. Conforme descrito anteriormente o sinal $s(t)$ é convertido em um sinal digital com nível TTL. Este sinal é amostrado na porta do processador DSP e convertido na seqüência ± 1 . A Figura 6.33 apresenta a estrutura lógica para converter o sinal TTL na seqüência PN. Este bloco é denominado de conversor TTL/PN.

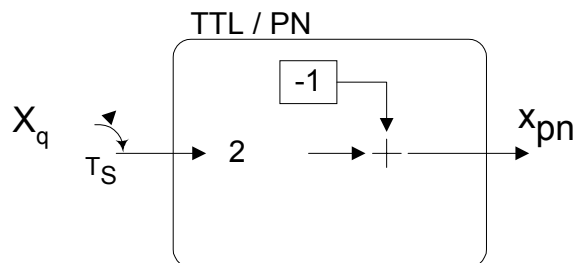


Figura 6.33 – Conversor TTL/PN

O conversor TTL/PN é conectado na saída do “quadrador SSC”, e gera uma seqüência $X_{pn}(n)$, composta somente de valores ± 1 . A amostragem do sinal é obtida pela leitura da

porta do processador onde o sinal está conectado. O gráfico da Figura 6.34 mostra a comparação entre o sinal SSC e a saída do conversor TTL/PN.

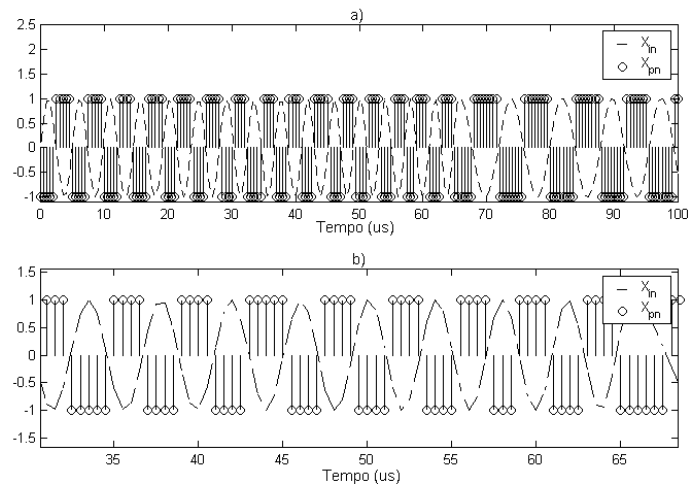


Figura 6.34 – a) Entrada (X_{in}) e saída (X_{pn}) do arranjo “quadrador SSC” + conversor TTL/PN; b) detalhe das formas de onda.

A Figura 6.35 mostra a sequência de autocorrelação de X_{pn} .

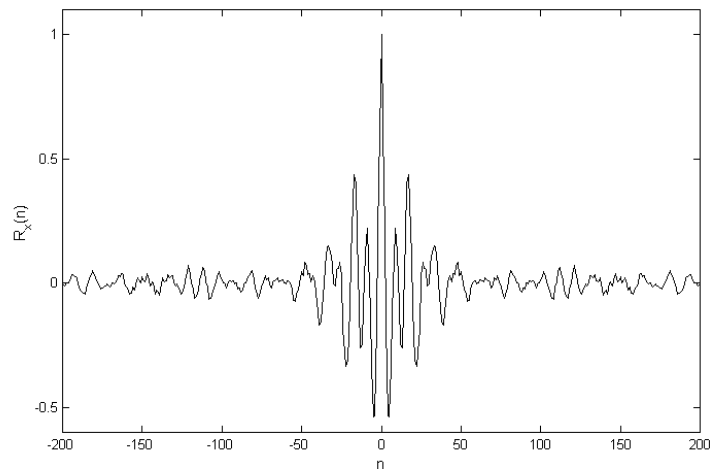


Figura 6.35 – Sequência de autocorrelação de X_{pn} .

Comparando a função de autocorrelação de uma sequência pseudo-aleatória mostrada na Figura 2.5 com a função de autocorrelação do sinal X_{pn} , vemos que, tanto o código X_{pn} gerado pelo “quadrador SSC” em cascata com o conversor TTL/PN quanto o *chirp*, têm um comportamento de uma sequência pseudo-aleatória, porém com um caráter oscilatório no entorno de $n = 0$. Para avaliar qual a função de autocorrelação mais se

aproxima de uma seqüência pseudo-aleatória, é necessário uma comparação dos gráficos das duas funções de correlação, $R(X_{in})$ e $R(X_{pn})$. A comparação das funções pode ser observada na Figura 6.36.

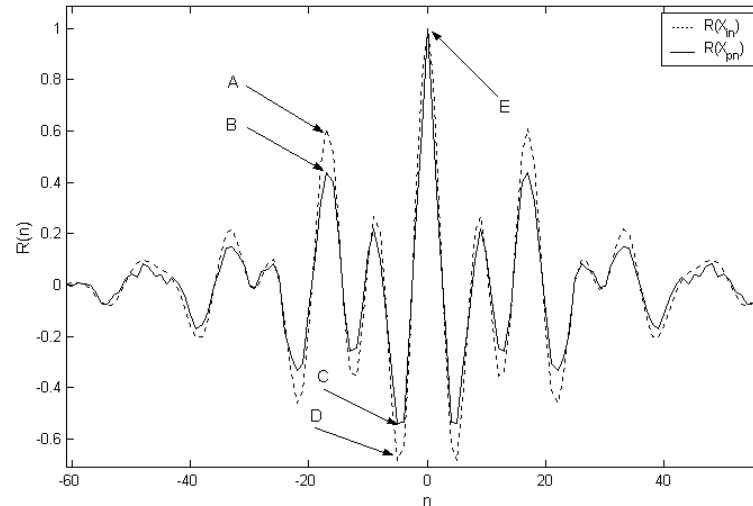


Figura 6.36 – Comparação dos gráficos das funções de autocorrelação de X_{in} e de X_{pn} .

Pode ser observado que a seqüência de autocorrelação de X_{pn} ($R(X_{pn})$) tem a mesma freqüência de oscilação que a seqüência de autocorrelação de X_{in} ($R(X_{in})$), apresentando, porém, oscilações mais amortecidas, como pode ser visto nos pontos A e B (parte positiva) e C e D (parte negativa) da Figura 6.36. Porém, o pico das funções $R(X_{pn})$ e $R(X_{in})$ ocorrem no mesmo instante, e com os mesmos valores, como mostra o ponto E. Logo, a função X_{pn} tem uma característica pseudo-aleatória maior que X_{in} , pois apresenta valores de $R(X_{pn})$ mais próximos de zero para valores de n diferentes de zero. Como X_{pn} não apresenta todos os requisitos necessários para que seja classificada como seqüência pseudo-aleatória (Tabela 2.2), X_{pn} pode ser denominada de seqüência “quasi pseudo-aleatória”.

Outra vantagem da utilização de X_{pn} como símbolo na detecção é que a função de correlação cruzada é independente da amplitude do sinal que é recebido. A Figura 6.37 mostra a função de correlação cruzada entre o sinal X_{in} e uma versão escalada de X_{in} por um fator de 0.5. Logo, função de correlação cruzada entre os dois sinais é reduzida pelo mesmo fator de escala, 0.5. Porém, utilizando o conjunto “quadrador SSC”/conversor TTL/PN com os dois sinais de entrada, X_{in} e $0.5 \cdot X_{in}$, obtemos dois sinais de saída, X_{pn} e X_{pn}' , respectivamente. A seqüência de correlação cruzada entre os dois sinais pode ser observada na Figura 6.38, aonde se mostra que o nível de correlação quase não sofre alterações devido à mudança de amplitude do sinal de entrada, ou seja, a conversão do

signal SSC em uma sequência “quasi pseudo-aleatória” aumenta a robustez do sistema de detecção, em função de permitir uma variação no nível de amplitude do sinal recebido.

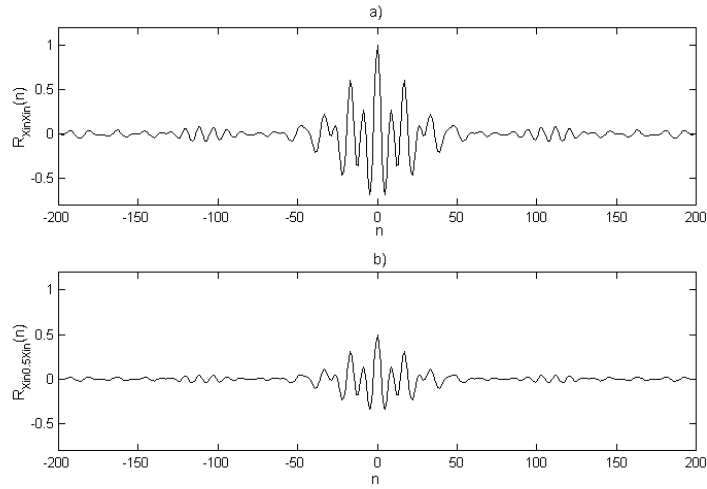


Figura 6.37 – a) Sequência de correlação cruzada entre X_{in} e X_{in} ; b) Sequência de correlação cruzada entre X_{in} e $0.5 \cdot X_{in}$.

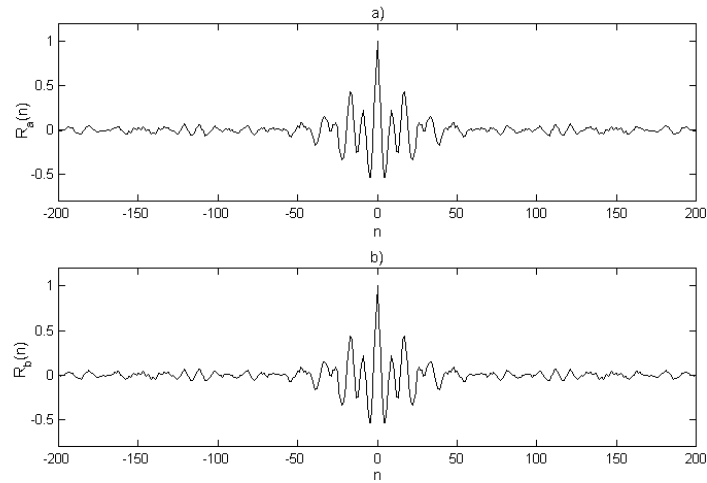


Figura 6.38 – a) Sequência de correlação cruzada entre X_{pn} e X_{pn} ; b) sequência de correlação cruzada entre X_{pn} e X_{pn}' , que são sinais derivados de X_{in} e $0.5X_{in}$.

6.6. Proposição de Algoritmo para Recepção de Sinais SSC no Protocolo *CEBus*.

6.6.1. Considerações iniciais

O algoritmo de recepção de símbolos SSC baseado no protocolo *CEBus* aqui proposto implementa as seguintes etapas no processo de detecção:

- Detecção do preâmbulo do pacote *CEBus*
- Sincronização dos símbolos recebidos com uma réplica do sinal (cópia local), armazenada na memória do sistema receptor e detecção dos símbolos SSC;
- Detecção dos marcadores especiais (EOF, EOP);
- Montagem do pacote *CEBus* a partir dos símbolos detectados;
- Verificação dos erros de recepção.

A estrutura do algoritmo de recepção é mostrada na Figura 6.39:

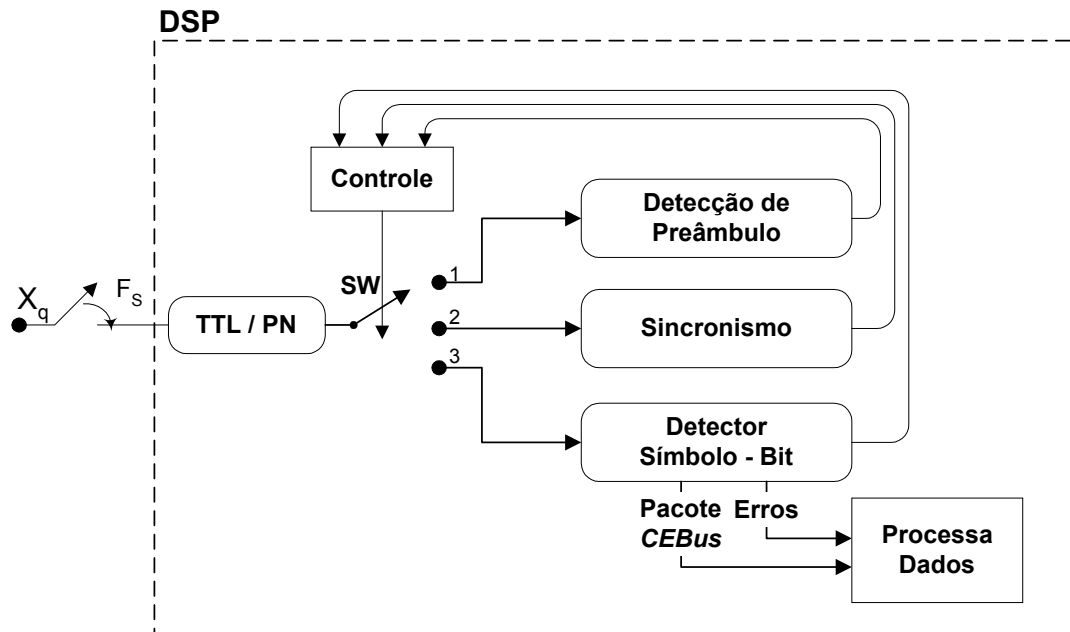


Figura 6.39 – Estrutura do algoritmo de recepção de pacotes *CEBus* proposta.

O controle do algoritmo se dá pelo controle da chave SW, que pode assumir três posições distintas:

- Estágio 1: Chave na posição 1. Ativa o bloco responsável pela detecção do pacote na rede elétrica (detecção de preâmbulo). Uma vez detectado o preâmbulo, o algoritmo passa ao estágio 2.
- Estágio 2: Chave na posição 2. Ativa o bloco de sincronismo, que tenta achar o melhor valor da correlação cruzada entre os símbolos recebidos e a cópia local do símbolo. Uma vez que o valor correto da correlação seja encontrado, o controle da chave muda para o estágio 3;

- c) Estágio 3: Chave na posição 3. Ativa o bloco de detecção de símbolos, que irá processar os símbolos recebidos e montar o pacote CEBus recebido.

Os estágios 1 (Detecção de Preâmbulo), 2 (Sincronismo) e 3 (Detector Símbolo-Bit) serão mostrados nas próximas seções.

6.6.2. Estágio 1: Detecção de Preâmbulo

O processo de detecção se inicia com a chave SW na posição 1, ativando o estágio de detecção de preâmbulo. Uma vez que um pacote de dados seja transmitido por algum nó da rede, o detector de preâmbulo deve reconhecer que símbolos SSC válidos estão sendo transmitidos e descobrir quando o preâmbulo termina.

A estrutura principal do bloco “Detecção de Preâmbulo” é mostrada a seguir, onde o parâmetro de entrada do algoritmo é a sequência X_{pn} e a saída é o valor máximo de correlação encontrado, **max_corr** e o valor do estado da chave **SW**, que determina quando a mudança de estágio deve acontecer.

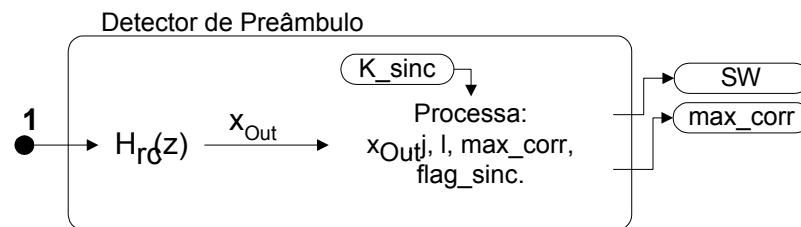


Figura 6.40 – Estrutura do bloco “Detector de Preâmbulo”

O parâmetro **K_sinc** é um valor percentual que diz qual o nível de correlação cruzada para a qual deve ser considerado para uma detecção de símbolo. A escolha do valor de **K_sinc** será mostrada adiante.

Para entender como o algoritmo de detecção de preâmbulo funciona, é necessária uma análise de como o preâmbulo é montado, e perceber se há alguma propriedade a ele intrínseco. Para isso, seja o preâmbulo de um pacote *CEBus*, mostrado na Figura 6.41. O preâmbulo, como mostrado no Capítulo 3, é codificado em ASK, e o byte que ele representa é gerado por um registrador de deslocamento, inicializado com valores aleatórios. Devido à modulação ASK e o fato do primeiro símbolo do pacote ser sempre diferente de zero (ponto A do gráfico da Figura 6.41), o preâmbulo sempre apresenta quatro espaços vazios (regiões B_1 , B_2 , B_3 e B_4 no gráfico da Figura 6.41) entre o primeiro símbolo e o símbolo de PRE_EOF (ponto C do gráfico da Figura 6.41). Logo, o processo de detecção do preâmbulo consiste em se detectar o primeiro símbolo SSC do preâmbulo, contar a ocorrência de quatro

espaços vazios e, logo em seguida, detectar o primeiro *chirp* do PRE_EOF. A ocorrência do primeiro símbolo SSC do PRE_EOF fornece o parâmetro para a mudança da chave SW, e conseqüentemente, o algoritmo passa ao estágio 2.

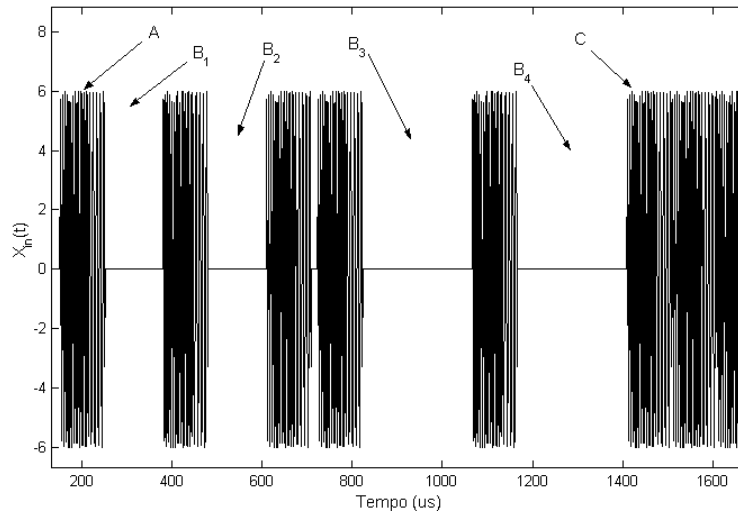


Figura 6.41 – Pré-âmbulo de um pacote *CEBus*. As letras A, B_i e C demarcam o início do pré-âmbulo, a ausência de sinal (devido à modulação ASK) e o início do PRE_EOF, respectivamente.

A estrutura utilizada para determinar o recebimento de símbolos SSC do pré-âmbulo de um pacote *CEBus* é um “*matched filter*”, descrito nas seções 6.3.2 e 6.3.3. Para isso, considere o símbolo utilizado para o processo de detecção, X_{pn} , mostrado na Figura 6.42a. Logo, o filtro de correlação que deve ser utilizado tem os coeficientes mostrados na Figura 6.42b, calculados de acordo com as equações (6.7) e (6.8).

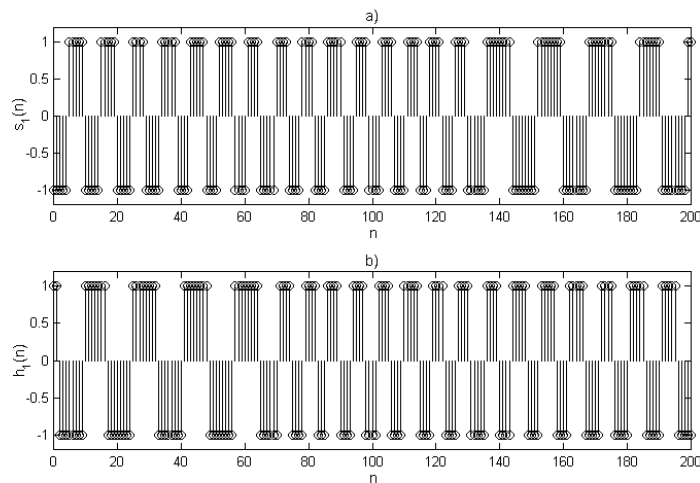


Figura 6.42 – a) Cópia local do símbolo a ser transmitido; b) Coeficientes do “*matched filter*”.

A estrutura geral proposta para o “*matched filter*”, denotada na Figura 6.40 por $H_{rc}(z)$, e acordo com (6.8), é mostrada a seguir, aonde a saída, $X_{out}(n)$ é o parâmetro $z(n)$ a ser calculado.

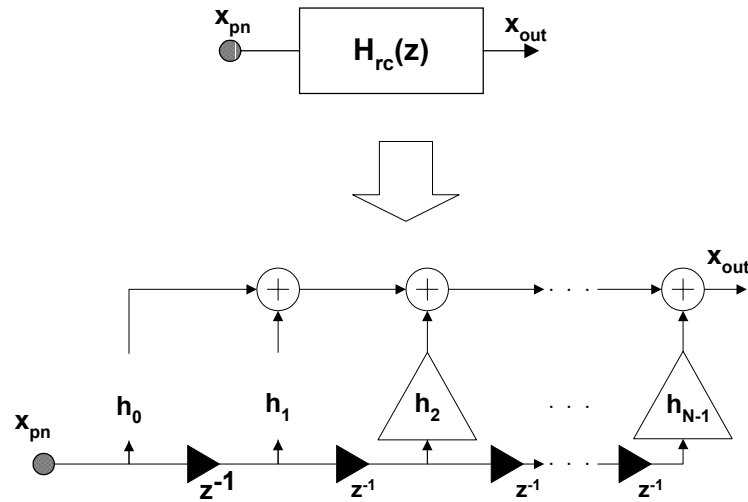


Figura 6.43 – Realização do “*matched filter*”.

O algoritmo que calcula os espaços e detecta o primeiro símbolo, tanto do preâmbulo quanto do PRE_EOF é mostrado a seguir:

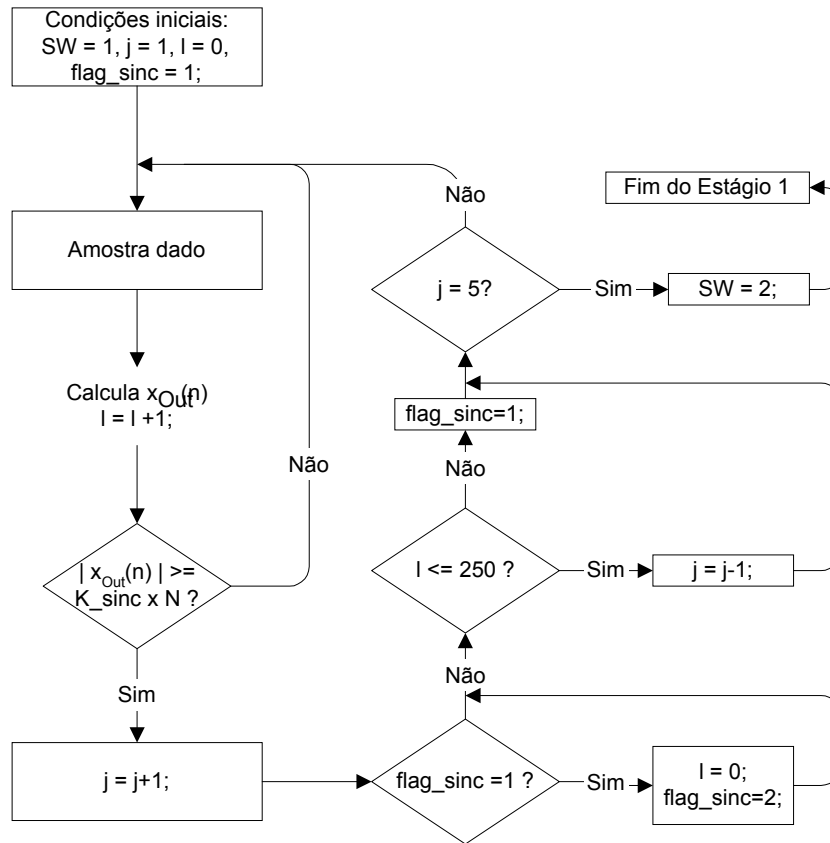


Figura 6.44 – Algoritmo do bloco “Detector de Preâmbulo”

No fluxograma da Figura 6.44, a variável j conta o número de símbolos detectados (que é igual ao número de espaços vazios) e l conta o número de amostras que foram recebidas, servindo para verificar se o símbolo detectado pertence ao mesmo bit ou pertence a bits separados, em virtude da codificação dos bits no pacote *CEBus* ter comprimento de símbolos diferentes para representar o bit “1” e o bit “0”. Uma vez que j tenha valor igual a 5 (o que significa que foi detectado o primeiro símbolo do PRE_EOF), a chave **SW** muda seu valor para 2, e o algoritmo entra no estágio de sincronização.

Um exemplo do algoritmo de detecção do preâmbulo do pacote *CEBus* é mostrado na Figura 6.45 e Figura 6.46. Na Figura 6.45 é mostrado a seqüência de preâmbulo transmitida, “10011010b”, juntamente com o contador j , indicando a presença dos sinais SSC. A Figura 6.46 mostra a saída do “*matched filter*” e o contador j . Observar que a mudança de j ocorre quando um novo símbolo SSC (que pertence a um novo bit) do preâmbulo é detectado, justamente como desejado.

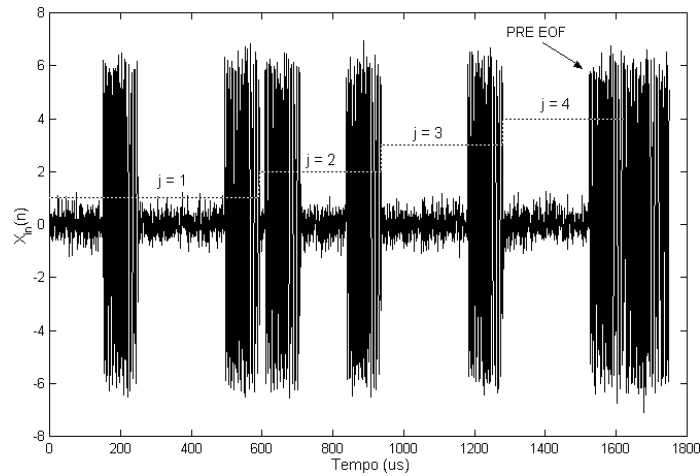


Figura 6.45 – Preâmbulo transmitido (“10011010b”), mostrando o contador de símbolos detectados, multiplicado por um fator de escala de 50, para efeito de visualização.

As simulações mostradas na Figura 6.45 e na Figura 6.46 foram obtidas com uma relação sinal ruído de 20 dB, e o sinal transmitido não percorreu nenhum canal até chegar ao receptor. O fator de escala **K_sinc** foi escolhido de forma que nem o ruído nem harmônicos de alta frequência (que possam estar na faixa de interesse, entre 100KHz e 400 KHz) pudessem disparar o processo. Desta forma, foram simulados vários níveis de ruído, e no pior caso, o nível de correlação cruzada (a saída $x_{out}(n)$) não ultrapassou 25% do valor máximo de correlação. Logo, foi escolhido um fator para **K_sinc** de 55% do valor máximo, o que garante que só o sinal SSC pode disparar o detector de preâmbulo.

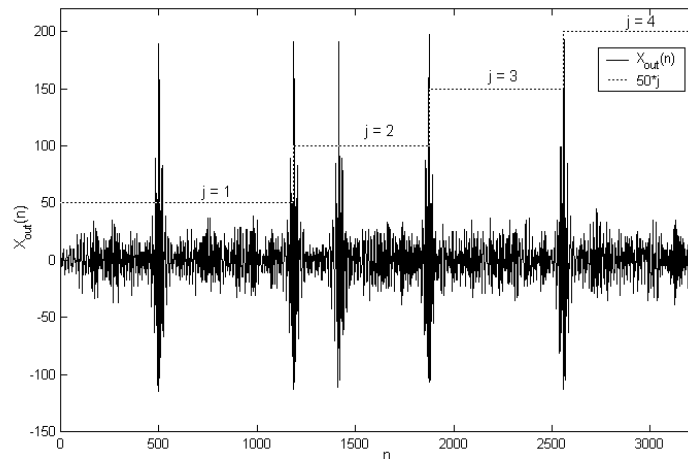


Figura 6.46 – Saída do “matched filter” ($x_{out}(n)$),.

6.6.3. Estágio 2: Sincronismo

Uma vez detectado o preâmbulo e o primeiro símbolo do PRE_EOF, o algoritmo de recepção comanda a chave para a posição 2, ativando o estágio de sincronismo. Isso é efetuado para se obter uma melhor avaliação do nível de correlação cruzada, e tenta explorar ao máximo o restante do símbolo de PRE_EOF (que é montado utilizando-se oito símbolos SSC seguidos, como visto no Capítulo 3), já que resta sete símbolos SSC (o primeiro foi detectado no estágio 1). Como o nível de decisão do estágio 1 é baixo (55%), é necessário encontrar qual o valor máximo da correlação cruzada que pode minimizar o erro de detecção. O estágio 1 não informa qual o máximo da correlação que pode ser alcançado na detecção do símbolo, logo, o estágio 2 procura “varrer” qual o maior valor de correlação que é possível de se obter. Para isso, considere a Figura 6.47, que mostra como o algoritmo de recepção calcula o valor de sincronismo.

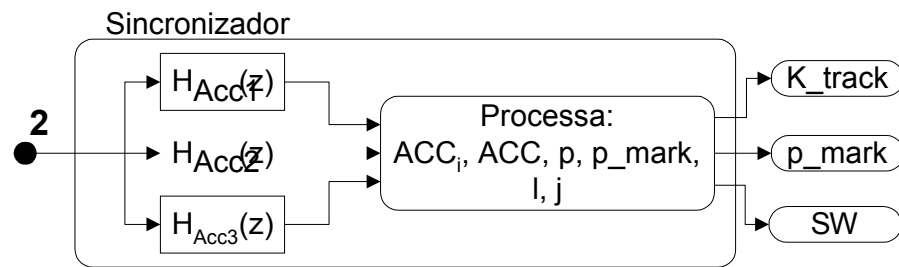


Figura 6.47 – Estrutura do bloco “Sincronismo”.

As funções de transferência $H_{Acci}(z)$ ($i=1,2,3$) calculam o valor da correlação cruzada por meio de filtros de correlação, vistos na seção 6.3.2. Porém, o algoritmo proposto deve tentar estimar o atraso que o sinal recebido tem em relação à cópia local do símbolo, logo, será utilizada uma alteração na função do filtro de correlação, como pode ser visto na Figura 6.48.

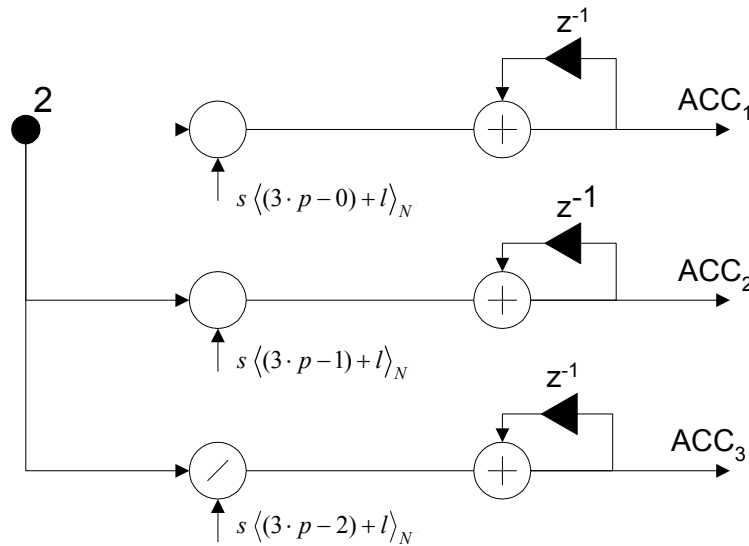


Figura 6.48 – Estrutura dos filtros $H_{Acc_i}(z)$ ($i=1,2,3$).

O arranjo proposto tenta, através da variação do parâmetro p , estimar o atraso entre o sinal recebido, $X_{pn}(n)$, e a cópia local do sinal, $s(n)$. Isto é feito através de uma operação módulo N , com o parâmetro p controlando o avanço e o atraso do símbolo local, $s(n)$. A condição inicial de p é zero, e a cada N amostras, o valor de p é decrementado de uma unidade, o que possibilita ao algoritmo “varrer” os atrasos possíveis de ocorrer, dentro de um intervalo especificado. A Tabela 6.2 mostra o atraso T que pode ser calculado em função do valor de p .

Tabela 6.2 – Atraso T de $s(n-T)$ calculado a partir do parâmetro p .

p	ACC_1	ACC_2	ACC_3
0	$s(n-0)$	$s(n-1)$	$s(n-2)$
1	$s(n-3)$	$s(n-4)$	$s(n-5)$
2	$s(n-6)$	$s(n-7)$	$s(n-8)$
3	$s(n-9)$	$s(n-10)$	$s(n-11)$
4	$s(n-12)$	$s(n-13)$	$s(n-14)$
5	$s(n-15)$	$s(n-16)$	$s(n-17)$
6	$s(n-18)$	$s(n-19)$	$s(n-20)$

O valor de p que indica o atraso T , denotado por p_mark , é calculado ao se avaliar para qual valor de p e de i a função $ACC_i(N-1)$ é máxima, $i=1,2,3$. A cada N amostras e para um p específico, é calculado o máximo entre os acumuladores ACC_i (guardado na variável **temp**), e comparado com o máximo guardado em uma variável auxiliar, **ACC**. Se **temp** for maior que **ACC**, **ACC** leva o valor de **temp**, e **p_mark** leva o valor de p . Quando todos os valores possíveis de p forem utilizados, o bloco retorna o valor de **p_mark** e

conseqüentemente o atraso sofrido pela cópia local $s(n)$. Neste instante, ACC é o valor da maior correlação cruzada encontrada, e um novo parâmetro é calculado, **K_{track}**. Similar em função ao parâmetro **K_{sinc}**, tem seu valor calculado de acordo com (6.18):

$$K_{track} = \frac{0.85 \cdot ACC}{N} \quad (6.18)$$

aonde o fator 0.85 (85%) escala o nível de decisão para 85% do maior valor de correlação encontrado. A quantificação do fator 0.85 é empírica e foi baseada em simulações efetuadas.

O número máximo de atrasos que podem ser calculados depende do número de ramos (número de acumuladores ACC_i) que são implementados, pois os símbolos para a avaliação do atraso (7 símbolos do PRE_EOF) são fixos. A equação que calcula o número máximo de atrasos é mostrada em (6.19):

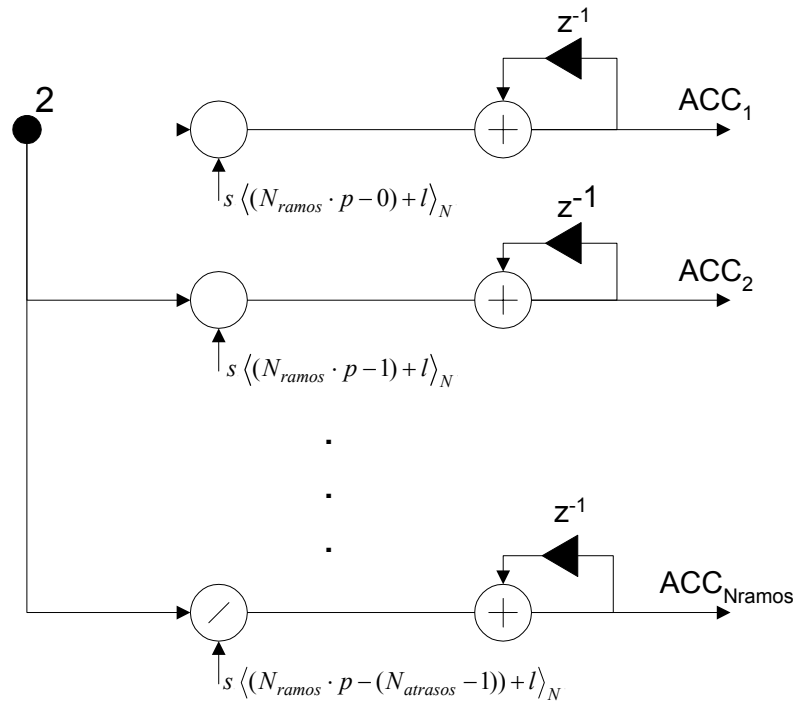
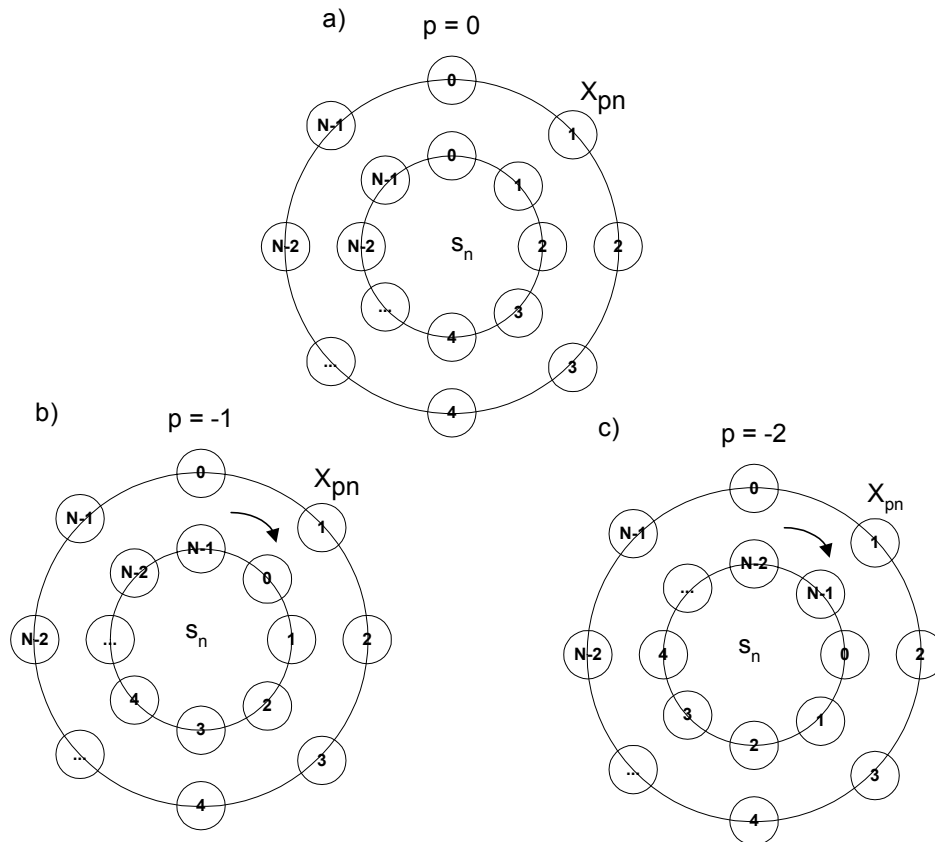
$$N_{atrasos} = 7 \cdot N_{ramos} \quad (6.19)$$

Para $N_{atrasos}$ atrasos, a estrutura geral para o cálculo de $H_{Acci}(z)$, $i = 1, 2, \dots, N_{ramos}$, é mostrada na Figura 6.49.

Para um maior entendimento de como a operação módulo N no índice de um vetor afeta o próprio índice, seja a multiplicação expressa por (6.20):

$$W(n) = X_{pn}(n) \cdot s(\langle p + n \rangle_N) \quad (6.20)$$

onde $W(n)$ é somente uma variável auxiliar. Para p igual a zero, a multiplicação de $X_{pn}(n)$ e $s(n)$ ocorre da forma mostrada na Figura 6.50a, ou seja, os índices dos vetores são iguais. Com p diferente de zero, ocorre um “deslocamento” circular no índice dos vetores, e os índices não são mais os mesmos para os dois vetores. O deslocamento pode ser visto como se os dois vetores fossem dois “discos” montados em um mesmo eixo, com o alinhamento controlado pela variável p . Uma variação unitária de p faz com que um dos “discos” se mova em torno do “eixo” comum, provocando o atraso, como mostrado na Figura 6.50b ($p=-1$) Figura 6.50c ($p=-2$).


 Figura 6.49 – Estrutura geral de $H_{Acc}(z)$.

 Figura 6.50 – Exemplo da operação módulo N em um vetor $s(n)$, para $p=0$, $p=-1$ e $p=-2$.

A Figura 6.51 mostra as saídas $ACC_i(n)$ do bloco de sincronismo em um pacote **CEBus**, simulados com $SNR = 10$ dB.

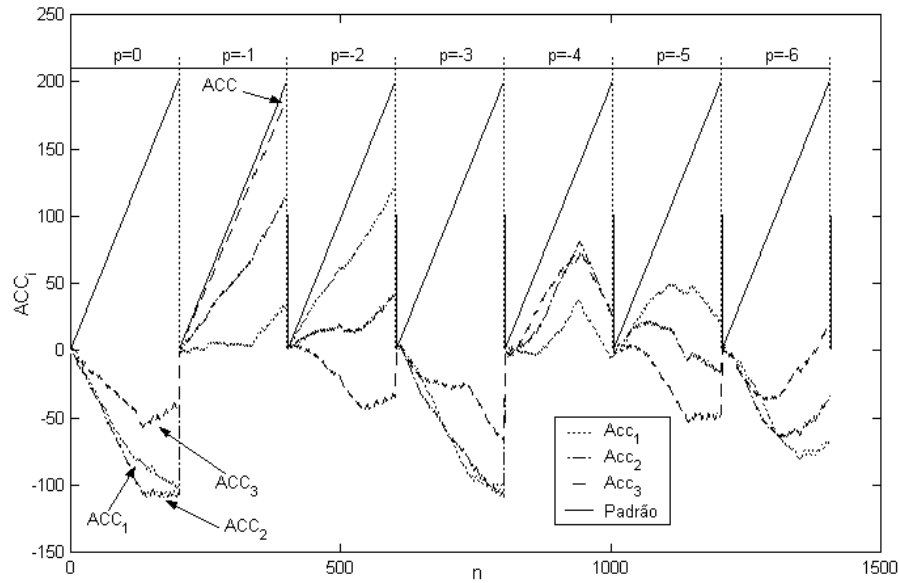


Figura 6.51 – Gráfico das saídas $ACC_i(n)$, para $N_{ramos} = 3$.

O valor máximo da correlação encontrado é assinalado por ACC na Figura 6.51, e ocorre para $p = -1$ no braço ACC_3 . O atraso **p_mark**, calculado a partir de **p** e **i**, é:

$$p_mark = 3 \cdot p - (i - 1) \quad (6.21)$$

aonde **i** em (6.21) indica qual braço foi achado o máximo. Logo, para o nosso caso, temos que **p_mark** vale -5, o que indica que a cópia local deve ser atrasada em 5 amostras para que o sincronismo seja alcançado. O valor de correlação ACC encontrado foi $ACC = 187$, logo, por (6.18), o valor encontrado para **K_track** foi 0.79 (79%). O algoritmo completo do estágio de sincronismo é mostrado na Figura 6.53.

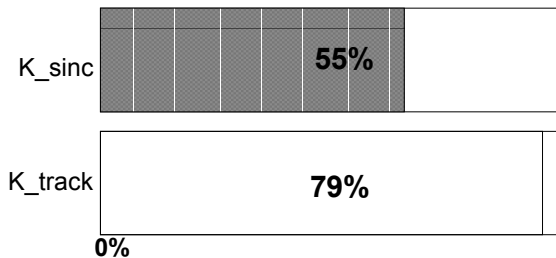


Figura 6.52 – Níveis de decisão do algoritmo encontrados (variáveis **K_sync** e **K_track**).

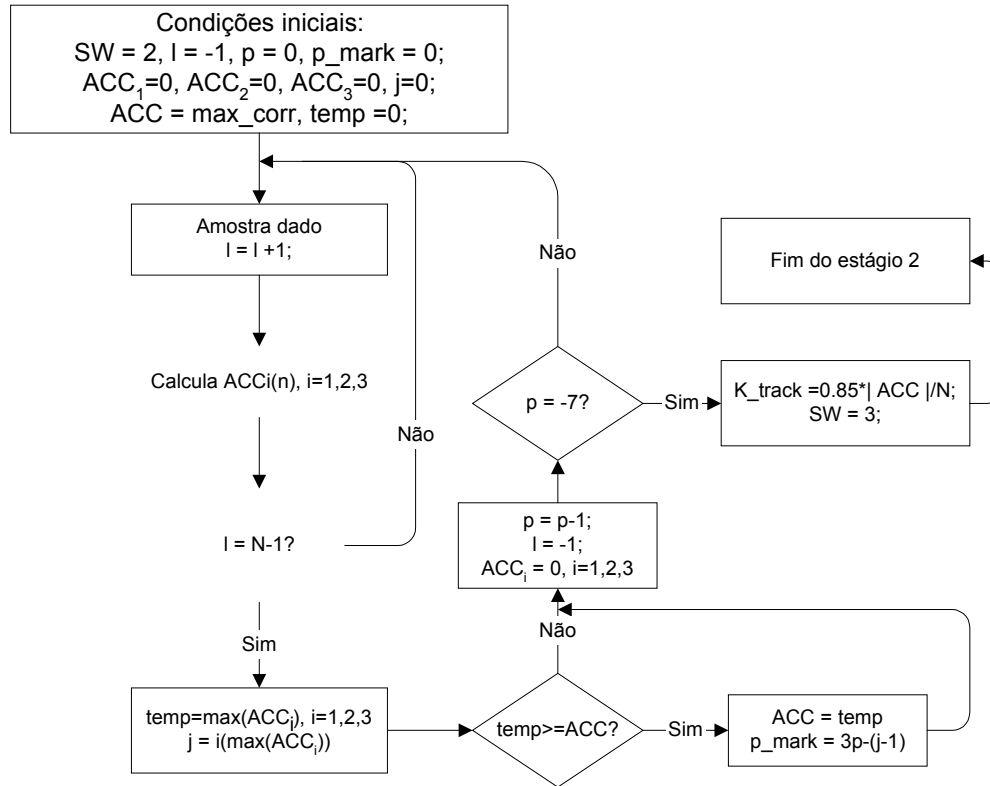


Figura 6.53 – Algoritmo do bloco de sincronismo

Uma vez encontrados os valores desejados (K_{track} e p_{mark}), o algoritmo de recepção ativa o controle da chave para a posição 3, ativando o estágio 3.

6.6.4. Estágio 3: Detector de Símbolo e Conversor Símbolo-Bit

Uma vez que o estágio de sincronismo esteja concluído, o algoritmo passa a detectar os símbolos e montar o pacote *CEBus*. O procedimento de detecção e montagem é efetuado no estágio 3 pelo bloco “Detector de Símbolo”, mostrado a seguir:

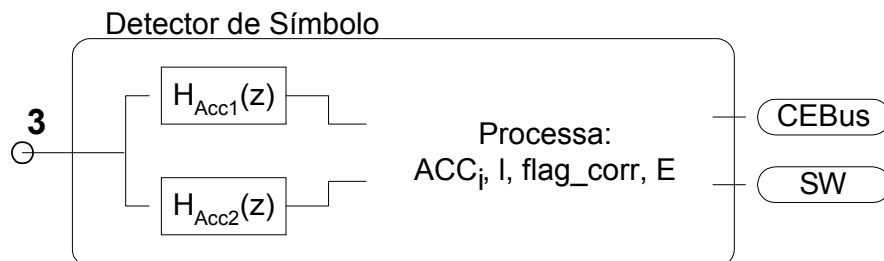


Figura 6.54 – Estrutura do bloco “Detector de Símbolo”

Para detectar os símbolos, é utilizada uma estrutura, $H_{ACCI}(z)$, que possui duas saídas: $H_{Acc1}(z)$ e $H_{Acc2}(z)$, que calcula a correlação cruzada entre o sinal recebido a cópia local, como o arranjo da Figura 6.49.

Uma modificação, porém, será introduzida, de forma a aumentar a robustez do processo de detecção. Seja duas seqüências pseudo-aleatórias, denotadas por $c_1(n)$ e $c_2(n)$, mostradas na Figura 6.55:

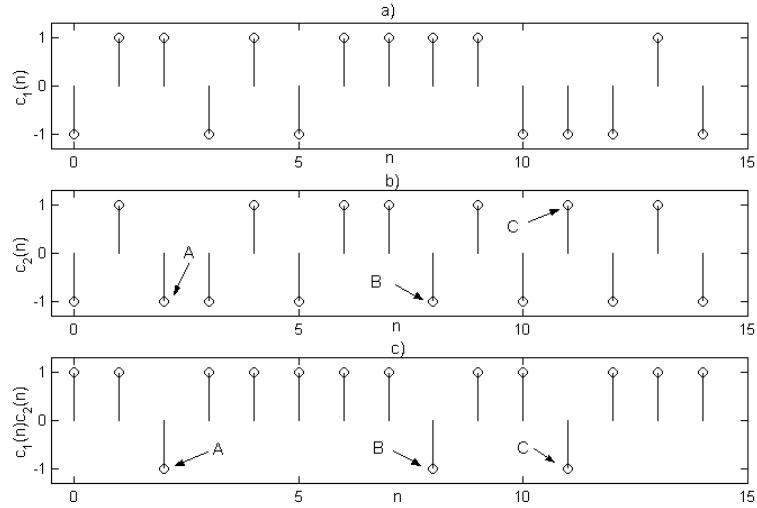


Figura 6.55 – a) seqüência pseudo-aleatória $c_1(n)$; b) seqüência pseudo-aleatória $c_2(n)$; c) multiplicação de $c_1(n)$ e $c_2(n)$;

Na Figura 6.55a, temos uma seqüência pseudo-aleatória $c_1(n)$, e na Figura 6.55b, temos uma seqüência pseudo-aleatória $c_2(n)$, que é uma cópia de $c_1(n)$, a não ser pelos pontos demarcados por A, B e C (que correspondem aos pontos $c_2(2)$, $c_2(8)$ e $c_2(12)$), que foram invertidos (multiplicados por -1). Logo, a correlação cruzada entre estes dois sinais é:

$$R_{c_1c_2}(0) = \sum_{n=0}^{N-1} c_1(n) \cdot c_2(n) \quad (6.22)$$

Como $c_1(n)$ e $c_2(n)$ são compostos por elementos bipolares (+1 ou -1), o cálculo da correlação cruzada entre os dois pode ser escrito como o número de elementos iguais menos o número de elementos diferentes, pois como pode ser observado na Figura 6.55c, a multiplicação de $c_1(n)$ por $c_2(n)$ gera elementos iguais para valores iguais e elementos opostos para valores diferentes, assinalados por A, B e C na Figura 6.55c.

Logo, seja o arranjo mostrado na Figura 6.56, utilizado no detector de símbolos da Figura 6.54, que calcula a seqüência de correlação cruzada entre o sinal $s(n)$ e o sinal $X_{pn}(n)$.

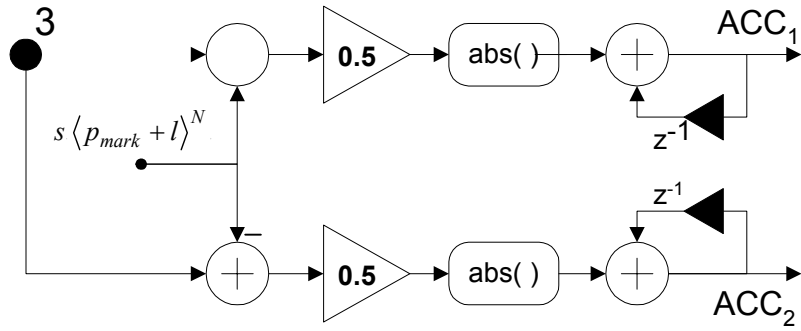


Figura 6.56 – Estrutura de correlação para seqüências pseudo-aleatórias

A saída ACC_1 fornece a quantidade de acertos, enquanto que a saída ACC_2 fornece a quantidade de erros. O fator de escala 0.5 não é necessário e somente serve para normalizar a função de correlação cruzada para que tenha um valor máximo de N . A correlação cruzada pode ser obtida a partir da diferença dos acumuladores, ou seja, da expressão $ACC_1 - ACC_2$.

Para comprovar o aumento da robustez da detecção, vamos analisar, sem perda de generalidade, as seqüências $c_1(n)$ e $c_2(n)$. O cálculo da correlação cruzada de $c_1(n)$ e $c_2(n)$, segundo a equação (6.22), é:

$$R_{c_1c_2}(0) = \sum_{n=0}^{N-1} c_1(n) \cdot c_2(n) = 12 - 3 = 9 \quad (6.23)$$

A relação entre o valor encontrado e o valor máximo de correlação (que é a autocorrelação de $c_1(n)$) nos dá o quanto que as duas curvas tem em comum. Logo, temos que:

$$\frac{R_{c_1c_2}(0)}{R_{c_1c_1}(0)} = \frac{\sum_{n=0}^{N-1} c_1(n) \cdot c_2(n)}{\sum_{n=0}^{N-1} c_1(n) \cdot c_1(n)} = \frac{9}{15} = 0.6 = 60\% \quad (6.24)$$

No processo de detecção, uma seqüência $c_1(n)$ e $c_2(n)$, que tem 15 elementos, com apenas 3 elementos diferentes, gerou um índice de correlação de apenas 60%. Analisando as mesmas seqüências na estrutura da Figura 6.56, temos que:

$$\begin{aligned} ACC_1 &= 12 \\ ACC_2 &= 3 \end{aligned} \quad (6.25)$$

Logo, a relação ACC_1/N nos dá o índice de coincidência entre as duas curvas, logo:

$$\frac{ACC_1}{N} = \frac{12}{15} = 0.8 = 80\% \quad (6.26)$$

O índice de coincidência (correlação) aumentou de 60% para 80%, em virtude de não ser computado o número de erros (ACC_2), situação que não acontece na expressão (6.24). O cômputo do índice de erros pode detectar se houve a inversão do símbolo, no caso de transmissão de sistemas de comunicação bipolares. Seja $c_3(n)$ uma nova sequência pseudo-aleatória, que é a sequência $c_2(n)$ multiplicada por -1 . Logo, a correlação de $c_3(n)$ e $c_1(n)$ é:

$$\frac{R_{c_1c_3}(0)}{R_{c_1c_1}(0)} = \frac{\sum_{n=0}^{N-1} c_1(n) \cdot c_3(n)}{\sum_{n=0}^{N-1} c_1(n) \cdot c_1(n)} = \frac{-9}{15} = -0.6 = -60\% \quad (6.27)$$

O nível de correlação indica que foi transmitido o símbolo inverso. Analisando as mesmas seqüências $c_1(n)$ e $c_3(n)$ na estrutura da Figura 6.56, temos que:

$$\begin{aligned} ACC_1 &= 3 \\ ACC_2 &= 12 \end{aligned} \quad (6.28)$$

Logo, a relação ACC_1/N nos dá o índice de coincidência entre as duas curvas, logo:

$$\frac{ACC_1}{N} = \frac{3}{15} = 0.2 = 20\% \quad (6.29)$$

O valor baixo de ACC_1 indica que não há muita semelhança entre $s_1(n)$ e $s_3(n)$, porém, analisando o fator ACC_2 , temos:

$$\frac{ACC_2}{N} = \frac{12}{15} = 0.8 = 80\% \quad (6.30)$$

Como os sinais $s_i(n)$ só possuem dois valores, o alto valor da relação ACC_2/N indica que o símbolo inverso foi transmitido, mantendo o alto valor para a detecção (80%). Uma outra estrutura pode calcular da mesma forma os valores de ACC_1 e ACC_2 , e é mostrada na Figura 6.57. Tal estrutura consegue operar com uma operação lógica XOR e uma operação lógica NOT, porém, precisa ter um conversor PN / TTL, que converta os sinais de $+1$ e -1 para “1” e “0”, respectivamente. Outra saída é retirar o conversor TTL / PN no estágio 3, de forma que não seja preciso colocar o conversor PN / TTL na entrada da operação lógica XOR.

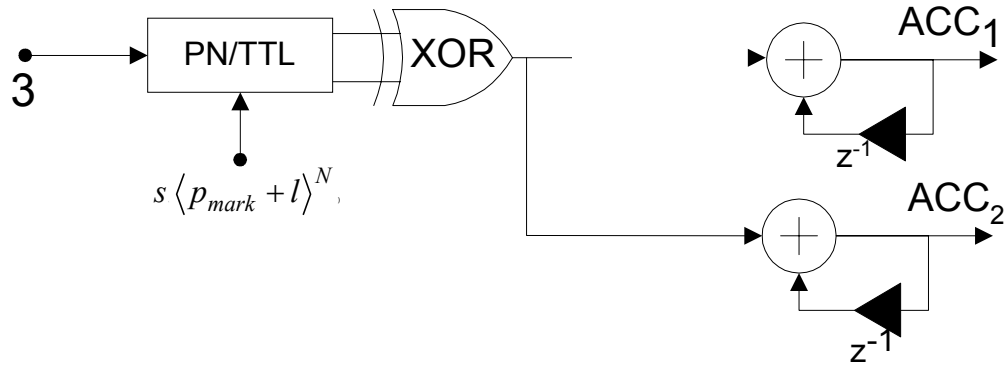


Figura 6.57 – Estrutura alternativa para o cálculo de ACC_1 e ACC_2 .

É necessário lembrar que a cada N amostras, o valor da memória do filtro ACC_i é zerado (valor inicial), e um novo processo de detecção se inicia.

Seja utilizando a estrutura da Figura 6.56 ou da Figura 6.57, a meta é utilizar as saídas ACC_i para monitorar nível de correlação, logo, a seguinte convenção será utilizada:

$$\begin{aligned} ACC_1 \geq K_{_track} \cdot N &\Rightarrow \text{Símbolo } S_1 \text{ recebido} \\ ACC_2 \geq K_{_track} \cdot N &\Rightarrow \text{Símbolo } S_2 \text{ recebido} \end{aligned} \quad (6.31)$$

Uma vez detectado o símbolo SSC, a próxima etapa é avaliar o recebimento do bit. Uma vez que o maior símbolo do corpo do pacote *CEBus* (em comprimento) é o EOP, que contém 4 símbolos SSC, é necessário examinar os símbolos anteriores para detectar qual bit foi transmitido. Logo, é necessário guardar os últimos quatro símbolos recebidos, e a partir deles, decidir se foi detectado “1”, “0”, EOF ou EOP. No algoritmo de recepção proposto, os símbolos SSC são armazenados em um vetor E , o símbolo mais recente armazenado em $E4$ e o símbolo mais antigo armazenado em $E1$. O vetor E é inicializado com o valor 0 para todas as suas posições, e a Tabela 6.3 informa o símbolo *CEBus* recebido em função dos símbolos SSC recebidos.

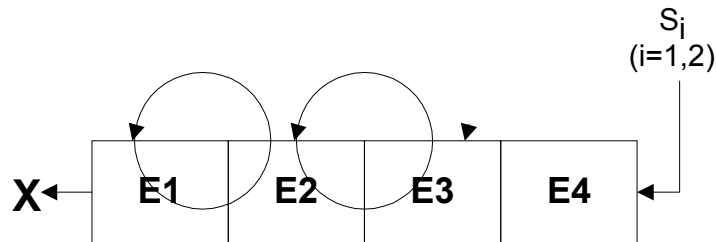


Figura 6.58 – Vetor de símbolos SSC

Tabela 6.3 – Tabela de detecção de símbolos *CEBus*

E1	E2	E3	E4	Símbolo <i>CEBus</i>
X	SA	SA	SB	"0"
X	SA	SB	SA	"1"
X	0	SA	SB	"1"
SA	SA	SA	SB	"EOF"
SA	SA	SA	SA	"EOP"

SA e SB representam estados antagônicos (se SA for igual a S_2 , SB é igual a S_1 , e vice-versa), e X representa qualquer estado. Se a configuração do vetor E não for igual à nenhuma apresentada na Tabela 6.3, deve-se esperar o próximo símbolo SSC para tentar determinar o símbolo *CEBus* recebido.

O fluxograma da Figura 6.58 mostra o algoritmo completo do bloco "Detector de Símbolos".

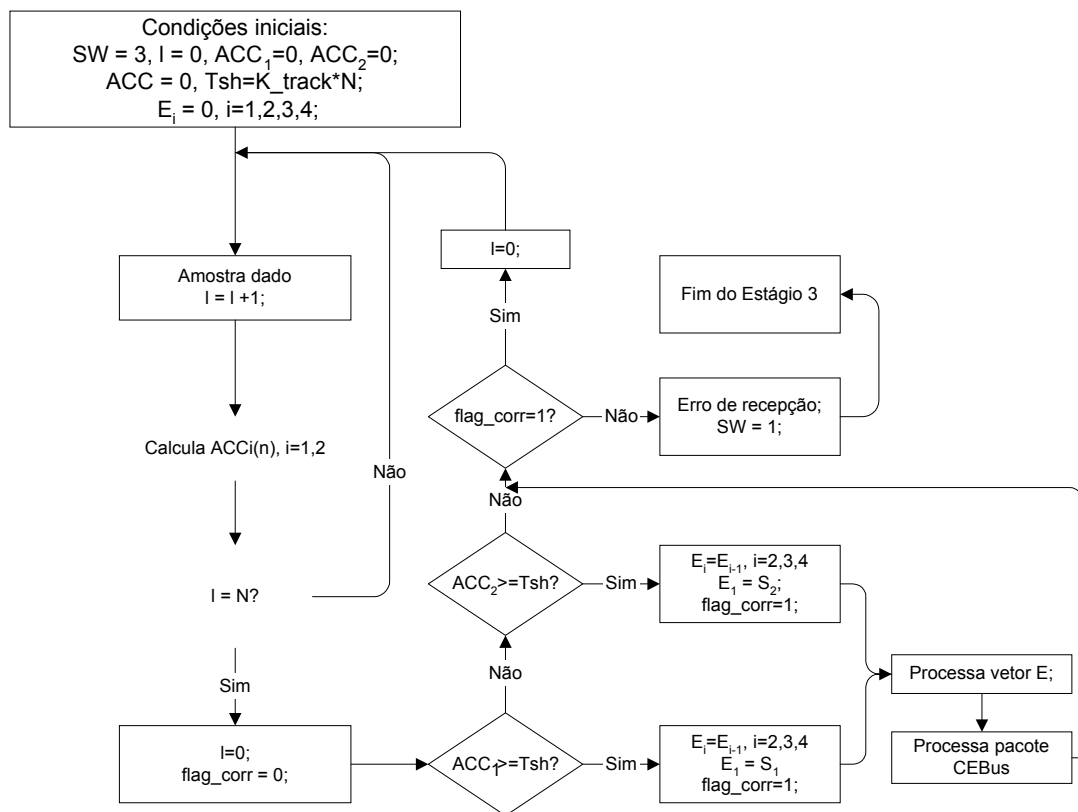


Figura 6.59 – Algoritmo do bloco "Detector de Símbolos"

O processo de detecção termina quando for detectado o símbolo de EOP, o que significa que o pacote *CEBus* foi recebido e está disponível para a leitura. Em seguida, o controle do algoritmo comanda a chave **SW** para a posição 1, o que faz com que o algoritmo

ative o bloco de detecção de preâmbulo para detectar um novo pacote na rede elétrica. O algoritmo também detecta a ausência de símbolos SSC antes da detecção do símbolo EOP, o que significa que a transmissão do pacote foi interrompida, o que gera uma condição de erro.

Os gráficos a seguir mostram algumas curvas da simulação do estágio 3, com SNR de 10 dB.

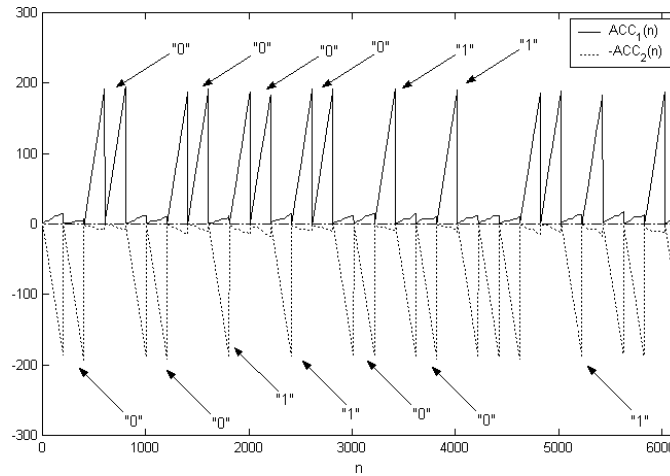


Figura 6.60 – Gráfico das funções $ACC_1(n)$ e $ACC_2(n)$.

O gráfico da Figura 6.60 mostra o início do estágio 3, com o cálculo das funções ACC_1 e ACC_2 e as marcações dos bits detectados em função dos símbolos SSC detectados (o padrão da codificação do corpo do pacote – PRK – é mostrado no Capítulo 3).

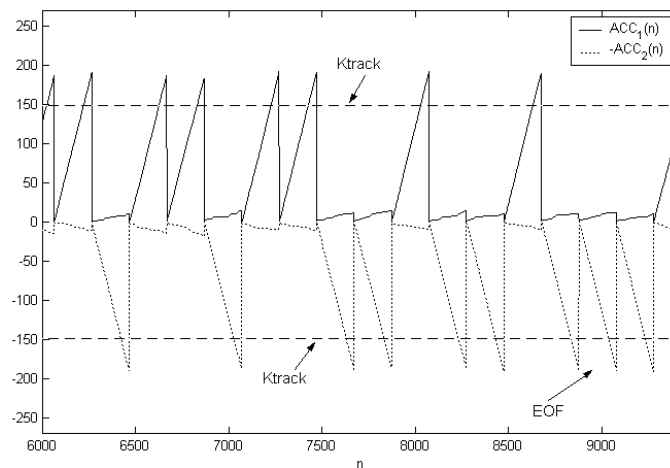


Figura 6.61 – Gráfico das funções $ACC_1(n)$ e $ACC_2(n)$, enfatizando o recebimento do símbolo EOF.

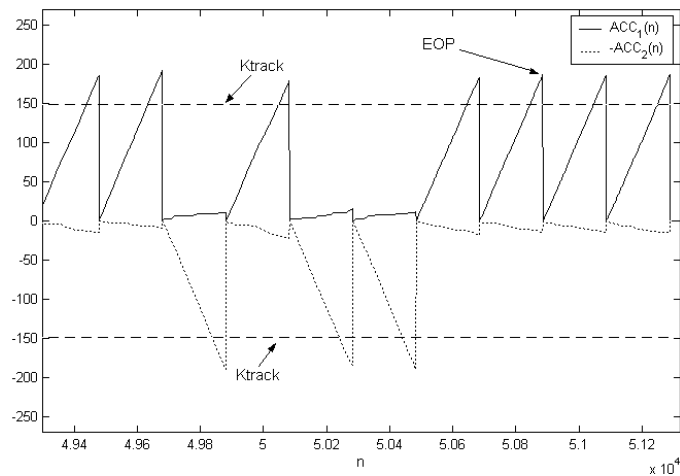


Figura 6.62 – Gráfico das funções $ACC_1(n)$ e $ACC_2(n)$, enfatizando o recebimento do símbolo EOP.

O gráfico da Figura 6.61 mostra o recebimento de um símbolo EOF, demarcando o fim do recebimento de um campo de endereço, enquanto que o gráfico da Figura 6.62 mostra o recebimento do símbolo EOP, que demarca o fim do pacote *CEBus* recebido. A linha *K_track* mostra para qual nível de correlação o estágio 3 valida os símbolos SSC recebidos.

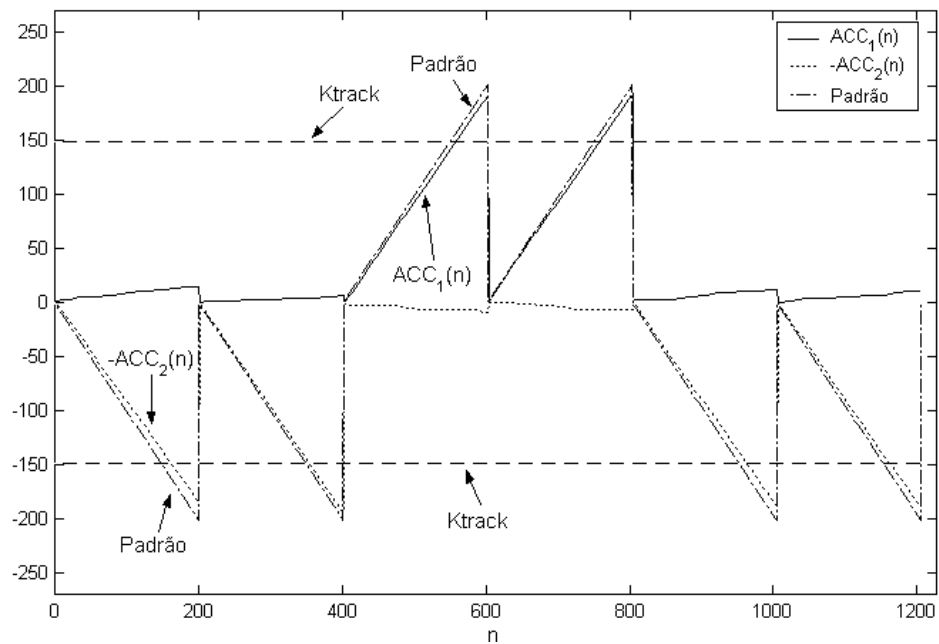


Figura 6.63 – Gráfico de correlação, mostrando o padrão ideal (0% de erros), as curvas $ACC_1(n)$ e $ACC_2(n)$ e o nível de decisão (*K_track*) para a validação do símbolo SSC.

A Figura 6.63 mostra mais detalhes das curvas de correlação $ACC_1(n)$ e $ACC_2(n)$. A linha denominada “Padrão” é uma reta, e significa 0% de erros no cálculo da correlação, ou seja, com 100% de correspondência entre o símbolo SSC local e o símbolo SSC recebido.

6.7. Simulação do Algoritmo Proposto

O algoritmo proposto na seção 6.6 será, agora, simulado para vários valores de relação sinal-ruído (SNR), juntamente com os canais aproximados na seção 6.4. O diagrama esquemático da simulação é mostrado na Figura 6.64, aonde TX(SSC) é o transmissor do pacote *CEBus*, $H(z)$ é o canal aproximado, $\omega(n)$ é o ruído gaussiano branco (AWGN) adicionado ao sinal recebido e RX(SSC) é o receptor do pacote *CEBus*.

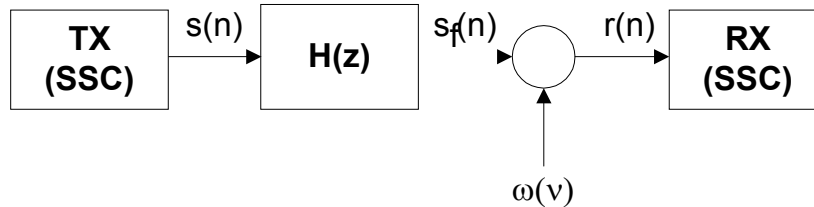


Figura 6.64 – Simulação do algoritmo proposto na seção 6.6.

Os parâmetros utilizados na simulação são fornecidos na Tabela 6.4.

Tabela 6.4 – Parâmetros de simulação do algoritmo de recepção SSC

Parâmetro	Valor do parâmetro
Amplitude do sinal $s(n)$	6 V
Canais simulados	Caso 1, 2, 3, 4, 5 e 6.
Comprimento do pacote <i>CEBus</i>	≈ 25 ms
Pacote <i>CEBus</i> : Control Field	0x0Ah
Pacote <i>CEBus</i> : Endereço de Destino	0x0Ah
Pacote <i>CEBus</i> : Endereço de Origem	0x16h
Pacote <i>CEBus</i> : Mensagem (ASCII)	'MESTRADO – UFJF'

A relação sinal-ruído desejada ($SNR_{desejada}$) é calculada da seguinte forma:

$$SNR_{desejada} (dB) = 10 \cdot \log_{10} \left(\frac{\sigma^2(s_f(n))}{\sigma^2(w(n))} \right) \quad (6.32)$$

onde $s_f(n)$ em (6.32) é o símbolo SSC que foi modificado pelo filtro $H(z)$. A atenuação que o símbolo sofre ao passar por cada canal $C(z)$ modelado é mostrada na Tabela 6.5, e é calculada com a equação (6.33).

$$A(dB) = 10 \cdot \log_{10} \left(\frac{\sigma^2(s_f(n))}{\sigma^2(s(n))} \right) \quad (6.33)$$

Tabela 6.5 – Atenuação média do símbolo SSC ao passar pelo canal $H(z)$

Caso	Atenuação do símbolo (dB)
S/C	0
1IS	-3.14
1I	-23.98
1A1	-33.14
1A2	-31.50
2I	-21.90
2A1	-29.15
2A2	-32.89
3I	-5.36
3A1	-21.78
3A2	-16.15
4ISH	-12.24
4IL	-25.33
4IH	-40.26
4OL	-35.16
4OH	-43.16
5I	-2.76
5O	-18.92
5OCD	-13.43
5OER	-6.47
6IER	-7.47
6ITV	-2.65
6OCD	-14.69
6OER	-8.08

O procedimento da simulação é verificar se houve erros na detecção do pacote (integridade do pacote), ou seja, verificar se os componentes do pacote *CEBus* recebido (Endereço,

Campo de Controle, etc) foram recebidos sem erros. A Tabela 6.6 mostra os resultados obtidos com as simulações. Um OK significa que o pacote foi recebido sem erros, enquanto que os erros são assinalados com um **F**. Cada erro é codificado pelo código do canal, seguido da relação sinal ruído simulada (por exemplo, um erro de detecção em 10 dB no canal 4IH gera o código de erro **F4IH10**). Já um símbolo “-” significa que não foi simulado, pelo fato do canal conter características muito semelhantes a outro simulado.

Tabela 6.6 – Resultados da simulação

Caso	20 dB	10 dB	5 dB	2 dB
SC	OK	OK	OK	OK
1IS	OK	OK	OK	OK
1I	OK	OK	OK	OK
1A1	OK	OK	OK	OK
1A2	OK	OK	F	F
2I	OK	OK	OK	OK
2A1	OK	OK	OK	OK
2A2	OK	OK	F	F
3I	OK	OK	OK	OK
3A1	OK	OK	OK	F
3A2	-	-	-	-
4ISH	OK	OK	OK	OK
4IL	OK	OK	OK	OK
4IH	F	F	F	F
4OL	OK	OK	OK	F
4OH	-	-	-	-
5I	OK	OK	OK	OK
5O	OK	F	F	F
5OCD	OK	OK	OK	OK
5OER	OK	OK	OK	OK
6IER	OK	OK	OK	OK
6ITV	OK	OK	OK	OK
6OCD	OK	OK	OK	OK
6OER	OK	OK	OK	OK

Os gráficos a seguir mostram os símbolos atenuados pela rede elétrica. Os valores da escala vertical estão em p.u. (o símbolo transmitido é igual a 1 p.u.).

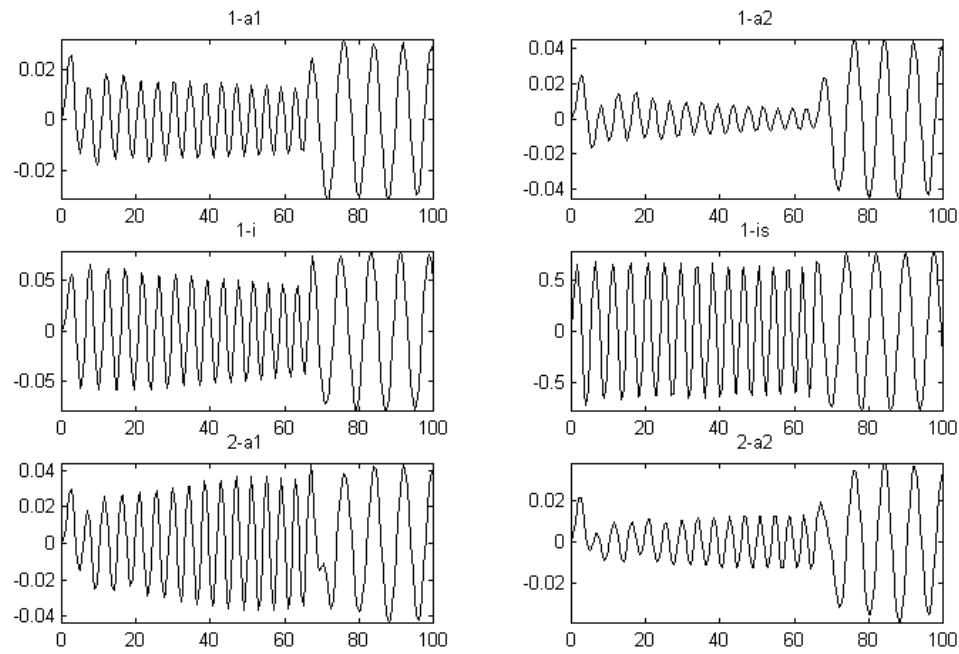


Figura 6.65 – Símbolos SSC atenuados pelos seguintes canais: 1-IS, 1-I, 1-A1, 1-A2, 2-A1 e 2-A2.

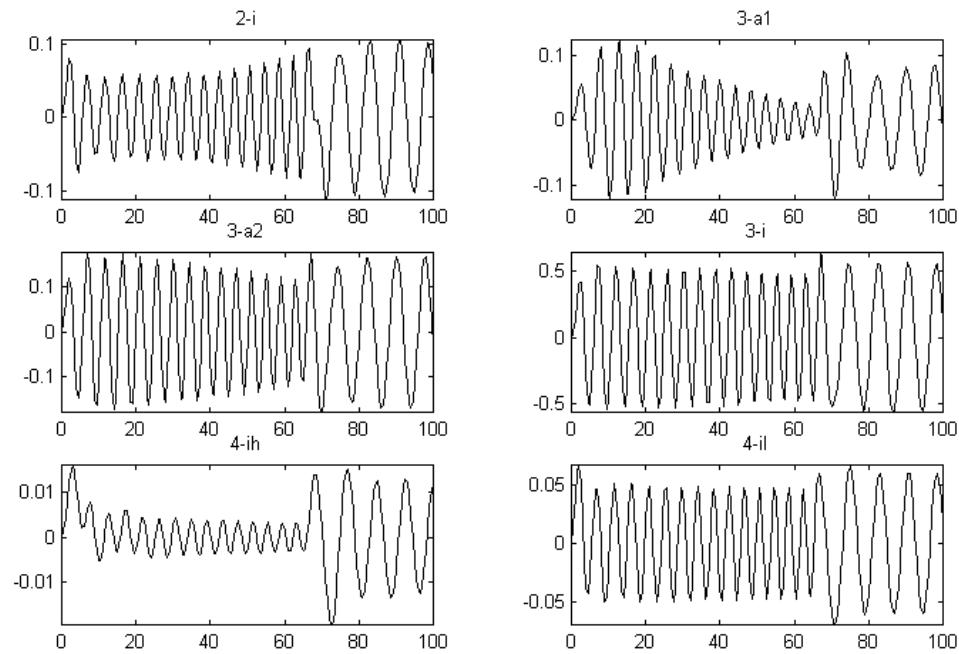


Figura 6.66– Símbolos SSC atenuados pelos seguintes canais: 2-I, 3-I, 3-A1, 3-A2, 4-IH e 4-IL.

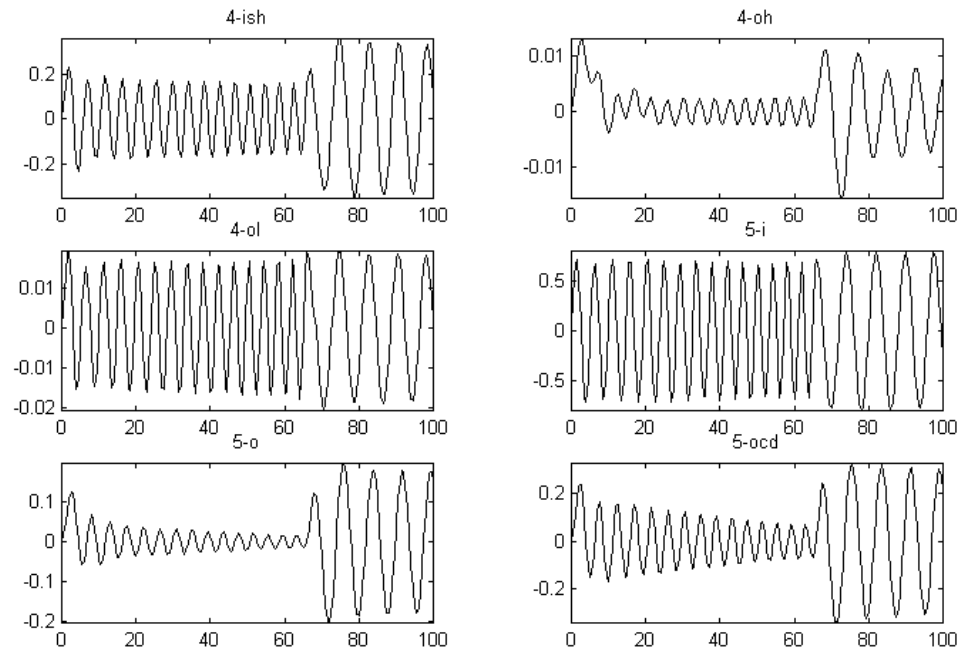


Figura 6.67– Símbolos SSC atenuados pelos seguintes canais: 4-ISH, 4-OH, 4-OL, 5-I, 5-O e 5-OCD.

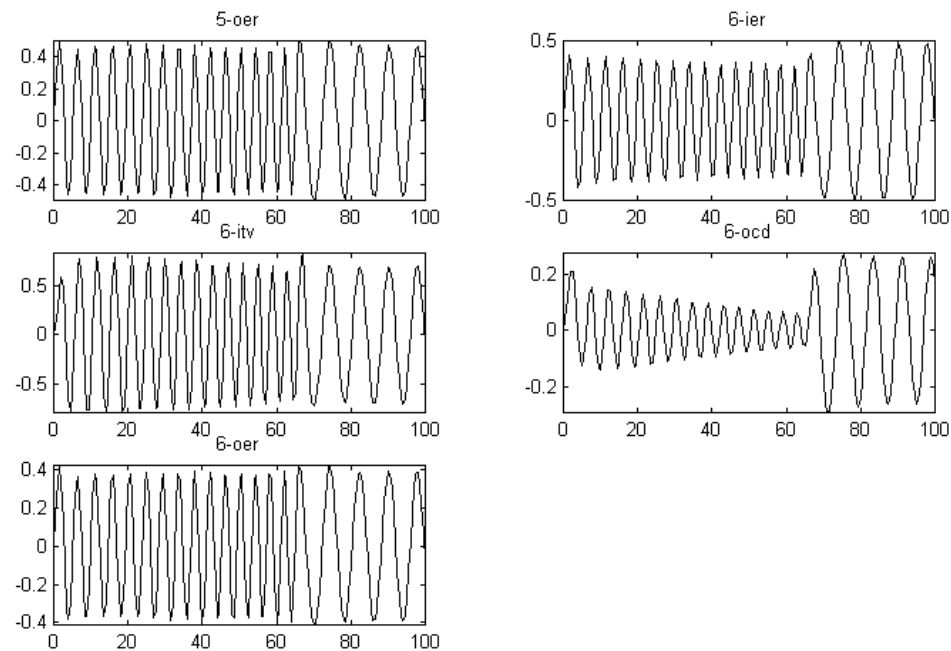


Figura 6.68 – Símbolos SSC atenuados pelos seguintes canais: 5-OER, 6-IER, 6-ITV, 6-OCD e 6-OER.

A análise dos resultados é mostrada na tabela a seguir:

Tabela 6.7 – Análise dos erros da simulação

F1-A2-05	Erro na detecção do primeiro símbolo EOF gerou campos de endereços diferentes, mas a string foi recebida sem erros.
F1-A2-02	Ruído em excesso não disparou o processo de detecção do preâmbulo (o fator K_{sinc} estava alto demais).
F2-A2-05	Erro na detecção do símbolo EOP (porém os bits do pacote foram transmitidos sem erros)
F2-A2-02	Erro na detecção do símbolo EOF
F3-A1-02	Ruído em excesso não disparou o processo de detecção do preâmbulo (o fator K_{sinc} estava alto demais).
F4-IH-20	A distorção dos símbolos <i>CEBus</i> pelo canal é excessiva
F4-IH-10	A distorção dos símbolos <i>CEBus</i> pelo canal é excessiva
F4-IH-05	A distorção dos símbolos <i>CEBus</i> pelo canal é excessiva
F4-IH-02	A distorção dos símbolos <i>CEBus</i> pelo canal é excessiva
F5-O-10	Erro na detecção do primeiro símbolo EOF gerou campos de endereços diferentes, mas a string foi recebida sem erros.
F5-O-05	Ruído em excesso mais a distorção do canal impossibilitaram o reconhecimento correto de alguns símbolos.
F5-O-02	Ruído em excesso

Como pode ser visto na Tabela 6.7, os erros em sua maioria foram causados pela severidade dos canais de comunicação, que distorcem e provocam erros de identificação dos símbolos SSC, comprometendo a integridade do pacote *CEBus*.

6.8. Otimização do “*Matched Filter*”

6.8.1. Simplificação por blocos

Seja o filtro FIR descrito em (6.8) e sua realização direta mostrada na Figura 6.43, que contém N multiplicações e $N-1$ adições. Em função do “*matched filter*” conter apenas coeficientes com valores ± 1 , os coeficientes relativos ao valor 1 podem ser retirados, e os coeficientes -1 podem ser agrupados, causando uma grande redução na complexidade computacional. Seja a Figura 6.69, que mostra os coeficientes do filtro, aonde B_i ($i=1,2,\dots,40$) são os blocos a serem agrupados.

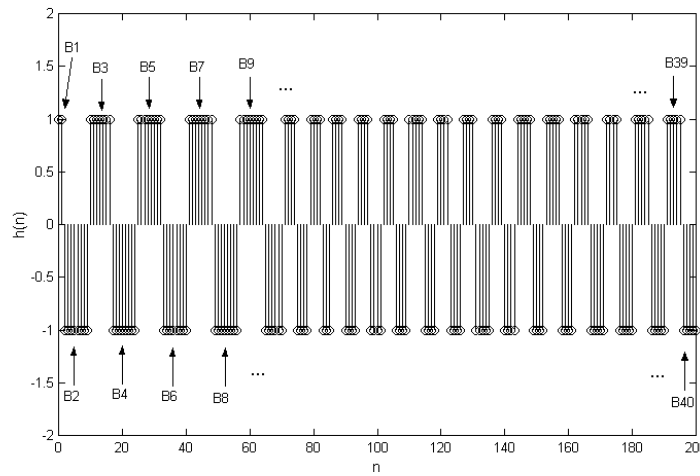


Figura 6.69 – Coeficientes do “*matched filter*”, com as marcações dos blocos.

A Figura 6.70 mostra o agrupamento dos blocos, de acordo com a Figura 6.69. Para um filtro FIR com N taps, com coeficientes ± 1 , com N_{+1} blocos iguais a 1 e N_{-1} blocos iguais a -1 , o esforço computacional pode ser calculado da seguinte forma:

$$EC_{MF} = N \text{ adições} + (N_{+1} + N_{-1} - 1) \text{ adições} \quad (6.34)$$

Para o nosso filtro, $N = 201$, $N_{+1} = 20$ e $N_{-1} = 20$, o esforço computacional é:

$$EC_{MF} = 201 \text{ adições} + (20 + 20 - 1) \text{ adições} = 240 \text{ adições} \quad (6.35)$$

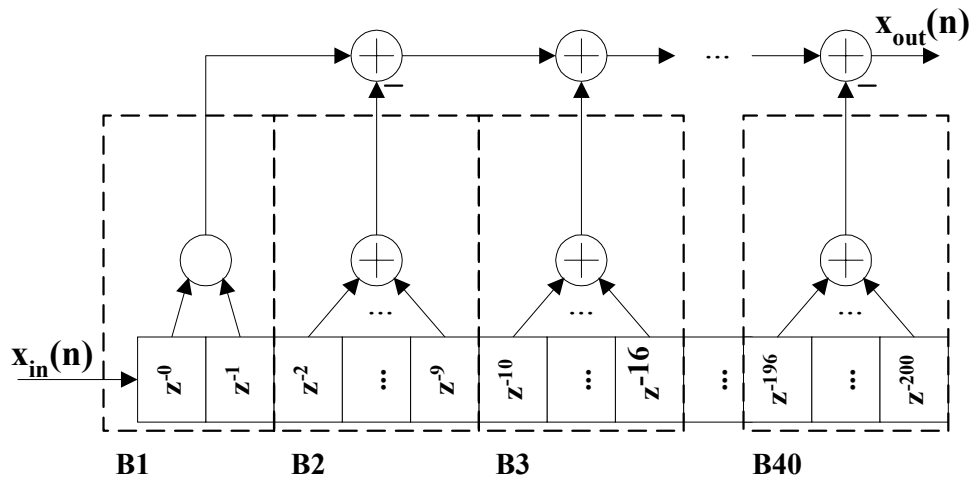


Figura 6.70 – Rearranjo dos coeficientes para otimizar a estrutura do cálculo de $H_{rc}(z)$.

6.8.2. Operações MACD

A operação anterior pode ser eficientemente implementada em processadores DSP. Por exemplo, os processadores da Texas[®] TMS320C10, C20 e C50 possuem a instrução MACD que requer apenas $N+17$ operações para computar a saída de um filtro FIR de ordem N . A instrução MACD é implementada na Figura 6.71, aonde os coeficientes do filtro ficam na memória de programa, e a memória do filtro é colocada na memória de dados. A instrução MACD incrementa o registro AR1, carregado inicialmente com o endereço da primeira amostra (memória de dados), e multiplica com o dado da memória de programa, indicado pelo *label* **coeff**. Após o ciclo de operações, a saída do filtro está no registro AR1, e a memória do filtro é atualizada.

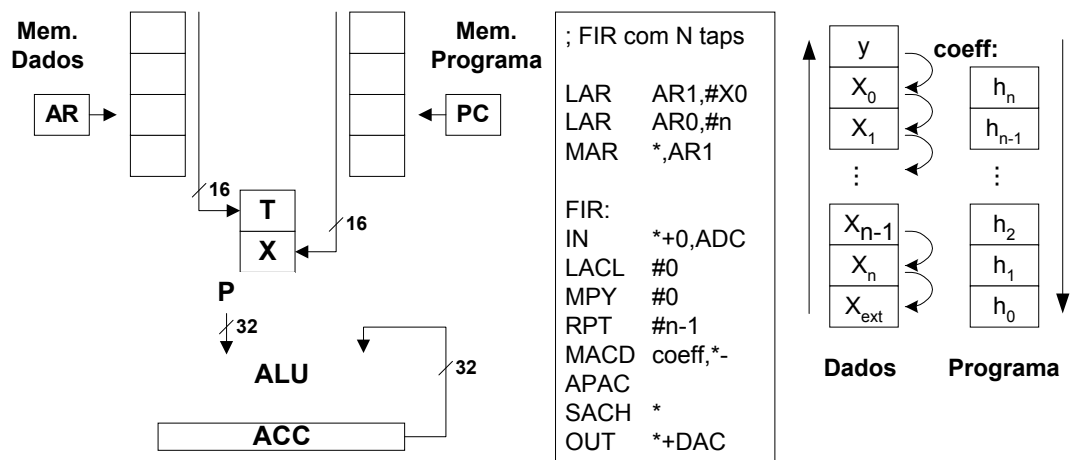


Figura 6.71 – Estrutura para o cálculo do “matched filter” utilizando a instrução MACD (somente processadores da Texas[®]).

6.8.3. Aproximação por operações XOR

Seja o filtro $H_{rc}(z)$ denominado “*matched filter*” mostrado na Figura 6.43. Uma outra realização do filtro $H_{rc}(z)$ é mostrada na Figura 6.72, para a implementação em processadores DSP de ponto fixo.

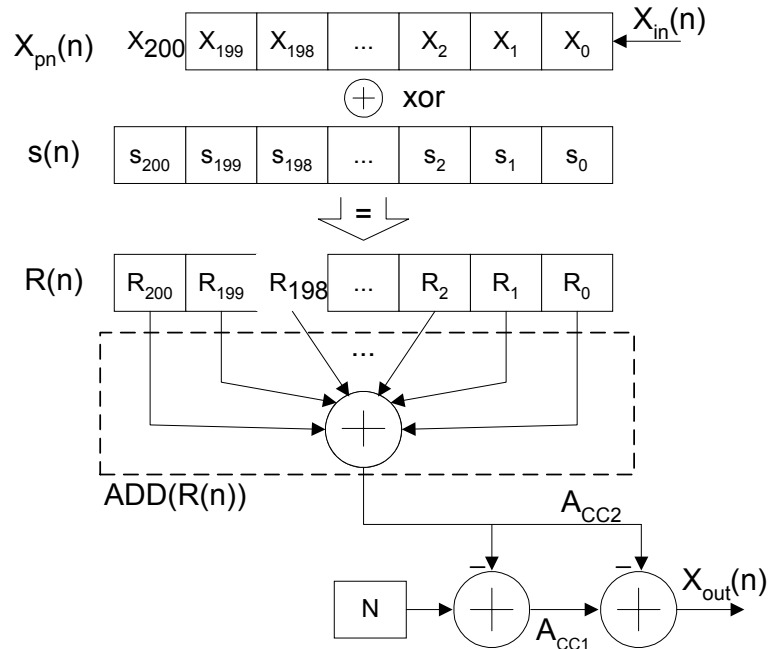


Figura 6.72 – Realização do “*matched filter*” $H_{rc}(Z)$ a partir de operações XOR

Aonde X são os dados que chegam do conversor AD, e s é a cópia local do símbolo. Os dados processados são fornecidos ao algoritmo na forma binária (1 e 0), segundo a equivalência mostrada em (2.20). O princípio de funcionamento se baseia em operações XOR entre duas “palavras” binárias de comprimento N (para $F_s = 2\text{MHz}$, $N = 201$), onde o resultado da operação XOR nos fornece os bits diferentes entre as duas palavras. Esta implementação é irrealizável na forma mostrada em Figura 6.72, devido às seguintes limitações:

- Os processadores DSP de ponto fixo tem tamanho de palavras de 16 e 32 bits;
- A operação demarcada por $\text{ADD}(R(n))$, que indica a soma de todos os bits de uma palavra binária não existe em processadores DSP.

Para tornar a estrutura realizável, é necessário “quebrar” X e s em palavras menores, que possam ser computadas. Para um processador DSP de 16 bits, o número de palavras necessárias é $N/16$. Para $N = 201$, são necessárias 13 palavras, denominadas de

wd_i e wc_i , $i = 1, 2, \dots, 13$, mostradas na Figura 6.74. As palavras wd_i representam a memória do filtro e as palavras wc_i representam os coeficientes do filtro. Como os processadores DSP são implementados com a arquitetura Harvard modificada (memória de dados $\neq$ memória de programa $\neq$ memória de IO), os coeficientes wd_i são implementados na memória de dados, que pode ser modificada, enquanto que a memória do filtro (coeficientes wc_i) é implementada na memória de programa, aonde os dados não podem ser modificados.

O algoritmo da operação $ADD(R(n))$ se baseia em uma tabela, com 256 posições, onde é fornecida a soma dos seus bits em função do resultado da operação XOR, como mostrado na Figura 6.73.

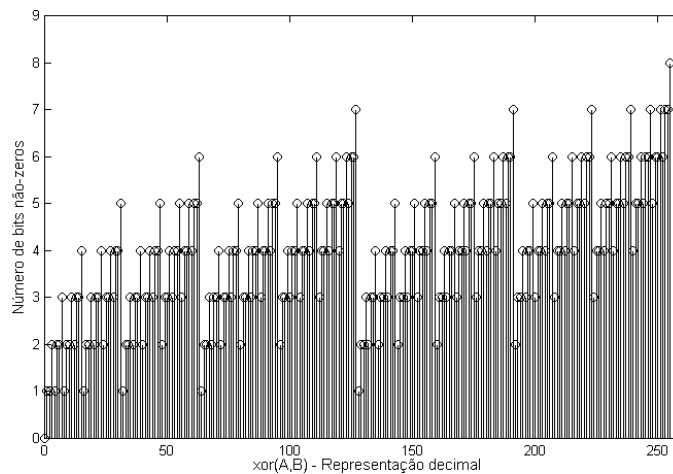


Figura 6.73 – Tabela da soma dos bits de uma operação XOR.

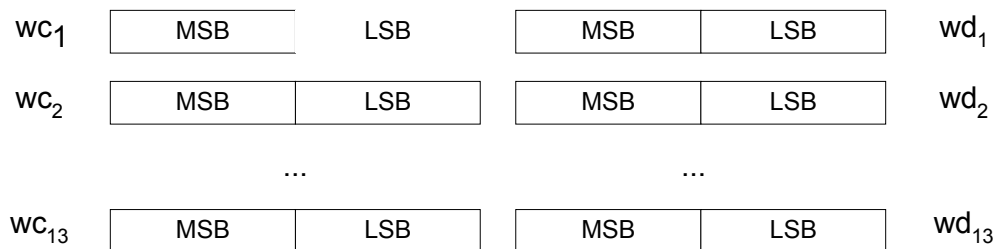


Figura 6.74 – Quebra de X e s em palavras menores (16 bits), denominadas wd_i e wc_i , respectivamente. Cada palavra é formada por 2 bytes (MSB e LSB).

O algoritmo para o cálculo de $x_{out}(n)$ é dividido em duas partes: a primeira, que consiste na atualização dos coeficientes na memória de dados, e a segunda, é o cálculo de $x_{out}(n)$ propriamente dito. Os algoritmos completos são mostrados nos fluxogramas da Figura 6.75, onde IN representa o dado amostrado no instante atual, e ACC_i é usado para calcular $x_{out}(n)$. O algoritmo da primeira parte (atualização da memória do filtro) é efetuada pelo

fluxograma da Figura 6.75a, enquanto que a segunda parte (cálculo de $x_{out}(n)$) é mostrada na Figura 6.75b.

Um detalhe é que a operação XOR de cada *word* (16 bits) é subdividida em duas partes de 8 bits (MSB e LSB) já que para 16 bits a tabela teria $0xFFFFh = 65536$ posições (com oito bits, a tabela tem $0xFFh = 256$ posições).

A justificativa para o uso do método proposto para o filtro $H_{rc}(z)$ é que ele gasta (no total) aproximadamente 160 ciclos de instruções para o cálculo de $x_{out}(n)$, enquanto que o método tradicional gasta aproximadamente 218 ciclos de instruções, ocasionando um ganho de 25% no tempo necessário para calcular $x_{out}(n)$.

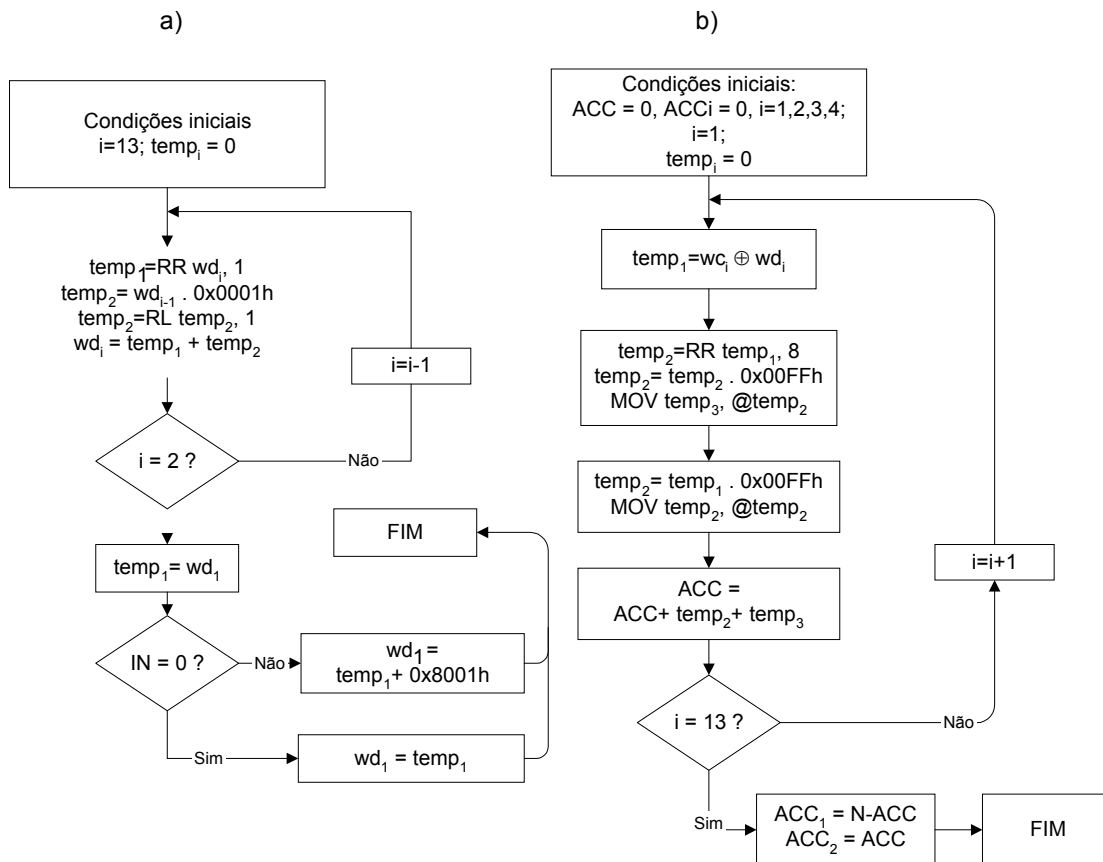


Figura 6.75 – a) Algoritmo para atualização da memória do filtro; b) Algoritmo para o cálculo de $x_{out}(n)$.

A simulação do algoritmo é mostrada na Figura 6.76, onde é comparado o cálculo da função de auto-correlação de $x_{out}(n)$ pelo método tradicional, que utiliza o “*matched filter*” proposto na Figura 6.43 e pelo método da operação XOR, proposto na Figura 6.72.

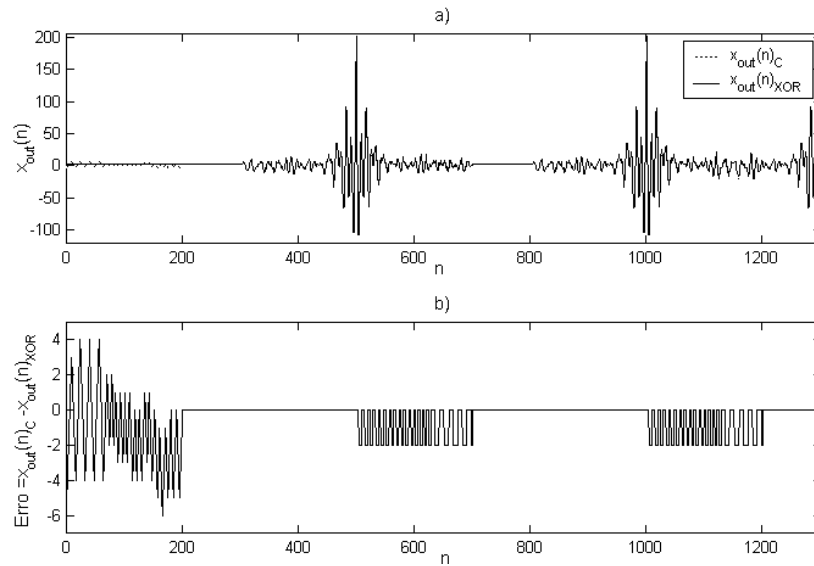


Figura 6.76 – Simulação no cálculo da função de correlação utilizando a Figura 6.72

6.9. Conclusões

Neste capítulo, as seguintes conclusões podem ser tomadas:

- o uso de técnicas de processamento digital de sinais atende os requisitos necessários para o projeto de um sistema de comunicação baseado em tecnologia SSC;
- a modelagem dos canais de comunicação por filtros digitais, embora não seja tão precisa, é de vital importância para a análise de sistemas que possam utilizar a rede eléctrica como caminho de propagação de dados;
- foram propostos, nas seções 6.5 e 6.6 deste capítulo, algoritmos de transmissão e recepção de pacotes *CEBus* para serem implementados em processadores digitais de sinais (DSP). Os algoritmos se baseiam nas características apresentadas do pacote *CEBus* e dos símbolos SSC, e foram desenvolvidos de forma a se obter um equilíbrio na relação robustez x esforços computacionais requeridos;
- com as simulações apresentadas nas seções 6.6 e 6.7, validou-se o algoritmo proposto;
- a redução computacional do “matched filter” pelos algoritmos propostos na seção 6.8, sub-seções 6.8.1, 6.8.2 e 6.8.3, foi validada pelos resultados de simulação apresentados;

7. Conclusões

Este trabalho apresentou como tema as técnicas de *Spread Spectrum Carrier*, utilizadas para a transmissão de dados em redes de energia elétrica. O desenvolvimento do trabalho se deu em duas frentes básicas: a primeira foi a construção de um protótipo de um dispositivo *powerline* (modem DLC – *Digital Line Carrier*) a partir de circuitos integrados dedicados da Intellon® e a partir de micro-controladores de baixo custo. Nesta linha de pesquisa, procurou-se dominar a tecnologia atual na transmissão de dados através de simulações e testes do modem em redes de comunicação *intra building*. Empregaram-se os programas de simulação PSPICE e MATLAB para validar o projeto do modem e adquirir o conhecimento necessário para o projeto da próxima etapa.

A segunda linha do trabalho foi o desenvolvimento de algoritmos de recepção de pacotes *CEBus* através da identificação dos sinais SSC que compõem o pacote. Para que isto fosse possível, foram utilizadas técnicas de processamento digital de sinais para o projeto dos algoritmos. O objetivo principal desta etapa foi o de identificar, projetar, otimizar e simular os algoritmos de detecção, sincronização e rastreamento do sinal SSC para serem implementados em processadores DSP.

Os principais aspectos do trabalho podem ser resumidos e são mostrados a seguir:

- a) O canal de comunicação (ou seja, a rede de energia elétrica) possui características difíceis de serem modeladas, com distorção elevada e características de atenuação aleatória e variante no tempo. Entretanto são disponibilizados, neste trabalho, alguns modelos de canais que foram úteis para a validação dos algoritmos propostos. Estes modelos foram aproximados como funções de transferências discretas, obtidas a partir de experimentos práticos;
- b) Um protótipo de um equipamento para a comunicação de dados em redes elétricas foi construído, a partir dos circuitos integrados Intellon®, e alguns testes preliminares foram efetuados. A construção envolveu várias etapas, desde o projeto de *hardware* até o projeto do *firmware* do micro-controlador do modem. Nestes testes utilizou-se o prédio do galpão do mestrado da Faculdade de Engenharia como planta-base para os experimentos. Tal planta-base é caracterizada por possuir equipamentos de ar-condicionado (bifásicos) e estações de computador em grande número. Conseguiu-se, nestes testes, transmitir dados e comandos a distâncias lineares de aproximadamente 100 metros e a taxas de 6.66 Kbps. Atenuações maiores que 10 dB não foram encontradas na planta-base;

- c) O protótipo do modem identificou dados em canais com atenuação de até 20 dB, em média. Atenuações maiores resultam em perda substancial do link de comunicação;
- d) Empregou-se, com sucesso, a tecnologia de acoplamento trifásico indutivo, que aumenta a confiabilidade do *link* de comunicação e permite a comunicação entre dispositivos *powerline* situados em fases diferentes. Tal tecnologia, derivada de acopladores *split-phase* e desenvolvida na UFJF, permite que o *link* de comunicação possa permanecer estável, mesmo na presença de dispositivos *powerline* conectados entre as fases de um sistema trifásico.
- e) A proposição de algoritmos para a recepção do pacote *CEBus*, baseados em técnicas de processamento digital de sinais, pôde ter seu desempenho validado, em função das simulações de transmissão de pacotes *CEBus* terem alta taxa de êxito. Os erros nas simulações ocorreram, em sua maioria, pela severidade excessiva na característica de atenuação dos canais de comunicação. Outros fatores de erros nas simulações são o nível de ruído excessivo, ou ainda, a combinação dos dois fatores citados anteriormente;
- f) A implementação de algoritmos alternativos para a detecção de símbolos do pacote *CEBus* se mostrou computacionalmente eficiente, gerando algoritmos que podem ser facilmente empregados em processadores DSP com um mínimo de esforço computacional;
- g) Um fator importante para a geração de algoritmos eficientes foi o uso de conversores AD de 1 bit. Com este tipo de conversão, os coeficientes dos filtros assumem valores de ± 1 , reduzindo assim o número de multiplicações para zero, além de eliminar a necessidade de conversores AD convencionais. Isto permite a redução do custo do projeto final, gerando equipamentos mais competitivos ao nível de mercado consumidor;
- h) A inclusão de equalizadores de canais pode ser uma opção bastante interessante para a redução de distorção de fase do canal, mas existem algumas dificuldades na sua implementação: a primeira é que na maioria dos casos, onde o sinal não foi recebido, foi devido a grande atenuação do canal, ou seja, este sinal possuía amplitude muito pequena para utilizar os algoritmos de equalização. Isso decorre do fato do algoritmo de detecção não conseguir detectar o pacote *CEBus*; a segunda é que o uso de algoritmos adaptativos implicam na não-utilização das estruturas de detecção projetadas, em função da perda da sequência pseudo-aleatória pela ação do filtro equalizador.

A dissertação, além de apresentar uma implementação prática, gerando um produto para o mercado abriu também vários horizontes para novas pesquisas. Algumas sugestões para novos projetos de pesquisa são:

- a) Implementação dos algoritmos propostos em processadores DSP gerando um produto aberto, onde poderão ser incluídas novas técnicas de modulação;
- b) Estudo da possibilidade do aumento da banda de transmissão de dados, através de tecnologias *multicarrier*, como por exemplo, OFDM (*Orthogonal Frequency Division Multiplexing*);
- c) Uso da técnica SSC para transmissão de dados em redes de distribuição e transmissão de energia elétrica;
- d) Estudo de técnicas que permitam a transferência do sinal SSC entre os terminais primário e secundário do transformador de distribuição, de modo eficiente e a custo reduzido;
- e) Uso de técnicas DSP para a equalização eficiente dos canais.

Apêndice A – Análise e Projeto de Um Acoplador de Fases para Sistemas SSC em Redes Trifásicas de Baixa Tensão.

Autor: Eng. Estevão Coelho Teixeira
Coordenador: Prof. Dr. Carlos Augusto Duque

A.1. Introdução

Quando dois dispositivos *CEBus* são conectados na mesma fase, existe um caminho simples entre os dois dispositivos, como mostra a Figura A.1. Em uma instalação predial ou industrial, no entanto, é muitas vezes necessária a comunicação entre dispositivos ligados em fases diferentes, como ilustra a Figura A.2. O único caminho existente é o enrolamento secundário do transformador de distribuição. Os transformadores de distribuição são projetados para um máximo rendimento na frequência de 60 Hz. Para as frequências do sinal SSC (de 100 kHz a 400 kHz), o sinal poderá ser atenuado de 30 dB ou mais, dependendo da configuração, tamanho e carregamento do transformador. Em muitos casos a atenuação provocada pelo transformador combinada com outros ruídos é suficiente para impedir a comunicação entre dois dispositivos.

É necessária a utilização, portanto, de um dispositivo de acoplamento que permita uma perfeita comunicação entre dois dispositivos *CEBus* ligados em fases diferentes da rede. Neste trabalho é relatado o projeto de um acoplador de fases para sistemas de comunicação utilizando a tecnologia SSC adaptado às redes de trifásicas com neutro, empregadas na distribuição de energia elétrica para consumidores em baixa tensão. O desenvolvimento partiu de um acoplador utilizado em sistema de distribuição monofásico, utilizado mais comumente em redes residenciais nos Estados Unidos [3]. O funcionamento deste transformador é brevemente explicado na próxima seção, e é o ponto de partida para o desenvolvimento do acoplador trifásico.

A.2. O Acoplador para Rede de Distribuição Monofásica

As redes de distribuição monofásicas em baixa tensão são usualmente alimentadas a partir de um transformador monofásico com derivação central, ou a três fios, que possui duas bobinas secundárias normalmente ligadas em série. Entre as bobinas há uma derivação central, como mostra a Figura A.3. Por simplicidade, na Figura as bobinas secundárias são representadas como uma única bobina com derivação central. Esta derivação central é aterrada, e constitui-se o neutro da rede secundária. As tensões fase-

neutro V_1 e V_2 são defasadas de 180° entre si. Os valores nominais de tensão secundária para estes transformadores são, normalmente, 120 V para tensão fase-neutro e 240 V para tensão entre fases. Estes transformadores são utilizados, no Brasil, em redes de distribuição rurais.

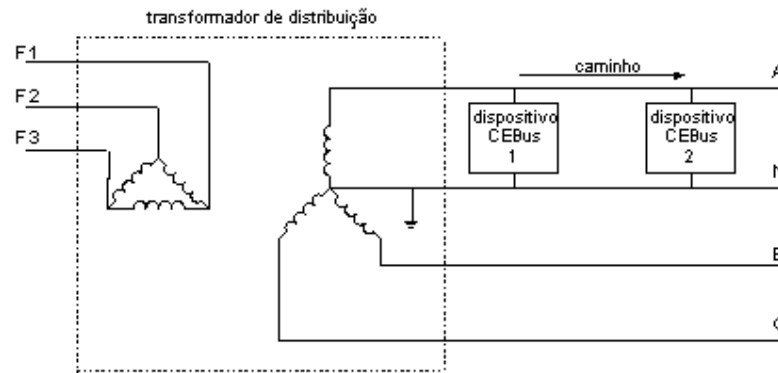


Figura A.1 – Dispositivos *CEBus* ligados na mesma fase

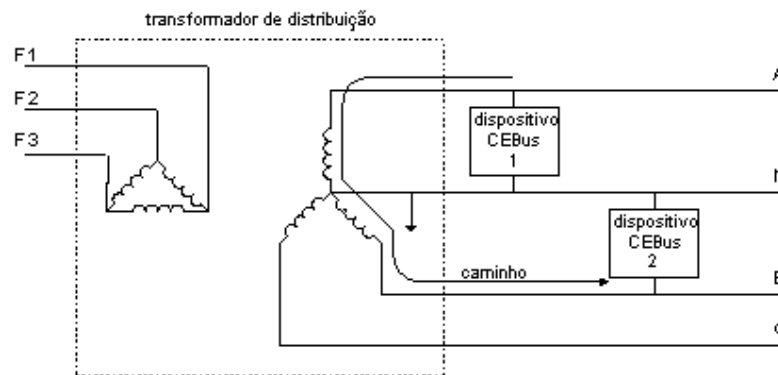


Fig. A.2 – Dispositivos *CEBus* ligados em fases diferentes

A Figura A.4 mostra esquematicamente dois dispositivos CEBus ligados na mesma fase. Para este caso, semelhantemente à Fig. A.1, existe um caminho direto para o sinal. A Figura A.5 mostra dois dispositivos CEBus ligados em fases diferentes. Neste caso, haverá atenuação, visto que o caminho para o sinal passa pelo secundário do transformador de distribuição. É necessário, neste caso, o uso de um dispositivo de acoplamento, como esquematiza a Figura A.6, com característica de filtro passa-alta, que possibilite um caminho para o sinal nas frequências de interesse (100 kHz – 400 kHz), e por outro lado bloqueie a passagem de sinal em baixas frequências. Neste caso, a simples utilização de um capacitor de valor apropriado seria viável.

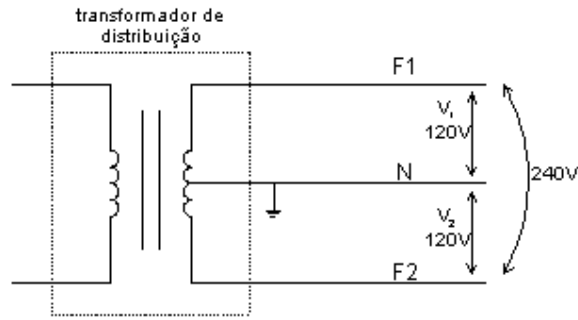


Figura A.3 – Transformador de distribuição monofásico com derivação central (três fios)

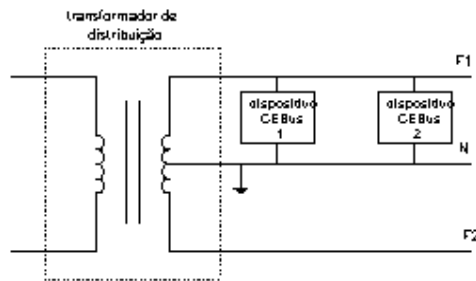


Figura. A.4 – Dispositivos CEBus ligados na mesma fase

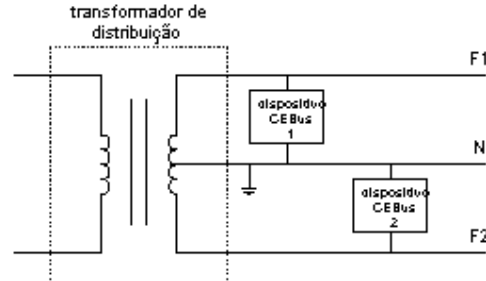


Fig. A.5 – Dispositivos CEBus ligados em fases opostas

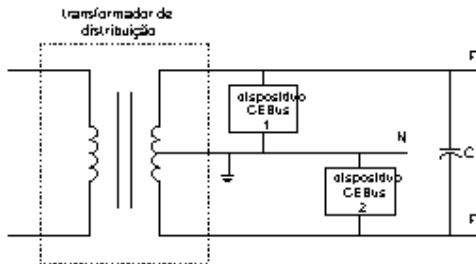


Figura A.6 – Uso de um capacitor como dispositivo de acoplamento entre dois dispositivos CEBus ligados em fases diferentes

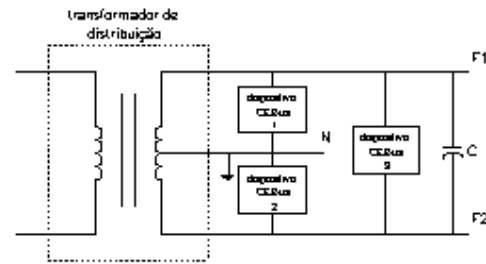


Figura A.7 – Caso um dispositivo CEBus esteja ligado entre fases, o capacitor irá curto-circuitar o sinal enviado ou recebido por este dispositivo

No entanto, se um dispositivo *CEBus* estiver ligado entre fases (240 V), como ilustra a Figura A.7, a presença do capacitor de acoplamento causaria um curto-circuito para o sinal, uma vez que a reatância do capacitor oferecerá um caminho de baixa impedância para o sinal injetado entre as fases.

Uma solução para este problema é a utilização de um acoplador com dois capacitores e um pequeno transformador de acoplamento com relação de espiras 1:1, como ilustra a Figura A.8. Note que o ponto neutro é conectado em comum às duas bobinas do

transformador. Para a comunicação entre dois dispositivos ligados em fases distintas, como na Fig. A.5, o sinal será acoplado magneticamente.

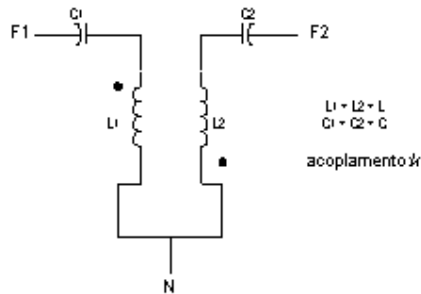


Figura A.8 – Acoplador avançado para o sistema de distribuição monofásico com derivação central

Considere um sinal aplicado entre a fase F1 e o neutro. O capacitor C1 é calculado de forma a oferecer uma baixa impedância nas frequências de interesse, de forma que, para o sinal SSC é visto como um curto-circuito. O sinal surge, então, nos terminais da bobina L1. Por indução eletromagnética, este sinal surge nos terminais da bobina L2, com fase oposta, como mostram os pontos na figura. Como o capacitor C2 também é visto como um curto-circuito nas frequências de interesse, o sinal resultante em L2 é aplicado ao dispositivo ligado entre a fase F2 e o neutro. Embora a fase do sinal resultante em F2 seja inversa à polaridade do sinal aplicado em F1, o protocolo *CEBus* permite que haja inversão de fase no sinal sem que haja perda ou distorção da informação.

O transformador de acoplamento, que deverá ser construído com núcleo de ferrite, oferece baixa impedância ao sinal da rede elétrica, em 60 Hz. A tensão da rede (120 V, 60 Hz) estará aplicada aos terminais dos capacitores de acoplamento. Recomenda-se ligar em paralelo com os capacitores resistores de valor elevado (470 k Ω é um valor típico) para descarregar os capacitores e prevenir choques elétricos quando o acoplador não estiver sendo utilizado.

Na prática, C1 e C2 possuem capacitância C e as bobinas L1 e L2 possuem indutância igual a L. Considerando o coeficiente de acoplamento entre as bobinas igual a k, temos que a indutância mútua entre as bobinas L1 e L2 será:

$$M = k \cdot \sqrt{L_1 \cdot L_2} = k \cdot \sqrt{L^2} = k \cdot L \quad (\text{A.1})$$

Estas considerações são importantes para o projeto de um acoplador para o sistema trifásico, como será mostrado na próxima seção.

A.3. Acoplador para Rede Trifásica com Neutro – Propostas Preliminares

O projeto do acoplador de fases para operar em uma rede de distribuição secundária trifásica com neutro é baseado na análise feita anteriormente para o acoplador para rede alimentada por transformador monofásico com derivação central. A proposta inicial foi a utilização de três bobinas acopladas magneticamente, como mostra a Figura A.9. Cada bobina tem uma de suas extremidades conectada a uma das fases por meio de um capacitor de acoplamento. A outra extremidade de cada uma das três bobinas é conectada ao neutro. Considere todos os capacitores com capacitância C e os indutores com indutância L .

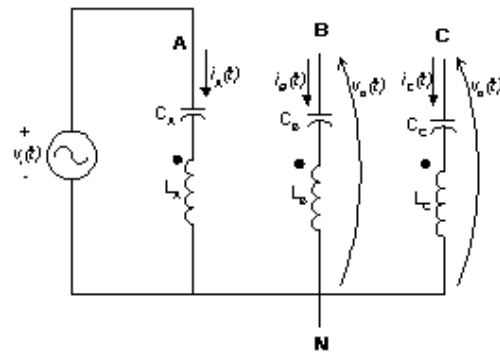
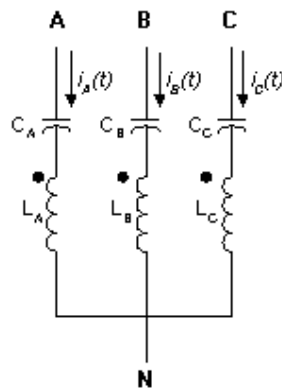


Figura A.9 – Acoplador para rede trifásica – proposta inicial

Figura A.10 – Se um sinal $v_i(t)$ for aplicado na fase A, teremos um sinal $v_o(t)$ nas fases B e C

Se um sinal $v_i(t)$ for aplicado entre a fase A e o neutro, obteremos sinais de saída $v_o(t)$ nas fases B e C, como ilustra a Figura A.10. O resultado será idêntico ao que seria obtido para o acoplador da seção anterior para bobinas com mesma polaridade. No entanto, se o acoplador da Fig. A.9 for empregado ligando-se o dispositivo transmissor, TX, entre fases (A e B, por exemplo) e o dispositivo receptor, RX, na outra fase (fase C), a impedância de entrada vista por TX será um valor muito baixo, para as frequências de interesse. Além disto, as tensões induzidas na bobina ligada à fase C devido às correntes nas bobinas ligadas às fases A e B, respectivamente, terão polaridades opostas. A tensão resultante induzida na fase C será, portanto, igual a zero. O acoplador apresentado na Fig. A.9 não pode ser utilizado no caso de um dos dispositivos ser ligado entre fases.

Uma alternativa para sistemas contendo dispositivos ligados entre duas das três fases do sistema é mostrada na Figura A.11. A ligação de uma das bobinas é invertida, de forma que um dispositivo TX ligado entre as fases A e B, por exemplo, verá uma impedância de entrada que não irá drenar do dispositivo uma corrente elevada. O sinal induzido na fase

C será aproximadamente a metade do sinal aplicado entre A e B. No entanto, é necessário que o dispositivo ligado entre fases esteja sempre ligado em fases que possuem bobinas do acoplador com polaridade invertida. Caso um dispositivo RX fosse ligado entre A e C, por exemplo, não teríamos sinal induzido em B.

Uma solução satisfatória para este problema é apresentada na próxima seção. O circuito a ser analisado corresponde ao projeto definitivo do acoplador para a rede trifásica.

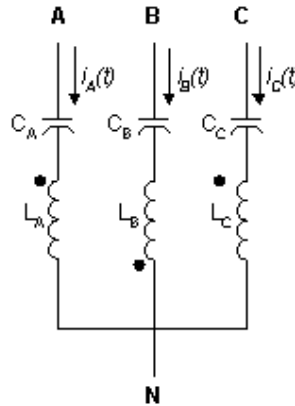


Figura A.11 – Modificação no circuito da Figura A.9

A.4. Acoplador para Rede Trifásica com Neutro – Circuito Modificado

O acoplador a ser analisado nesta seção é uma modificação do circuito mostrado na Figura A.11. Em uma das fases, é adicionada mais uma bobina, com auto-indutância também igual a L , magneticamente acoplada com as demais. O circuito é apresentado na Figura A.12.

Este circuito se propõe a permitir o acoplamento entre dispositivos ligados em qualquer uma das três fases, fase-neutro ou entre fases. É necessário que a ligação de uma das bobinas ainda se mantenha invertida, como mostra a figura.

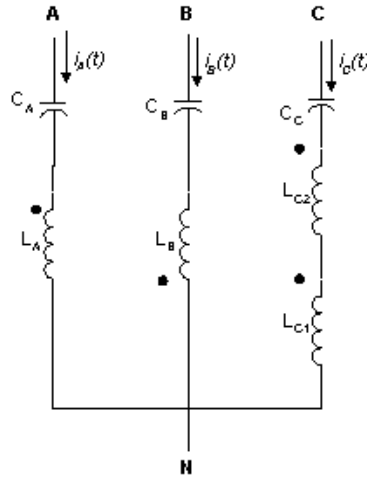


Figura A.12 – Acoplador para rede trifásica – circuito modificado

A.4.1) Acoplador para sistema trifásico – funções de transferência e impedâncias de entrada

O circuito da Fig. A.11 apresentará diferentes funções de transferência $V_o(s)/V_i(s)$, dependendo da fase ou das fases às quais estiverem ligados os dispositivos TX e RX. O sistema de equações que descreve o circuito é dado a seguir:

$$\begin{bmatrix} \left(\frac{1}{Cs} + Ls \right) & -kLs & 2kLs \\ -kLs & \left(\frac{1}{Cs} + Ls \right) & -2kLs \\ 2kLs & -2kLs & \left(\frac{1}{Cs} + 2(1+k)Ls \right) \end{bmatrix} \begin{bmatrix} I_A(s) \\ I_B(s) \\ I_C(s) \end{bmatrix} = \begin{bmatrix} V_A(s) \\ V_B(s) \\ V_C(s) \end{bmatrix} \quad (\text{A.2})$$

A partir de (A.2), obtém-se as funções de transferência para diferentes configurações dos dispositivos RX e TX. As impedâncias de entrada para cada caso também são calculadas a partir de (A.2). Os diferentes casos são resumidos na Tabela A.1 a seguir, e podem ser facilmente demonstrados.

TABELA A.1 – Acoplador modificado – funções de transferência

Entrada, $V_i(s)$	Saída, $V_o(s)$	Função de transferência, $V_o(s)/V_i(s)$	Impedância de entrada, $Z_i(s)$
1º caso: Dispositivo TX entre fase e neutro (F_1 -N) Dispositivo RX entre fase e neutro (F_2 -N, F_3 -N)			
A-N	B-N	$\frac{-kLC s^2}{1+LC s^2}$	$\frac{1}{Cs} + Ls$
	C-N	$\frac{2kLC s^2}{1+LC s^2}$	
B-N	A-N	$\frac{-kLC s^2}{1+LC s^2}$	$\frac{1}{Cs} + Ls$
	C-N	$\frac{-2kLC s^2}{1+LC s^2}$	
C-N	A-N	$\frac{2kLC s^2}{1+2(1+k)LC s^2}$	$\frac{1}{Cs} + 2(1+k)Ls$
	B-N	$\frac{-2kLC s^2}{1+2(1+k)LC s^2}$	
2º caso: Dispositivo TX entre fases (F_1 - F_2) Dispositivo RX entre fase e neutro (F_3 -N)			
A-B	C-N	$\frac{2kLC s^2}{1+(1+k)LC s^2}$	$2\left[\frac{1}{Cs} + 2(1+k)Ls\right]$
B-C	A-N	$\frac{-1,5kLC s^2}{1+1,5(1+k)LC s^2}$	$\frac{2}{Cs} + 3(1+k)Ls$
A-C	B-N	$\frac{kLC s^2}{1+1,5(3-2k)LC s^2}$	$\frac{2}{Cs} + (3-2k)Ls$
3º caso: Dispositivo TX entre fase e neutro (F_1 -N) Dispositivo RX entre fases (F_2 - F_3)			
A-N	B-C	$\frac{-3kLC s^2}{1+LC s^2}$	$\frac{1}{Cs} + Ls$
B-N	A-C	$\frac{kLC s^2}{1+LC s^2}$	$\frac{1}{Cs} + Ls$
C-N	A-B	$\frac{4kLC s^2}{1+2(1+k)LC s^2}$	$\frac{1}{Cs} + 2(1+k)Ls$

A.4.2) Diagramas de Bode

Observa-se que todas as funções de transferência são do tipo

$$\frac{a s^2}{1 + b s^2} \quad (\text{A.3})$$

Onde os coeficientes a e b irão depender das fases às quais estiverem ligados os dispositivos TX e RX. Fazendo $s = j\omega$, temos:

$$\frac{a (j\omega)^2}{1 + b (j\omega)^2} \quad (\text{A.4})$$

O diagrama de Bode para uma função deste tipo é dado a seguir:

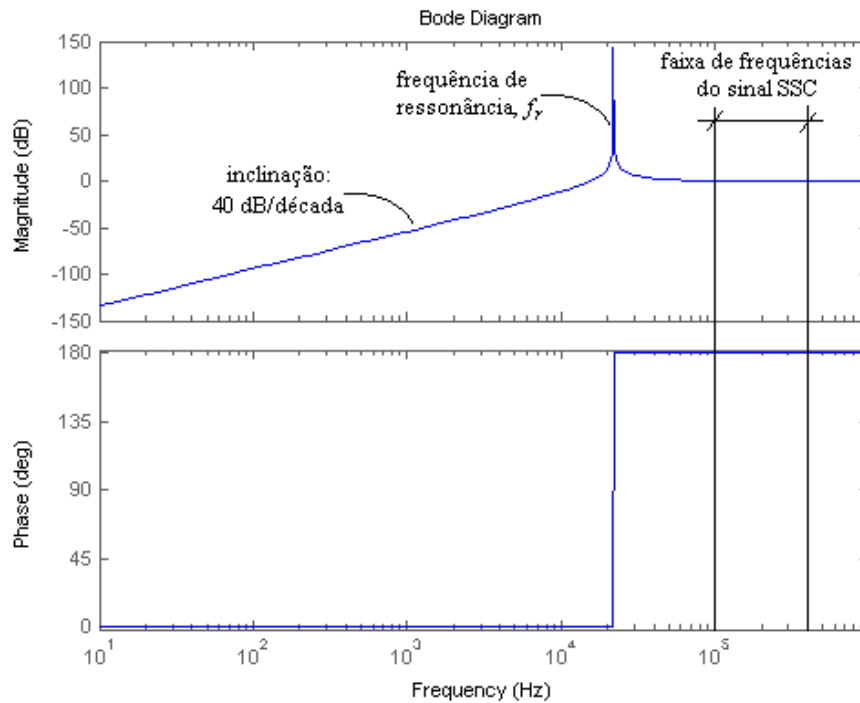


Fig. A.14 – Diagrama de Bode para uma função de transferência do tipo apresentado em (A.5)

Nota-se um pico na *frequência de ressonância* do sistema, f_r , dada por

$$f_r = \frac{1}{2\pi\sqrt{b}} \quad (\text{A.5})$$

Abaixo de f_r , o gráfico acompanha uma assíntota com inclinação de 40 dB/década. Acima de f_r , o gráfico mantém-se praticamente em linha reta, significando um ganho aproximadamente constante acima desta frequência.

Na prática, a presença de elementos dissipativos (resistências dos enrolamentos, resistores de descarga dos capacitores, resistência da linha, dentre outros) faz com que o pico de ressonância seja reduzido. No entanto, observa-se, na tabela I, que os valores de impedância de entrada são iguais a zero na frequência de ressonância. Portanto, um dos procedimentos de projeto a serem tomados é escolher os componentes C e L de forma que, para todos os casos, a frequência de ressonância situe-se suficientemente abaixo da faixa de frequências do sinal SSC. Por outro lado, ela deve se situar suficientemente acima de uma determinada harmônica do sinal de 60 Hz, de forma a garantir que a presença de harmônicas de ordem inferior seguramente não causem interferência de uma fase em outra.

O projeto completo do acoplador para rede trifásica desenvolvido é apresentado na próxima seção.

A.5 – Projeto do Acoplador para Rede Trifásica com Neutro

Esta seção apresenta os procedimentos de cálculo dos valores de L e C para o acoplador para rede trifásica apresentado na Figura A.12. O valor exato da indutância das bobinas do acoplador não é crítico, desde que seja respeitado o critério referido na seção anterior quanto à frequência de ressonância do dispositivo.

Para estimar o valor da indutância a ser utilizado, tomamos por base a indutância das bobinas do transformador de acoplamento utilizada no modem ATI/UFJF DLC P300, cujo valor é aproximadamente 0,125 mH.

Para o acoplador trifásico, optou-se também por um núcleo de ferrite tipo T (toroidal), cujas dimensões permitam o enrolamento de 4 bobinas. Foi escolhido um núcleo Thornton® [44] tipo T, código NT-27/16/12-2700-IP6, cujas especificações são as seguintes:

AL (fator de indutância): 2700 nH

Tolerância: $\pm 25\%$

Material tipo IP6

Dimensões: D = 27,0 mm (diâmetro externo)

d = 16,0 mm (diâmetro interno)

h = 12,0 mm (espessura)

Peso: ~ 23,50 g

O fator de indutância de um determinado núcleo representa a indutância por unidade de espira. Uma expressão aproximada para AL é:

$$AL = \frac{L}{N^2} \quad (A.6)$$

Através de (A.6), podemos calcular o número de espiras que nos permite chegar ao valor de indutância desejado. Para o núcleo escolhido e a indutância base utilizada, temos:

$$N^2 = \frac{L}{AL} = \frac{0,125 \text{ mH}}{2700 \text{ nH}} = 46,30 \quad (A.7)$$

$$N \cong 6,8 \rightarrow 7 \text{ espiras} \quad (A.8)$$

Assim, foram enroladas 4 bobinas no núcleo especificado com 7 espiras cada, utilizando fio # 26 AWG isolado. A medição experimental da indutância de cada bobina levou a um valor de 0,1624 mH em 100 kHz. A medida estimada para o fator de acoplamento k conduz a um valor superior a 0,97.

Os capacitores empregados são de 330 nF / 400 V (poliéster). Em paralelo com cada capacitor, foi conectado um resistor de 470 k Ω / 1W, para permitir a descarga dos capacitores quando o acoplador não estiver sendo utilizado.

Para estes valores de C, L e k obtivemos, de acordo com a tabela I, as seguintes frequências de ressonância, e os valores *aproximados* de ganho na frequência do sinal SSC (100 kHz – 400 kHz) para os diferentes pontos de aplicação do sinal do dispositivo TX e recepção do sinal nos dispositivos RX, listados na tabela II. Ressalta-se que estamos considerando aqui a impedância de saída de TX igual a zero, e a impedância de entrada de RX infinita (caso ideal).

TABELA A.2 – Ganhos aproximados e frequências de ressonância para o acoplador modificado

Entrada, V_i	Saída, V_o	Ganho $ V_o(s)/V_i(s) $ (aprox.)	Ganho em dB (aprox.)	Frequência de ressonância, f_r
1º caso: Dispositivo TX entre fase e neutro (F_1 -N) Dispositivo RX entre fase e neutro (F_2 -N, F_3 -N)				
AN	BN	1	0	21,7 kHz
	CN	2	6	
BN	AN	1	0	21,7 kHz
	CN	2	6	
CN	AN	0,5	-6	11 kHz
	BN	0,5	-6	
2º caso: Dispositivo TX entre fases (F_1 - F_2) Dispositivo RX entre fase e neutro (F_3 -N)				
AB	CN	1	0	15,5 kHz
BC	AN	0,5	-6	12,6 kHz
AC	BN	0,61	-4,3	17,2 kHz
3º caso: Dispositivo TX entre fase e neutro (F_1 -N) Dispositivo RX entre fases (F_2 - F_3)				
AN	BC	3	9,5	21,7 kHz
BN	AC	1	0	21,7 kHz
CN	AB	1	0	11 kHz

A menor frequência de ressonância obtida (11 kHz) é para o caso do dispositivo TX ligado entre a fase C e o neutro. Este valor é 6,1 vezes maior que a 30ª harmônica do sinal ac da rede elétrica em 60 Hz. Por sua vez, a maior frequência obtida (21,7 kHz) é 4,6 menor que a menor frequência do sinal SSC. Para a frequência de 60 Hz, a atenuação do sinal é de, *no mínimo*, 90 dB (para TX ligado entre fase C e neutro). A verificação destes parâmetros valida a escolha dos componentes utilizados.

Embora se constate a diferença nos ganhos, dependendo dos pontos de aplicação e recepção do sinal SSC, este dispositivo permite o acoplamento de dispositivos CEBus independentemente da fase ou das fases às quais estiverem conectados. Se, todavia, os dispositivos CEBus forem conectados apenas entre fase e neutro, o circuito apresentado na Fig. A.10 representa uma proposta mais adequada que o circuito apresentado na Fig. A.12, por proporcionar os mesmos valores de ganho para quaisquer fases às quais estiverem ligados os dispositivos. A única diferença entre os dois circuitos é a presença de mais uma

bobina ligada na fase C. Assim, recomenda-se curto-circuitar a bobina L_{C2} caso o sistema utilize apenas dispositivos ligados entre fase e neutro.

A Figura A.15 apresenta o circuito do acoplador, incluindo uma chave (S_1) que permite ao usuário curto-circuitar a bobina L_{C2} , além de representar aqui os resistores de $470\text{ k}\Omega$ para descarga dos capacitores. A chave tripolar S_2 permite ao usuário desconectar o acoplador da rede, se necessário. Este circuito serve como base para o projeto de uma placa que permita que o dispositivo seja devidamente instalado no quadro de distribuição de energia, local mais recomendado para a instalação do acoplador. A confecção desta placa é um trabalho ainda a ser desenvolvido.

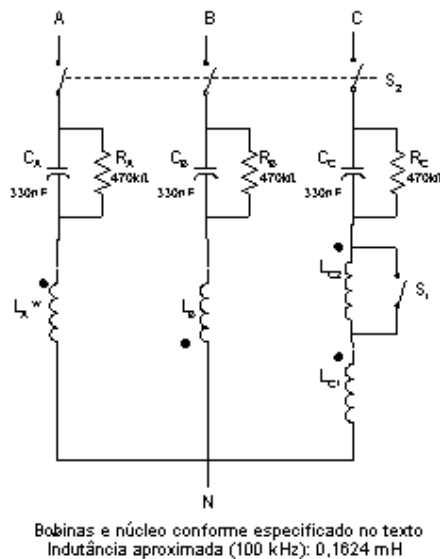


Fig. A.15 – Acoplador para rede trifásica – circuito modificado. Este circuito apresenta os resistores de descarga dos capacitores, a chave para curto-circuitar L_{C2} e as chaves para desligar o acoplador da rede

A.6 – Simulação do Circuito

Nesta seção são apresentados resultados obtidos através de simulação do acoplador modificado utilizando o programa PSpice®. Na simulação, os valores adotados para a resistência de saída da fonte, r_s , e para a resistência da carga, R_L são, respectivamente, $1\ \Omega$ e $10\text{ k}\Omega$. A amplitude do sinal aplicado é de 3 V. O circuito foi simulado com os resistores de descarga dos capacitores. É representada ainda a chave S_1 , mostrada na Fig. 5.1. Foram escolhidas 8 das 12 configurações possíveis, simuladas para frequências de 100 kHz, 200 kHz e 400 kHz, para dois casos: com S_1 fechada e com S_1 aberta.

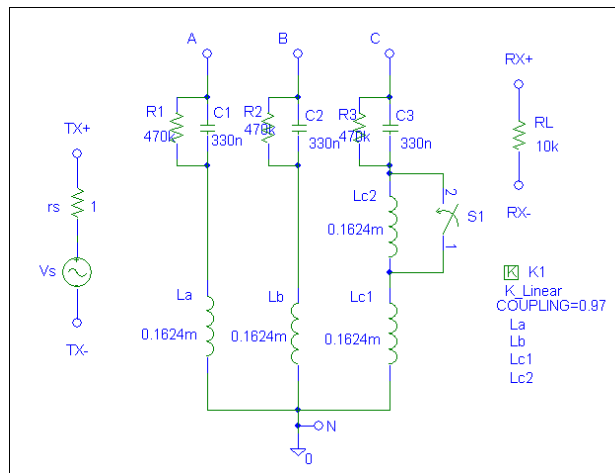


Fig. A.16 – Acoplador para rede trifásica modificado – circuito para simulação no PSpice

Os resultados são apresentados nas Tabelas A.3 e A.4, a seguir. Os valores mostrados são referentes à tensão de pico obtida em RX.

TABELA A.3 – Resultados de simulação – chave S_1 fechada; $V_i = 3$ V (pico)

TX	RX	Tensão de saída, em volts (pico)		
		$f = 100$ kHz	$f = 200$ kHz	$f = 400$ kHz
AN	BN	5,6477	1,8348	1,5524
	CN	5,6483	1,8348	1,5447
CN	AN	5,5818	1,8343	1,5440
AB	CN	2,1645	1,1624	1,0308
AC	BN	7,9507e-03	3,8695e-03	1,3099e-03
NA	BC	11,283	3,6691	3,1045
BN	AC	18,451e-03	2,9940e-03	1,2607e-03
CN	AB	11,150	3,6681	3,0884

TABELA A.4 – Resultados de simulação – chave S_1 aberta; $V_i = 3$ V (pico)

TX	RX	Tensão de saída, em volts (pico)		
		$f = 100 \text{ kHz}$	$f = 200 \text{ kHz}$	$f = 400 \text{ kHz}$
AN	BN	3,0563	2,9460	2,9194
	CN	6,1078	5,8887	5,8259
CN	AN	1,5372	1,5029	1,4853
AB	CN	3,1035	3,0104	2,9731
AC	BN	3,0180	2,8098	2,7564
AN	BC	9,1538	8,8271	8,7443
BN	AC	3,0564	2,9461	2,9143
CN	AB	3,0601	2,9986	2,9696

Apêndice B – Coeficientes dos filtros digitais dos canais de comunicação

Seja a equação (6.16), que representa um filtro digital IIR e que é reproduzida a seguir:

$$C(z) = \frac{\sum_{i=0}^{L-1} b_i \cdot z^{-i}}{\sum_{i=0}^{M-1} a_i \cdot z^{-i}} \quad (\text{B.1})$$

onde a_i e b_i em (B.1) são os coeficientes do denominador e do numerador do filtro, respectivamente. As aproximações dos canais de comunicação pela função *yulewalk* do software Matlab gera os coeficientes dos filtros para cada caso. Os coeficientes são listados nas tabelas abaixo, com oito casas decimais após a vírgula. Cada tabela fornece o conjunto de coeficientes de dois filtros, totalizando um conjunto de 23 canais de comunicação.

Tabela B.1 – Coeficientes dos filtros digitais dos casos 1IS e 1I

Casos:	Caso 1 (IS)		Caso 1 (I)	
i	b_i	a_i	b_i	a_i
0	0.41614651	1	0.00598897	1
1	-2.04871581	-5.47983157	-0.03364275	-7.73893271
2	4.86519710	14.60000295	0.09141311	28.56965355
3	-7.03465420	-24.29371055	-0.15462684	-65.89916590
4	5.79943852	25.66283168	0.16667237	103.12088001
5	-0.28547496	-12.36733921	-0.08576527	-107.48490512
6	-6.64077754	-11.15191107	-0.06168912	56.92076460
7	10.26802746	30.76615974	0.18791294	32.16218279
8	-8.15512461	-33.90895182	-0.21488324	-109.34224609
9	2.29674686	20.26773268	0.14005589	129.14463042
10	2.90608771	-1.16870657	-0.03020893	-87.86174032
11	-4.55467787	-11.00148634	-0.04441336	22.31914278
12	3.07933424	12.17979835	0.06211754	24.89980388
13	-0.80884150	-6.81314047	-0.04574783	-37.02813132
14	-0.49262042	1.36475732	0.02575092	25.28006323
15	0.62920444	1.06811280	-0.01459075	-9.33278955
16	-0.28948724	-1.09986970	0.00951315	0.24941118
17	0.03362166	0.45811587	-0.00586923	1.86122850
18	0.03049340	-0.07488059	0.00272985	-1.12258650
19	-0.01456016	-0.00992156	-0.00083526	0.32330915
20	0.00173674	0.00402353	0.00013686	-0.04042976

Tabela B.2 – Coeficientes dos filtros digitais dos casos 1A1 e 1A2

Casos:	Caso 1 (A1)		Caso 1 (A2)	
i	b_i	a_i	b_i	a_i
0	0.00571091	1	0.00309485	1
1	-0.03229665	-7.06355324	-0.01908198	-8.07932904
2	0.08635548	23.52090795	0.05622765	31.02760955
3	-0.13889082	-48.04919900	-0.10147167	-74.08688568
4	0.13269858	64.06198700	0.11537422	119.27678587
5	-0.03688842	-50.35586051	-0.06351009	-126.95448681
6	-0.10435725	3.37968683	-0.04025281	67.87906144
7	0.19561022	51.98828726	0.13160070	39.45724660
8	-0.17488467	-80.03303498	-0.14861304	-132.01737658
9	0.06931021	65.12717502	0.08752954	152.69375160
10	0.03660555	-23.88882719	-0.00188657	-98.85733685
11	-0.07974159	-12.69182330	-0.05024890	19.11711491
12	0.06104377	26.09109314	0.05194748	33.97470194
13	-0.02188541	-19.72869646	-0.02587876	-43.50808988
14	-0.00363517	7.75646626	0.00178881	26.86811588
15	0.00934564	-0.15908377	0.00771082	-8.38418076
16	-0.00551184	-1.79791381	-0.00662474	-0.70234547
17	0.00157172	1.15479772	0.00297031	2.11866446
18	-0.00007715	-0.36584077	-0.00075092	-1.05634588
19	-0.00010369	0.05562938	0.00006630	0.26079264
20	0.00003332	-0.00201497	0.00001236	-0.02740341

Tabela B.3 – Coeficientes dos filtros digitais dos casos 2I e 2A1

Casos:	Caso 2 (I)		Caso 2 (A1)	
i	b_i	a_i	b_i	a_i
0	0.01859463	1	0.00399873	1
1	-0.10613915	-7.27590841	-0.02357106	-7.99373003
2	0.29138939	25.47342666	0.06701475	30.78493885
3	-0.49149916	-55.85788451	-0.11885896	-74.97792430
4	0.51610447	82.50232233	0.13690692	126.08829428
5	-0.23471985	-78.84913329	-0.08393675	-146.56314036
6	-0.24325309	31.92968827	-0.02902985	101.34139162
7	0.60991183	38.06344820	0.13749311	4.56279789
8	-0.62136959	-88.74060629	-0.17240475	-116.90178916
9	0.31405400	90.65637060	0.11861665	169.93132673
10	0.05029132	-51.25328178	-0.02442778	-139.19537112
11	-0.23366283	4.07172009	-0.04511193	58.10999446
12	0.19867802	22.32747461	0.06026776	15.54572908
13	-0.06889804	-23.43412297	-0.03670977	-46.52150712
14	-0.02720391	12.48319392	0.00743978	39.01757978
15	0.05089807	-2.97592827	0.00761723	-18.03985906
16	-0.03329466	-0.72988002	-0.00870113	3.01760945
17	0.01257575	0.86931556	0.00458269	1.97443095
18	-0.00258294	-0.30065284	-0.00135502	-1.65061779
19	0.00016025	0.04069499	0.00016375	0.54531491
20	0	0	0.00001425	-0.07535174

Tabela B.4 – Coeficientes dos filtros digitais dos casos 2A2 e 3I

Casos:	Caso 2 (A2)		Caso 3 (I)	
I	b_i	a_i	b_i	a_i
0	0.00532188	1	0.06703988	1
1	-0.03278684	-7.58962409	-0.39798960	-7.87977304
2	0.09816363	27.85411543	1.13596794	29.85178083
3	-0.18519004	-64.98408878	-1.99942489	-71.07961269
4	0.23277973	105.18122405	2.23571716	115.66305791
5	-0.17451852	-117.95872592	-1.22424279	-127.55500539
6	0.00734053	78.10471738	-0.78006703	77.95917931
7	0.18338708	6.70198977	2.65467165	20.97651166
8	-0.28295404	-92.80079133	-3.30396489	-114.96496307
9	0.24072217	131.50791351	2.54682946	148.37584660
10	-0.10620953	-107.19802421	-1.10343765	-109.59401137
11	-0.02315333	46.33614954	-0.17039547	36.24984948
12	0.08307730	8.64916588	0.90548556	21.56373244
13	-0.07573164	-32.59321512	-1.18600274	-40.25316648
14	0.04155723	28.52075753	1.18530328	29.62841442
15	-0.01441766	-14.07381554	-0.99021516	-12.02022338
16	0.00305043	3.15924279	0.67330850	1.25550760
17	-0.00087853	0.88722199	-0.35377420	1.64229856
18	0.00081710	-1.01462958	0.13457394	-1.10338326
19	-0.00049503	0.36291132	-0.03336354	0.32513333
20	0.00012676	-0.05237190	0.00413082	-0.04090115

Tabela B.5 – Coeficientes dos filtros digitais dos casos 3A1 e 3A2

Casos:	Caso 3 (A1)		Caso 3 (A2)	
i	b_i	a_i	b_i	a_i
0	0.00726565	1	0.03921597	1
1	-0.03972212	-7.32088757	-0.20811423	-6.50769383
2	0.10446271	25.39297383	0.53183325	20.37256397
3	-0.16640844	-54.20585884	-0.83425902	-39.87678913
4	0.15807043	76.18339355	0.80032910	51.96859567
5	-0.04446800	-65.66643432	-0.26690626	-41.20782308
6	-0.12184212	14.34658697	-0.53213376	5.39371445
7	0.22994064	52.73182272	1.08883976	37.41209621
8	-0.20713899	-93.17881281	-1.04797594	-61.14905251
9	0.08416866	83.23935847	0.51010793	53.22878901
10	0.04076941	-37.20604259	0.09479006	-23.88773966
11	-0.09333732	-8.88054752	-0.40144979	-4.90089526
12	0.07301980	29.48485909	0.36152339	18.07640510
13	-0.02694973	-24.93853621	-0.16678486	-15.84386252
14	-0.00432849	11.11506485	0.01295651	7.56302215
15	0.01216763	-1.23347531	0.03823833	-1.25159438
16	-0.00803086	-1.82811027	-0.02744912	-0.99346066
17	0.00309105	1.36002758	0.00770474	0.87541433
18	-0.00072086	-0.48038898	0.00053171	-0.32633355
19	0.00006848	0.09496740	-0.00108323	0.05884493
20	0.00001643	-0.00931222	0.00028488	-0.00322954

Tabela B.6 – Coeficientes dos filtros digitais dos casos 4IH e 4IL

Casos:	Caso 4 (IH)		Caso 4 (IL)	
I	b_i	a_i	b_i	a_i
0	0.00248617	1	0.02538579	1
1	-0.01387121	-6.99437280	-0.13198531	-6.237167
2	0.03726432	23.08562782	0.32826127	18.62678274
3	-0.06089865	-46.83042359	-0.49137339	-34.37360821
4	0.06055320	62.35248176	0.41696856	40.65172935
5	-0.02274050	-50.25548063	-0.02627194	-24.84129142
6	-0.03468300	8.33614088	-0.47739763	-10.54364545
7	0.07290063	40.25199927	0.73974779	44.73913519
8	-0.06726809	-64.70579628	-0.57531364	-54.79599886
9	0.02918348	53.45112235	0.13800871	35.85932595
10	0.00842779	-21.87210860	0.23544687	-4.60828033
11	-0.02299819	-5.17944352	-0.33444796	-17.10095644
12	0.01674127	14.90430350	0.20838479	20.29023229
13	-0.00500562	-11.28696831	-0.04055763	-11.55421896
14	-0.00150593	4.53157893	-0.04391278	2.24729827
15	0.00223917	-0.59827294	0.04454934	1.87837545
16	-0.00099653	-0.36472552	-0.01807263	-1.85173935
17	0.00018532	0.21326818	0.00154282	0.73423183
18	-0.00000143	-0.03879874	0.0016852	-0.10857261
19	0	0	-0.00062588	-0.01731313
20	0	0	0.00007729	0.00633696

Tabela B.7 – Coeficientes dos filtros digitais dos casos 4ISH e 4OH

Casos:	Caso 4 (ISH)		Caso 4 (OH)	
i	b_i	a_i	b_i	a_i
0	0.09790496	1	0.00322666	1
1	-0.5364184	-6.37628315	-0.01906536	-7.03098
2	1.38405785	19.21756745	0.0535936	23.31656
3	-2.11080464	-35.21020885	-0.08982853	-47.1008
4	1.79143865	40.44658427	0.08720597	60.64302
5	-0.06444111	-22.24664121	-0.01976355	-41.9813
6	-2.15337541	-15.05233862	-0.08208413	-10.0519
7	3.25705125	48.42925016	0.14478774	65.06272
8	-2.4384187	-54.85069609	-0.11952216	-84.0776
9	0.45686094	32.1792786	0.03131112	55.79284
10	1.15110667	0.04365228	0.04770889	-6.3004
11	-1.49218418	-19.7691442	-0.06861506	-27.7777
12	0.86456642	20.24699965	0.04101732	31.486
13	-0.11832761	-10.16617519	-0.00577936	-16.5887
14	-0.22270793	1.08508407	-0.01032188	2.087731
15	0.19865708	2.29825697	0.00882777	3.470425
16	-0.07455137	-1.83465133	-0.00285832	-2.75538
17	0.00480808	0.65468741	-0.00014963	0.907541
18	0.00781866	-0.07612027	0.00043927	-0.07075
19	-0.00300763	-0.02731084	-0.00012334	-0.0399
20	0.00022191	0.00911401	0.00000644	0.008768

Tabela B.8 – Coeficientes dos filtros digitais dos casos 4OL e 5I

Casos:	Caso 4 (OL)		Caso 5 (I)	
i	b_i	a_i	b_i	a_i
0	0.00855542	1	0.60658	1
1	-0.04993335	-6.82844	-2.76786	-4.83758
2	0.14148339	22.7009	6.02595	11.32402
3	-0.24803212	-47.5868	-7.67258	-16.1033
4	0.27504062	67.1752	4.685411	13.0526
5	-0.14447944	-60.15	2.705627	-0.56198
6	-0.10611239	18.25874	-9.87097	-14.7276
7	0.32291531	39.51949	11.4601	22.44439
8	-0.36105086	-78.0244	-6.47657	-17.6696
9	0.21289116	74.92332	-0.90815	5.063498
10	-0.00553539	-38.7928	5.398057	5.905753
11	-0.12203392	-1.41813	-5.10447	-9.28965
12	0.12799522	22.43627	2.160161	6.258338
13	-0.06624764	-21.6661	0.335055	-1.71102
14	0.00868026	11.06576	-1.06844	-0.84471
15	0.01428892	-2.28559	0.65204	1.126848
16	-0.01253763	-1.06321	-0.12919	-0.51884
17	0.00513052	1.080401	-0.06308	0.076555
18	-0.00098526	-0.41755	0.041977	0.027431
19	-0.00003055	0.079312	-0.00595	-0.01244
20	0.0000383	-0.00543	-0.00041	0.001736

Tabela B.9 – Coeficientes dos filtros digitais dos casos 5O e 5OCD

Casos:	Caso 5 (O)		Caso 5 (OCD)	
i	b_i	a_i	b_i	a_i
0	0.031867	1	0.072846	1
1	-0.16539	-6.15602	-0.40029	-6.50491
2	0.41339	17.81111	1.06259	20.15268
3	-0.62452	-31.4694	-1.70297	-38.3466
4	0.546058	35.34113	1.596263	46.76077
5	-0.07317	-19.6664	-0.34421	-30.3169
6	-0.56148	-11.5315	-1.4818	-9.50322
7	0.923211	39.68291	2.622299	49.74679
8	-0.75865	-46.3213	-2.24126	-63.3001
9	0.229037	29.21282	0.721616	42.93014
10	0.257298	-3.10256	0.734247	-6.90535
11	-0.41669	-14.378	-1.2331	-19.1153
12	0.280256	16.59819	0.83558	23.69376
13	-0.06809	-9.35034	-0.20126	-13.86
14	-0.04995	1.812215	-0.15232	2.871328
15	0.057707	1.520195	0.174072	2.176486
16	-0.0239	-1.487	-0.07183	-2.22189
17	0.001146	0.564428	0.004214	0.884668
18	0.003124	-0.06426	0.008686	-0.12515
19	-0.00106	-0.02153	-0.00337	-0.02606
20	4.21E-05	0.005864	0.000299	0.009253

Tabela B.10 – Coeficientes dos filtros digitais dos casos 5OER e 6IER

Casos:	Caso 5 (OER)		Caso 6 (IER)	
i	b_i	a_i	b_i	a_i
0	0.31856	1	0.252003	1
1	-1.42349	-4.98945	-1.21103	-5.38597
2	3.046179	12.01285	2.76341	13.855
3	-3.70593	-17.2572	-3.6726	-21.5213
4	1.844192	13.50572	2.391609	19.62032
5	2.202206	1.250612	1.147628	-3.9359
6	-5.69618	-18.8968	-4.77292	-18.177
7	5.789687	26.6303	5.728304	31.75399
8	-2.37335	-18.7251	-3.29702	-27.2618
9	-1.75544	2.03411	-0.48959	9.350902
10	3.613132	10.7857	2.844043	8.037685
11	-2.63094	-12.7159	-2.68717	-14.4585
12	0.543027	6.678543	1.092745	10.26288
13	0.757679	-0.10838	0.260194	-2.86554
14	-0.80876	-2.54719	-0.6323	-1.56134
15	0.304	1.885709	0.364166	2.07932
16	0.048513	-0.49637	-0.05415	-0.94568
17	-0.09708	-0.14927	-0.05165	0.095058
18	0.03146	0.148551	0.030394	0.097952
19	0.002919	-0.03213	-0.00267	-0.03712
20	-0.00293	-0.00234	-0.0014	0.001177

Tabela B.11 – Coeficientes dos filtros digitais dos casos 6ITV e 6OCD

Casos:	Caso 6 (ITV)		Caso 6 (OCD)	
i	b_i	a_i	b_i	a_i
0	0.136785	1	0.07489	1
1	-0.69189	-6.68034	-0.41456	-6.43941
2	1.730116	21.86981	1.088556	19.51778
3	-2.69567	-45.4055	-1.68261	-35.684
4	2.599856	63.97441	1.438033	40.4159
5	-0.92883	-58.0606	-0.0572	-21.1127
6	-1.61258	19.75206	-1.70673	-16.6251
7	3.461633	34.64631	2.560209	48.77512
8	-3.46398	-73.4724	-1.88605	-53.1321
9	1.833965	74.6364	0.328533	29.57406
10	0.136275	-42.9458	0.893656	1.434592
11	-1.25162	3.487107	-1.12237	-19.0895
12	1.248296	20.50233	0.633453	18.54578
13	-0.6545	-23.1913	-0.08144	-8.95426
14	0.107672	13.78678	-0.16103	0.851783
15	0.12279	-3.99943	0.139054	2.008114
16	-0.11719	-0.78104	-0.04877	-1.54201
17	0.049138	1.442333	0.000422	0.527296
18	-0.00892	-0.72581	0.006162	-0.05928
19	-0.00155	0.187373	-0.00204	-0.01599
20	0.001385	-0.02042	0.000199	0.004541

Tabela B.12 – Coeficientes dos filtros digitais do caso 6OER

Casos:	Caso 6 (OER)		-	
i	b_i	a_i	b_i	a_i
0	0.274483	1	-	-
1	-1.23105	-4.99868	-	-
2	2.577853	11.83709	-	-
3	-2.91316	-16.1557	-	-
4	0.892243	10.45572	-	-
5	2.838176	6.105716	-	-
6	-5.47426	-23.1935	-	-
7	4.585682	27.065	-	-
8	-0.76453	-14.1525	-	-
9	-2.83617	-4.89364	-	-
10	3.636927	15.69845	-	-
11	-1.88567	-13.3319	-	-
12	-0.23346	4.286302	-	-
13	1.080122	2.4563	-	-
14	-0.72818	-3.65375	-	-
15	0.130895	1.795489	-	-
16	0.124713	-0.14418	-	-
17	-0.09269	-0.30235	-	-
18	0.016711	0.154312	-	-
19	0.005821	-0.02015	-	-
20	-0.00185	-0.00284	-	-

Referências Bibliográficas

- [1] – HOUSE, P. *Cebus for The Masses*. Intellon Home Page.
http://www.intellon.com/docs/articles/ph_mass1.pdf, 2000.
- [2] – RADFORD, D. *New Spread Spectrum Technologies Enable Low Cost Applicatoins For Residential And Commercial Use*. Intellon Home Page.
<http://www.intellon.com/docs/articles/ieee.pdf>, 2000.
- [3] – INTELLON. *Phase Coupling for Power Line Communications*. Intellon Home Page.
<http://www.intellon.com/docs/appnotes/26001087.pdf>, 2000.
- [4] – INTELLON. *CEBus Power Line Encoding and Signaling*. Intellon Home Page.
<http://www.intellon.com/docs/appnotes/26001087.pdf>, 2000.
- [5] – INTELLON. *Your First CEBus Packet*. Intellon Home Page.
<http://www.intellon.com/docs/appnotes/26000686.pdf>, 2000.
- [6] – INTELLON. *SSC P200 AND P300 Hardware Design Reference*. Intellon Home Page.
<http://www.intellon.com/docs/refnotes/26000679.pdf>, 2000.
- [7] – INTELLON. *SSC P111 PL Media Interface IC*. Intellon Home Page.
<http://www.intellon.com/docs/datasht/24000624.pdf>, 2000.
- [8] – INTELLON. *SSC P300 PL Network Interface Controller*. Intellon Home Page.
<http://www.intellon.com/docs/datasht/24000739.pdf>, 2000.
- [9] – SKLAR, B. *Digital Communications*. Fundamentals and Applications. New Jersey, Prentice-Hall, 1988.
- [10] – GLISIC, S., VUCEVIT, B. *Spread Spectrum CDMA Systems for Wireless Communications*. Norwood, Artech House, 1997.
- [11] – KOSOW, I. *Máquinas Elétricas e Transformadores*. 3a. ed. Porto Alegre, Globo, 1979.
- [12] – SHWEHDI, M. H.; KHAN, A. Z. *A power line data communication interface using spread spectrum technology in home automation*. IEEE Transactions on Power Delivery, Vol. 11, Nº. 3, pp. 1232-1237, Julho de 1996.
- [13] – GORIZ, J. *Comunicação por Linha de Alta Tensão*. AN-EP, pp.337-344, Setembro/Outubro, 1986.
- [14] – CHAFFANJON, D., DUVAL, G., MEUNIER, M., PACAUD, A. *Differential and Commom Mode Propagation in PLC Low Voltage Networks*. IEEE Transactions on Power Delivery, Vol. 14, Nº. 2, pp. 327-334, Abril de 1999.
- [15] – CHAN, M., DONALDSON, R. *Attenuation of Communication Signals on Residential and Commercial Intrabuilding Power-Distribution Circuits*. IEEE Transactions on Electromagnetic Compatibility, Vol. 28, Nº. 4, pp. 220-230, Novembro de 1986.
- [16] – TRETER, S. *Communication System Design Using DSP Algorithms*. New York, Plenium Press, 1995.

- [17] – JR., VIDAL P. *Aplicações Práticas do Microcontrolador 8051*. 4 ed. São Paulo, Érica, 1994.
- [18] – FLIKKEMA, PAUL G. *Spread Spectrum Techniques for Wireless Communication*. IEEE Signal Processing Magazine, Vol. 14, N°. 3, pp. 26-36, Maio de 1997.
- [19] – JARDINI, JOSÉ A. *Sistemas Digitais para Automação da Geração, Transmissão e Distribuição de Energia Elétrica*. São Paulo, FCA, 1996.
- [20] – OPPENHEIM, A. V; SCHAFER, R. W. *Discrete-Time Signal Processing*. New Jersey, Prentice-Hall, 1989.
- [21] – ANH VU. *Error Detection And Error Correction*. ANH VU Home Page. <http://home.apu.edu/~jcox/projects/individual/Vu/>, 2001.
- [22] – VG-AnyLAN. *Cyclical Redundancy Check*. VG-AnyLAN Home Page. <http://www.ktinet.hu/100vg/teach/vgconcepts/frame/crcsum.html>, 2001.
- [23] – FOLDOC. *Free On-Line Dictionary of Computing*. FOLDOC Home Page. <http://foldoc.doc.ic.ac.uk/foldoc/foldoc.cgi?query=CSMA> , 2001.
- [24] – FAIRHURST, G. *CSMA-CD, A Shared Medium*. Godred Fairhurst Home Page. <http://www.erg.abdn.ac.uk/users/gorry/course/lan-pages/csma-cd.html>, 2001.
- [25] – MITRA, S. *Digital Signal Processing. A Computer-Based Approach*. New York, McGraw-Hill, 1998.
- [26] – QIAN, S., CHEN, D. *Joint Time-Frequency Analysis*. IEEE Signal Processing Magazine, Vol. 16, No. 2, pp. 53-67, Março de 1999.
- [27] – INTELLON. *Using Intellon's Serial Peripheral Interface*. Intellon Home Page. <http://www.intellon.com/docs/appnotes/26000722.pdf>, 2000.
- [28] – INTELLON. *SSC P200 Software Design Reference*. Intellon Home Page. <http://www.intellon.com/docs/appnotes/26000806.pdf>, 2000.
- [29] – VINES, R.M., TRUSSEL, K.C. *Impedance for the residential power distribution circuits*. IEEE Transactions on Electromagnetic Compatibility, Vol. 27, pp. 06-12, Fevereiro de 1985.
- [30] – HAYKIN, S. *Adaptive Filter Theory*. 2 ed. New Jersey, Prentice-Hall, 1991.
- [31] – FRIEDLANDER, B., PORAT, B. *The Modified Yule-Walker Method of ARMA Spectral Estimation*. IEEE Transactions on Aerospace Electronic Systems, Vol. 20, N°. 2, pp. 158-173, Março de 1984.
- [32] – PARKS, T. W., BURRUS, C. S. *Digital Filter Design*. New York, John Wiley & Sons, Inc., 1987.
- [33] – X10. *X10 Powerline Carrier (PLC) Technology*. X10 Home Page. <http://www.x10.com/support/technology1.htm>, 2001.
- [34] – ECHELON. *PLT Power Line Technology*. Echelon Home Page. <http://www.echelon.com>, 2001.
- [35] – HOME AUTOMATION. *X-10 vs. CEBus vs. LONworks*. Technocopia Home Page. <http://www.technocopia.com/ha-19991109-protocols.html>, 2001.

- [36] – RADFORD, D. *Spread-Spectrum Data Leap Through AC Power Wiring*. IEEE Spectrum, Vol. 32, Nº. 11, pp. 48-53, Novembro de 1996.
- [37] – REITER, L. F. *Implementação de Comunicação Serial no modem de linha DLC-P300*. Juiz de Fora: 2000. 55p. Trabalho Final de Curso (Faculdade de Engenharia) – UFJF.
- [38] – INTELLON. *Spread Spectrum Communications for AC and DC Power Line Networks*. Intellon Home Page. <http://www.intellon.com/docs/appnotes/26001087.pdf>, 2000.
- [39] – 3R. *Data Transmission/Networks. PLC Technical trend*. 3R Home Page. http://www.3r.co.kr/eng/bodyc01c_a.htm, 2001.
- [40] – IGNÁCIO, H. K. *TMS320C24x DSP 1-Day Workshop*. Belo Horizonte: 2000. 158p. Workshop (Texas Instrument's).
- [41] – SUTTERLIN, P., DOWNEY, W. *A Power Line Communication Tutorial—Challenges and Technologies*. Echelon Home Page. <http://www.echelon.com>, 2000.
- [42] – SUTTERLIN, P., DOWNEY, W. *Power Line Communication Technology Update*. Echelon Home Page. <http://www.echelon.com>, 2001.
- [43] – ECHELON. *PLT-22 Power Line Transceiver Model 50090-03*. Echelon Home Page. <http://www.echelon.com>, 2000.
- [44] – THORNTON INPEC ELETRÔNICA LTDA. *Catálogo de Ferrites*, 1994