

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FUNÇÕES DE MULTIPLEXAÇÃO E1

por

GUSTAVO WEYMAR KAISER

Dissertação submetida como requisito parcial  
para a obtenção do grau de  
Mestre em Ciência da Computação

Prof. Tiaraju Vasconcellos Wagner  
Orientador

Porto Alegre, dezembro de 1995.

UFRGS  
INSTITUTO DE INFORMÁTICA  
BIBLIOTECA

## CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Kaiser, Gustavo Weymar

Funções de Multiplexação E1 / Gustavo Weymar Kaiser. — Porto Alegre : CPGCC da UFRGS, 1995.

202 p. : il.

Dissertação (mestrado) — Universidade Federal do Rio Grande do Sul. Curso de Pós-Graduação em Ciência da Computação, Porto Alegre, BR-RS, 1995. Orientador: Wagner, Tiaraju Vasconcellos

1. Microeletrônica. 2. Multiplexadores. 3. Comunicação Digital de Dados. 4. Redes Corporativas. 5. Projeto de Sistemas Digitais. I. Wagner, Tiaraju Vasconcellos. II. Título.

| UFRGS<br>INSTITUTO DE INFORMÁTICA<br>BIBLIOTECA |                  |                   |
|---|------------------|-------------------|
| N.º CHAMADA<br>621.38-181.4(043)<br>K13F        | N.º REG:<br>2055 | DATA:<br>24.01.97 |
| ORIGEM: D                                       | DATA: 11.12.96   | PREÇO: R\$ 30.00  |
| FUNDO: II                                       | FORN.: II        |                   |

microeletronica -  
SEU/II

Multiplexadores:  
Comunicações:

Dados

Redes corporativas

Sistemas digitais

CNPq 3.04.03.00-6

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Dr. Héglio Casses Trindade

Pró-Reitor de Pesquisa e Pós-Graduação: Prof. Dr. Cláudio Scherer

Diretor do Instituto de Informática: Prof. Dr. Roberto Tom Price

Coordenador do CPGCC: Prof. Dr. José Palazzo Moreira de Oliveira

Bibliotecária-Chefe do Instituto de Informática: Zita Prates de Oliveira



## AGRADECIMENTOS

Ao Prof. Msc. **Tiaraju Vasconcellos Wagner** que, como orientador e amigo, mostrou-se sempre prestativo e solícito. Sua experiência, competência e dedicação foram fundamentais para que a realização deste trabalho fosse possível.

Agradeço também penhoradamente ao Prof. Msc. **Juergen Rochol**, pela valiosa orientação complementar, amizade e apoio recebidos durante o andamento deste trabalho.

Aos Profs. **Dr. Ricardo A. L. Reis** e **Dr. Sérgio Bampi** pelos conhecimentos transmitidos nas disciplinas cursadas e atividades extra-classe.

A todos os colegas do Grupo de Microeletrônica (GME) pelo companheirismo, espírito de camaradagem e amizade, que tornam mais agradáveis as horas de trabalho. Sendo impossível citar a todos nominalmente, agradeço de coração a todos em geral e a cada um, em particular.

À minha esposa, **Glória**, pelo incentivo, amor e compreensão pelos momentos de convívio subtraídos nas horas de trabalho intenso.

À minha mãe, **Léia**, a quem devo minha vida e meu caráter.

A todo o corpo de funcionários do Instituto de Informática e, em especial, o atendimento carinhoso do pessoal da secretaria do CPGCC e da biblioteca. Agradeço também de forma especial às "gurias" da portaria, Eliane e Márcia.

Aos funcionários do Restaurante Universitário, pela "bóia".

Por fim, agradeço aos trabalhadores brasileiros, de todos os níveis, que efetivamente financiaram as bolsas de estudo por mim recebidas. A eles dedico este trabalho.

# SUMÁRIO

|  |           |
|--|-----------|
| <b>LISTA DE ABREVIATURAS</b> . . . . .   | <b>8</b>  |
| <b>LISTA DE FIGURAS</b> . . . . .  | <b>9</b>  |
| <b>RESUMO</b> . . . . .  | <b>16</b> |
| <b>ABSTRACT</b> . . . . .  | <b>17</b> |
| <b>1 INTRODUÇÃO</b> . . . . .  | <b>18</b> |
| <b>1.1 Digitalização em Redes de Telefonia Pública</b> . . . . .                     | <b>19</b> |
| <b>1.2 Evolução da Comunicação de Dados</b> . . . . .                                | <b>21</b> |
| <b>2 APLICAÇÕES DO MULTIPLEXADOR <i>E1</i></b> . . . . .                             | <b>26</b> |
| <b>2.1 Rede <i>E1</i> Embratel</b> . . . . .   | <b>26</b> |
| 2.1.1 Tipos de Acesso à Rede <i>E1</i> Embratel . . . . .                            | 29        |
| 2.1.2 Métodos de Sincronismo na Rede <i>E1</i> Embratel . . . . .                    | 30        |
| <b>2.2 Redes Corporativas</b> . . . . .  | <b>30</b> |
| <b>2.3 Acesso da Rede Corporativa à Rede Pública de Comunicações</b>                 | <b>33</b> |
| <b>3 PADRONIZAÇÃO DA HIERARQUIA DIGITAL <i>ITU-T</i></b> . . . . .                   | <b>35</b> |
| <b>3.1 Rec. <i>G. 702</i> - Hierarquias Digitais e Taxas de Bit Associadas</b> .     | <b>36</b> |
| <b>3.2 Rec. <i>G. 703</i> - Características Físicas e Elétricas das Interfaces</b> . | <b>38</b> |
| 3.2.1 Interface de 64 kbit/s . . . . .   | 38        |
| 3.2.1.1 Temporização de Transmissão . . . . .  | 39        |
| 3.2.1.2 Características Elétricas da Interface Co-direcional . . . . .               | 41        |
| 3.2.2 Interface de 2048 kbit/s . . . . .   | 44        |
| <b>3.3 Rec. <i>G. 704</i> - Estruturas de Quadros Síncronos dos Níveis 1 e 2</b>     | <b>47</b> |

|            |  |           |
|------------|--|-----------|
| 3.3.1      | Estrutura Básica de Quadro <i>E1</i> . . . . .   | 47        |
| 3.3.2      | Estrutura de Multiquadro <i>CRC-4</i> . . . . .  | 49        |
| 3.3.3      | Implementação da Verificação Cíclica de Redundância <i>CRC-4</i> . . . . .   | 52        |
| <b>3.4</b> | <b>Rec. G.706 - Procedimentos de Controle em Estruturas Básicas de Quadro</b> . . . . .  | <b>53</b> |
| 3.4.1      | Perda de Sincronismo de Quadro . . . . .   | 54        |
| 3.4.2      | Estratégia para Alinhamento de Quadro . . . . .  | 56        |
| 3.4.3      | Alinhamento de Multiquadro . . . . .   | 57        |
| <b>3.5</b> | <b>Rec. G.736 - Características de um MUX de 2048 kbit/s</b> . . . . .   | <b>59</b> |
| 3.5.1      | Taxa de Bit . . . . .  | 59        |
| 3.5.2      | Sinal de Relógio . . . . .   | 59        |
| 3.5.3      | Condições de Falha e Ações . . . . .   | 60        |
| <b>3.6</b> | <b>Rec. G.823 - O Controle do <i>Jitter</i> e <i>Wander</i> nas Redes Digitais Baseadas na Hierarquia de 2048 kbit/s</b> . . . . . | <b>62</b> |
| 3.6.1      | <i>Jitter</i> e <i>Wander</i> . . . . .  | 62        |
| 3.6.2      | <i>Jitter</i> e <i>Wander</i> nas Interfaces de 2048 kbit/s e 64 kbit/s do Multiplexador <i>E1</i> . . . . .                       | 65        |
| <b>4</b>   | <b>ASPECTOS FUNCIONAIS DO MULTIPLEXADOR <i>E1</i></b> . . . . .  | <b>68</b> |
| 4.1        | Implementação de um Enlace de Multiplexação <i>E1</i> . . . . .  | 68        |
| 4.2        | Pré-requisitos para a Multiplexação . . . . .  | 71        |
| 4.3        | Terminologia para Sinais em Redes Digitais de Comunicação . . . . .  | 74        |
| 4.4        | Compatibilização Entre MUX e Tributários . . . . .   | 77        |
| 4.5        | Armazenamento Elástico . . . . .   | 78        |
| 4.6        | Objetivos de <i>Slips</i> . . . . .  | 86        |
| <b>5</b>   | <b>MÓDULOS FUNCIONAIS PARA MULTIPLEXAÇÃO <i>E1</i></b> . . . . .   | <b>88</b> |

|                |   |            |
|----------------|---|------------|
| <b>5.1</b>     | <b>Módulo Gerador de Base de Tempo . . . . .</b>                                      | <b>88</b>  |
| <b>5.2</b>     | <b>Módulos para Multiplexação . . . . .</b>   | <b>91</b>  |
| 5.2.1          | Módulos de Recepção para Interface <i>G.703</i> de 64 kbit/s . . . . .                | 93         |
| 5.2.1.1        | Circuito de Recuperação de Relógio de 64 kHz . . . . .                                | 94         |
| 5.2.1.2        | Decodificação <i>G.703</i> . . . . .  | 98         |
| 5.2.1.3        | Detecção de Violações . . . . .   | 99         |
| 5.2.1.4        | Armazenamento Elástico para Tributários . . . . .                                     | 100        |
| 5.2.2          | Codificação <i>HDB-3</i> . . . . .  | 117        |
| <b>5.3</b>     | <b>Módulos para Demultiplexação . . . . .</b>   | <b>119</b> |
| 5.3.1          | Recepção do Sinal Codificado de 2048 kbit/s . . . . .                                 | 122        |
| 5.3.1.1        | Recuperação de Relógio de 2048 kHz . . . . .  | 122        |
| 5.3.1.2        | Decodificação <i>HDB-3</i> . . . . .  | 123        |
| 5.3.1.3        | <i>Buffer Elástico</i> de Agregado . . . . .  | 125        |
| 5.3.2          | Alinhamento de Quadro . . . . .   | 139        |
| 5.3.3          | Geração de <i>Time-Slots</i> . . . . .  | 144        |
| 5.3.4          | Módulos de Transmissão de Tributários . . . . .                                       | 147        |
| <b>6</b>       | <b>CONCLUSÕES . . . . .</b>   | <b>153</b> |
| <b>ANEXO 1</b> | <b>SUBCIRCUITOS PARA OS MÓDULOS DE RECEPÇÃO DE TRIBUTÁRIOS . . . . .</b>              | <b>157</b> |
| <b>ANEXO 2</b> | <b>SUBCIRCUITOS PARA OS MÓDULOS CODIFICADOR E DECODIFICADOR <i>HDB-3</i>. . . . .</b> | <b>165</b> |
| <b>ANEXO 3</b> | <b>SUBCIRCUITOS PARA O <i>BUFFER ELÁSTICO</i> DE AGREGADO . . . . .</b>               | <b>171</b> |

|              |  |     |
|--------------|--|-----|
| ANEXO 4      | SUBCIRCUITOS PARA O MÓDULO DE ALINHAMENTO DE QUADRO . . . . .        | 182 |
| ANEXO 5      | SUBCIRCUITOS PARA O MÓDULO GERADOR DE <i>TIME-SLOTS</i> . . . . .    | 189 |
| ANEXO 6      | SUBCIRCUITOS PARA OS MÓDULOS DE TRANSMISSÃO DE TRIBUTÁRIOS . . . . . | 196 |
| BIBLIOGRAFIA | . . . . .  | 199 |

## LISTA DE ABREVIATURAS

|       |  |
|-------|--|
| ABNT  | – Associação Brasileira de Normas Técnicas                             |
| A/D   | – <i>Analog to Digital</i>   |
| AMI   | – <i>Alternate Mark Inversion Code</i>                                 |
| CCITT | – Comitê Consultativo Internacional de Telegrafia e Telefonia          |
| CMOS  | – <i>Complementar Metal Oxid Semiconductor</i>                         |
| CRC-4 | – Verificação de redundância cíclica a 4 bits                          |
| D/A   | – <i>Digital to Analog</i>   |
| DC    | – <i>Direct Current</i>  |
| DEMUX | – Demultiplexador  |
| FDM   | – Multiplexação por Divisão em Frequência                              |
| HDB-3 | – <i>High Density Bipolar Code With Three Zero Excess Substitution</i> |
| ITU-T | – <i>International Telecommunications Union</i>                        |
| LPF   | – <i>Low-Pass Filter</i>   |
| MUX   | – Multiplexador  |
| NRZ   | – <i>Non Return to Zero</i>  |
| PABX  | – <i>Private Automatic Branch Exchange</i>                             |
| PC    | – <i>Phase Comparator</i>  |
| PCM   | – <i>Pulse Code Modulation</i>   |
| PLL   | – <i>Phase Locked Loop</i>   |
| PPM   | – <i>Part Per Million</i>  |
| TDM   | – Multiplexação por Divisão no Tempo                                   |
| UI    | – <i>Unit Interval</i>   |
| VCO   | – <i>Voltage Controlled Oscillator</i>                                 |

## LISTA DE FIGURAS

|   |    |
|---|----|
| Figura 2.1 - Topologia básica da Rede <i>E1</i> Embratel. . . . .                               | 27 |
| Figura 2.2 - Rede <i>E1</i> como <i>backbone</i> . . . . .                                      | 31 |
| Figura 2.3 - Rede <i>E1</i> com controle centralizado. . . . .                                  | 33 |
| Figura 3.1 - Hierarquia Digital <i>ITU-T T1</i> . . . . .                                       | 36 |
| Figura 3.2 - Hierarquia Digital <i>ITU-T E1</i> . . . . .                                       | 37 |
| Figura 3.3 - Interface co-direcional. . . . .   | 39 |
| Figura 3.4 - Interface com relógio centralizado. . . . .  | 40 |
| Figura 3.5 - Interface contra-direcional. . . . .   | 41 |
| Figura 3.6 - Conversão de código para interface co-direcional. . . . .                          | 42 |
| Figura 3.7 - Máscara para pulso único. . . . .  | 43 |
| Figura 3.8 - Máscara para pulso duplo. . . . .  | 44 |
| Figura 3.9 - Códigos <i>HDB-3</i> e <i>AMI</i> na interface <i>G.703</i> . . . . .              | 45 |
| Figura 3.10 - Máscara para pulsos na interface de 2048 kbit/s. . . . .                          | 46 |
| Figura 3.11 - Estrutura básica de quadro <i>E1</i> . . . . .                                    | 49 |
| Figura 3.12 - Estrutura de multiquadro. . . . .   | 50 |
| Figura 3.13 - Circuito para obtenção dos coeficientes de <i>CRC-4</i> . . . . .                 | 53 |
| Figura 3.14 - Procedimento para declaração do estado de perda de sincronismo de quadro. . . . . | 56 |
| Figura 3.15 - Procedimento para obtenção de sincronismo de quadro. . . . .                      | 58 |
| Figura 3.16 - Condições de falha e ações. . . . .   | 60 |
| Figura 3.17 - Efeito do <i>jitter</i> em um sinal digital. . . . .                              | 65 |
| Figura 3.18 - Níveis de <i>jitter</i> nas interfaces. . . . .                                   | 66 |
| Figura 4.1 - Enlace de multiplexação <i>E1</i> . . . . .  | 69 |

|             |  |     |
|-------------|--|-----|
| Figura 4.2  | - Geração de <i>time-slots</i> do quadro básico <i>E1</i> . . . . .    | 70  |
| Figura 4.3  | - Registrador de deslocamento. . . . .                                 | 72  |
| Figura 4.4  | - Paralelização de octetos. . . . .                                    | 73  |
| Figura 4.5  | - Defasamento necessário à viabilidade da multiplexação. . . . .       | 74  |
| Figura 4.6  | - Classificação dos sinais em uma rede de comunicação digital. . . . . | 75  |
| Figura 4.7  | - Modelo dos comutadores para armazenamento elástico. . . . .          | 80  |
| Figura 4.8  | - Escrita e leitura síncronas. . . . .                                 | 81  |
| Figura 4.9  | - Escrita e leitura mesócronas. . . . .                                | 82  |
| Figura 4.10 | - Escrita e leitura plesiócronas. . . . .                              | 83  |
| Figura 4.11 | - <i>Buffer elástico</i> e gráfico de ocupância. . . . .               | 84  |
| Figura 5.1  | - Módulo gerador de base de tempo. . . . .                             | 89  |
| Figura 5.2  | - Sinais de saída do módulo gerador de base de tempo. . . . .          | 90  |
| Figura 5.3  | - Diagrama de blocos funcionais para multiplexação. . . . .            | 92  |
| Figura 5.4  | - Diagrama de blocos de um módulo de recepção <i>G.703</i> . . . . .   | 93  |
| Figura 5.5  | - Formas de onda para o módulo de recepção <i>G.703</i> . . . . .      | 94  |
| Figura 5.6  | - Circuito de recuperação de relógio de 64 kHz. . . . .                | 95  |
| Figura 5.7  | - Simulação do circuito de recuperação de relógio. . . . .             | 97  |
| Figura 5.8  | - Circuito decodificador <i>G.703</i> . . . . .                        | 98  |
| Figura 5.9  | - Simulação do circuito decodificador <i>G.703</i> . . . . .           | 99  |
| Figura 5.10 | - Circuito detector de violações. . . . .                              | 100 |
| Figura 5.11 | - Simulação do circuito detector de violações. . . . .                 | 101 |
| Figura 5.12 | - Sinais para operação do <i>buffer elástico</i> . . . . .             | 102 |
| Figura 5.13 | - Diagrama de blocos do <i>buffer elástico</i> de tributários. . . . . | 103 |
| Figura 5.14 | - Intervalo entre escrita e leitura próximo de zero. . . . .           | 106 |

|  |     |
|--|-----|
| Figura 5.15 - Intervalo entre escrita e leitura próximo de 125 $\mu$ s. . . . .                                      | 107 |
| Figura 5.16 - <i>Slip</i> por <i>underflow</i> . . . . .   | 108 |
| Figura 5.17 - <i>Slip</i> por <i>overflow</i> . . . . .  | 109 |
| Figura 5.18 - Circuito para simulação conjunta dos módulos de recepção para interface <i>G.703</i> . . . . .         | 110 |
| Figura 5.19 - Sinais de dados, relógio e temporização de octeto, obtidos por simulação do circuito completo. . . . . | 111 |
| Figura 5.20 - Oito ciclos de 2048 kHz durante a ativação do <i>time-slot</i> . . . . .                               | 112 |
| Figura 5.21 - Simulação para escrita e leitura mesócronas. . . . .   | 113 |
| Figura 5.22 - Simulação para escrita e leitura mesócronas; vista em detalhe. . . . .                                 | 114 |
| Figura 5.23 - Simulação para escrita e leitura plesiócronas; <i>slip</i> por <i>overflow</i> . . . . .               | 115 |
| Figura 5.24 - Simulação para escrita e leitura plesiócronas; <i>slip</i> por <i>underflow</i> . . . . .              | 116 |
| Figura 5.25 - Saída do <i>buffer elástico</i> a partir do sinal codificado. . . . .                                  | 117 |
| Figura 5.26 - Saída do <i>buffer elástico</i> a partir do sinal codificado; vista em detalhe. . . . .                | 118 |
| Figura 5.27 - Módulo de codificação <i>HDB-3</i> . . . . .   | 119 |
| Figura 5.28 - Simulação do codificador <i>HDB-3</i> . . . . .  | 120 |
| Figura 5.29 - Diagrama de blocos funcionais para demultiplexação. . . . .  | 121 |
| Figura 5.30 - Diagrama funcional e formas de onda para o circuito de recuperação de relógio de 2048 kHz. . . . .     | 123 |
| Figura 5.31 - Módulo de decodificação <i>HDB-3</i> . . . . .   | 123 |
| Figura 5.32 - Simulação do decodificador <i>HDB-3</i> . . . . .  | 124 |
| Figura 5.33 - <i>Buffer elástico</i> para o agregado. . . . .  | 126 |
| Figura 5.34 - Sinais de relógio, entrada e saída para o <i>buffer elástico</i> . . . . .                             | 127 |
| Figura 5.35 - Funcionamento com temporização mesócrona . . . . .   | 131 |
| Figura 5.36 - Proximidade dos endereços de escrita e leitura. . . . .  | 133 |

|  |     |
|--|-----|
| Figura 5.37 - <i>Slip</i> por <i>overflow</i> . . . . .  | 134 |
| Figura 5.38 - Alteração do endereço de leitura ( <i>slip</i> por <i>overflow</i> ). . . . .        | 135 |
| Figura 5.39 - Proximidade dos endereços de escrita e leitura. . . . .                              | 136 |
| Figura 5.40 - <i>Slip</i> por <i>underflow</i> . . . . .   | 137 |
| Figura 5.41 - Alteração do endereço de leitura ( <i>slip</i> por <i>underflow</i> ). . . . .       | 138 |
| Figura 5.42 - Circuito para obtenção do alinhamento de quadro. . . . .                             | 140 |
| Figura 5.43 - Simulação do circuito de alinhamento de quadro. . . . .                              | 142 |
| Figura 5.44 - Simulação de falha para a segunda condição. . . . .                                  | 143 |
| Figura 5.45 - Simulação de falha para a terceira condição. . . . .                                 | 144 |
| Figura 5.46 - Módulo gerador de <i>time-slots</i> . . . . .  | 145 |
| Figura 5.47 - Simulação do circuito gerador de <i>time-slots</i> . . . . .                         | 146 |
| Figura 5.48 - Módulo de transmissão para canais tributários. . . . .                               | 147 |
| Figura 5.49 - Diagrama de temporização para os módulos de transmissão de tributários. . . . .      | 149 |
| Figura 5.50 - Carga de octetos a 2048 kbit/s nos módulos <i>8.bit.shift.reg</i> . . . . .          | 150 |
| Figura 5.51 - Saídas <i>G.703</i> codificadas a 64 kbit/s para os canais 1, 2, 3 e 4. . . . .      | 151 |
| Figura 1.1 - Módulo <i>sr.ff</i> ; (flip-flop set-reset). . . . .                                  | 158 |
| Figura 1.2 - Módulo <i>8.bit.sync.count</i> ; (contador síncrono de oito bits). . . . .            | 158 |
| Figura 1.3 - Módulo <i>mono z</i> ; (mono-estável). . . . .  | 159 |
| Figura 1.4 - Módulo <i>del</i> ; (retardo de propagação). . . . .                                  | 159 |
| Figura 1.5 - Módulo <i>elastic.str</i> ; ( <i>buffer elástico</i> de tributários). . . . .         | 159 |
| Figura 1.6 - Módulo <i>u.d.logic</i> ; (lógica de controle do módulo <i>u.d.counter</i> ). . . . . | 159 |
| Figura 1.7 - Módulo <i>3.bit.jcount</i> ; (contador <i>johnson</i> de 3 bits). . . . .             | 160 |
| Figura 1.8 - Módulo <i>u.d.counter</i> ; (contador <i>up/down</i> ). . . . .                       | 161 |

|             |  |     |
|-------------|--|-----|
| Figura 1.9  | - Módulo <i>stack</i> ; (registrador de deslocamento, registradores <i>R1</i> , <i>R2</i> e <i>R3</i> , tristates <i>T1</i> , <i>T2</i> e <i>T3</i> , retentor de barramento e <i>latch</i> de saída). . . . . | 162 |
| Figura 1.10 | - Módulo <i>8.bit.shift.reg</i> ; (registrador de deslocamento de oito bits). . . . .  | 162 |
| Figura 1.11 | - Módulo <i>8.bit.register</i> ; (registrador de carga paralela de oito bits). . . . .   | 163 |
| Figura 1.12 | - Módulo <i>three.state</i> ; (tristate). . . . .  | 163 |
| Figura 1.13 | - Módulo <i>bus.ret</i> ; (retentor de barramento). . . . .  | 163 |
| Figura 1.14 | - Módulo <i>8.bit.latch</i> ; ( <i>latch</i> de oito bits). . . . .  | 164 |
| Figura 2.1  | - Módulo <i>hdb cod</i> ; (codificador <i>HDB-3</i> ; esquema parcial 1 de 3). . . . .   | 166 |
| Figura 2.2  | - Módulo <i>hdb cod</i> ; (codificador <i>HDB-3</i> ; esquema parcial 2 de 3). . . . .   | 166 |
| Figura 2.3  | - Módulo <i>hdb cod</i> ; (codificador <i>HDB-3</i> ; esquema parcial 3 de 3). . . . .   | 167 |
| Figura 2.4  | - Módulo <i>hdb dec</i> ; (decodificador <i>HDB-3</i> ; esquema parcial 1 de 2). . . . .   | 167 |
| Figura 2.5  | - Módulo <i>hdb dec</i> ; (decodificador <i>HDB-3</i> ; esquema parcial 2 de 2). . . . .   | 168 |
| Figura 2.6  | - Módulo <i>jk</i> ; (flip-flop <i>jk</i> ). . . . .   | 168 |
| Figura 2.7  | - Módulo <i>ms res</i> ; (flip-flop tipo <i>D master/slave</i> com reset). . . . .   | 169 |
| Figura 2.8  | - Módulo <i>ms rs2</i> ; (flip-flop tipo <i>D master/slave</i> com resets independentes). . . . .  | 169 |
| Figura 2.9  | - Módulo <i>ms set</i> ; (flip-flop tipo <i>D master/slave</i> com set). . . . .   | 170 |
| Figura 3.1  | - Módulo <i>wr.add.gen</i> ; (gerador de endereços de escrita). . . . .  | 172 |
| Figura 3.2  | - Módulo <i>rd.add.gen</i> ; (gerador de endereços de leitura). . . . .  | 173 |
| Figura 3.3  | - Módulo <i>logic</i> ; (lógica de controle do módulo <i>rd.add.gen</i> ). . . . .   | 174 |
| Figura 3.4  | - Módulo <i>10.bit.sync.count</i> ; (contador síncrono de 10 bits). . . . .  | 175 |
| Figura 3.5  | - Módulo <i>528.bit.cell</i> ; (célula de armazenagem; capacidade: 528 bits). . . . .  | 176 |

|             |  |     |
|-------------|--|-----|
| Figura 3.6  | - Módulo <i>512.bit.cell</i> ; (célula de armazenagem; capacidade: 512 bits). . . . .                        | 177 |
| Figura 3.7  | - Módulo <i>64.bit.cell</i> ; (célula de armazenagem; capacidade: 64 bits). . . . .                          | 178 |
| Figura 3.8  | - Módulo <i>16.bit.cell</i> ; (célula de armazenagem; capacidade: 16 bits). . . . .                          | 179 |
| Figura 3.9  | - Módulo <i>8.bit.cell</i> ; (célula de armazenagem; capacidade: 8 bits). . . . .                            | 180 |
| Figura 3.10 | - Módulo <i>cell</i> ; (célula de armazenagem; capacidade: 1 bit). . . . .                                   | 180 |
| Figura 3.11 | - Módulo <i>wr.dec</i> ; (decodificador de escrita). . . . .   | 181 |
| Figura 3.12 | - Módulo <i>rd.dec</i> ; (decodificador de leitura). . . . .   | 181 |
| Figura 4.1  | - Módulo <i>step1</i> ; (verificador da primeira condição do procedimento de alinhamento de quadro). . . . . | 183 |
| Figura 4.2  | - Módulo <i>shift.reg</i> ; (registrador de deslocamento). . . . .   | 184 |
| Figura 4.3  | - Módulo <i>step2</i> ; (verificador da segunda condição do procedimento de alinhamento de quadro). . . . .  | 185 |
| Figura 4.4  | - Módulo <i>step3</i> ; (verificador da terceira condição do procedimento de alinhamento de quadro). . . . . | 186 |
| Figura 4.5  | - Módulo <i>counter</i> ; (contador temporizador). . . . .   | 186 |
| Figura 4.6  | - Módulo <i>9.bit.sync.count</i> ; (contador síncrono de 9 bits). . . . .                                    | 187 |
| Figura 4.7  | - Módulo <i>st.m</i> ; (máquina de estados). . . . .   | 187 |
| Figura 4.8  | - Módulo <i>j.count</i> ; (contador <i>johnson</i> ). . . . .  | 188 |
| Figura 5.1  | - Módulo <i>ring.counter</i> ; (contador em anel; esquema parcial 1 de 4). . . . .                           | 190 |
| Figura 5.2  | - Módulo <i>ring.counter</i> ; (contador em anel; esquema parcial 2 de 4). . . . .                           | 191 |
| Figura 5.3  | - Módulo <i>ring.counter</i> ; (contador em anel; esquema parcial 3 de 4). . . . .                           | 192 |
| Figura 5.4  | - Módulo <i>ring.counter</i> ; (contador em anel; esquema parcial 4 de 4). . . . .                           | 193 |

|            |  |     |
|------------|--|-----|
| Figura 5.5 | - Módulo <i>3.bit.sync.count</i> ; (contador síncrono de 3 bits). . . .                      | 194 |
| Figura 5.6 | - Módulo <i>tmg.gen</i> ; (gerador de temporização para a transmissão de 64 kbit/s). . . . . | 194 |
| Figura 5.7 | - Módulo <i>5.bit.count</i> ; (contador assíncrono de 5 bits). . . . .                       | 195 |
| Figura 6.1 | - Módulo <i>8.bit.shift.reg</i> ; (registrador de deslocamento de 8 bits). . . . .           | 197 |
| Figura 6.2 | - Módulo <i>line.coder</i> ; (codificador <i>G.703</i> ). . . . .                            | 197 |
| Figura 6.3 | - Módulo <i>5.bit.count</i> ; (contador assíncrono de 5 bits). . . . .                       | 198 |

## RESUMO

Este trabalho tem como objetivo a especificação de um conjunto de módulos de circuito, capazes de executar as funções de multiplexação relativas à Hierarquia Digital *E1*, do *ITU-T*.

A fim de alcançar o objetivo proposto, são realizadas consultas às normas nacionais (Práticas Telebrás) e internacionais (*ITU-T*) e estudos bibliográficos acerca do projeto e implementação dos circuitos executores das funções de multiplexação *E1*.

Após a fase inicial de especificação e projeto dos módulos de circuito, os mesmos foram submetidos a intensa rotina de simulações, a fim de validar as arquiteturas propostas.

Os módulos de circuito projetados são apresentados e descritos. Os resultados das simulações desenvolvidas são mostrados e comentados.

Os módulos poderão ser empregados na produção de um *chip-set* de circuitos integrados para implementação de equipamentos multiplexadores *E1*.

### **PALAVRAS-CHAVES:**

Microeletrônica, Multiplexadores, Comunicação Digital de Dados, Redes Corporativas, Projeto de Sistemas Digitais.

**TITLE:** “*E1* MULTIPLEXING FUNCTIONS”

## **ABSTRACT**

The aim of this work is the specification of a set of circuit modules, capable to perform the multiplexing functions related to the *ITU-T's E1* Digital Hierarchy.

In order to reach the proposed objective, the national (Telebrás Practices) and international (*ITU-T*) standards are consulted, as well as bibliographical references for the design and implementation of *E1* multiplexing functions performing circuits.

After the initial step of specification and design of the circuit modules, they are submitted to an intensive simulation routine, in order to validate the proposed architectures.

The designed circuit modules are presented and described. The simulation results are shown and discussed.

These modules may be used in the production of an integrated circuits chip-set, for implementation of *E1* multiplexing equipments.

### **KEYWORDS:**

Microelectronics, Multiplexers, Digital Data Communication, Corporate Networks, Digital Systems Design.

# 1 INTRODUÇÃO

Desde os primórdios da telefonia, a técnica de multiplexação é empregada com a finalidade de compartilhar os meios de comunicação, visando uma otimização do aproveitamento destes. Os primeiros sistemas de multiplexação, então analógicos, utilizavam a técnica *FDM* [KAI 94] para acomodar diversos canais telefônicos de banda passante limitada, em um único meio de transmissão de banda larga, resultando em menor número de cabos necessários à implementação de circuitos telefônicos. Com isto conseguiu-se realizar economia de escala, reduzindo-se os custos dos serviços. Com o passar do tempo, à medida em que a tecnologia avançou, foram surgindo novos dispositivos capazes de tornar mais efetivo o emprego dos serviços de telecomunicações. Assim, com a introdução das técnicas de multiplexação no tempo conhecida como *TDM* [KAI 94], as redes públicas de telefonia sofreram mudanças estruturais significativas, num processo que viria a resultar na digitalização das telecomunicações de um modo geral e em um redirecionamento de suas atribuições na sociedade moderna.

Dentro deste contexto, os equipamentos sobre os quais estão atualmente estruturadas as operadoras de serviços de telecomunicações, assumem um papel fundamental. Um destes equipamentos é o multiplexador *TDM E1*, que será objeto de análise neste trabalho.

Inicialmente será realizada uma análise retrospectiva das redes telefônicas e da comunicação de dados, a partir da introdução de técnicas digitais nas redes públicas de telefonia, até os dias atuais, buscando com isto caracterizar o ambiente natural de aplicação do *MUX E1*, as *Redes Integradas*. Em seqüência, será feito um estudo das características do *MUX E1* e das normas internacionais que definem sua estrutura visando a padronização de sistemas, culminando com uma proposta de arquitetura direcionada à implementação em microeletrônica dos circuitos lógicos capazes de desenvolver as funções de multiplexação propriamente ditas. Paralela-

mente a isto, serão abordados tópicos considerados relevantes ao desenvolvimento deste trabalho. Desta forma, aspectos de *timing*, aplicabilidade da arquitetura proposta entre outros, serão também citados ou analisados.

Do ponto de vista da implementação, parte da arquitetura proposta será mapeada a nível de portas lógicas, utilizando a metodologia *standard-cells*, dentro do ambiente de projeto *SOLO-CADENCE* [CAD 88], onde também serão realizadas as simulações dos circuitos. Concluindo o trabalho, serão analisados os resultados, sendo então apontadas as diretrizes para sua consecução e futuro desenvolvimento.

## 1.1 Digitalização em Redes de Telefonia Pública

O início da década de sessenta representou um marco para as telecomunicações de um modo geral e para a telefonia em particular. Até então, as redes telefônicas eram suportadas por tecnologia totalmente analógica, sendo que a estrutura das centrais de telefonia era constituída, entre outros dispositivos, por multiplexadores, que eram empregados com a finalidade de racionalizar o uso dos meios de transmissão. Estes equipamentos, empregando a técnica de divisão espectral de frequência, proporcionavam a multiplexação *FDM*, a qual apresenta determinadas características indesejáveis.

Um sinal analógico presente à saída de um multiplexador *FDM*, ao percorrer um meio de transmissão, sofre crescente degradação à medida em que se propaga, comprometendo a qualidade da informação na extremidade receptora do sistema. Do ponto de vista econômico, a implementação de um multiplexador *FDM* apresenta custos intrínsecos bastante elevados, devido a complexidade de sua arquitetura. Este fato tornava viável sua aplicação somente para enlaces de longo alcance. Desta forma, dependendo da extensão e da quantidade de pares necessários à implementação de um conjunto de linhas de assinante, era preferível optar-se pelo

emprego de cabos multipares, cada um destes pares suportando um único canal telefônico, ao invés de utilizar equipamentos de multiplexação *FDM*.

Entretanto, a partir de 1962, teve início dentro das redes públicas de telefonia, o processo de migração da tecnologia analógica, para a tecnologia digital. Neste ano, a AT&T colocava em operação comercial nos *EUA*, o sistema de multiplexação *T1*, desenvolvido nos Laboratórios Bell. Este sistema, utiliza a técnica *PCM* idealizada na França em 1939, para codificar 24 canais telefônicos de voz analógicos, convertendo-os em palavras binárias que são multiplexadas em tempo (multiplexação *TDM*), agregando os 24 canais de baixa velocidade (64 kbit/s), para serem transmitidos através de um único canal de alta velocidade de 1,544 Mbits/s. Por empregar transmissão digital, a degradação do sinal de saída de um *MUX T1* ao longo do meio de transmissão não é cumulativa, uma vez que o mesmo é de trechos em trechos regenerado. Na regeneração, o sinal de entrada degradado é identificado simbolicamente e em seguida é descartado, sendo substituído por um novo, de mesmo símbolo, livre de distorções. Este processo difere radicalmente do processo de amplificação dos sinais analógicos em que é impossível separar a informação do ruído.

Com a introdução dos equipamentos *TDM*, suportados por técnicas digitais e implementados de forma econômica devido à aplicação da microeletrônica, o emprego dos multiplexadores *TDM* tornou-se extensivo mesmo para pequenos trechos de rede. Com isto, possibilitou-se o assim chamado ganho de pares [BEL 91], em que vários usuários geograficamente próximos entre si, eram conectados a uma central de comutação local, através de um único par telefônico. Atualmente, a multiplexação por divisão de tempo, constitui a base de qualquer rede digital de comunicações em todo o mundo.

Pode-se então concluir que o processo de introdução de tecnologia digital nas redes telefônicas foi motivado pela necessidade de melhorar a qualidade dos serviços, conferindo-lhes maiores confiabilidade e robustez, acrescentar novas facilidades e reduzir os custos decorrentes da tecnologia analógica até então empregada.

A migração da tecnologia analógica para a digital não visava diretamente portanto, o atendimento das necessidades da emergente indústria de comunicação de dados por suportes mais adequados à transmissão de dados. Justifica-se esta afirmação ao constatar-se que no início do processo, a maior parte da tecnologia digital introduzida nas redes era inacessível ao tráfego de dados. Isto devia-se ao fato de que ainda que a transmissão entre centrais fosse digital, toda a comutação e linhas de assinantes ainda eram implementadas de forma analógica. É evidente que uma rede digital é um ambiente natural para os serviços de comunicação de dados, sendo que à medida em que o processo de digitalização vai alastrando-se de dentro para fora das redes em direção ao usuário final, a utilização das facilidades digitais torna-se disponível em maior grau para as aplicações de comunicação de dados. Assim, a telefonia e a comunicação de dados tendem a se encontrar nas chamadas *Redes Digitais Integradas*. Para justificar esta assertiva, é interessante analisar também a evolução histórica da Comunicação de Dados.

## 1.2 Evolução da Comunicação de Dados

Com a viabilização comercial do uso de computadores, no início dos anos sessenta, surgiram os primeiros sistemas de teleprocessamento: as redes de terminais. Utilizando a rede telefônica pública como suporte, as linhas telefônicas eram o meio de transmissão empregado para interligar terminais de consulta a um computador central. Tipicamente, um terminal é caracterizado por um teclado (entrada de dados) e um monitor de vídeo (saída de dados), constituindo um sistema de pouca ou nenhuma capacidade de processamento local; todo o processamento era realizado no computador central. Este por sua vez, caracterizava-se por apresentar dimensões físicas muito grandes, sendo um equipamento de custo elevadíssimo, além de exigir instalações específicas para seu funcionamento e manutenção operacional constante.

Durante a década de setenta, intensificou-se a utilização de sistemas de teleprocessamento, introduzidos na década anterior. Na indústria, os (grandes) fa-

bricantes de computadores consolidam o conceito de arquiteturas proprietárias, na qual os sistemas são fechados, ou seja, não aceitam a incorporação de equipamentos de outros fabricantes em seu ambiente. Assim, a título de exemplo, era comum um determinado usuário possuir um sistema constituído por equipamentos *IBM* e outro constituído por equipamentos *Burroughs*. O processamento centralizado continuava sendo a topologia adotada na implementação dos sistemas. Ainda neste período, surgiram os primeiros movimentos para padronização de sistemas, com a introdução do modelo de referência *OSI* [TAN 88].

A intensificação do uso das facilidades oferecidas pela rede pública de telefonia por parte dos usuários de comunicação de dados, levou ao desenvolvimento de técnicas especiais para otimizar a utilização da estreita banda passante das linhas telefônicas. Entre estas técnicas destacam-se a Modulação em Amplitude por Quadratura, a Equalização Adaptativa e a Codificação em Treliça [LEE 90].

No início da década de 80, podia-se detectar o rápido crescimento da utilização da tecnologia de comunicação de dados. Nesta época, de acordo com [OWE 82], em muitos países o número de assinantes telefônicos havia dobrado em relação à década passada, ao passo que, curiosamente, a taxa de utilização do serviço telefônico apresentava um crescimento bem mais elevado, para o mesmo período. A área de comunicação de dados e o início do uso extensivo de facilidades como fax, vídeo-conferência e outras aplicações, foi identificada como responsável pelo aumento na utilização dos recursos telefônicos. Assim, o uso das linhas telefônicas para transportar dados começou a tornar-se significativo.

Com base na massificação dos microprocessadores e na miniaturização dos equipamentos, abriu-se espaço para uma nova proposta de topologia para as redes de comunicação de dados: ao invés de redes de terminais conectados a um processador de grande porte, surgiram as redes de computadores, nas quais os terminais passaram a apresentar boa capacidade de processamento local, descentralizando a inteligência do sistema e distribuindo-a ao longo da rede. Este processo, batizado por *Downsizing*, viria a revolucionar a área de comunicação de dados. Com a popu-

larização dos microcomputadores, surgiram as redes locais de computadores (*LAN*) [SPO 93], aparecendo também a pressão por parte de seus usuários, pelo estabelecimento de padrões a serem adotados pelos fabricantes, de modo a tornar possível a interoperabilidade dos sistemas.

Atualmente, tem sido crescente a demanda dos usuários por canais capazes de operar com taxas de transmissão mais elevadas. Novas aplicações como a multimídia, requerem meios de comunicação capazes de oferecer bandas mais largas. A linha de assinante telefônico, com sua inerente limitação de banda passante em torno de taxas não maiores que 30 kbit/s, mostra-se inadequada, ficando evidente a necessidade de uma infra-estrutura de comunicações mais apropriada a estas atuais demandas.

Assim, a partir de dois pontos de partida diferentes, quais sejam o processo de digitalização das redes telefônicas e o surgimento e evolução da comunicação de dados, percebe-se uma tendência natural de convergência destas duas áreas para um mesmo ponto: um sistema de comunicação digital, na qual um canal de usuário telefônico pode ser empregado para transmissão de dados. A este sistema deu-se o nome de *Rede Digital Integrada*. Uma evolução da Rede Digital Integrada é a chamada *Rede Digital de Serviços Integrados* em que é possível a co-existência de múltiplos tipos de serviços tais como telefonia digital, comunicação de dados, vídeo digital por demanda, áudio digital de alta fidelidade, fax, telemetria e outros. A integração de diferentes serviços de comunicação é e continuará a ser daqui para a frente, uma tendência mundial no campo das comunicações em geral.

Entretanto, para que seja possível a viabilização das redes integradas, é fundamental que os aspectos estruturais dos equipamentos sobre os quais as redes serão implementadas, sejam definidos através de normas técnicas ou padrões.

Assim, como resultado de um esforço conjunto de entidades ligadas às comunicações de um modo geral em diferentes países, o *ITU-T*, antigo *CCITT* (Comitê Consultativo Internacional para Telefonia e Telegrafia) através das recomendações

da série "G" [ITU 88], foi o responsável pela edição de normas técnicas visando a padronização dos sistemas de comunicação digital. Desta forma, no volume III da série "*Blue Book*", fascículo III.4, estão descritos os aspectos gerais para sistemas digitais, no que tange aos equipamentos de transmissão.

Especificamente, este trabalho tem por objetivo definir estruturalmente um conjunto de módulos na forma de circuitos integráveis em silício, capaz de prover as funções de multiplexação definidas nas recomendações da série "G" do *ITU-T*, particularmente no que diz respeito ao equipamento de multiplexação do nível 1 daquelas recomendações.

O multiplexador de nível 1 do *ITU-T*, denominado *MUX E1*, é um multiplexador digital *TDM* síncrono, a partir do qual é possível a estruturação de Redes Digitais Integradas.

As funções de multiplexação *E1* constituem-se portanto no conjunto de procedimentos necessários à implementação dos equipamentos multiplexadores *E1*. Entre estas funções, pode-se citar a codificação e decodificação de linha dos sinais de informação, recuperação de relógio, procedimentos de verificação de redundância cíclica, montagem, desmontagem e alinhamento de quadros e multiquadros, procedimentos de supervisão das condições do enlace de comunicação, funções de teste, entre outras.

Cabe ressaltar entretanto, que neste trabalho não serão abordadas de maneira enfática as funções ligadas a interfaceamento analógico de sinais, tais como conversão *A/D* e *D/A*, amplificação e regeneração de sinais de recepção, equalização, excitação de linha e funções implementáveis por técnicas analógicas. Somente as funções passíveis de implementação mediante projeto digital de circuitos *CMOS*, é que serão estudadas com maior nível de detalhe.

Assim sendo, no capítulo que segue, serão apresentadas as aplicações do multiplexador *E1* na estruturação da Rede Pública *E1* da Embratel, bem como

exemplos de aplicação do multiplexador *E1* na implementação de redes integradas de uso privado. Este capítulo tem por finalidade contextualizar o trabalho a partir de sua aplicabilidade.

Nos capítulos subseqüentes, iniciar-se-á uma abordagem mais objetiva, orientada mesmo à arquitetura do *MUX E1*, do ponto de vista da especificação dos circuitos executores das funções de multiplexação *E1*. Para tanto, um estudo detalhado e cuidadoso das recomendações do *ITU-T* que orientam a implementação das funções de multiplexação *E1*, também é incluído neste trabalho. Uma consistente aderência às normas é fundamental para a existência de compatibilidade que venha a permitir a interoperabilidade de sistemas de diferentes fabricantes. As normas e recomendações que orientam a arquitetura dos circuitos executores das funções de multiplexação estudadas, estão relacionadas na listagem bibliográfica, constante no final deste trabalho.

## 2 APLICAÇÕES DO MULTIPLEXADOR *E1*

A consolidação das diferentes formas de intercâmbio de informação surgidas na sociedade contemporânea a partir da década passada, tem exigido a disponibilização de recursos para um atendimento global das premissas de comunicação (dados, voz, fax e outros), na forma de redes digitais integradas com acessos em diversas taxas. Neste capítulo será destacada a importância do *MUX E1* na estruturação destas redes, bem como exemplos de implementação.

### 2.1 Rede *E1* Embratel

No Brasil, utilizando multiplexadores *E1* como elemento básico de rede, a Embratel disponibiliza a sua Rede de Serviços Digitais *E1*, que consiste estruturalmente de:

- Roteadores de 2048 kbit/s com capacidade para rotear canais de 64 kbit/s e canais de baixa velocidade.
- Multiplexadores de 2048 kbit/s (*E1*)
- Sub-multiplexadores capazes de agregar canais de até 19200 bit/s em feixes de 64 kbit/s. Com estes equipamentos, é possível o acesso à rede de canais com taxas de dados inferiores a 64 kbit/s (canais *sub-rate*).
- Sistema de Gerência e Supervisão estruturado de acordo com a recomendação *M.30* do *ITU-T*.

A Rede *E1* da Embratel foi projetada para oferecer uma grande capacidade de roteamento, sendo configurada nos assim chamados Centros Básicos, nos prédios da Embratel espalhados pelo país. Estes centros interligam os equipamentos de acesso instalados nos locais do usuário.

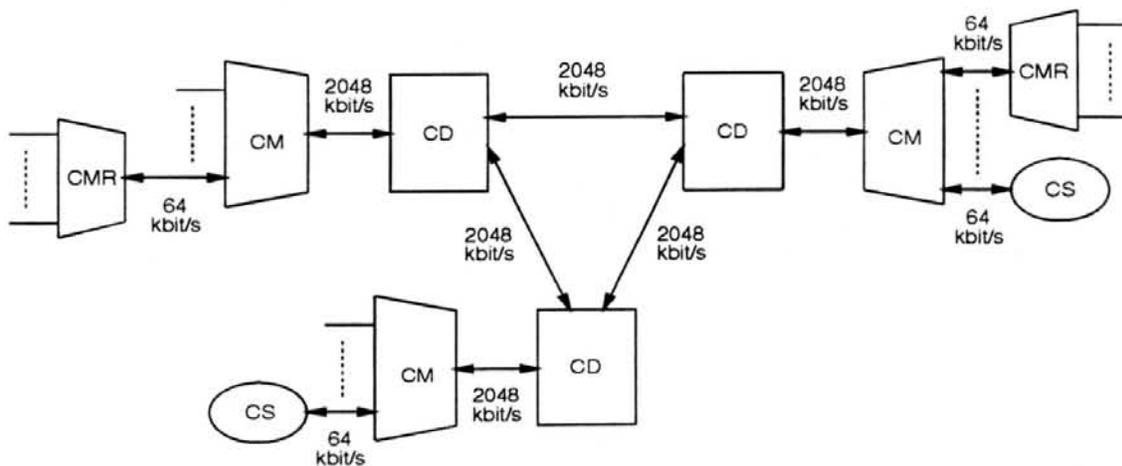


Figura 2.1 - Topologia básica da Rede *E1* Embratel.

A topologia da rede permite a crescente integração dos serviços de voz, dados, fax, vídeo e outros. Esta topologia pode ser vista na figura 2.1.

A arquitetura da rede compõe-se de cinco tipos de centros de equipamentos; estes são:

- CD: Centro Distribuidor (roteadores);
- CM: Centro Multiplexador (multiplexadores *E1*);
- CMR: Centro Multiplexador Remoto a 64 kbit/s (sub-multiplexadores);
- CS: Concentrador de Serviços (sub-multiplexadores);
- CSG: Centro de Supervisão e Gerência de Rede (*workstations e software*, não mostrados na figura).

Os Centros Distribuidores (*CD*) são responsáveis pelo roteamento e distribuição da rede. Na rede da Embratel, estes equipamentos tem capacidade máxima de 256 agregados de entrada e saída (um agregado corresponde a um feixe *E1* de 2048 kbit/s), existindo contudo equipamentos menores, capazes de rotear 32 agregados de entrada e saída. Os Centros Distribuidores localizam-se principalmente nas capitais e grandes centros urbanos do país.

Os Centros Multiplexadores são responsáveis pelo atendimento das altas concentrações de acesso, através do emprego do multiplexador *E1*, que agrega 31 canais de informação de 64 kbit/s e um canal de serviço também de 64 kbit/s em um canal de 2048 kbit/s. Os Centros Multiplexadores podem ser instalados tanto nos prédios da Embratel das capitais e grandes cidades, como também no ambiente do usuário.

Os Centros Multiplexadores Remotos atendem as médias concentrações de acesso (canais *sub-rate*). Assim, taxas como 4800 bit/s, 9600 bit/s, até 19200 bit/s são multiplexadas, dando origem a um agregado de 64 kbit/s, que é encaminhado ao centro multiplexador. O Centro Multiplexador Remoto, é instalado nas dependências do usuário.

Os Concentradores de Serviços destinam-se ao atendimento de baixas concentrações de acesso (canais *sub-rate*), também no ambiente do usuário. São os menores equipamentos da rede *E1* da Embratel. Sua saída consiste em um agregado de 64 kbit/s formado por canais de baixa velocidade, que é encaminhado a um centro multiplexador.

A supervisão e gerência da rede é realizada através da instalação de *software* dedicado, rodando sobre *workstations* em locais da rede estrategicamente definidos. Assim, um Centro Distribuidor, um Centro Multiplexador ou mesmo um Centro Multiplexador Remoto, podem vir a constituir-se em um Centro de Supervisão e Gerência de Rede, podendo ainda haver vários Centros de Supervisão e Gerência em uma mesma rede, caracterizando uma descentralização e distribuição da capacidade de gerenciamento e controle.

As funções do Centro de Supervisão e Gerência de Rede, são as seguintes:

- Gerenciamento de configurações: Esta função está relacionada a circuitos, enlaces, partições, redundâncias, delegações e re-roteamentos.

- Gerenciamento de falhas: Fornece representação gráfica do(s) dispositivo(s) em falha, centralização de alarmes, relatórios e facilidades de manutenção (estabelecimento de *loopbacks* de teste).
- Gerenciamento de performance: Fornece diagnósticos e estatísticas de desempenho de toda a rede.

A comunicação entre Centros de Supervisão e Gerência de Rede é suportada por uma rede interna de comutação de pacotes *X.25*. Os canais para tráfego das informações são criados automaticamente quando da interconexão de equipamentos entre os nós de rede.

### 2.1.1 Tipos de Acesso à Rede E1 Embratel

Os usuários da rede *E1* da Embratel podem dispor de vários modos para acessá-la. Assim, nos Centros Distribuidores e nos Centros Multiplexadores, os acessos em baixa velocidade podem ser feitos nas taxas de 1200 bit/s, 2400 bit/s, 4800 bit/s, 9600 bit/s e 19200 bit/s. Em alta velocidade, são possíveis os acessos a 48 kbit/s, 64 kbit/s e  $n \times 64$  kbit/s, este último até 1920 kbit/s (canais *E1* fracionários), além de um acesso em 2048 kbit/s. É possível ainda, acessos para sinais de voz analógicos que podem ser digitalizados em 8 kbit/s (modulação delta), 16 kbit/s (modulação *PCM* adaptativa diferencial) e 64 kbit/s (modulação *PCM*). A rede disponibiliza também acesso *Ethernet* para rede local e sinais de fax do grupo 3.

Nos Centros Multiplexadores Remotos, os acessos podem ser feitos nas taxas de 1200 bit/s, 2400 bit/s, 4800 bit/s, 9600 bit/s e 19200 bit/s, sendo que em alta velocidade é possível acessar a rede nas taxas de 48 kbit/s e 64 kbit/s. Sinais de voz analógicos podem acessar a rede e ser digitalizados em taxas de 8 kbit/s e 16 kbit/s. Ainda nestes centros, existem acessos para sinais de fax do grupo 3.

Nos Concentradores de Serviços, existem acessos de baixa velocidade nas taxas de 1200 bit/s, 2400 bit/s, 4800 bit/s, 9600 bit/s e 19200 bit/s. Sinais de voz analógicos também acessam a rede, podendo ser digitalizados em 8 kbit/s e 16 kbit/s. Os sinais de fax do grupo 3 também podem ter acesso à rede, através dos Concentradores de Serviços.

### 2.1.2 Métodos de Sincronismo na Rede E1 Embratel

O método utilizado para sincronização da rede *E1*, emprega uma topologia mestre-escravo, podendo ser de duas formas:

- *Stand Alone*: quando são utilizadas apenas fontes de sincronismo configuráveis no próprio nó da rede. Desta forma, pode ser possível obter sinal de relógio a partir de entradas externas de 8 kHz, 64 kHz e 2048 kHz ou da regeneração de um sinal de relógio a partir do sinal de uma entrada, em múltiplos de 8 kHz. Uma terceira possibilidade, é o emprego de um gerador de relógio interno ao equipamento.
- *Automatic Network Synchronization (ANS)*: Nesta modalidade, todos os nós recebem sinais de temporização gerados pela mesma fonte, de forma a assegurar-se assim, a manutenção do sincronismo em toda a rede. Este recurso é também conhecido como Função de Sincronismo de Rede [LEE 90].

## 2.2 Redes Corporativas

Até o ano de 1990, todo e qualquer tipo serviço de comunicação de dados no Brasil somente poderia ser explorado pela Embratel. Entretanto, a partir de 1990, com a quebra do monopólio estatal neste setor, passou a ser possível a

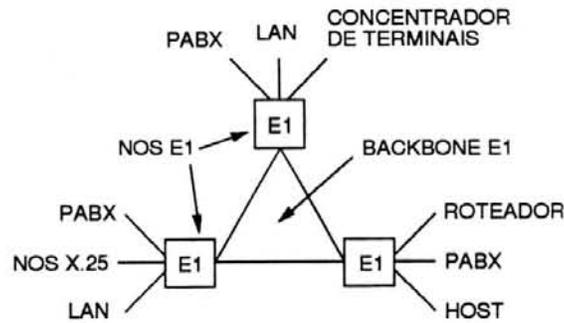


Figura 2.2 - Rede E1 como *backbone*.

estruturação de redes particulares para comunicação de dados, utilizando-se nós de rede (roteadores e multiplexadores) E1. Estas redes particulares foram denominadas Redes Corporativas.

As Redes Corporativas tem como finalidade, atender às necessidades de troca de informações entre os diferentes setores dentro de uma corporação. Genericamente, uma corporação pode ser uma indústria, uma universidade, um banco, etc. Deste modo, a corporação estabelece sua própria infra-estrutura de comunicações, sempre adequando-a a suas particulares necessidades. Isto resulta inicialmente na necessidade de investimentos em equipamentos e instalações, por parte da corporação. Entretanto, a economia proporcionada pelo baixo custo operacional em relação aos serviços públicos, compensa em muito o investimento inicial. Isto se deve ao fato de que ao longo dos anos, o custo dos serviços públicos de comunicações vem se mantendo estável, ao passo que o dos equipamentos vem apresentando constante redução. Entre as causas para a redução do custo dos equipamentos, a microeletrônica destaca-se como uma das mais relevantes.

Atualmente, os sistemas de informação utilizados pelas corporações, são constituídos basicamente por redes locais de computadores. Frequentemente é necessário interligar estas redes, de forma que assim possam compor um sistema de informação global da corporação. Esta rede deve ainda ser capaz de suportar além dos serviços de comunicação de dados, o tráfego de canais telefônicos e, em não raras vezes, vídeo digital, caracterizando portanto, uma Rede Corporativa Integrada. Em

uma rede deste tipo, o multiplexador *E1* pode ser empregado como elemento básico de implementação, em diferentes topologias de rede, sendo importante destacar:

- Rede *E1* como *backbone* para interligação de centros de tráfego;
- Rede *E1* com controle fixo e centralizado;
- Rede *E1* com gerenciamento e controle distribuídos e auto-configurável.

Uma rede do tipo *backbone E1* pode ser empregada para interligar os principais centros de tráfego da corporação. Para tanto, além da multiplexação, os nós *E1* devem ser providos de funções de roteamento. Um exemplo para esta topologia pode ser visto na figura 2.2, em uma rede integrada que interliga redes locais, *PABX* digital, redes de pacotes, além de outros serviços.

As redes *E1* com controle fixo e centralizado, são aplicáveis a estruturas mais modestas, com no máximo três nós e roteamento fixo, controlado centralmente. Os nós são constituídos por multiplexadores *E1*, com capacidade de comutação por comando de operador. Na figura 2.3, é apresentado um exemplo de uma rede com controle centralizado, numa ligação ponto a ponto onde os multiplexadores implementam uma rede integrada voz/dados.

Nas redes *E1* com gerenciamento e controle distribuídos, todos os nós possuem capacidade de processamento, podendo assim cada um destes nós executar tarefas de operação e controle localmente. Cada nó recebe recursos de *hardware* e *software*, que o habilitam para a execução de funções de controle e gerenciamento da rede. Um canal de comunicação entre nós é estabelecido para troca das informações relativas ao controle da rede constituindo assim, uma sub-rede dedicada a esta finalidade.

A topologia com gerenciamento e controle distribuídos proporciona maior autonomia e robustez à rede, uma vez que não existe um nó essencial ao funcionamento de todo o sistema. Um nó ou rota que deixe de operar devido a uma pane

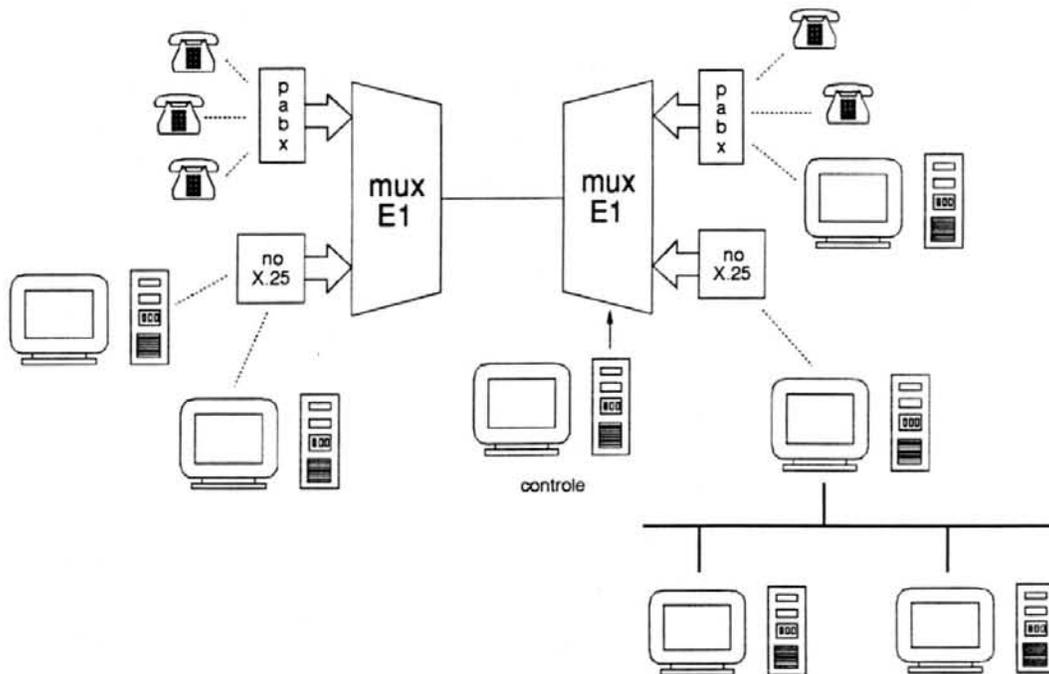


Figura 2.3 - Rede *E1* com controle centralizado.

qualquer, faz com que a rede seja reconfigurada, estabelecendo rotas alternativas automaticamente, de modo a manter o menor número de usuários fora da rede, no período de restabelecimento do setor em falha. Naturalmente, uma rede deste tipo apresenta complexidade e custos de implementação mais elevados, tornando-a indicada para grandes redes, ou redes de alta confiabilidade.

## 2.3 Acesso da Rede Corporativa à Rede Pública de Comunicações

Normalmente, as Redes Corporativas são providas de algum tipo de acesso à rede pública de comunicação (Rede *E1* da Embratel, p.ex.), de modo a ser possível o estabelecimento de enlaces de comunicação sobre distâncias muito grandes, não cobertas pela estrutura própria da rede. Isto é particularmente interessante a pequenas e médias corporações, para as quais não é economicamente viável manter o aluguel de canais de satélite, por exemplo.

Uma outra razão para prover a rede corporativa de acesso à rede pública, é a interligação eventual de redes corporativas distintas. Desta forma, a estrutura pública serve como elemento de interligação entre as redes corporativas.

No capítulo seguinte serão estudados os aspectos relativos à padronização em redes digitais de comunicação.

### 3 PADRONIZAÇÃO DA HIERARQUIA DIGITAL *ITU-T*

A fim de que equipamentos de diferentes fabricantes possam coexistir em uma mesma rede de comunicações, é necessária a existência de padrões que definam a interoperabilidade dos dispositivos, de forma a serem evitadas as arquiteturas proprietárias. Assim, aspectos tais como formato de quadros, taxas de bit, interfaces, padrões de sincronização e outros, precisam ser consistentemente definidos.

Os estudos normativos das telecomunicações a nível internacional, estão a cargo do *ITU* (*International Telecommunications Union*). Os aspectos relacionados com a comunicação de dados no *ITU*, são elaborados no âmbito do *ITU-T*. No Brasil, as recomendações do *ITU-T* são acolhidas e adaptadas pela *ABNT* (Associação Brasileira de Normas Técnicas), através das assim chamadas Práticas Telebrás.

Tendo em vista que os multiplexadores *TDM E1* são equipamentos do tipo terminal, dentro de um sistema de transmissão digital, as recomendações e especificações técnicas da série "G" do *ITU-T*, aplicáveis a estes equipamentos, tratam dos aspectos gerais de sistemas de transmissão e das principais características da multiplexação digital, sendo interessante destacar especificamente neste trabalho, algumas destas recomendações:

- *G. 702* - Hierarquias Digitais e taxas de bit associadas.
- *G. 703* - Características físicas e elétricas das interfaces na Hierarquia Digital.
- *G. 704* - Estruturas de quadros síncronos dos níveis 1 e 2 da Hierarquia Digital.
- *G. 706* - Procedimentos relacionados com sincronismo de quadro e *CRC*, em estruturas básicas de quadro.

- *G. 736* - Características de um equipamento multiplex digital síncrono, operando a 2.048 Kbit/s.
- *G. 823* - O Controle do *Jitter* e *Wander* nas redes digitais baseadas na hierarquia de 2048 kbit/s.

### 3.1 Rec.G. 702 - Hierarquias Digitais e Taxas de Bit Associadas

O *ITU-T* definiu duas hierarquias para multiplexação digital, cada uma das quais constituída de 4 níveis. Assim, para redes cuja taxa de dados de nível 1 é igual a 1.544 kbit/s, foi definida a hierarquia digital denominada *T1*.

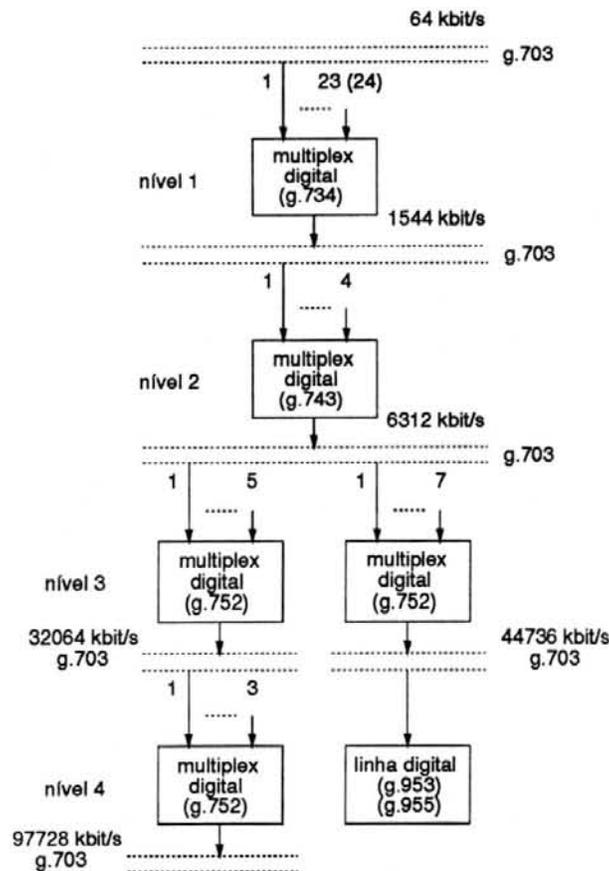


Figura 3.1 - Hierarquia Digital *ITU-T T1*.

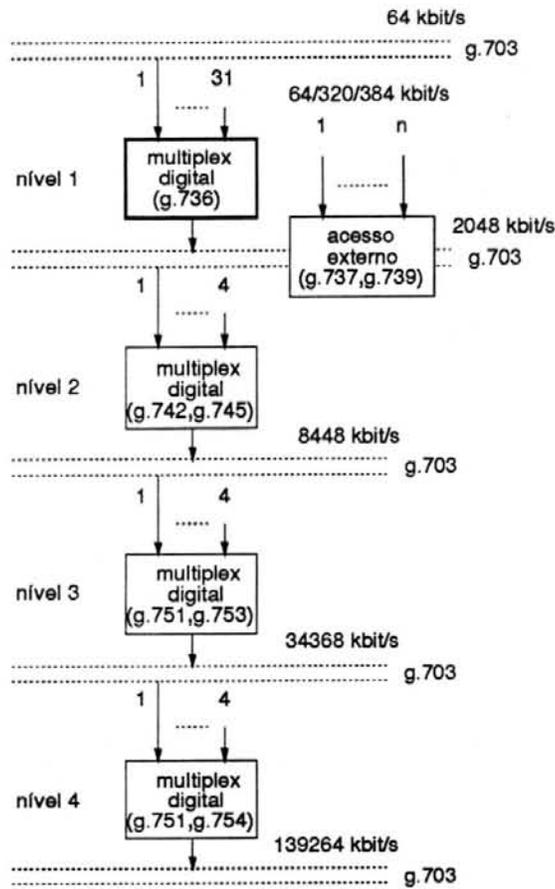


Figura 3.2 - Hierarquia Digital ITU-T E1.

Para redes em que a taxa de dados de nível 1 é de 2.048 kbit/s, foi definida a hierarquia digital conhecida como *E1*.

A hierarquia *T1* é adotada na América do Norte, ao passo que a hierarquia *E1* é adotada na Europa e México. A Ásia dividiu-se na adoção dos padrões, tendo alguns países optado pela hierarquia *E1* e outros pela hierarquia *T1*.

As normas técnicas têm sido elaboradas com base na dominância tecnológica histórica das nações. Assim, como a maioria dos satélites de comunicação de dados têm sido produzidos e postos em órbita pelos EUA, consolidou-se nesta área a hierarquia *T1*. Por outro lado, os cabos submarinos de fibra óptica lançados no Atlântico e no Pacífico, foram estruturados de acordo com a hierarquia *E1*, denotando a influência européia.

No Brasil, a exemplo da Europa, foi adotada a hierarquia *E1*. As figuras 3.1 e 3.2 mostram os equipamentos multiplexadores pertencentes às hierarquias *T1* e *E1*. O multiplexador que aparece em destaque na figura 3.2, é o multiplexador *E1*, que será objeto de estudo detalhado, ao longo deste trabalho.

Conforme pode ser visto na figura 3.2, o *MUX E1* disponibiliza 31 canais operando a 64 kbit/s cada um. Estes canais são também conhecidos como **tributários**. Internamente, o *MUX* reserva um canal de 64 kbit/s (canal zero) para sincronização, controle e comunicação entre multiplexadores. Este canal é denominado **canal de serviço**. Assim, os 31 canais tributários e o canal de serviço são multiplexados em tempo, originando o canal de alta velocidade a 2048 kbit/s, ou **agregado**, como também é conhecido.

## 3.2 Rec. *G.703* - Características Físicas e Elétricas das Interfaces

A recomendação *G.703* especifica e padroniza as interfaces necessárias à viabilização da interconexão dos componentes de uma rede digital. Assim, para os tributários foi definida a interface de 64 kbit/s e para o agregado definiu-se a interface de 2048 kbit/s. A seguir, estas duas interfaces serão detalhadas.

### 3.2.1 Interface de 64 kbit/s

Na interface de 64 kbit/s, três sinais devem ser transportados em ambos os sentidos de transmissão:

- Sinal de informação a 64 kbit/s;
- sinal de temporização de 64 kHz;

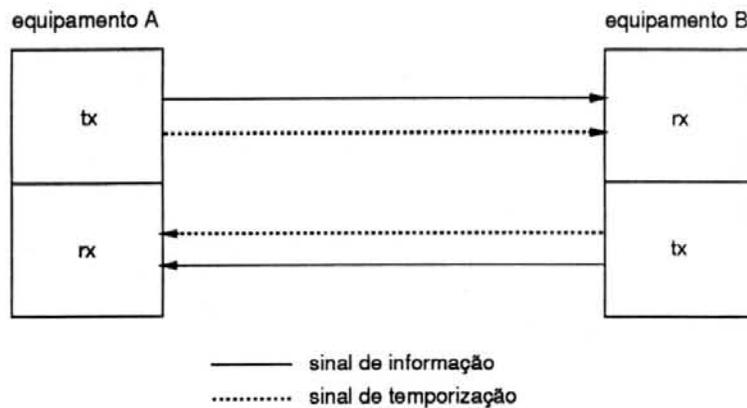


Figura 3.3 - Interface co-direcional.

- sinal de temporização de 8 kHz.

O sinal de informação a 64 kbit/s são os dados *NRZ* codificados segundo a regra *G.703*. O sinal de temporização de 64 kHz é incorporado à seqüência transmitida ao longo da linha, através das transições inter-níveis do sinal codificado. Estas transições, no lado receptor, possibilitarão a recuperação do relógio de temporização, que será fundamental para a decodificação da informação. O sinal de temporização de 8 kHz, é gerado a partir da regra de violação do código *G.703*; sua finalidade é possibilitar a organização dos dados de tributário, em palavras de oito bits denominadas **octetos**, que serão utilizadas na composição do quadro de dados a 2.048 kbit/s.

#### 3.2.1.1 Temporização de Transmissão

Quanto à temporização, o *ITU-T* recomenda três maneiras possíveis:

- Interface co-direcional: O termo co-direcional é empregado para descrever uma interface sobre a qual a informação e sua temporização associada, são transmitidas no mesmo sentido. A figura 3.3 ilustra a interface co-direcional.

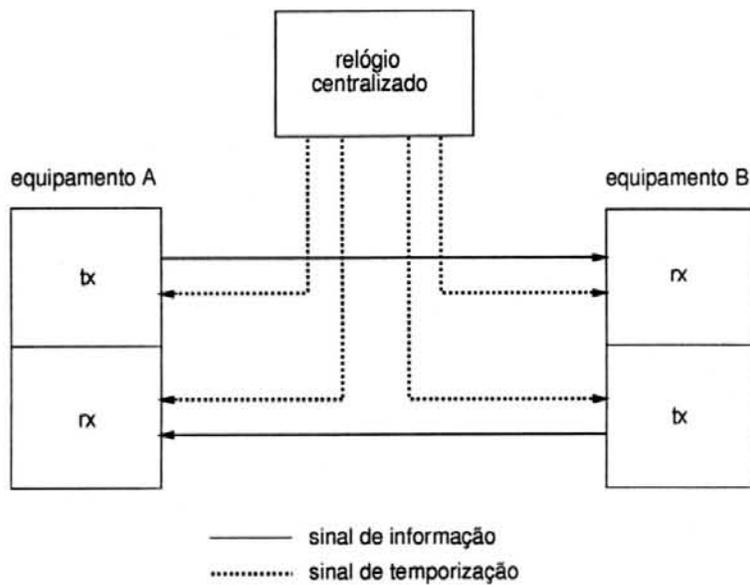


Figura 3.4 - Interface com relógio centralizado.

- Interface com relógio centralizado: O termo relógio centralizado é empregado para descrever uma interface em que, para ambos os sentidos de transmissão do sinal de informação, o sinal de temporização associado é fornecido a partir de um relógio centralizado, que pode ser derivado por exemplo, de sinais que chegam da linha de transmissão, ou fornecido externamente ao equipamento. A figura 3.4, mostra a interface com relógio centralizado.
- Interface contra-direcional: O termo contra-direcional é empregado para descrever uma interface sobre a qual os sinais de temporização associados a ambos os sentidos de transmissão, são dirigidos do equipamento controlador ao equipamento subordinado, conforme pode ser visto na figura 3.5.

As funções de multiplexação *E1* descritas neste trabalho, serão especificadas para operar com interfaces co-direcionais.

De acordo com a nota 2 do parágrafo 5, da recomendação *G.736*, a porta de entrada para esta interface deve ser capaz de identificar os octetos, a partir da regra de violação inerente ao código *G.703*, a fim de controlar a ocorrência de *slips*,

quando as temporizações do tributário e do multiplexador forem *pleσιόcronas*. O conceito de *slip* e *pleσιόcrono*, será apresentado no capítulo "Aspectos Funcionais do Multiplexador E1". A especificação das portas de entrada para os canais tributários, deve também prever meios para absorção de *jitter* e *wander*, dentro dos limites estabelecidos na recomendação G.823, analisados na seção 3.6 deste capítulo.

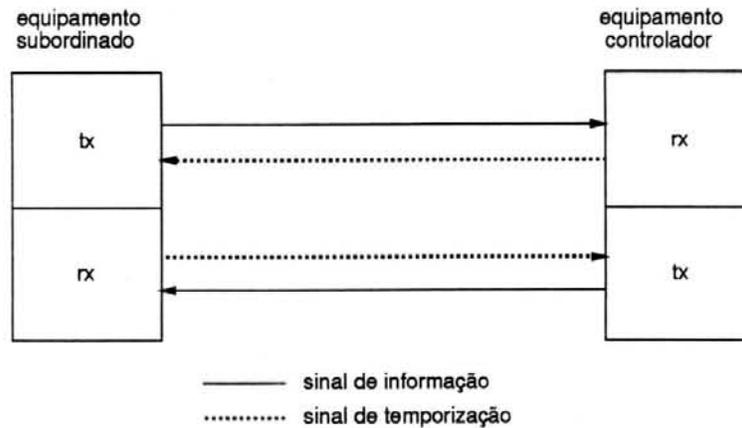


Figura 3.5 - Interface contra-direcional.

### 3.2.1.2 Características Elétricas da Interface Co-direcional

Esta interface é atualmente a única adotada em equipamentos multiplexadores digitais com canais tributários de 64 kbit/s. As recomendações para as características elétricas da interface co-direcional, estabelecem uma taxa nominal de 64 kbit/s, com uma tolerância máxima de +/- 100 ppm. As regras para converter os dados *NRZ* em código de linha, são as seguintes:

- O intervalo de tempo correspondente a um bit de dados *NRZ*, é dividido em quatro sub-intervalos de tempo;
- o dígito binário *um* é codificado como um bloco composto pelos seguintes quatro bits: 1 1 0 0;
- o dígito binário *zero* é codificado como um bloco composto pelos seguintes quatro bits: 1 0 1 0;

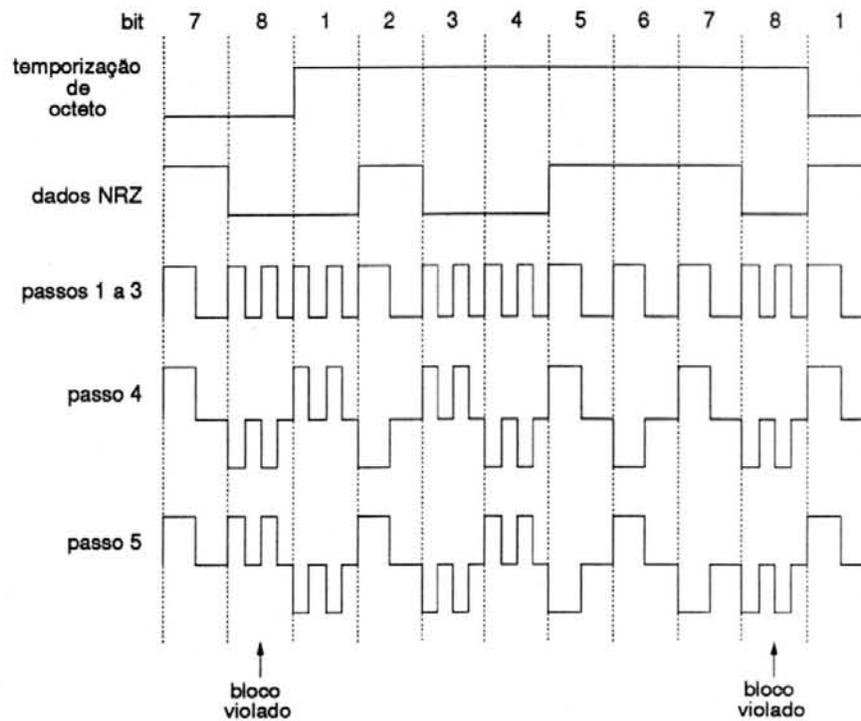


Figura 3.6 - Conversão de código para interface co-direcional.

- o sinal binário é convertido em um sinal de três níveis, alternando-se a polaridade de blocos consecutivos;
- a alternância na polaridade dos blocos é violada a cada oito blocos. Com isto, é possível a identificação dos octetos. A figura 3.6, mostra a codificação de uma seqüência de dados genérica, como exemplo.

As características elétricas dos sinais de entrada e saída da interface *G.703* de 64 kbit/s co-direcional, são as seguintes:

- Taxa de símbolos: 256 kbaud.
- Forma de pulso (nominalmente retangular): de acordo com as figuras 3.7 e 3.8.
- Pares trançados em cada direção (saída/entrada): um par simétrico.
- Impedância de teste (entrada e saída): 120 ohm, resistiva.

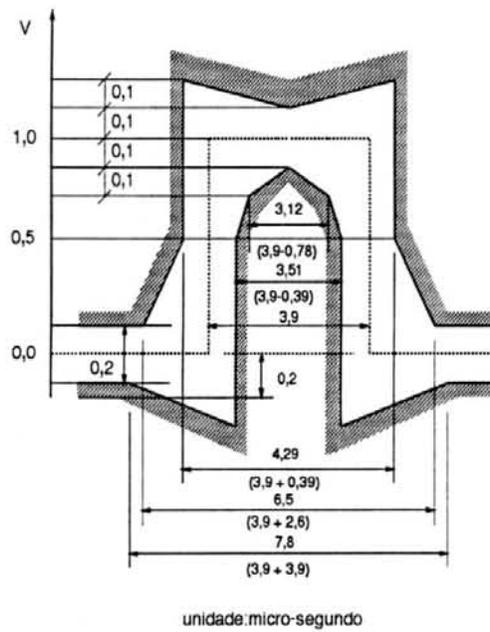


Figura 3.7 - Máscara para pulso único.

- Tensão de pico nominal de um pulso de marca: 1,0 volt.
- Tensão de pico nominal de um pulso de espaço: 0,0 +/- 0,1 volt.
- Largura nominal de um intervalo de bit de 64 kbit/s: 3,90625  $\mu$ s.
- Relação entre a amplitude positiva e negativa dos pulsos no centro do intervalo: 0,95 a 1,05.
- Relação entre a largura dos pulsos positivos e negativos no centro do intervalo: 0,95 a 1,05.
- *Jitter* (flutuação de fase) pico a pico máxima na porta de saída: De acordo com o especificado na recomendação *G.823*.

As interfaces de 64 kbit/s com relógio centralizado e contra-direcional, não são atualmente empregadas em equipamentos multiplex, apesar de estarem definidas nos parágrafos 1.2.2 e 1.2.3, da recomendação *G.703*. O código de pulsos definido para estas interfaces é o *AMI*.

### 3.2.2 Interface de 2048 kbit/s

O sinal nesta interface possui uma taxa nominal de 2048 kbit/s, com uma tolerância de  $\pm 50$  ppm e combina o sinal de dados com o de sincronismo de bit por meio do código *HDB-3*, uma variante do código *AMI*, que visa eliminar algumas características indesejáveis deste último.

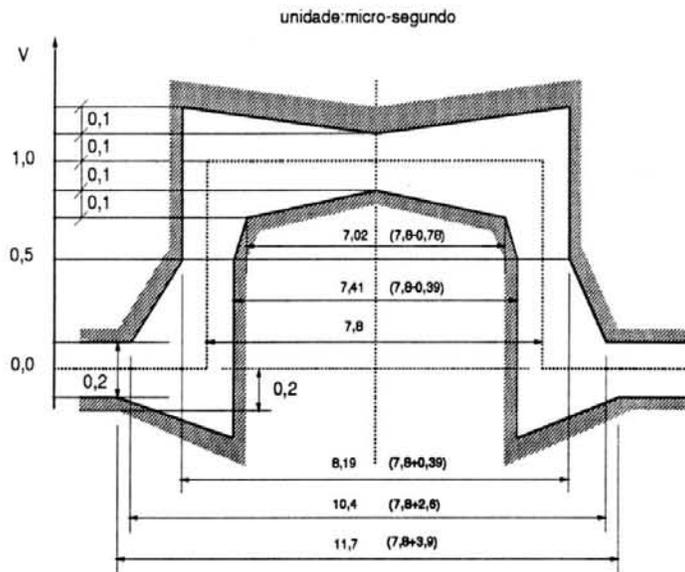


Figura 3.8 - Máscara para pulso duplo.

A técnica de codificação *AMI* é histórica e comercialmente importante, pois tem sido utilizada desde a introdução dos sistemas de transmissão *T1* pela AT&T em 1962. Este código, ainda largamente utilizado, satisfaz a maior parte dos requisitos importantes para sistemas de transmissão digitais, como por exemplo ausência de componente *DC* significativa na linha e baixo nível de energia em frequências baixas, além de evitar a propagação de erros.

Na codificação *AMI* um dígito binário *zero* é representado por ausência de sinal na linha, sendo que para um dígito binário *um*, são transmitidos impulsos *RZ*, cuja polaridade sempre é invertida em relação ao impulso anterior. A característica indesejável do código *AMI*, é a sua incapacidade de representar longas seqüências de dígitos binários *zero*, sem suprimir a informação de temporização do sinal transmitido. Com isto, torna-se problemática a recuperação do sinal de relógio associado

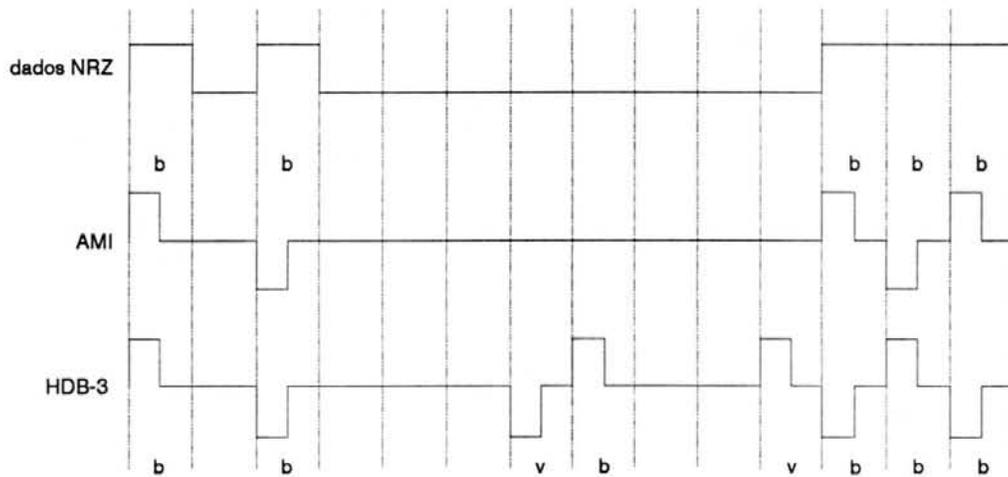


Figura 3.9 - Códigos *HDB-3* e *AMI* na interface *G.703*.

à informação, na extremidade receptora do enlace. Para contornar este problema, geralmente o codificador *AMI* é utilizado conjuntamente com um circuito capaz de decorrelacionar os dados que serão transmitidos, quebrando as longas cadeias de dígitos binários *zero*, substituindo-as por seqüências de dados onde têm-se tanto dígitos binários *zero*, como dígitos binários *um*. Este circuito é denominado *Scrambler*, ou embaralhador. Os dados são então embaralhados, codificados e transmitidos ao equipamento receptor. Este por sua vez, decodifica o sinal *AMI*, resultando em dados binários embaralhados. Um circuito desembaralhador, ou *descrambler*, recupera o formato original da informação.

No código *HDB-3*, a condição de marca (dígito binário *um*) é codificada segundo o código *AMI*, sendo que para quatro espaços (dígitos binários *zero*) consecutivos, esta seqüência é substituída pela seqüência *000V* ou *B00V*, onde *V* é o pulso de violação e *B* é um bit de recheio, ou *stuffing*. A escolha entre uma ou outra seqüência, é feita de forma que o número de pulsos *B* entre pulsos *V* consecutivos seja ímpar ou seja, pulsos *V* sucessivos são de polaridade alternada, a fim de que não seja introduzida componente *DC* no sinal. A figura 3.9, ilustra um esquema das codificações *HDB-3* e *AMI*, para uma seqüência genérica de bits.

As características físicas e elétricas da interface de 2048 kbit/s, são as seguintes:

- Forma dos pulsos  $B$  e  $V$  (nominalmente retangular): conforme a figura 3.10.
- Pares trançados em cada direção: um par simétrico.
- Impedância de teste: 120 ohm, resistiva.
- Tensão de pico nominal dos pulsos  $B$  e  $V$ : 3,0 volt.
- Tensão máxima na condição de espaço:  $0,0 \pm 0,3$  volt.
- Largura nominal de um intervalo de bit de 2048 kbit/s: 244 ns.
- Relação entre a amplitude positiva e negativa dos pulsos, no centro do intervalo: 0,95 a 1,05.

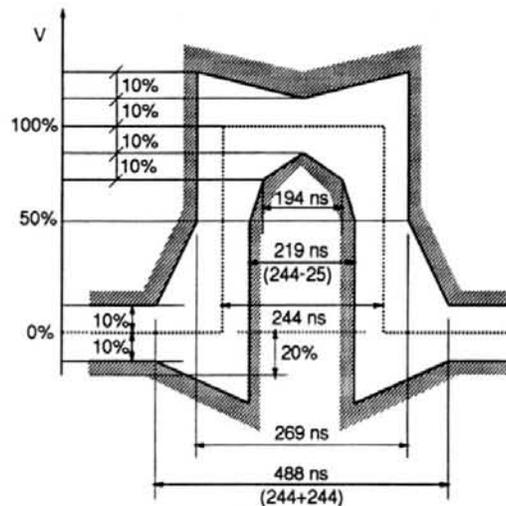


Figura 3.10 - Máscara para pulsos na interface de 2048 kbit/s.

- Relação entre a largura de pulsos positivos e negativos no centro do intervalo: 0,95 a 1,05.
- *Jitter* pico a pico máximo na porta de saída: De acordo com a recomendação *G.823*.

### 3.3 Rec. *G.704* - Estruturas de Quadros Síncronos dos Níveis 1 e 2

Esta recomendação do *ITU-T*, fornece características funcionais de interfaces associadas com equipamentos multiplex digitais, no que concerne à estrutura básica de quadros, incluindo detalhes de comprimento de quadro, padrão de alinhamento de quadro, procedimentos para *CRC-4* (verificação de redundância cíclica a 4 bits) e outras informações básicas.

A estrutura básica de quadro para os multiplexadores da hierarquia *T1*, está descrita no parágrafo 2.1 da recomendação *G.704*, sendo que para a hierarquia *E1*, as especificações estão contidas no parágrafo 2.3.

A seguir, serão detalhadas as especificações para a hierarquia *E1*, no que diz respeito à estrutura de quadros.

#### 3.3.1 Estrutura Básica de Quadro *E1*

No sistema *E1*, são transmitidos 32 canais, sendo que destes, dois canais são reservados à transmissão de sinalização de serviço e sincronismo do sistema. O canal zero (primeiros oito bits) transporta informações de sincronismo e controle do enlace, disponibilizando ainda uma banda de 20 kbit/s, utilizada para comunicação entre equipamentos. O canal 16 transporta sinalização telefônica, sendo utilizado quando o *MUX* opera com sinais de voz digitalizada (*PCM* telefônico). Desta forma, têm-se 30 canais telefônicos digitais; no caso do *MUX* estar operando somente com sinais de dados propriamente ditos, o canal 16 é utilizado como um canal a mais, fornecendo portanto, 31 canais de 64 kbit/s.

A estrutura de quadro básico para o padrão *E1*, determina um quadro composto de 256 bits numerados de 0 a 255, com taxa de repetição de 8000 Hz. Assim, a duração de um quadro básico *E1* é de 125  $\mu$ s. Os primeiros oito bits em

cada quadro básico (canal zero), são alocados de acordo com o que mostra a figura 3.11, com as seguintes observações:

- OBS 1:  $S_i$  são bits reservados para uso internacional, sendo que um uso específico para estes é na implementação do *CRC-4*. Outros usos possíveis para estes bits, estão sendo estudados pelo *ITU-T*. Caso os bits  $S_i$  não sejam utilizados para nenhuma função, estes devem ser mantidos em nível lógico *um*, caso a informação atravessasse fronteiras internacionais. Se a informação ficar restrita a tráfego dentro de um mesmo país, a recomendação permite que eles possam ser utilizados genericamente.
- OBS 2: Este bit é fixado em nível lógico *um*, a fim de que sejam evitadas simulações do padrão de alinhamento de quadro.
- OBS 3:  $A$  é um bit para indicação de alarme remoto. No caso de operação normal, este bit deve estar em nível lógico *zero*, caso contrário, em nível lógico *um*.
- OBS 4:  $Sa_4$  a  $Sa_8$ , são bits de reserva que podem ser utilizados em aplicações específicas ponto a ponto, tais como equipamentos *transcoders*, de acordo com a recomendação *G.761*. Para o bit  $Sa_4$ , uma outra utilização pode ser na forma de um canal de 4 kbit/s empregado em enlaces de dados baseados em mensagens para operação, manutenção e monitoração de desempenho. Este canal origina-se no ponto onde o quadro é montado e termina no ponto onde o quadro é desfeito. Esta aplicação também é objeto de estudos posteriores por parte do *ITU-T*. Os bits  $Sa_5$  a  $Sa_7$  podem ser utilizados dentro de um mesmo país, caso não estejam sendo empregados nas aplicações de *transcoder* (Rec. *G.761*). Os bits  $Sa_4$  a  $Sa_8$ , caso não sejam utilizados, devem estar em nível lógico *um*, sempre que a informação atravessar fronteiras internacionais. Utilizados integralmente, na forma de um canal de 20 kbit/s, os bits  $Sa_4$  a  $Sa_8$  possibilitam uma sub-rede para troca de mensagens de gerenciamento e controle entre os nós da rede propriamente dita.

| quadros alternados                                   | bit # |                                |      |      |     |     |     |     |
|--|-------|--------------------------------|------|------|-----|-----|-----|-----|
|  | b0    | b1                             | b2   | b3   | b4  | b5  | b6  | b7  |
| quadro que contém sinal de alinhamento de quadro     | si    | 0                              | 0    | 1    | 1   | 0   | 1   | 1   |
|  | obs1  | sinal de alinhamento de quadro |      |      |     |     |     |     |
| quadro que não contém sinal de alinhamento de quadro | si    | 1                              | A    | sa4  | sa5 | sa6 | sa7 | sa8 |
|  | obs1  | obs2                           | obs3 | obs4 |     |     |     |     |

Figura 3.11 - Estrutura básica de quadro *E1*.

### 3.3.2 Estrutura de Multiquadro *CRC-4*

Conforme pode ser visto na figura 3.11, alternadamente são transmitidos quadros que contém e quadros que não contém o padrão de alinhamento de quadro. Após os primeiros oito bits pertencentes ao canal zero terem sido transmitidos, segue-se uma seqüência de 248 bits (31 vezes 8 bits), correspondente aos canais 1 a 31. O primeiro quadro correspondente ao canal zero a ser transmitido, é o que contém o padrão de alinhamento de quadro. A partir daí, os quadros correspondentes ao canal zero irão se alternando, ora um quadro que não contenha o padrão de alinhamento de quadro, ora um quadro que contenha o padrão de alinhamento de quadro. Este esquema pode ser visualizado na figura 3.12, que mostra a estrutura de multiquadro *CRC-4*, na qual os bits  $S_i$  são empregados para implementação do procedimento de verificação de redundância cíclica, *CRC-4*.

O *CRC-4* pode ser utilizado onde haja necessidade de uma proteção adicional contra a simulação do padrão de alinhamento de quadro, ou ainda para monitoração da ocorrência de erros no enlace.

A recomendação *G.704*, em seu parágrafo 2.3.3.1, estabelece que os equipamentos que incorporarem o procedimento *CRC-4*, devem estar especificados de tal forma que seja possível desativar este recurso através de, por exemplo, *straps*. Esta orientação baseia-se na premissa de que deve ser possível interconectar equipa-

|   | sub-multiquadro | quadro | bits 0 a 7 do quadro básico                         |   |     |     |     |     |     |     |
|---|-----------------|--------|---|---|-----|-----|-----|-----|-----|-----|
|   |                 |        | m<br>u<br>l<br>t<br>i<br>q<br>u<br>a<br>d<br>r<br>o | 1 | 0   | c1  | 0   | 0   | 1   | 1   |
| 1 | 0               | 1      |   |   | A   | sa4 | sa5 | sa6 | sa7 | sa8 |
| 2 | c2              | 0      |   |   | 0   | 1   | 1   | 0   | 1   | 1   |
| 3 | 0               | 1      |   |   | A   | sa4 | sa5 | sa6 | sa7 | sa8 |
| 4 | c3              | 0      |   |   | 0   | 1   | 1   | 0   | 1   | 1   |
| 5 | 1               | 1      |   |   | A   | sa4 | sa5 | sa6 | sa7 | sa8 |
| 6 | c4              | 0      |   |   | 0   | 1   | 1   | 0   | 1   | 1   |
| 7 | 0               | 1      |   |   | A   | sa4 | sa5 | sa6 | sa7 | sa8 |
| 2 | 8               | c1     |   | 0 | 0   | 1   | 1   | 0   | 1   | 1   |
|   | 9               | 1      |   | 1 | A   | sa4 | sa5 | sa6 | sa7 | sa8 |
|   | 10              | c2     |   | 0 | 0   | 1   | 1   | 0   | 1   | 1   |
|   | 11              | 1      |   | 1 | A   | sa4 | sa5 | sa6 | sa7 | sa8 |
|   | 12              | c3     |   | 0 | 0   | 1   | 1   | 0   | 1   | 1   |
|   | 13              | E      |   | 1 | A   | sa4 | sa5 | sa6 | sa7 | sa8 |
|   | 14              | c4     |   | 0 | 0   | 1   | 1   | 0   | 1   | 1   |
|   | 15              | E      | 1   | A | sa4 | sa5 | sa6 | sa7 | sa8 |     |

Figura 3.12 - Estrutura de multiquadro.

mentos que não possuam o procedimento *CRC-4* incorporado ao seu projeto, com outros que disponibilizem este recurso.

Nos quadros que contêm o padrão de alinhamento de quadro, o bit 0 de cada um destes é utilizado para transmitir os bits de *CRC-4*, designados por *C1*, *C2*, *C3* e *C4*.

O quadro de número 0, na figura 3.12, transmite seu primeiro bit, *C1*; em seguida, transmite o padrão de alinhamento de quadro ( sete bits restantes), após o que seguem-se os bits de número 8 até 255. Quando esta seqüência termina, tem-se o envio completo do primeiro quadro (quadro zero). Em seguida, será transmitido o quadro de número 1 (quadro que não contém padrão de alinhamento de quadro), sempre com o segundo bit fixado em nível lógico *um*, a fim de evitar simulações do padrão de alinhamento de quadro, seguindo-se os bits de alarme e de reserva. Após

os primeiros oito bits do quadro um terem sido transmitidos, seguem-se os bits de número 8 até 255, deste quadro. O próximo quadro a ser transmitido (quadro dois), novamente conterà um bit de *CRC-4* (*C2*), e os bits do padrão de alinhamento de quadro, e assim por diante, com *C3* e *C4* nos quadros 4 e 6 respectivamente. Quando tiverem sido transmitidos oito quadros, completa-se um total de  $8 \times 256 = 2048$  bits transmitidos. Este conjunto de 2048 bits, o qual é transmitido em um milissegundo, é denominado de sub-multiquadro, sendo que o conjunto de dois sub-multiquadros, é denominado multiquadro *CRC-4*.

Nos quadros que não contêm o padrão de alinhamento de quadro (quadros 1, 3, 5, 7, 9 e 11), o primeiro bit é empregado para transmissão do padrão de alinhamento de multiquadro, nominalmente a seqüência "001011". Os dois últimos bits (bits *E* do multiquadro na figura 3.12), nos quadros que não contêm o sinal de alinhamento de quadro (quadros 13 e 15), são os bits de indicação de erro *CRC-4*.

Os bits *E* devem ser utilizados para indicar a recepção com erro dos bits de *CRC-4* dos sub-multiquadros 1 e 2. Desta forma, o bit *E* do quadro básico 13 será resetado quando for detectado erro na recepção do sub-multiquadro 1, ao passo que se houver erro no sub-multiquadro 2, o bit *E* do quadro básico 15 deverá ser resetado.

Os bits *E* devem ser sempre levados em consideração, mesmo para o caso em que o sub-multiquadro que os contenha possa estar sendo recebido com erro, uma vez que é muito pequena a probabilidade de que os próprios bits *E* estejam errados.

Pode ocorrer que equipamentos que não utilizem os bits *E*, venham a operar com equipamentos providos deste recurso; para este caso, visando a compatibilidade do sistema, deve ser prevista uma maneira de setar estes bits, no equipamento que os empregue.

### 3.3.3 Implementação da Verificação Cíclica de Redundância *CRC-4*

Uma palavra de *CRC-4* localizada em um sub-multiquadro qualquer, é o restante de uma operação de multiplicação da representação polinomial do sub-multiquadro anterior por  $x^4$ , seguido de divisão pelo gerador polinomial  $x^4 + x + 1$ .

O processo de codificação para geração dos bits de *CRC-4* que serão transmitidos ao multiplexador remoto, é o seguinte:

- Inicialmente, os bits de *CRC-4* do sub-multiquadro são resetados;
- todo o sub-multiquadro é submetido às operações de multiplicação e divisão pelo polinômio gerador;
- o restante (bits *C1*, *C2*, *C3* e *C4*) é então armazenado até que seja inserido no próximo sub-multiquadro.

Os bits de *CRC-4* assim gerados não afetam o resultado do processo de multiplicação/divisão para o próximo sub-multiquadro, uma vez que na codificação, suas posições no sub-multiquadro são sempre substituídas por *zeros* binários.

O processo de decodificação, no equipamento remoto, dá-se da seguinte maneira:

- Os bits de *CRC-4* de um sub-multiquadro recebido, são inicialmente extraídos de suas posições e armazenados em memória, sendo suas posições preenchidas com *zeros* binários;
- o sub-multiquadro é então submetido ao processo de multiplicação e divisão pelo polinômio gerador, resultando em quatro bits que são armazenados e subseqüentemente comparados bit a bit com os bits de *CRC-4* do próximo sub-multiquadro;

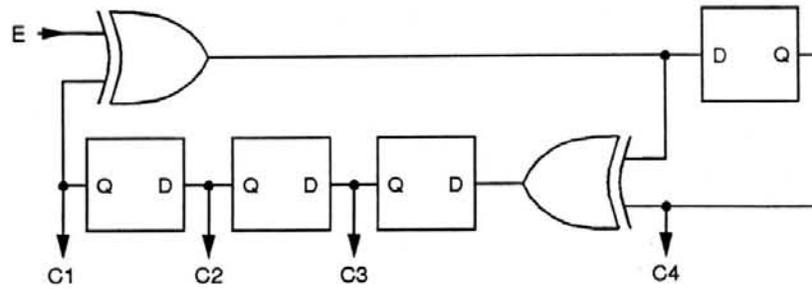


Figura 3.13 - Circuito para obtenção dos coeficientes de  $CRC-4$ .

- se o resultado obtido na decodificação corresponder exatamente aos quatro bits  $CRC-4$  recebidos no próximo sub-multiquadro, assume-se que o sub-multiquadro verificado está livre de erros.

O apêndice *A3* da recomendação *G.704* apresenta um circuito para implementação do  $CRC-4$ . Este circuito pode ser visto na figura 3.13.

A implementação do circuito para obtenção dos coeficientes de  $CRC-4$  é realizada através do emprego de flip-flops tipo *D* e portas OR-exclusivo.

Com os bits  $C1$ ,  $C2$ ,  $C3$  e  $C4$  resetados, o sub-multiquadro ( $N$ ) é introduzido serialmente no circuito, através do terminal de entrada  $E$ . Quando o último bit do sub-multiquadro (bit 256 do quadro básico 7, ou bit 256 do quadro básico 15) tiver sido introduzido no circuito, os bits de  $CRC-4$  estarão disponíveis nas saídas  $C1$  a  $C4$ , sendo  $C1$  o bit mais significativo. Assim, estes bits serão transmitidos no próximo sub-multiquadro ( $N+1$ ). Depois de cada sub-multiquadro, as saídas dos flip-flops deverão ser resetadas.

### 3.4 Rec. *G.706* - Procedimentos de Controle em Estruturas Básicas de Quadro

Esta recomendação diz respeito aos procedimentos de controle aplicados nas estruturas de quadro básico  $T1$  e  $E1$ , definidas na recomendação *G.704*. Neste

trabalho, serão analisados estes procedimentos, especificamente para o multiplexador *E1*, o qual deverá dispor de funções que permitam implementar estes procedimentos de controle do enlace, entre os equipamentos local e remoto.

### 3.4.1 Perda de Sincronismo de Quadro

Na recomendação *G.704*, foi definido o padrão de alinhamento de quadro, responsável pelo sincronismo de quadros entre um multiplexador local e outro remoto. Assim, por ocasião da entrada em operação de um enlace *E1*, os multiplexadores deverão inicialmente sincronizar a transmissão e recepção dos quadros de dados. Durante a operação normal, com os multiplexadores já sincronizados, o padrão de alinhamento de quadro é verificado periodicamente na recepção, a fim de confirmar a manutenção de sincronismo de quadro entre os equipamentos; isto é fundamental para a correta desmontagem dos quadros e posterior transmissão dos octetos para seus respectivos tributários.

A confirmação do sincronismo de quadro é realizada através do reconhecimento da seqüência de bits que constitui o padrão de alinhamento de quadro, durante o intervalo de tempo a este reservado no quadro básico, ou seja os bits  $b_1$  a  $b_7$  do quadro que contém o padrão de alinhamento de quadro. Assim, se a operação síncrona entre os equipamentos local e remoto for mantida, o intervalo de tempo estabelecido localmente, para a verificação do padrão de alinhamento de quadro, coincidirá temporalmente com a recepção deste padrão, proveniente do equipamento remoto.

A perda do sincronismo de quadro portanto, é caracterizada pela não-confirmação da seqüência de bits do padrão de alinhamento de quadro, no intervalo de tempo a ele atribuído pelo equipamento local. Contudo, para que um estado de perda de sincronismo de quadro possa ser assumido com uma margem de incerteza muito pequena, é necessário que ocorra a não-confirmação na detecção do padrão de

alinhamento de quadro, por três vezes consecutivas. Opcionalmente, a recomendação *G.706* estabelece que o bit  $b_1$  do canal de serviço, no quadro que não contém o padrão de alinhamento de quadro, pode vir a ser utilizado para auxiliar na detecção do estado de perda de sincronismo. Assim, a condição de perda de alinhamento de quadro será assumida se este bit for detectado com erro (nível lógico *zero*), durante o processo de verificação do padrão de alinhamento de quadro.

Uma vez que tenha sido assumida a perda de sincronismo de quadro, o equipamento receptor deve gerar alarme de indicação de manutenção, indicar alarme ao equipamento transmissor e gerar alarme aos canais tributários, de acordo com o que especifica a recomendação *G.736*, no que diz respeito aos procedimentos a serem adotados pelo equipamento, em caso de falhas. Estes procedimentos serão abordados na seção "*Rec. G.736 - Características de um MUX de 2048 kbit/s*". Adicionalmente, as funções de controle do equipamento deverão fazer com que o multiplexador passe do estado de operação normal, para um estado de busca de sincronismo de quadro. Este estado será analisado na sub-seção "*Estratégia para Alinhamento de Quadro*".

Na figura 3.14, pode ser visto um diagrama de fluxo para o procedimento de verificação periódica do padrão de alinhamento de quadro, a partir do estado de operação normal.

Uma outra condição que caracteriza a perda de sincronismo de quadro, é a incapacidade do multiplexador em obter o sincronismo de multiquadro (sincronismo de *CRC*), conforme especifica o parágrafo 4.2 da recomendação *G.706*, ou ainda devido a excessiva ocorrência de blocos de mensagem *CRC* recebidos com erro, conforme os limites estipulados no parágrafo 4.3.2 da recomendação *G.706*. Estas duas condições serão analisadas na seção 3.5, referente às características dos equipamentos multiplexadores de 2048 kbit/s.

Sempre que for detectada a perda de sincronismo de quadro, o multiplexador deve iniciar um processo que visa a recuperação deste sincronismo, de forma

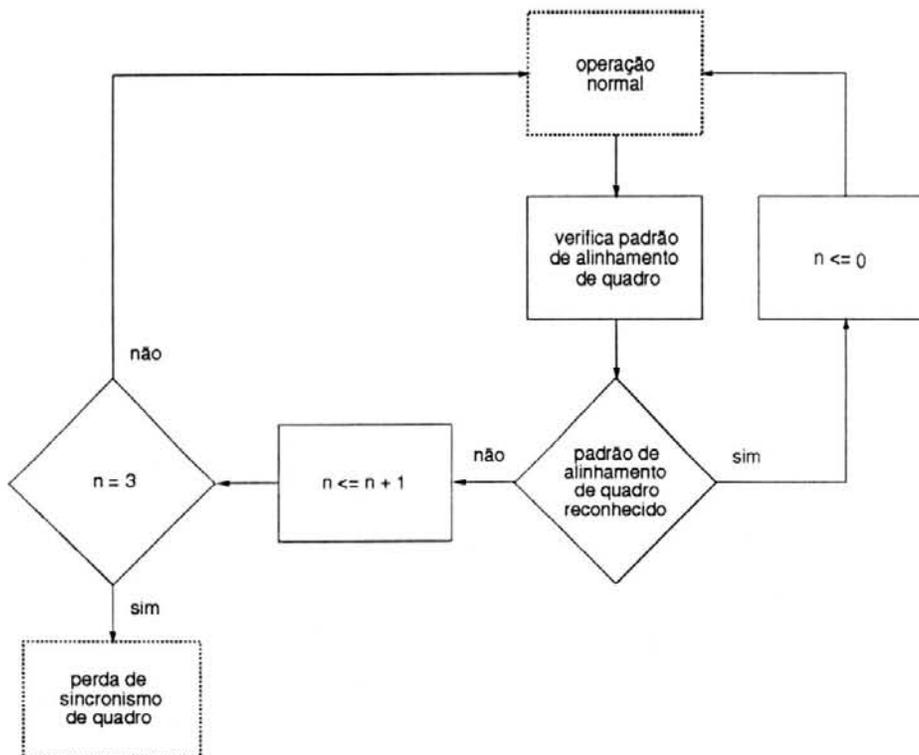


Figura 3.14 - Procedimento para declaração do estado de perda de sincronismo de quadro.

que os equipamentos, local e remoto possam ter seus quadros alinhados. A seguir, será analisada a estratégia para alinhamento de quadros, entre dois equipamentos multiplexadores, em um enlace *E1*.

### 3.4.2 Estratégia para Alinhamento de Quadro

O alinhamento de quadro caracteriza o sincronismo entre a transmissão e a recepção dos quadros de dados entre os equipamentos multiplexadores. Assim, a recomendação *G.706* especifica um procedimento a ser adotado quando for necessário o estabelecimento do sincronismo (alinhamento de quadro). O estabelecimento do alinhamento de quadro é necessário por ocasião da ativação do enlace *E1* (*start-up* do sistema) ou durante sua operação, quando determinadas condições de falha se fizerem presentes.

A estratégia para alinhamento de quadro consiste em verificar determinadas condições, na seguinte seqüência:

- monitorar o sinal de recepção e reconhecer o padrão de alinhamento de quadro ou seja, a seqüência "0011011"; isto feito, assume-se que esta seqüência está localizada entre os bits  $b_1$  a  $b_7$  de um suposto quadro básico, que contém o padrão de alinhamento de quadro;
- feita a suposição descrita no item anterior, o próximo quadro deverá ser um que não contenha o padrão de alinhamento de quadro, logo o bit  $b_1$  deste quadro deve ser testado e reconhecido como sendo um nível lógico *um*;
- por fim, no quadro seguinte ao quadro mencionado no item anterior, novamente devem ser monitorados os bits  $b_1$  a  $b_7$ , a fim de reconhecer o padrão de alinhamento de quadro.

Se as condições acima forem satisfeitas em seqüência, assume-se que estabeleceu-se sincronismo ou alinhamento de quadro, devendo esta condição ser sinalizada ao módulo de controle do equipamento.

No caso de não confirmação de uma ou mais das condições descritas, o processo de alinhamento de quadro deve recomeçar a partir do primeiro item. A figura 3.15 mostra o procedimento para recuperação de sincronismo de quadro, na forma de um diagrama de fluxo.

### 3.4.3 Alinhamento de Multiquadro

Uma vez que tenha sido obtido alinhamento de quadro, deve ser verificado o alinhamento de multiquadro. Para tanto, o padrão de alinhamento de multiquadro deverá ser detectado dentro de um intervalo de tempo não maior que 8 ms, contado

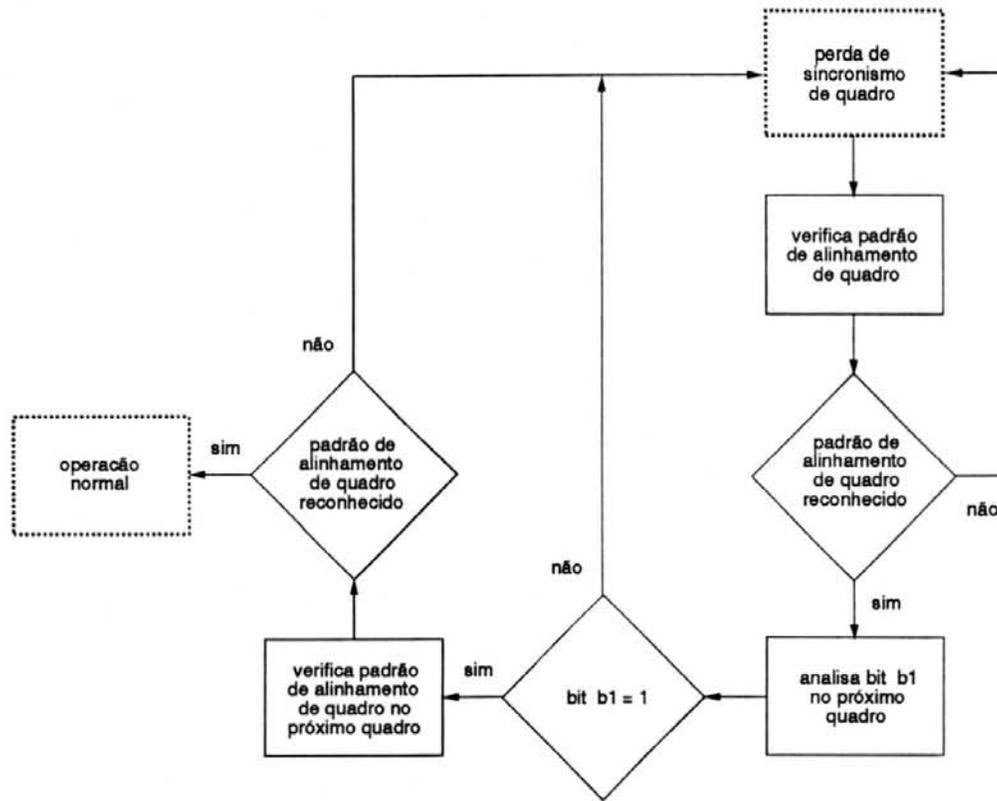


Figura 3.15 - Procedimento para obtenção de sincronismo de quadro.

a partir do momento em que seja sinalizado o alinhamento de quadro. O padrão de alinhamento de multiquadro deve ser procurado no primeiro bit dos quadros que não contêm o padrão de alinhamento de quadro.

Se não for possível obter alinhamento de multiquadro dentro do intervalo de tempo previsto (8 ms), deve ser assumido que o alinhamento de quadro anteriormente detectado, foi devido a sinais espúrios, simulantes do padrão de alinhamento de quadro verdadeiro. Desta forma, deverá ser iniciado um novo procedimento para alinhamento de quadro.

### 3.5 Rec. G.736 - Características de um MUX de 2048 kbit/s

Esta recomendação, especifica as características funcionais de um equipamento multiplex da hierarquia *E1*, capaz de multiplexar 31 canais tributários de 64 kbit/s, em um agregado de 2048 kbit/s. Especial ênfase é dada nesta recomendação, com relação à detecção e sinalização de falhas.

Entre as especificações apresentadas, estão aquelas relacionadas à estrutura de quadros, interfaces, procedimentos para *CRC* e alinhamento de quadro e especificações relativas a *jitter* nos sinais, sendo a seguir detalhadas algumas destas especificações.

#### 3.5.1 Taxa de Bit

A taxa de bit nominal especificada por esta recomendação é de 2048 kbit/s, com uma tolerância de +/- 50 ppm.

#### 3.5.2 Sinal de Relógio

São previstos três casos a partir dos quais deve ser possível derivar o sinal de relógio, a fim de sincronizar os circuitos do equipamento. Estes são:

- relógio derivado do sinal recebido de 2048 kbit/s;
- relógio proveniente de uma fonte externa de 2048 kHz;
- relógio gerado em uma fonte interna (oscilador local).

A possibilidade de obter-se o sinal de relógio derivado do sinal de 64 kbit/s de um dos canais tributários, está em estudo por parte do *ITU-T*.

| parte do equipamento | tipo de falha                                       | procedimentos                           |                              |                                     |   |
|----------------------|---|---|------------------------------|-------------------------------------|---|
|                      |   | gerar alarme de indicação de manutenção | indicar alarme ao eq. remoto | gerar alarme aos canais tributários | gerar alarme aos slots relevantes do agregado |
| mux e demux          | falta de energia                                    | sim                                     | se possível                  | se possível                         | se possível                                   |
| mux                  | perda do sinal entrante de 64 kbit/s                | sim                                     |                              |                                     | sim   |
| demux                | perda do sinal entrante de 2048 kbit/s              | sim                                     | sim                          | sim                                 |   |
|                      | perda do sinal de alinhamento de quadro             | sim                                     | sim                          | sim                                 |   |
|                      | alta taxa de erro no sinal de alinhamento de quadro | sim                                     | sim                          | sim                                 |   |
|                      | alarme proveniente do equipamento remoto            | sim                                     |                              |                                     |   |

Figura 3.16 - Condições de falha e ações.

O equipamento poderá prover uma saída através da qual o sinal de relógio localmente gerado, possa ser utilizado por outros equipamentos que necessitem estar em sincronismo com o multiplexador. No entanto, esta opção fica a cargo das regulamentações impostas pelas concessionárias em cada país.

### 3.5.3 Condições de Falha e Ações

A recomendação para as condições de falha e ações, estabelece os procedimentos que devem ser desencadeados pelo equipamento, caso ocorram determinadas falhas no funcionamento do sistema.

A figura 3.16, mostra as condições de falha e as respectivas ações a serem desencadeadas.

As falhas especificadas são falta de energia, perda do sinal tributário entrante de 64 kbit/s, perda do padrão de alinhamento de quadro de 2048 kbit/s,

elevado grau de detecção do bit de erro através do monitoramento do padrão de alinhamento de quadro e indicação de alarme proveniente do equipamento remoto. Cada uma destas condições de falha está relacionada com uma determinada parte do equipamento, quando mais não sejam, a parte de multiplexação (transmissão), ou a parte de demultiplexação (recepção), com exceção da falta de energia, que relaciona-se às duas partes do equipamento.

A condição de falha caracterizada por uma alta taxa de erro na recepção do padrão de alinhamento de quadro deve gerar um alarme de indicação de manutenção, tal que:

- Uma taxa de erro maior ou igual a  $10^{-3}$ , monitorada por um período contínuo de 4 segundos, é considerada alta taxa de erro na recepção do padrão de alinhamento de quadro, devendo a indicação de alarme ser ativada.
- A indicação de alarme ativada pela condição de falha especificada no item anterior, somente poderá ser desativada, se a taxa de erro cair abaixo de  $10^{-4}$ , assim permanecendo por um período mínimo contínuo de 4 segundos.

Do ponto de vista da implementação, a indicação de alarme mencionada nos itens anteriores, pode ser feita através de um dispositivo luminoso, instalado no painel do equipamento. As demais ações conseqüentes especificadas na figura 3.16, são implementadas através de manipulação de bits, nos quadros do agregado, ou nos octetos dos tributários. Assim, a indicação de alarme ao equipamento remoto deve ser realizada setando-se o bit "A" do canal de serviço a ser transmitido.

A geração de alarme aos canais tributários e aos *slots* relevantes do agregado, é realizada através do *Sinal de Indicação de Alarme*. O conteúdo binário equivalente a este sinal consiste de uma seqüência contínua de bits *um*, sendo que a estratégia para detectar sua presença deve ser tal que, com uma probabilidade

muito grande, possa ser detectado mesmo na presença de taxas de erro na faixa de  $10^{-3}$ . Adicionalmente, deve ser possível distinguir um sinal de indicação de alarme de um sinal em que todos os bits estejam setados.

O sinal de indicação de alarme é aplicável tanto aos tributários como ao agregado. Para os tributários, este sinal é aplicado nos casos de falta de energia para o equipamento, perda do sinal entrante de 2048 kbit/s, perda de sincronismo de quadro e alta taxa de erro na detecção do sinal de alinhamento de quadro. Para o agregado, o sinal de indicação de alarme é aplicado no caso de falta de energia e perda do sinal entrante de 64 kbit/s em uma ou mais entradas. Neste caso, o sinal de indicação de alarme deve ser aplicado ao agregado, durante o *time-slot* dos canais tributários em falha.

Quando o sinal de indicação de alarme for detectado no sinal entrante de 2048 kbit/s, as indicações de alarme de manutenção geradas pela detecção de perda de alinhamento de quadro e taxa de erro acima de  $10^{-3}$  devem ser inibidas, sendo que as demais ações consequentes relativas a estas duas falhas devem ser mantidas.

### **3.6 Rec. G.823 - O Controle do *Jitter* e *Wander* nas Redes Digitais Baseadas na Hierarquia de 2048 kbit/s**

#### **3.6.1 *Jitter* e *Wander***

Quando é necessário transferir informação de um ponto para outro, frequentemente há que se fornecer um sinal de temporização, que cadencie a operação de transferência da informação. Este é o caso dos sistemas digitais síncronos.

Em sistemas onde a distância é pequena (em termos gerais, até cerca de alguns metros), é comum fornecer-se o sinal de informação e temporização separa-

damente, utilizando-se meios de transmissão exclusivos para cada um destes sinais. Nestes casos, a informação é transferida na forma de sinais binários *NRZ*, sendo que a temporização é fornecida na forma de um sinal de relógio de frequência constante.

Em sistemas de comunicação digital entretanto, torna-se anti-econômica a utilização de suportes exclusivos à transmissão de informação e relógio, devido à distância relativamente longa entre o equipamento transmissor e o equipamento receptor. Desta forma, na extremidade receptora do enlace, o sinal de relógio é regenerado aproveitando-se as transições interníveis do sinal transmitido, como referências temporais.

Entretanto, o sinal *NRZ* apresenta características intrinsecamente indesejáveis para transmissão a longas distâncias: forte componente *DC* e ausência de transições interníveis em longas seqüências de *zeros* ou *uns* binários, tornam impraticável a transmissão direta de sinais *NRZ* à longas distâncias.

Para contornar os problemas inerentes à transmissão direta de sinais *NRZ* a longas distâncias, utiliza-se a *Codificação* [BEL 91] de ambos os sinais (informação e temporização) em um único sinal, cujo formato é tal que proporciona nível *DC* desprezível no meio de transmissão e densidade de transições suficiente para o processo de regeneração da temporização associada à informação, no extremo receptor do enlace.

Entretanto, o sinal codificado sofre diferentes tipos de degradação ao ser propagado ao longo do meio de transmissão. Assim, a característica passa-baixas do meio de transmissão, bem como o efeito de diferentes tipos de ruído interferente e variações do comprimento dos meios de transmissão e da velocidade de propagação do sinal nos meios de transmissão, provocam instabilidades de temporização conhecidas como *Jitter* e *Wander*.

O *jitter* é definido como sendo variações a curto termo (acima de 10 Hz) dos instantes significativos de um sinal digital e sua ocorrência está associada

sistematicamente aos efeitos da característica passa-baixas do meio de transmissão e, de forma aleatória, à ação do ruído interferente sobre o sinal codificado.

O *Wander* por sua vez, é definido como sendo variações lentas (abaixo de 10 Hz) dos instantes significativos do sinal, sendo que sua ocorrência dá-se basicamente sobre enlaces muito longos, devido aos efeitos da variação de temperatura sobre os meios de transmissão, que provoca contração ou dilatação dos mesmos, bem como alteração de suas constantes de propagação.

Os instantes significativos de um sinal digital são os instantes no tempo em que ocorrem as transições entre níveis lógicos. Para um sinal digital *NRZ*, os instantes significativos são as transições entre níveis lógicos *um* e *zero*.

Assim, para um sinal degradado por *jitter* ou *wander*, uma transição internível esperada para um instante  $t$ , poderá ocorrer em qualquer instante localizado entre  $t - \frac{i}{2}$  e  $t + \frac{i}{2}$ , onde  $i$  é a margem de incerteza determinada pela intensidade da degradação sofrida pelo sinal.

A incerteza na localização dos instantes significativos é prejudicial ao processo de regeneração do sinal de relógio associado ao sinal de informação.

A figura 3.17 mostra o efeito do *jitter*, sobre um sinal do tipo código de linha, em que existem três níveis lógicos ( $+V$ ,  $0$  e  $-V$ ). As áreas hachuradas na figura são as margens de incerteza das transições interníveis do sinal afetado por *jitter*. Uma transição inter-nível pode estar localizada em qualquer ponto da área de incerteza.

O *jitter* em excesso aumenta a taxa líquida de erro em sistemas digitais de comunicação, pela elevação da interferência intersimbólica [LEE 90] que causa redução da imunidade a ruído. Em sistemas telefônicos digitais, o sinal de voz reconstruído resultará distorcido se os quadros *PCM* estiverem excessivamente afetados por *jitter* [BEL 91]. Em enlaces de longo alcance, pode ocorrer também o fenômeno de acumulação de *jitter*, descrito em [BYR 63].

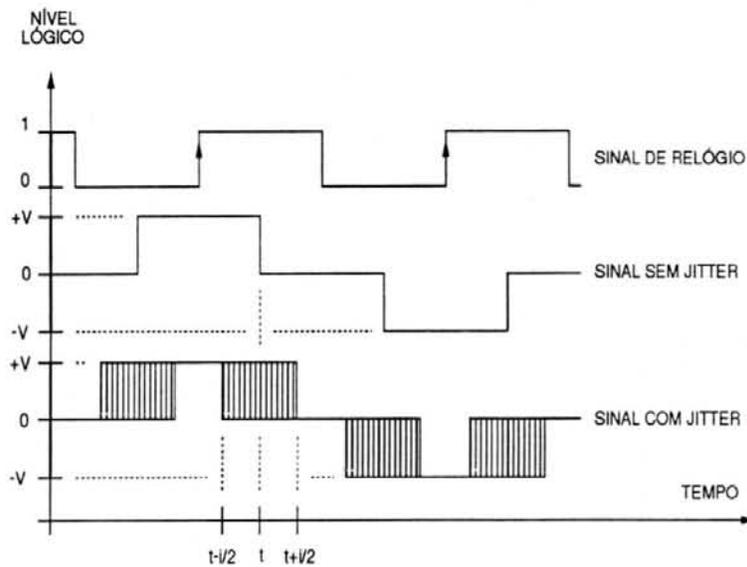


Figura 3.17 - Efeito do *jitter* em um sinal digital.

Assim, para funcionamento adequado dos sistemas digitais de comunicação, o nível de *jitter* deve ser mantido dentro de determinados limites. Estes limites estão definidos na recomendação *G.823*, do *ITU-T*. Esta recomendação estabelece as diretrizes para o controle do *jitter* e *wander*, dentro das redes digitais baseadas na hierarquia de 2048 kbit/s.

### 3.6.2 *Jitter* e *Wander* nas Interfaces de 2048 kbit/s e 64 kbit/s do Multiplexador *E1*

A recomendação *G.823* estabelece que as portas de entrada de equipamentos que operam a partir das interfaces hierárquicas, devem suportar níveis de *jitter* e *wander* típicos para redes digitais de comunicação.

Para o caso do multiplexador *E1*, é necessário especificar os circuitos de entrada para as interfaces de 64 kbit/s (tributários) e 2048 kbit/s (agregado), de tal sorte que estes sejam capazes de operar a partir de sinais contaminados com níveis de *jitter* tais quais os mostrados nos gráficos da figura 3.18.

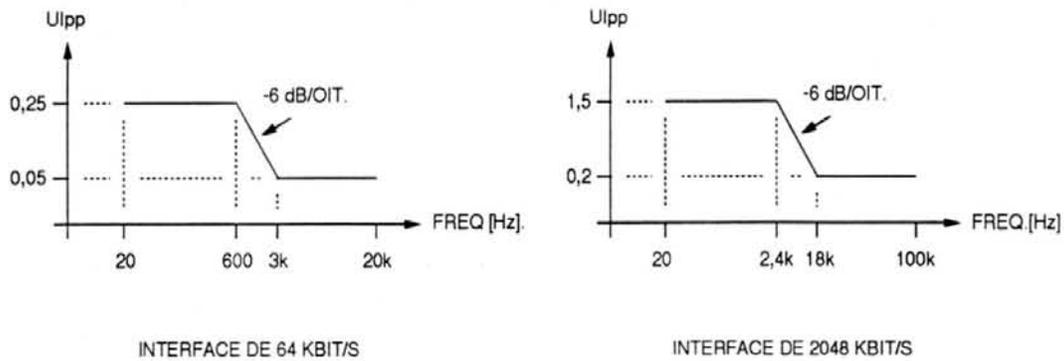


Figura 3.18 - Níveis de *jitter* nas interfaces.

Os níveis de *jitter* e *wander* são expressos através do número de *Intervalos Unitários* ( $UI_{pp}$ , eixo vertical) em função da rapidez com que os instantes significativos do sinal são deslocados de suas posições ideais no tempo (frequência de *jitter*, eixo horizontal).

O intervalo unitário é o período de bit do sinal de informação na interface considerada. Assim, para o sinal de 64 kbit/s, o intervalo unitário é 15,625  $\mu$ s, sendo que para o sinal de 2048 kbit/s, o intervalo unitário é 488,281 ns.

Para o caso de determinados equipamentos de transmissão, entre estes os multiplexadores, a recomendação *G.823* desobriga a acomodação de instabilidades temporais do tipo *wander*. Desta forma, no caso do multiplexador *E1*, para a interface de 64 kbit/s, o espectro de frequências de *jitter* se estende de 20 Hz a 20 kHz, sendo que para a interface de 2048 kbit/s, esta faixa vai de 20 Hz a 100 kHz.

Assim, de acordo com a figura 3.18, a máxima variação pico a pico dos instantes significativos do sinal na interface de 64 kbit/s é de 0,25 UI na faixa de frequências de *jitter* que se estende de 20 Hz a 600 Hz até 0,05 UI na faixa que se estende de 3,0 kHz até 20 kHz. Para um *jitter* igual a 0,25  $UI_{pp}$ , os instantes significativos do sinal entrante na interface deslocam-se de suas posições ideais no tempo a uma taxa de 36 ns por intervalo de bit. Esta taxa de deslocamento mantém-se constante na faixa de frequências de *jitter* compreendida entre 600 Hz e 3,0 kHz, embora o *jitter* diminua de acordo com a pendente de -6 dB/oitava, dentro desta

faixa. Na faixa de frequências de *jitter* de 3,0 kHz até 20 kHz, o *jitter* mantém-se constante, ainda que a taxa de deslocamento das transições aumente gradativamente, até cerca de 244 ns por intervalo de bit, na frequência de 20 kHz.

Para a interface de 2048 kbit/s, o máximo *jitter* esperado para um sinal entrante é de  $1,5 UI_{pp}$  na faixa de 20 Hz até 2,4 kHz, sendo que o mínimo é igual a  $0,2 UI_{pp}$ , na faixa que se estende de 18 kHz até 100 kHz. Para o valor máximo de *jitter* de  $1,5 UI_{pp}$  com frequência de 2,4 kHz, os instantes significativos do sinal entrante na interface deslocam-se de suas posições ideais no tempo a uma taxa de 0,86 ns por intervalo de bit. Este valor de taxa de deslocamento mantém-se constante ao longo da faixa de -6 db/oitava de variação da frequência de *jitter*, até a frequência de 18 kHz. A partir daí, a taxa de deslocamento de fase do sinal cresce com a frequência de *jitter*, até o valor máximo de 4,8 ns por intervalo de bit, na frequência de *jitter* de 100 kHz.

Na fase de especificação dos circuitos de entrada das interfaces de tributários e de agregado, a simulação destes circuitos deve portanto considerar os níveis de *jitter* e as taxas de deslocamento de fase dos sinais (frequências de *jitter*) descritas nesta análise, a fim de que correspondam à recomendação *G.823*.

Com a análise desta recomendação *G.823*, encerra-se neste trabalho o estudo das recomendações pertinentes ao multiplexador *E1*. No capítulo seguinte, serão analisados os aspectos funcionais deste multiplexador.

## 4 ASPECTOS FUNCIONAIS DO MULTIPLEXADOR *E1*

A partir do estudo das recomendações que definem a estrutura do multiplexador *E1*, serão apresentados neste capítulo os aspectos funcionais essenciais do processo de multiplexação desenvolvido por este equipamento.

### 4.1 Implementação de um Enlace de Multiplexação *E1*

A figura 4.1 mostra o diagrama de blocos de um enlace de multiplexação *E1*. Através de suas 31 entradas (interfaces de recepção *G.703* representadas pelos quadrinhos escuros na figura), o equipamento multiplexador (designado por MUX na figura) recebe o sinal proveniente de cada um dos canais tributários locais (designados na figura pelos quadrinhos numerados de 1 a 31). Os canais tributários locais estão localizados geograficamente próximos ao equipamento multiplexador, sendo a este ligados por meio de linha física tipo par trançado, conforme especificado na recomendação *G.703* do *ITU-T*.

Um canal tributário local transfere informação a uma das entradas do multiplexador através de interfaces co-direcionais, na forma de octetos *G.703* a 64 kbit/s. Ao receber os octetos *G.703* provenientes dos tributários locais, o multiplexador os decodifica, obtendo octetos de dados binários *NRZ*, próprios para serem multiplexados. Uma vez multiplexados, estes octetos de dados dão origem aos quadros *E1* (agregado de 2048 kbit/s).

Os quadros *E1*, conforme a recomendação *G.704* do *ITU-T*, são formados por 32 octetos, totalizando 256 bits dos quais os primeiros oito bits constituem-se no denominado canal de serviço ou canal zero e os 248 restantes correspondem aos

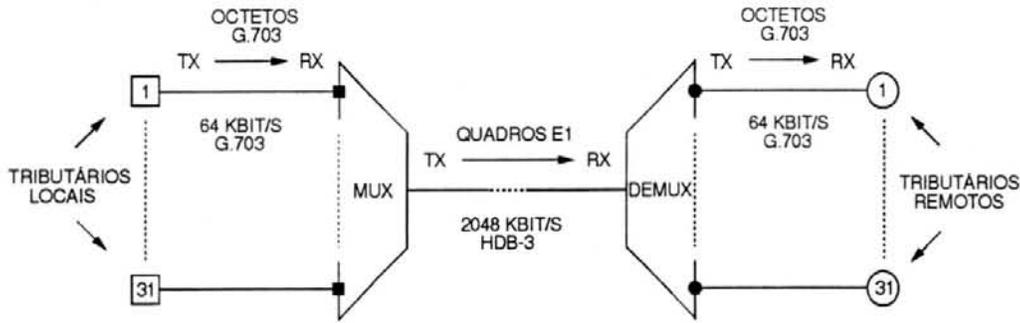


Figura 4.1 - Enlace de multiplexação *E1*.

canais tributários de número 1 a 31. O conteúdo dos quadros *E1* é então codificado segundo o código *HDB-3*, sendo assim bit a bit transmitido via par trançado, ao equipamento demultiplexador (designado por *DEMUX* na figura), localizado na extremidade remota do enlace.

A transmissão dos quadros *E1* do equipamento multiplexador para o equipamento demultiplexador, dá-se através de interfaces *G.703* a 2048 kbit/s. Ao receber o sinal *HDB-3* proveniente do multiplexador, o demultiplexador decodifica e separa os octetos que deverão ser transmitidos a cada um dos tributários remotos (designados na figura pelos círculos numerados de 1 a 31). Após codificar os octetos segundo o código *G.703*, estes são então transmitidos via linha de 64 kbit/s, aos tributários remotos. Os círculos escuros na figura representam as interfaces de transmissão *G.703* do demultiplexador. A interface entre linha e tributários (locais e remotos), também é a *G.703*.

O multiplexador *E1*, ao atribuir a cada um dos canais tributários uma posição correspondente a um octeto em seu quadro básico, está na verdade disponibilizando o meio de transmissão durante um determinado intervalo de tempo, para que os dados de um canal tributário local sejam transmitidos a um canal tributário remoto.

Como no caso do multiplexador *E1* o período de um quadro básico é determinado pela soma dos períodos dos 32 octetos que o compõem, um canal tributário local efetivamente dispõe do meio de transmissão durante  $1/32$  do período

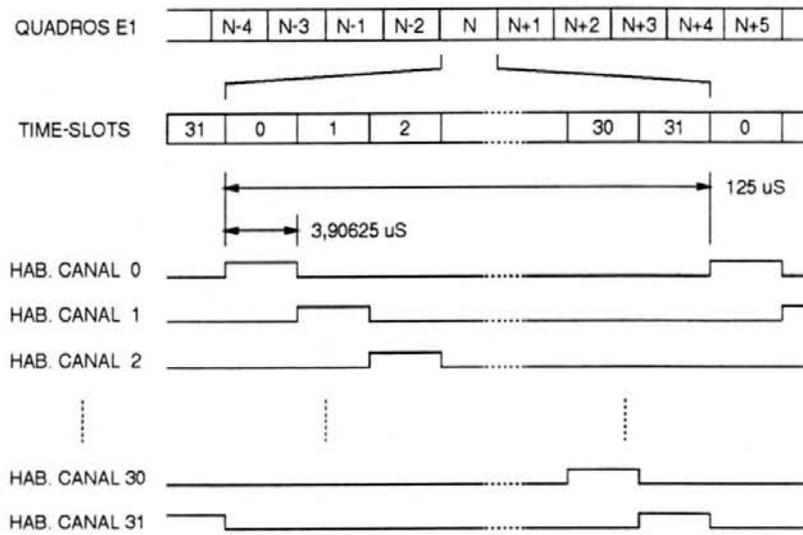


Figura 4.2 - Geração de *time-slots* do quadro básico *E1*.

de quadro, sendo os restantes 31/32 do tempo destinados à transmissão dos dados dos demais 30 canais tributários e do canal de serviço (canal zero).

Assim, em essência a multiplexação *TDM* síncrona, tal qual a realizada pelo multiplexador *E1*, consiste em dividir o tempo de utilização do meio de transmissão em 32 períodos de tempo de igual duração, durante os quais os dados dos canais tributários e do canal de serviço são transmitidos na forma de octetos ao equipamento demultiplexador.

Formalmente, estes 32 períodos de tempo são denominados de *Time-slots*. O quadro básico *E1* é portanto formado por 32 *time-slots*. A figura 4.2 mostra um diagrama de temporização em que podem ser vistos os *time-slots* de um quadro básico *E1* (quadro *N*), para cada um dos canais tributários, em um período equivalente a um quadro *E1* (125 µs).

A transmissão dos dados (alocação de *time-slots*) é controlada por sinais de habilitação (*HAB. CANAL "n"*). Um nível lógico alto em um destes sinais significa a disponibilidade do meio de transmissão para o canal "*n*" em questão. Assim, no início do quadro *N* mostrado na figura, a informação referente ao canal 0 é transmitida durante os primeiros 3,90625 µs, seguindo-se a transmissão da informação do canal 1 nos subseqüentes 3,90625 µs e assim por diante. Ao final do *time-slot* reser-

vado ao canal 31 ou seja, após  $125 \mu\text{s}$  decorridos do início do quadro  $N$ , novamente o canal 0 transmite informação, agora relativa ao próximo quadro (quadro  $N+1$ ).

O processo de multiplexação ora descrito deixa transparecer determinadas condições que devem ser satisfeitas, a fim de que seja possível a multiplexação. Estas condições serão a seguir analisadas com maior detalhe.

## 4.2 Pré-requisitos para a Multiplexação

A transmissão da informação entre os tributários locais e o multiplexador, assim como entre o demultiplexador e os tributários remotos, dá-se a  $64 \text{ kbit/s}$  através de octetos *G.703* de duração igual a  $125 \mu\text{s}$ . Por sua vez, o multiplexador realiza a transmissão de 32 octetos ao demultiplexador, na forma de um quadro *E1*, cujo período é igualmente de  $125 \mu\text{s}$ . Assim, o tempo efetivamente disponível para utilização do meio de transmissão por parte de cada um dos canais tributários é de  $3,90625 \mu\text{s}$ . Uma vez que cada octeto é composto de oito bits, estes devem portanto ser transmitidos bit a bit durante o *time-slot* de  $3,90625 \mu\text{s}$ , resultando no intervalo de bit  $488,28125 \text{ ns}$ , que corresponde precisamente à taxa de transmissão de agregado de  $2048 \text{ kbit/s}$ .

Assim, para que seja possível a transmissão de um octeto durante o *time-slot* de  $3,90625 \mu\text{s}$  alocado para seu respectivo canal, é necessário que os oito bits deste octeto sejam simultaneamente disponibilizados ao multiplexador ou seja, é necessário concentrá-los previamente ao longo de  $125 \mu\text{s}$  e então transmití-los durante  $3,90625 \mu\text{s}$ , segundo a taxa de agregado. Este processo pode ser acompanhado com auxílio das figuras 4.3 e 4.4.

A figura 4.3 mostra o diagrama lógico de um registrador de deslocamento. Neste dispositivo, uma transição ascendente do sinal de relógio, faz com que um bit presente à entrada *D* seja transferido à saída *Q7*, sendo que um bit presente a uma

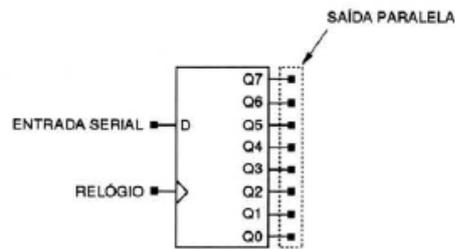


Figura 4.3 - Registrador de deslocamento.

saída  $Q_N$  é transferido à saída  $Q_{N-1}$ , com exceção do bit presente à saída  $Q_0$ , que é perdido. Desta forma, a cada 8 ciclos de relógio, dados seriais à entrada  $D$  são convertidos em dados paralelos e disponibilizados na forma de octetos na saída do registrador de deslocamento.

Na figura 4.4, podem ser vistos os octetos a 64 kbit/s, bem como o relógio de 64 kHz que os cadencia. Estes dois sinais são levados respectivamente à entrada serial e de relógio de um registrador de deslocamento. Assim, a cada transição ascendente do relógio de 64 kHz, as oito saídas ( $Q_0$  a  $Q_7$ ) do registrador de deslocamento são atualizadas. Estas saídas, bem como os *time-slots* de habilitação do canal também são mostrados na figura 4.4.

Quando a saída do registrador de deslocamento apresentar um octeto completo (bit  $BN$  correspondendo à saída  $Q_N$ ), este estará pronto para ser codificado e transmitido serialmente ao demultiplexador, no intervalo de tempo correspondente ao seu *time-slot* de 3,90625  $\mu$ s. Pode ser notado na figura, que o período do *time-slot* corresponde a 1/4 do período de bit a 64 kbit/s. No destaque da figura 4.4, é mostrado ampliadamente o *time-slot*, assim como oito ciclos de relógio a 2048 kHz, responsáveis pela serialização do octeto presente à saída do registrador de deslocamento. Na serialização, a cada transição ascendente do relógio de 2048 kHz, um bit presente à saída do registrador de deslocamento é transmitido, sendo  $B_0$  o primeiro e  $B_7$  o último. Dado que o *time-slot* repete-se também a intervalos de 125

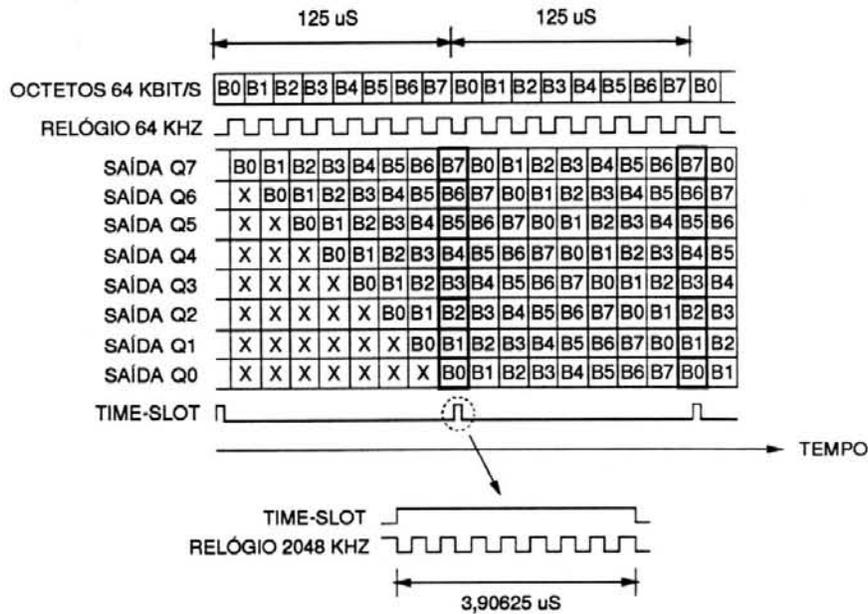


Figura 4.4 - Paralelização de octetos.

$\mu\text{s}$ , o octeto seguinte já estará disponível à saída do registrador de deslocamento, quando da ocorrência do próximo *time-slot*.

Assim, para que o processo de multiplexação seja viável, percebe-se que é necessário que os oito bits de cada um dos octetos provenientes de um tributário remoto qualquer, estejam disponíveis na forma paralela, precisamente no momento em que o multiplexador disponibiliza o meio de transmissão a este tributário, através da alocação de seu *time-slot*. Para que isto seja efetivo com relação não somente a um tributário mas a todos, conclui-se então que é necessária a existência de uma defasagem temporal de precisamente  $3,90625 \mu\text{s}$  entre a transmissão dos octetos de um tributário genérico de número  $n$  e seu adjacente de número  $n+1$ , já que *time-slots* adjacentes alocados pelo multiplexador apresentam este defasamento de  $3,90625 \mu\text{s}$ . Esta situação é mostrada na figura 4.5.

Entretanto, nas interfaces de recepção *G.703* de um multiplexador, os octetos provenientes dos tributários remotos apresentam relações de fase indeterminísticas entre si. A recomendação *G.703*, que define estas interfaces, não impõe qualquer regra com relação à fase do sinal emitido por um tributário. Um tributário que inicie a transmissão de informação ao multiplexador, deverá poder fazê-lo em

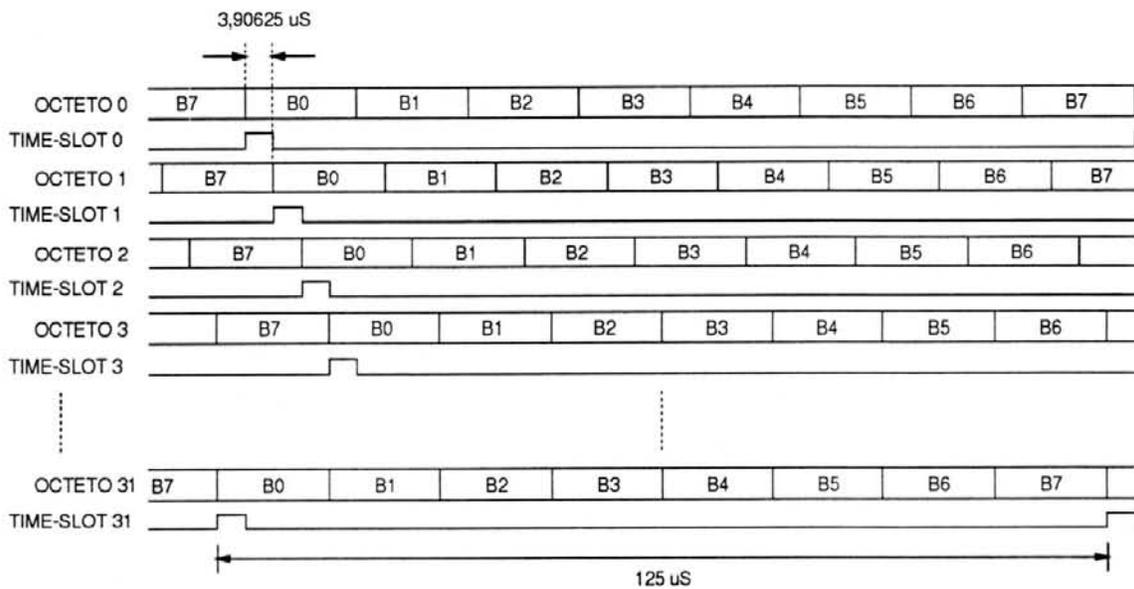


Figura 4.5 - Defasamento necessário à viabilidade da multiplexação.

um instante de tempo totalmente arbitrário em relação aos demais tributários, caracterizando assim, um desalinhamento temporal entre eles.

A relação de fases requerida para a viabilização do processo de multiplexação deve então ser obtida artificialmente, tão logo os octetos sejam recebidos pelo multiplexador. Isto constitui-se em um problema relativo à implementação do processo de multiplexação. A seguir serão apresentados aspectos relacionados a este problema, bem como uma terminologia apropriada para sua análise.

### 4.3 Terminologia para Sinais em Redes Digitais de Comunicação

Uma seqüência de bits presente em um determinado ponto de uma rede digital de comunicações, apresenta uma taxa de bit que pode ser nominalmente constante, onde as únicas variações na taxa são devidas a efeitos tais como desvio térmico dos osciladores (*drift*), ou pode ser variável no tempo, o que é comum em uma rede onde várias seqüências de bits interagem entre si, após terem passado por diferentes equipamentos e meios de transmissão.

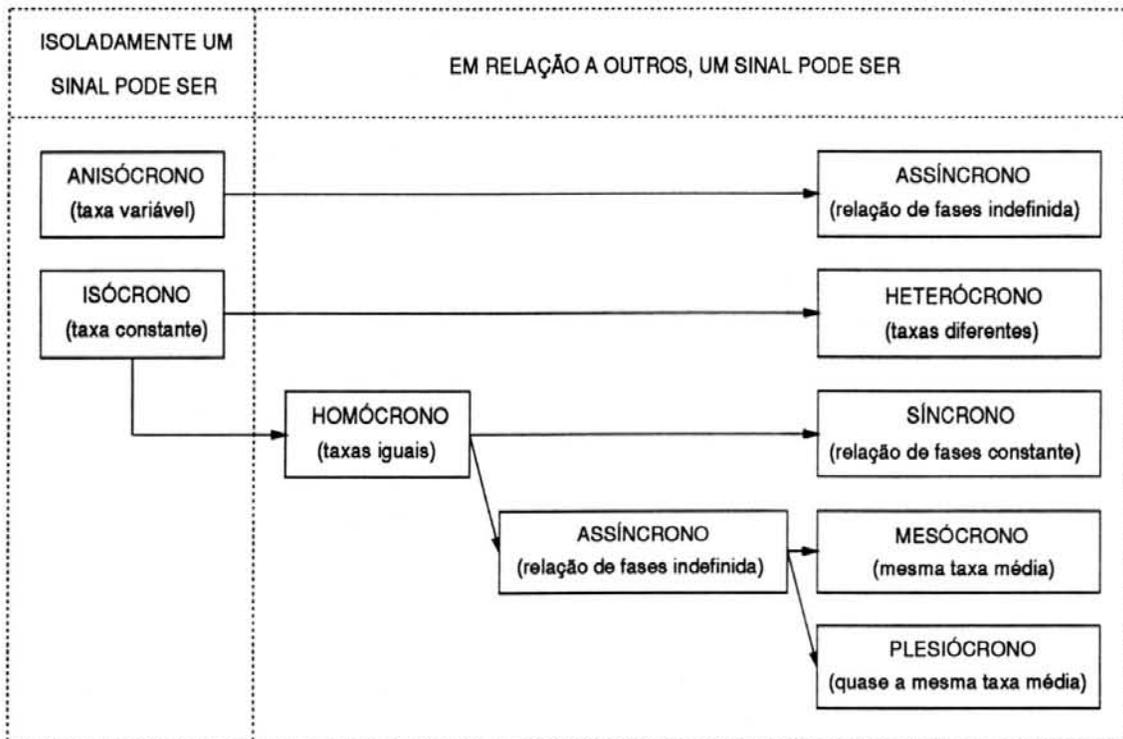


Figura 4.6 - Classificação dos sinais em uma rede de comunicação digital.

Assim, uma seqüência de bits cuja taxa é constante, é dita **Isócrona**, (de **iso**, radical grego para **igual** e **cronos**, radical grego para **tempo**). Analogamente, uma seqüência de bits cuja taxa é variável no tempo é dita **Anisócrona** (ou **não-igual**).

Quando se toma duas seqüências de bits quaisquer em uma rede digital de comunicações, pode-se estar diante de diferentes situações, as quais serão descritas a seguir. Se estas seqüências forem isócronas, apresentarem a mesma taxa média de bit e, além disso, possuírem uma relação de fases constante, então estas duas seqüências de bits são ditas **Síncronas** (de **sin**, radical grego para **juntos**). Por outro lado, se estas duas seqüências de bits não forem síncronas, diz-se então que são **Assíncronas**.

Devido aos retardos de propagação e processamento e também aos mecanismos de *jitter* e *wander* presentes na rede, dois sinais isócronos serão síncronos somente no ponto de origem, onde tiverem sido gerados a partir de um sinal de

relógio comum a ambos. Assim, freqüentemente é necessário tratar-se com sinais assíncronos, que podem apresentar-se de diferentes maneiras.

Uma seqüência de bits que seja anisócrona, será assíncrona em relação a qualquer outra seqüência de bits, uma vez que para um sinal de freqüência variável, sua fase não pode ser comparada com a de qualquer outro sinal. Se duas seqüências de bits possuem a mesma taxa média de bit, não existindo porém uma relação de fases constante entre elas, diz-se então que são **Mesócronas** (de **meso**, radical grego para **médio**). Desta forma, pode ser dito que duas seqüências de bits que sejam síncronas no ponto de origem, serão mesócronas após terem sido transmitidas ao longo da rede.

Duas seqüências de bits que tenham nominalmente a mesma taxa de bit, mas não exatamente a mesma, são ditas serem **Plesiócronas** (de **plesio**, radical grego para **perto**). Na prática, seqüências de bits plesiócronas são geradas a partir de osciladores de relógio independentes, cujos sinais produzidos possuam a mesma freqüência nominal mas que, em realidade difiram levemente, uma vez que não existem osciladores que possam produzir sinais síncronos, sem que seja necessária alguma técnica especial para tal (malha de travamento de fase entre os osciladores).

Por fim, duas seqüências de bits que sejam isócronas, mas que possuam taxas de bit nominalmente diferentes, são chamadas **Heterócronas** (de **hetero**, radical grego para **diferentes**), sendo que se estas seqüências apresentarem taxas nominalmente iguais, serão ditas **homócronas** (de **homo**, radical grego para **mesmo**). Na figura 4.6, pode ser visto um diagrama que oferece uma visão de contexto da classificação de sinais ora apresentada.

## 4.4 Compatibilização Entre MUX e Tributários

A multiplexação *TDM*, tal como a efetuada pelo multiplexador *E1*, é também denominada multiplexação síncrona, numa referência ao fato de que os dados dos tributários após terem sido agrupados na forma de um quadro básico *E1*, apresentam precisamente a mesma taxa de bit. Além disto, possuem ainda uma relação de fases constante entre si ou seja, conhecendo-se a posição de um octeto no quadro básico, pode-se determinar com precisão a posição dos demais.

O processo de multiplexação *TDM* conforme visto na seção 4.2, consiste em acumular os octetos de tributário que chegam ao multiplexador com taxa de 64 kbit/s e então transmití-los durante um *time-slot* de 3,90625  $\mu$ s, utilizando para tanto, oito ciclos de relógio de 2048 kHz, um para cada bit do octeto a ser transmitido (taxa de agregado de 2048 kbit/s).

Entretanto, sabe-se que os *time-slots* repetem-se a intervalos de 125  $\mu$ s. Assim, efetivamente a taxa média de um octeto inserido no quadro básico *E1* é:

$$B = (2048 \text{ kbit/s} \times 3,90625 \mu\text{s}) / 125 \mu\text{s} = 64 \text{ kbit/s}$$

Portanto, cada tributário "vê" um meio de transmissão capaz de transportar informações a 64 kbit/s.

Assim, ainda que a taxa instantânea de transmissão dos dados de um tributário seja igual a 2048 kbit/s no quadro básico *E1*, em média esta taxa é de 64 kbit/s. Esta particularidade por si só já seria suficiente para afirmar-se que os sinais provenientes de cada um dos 31 tributários são mesócronos em relação ao agregado. Entretanto, existe ainda a degradação por *jitter* e *wander* sofrida pelos sinais dos tributários, que provoca indeterminação de fases entre a temporização dos tributários e do multiplexador.

Em relação aos demais tributários, um tributário qualquer também é mesócrono, uma vez que todos apresentam taxas iguais, porém com relações de fase arbitrárias.

Para que sejam mesócronos é necessária ainda a implementação da *Função de Sincronismo de Rede*, responsável pela sincronização dos osciladores a partir dos quais as seqüências de bits são geradas. Basicamente, a função de sincronismo de rede é implementada através dos esquemas de *Sincronização Mútua* e *Sincronização Mestre-escravo* [PAN 72].

Por outro lado, não raras as vezes, somente é exequível que os tributários apresentem a mesma taxa nominal de bit ou seja, que sejam plesiócronos. Neste caso, ao invés de sincronismo de rede, a temporização do sistema é suportada por osciladores de relógio independentes e altamente estáveis, instalados nas extremidades de transmissão dos tributários e no equipamento de multiplexação. Por outro lado, uma rede operando sobre temporização mesócrona, tornar-se-á plesiócrona caso a Função de Sincronismo de Rede venha a falhar.

De qualquer forma, torna-se claro que é necessário compatibilizar o funcionamento do multiplexador, um dispositivo essencialmente síncrono, com sinais que são plesiócronos ou, na melhor das hipóteses, mesócronos em relação ao multiplexador [KAI 95]. Este será o assunto abordado na seção seguinte.

## 4.5 Armazenamento Elástico

Na seção anterior ficou evidente a natureza mesócrona ou plesiócrona dos dados dos tributários em relação ao multiplexador e a necessidade de torná-los síncronos, para que possam assim constituir o quadro básico *E1*. Para o caso em que os dados dos tributários sejam mesócronos, estes podem ser convertidos em síncronos, submetendo-os a um retardo controlável de forma que, por meio do

ajuste individual da quantidade de retardo imposto a cada tributário, estes passem a apresentar uma relação de fases constante. Esta relação de fases assim obtida, é aquela necessária à viabilização do processo de multiplexação, tal como descrito na seção 4.2.

No caso em que os dados dos tributários sejam plesiócronicos, não é possível fazer com que apresentem a relação de fases requerida indefinidamente. Neste caso, apenas por períodos de tempo determinados é possível controlar a relação de fases. A seguir, será descrito como compatibilizar sinais mesócronicos e plesiócronicos em um mesmo ambiente, a partir da técnica denominada **Armazenamento Elástico** [LEE 90].

Um modelo simples para representar um dispositivo capaz de executar a função de armazenamento elástico, pode ser visto na figura 4.7. O modelo dos comutadores como é denominado, em realidade é implementado utilizando-se memórias de acesso aleatório e lógica digital para endereçamento e controle, sendo que o arranjo apresentado na figura é apenas uma forma didática de apresentar e compreender a técnica de armazenamento elástico.

O modelo mostrado na figura, é constituído por um conjunto de células de armazenagem (designadas por *sc* na figura) capazes de armazenar um bit, cada uma. Assim, um bit pode ser escrito em uma célula através de seu terminal *WR* e lido por meio de seu terminal *RD*. Fazem parte também do dispositivo, dois comutadores: um para efetuar a operação de escrita dos bits nas células (comutador *WR*) e outro para efetuar a operação de leitura dos bits armazenados nas células (comutador *RD*). A este dispositivo é dado o nome de *Buffer Elástico*.

Os dados que chegam ao *buffer elástico* vão sendo sucessivamente escritos nas células de armazenagem através do comutador *WR*, ao passo que a leitura destes é posteriormente realizada pelo comutador *RD*, também de forma sucessiva. Uma escrita (ou leitura) de um bit somente é efetivada quando o comutador de escrita (ou leitura) passa pelo terminal *WR* (ou *RD*) de uma célula. Os dois comutadores

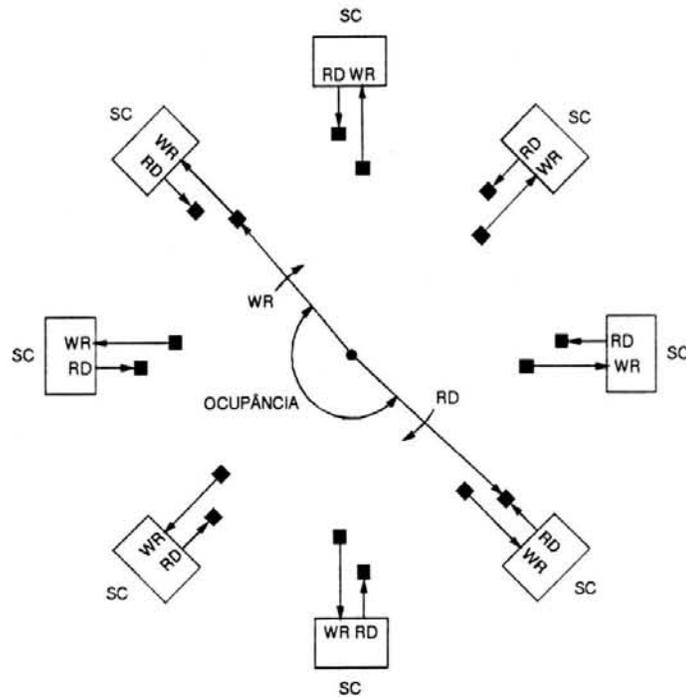


Figura 4.7 - Modelo dos comutadores para armazenamento elástico.

estão portanto a girar, a uma velocidade que representa a taxa de bit do sistema. Como as posições angulares dos comutadores são diferentes, um determinado bit escrito pelo comutador *WR* em um instante de tempo  $t_0$ , será lido pelo comutador *RD* em um instante de tempo  $t_0 + d$ , onde  $d$  é o retardo de tempo transcorrido desde a escrita até a leitura de um bit, em uma determinada célula. Este retardo pode também ser expresso pelo número de bits compreendido entre os dois comutadores, conforme mostrado na figura 4.7.

Desta forma, define-se *Ocupância* do *buffer*, como sendo o número de bits de retardo, determinado pelas posições relativas dos dois comutadores. No exemplo da figura, a ocupância é portanto de quatro bits. Deve ser ressaltado entretanto, que a ocupância expressa em função do número de bits não é uma entidade rigorosamente definida, uma vez que uma operação de escrita não ocorre necessariamente no mesmo instante em que ocorre uma operação de leitura. Assim, é mais usual expressá-la em unidades de tempo.

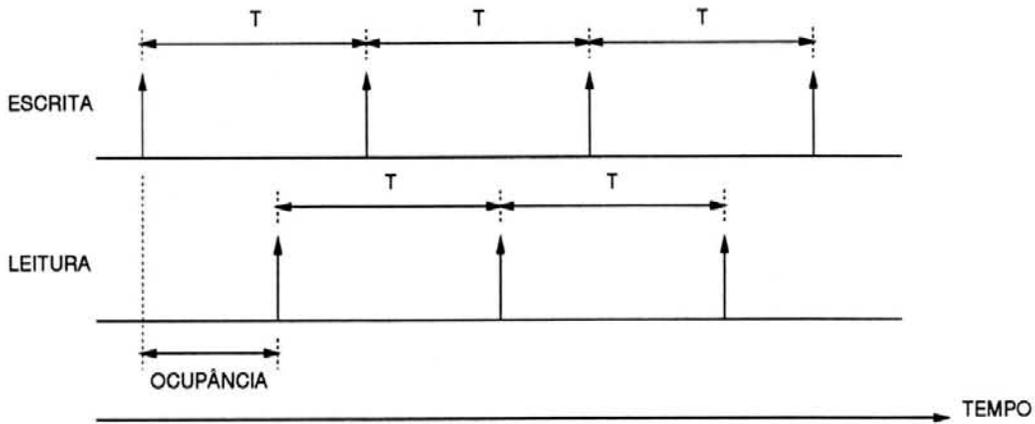


Figura 4.8 - Escrita e leitura síncronas.

Se os comutadores de escrita e leitura estiverem girando com a mesma velocidade angular (mesma taxa de bit) e ainda suas posições angulares relativas não se alterarem em instante algum (relação de fases constante), a ocupância será constante e as operações de escrita e leitura serão síncronas. Esta situação pode ser visualizada com o auxílio da figura 4.8, onde os instantes de escrita e leitura sobre uma célula *SC* genérica são representados por setas verticais. Nesta figura,  $T$  é o tempo necessário para que os comutadores realizem uma volta completa sobre o *buffer*, passando por todas as células.

Analisemos agora, a situação em que as operações de escrita e leitura sejam mesócronas. Neste caso, estas operações ocorrem à mesma taxa média, havendo porém variações instantâneas na posição de um dos comutadores em relação ao outro. Entretanto, na operação mesócrona, os respectivos relógios de escrita e leitura estão amarrados pela ação da função de sincronismo de rede, fazendo com que ao menos o valor médio das variações de fase entre escrita e leitura permaneça constante, garantindo assim um valor médio também constante para a ocupância do *buffer*, ainda que instantaneamente possa esta assumir valores indeterminados, em torno de um valor médio.

A figura 4.9 mostra uma representação dos eventos de escrita e leitura ao longo do tempo, sobre uma célula genérica do *buffer*, para a temporização mesócrona. Para análise desta figura, assumir-se-á como referência temporal, as operações de

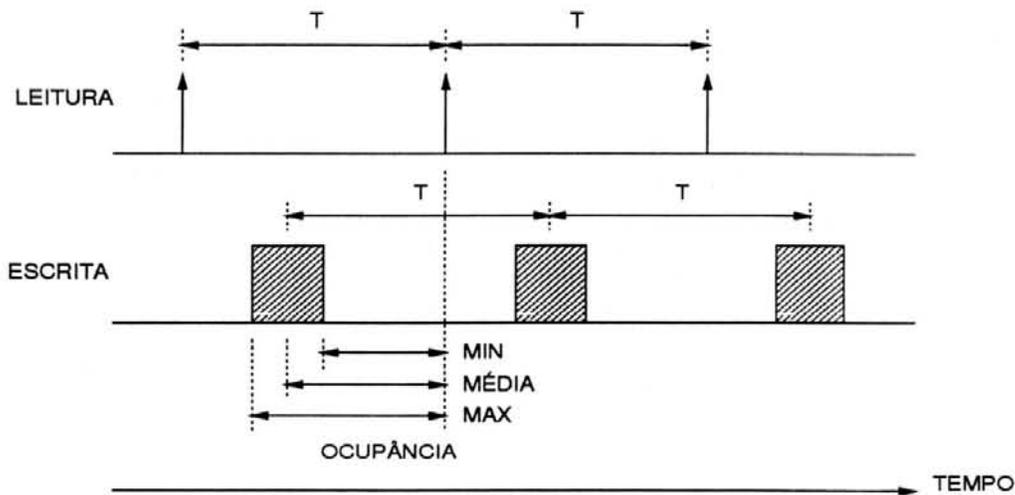


Figura 4.9 - Escrita e leitura mesócronas.

leitura. Assim, não é possível definir a exata posição no tempo, de um evento de escrita em relação a outro de leitura. As áreas hachuradas na figura determinam regiões do eixo do tempo em que os eventos de escrita podem ocorrer, determinando assim, valores máximo e mínimo para a ocupância. Estes valores máximo e mínimo são determinados pelo nível de *jitter* presente no sinal a ser armazenado no *buffer*. Quanto maior o *jitter*, maior a diferença entre os valores máximo e mínimo da ocupância.

A temporização mesócrona é garantida pela função de sincronismo de rede. Quando esta função falha ou simplesmente não existe, o sistema passa a operar com temporização plesiócrona.

Para operações de escrita e leitura plesiócronas, além da incerteza temporal determinada pelo nível de *jitter* presente no sinal de entrada do *buffer*, existe ainda uma pequena diferença numérica entre as taxas de escrita e leitura. Com isto, a ocupância ao longo do tempo não apresenta um valor médio constante; ao contrário, à medida em que o tempo passa, seu valor pode aumentar ou diminuir.

A figura 4.10 mostra uma representação dos eventos de escrita e leitura ao longo do tempo, sobre uma célula genérica do *buffer*, em que a taxa de leitura excede a de escrita por um *offset* igual a  $e$ . Assim, a ocupância determinada pela

diferença entre  $t_1$  e  $t_0$  é maior do que aquela determinada pela diferença entre  $t_3$  e  $t_2$ , que por sua vez é maior do que a ocupância determinada pela diferença entre  $t_5$  e  $t_4$ . Neste exemplo portanto, a ocupância torna-se cada vez menor ao longo do tempo, sendo que quando igualar-se a zero, terá ocorrido um *underflow* do *buffer*. Assim, o *buffer* irá sofrer um *underflow* sempre que dados sejam lidos mais rapidamente do que possam ser escritos. No caso em que o *offset*  $\epsilon$  seja negativo, os dados serão escritos mais rapidamente em relação à leitura. Desta forma, a ocupância aumenta ao longo do tempo, até que seu valor iguale-se ao período dos eventos de leitura ou seja, o intervalo de tempo  $T$  indicado na figura. Neste caso, o *buffer* terá sofrido um *overflow*.

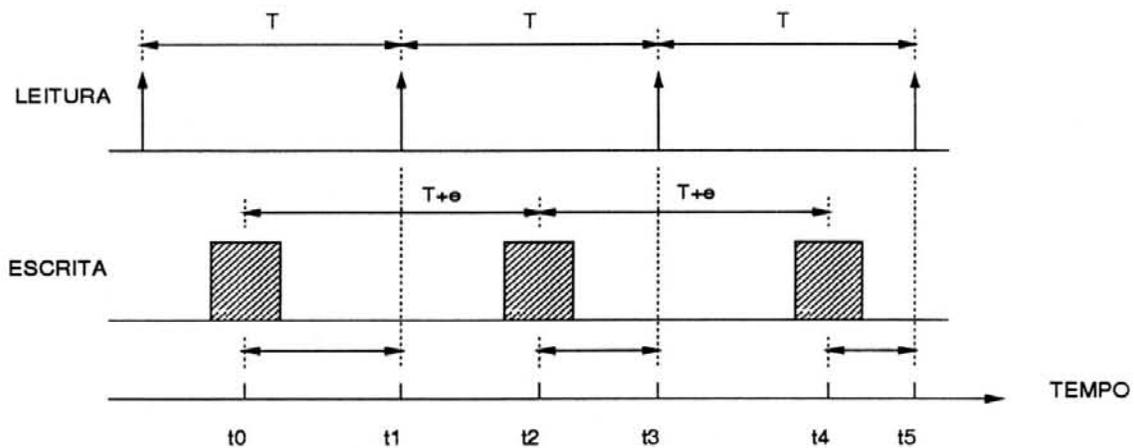


Figura 4.10 - Escrita e leitura plesiócronicas.

A ocorrência de *underflow* ou *overflow* sobre o *buffer* é denominada *Slip*. Sempre que ocorre um *slip*, o resultado é a repetição de leitura ou perda de bits de dados.

Ainda que inevitáveis para temporização plesiócrona, a ocorrência de *slips* precisa ser limitada a taxas toleráveis. Isto pode ser feito a partir da definição apropriada do número de células de armazenagem do *buffer elástico*. Para esta análise, seja o *buffer elástico* mostrado em (a) na figura 4.11 e o gráfico de sua ocupância em função do tempo, mostrado em (b) da mesma figura.

O *buffer* mostrado na figura 4.11 apresenta uma capacidade total de armazenamento igual a  $2N$  bits. Entretanto, efetivamente sua capacidade líquida equivale à metade deste valor ou seja,  $N$  bits. Isto se deve ao fato de não ser possível prever se um *slip* ocorrerá por *underflow* ou por *overflow*. Por isto, é necessário estabelecer uma condição de partida, qual seja a de inicialmente permitir que o *buffer* seja preenchido (escrita de dados) até a metade de sua capacidade total - no caso  $N$  bits - e só então iniciar a operação de leitura. Utilizando o modelo dos comutadores da figura 4.7, esta condição de partida significa que o comutador de leitura somente passará a movimentar-se, quando o ângulo formado pelos dois comutadores tornar-se igual a  $180$  graus. Desta forma, se o sinal do *offset* e for positivo, um *slip* ocorrerá por *overflow* quando o comutador de escrita tiver alcançado o de leitura, após ter vencido uma ocupância de  $N$  bits. De outra forma, se o sinal do *offset* for negativo, o comutador de leitura é que irá alcançar o de escrita (*underflow*), não sem antes vencer uma ocupância também igual a  $N$  bits.

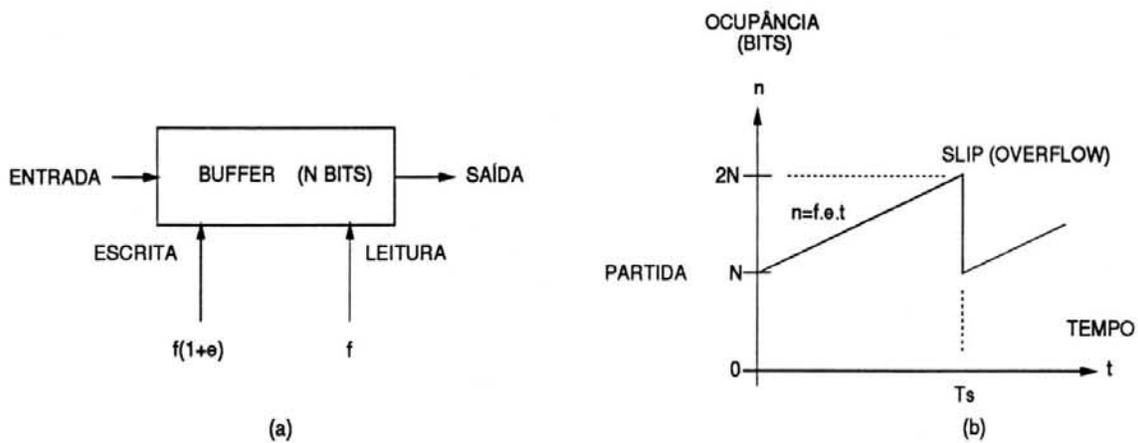


Figura 4.11 - *Buffer* elástico e gráfico de ocupância.

No sistema de armazenamento elástico da figura 4.11, o *buffer* então armazena a seqüência de dados entrante a uma taxa  $f(1+e)$ , com  $e$  positivo, sendo que a leitura ocorre a uma taxa  $f$ . Assim, o acúmulo de dados no *buffer* ocorre a uma taxa líquida de  $f.e$ , sendo que a ocupância do dispositivo cresce de acordo com a reta  $n = f.e.t$ , como mostrado em (b). Para  $n = 2N - N = N$ , terá ocorrido um *overflow* no instante de tempo  $t = T_s$ . Assim, a equação que relaciona os intervalos entre *slips* com a capacidade de armazenamento elástico é:

$$Ts = \frac{N}{f \cdot |e|} \quad (4.1)$$

Para o caso de um canal tributário *E1*, cuja taxa é de 64 kbit/s ( $f = 64$  kHz), com um *offset* de frequência  $e = 1,74 \cdot 10^{-9}$ , um *buffer elástico* com capacidade de armazenamento líquida de um octeto ( $N = 8$  bits), o intervalo mínimo entre slips ( $Ts$ ) será de 72.000 s, que corresponde a um *slip* a cada 20 horas. Considerando a hipótese de que o *offset* de frequência na extremidade geradora da seqüência de bits transmitida é positivo e, ao contrário, na extremidade receptora o *offset* é negativo, os osciladores em ambas as extremidades do enlace devem portanto apresentar uma estabilidade de  $0,87 \cdot 10^{-9}$  (osciladores a cristal de quartzo estabilizados termicamente).

O aumento da capacidade de armazenamento ( $N$ ) do *buffer elástico*, permite a utilização de osciladores menos precisos (mais simples, mais baratos). Entretanto, o retardo imposto ao sinal desde sua entrada até sua saída do *buffer* (nominalmente a *Latência* do dispositivo), deve ser mantida a menor possível. Assim, como diretriz para o projeto de dispositivos de armazenamento elástico, a capacidade de armazenamento ( $N$ ) deve ser feita tão pequena quanto possível. Esta orientação consta na recomendação *G.810* do *ITU-T* ("*Considerações Sobre Problemas de Sincronização e Temporização*"). Ainda de acordo com esta recomendação, os *slips* para as interfaces de 2048 kbit/s (agregado) devem ocorrer de tal sorte que as perdas ou repetições de dados sejam na forma de quadros inteiros, evitando a perda do alinhamento de quadro no enlace. Esta orientação evita o assim denominado *Slip Não-Controlado*, em que o multiplexador é forçado a assumir o estado de perda de sincronismo de quadro e assim, portanto, executar o procedimento de alinhamento de quadro.

Quanto aos osciladores de alta estabilidade, já há alguns anos encontram-se disponíveis comercialmente na forma de módulos, a custos bastante módicos. As

dimensões destes módulos, na ordem de  $1 \text{ cm}^3$ , permitem montagem diretamente sobre cartões de circuito impresso. Estas unidades osciladoras são capazes de fornecer sinais de relógio com estabilidade da ordem de  $10^{-9}$ , sobre uma variada gama de frequências de operação. Os módulos integram, no mesmo encapsulamento, um oscilador a cristal de quartzo, um calefator e um sistema de controle de temperatura capaz de manter o módulo aquecido a uma temperatura constante em torno de 50 a 60 graus celsius, independentemente da temperatura externa. Devido à sua construção integrada, a questão do custo destas unidades deixou de ser relevante quando comparado ao custo global do sistema em que está inserido.

## 4.6 Objetivos de *Slips*

De acordo com a recomendação *G.703*, a taxa de bit nas interfaces tributários/multiplexador é nominalmente 64 kbit/s, com uma tolerância máxima de +/- 100 ppm. Para interfaces de 2048 kbit/s, a tolerância é de +/- 50 ppm. Estes níveis de tolerância entretanto, são aplicáveis somente às interfaces operando sobre temporização mesócrona. Para redes com temporização plesiócrona, a recomendação *G.703* determina que os sinais de relógio utilizados na geração dos sinais codificados que são transmitidos de um equipamento para outro, sejam originados a partir de osciladores com estabilidade tal que garanta a existência de intervalos mínimos de *slip*. Estes intervalos mínimos são denominados de *Objetivos de Slips*.

O efeito de *slips* sobre sinais de voz digitalizada (quadros *PCM*) produz uma perturbação no formato do sinal analógico reconstruído. Entretanto, apenas um em cada 25 *slips* é audível pelo usuário, na forma de um pequeno "click", fazendo com que sinais de voz possam portanto tolerar vários *slips* por minuto [BEL 91].

Sinais (*PCM*) encriptados são mais sensíveis a *slips*, uma vez que o processo de encriptação/decriptação consiste na numeração e embaralhamento de bits na transmissão e posterior contagem e reordenação destes na recepção. Com a al-

teração na contagem dos bits provocada pela perda ou repetição de quadros *PCM*, inevitavelmente todos os *slips* serão audíveis pelo usuário, podendo mesmo o sinal reconstruído tornar-se ininteligível, em uma situação de pior caso.

Para sinais de dados propriamente ditos, a ocorrência de *slips* pode ter maior ou menor impacto. Alguns equipamentos requisitam a retransmissão completa de blocos de dados que não satisfaçam alguns requisitos de verificação de redundância cíclica. Se este for o caso, a ocorrência de um *slip* não será pior do que a ocorrência de simples erro de canal devido à ação de ruído interferente. Entretanto, para determinados tipos de protocolos de comunicação que empregam procedimentos de contagem de *bytes* para delimitação de blocos de mensagem, um *slip* provoca dessincronização do contador na recepção em relação à transmissão e a troca de informações será interrompida até que a perda de sincronismo seja reconhecida.

Por todos estes inconvenientes, a taxa de *slips* admissível para conexões fim-a-fim é de um a cada 5 horas. Como os *slips* podem ocorrer em diversos pontos dentro de uma rede, os objetivos de *slip* para interfaces individuais foi estabelecido em um *slip* a cada 20 horas [LEE 90].

Para enlaces internacionais, o *ITU-T* estabelece através da recomendação *G.811*, uma taxa de *slips* não maior do que um a cada 70 dias. Para atingir objetivos de *slip* desta ordem, a temporização do sistema deve ser suportada por osciladores atômicos [ITU 88c].

Os assuntos abordados nesta primeira parte deste trabalho, constituem-se nos elementos básicos que fundamentam a especificação dos módulos de circuito capazes de implementar as funções de multiplexação *E1*.

Na segunda parte, que iniciar-se-á no próximo capítulo, serão apresentados os módulos funcionais de circuito, bem como descrições de funcionamento e simulações para cada um deles.

## 5 MÓDULOS FUNCIONAIS PARA MULTIPLEXAÇÃO E1

Neste capítulo serão apresentadas as especificações e projeto de módulos funcionais para equipamentos *mux/demux E1*. Os módulos cuja implementação seja essencialmente analógica, serão apenas especificados.

Neste capítulo, dividiu-se a abordagem dos assuntos em três partes quais sejam, a geração de sinais de temporização para o sistema *mux/demux*, módulos para multiplexação e módulos para demultiplexação.

A edição dos diagramas esquemáticos dos circuitos que compõem os módulos, bem como as simulações dos mesmos, foram realizadas no ambiente de projeto *SOLO/CADENCE* [CAD 88] que possibilita o projeto de circuitos integrados *CMOS*, segundo a metodologia *standard cells*.

O *framework SOLO/CADENCE* disponibiliza também o simulador *SILOS* [SIL 88], utilizado na simulação dos circuitos projetados. *SILOS* é um simulador lógico que modela os transistores *CMOS* como chaves *ON/OFF*, levando em consideração o efeito do compartilhamento de cargas nas redes formadas pelas capacitâncias dos transistores, em decorrência da abertura e fechamento das chaves que modelam os dispositivos. Desta forma é possível avaliar os atrasos inerentes às células lógicas e seu efeito no desempenho global do circuito.

### 5.1 Módulo Gerador de Base de Tempo

De acordo com a recomendação *G.736*, os equipamentos *mux/demux* devem estar providos de uma maneira de selecionar uma fonte primária de sinal de relógio, a partir da qual todos os demais sinais de relógio sejam derivados.

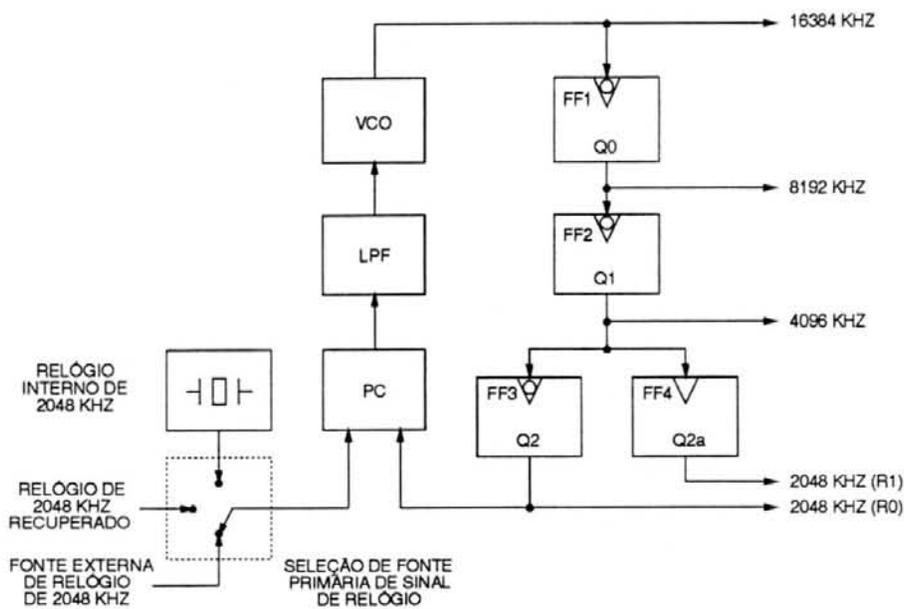


Figura 5.1 - Módulo gerador de base de tempo.

Esta fonte de sinal de relógio primária pode ser um oscilador interno de 2048 kHz, uma fonte externa de 2048 kHz (função de sincronização de rede), ou ainda o próprio sinal de relógio de 2048 kHz, recuperado a partir do sinal de linha proveniente do equipamento remoto.

Entretanto, determinadas funções somente podem ser implementadas a partir de circuitos que utilizam sinais de relógio com freqüências maiores que 2048 kHz, tais como 16384 kHz, 8192 kHz ou 4096 kHz.

Utilizando-se um circuito *PLL* clássico, é possível implementar um gerador de base de tempo, obtendo assim um sinal de relógio com freqüência de 16384 kHz (multiplicação de freqüência), a partir de qualquer uma das fontes primárias selecionadas de 2048 kHz. Este gerador pode ainda fornecer outras freqüências, se necessárias. A figura 5.1 mostra um diagrama de blocos do módulo gerador de base de tempo proposto.

O circuito é formado por um oscilador controlado por tensão (bloco *VCO*), um filtro passa-baixas (bloco *LPF*), um comparador de fases (bloco *PC*) e uma cadeia de três divisores de freqüência (blocos *FF1*, *FF2* e *FF3*). O oscilador controlado

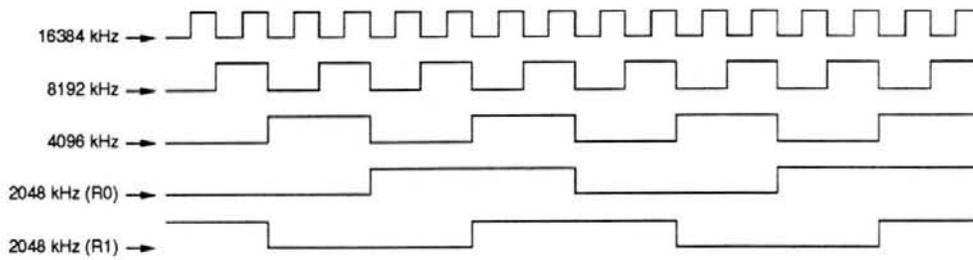


Figura 5.2 - Sinais de saída do módulo gerador de base de tempo.

por tensão fornece uma frequência nominal de 16384 kHz. Através de sucessivas divisões, obtém-se a partir da frequência de 16384 kHz, três outras frequências, quais sejam, 8192 kHz, 4096 kHz e 2048 kHz, disponibilizadas respectivamente à saída dos divisores  $FF1$ ,  $FF2$  e  $FF3$ . A frequência de 2048 kHz fornecida à saída de  $FF3$  é denominada  $R_0$ . Um quarto divisor de frequência (bloco  $FF4$ ), gera um segundo sinal de relógio de 2048 kHz, atrasado de 90 graus elétricos de  $R_0$ . Este segundo sinal de relógio é denominado  $R_1$ . Assim, o módulo gerador de base de tempo fornece duas fases de sinal de relógio de 2048 kHz, sendo  $R_0$  a fase principal e  $R_1$  a fase auxiliar, além das frequências superiores de 4096 kHz, 8192 kHz e 16384 kHz. Na figura 5.2 podem ser vistos os sinais de relógio provenientes do módulo gerador de base de tempo.

A fonte primária de sinal de relógio selecionada, fornece uma referência de 2048 kHz que será comparada em fase com  $R_0$ . A saída do comparador de fase produz um sinal cuja largura é proporcional ao nível de defasagem entre a fonte primária selecionada e a fase principal do relógio de 2048 kHz  $R_0$ . O filtro passa-baixas exibe em sua saída uma tensão  $DC$  proporcional à largura do sinal em sua entrada e, portanto, ao defasamento entre os sinais de 2048 kHz. Finalmente, esta tensão atua sobre o oscilador controlado por tensão, de modo a manter a fase do sinal de 2048 kHz derivado (sinal  $R_0$ ), amarrada à fase do sinal de 2048 kHz proveniente da fonte de sinal de relógio selecionada.

A alternativa ora apresentada de utilizar-se um  $PLL$  para derivação de frequências superiores à da fonte primária selecionada, mostrou como exemplo de implementação um diagrama de blocos de um sistema analógico. Esta implementação

entretanto, poderia ser feita a partir de *PLL*'s digitais. Contudo, uma das desvantagens dos *PLL*'s digitais, é que a frequência de saída nestes dispositivos é obtida a partir da divisão da frequência de um sinal de relógio primário, por um fator que pode ser tipicamente tão grande quanto 64 [LEE 90], para que fique assegurada a obtenção de um sinal de relógio com razoável qualidade, em termos de nível de *jitter*. Assim, para o caso em questão, qual seja o de obter uma frequência de 16384 kHz, seria necessário um sinal de relógio primário com frequência em torno de 1,05 GHz! O projeto de circuitos digitais em frequências desta ordem é intrinsecamente complexo e custoso, além de exigir o emprego de tecnologias próprias para altas frequências, de modo que do ponto de vista da implementação prática visada neste projeto, parece razoável optar por um *PLL* analógico.

Não é entretanto objetivo deste trabalho especificar circuitos que não sejam passíveis de integração em silício, utilizando tecnologia *CMOS*, razão pela qual o projeto deste *PLL* passa a integrar o elenco de atividades classificadas como trabalhos futuros, quando da implementação do equipamento multiplexador propriamente dito.

## 5.2 Módulos para Multiplexação

O processo de multiplexação consiste na montagem e transmissão dos quadros *E1*, a partir dos dados do canal de serviço e dos octetos recebidos dos canais tributários. A figura 5.3 apresenta um diagrama de interconexão dos módulos de circuito executores das funções envolvidas neste processo.

Os módulos de recepção *G.703* (*RX1* a *RX31*) acolhem o sinal tributário entrante nas interfaces de 64 kbit/s (*T1* a *T31*), disponibilizando ao multiplexador um octeto a cada 125  $\mu$ s. Estes módulos são responsáveis pela regeneração do sinal recebido da linha, recuperação de relógio de 64 kHz, decodificação *G.703* e detecção

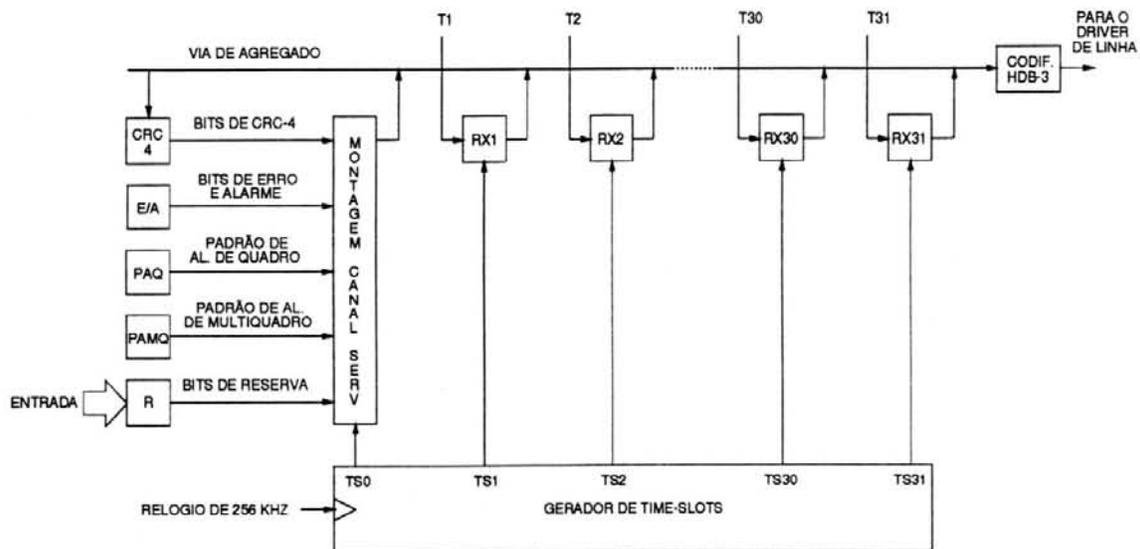


Figura 5.3 - Diagrama de blocos funcionais para multiplexação.

de violações à regra de codificação. A função de armazenamento elástico também faz parte dos módulos de recepção *G.703*.

O módulo gerador de *time-slots* temporiza a transferência destes octetos, à via de agregado. Através da via de agregado, os bits do canal de serviço e dos octetos recebidos são levados ao módulo codificador *HDB-3* e transferidos à linha de transmissão, por meio do circuito *driver* de linha (não mostrado na figura).

Nos quadros que contêm o padrão de alinhamento de quadro, durante o *time-slot* do canal de serviço (canal 0), o primeiro bit a ser inserido na via de agregado é um bit proveniente do módulo gerador dos coeficientes *CRC-4*. Em seguida são inseridos na via de agregado, os sete bits constituintes do padrão de alinhamento de quadro, finalizando o *time-slot* do canal 0. Com a ativação em seqüência dos *time-slots* dos canais 1 a 31, têm-se a montagem e transmissão completa de 256 bits de um quadro *E1*.

Nos quadros que não contêm o padrão de alinhamento de quadro, os bits são inseridos na via de agregado, a partir do bit de alinhamento de multiquadro, seguido do segundo bit (este sempre fixado em nível lógico "1"), o bit de alarme (*A*) e os bits de reserva. Os bits de reserva são provenientes de uma porta de entrada,

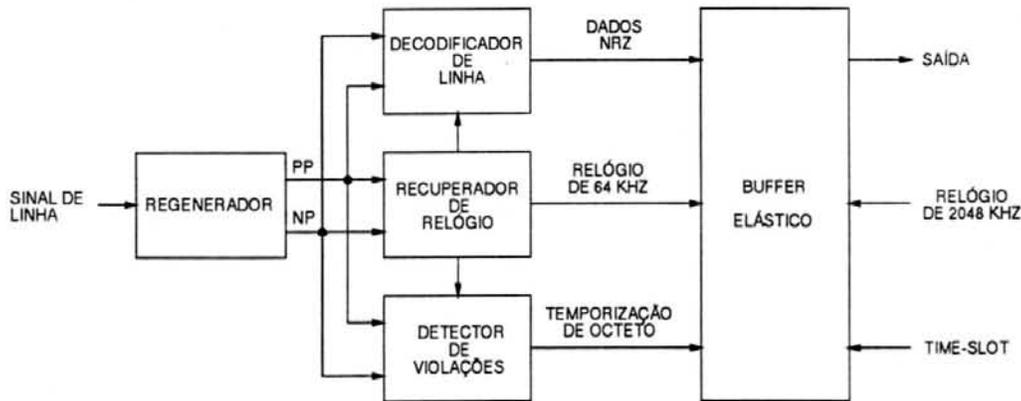


Figura 5.4 - Diagrama de blocos de um módulo de recepção *G.703*.

a qual possibilita acesso de um terminal ao canal interno de comunicação de 20 kbit/s, para troca de mensagens de gerenciamento e controle de rede. Nos quadros de número 13 e 15 do multiquadro, são transmitidos os bits de de indicação de erro *CRC-4* (bits *E*). Estes bits de erro são fornecidos pelo verificador *CRC-4* da seção demultiplexadora. Desta forma, sempre que for detectado erro na verificação *CRC-4* sobre o agregado recebido do equipamento remoto, os bits de erro são ativados.

A seguir, será apresentada a arquitetura proposta para os módulos empregados na multiplexação. No apêndice deste trabalho podem ser vistos os circuitos contidos nos blocos que compõem os módulos apresentados nesta seção.

### 5.2.1 Módulos de Recepção para Interface *G.703* de 64 kbit/s

Os módulos de recepção para interface *G.703* de 64 kbit/s são os circuitos recuperador de relógio de 64 kHz, decodificador de linha, detector de violações e *buffer elástico*. A figura 5.4 mostra um diagrama de blocos do canal de recepção *G.703*. O sinal codificado proveniente da linha de recepção é levado ao circuito regenerador, que fornece os sinais unipolares *NRZ PP* e *NP*, correspondentes aos pulsos positivos e negativos que formam o sinal bipolar de linha. Este circuito regenerador é implementado por meio de técnicas analógicas, não sendo especificado neste trabalho.

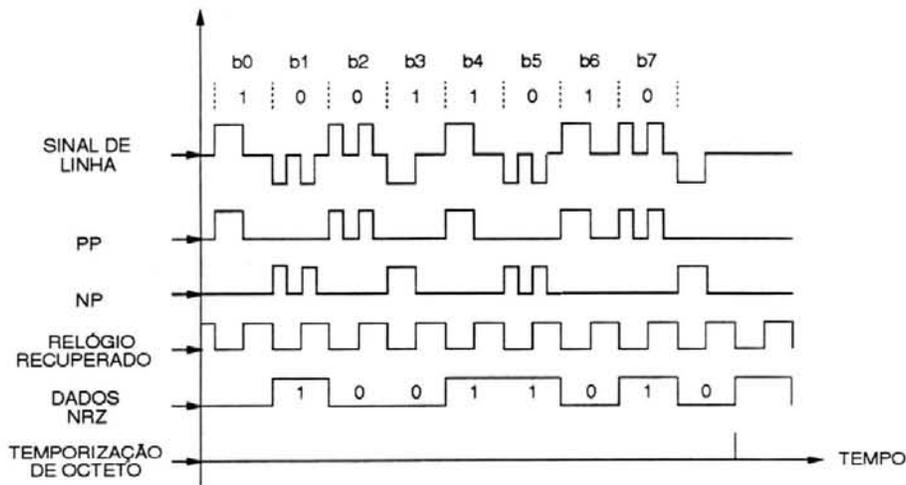


Figura 5.5 - Formas de onda para o módulo de recepção *G.703*.

A partir dos sinais unipolares *PP* e *NP*, o circuito recuperador de relógio fornece um sinal de relógio de 64 kHz, que é utilizado pelos circuitos decodificador de linha e detector de violações. O sinal de relógio recuperado de 64 kHz é ainda utilizado para cadenciar a escrita dos dados no *buffer elástico*. Os sinais de temporização de octeto, relógio de 2048 kHz e *time-slot* também são empregados na escrita e leitura de dados sobre o *buffer elástico*. A saída do *buffer elástico* constitui-se em rajadas de 8 bits (octetos) a 2048 kbit/s, que repetem-se a intervalos de 125  $\mu$ s. Este sinal é levado à via de agregado, sendo após codificado e transmitido ao equipamento demultiplexador remoto.

A figura 5.5 mostra as formas de onda nos diferentes pontos do diagrama de blocos do canal de recepção *G.703* mostrado na figura 5.4.

A seguir serão apresentados e descritos cada um dos circuitos pertencentes ao canal de recepção *G.703*.

#### 5.2.1.1 Circuito de Recuperação de Relógio de 64 kHz

Este circuito tem por finalidade produzir um sinal de relógio com frequência de 64 kHz, a partir do sinal tributário recebido da linha. O sinal de relógio

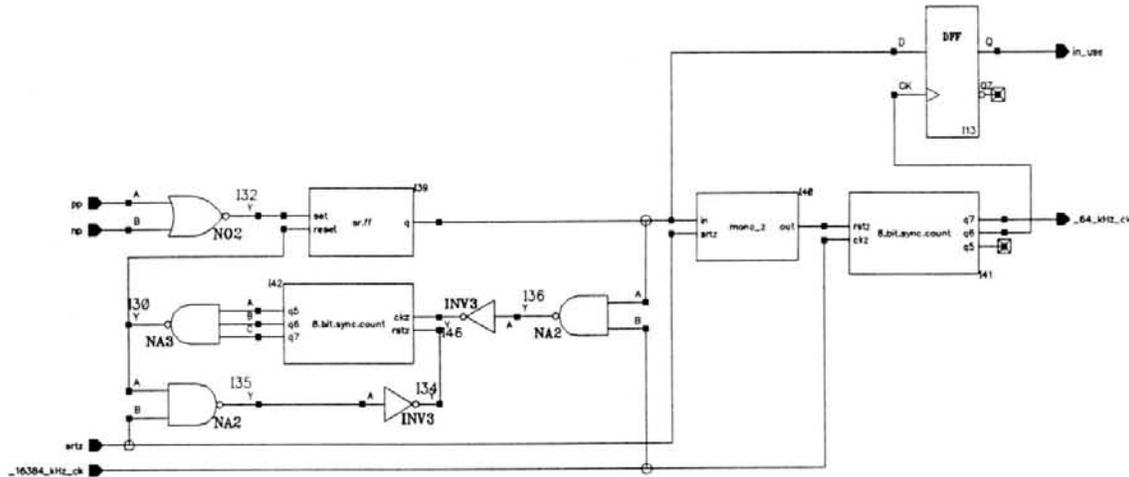


Figura 5.6 - Circuito de recuperação de relógio de 64 kHz.

recuperado é então utilizado na decodificação do sinal de linha recebido, na geração do sinal de temporização de octeto e na escrita dos dados recebidos sobre o *buffer elástico* de tributário.

Para a implementação deste circuito, explorou-se uma característica da codificação *G.703* a 64 kbit/s, qual seja a de que um bloco-código sempre inicia por uma transição lógica ascendente, seja este bloco representativo de um bit *zero* ou *um*. Esta característica pode ser comprovada na figura 3.6, da seção 3.2.

As transições ascendentes que caracterizam o início de um bloco-código são utilizadas como referências de tempo para a recuperação do sinal de relógio de 64 kHz. Assim, um período do sinal de relógio de 64 kHz é obtido através da contagem de 256 ciclos de um sinal de relógio com frequência de 16384 kHz, em que o processo de contagem sempre é inicializado através de um sinal de reset aplicado aos contadores. Este sinal de reset é gerado a partir das transições ascendentes, no início dos blocos código *G.703*. Assim, o período do sinal de relógio recuperado é corrigido a cada bloco código, traduzindo portanto a temporização do sinal recebido da linha.

O circuito de recuperação de relógio de 64 kHz fornece ainda uma indicação de *status* da linha de recepção. Caso haja sinal na linha, uma saída denomi-

nada *in use* permanece ativa, em nível lógico *um*, caso contrário, esta saída assume nível lógico *zero*. Este recurso é fundamental para o controle de falhas do sistema, de que trata a recomendação *G.736*.

Nas figuras 5.6 e 5.7 podem ser vistos respectivamente o esquemático e as formas de onda obtidas através da simulação do circuito de recuperação de relógio de 64 kHz. Os sinais *pp* e *np* são os sinais fornecidos pelo circuito regenerador de linha. O sinal *srtz* é um sinal de reset, necessário apenas à simulação do circuito. O sinal de relógio *16384 kHz ck* é fornecido pelo módulo gerador de base de tempo. Os sinais *64 kHz ck* e *in use* são respectivamente o relógio de 64 kHz recuperado e o sinal de *status* de linha. Na figura 5.7 pode ser vista a desativação do sinal *in use* (em torno de 150  $\mu$ s), devido a supressão do sinal de linha (sinais *pp* e *np* em nível lógico zero). É interessante observar que o sinal de relógio de 64 kHz continua a ser gerado mesmo com a falta de sinal na linha.

Devido à topologia do circuito, as transições ascendentes do sinal de relógio recuperado ocorrem sincronamente a transições descendentes do relógio de 16384 kHz e nominalmente a meio período de um bloco-código *G.703*. O termo "nominalmente" significa que, devido ao *jitter*, as transições mencionadas podem não ocorrer precisamente no centro dos intervalos dos blocos-código.

Na simulação do circuito de recuperação de relógio de 64 kHz, os sinais de entrada na interface de 64 kbit/s (sinais *pp* e *np*) foram especificados da seguinte maneira: um bloco-código representativo de um dígito binário *um*, é formado por 64 sub-intervalos de tempo em nível lógico *um* e 64 sub-intervalos de tempo em nível lógico *zero*; o bloco-código que representa o dígito binário *zero*, é formado por 32 sub-intervalos de tempo em nível lógico *um*, seguidos de 32 sub-intervalos de tempo em nível lógico *zero*, mais 32 sub-intervalos de tempo novamente em nível lógico *um*, seguindo-se outros 32 sub-intervalos de tempo em nível lógico *zero*. Com esta especificação, a granularidade dos sinais *pp* e *np* é de 128 sub-intervalos de tempo.

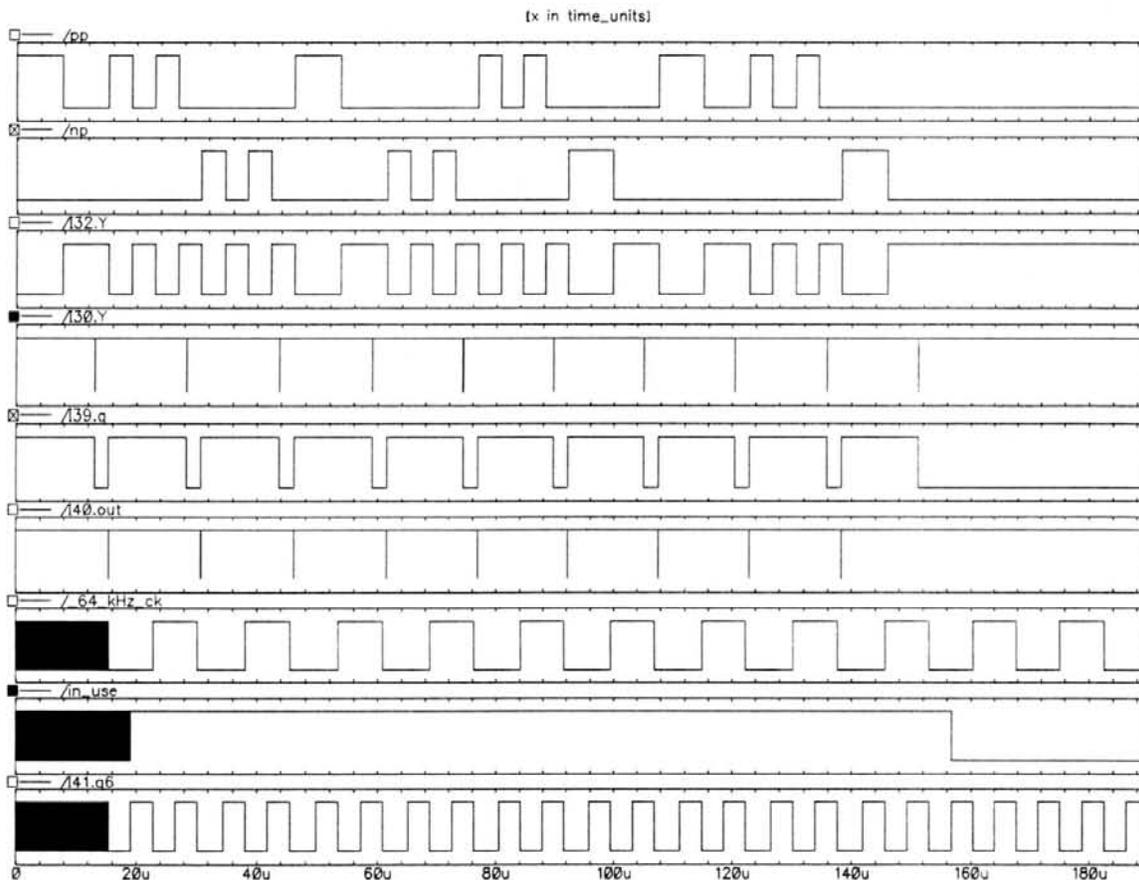


Figura 5.7 - Simulação do circuito de recuperação de relógio.

Como o período de um bloco código *G.703* à 64 kbit/s é nominalmente  $15,625 \mu\text{s}$ , cada sub-intervalo de tempo utilizado na formação dos blocos-código tem portanto, um período de  $122,0703125 \text{ ns}$ . Cada sub-intervalo acrescido ou suprimido, modifica portanto o período de um bloco-código em  $122,0703125 \text{ ns}$ . Com isto é possível a simulação de sinais de recepção contaminados por *jitter* com níveis tais quais os estabelecidos na recomendação *G.823*. Por meio de acréscimo ou supressão de sub-intervalos formadores dos blocos-código *G.703*, torna-se possível o aumento ou diminuição do período nominal de bit, alterando-se portanto de forma simulada, a posição no tempo dos instantes significativos do sinal de 64 kbit/s, na interface de recepção *G.703*.

Na curva de admissibilidade de *jitter* na interface de 64 kbit/s mostrada na seção 3.6, o pior caso acha-se na frequência de *jitter* de 20 kHz, onde é possível a ocorrência de deslocamentos instantâneos na fase do sinal entrante de até 244 ns

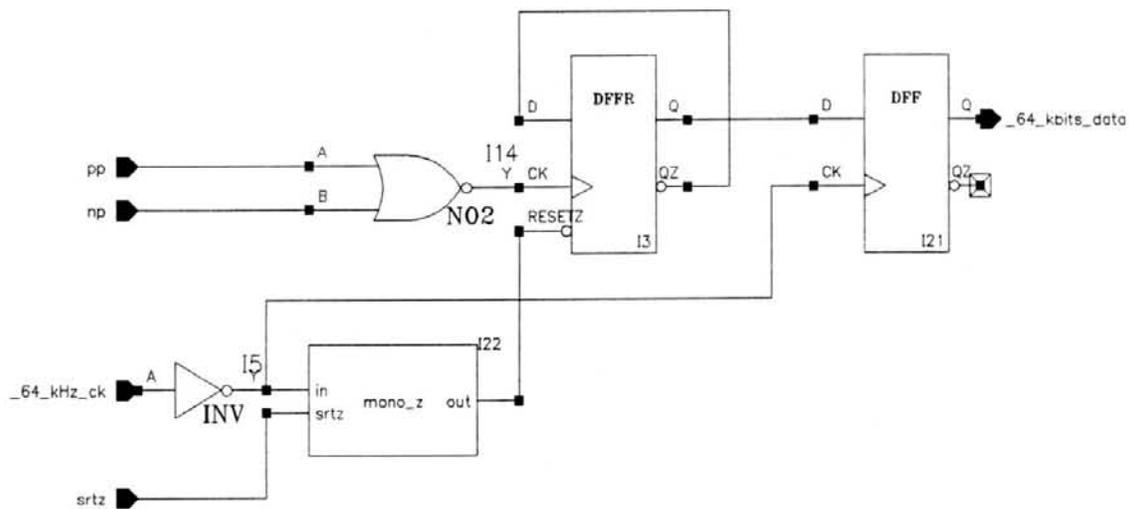


Figura 5.8 - Circuito decodificador *G.703*.

por intervalo de bit. Um deslocamento de fase desta ordem, representa em torno de 4 ciclos do relógio de 16384 kHz empregado na derivação do relógio de 64 kHz recuperado. Se for considerado que um período deste relógio de 64 kHz é formado por 256 ciclos do relógio de 16384 kHz, conclui-se que o desempenho da topologia adotada é suficiente para garantir um relógio recuperado de boa qualidade, já que uma diferença de 4 ciclos de relógio em 256 é perfeitamente tolerável.

#### 5.2.1.2 Decodificação *G.703*

Este circuito tem por finalidade decodificar o sinal de linha recebido, sendo o mais simples dos circuitos do canal de recepção *G.703*. Nas figuras 5.8 e 5.9 podem ser vistos respectivamente o esquemático e as formas de onda obtidas por simulação do circuito decodificador *G.703*.

O circuito decodificador recebe o sinal de relógio recuperado de 64 kHz e os sinais *pp* e *np*, além do sinal *srtz* necessário apenas à simulação. O sinal *64 kbits data* é o sinal de dados *NRZ* já decodificado. Este sinal, na saída do decodificador, apresenta um atraso de um bit em relação ao sinal codificado da linha, conforme pode ser constatado na figura 5.9.

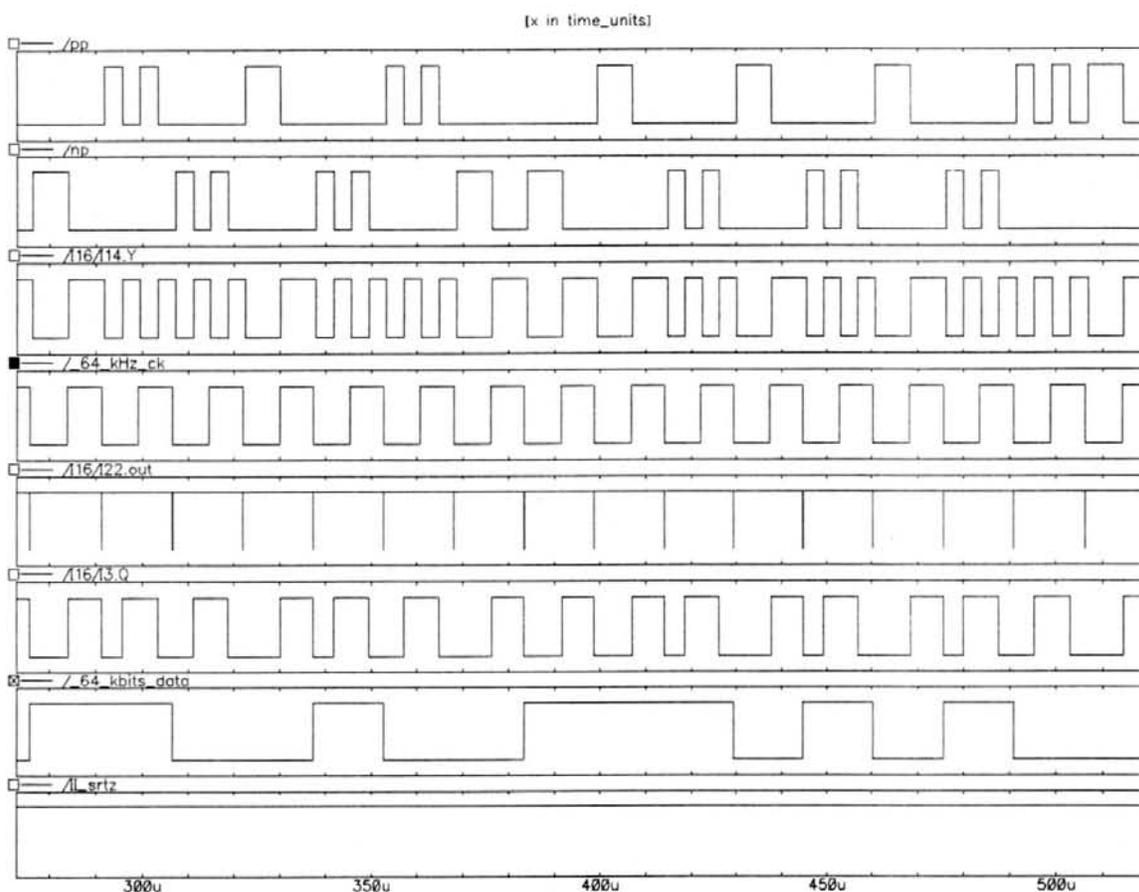


Figura 5.9 - Simulação do circuito decodificador *G.703*.

O sinal de dados a 64 kbit/s é então levado ao *buffer* elástico, a partir do qual será utilizado na formação dos quadros de transmissão a 2048 kbit/s.

### 5.2.1.3 Detecção de Violações

O circuito detector de violações produz em sua saída um impulso com duração igual a meio período do relógio de 8192 kHz ou seja, aproximadamente 61 ns. A transição ascendente deste impulso ocorre nominalmente no final do intervalo de bit do oitavo bit dos octetos de 64 kbit/s. Sua finalidade é portanto demarcar octetos. As transições ascendentes deste sinal ocorrem sincronamente a transições descendentes do sinal de relógio de 8192 kHz. Esta característica é particularmente importante para o processo de escrita e leitura sobre o *buffer elástico* e será explorada quando da apresentação da arquitetura proposta para aquele dispositivo.

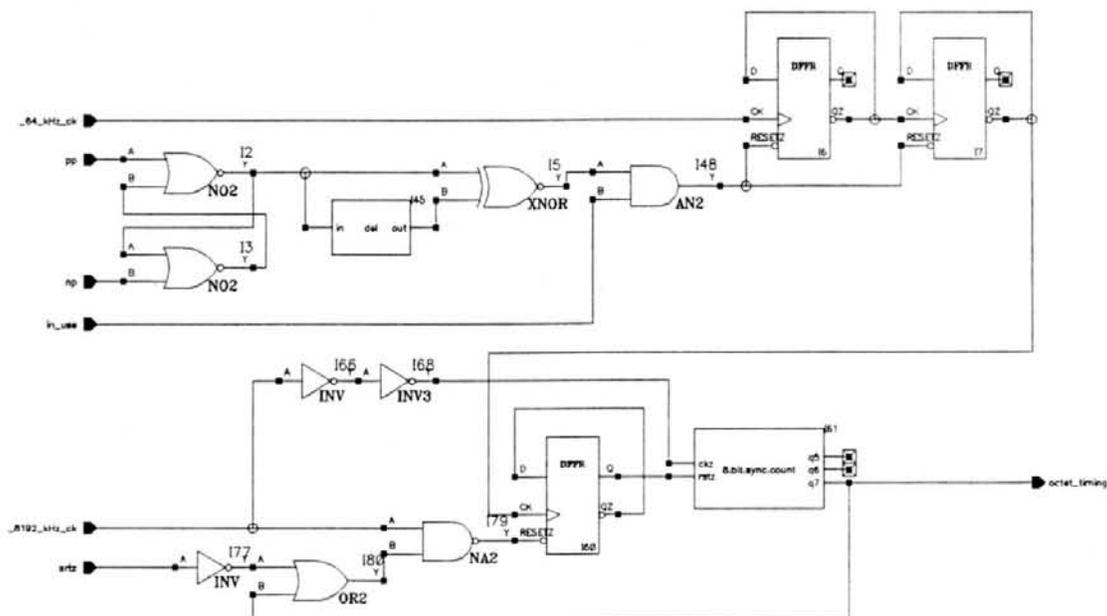


Figura 5.10 - Circuito detector de violações.

Nas figuras 5.10 e 5.11 podem ser vistos respectivamente o esquemático e as formas de onda resultantes da simulação do circuito detector de violações. O detector de violações recebe os sinais *pp* e *np*, relógio recuperado de 64 kHz, relógio de 8192 kHz, sinal *in use* e um sinal de reset (*srtz*) necessário apenas à simulação. O sinal *in use* é necessário para manter a saída *octet timing* inativa, caso não haja sinal na linha de recepção de 64 kbit/s.

#### 5.2.1.4 Armazenamento Elástico para Tributários

No capítulo 4 ficou evidente a necessidade do emprego de armazenamento elástico para compatibilizar a recepção dos octetos tributários com a temporização do multiplexador e viabilizar a montagem dos quadros *E1*. Assim, na figura 5.13 pode ser visto o diagrama de blocos do *buffer elástico* proposto neste trabalho. Cada um dos 31 canais tributários necessita de um *buffer elástico* tal qual o mostrado na figura.

Para análise do funcionamento da estrutura de armazenamento elástico proposta, é necessária a familiarização com os sinais a partir dos quais o dispositivo

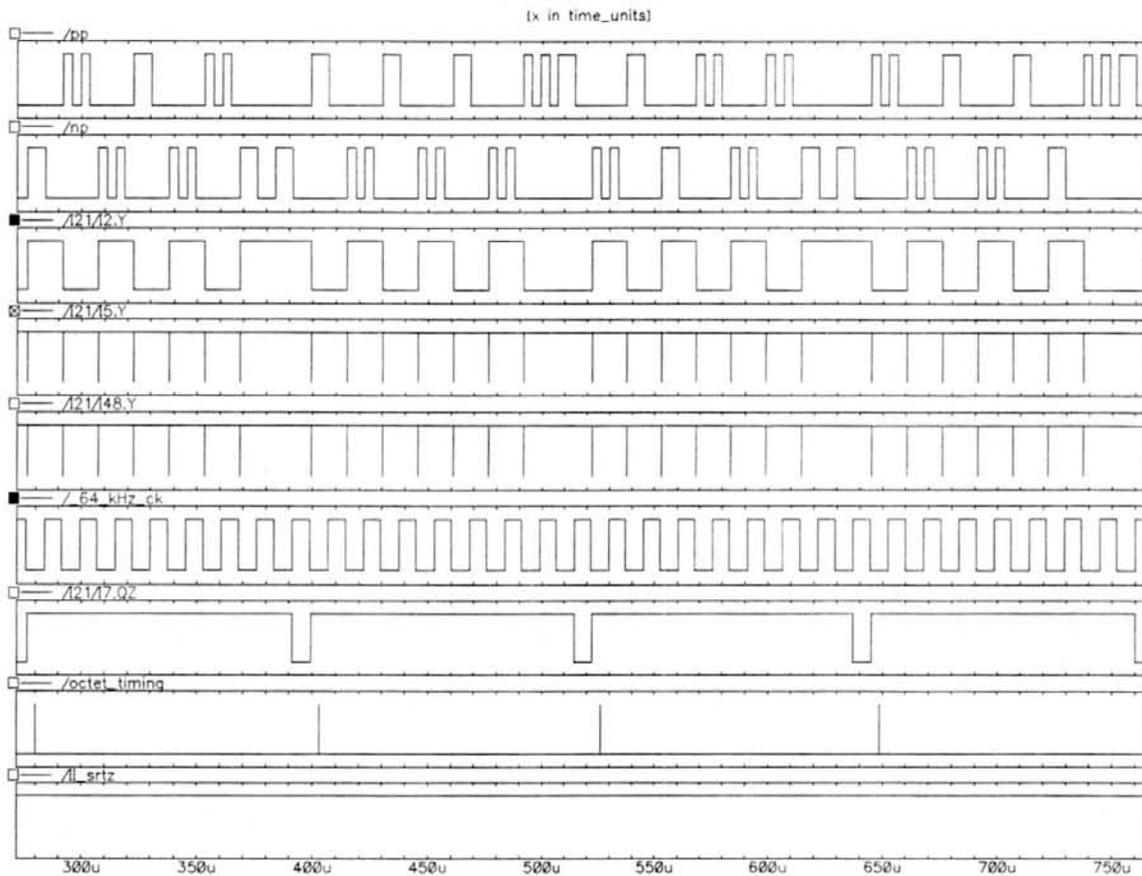


Figura 5.11 - Simulação do circuito detector de violações.

irá operar. Desta forma, na figura 5.12 podem ser vistas as formas de onda dos sinais em questão, os quais serão descritos a seguir.

O sinal *2048 kHz ck* é a fase principal do sinal de relógio do multiplexador. Este sinal é utilizado para cadenciar a codificação e transmissão dos dados dos tributários, na geração dos quadros *E1* de 2048 kbit/s. O sinal *2048 kHz ck* é proveniente do circuito gerador de base de tempo descrito na seção 5.1.

O sinal *mux read req* é o próprio *time-slot* atribuído ao canal tributário. Este sinal é fornecido por um circuito denominado gerador de *time-slots*, responsável pela geração dos 32 *time-slots*, para os canais tributários e de serviço. O circuito gerador de *time-slots* está descrito na seção 5.3.3.

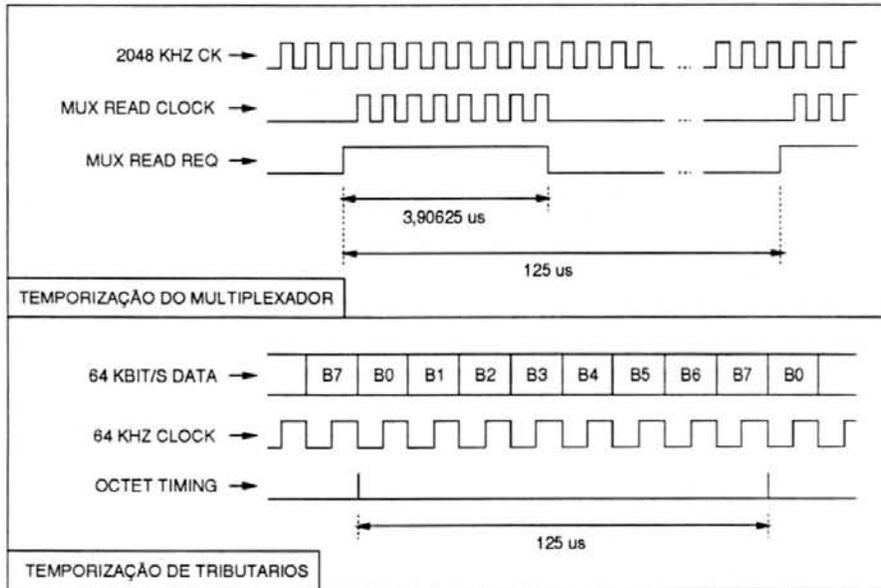


Figura 5.12 - Sinais para operação do *buffer elástico*.

Durante o intervalo de tempo em que o *time-slot* de um determinado canal tributário está ativo (em nível lógico *um*), um octeto é buscado no *buffer elástico*, codificado e transmitido ao equipamento remoto. A extração de um octeto do *buffer elástico* ocorre sincronamente à transição ascendente do sinal *mux read req*, sendo que este tem suas transições (ascendentes ou descendentes) sincronizadas a transições ascendentes do relógio de 8192 kHz. Portanto, a extração de um octeto do *buffer elástico* sempre ocorre sincronamente a uma transição ascendente do relógio de 8192 kHz.

Conforme pode ser visto na figura 5.12, um octeto é extraído do *buffer elástico* a cada 125  $\mu$ s, intervalo de tempo entre duas transições ascendentes do sinal *mux read req*. O sinal *mux read clock* é uma função lógica *E* dos sinais 2048 kHz *ck* e *mux read req*. As transições ascendentes deste sinal temporizam a codificação e transmissão do octeto extraído do *buffer elástico*.

Os sinais relativos aos octetos de tributário também podem ser vistos na figura 5.12. As transições ascendentes do relógio de 64 kHz (*64 kHz ck*) ocorrem a meio período do sinal de dados (*64 kbits data*) e o sinal de temporização de octeto

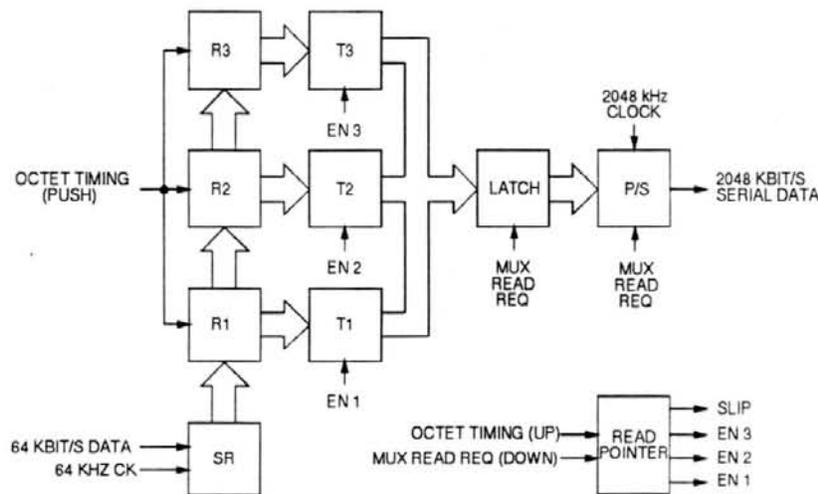


Figura 5.13 - Diagrama de blocos do *buffer elástico* de tributários.

(*octet timing*) ocorre entre dois octetos adjacentes, a cada oito ciclos de relógio de 64 kHz, a intervalos de nominalmente 125  $\mu$ s.

Tendo sido descritos os sinais a partir dos quais o *buffer elástico* irá operar, inicia-se a análise funcional de seu diagrama de blocos, mostrado na figura 5.13. Os diagramas de circuito dos blocos mostrados na figura 5.13, podem ser vistos no apêndice deste trabalho.

O bloco *SR* é um registrador de deslocamento. Este bloco recebe dados a 64 kbit/s (64 kbit/s data) que são nele escritos, cadenciados pelo relógio recuperado 64 kHz ck. Portanto, a cada oito ciclos deste relógio, tem-se à saída do registrador de deslocamento, um octeto completo, quando concomitantemente ocorre a ativação do sinal *octet timing*.

Os blocos *R1*, *R2* e *R3*, são registradores de oito bits de carga e leitura paralelas. A carga de dados nos registradores é realizada através da ativação das entradas *push* em cada registrador, pelo sinal *octet timing*. Nestas condições, o registrador *R1* carrega o conteúdo da saída do registrador de deslocamento (*SR*) e os registradores *R2* e *R3* carregam os conteúdos das saídas dos registradores *R1* e *R2*, respectivamente. Desta forma, os octetos vão sendo "empurrados" sucessivamente

nos registradores, à medida em que novos octetos vão sendo recebidos. Um octeto armazenado à saída do registrador *R3* é perdido após a ativação do sinal *octet timing*.

A extração de octetos dos registradores *T1*, *T2* e *T3* dá-se da maneira descrita a seguir. O conteúdo de um único registrador é levado à entrada de um *latch*, através da habilitação de um dos três *tristates*, *T1*, *T2* e *T3*, mostrados no diagrama de blocos. A ativação do sinal *mux read req*, durante o *time-slot* do canal, salva no *latch* o octeto presente em sua entrada. Em seguida, o octeto extraído de um dos registradores é convertido da forma paralela para a serial, utilizando-se oito ciclos do relógio de 2048 kHz, durante o período de ativação do *time-slot*.

A habilitação de um dos três *tristates* é realizada por um ponteiro de leitura, o módulo *READ POINTER* no diagrama de blocos. Este apresenta quatro saídas quais sejam, *EN1*, *EN2*, *EN3* e *SLIP*, cujos estados lógicos são dependentes das duas entradas, *UP* e *DOWN*. Uma única saída do ponteiro de leitura pode estar ativa (em nível lógico *um*) por vez. Assim, se a saída ativa for *EN1*, uma transição ascendente na entrada *UP* faz com que *EN1* seja desativada e *EN2* passe a ser a saída ativa. Uma nova transição ascendente na entrada *UP*, provoca a desativação da saída *EN2* e a ativação da saída *EN3*. Se a saída *EN3* encontra-se ativada e a entrada *UP* recebe uma nova transição ascendente, a saída *SLIP* torna-se ativa.

De maneira análoga, se a saída ativa for a saída *SLIP*, três sucessivas transições ascendentes na entrada *DOWN* terminarão por ativar a saída *EN1*. Uma nova transição ascendente na entrada *DOWN* desativará a saída *EN1*, sendo que nestas condições nenhuma das saídas do ponteiro de leitura estará ativa. Entretanto, se uma próxima transição ascendente ocorrer na entrada *DOWN*, a saída *SLIP* tornar-se-á ativa novamente. Assim, diz-se que o ponteiro de leitura é incrementado por uma transição ascendente na entrada *UP* e decrementado por uma transição ascendente na entrada *DOWN*.

As entradas *UP* e *DOWN* estão sujeitas aos sinais *octet timing* e *mux read req* respectivamente. Sabe-se que o sinal *octet timing* é responsável pela escrita de um

novo octeto de dados sobre o *buffer elástico* e que o sinal *mux read req* é responsável pela extração (leitura) de um octeto. Desta maneira, sempre que um novo octeto de dados é escrito no *buffer* elástico, o ponteiro de leitura é incrementado, sendo que sempre que o multiplexador extrai um octeto do *buffer*, o ponteiro de leitura é decrementado.

Como o sinal *octet timing* é ativado sincronamente a transições descendentes do relógio de 8192 kHz e o sinal *mux read req* é ativado sincronamente a transições ascendentes do relógio de 8192 kHz, conclui-se que as entradas *UP* e *DOWN* do ponteiro de leitura nunca serão ativadas ao mesmo tempo. Portanto, não existe a possibilidade da extração de um octeto do *buffer* elástico ocorrer no mesmo instante em que um novo octeto é escrito, o que garante um esquema confiável de escrita e leitura sobre o dispositivo.

Durante o *start-up* do sistema, por *default EN2* é a saída ativada. Ainda como condição de *start-up*, o ponteiro de leitura não sofre ação das entradas *UP* e *DOWN*, até que tenham sido contabilizadas duas ocorrências do sinal *octet timing*. Esta condição de partida é necessária para que o primeiro octeto válido para leitura, seja disponibilizado no registrador *R2*. A partir daí então, o ponteiro de leitura começa a ser sistematicamente incrementado e/ou decrementado pela ação dos sinais *octet timing* e *mux read req*, podendo isto se dar de diferentes maneiras, as quais serão analisadas a seguir.

Como por *default* a saída do ponteiro de leitura habilitada é *EN2*, o multiplexador inicia a extração de octetos a partir do registrador *R2*. Se a temporização do sistema é mesócrona, resulta que a cada operação de leitura de um octeto sobre o *buffer* corresponderá uma operação de escrita. Assim, após um octeto ter sido extraído do *buffer*, o ponteiro de leitura é decrementado. Entretanto, um novo octeto sendo escrito logo após no *buffer*, fará com que o ponteiro seja novamente incrementado e o próximo octeto a ser lido do *buffer* pelo multiplexador, o será novamente a partir do registrador *R2*. Na medida em que a temporização do sistema é

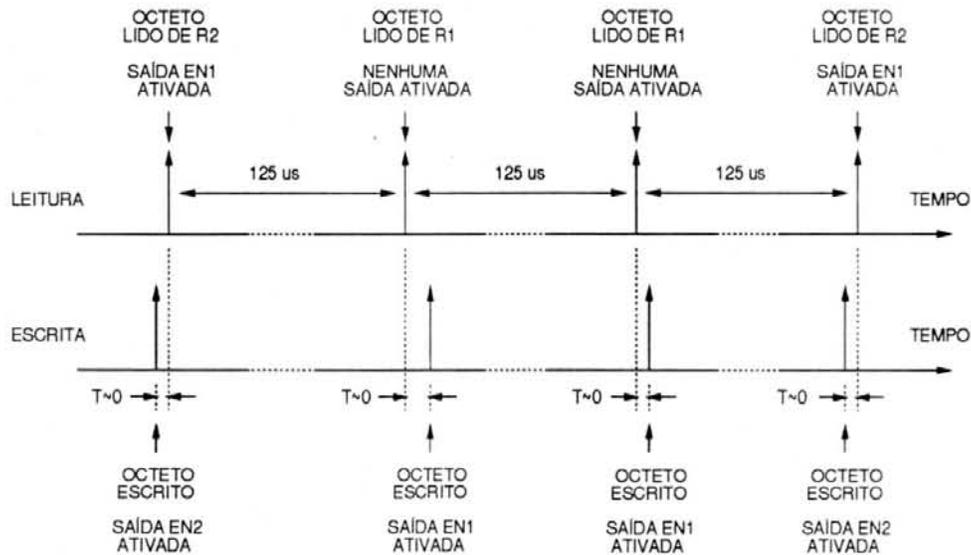


Figura 5.14 - Intervalo entre escrita e leitura próximo de zero.

mesócrona, este processo perdura indefinidamente, e o multiplexador extrai octetos sempre a partir do registrador *R2*.

O efeito do *jitter* no processo de escrita e leitura mesócrona sobre o *buffer elástico*, será a seguir analisado.

Sabe-se que a temporização dos octetos provenientes dos canais tributários pode estar contaminada por *jitter*, com os níveis máximos suportáveis pela interface de 64 kbit/s, estipulados na recomendação *G.823*. Desta forma, o intervalo de tempo observado entre a ocorrência dos sinais *octet-timing* (que traduz a temporização do sinal recebido) e *mux read req* (que traduz a temporização do multiplexador) pode apresentar variações instantâneas, ainda que na média este intervalo seja constante para, resalte-se, temporização mesócrona.

Se os eventos de escrita (*octet timing*) e leitura (*mux read req*) ocorrem demasiadamente próximos no tempo, dois casos há que se considerar. No primeiro caso, que pode ser visualizado na figura 5.14, um evento de escrita antecede um evento de leitura por um intervalo de tempo próximo a zero. As setas verticais indicam os instantes de escrita e leitura. Nestas condições, à esquerda da figura, o evento de escrita encontra-se posicionado ligeiramente à esquerda do evento de

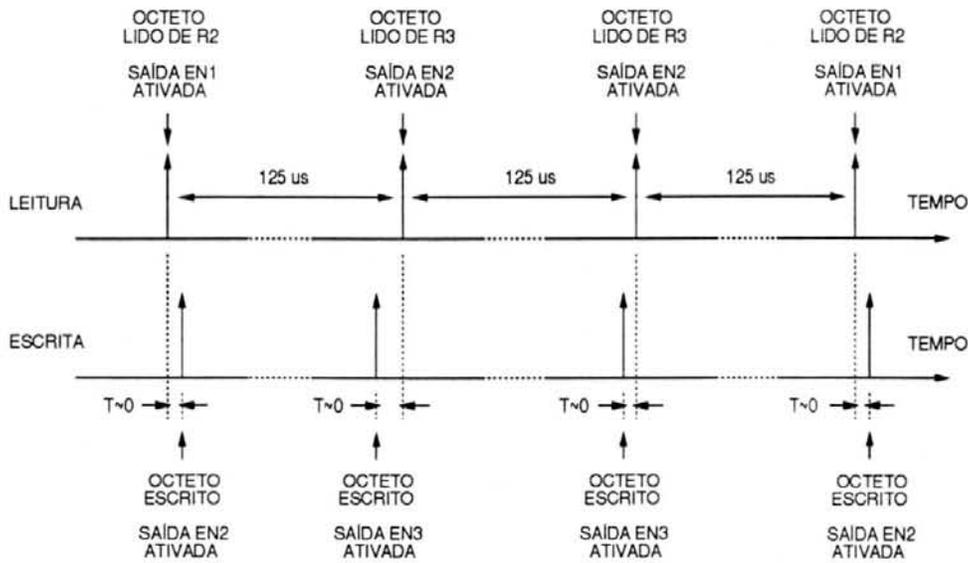


Figura 5.15 - Intervalo entre escrita e leitura próximo de 125  $\mu$ s.

leitura no eixo do tempo e o multiplexador extrai um octeto a partir do registrador *R2*.

Devido aos deslocamentos temporais provocados por *jitter*, poderá ocorrer que o próximo evento de escrita de um octeto sobre o *buffer* (segunda seta da esquerda para a direita, nominalmente após 125  $\mu$ s), seja atrasado. Assim, este evento apareceria neste instante posicionado à direita do evento de leitura. Com isto, têm-se dois eventos de leitura consecutivos sobre o *buffer* elástico. Neste caso, o multiplexador extrai o próximo octeto do registrador *R1* e não mais do registrador *R2*, onde acha-se o octeto anteriormente lido. Portanto, nesta situação em que o intervalo entre os eventos de escrita e leitura é próximo a zero, podem ocorrer duas escritas ou duas leituras consecutivas sobre o *buffer* elástico, fazendo com que o octeto a ser lido pelo multiplexador seja buscado ora no registrador *R2*, ora no registrador *R1*, tal como mostrado na figura.

No segundo caso, cuja representação gráfica pode ser vista na figura 5.15, o intervalo entre dois eventos de escrita e leitura aproxima-se de 125  $\mu$ s. Nestas condições, um evento de leitura antecede um evento de escrita por um intervalo de tempo próximo a zero, de forma que o evento de escrita encontra-se posicionado ligeiramente à direita do evento de leitura no eixo do tempo. Assim, conforme mos-

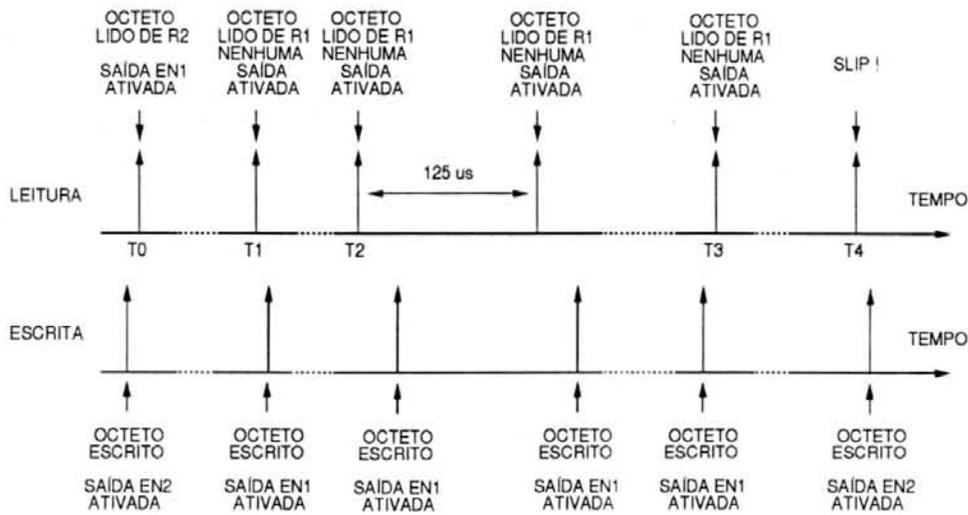


Figura 5.16 - Slip por *underflow*.

trado à esquerda da figura, o multiplexador extrai um octeto a partir do registrador *R2*. Entretanto, devido à ação do *jitter*, o próximo evento de escrita (segunda seta da esquerda para a direita) é deslocado de sua posição nominal, sendo posicionado agora à esquerda do evento de leitura. Nestas condições, ocorrem dois eventos consecutivos de escrita sobre o *buffer* e a próxima leitura de um octeto do *buffer*, será realizada a partir do registrador *R3*. Portanto, na situação em que o intervalo entre eventos de escrita e leitura é próximo a  $125 \mu\text{s}$ , podem ocorrer também duas escritas ou duas leituras consecutivas sobre o *buffer* elástico, fazendo com que a leitura de octetos do *buffer* seja realizada ora no registrador *R2*, ora no registrador *R3*.

Em qualquer um dos casos ora apresentado, é importante notar a ação dinâmica do ponteiro de leitura que sempre fornece a indicação correta acerca de qual dos três registradores contém um octeto válido para leitura, por parte do multiplexador. O efeito do *jitter* no sinal tributário recebido é portanto compensado pela ação do *buffer elástico*.

Para temporização plesiócrona, existe um *offset* entre os períodos dos eventos de escrita e de leitura. Com isto, a ocorrência de *slips* dar-se-á periodicamente, ou por *underflow* ou por *overflow* do *buffer*. Foi demonstrado que o ponteiro de leitura é capaz de indicar um dos três registradores a partir do qual será ex-

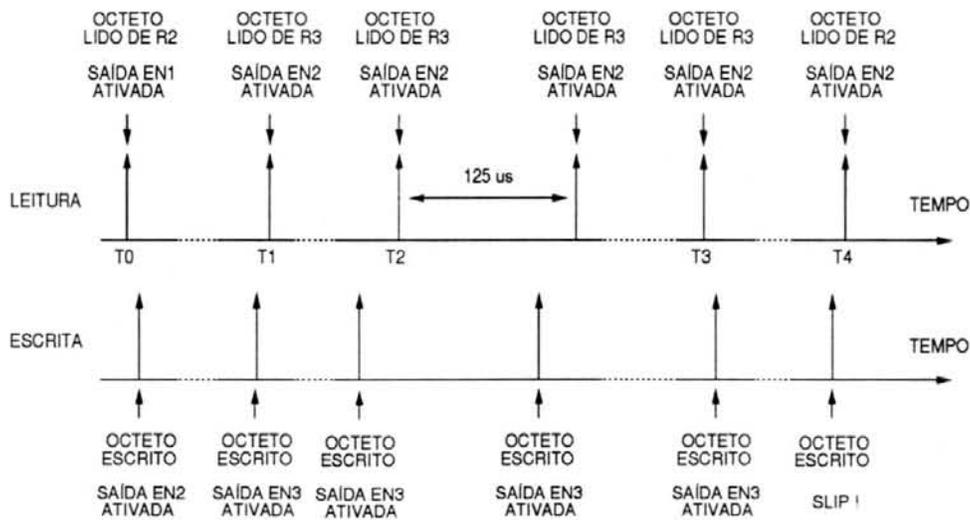


Figura 5.17 - *Slip* por *overflow*.

traído um octeto válido para leitura. Por isto, é possível a ocorrência de eventos consecutivos de escrita ou de leitura sobre o *buffer*.

Na temporização plesiócrona, devido ao deslocamento temporal relativo dos eventos de escrita e leitura, haverá de fato instantes em que devido à proximidade temporal entre estes aliada aos efeitos do *jitter* na temporização de escrita, irão ocorrer duas operações de escrita consecutivas ou duas operações de leitura consecutivas, conforme o sinal (positivo ou negativo) do *offset* entre os mesmos. A ação do ponteiro de leitura permite a ocorrência destas situações, de maneira similar àquela mostrada na figura 5.14 e também na figura 5.15. Entretanto, se após já terem ocorrido dois eventos consecutivos de leitura e, sem que tenham ocorrido dois eventos consecutivos de escrita, ocorrerem outros dois eventos consecutivos de leitura, estará caracterizado um *slip* por *underflow* do *buffer* elástico.

De maneira análoga, se após já terem ocorrido dois eventos consecutivos de escrita e, sem que tenham ocorrido dois eventos consecutivos de leitura ocorrerem dois outros eventos consecutivos de escrita, estará caracterizado um *slip* por *overflow* no dispositivo. Estas situações podem ser representadas graficamente, conforme mostrado nas figuras 5.16 e 5.17.

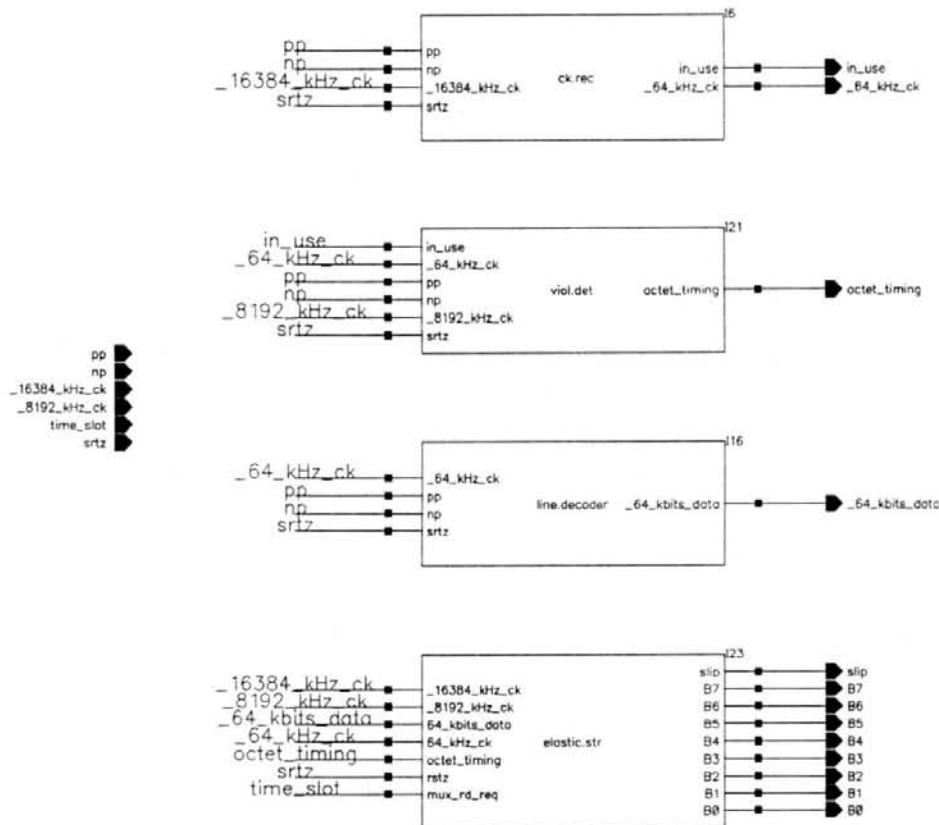


Figura 5.18 - Circuito para simulação conjunta dos módulos de recepção para interface *G.703*.

Na figura 5.16, entre os instantes de tempo  $T_0$  e  $T_1$ , ocorrem duas operações consecutivas de leitura. A partir de  $T_1$ , os octetos que vinham sendo extraídos do *buffer* a partir do registrador *R2*, passam a ser lidos a partir do registrador *R1* (intervalo de tempo entre  $T_2$  e  $T_3$ ). Entre os instantes de tempo  $T_3$  e  $T_4$ , ocorrem novamente duas operações consecutivas de leitura, ocasionando desta forma um *slip*.

Na figura 5.17, entre os instantes de tempo  $T_0$  e  $T_1$  ocorrem duas operações consecutivas de escrita. A partir daí, os octetos que vinham sendo lidos do registrador *R2*, passam a ser lidos a partir do registrador *R3* (intervalo de tempo entre  $T_2$  e  $T_3$ ). Entre os instantes de tempo  $T_3$  e  $T_4$ , ocorrem novamente duas operações consecutivas de escrita, ocasionando assim um *slip*. A ocorrência de um *slip* faz com que o *buffer elástico* seja reinicializado, tal como ocorre durante o *start-up* do sistema.

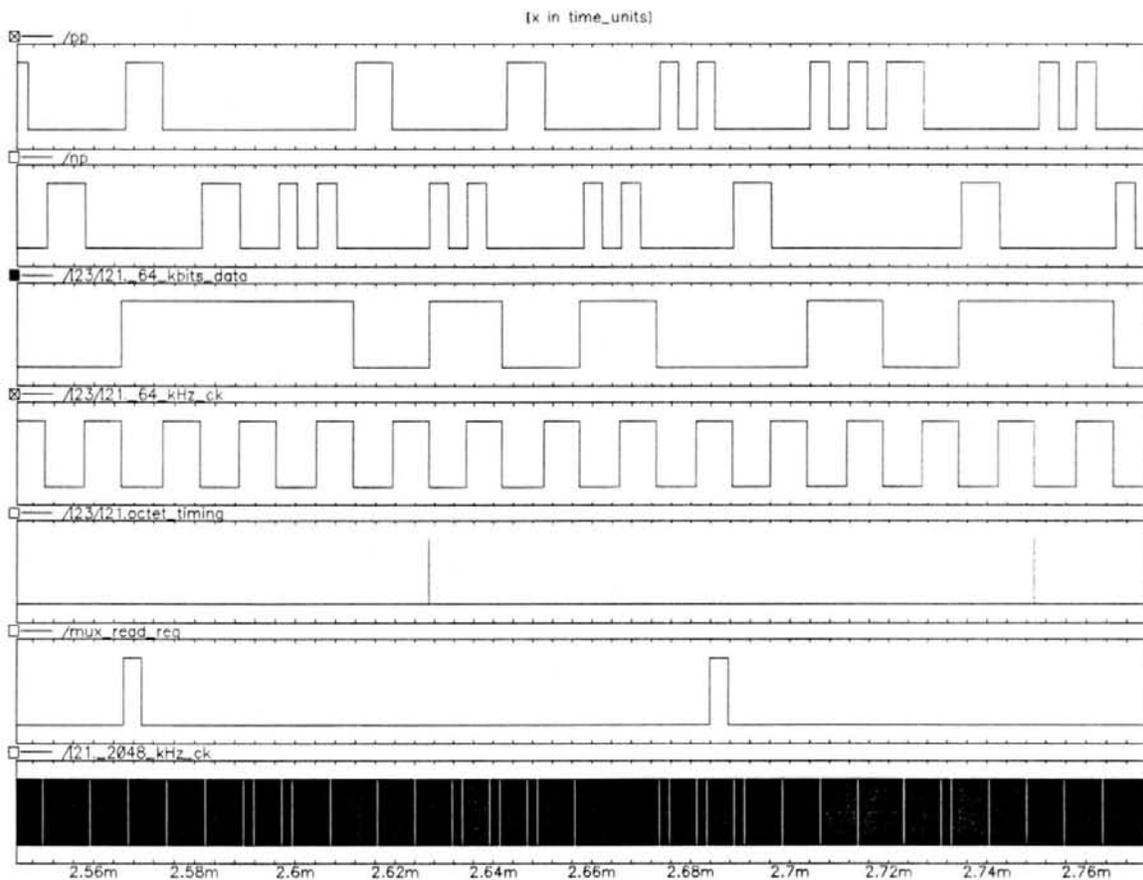


Figura 5.19 - Sinais de dados, relógio e temporização de octeto, obtidos por simulação do circuito completo.

Tendo sido brevemente descrito o funcionamento do *buffer elástico* proposto para a interface de recepção *G.703* de 64 kbit/s, serão a seguir apresentados os resultados das simulações desenvolvidas.

A fim de verificar o funcionamento conjunto dos módulos de recepção para interface *G.703* de 64 kbit/s, as simulações desenvolvidas foram realizadas através da interligação dos módulos de recuperação de relógio, decodificador, detector de violações e *buffer elástico*. Assim, na figura 5.18 pode ser visto o diagrama de blocos da estrutura simulada.

Os sinais de saída dos módulos que compõem o circuito, foram obtidos por simulação, a partir da definição dos sinais codificados de linha unipolares *pp* e *np*, dos relógios de 16384 kHz e 8192 kHz, além do *time-slot* do canal. Assim,

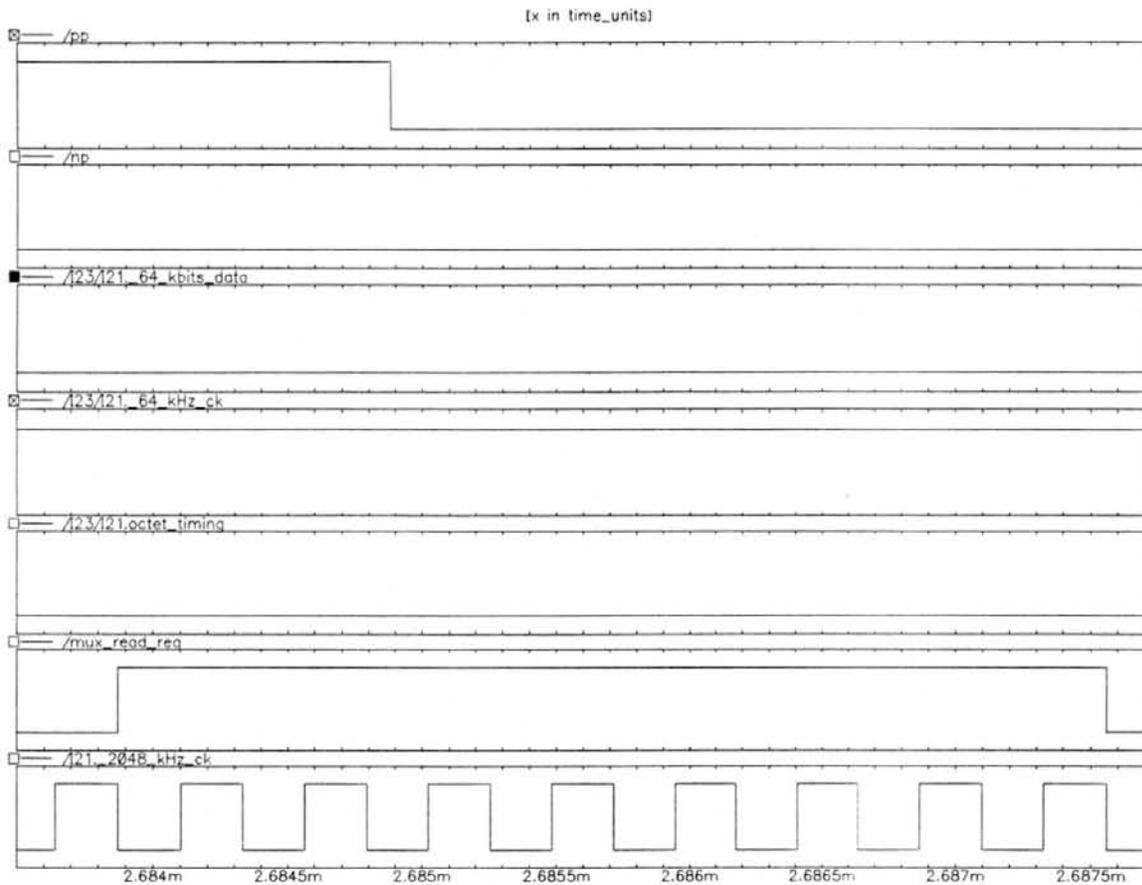


Figura 5.20 - Oito ciclos de 2048 kHz durante a ativação do *time-slot*.

na figura 5.19, são mostrados os sinais de saída *64 kbits data*, *64 kHz ck* e *octet timing*, resultantes do processamento dos sinais de entrada *pp* e *np*, respectivamente pelos circuitos decodificador, recuperador de relógio e detector de violações. Nesta mesma figura, são mostrados também o *time-slot* do canal (denominado na figura de *mux.read.req*) e o relógio de multiplexação de 2048 kHz.

Na definição dos sinais codificados de linha unipolares *pp* e *np*, simulou-se a existência de *jitter* nestes sinais, através da supressão ou adição de sub-intervalos com período de 122,0703125 ns, tal como realizado na simulação do circuito de recuperação de relógio.

Na figura 5.20, podem ser vistos em detalhe, oito ciclos do sinal de relógio de 2048 kHz, durante a ativação do *time-slot* do canal, que são utilizados para a conversão paralelo/serial do octeto presente à saída do *buffer elástico*.

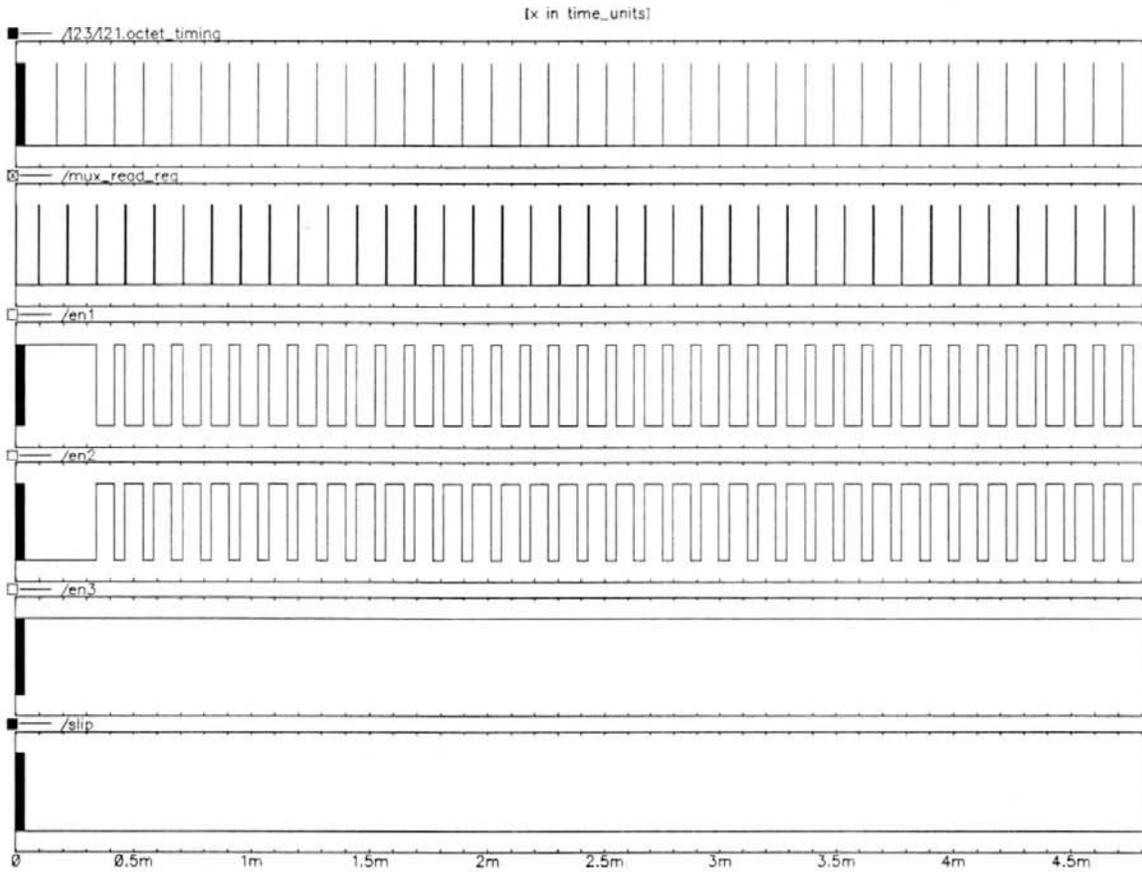


Figura 5.21 - Simulação para escrita e leitura mesócronas.

Nas simulações mostradas a seguir, poder-se-á observar os instantes de ocorrência dos eventos de escrita e leitura sobre o *buffer elástico*, bem como a ação dinâmica de endereçamento desenvolvida pelo ponteiro de leitura. Desta forma, na figura 5.21 é mostrada a simulação de escrita e leitura mesócronas sobre o *buffer*. Os sinais *en1*, *en2* e *en3*, quando em nível lógico *zero*, habilitam respectivamente os registradores *R1*, *R2* e *R3*.

A posição relativa no eixo do tempo entre os eventos de escrita e leitura (respectivamente os sinais *octet timing* e *mux read req*), permanece constante ao longo de todo o intervalo de simulação, devido ao fato de que os períodos médios de ambos os sinais é o mesmo. Assim, os octetos são lidos do *buffer* sempre a partir do registrador *R2*. Esta particularidade pode ser bem apreciada na visão de conjunto que a simulação da figura 5.21 oferece.

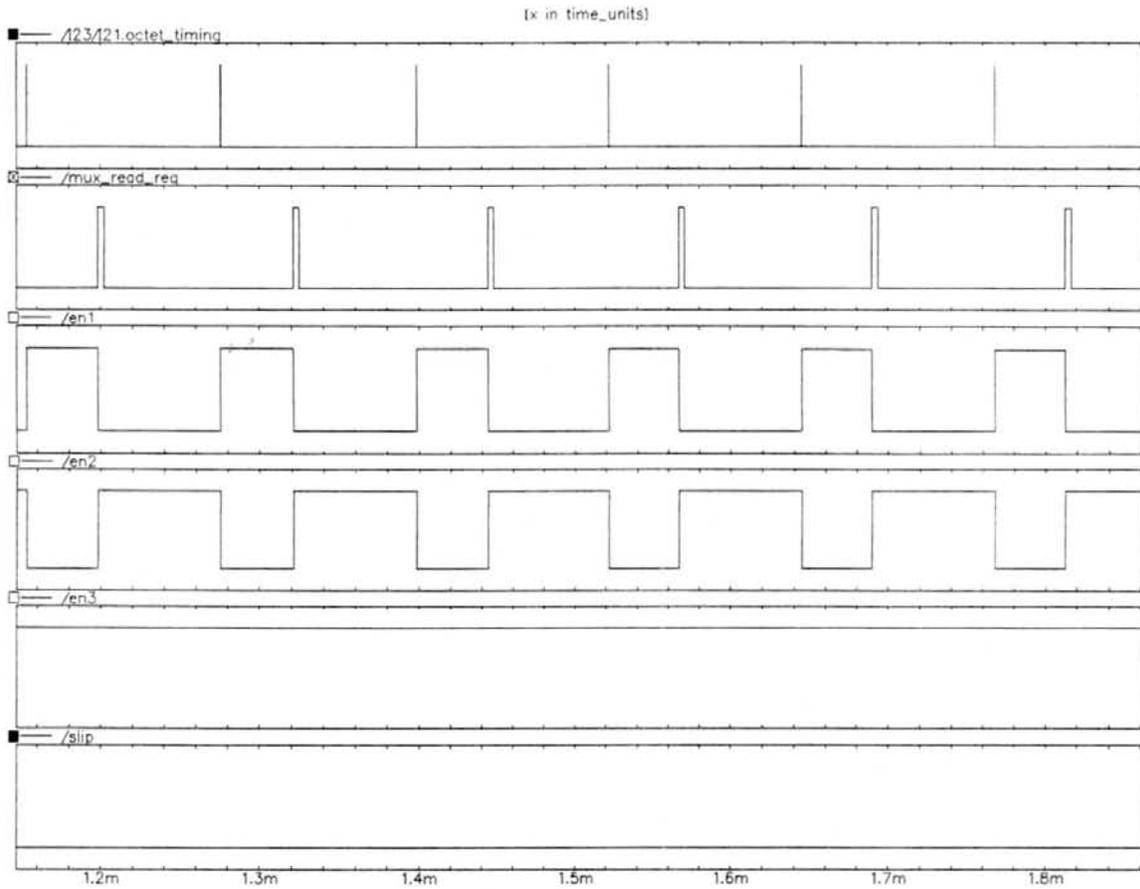


Figura 5.22 - Simulação para escrita e leitura mesócronas; vista em detalhe.

Na figura 5.22, que oferece uma visão mais localizada dos eventos de escrita e leitura, é interessante notar que a cada ativação do sinal *mux read request* (leitura de um octeto do *buffer*), o ponteiro de leitura é decrementado. Por isto, o sinal *en2* é desativado (nível lógico *um*) e o sinal *en1* é ativado (nível lógico *zero*).

Analogamente, a ativação do sinal *octet timing*, por ocasião da escrita de um octeto no *buffer*, faz com que o ponteiro de leitura seja incrementado, quando então o sinal *en1* é ativado (nível lógico *zero*) e o sinal *en2* é desativado (nível lógico *um*). Com a temporização mesócrona, não existe a ocorrência de *slips*, o que pode ser comprovado pela inatividade do sinal *slip* mostrado nas duas figuras anteriores.

A simulação para operação plesiócrona, cujos resultados são mostrados nas figuras a seguir, foi realizada a partir da especificação de temporizações distintas; uma que define o período do sinal codificado *G.703* de 64 kbit/s na interface de

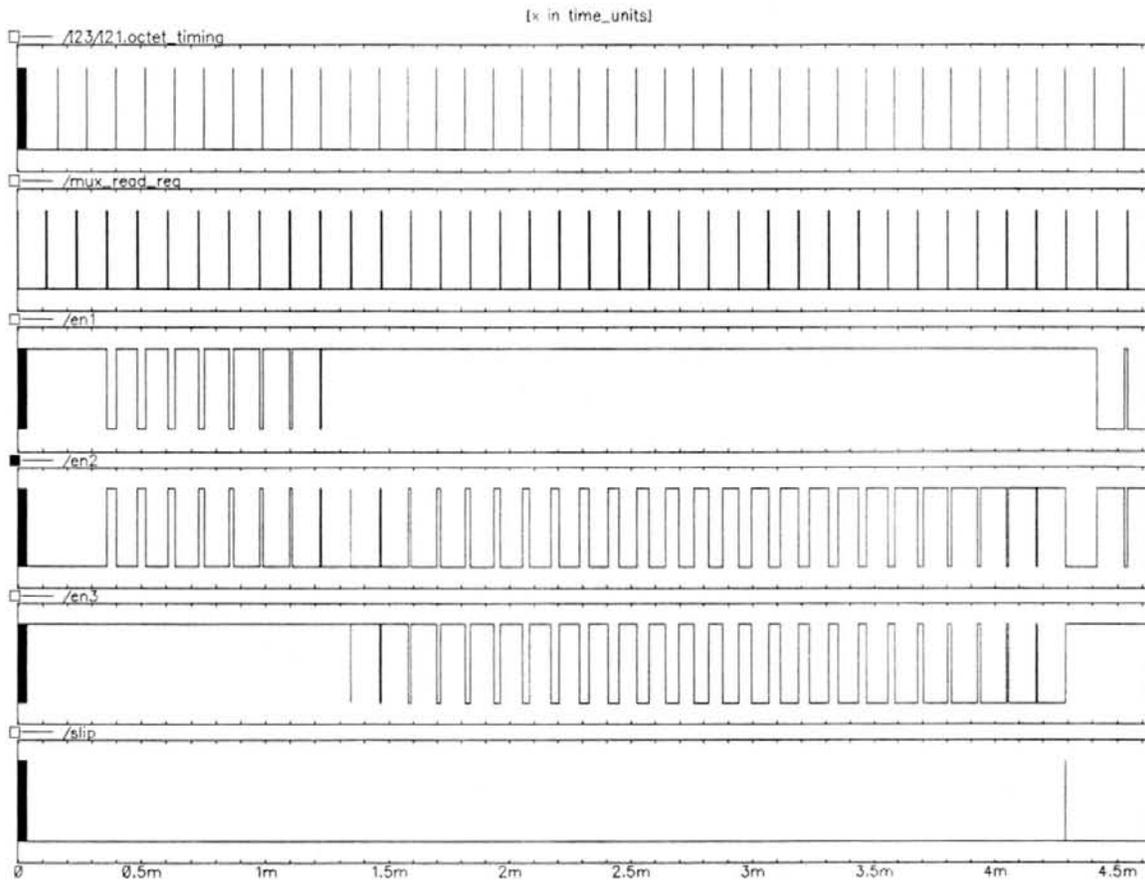


Figura 5.23 - Simulação para escrita e leitura plesiócronicas; *slip* por *overflow*.

recepção e outra que define o período do sinal de relógio de 16384 kHz, do qual é derivado o sinal *mux.read.req*. Um pequeno *offset* foi introduzido entre estas duas temporizações. Cabe ressaltar que o *offset* entre os eventos de escrita e leitura foi propositadamente super-dimensionado, a fim de que fosse possível observar a ocorrência de *slips* em um tempo exequível de simulação.

Na simulação mostrada na figura 5.23, os eventos de escrita ocorrem a intervalos ligeiramente menores que os eventos de leitura. Por isto, a leitura dos octetos inicia a ser feita a partir do registrador *R2* e, próximo ao instante de tempo 1,5 ms, passa a ser feita a partir do registrador *R3*. Próximo ao instante de tempo 4,5 ms, ocorre um *slip* por *overflow* do *buffer*. Note-se que após a ocorrência do *slip*, a leitura de octetos retorna a ser realizada a partir do registrador *R2*.

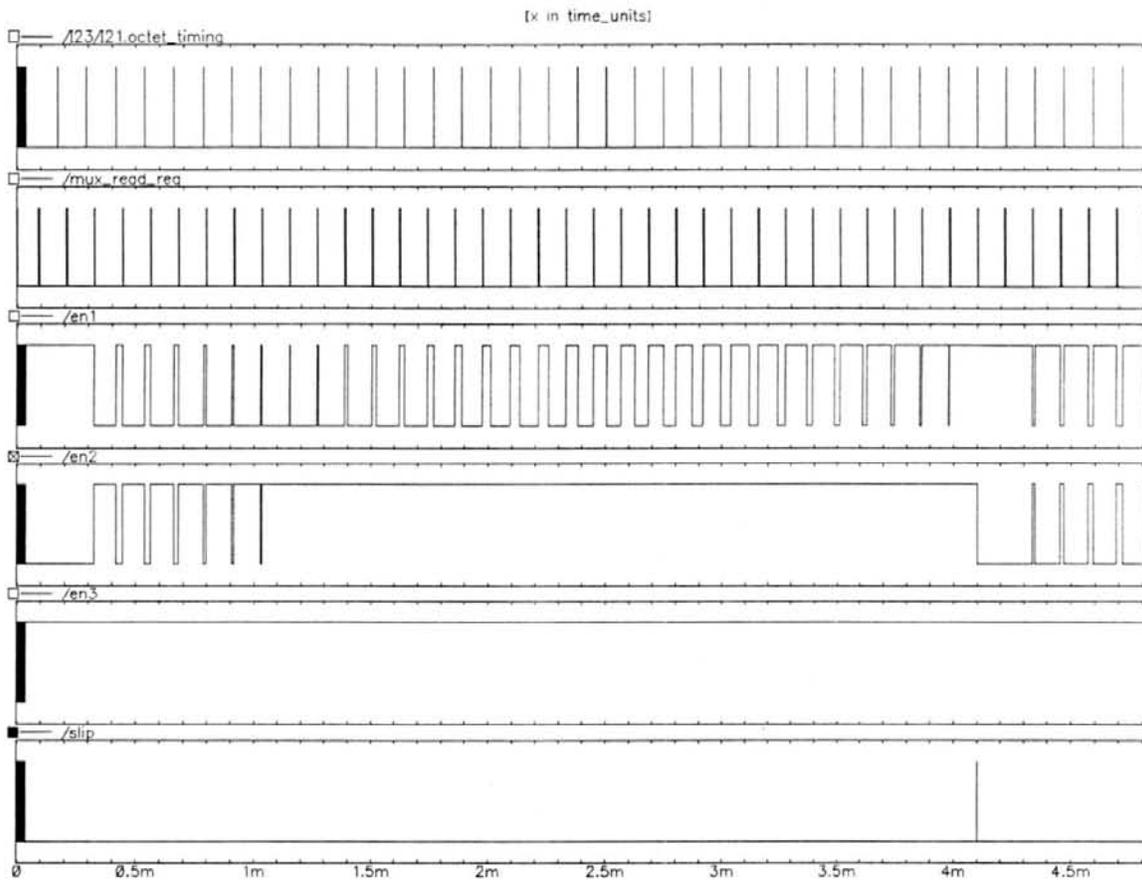


Figura 5.24 - Simulação para escrita e leitura plesiócronas; *slip* por *underflow*.

A ocorrência de um *slip* por *underflow* é o que mostra a simulação apresentada na figura 5.24. Neste caso, o período dos eventos de leitura é ligeiramente menor que o período dos eventos de escrita. Assim, a leitura que inicia a ser realizada a partir do registrador *R2*, passa a ser feita a partir do registrador *R1* (próximo ao instante de tempo 1 ms) e, próximo ao instante de tempo 4ms, ocorre o *slip*. Note-se que após a ocorrência do *slip*, a leitura de octetos retorna a ser realizada a partir do registrador *R2*.

As figuras 5.25 e 5.26 mostram as saídas *B0* a *B7* do *buffer elástico* e os sinais *pp* e *np*, além do *time-slot* do canal (sinal *ch16*) e sinal de relógio de 2048 kHz. Na figura 5.25, o octeto **1001 0011**, que ocorre aproximadamente entre os instantes de tempo 2,73 ms e 2,88 ms, é disponibilizado à saída do *buffer elástico* no flanco ascendente do *time-slot*, próximo ao instante de tempo 2,9 ms. Na figura 5.26, este *time-slot* é mostrado em detalhe, bem como o sinal de relógio de 2048 kHz, cujos

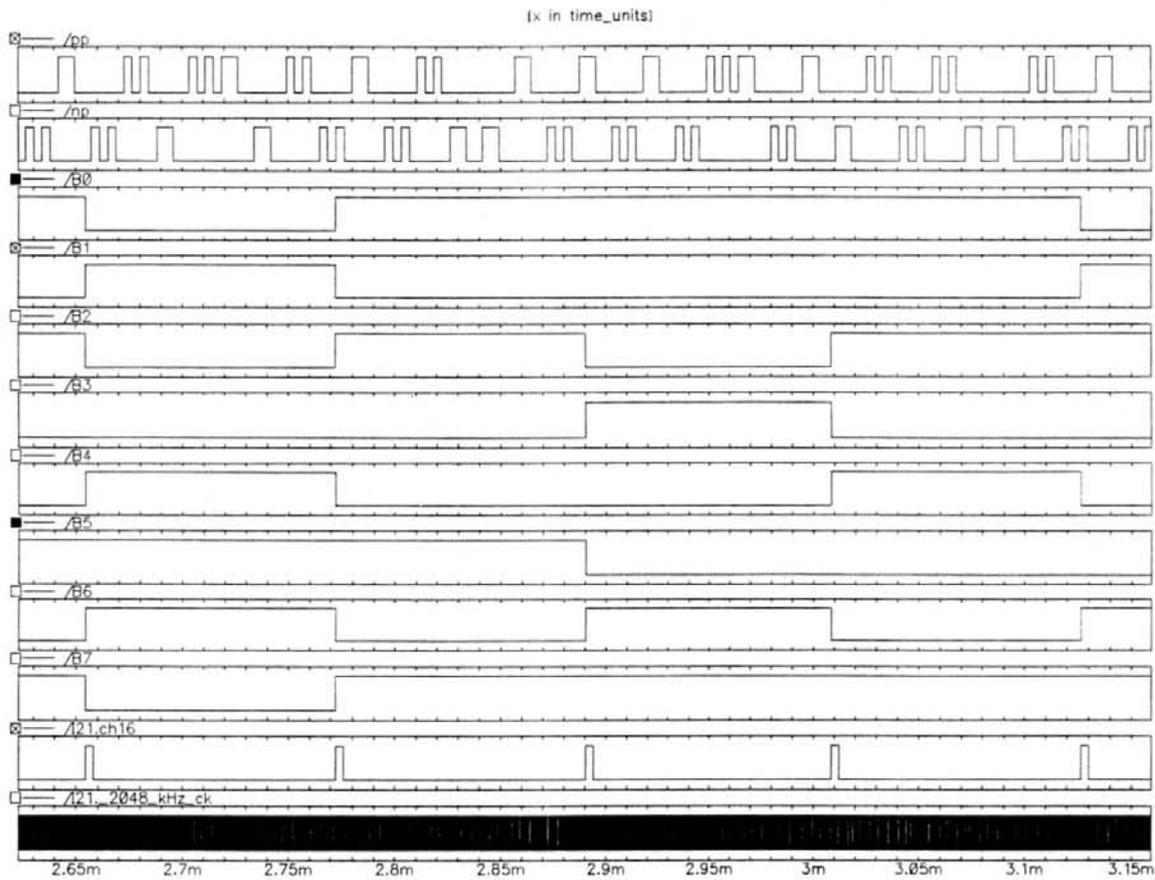


Figura 5.25 - Saída do *buffer elástico* a partir do sinal codificado.

flancos ascendentes serializam o octeto em paralelo formado pelos bits  $B0$  a  $B7$ , no período em que o *time-slot* está ativo. O octeto assim serializado, é disponibilizado na via de agregado, sendo após codificado e por fim transmitido ao equipamento demultiplexador remoto.

### 5.2.2 Codificação *HDB-3*

Na figura 5.27 pode ser visto o diagrama funcional do módulo de codificação *HDB-3*. Este módulo foi desenvolvido inicialmente para aplicação em um *chip-set* para modens banda base [AIT 94], sendo re-utilizado neste trabalho, uma vez que sua arquitetura encontra-se já validada. Os circuitos que compõem o módulo codificador *HDB-3*, encontram-se no apêndice deste trabalho.

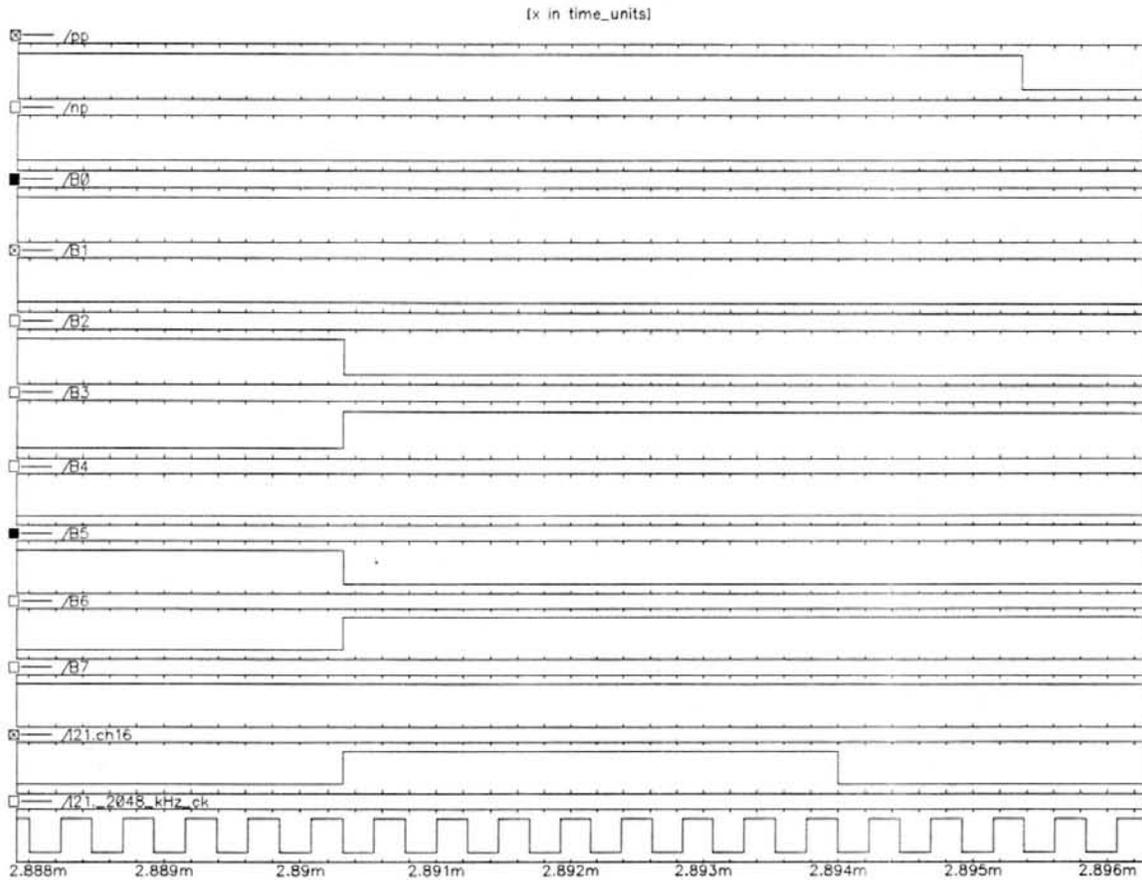


Figura 5.26 - Saída do *buffer elástico* a partir do sinal codificado; vista em detalhe.

Os dados a 2048 kbit/s provenientes da via de agregado, são disponibilizados na entrada *td* do codificador, sendo avaliados nas transições descendentes do sinal de relógio de 2048 kHz (entrada *clk master*). A entrada *reset* inicializa os *flip-flops* internos do módulo, sendo necessária apenas para a simulação. A entrada *ami* possibilita opcionalmente, a transmissão dos dados a 2048 kbit/s, segundo o código *ami* (*alternate mark inversion*). Este recurso entretanto, não é necessário para a transmissão dos quadros *E1*, uma vez que para o multiplexador *E1*, o código recomendado na interface *G.703* é o *HDB-3*.

Os dados codificados são disponibilizados nas saídas *tp* e *tn*. Na figura 5.28 pode ser vista a simulação do codificador. Note-se que para a codificação *ami* (entrada *ami* ativada, em nível lógico *um*), não há substituição de bits para seqüências com mais de três *zeros*. Entretanto, quando seleciona-se a codificação



Figura 5.27 - Módulo de codificação *HDB-3*.

*HDB-3* (entrada *ami* desativada), as seqüências com mais de três bits *zero* são substituídas pelas seqüências *000V* ou *B00V*.

Os sinais *tp* e *tn* na saída do codificador, são a seguir levados ao circuito *driver* de linha e transmitidos na forma bipolar, ao equipamento demultiplexador remoto.

### 5.3 Módulos para Demultiplexação

A demultiplexação consiste na separação dos 32 octetos que formam os quadros *E1* e posterior transmissão dos octetos de tributário aos seus respectivos canais. A figura 5.29 mostra um diagrama de interconexão dos módulos funcionais que tomam parte na demultiplexação.

Os sinais *pp* e *pn* constituem o sinal codificado de linha regenerado e unipolarizado. Estes sinais são fornecidos por um circuito analógico de regeneração que amplifica e reconstrói o sinal degradado pelo meio de transmissão. O circuito regenerador, por ser implementado à base de técnicas analógicas, não será especificado neste trabalho.

O circuito recuperador de relógio fornece sinais de relógio que traduzem a temporização do sinal codificado de recepção. Estes sinais são empregados para decodificação *HDB-3*, escrita de dados no *buffer elástico*, podendo ainda um destes sinais ser utilizado como referência para a geração de sinais de relógio para a

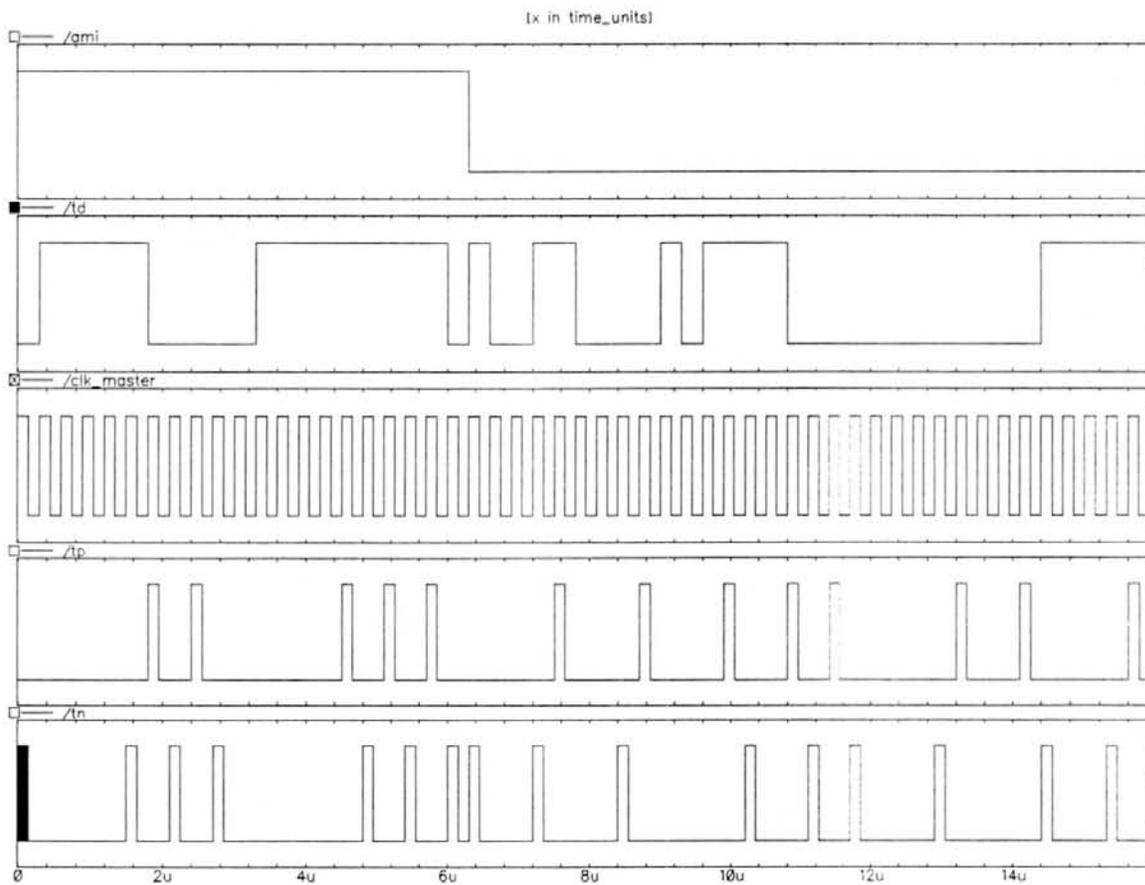


Figura 5.28 - Simulação do codificador *HDB-3*.

temporização do sistema *mux/demux* local. A geração de sinais de relógio para a temporização do sistema *mux/demux* local é realizada pelo módulo gerador de base de tempo.

Na saída do *buffer elástico* (via de agregado), a temporização do agregado é síncrona ao sinal de relógio do sistema *mux/demux*, podendo a partir daí iniciar o processo de alinhamento de quadro, o que é feito no módulo de alinhamento de quadro.

Tendo sido obtido o alinhamento de quadro, o módulo gerador de *time-slots* é inicializado, fornecendo sinais para habilitação dos módulos de transmissão de 64 kbit/s. Assim, os *time-slots* são gerados sincronamente à temporização de octeto, possibilitando a desmontagem dos quadros *E1* e a transmissão dos octetos a seus respectivos canais tributários (módulos *TX1* a *TX31*).

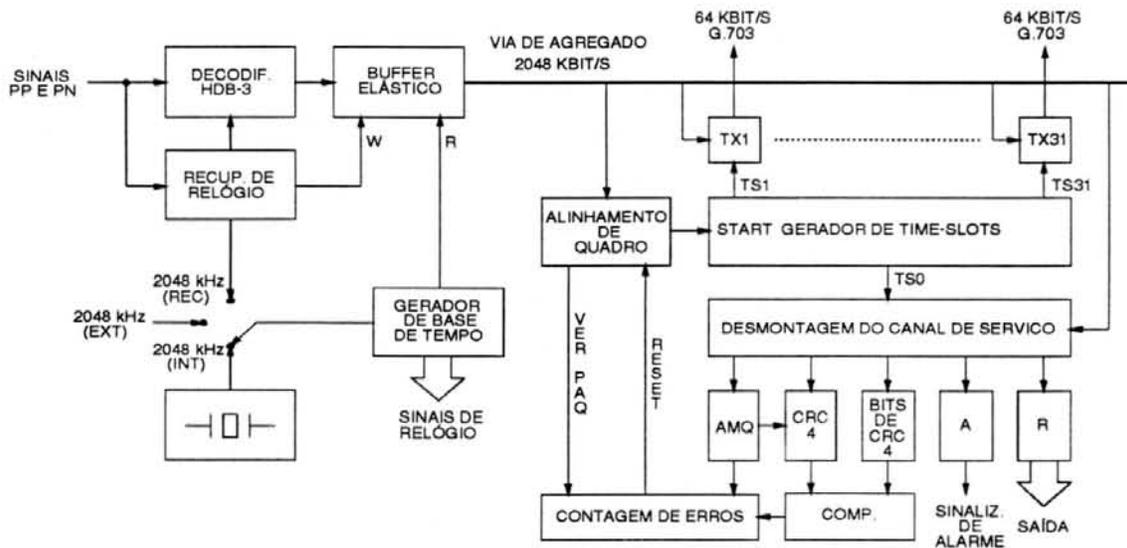


Figura 5.29 - Diagrama de blocos funcionais para demultiplexação.

Durante o *time-slot* relativo ao canal de serviço, inicia-se também após o alinhamento de quadro, o procedimento para alinhamento de multiquadro *CRC-4* (módulo *AMQ*). Tendo sido obtido o alinhamento de multiquadro, inicia-se o procedimento de verificação cíclica *CRC-4*, sobre os quadros *E1*. Para tanto, os bits de *CRC-4* são separados e comparados (módulo *COMP*) com os bits obtidos pelo verificador *CRC-4* (módulo *CRC-4*).

O módulo de *contagem de erros* avalia periodicamente o padrão de alinhamento de quadro, o padrão de alinhamento de multiquadro e a taxa de erros do verificador *CRC-4*, podendo forçar o procedimento de alinhamento de quadro se necessário. Os bits de reserva são encaminhados à porta de saída do canal interno de 20 kbit/s. O bit de alarme é disponibilizado para sinalização de manutenção.

A seguir serão descritos os módulos funcionais propostos neste trabalho, relativos à demultiplexação.

### 5.3.1 Recepção do Sinal Codificado de 2048 kbit/s

A recepção do sinal multiplexado de 2048 kbit/s, proveniente do equipamento multiplexador remoto é realizada pelos circuitos regenerador de linha, recuperador de relógio, decodificador *HDB-3* e *buffer elástico*.

Devido aos efeitos do *jitter*, a temporização do sinal proveniente do sistema *mux/demux* remoto é mesócrona em relação à temporização gerada pelo sistema *mux/demux* local. Em uma situação de pior caso, devido à ausência de função de sincronismo de rede, a temporização dos sistemas será plesiócrona.

Na saída do *buffer elástico* contudo, o agregado será síncrono à temporização local do sistema, estando apto a ser demultiplexado para que os octetos de tributário possam ser transmitidos a seus respectivos canais.

#### 5.3.1.1 Recuperação de Relógio de 2048 kHz

A recuperação do sinal de relógio de 2048 kHz, a partir do sinal codificado de linha, deverá ser realizada por meio de circuitos analógicos. Uma implementação digital para este circuito cerca-se de dificuldades tais quais aquelas já identificadas na seção 5.1. Desta forma, especificar-se-á simplesmente os sinais de entrada para o circuito de recuperação de relógio, bem como os sinais de saída que deverão ser fornecidos por este circuito.

A figura 5.30 mostra o diagrama funcional do circuito de recuperação de relógio de 2048 kHz, bem como formas de onda dos sinais de entrada e saída. O sinal bipolar *HDB-3* proveniente da linha é regenerado e convertido nos sinais unipolares *PP* e *NP*. Na saída do circuito recuperador de relógio, o sinal *P0* é um sinal com frequência de 2048 kHz (fase principal), sendo que *P1* (fase auxiliar) é um sinal de mesma frequência, atrasado entretanto de 90 graus em relação a *P0*.

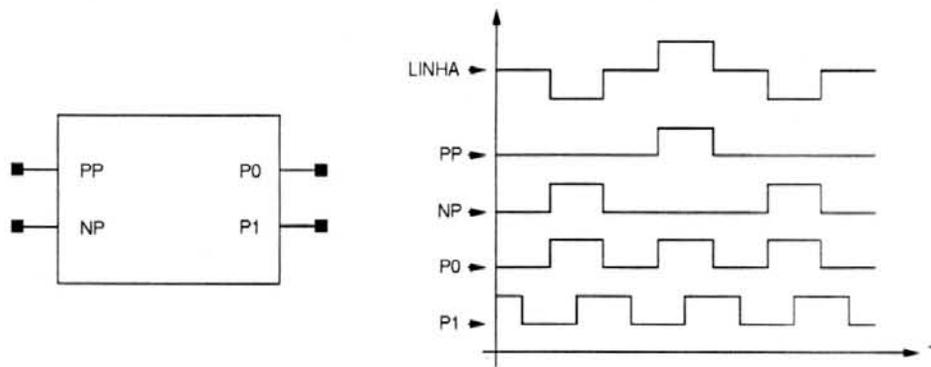


Figura 5.30 - Diagrama funcional e formas de onda para o circuito de recuperação de relógio de 2048 kHz.

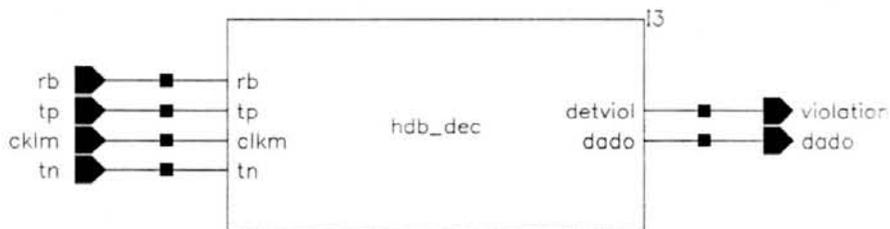


Figura 5.31 - Módulo de decodificação *HDB-3*.

### 5.3.1.2 Decodificação *HDB-3*

Tendo sido recuperado o sinal de relógio, é possível realizar a decodificação do sinal de linha *HDB-3*, proveniente do módulo regenerador. Esta função é executada pelo módulo decodificador *HDB-3*, cujo diagrama funcional pode ser visto na figura 5.31.

Igualmente ao módulo codificador *HDB-3*, o decodificador mostrado na figura 5.31 foi inicialmente especificado para implementação de um *chip-set* para modens banda base de 2048 kbit/s [AIT 94]. Por ter sido validada sua arquitetura através de intensa rotina de simulações, este módulo será re-utilizado neste trabalho, como parte dos módulos para implementação das funções de multiplexação *E1*.

Os sinais *tp* e *tn* na entrada do módulo decodificador, são os impulsos unipolarizados do sinal codificado de linha, fornecidos pelo circuito regenerador. O sinal *cklm* é o sinal de relógio de 2048 kHz fornecido pelo circuito de recuperação de



### 5.3.1.3 *Buffer Elástico* de Agregado

A necessidade do emprego de armazenamento elástico para o agregado repousa no fato de que é preciso acomodar os níveis de *jitter* previstos na recomendação *G.823*, tornando o agregado de recepção síncrono à temporização do equipamento demultiplexador. Adicionalmente, o armazenamento elástico para o agregado possibilita a implementação de enlaces plesiócronicos.

Assim, neste trabalho é proposta e simulada a arquitetura de um *buffer elástico*, também para o agregado. Esta arquitetura pode ser vista na figura 5.33, na forma de diagrama de blocos. No apêndice deste trabalho, encontram-se os sub-circuitos que compõem internamente os blocos que implementam o *buffer elástico* apresentado.

O *buffer elástico* mostrado na figura 5.33 é formado por um conjunto de 528 células de armazenamento (módulo *528.bit.cell*), um gerador de endereços de leitura (módulo *rd.add.gen*), um gerador de endereços de escrita (módulo *wr.add.gen*) e um circuito de controle lógico (módulo *logic*).

O módulo *528.bit.cell* é formado por 528 células de armazenagem, cada uma das quais capaz de armazenar um bit de dados. A escrita e leitura de dados sobre este módulo pode ser feita independentemente ou seja, é possível estar ocorrendo a leitura de um bit de uma célula simultaneamente à escrita de um bit em outra célula. Para tanto, existem dez linhas de endereçamento de escrita ( $a_0$  a  $a_9$ ) e dez linhas de endereçamento de leitura ( $b_0$  a  $b_9$ ). Os endereços de escrita (linhas  $a_0$  a  $a_9$ ) são fornecidos pelo módulo gerador de endereços de escrita e os endereços de leitura (linhas  $b_0$  a  $b_9$ ) são fornecidos pelo gerador de endereços de leitura.

O módulo de controle lógico desempenha duas funções:

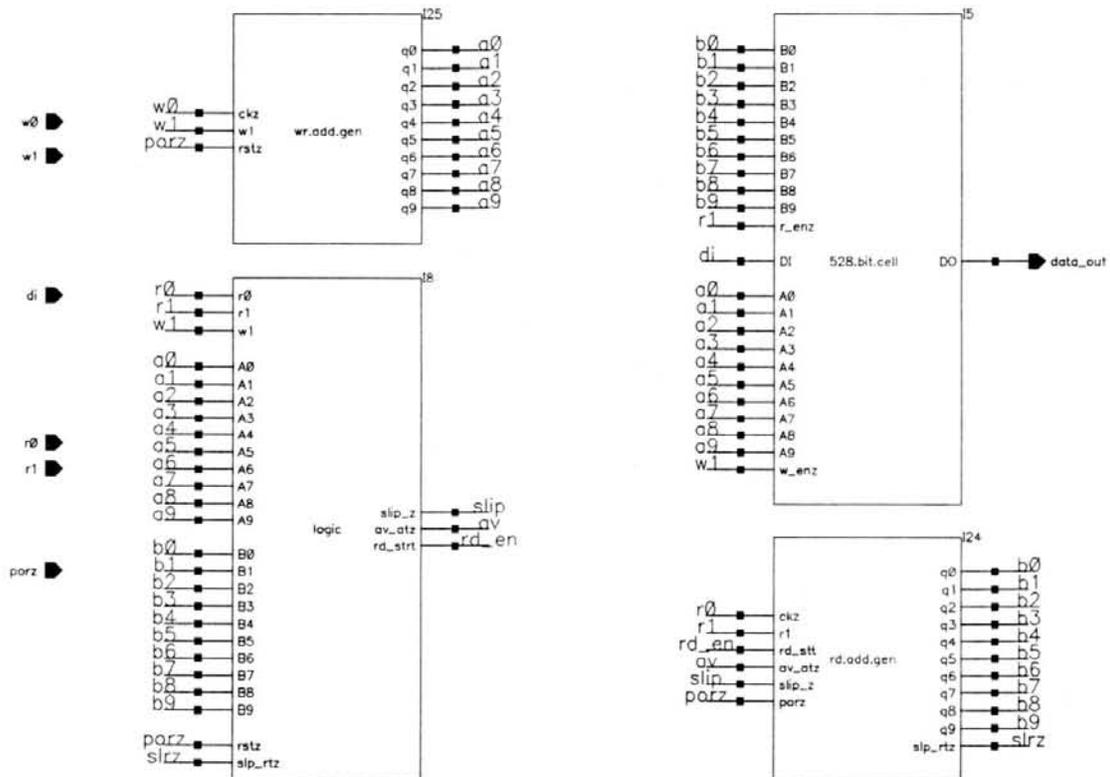


Figura 5.33 - Buffer elástico para o agregado.

- Manter o gerador de endereços de leitura inicializado (saídas  $b_0$  a  $b_9$  em nível lógico *zero*), até que o módulo *528.bit.cell* tenha sido preenchido com 264 bits (metade de sua capacidade).
- Forçar a ocorrência de *slip*, quando a diferença entre os endereços de escrita e leitura for menor que 8 bits.

O funcionamento do *buffer elástico* proposto será a seguir descrito. Durante o *start-up* do sistema (energização dos circuitos), o sinal *porz* (*power-on-reset*, ativo em nível lógico *zero*) inicializa os geradores de endereços de escrita e de leitura (linhas de endereçamento  $a_0$  a  $a_9$  e  $b_0$  a  $b_9$  em nível lógico *zero*). O circuito de controle lógico também é inicializado, tornando as saídas *slip*, *av* e *rd en* desativadas. Estas são portanto, as condições de partida do dispositivo.

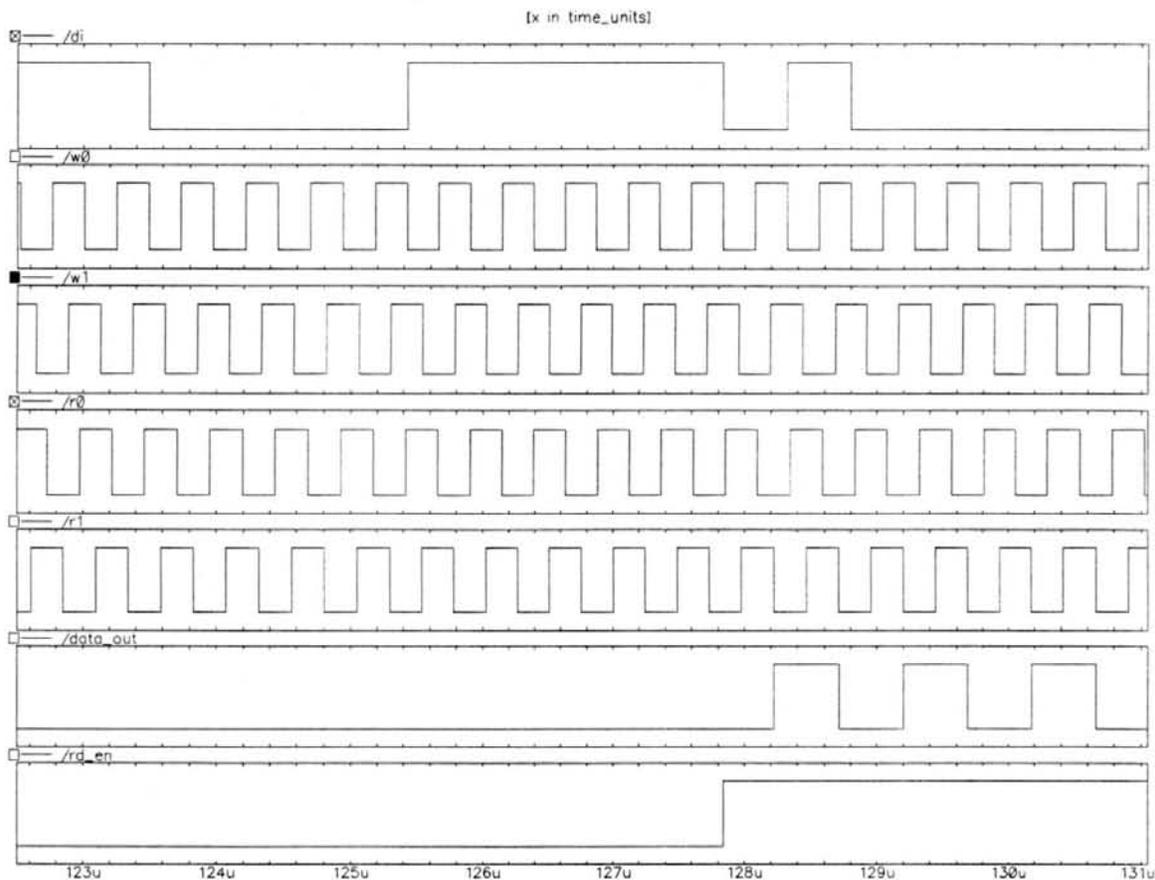


Figura 5.34 - Sinais de relógio, entrada e saída para o *buffer elástico*.

Além do sinal *porz*, o *buffer elástico* necessita outros cinco sinais de entrada, quais sejam *di*, *w0*, *w1*, *r0* e *r1*, sendo *di* o agregado de entrada (assíncrono) que será armazenado bit a bit, em sucessivas células do módulo *528.bit.cell*. Os sinais *w0* e *w1* são sinais de relógio de escrita, sendo *w0* a fase principal e *w1* a fase auxiliar. Os sinais *w0* e *w1* correspondem respectivamente aos sinais *P0* e *P1* fornecidos pelo circuito de recuperação de relógio de 2048 kHz (seção 5.3.3.1). Os sinais *r0* e *r1* são sinais de relógio para leitura, sendo *r0* a fase principal e *r1* a fase auxiliar. Os sinais *r0* e *r1* correspondem respectivamente aos sinais *R0* e *R1* fornecidos pelo circuito gerador de base de tempo descrito na seção 5.1.

Todos os quatro sinais de relógio apresentam frequências nominais de 2048 kHz, sendo que as fases auxiliares (dos relógios de escrita ou de leitura) estão atrasadas de 90 graus em relação às respectivas fases principais. Quanto à temporização, tomando-se as fases principais (*w0*, escrita e *r0*, leitura) dos sinais de

relógio, se ambas apresentarem a mesma frequência média, a temporização do sistema será mesócrona, sendo que se existir um *offset* entre suas frequências nominais, a temporização será plesiócrona. A figura 5.34 mostra os sinais de relógio e os sinais de entrada e saída para o diagrama de blocos mostrado na figura 5.33. Esta figura foi obtida por simulação da arquitetura proposta.

De acordo com a figura, os flancos descendentes da fase principal do sinal de relógio de escrita ( $w0$ ) coincidem com o início do período de bit do agregado de entrada ( $di$ ). O nível lógico *zero* da fase auxiliar de escrita  $w1$  está localizado exatamente no centro do período de bit do agregado de entrada. O endereço para escrita de um bit em uma célula de armazenagem é fornecido pelo gerador de endereços de escrita no flanco descendente da fase principal  $w0$  do relógio de escrita. O bit é efetivamente escrito na célula endereçada, no nível lógico *zero* da fase auxiliar  $w1$ , através da ativação do sinal  $w\ enz$ , no módulo *528.bit.cell*. Para a leitura de um bit do *buffer*, o procedimento é similar. Nos flancos descendentes da fase principal  $r0$  do relógio de leitura, o gerador de endereços de leitura fornece um endereço a partir do qual um bit será lido. O nível lógico *zero* da fase auxiliar  $r1$  efetiva a operação de leitura, através da ativação do sinal  $r\ enz$  no módulo *528.bit.cell*, disponibilizando o bit lido na saída deste dispositivo.

Após o período de *start-up* do sistema, o agregado de entrada começa a ser escrito bit a bit nas células de armazenamento, endereçadas pelo gerador de endereços de escrita. O módulo de controle lógico habilita a geração de endereços de leitura, através da ativação do sinal  $rd\ en$ , tão logo tenham sido armazenados 264 bits do agregado de entrada nas células de armazenagem. Isto significa que a operação de leitura somente inicia após ter sido estabelecida uma ocupância equivalente à metade da capacidade de armazenamento do *buffer elástico*. Na figura 5.34, pode ser visto o instante de ativação do sinal  $rd\ en$  (próximo a 128  $\mu s$ ), habilitando a geração dos endereços de leitura. Nestas condições, o retardo imposto ao agregado desde sua entrada até sua saída (latência do sistema) é de um quadro (256 bits) mais 8 bits. No caso da temporização do sistema ser mesócrona, a ocupância média manter-

se-á neste valor (264 bits). Para temporização plesiócrona contudo, a ocupância poderá aumentar (escrita mais rápida que leitura) ou diminuir (leitura mais rápida que escrita). Para qualquer um destes casos, o módulo de controle lógico força a ocorrência de um *slip* sempre que a ocupância for tal que a diferença entre os endereços de escrita e de leitura seja menor que 8 bits. É portanto necessário avaliar periodicamente o comportamento da ocupância, o que é realizado conforme descrito a seguir.

O módulo de controle lógico possui internamente um contador (se necessário, pode ser visto no apêndice deste trabalho), o qual é inicializado sempre que o endereço de escrita for o binário **00 0000 0000**. Para tanto, o módulo de controle monitora o conteúdo binário das linhas de endereçamento de escrita ( $a_0$  a  $a_9$ ), ativando (nível lógico zero) um sinal denominado  $wad0$ , que inicializa o contador. O sinal de relógio para este contador é a fase principal  $r0$  do relógio de leitura, sendo que sua saída é avaliada sempre que o gerador de endereços de leitura apresentar o endereço binário **00 0000 0000**. Para isto, o módulo de controle lógico monitora as saídas ( $b_0$  a  $b_9$ ), ativando (nível lógico zero) um sinal denominado  $rad0$ . Com este arranjo, o conteúdo presente à saída do contador, avaliado no instante em que o sinal  $rad0$  está ativo, é a própria ocupância do *buffer*.

Assim, para temporização plesiócrona duas situações são passíveis de ocorrência, as quais se descreve a seguir.

- Caso 1: *Slip* por *overflow*.

Supondo-se que a frequência do sinal de relógio de escrita seja ligeiramente superior à frequência do sinal de relógio de leitura, o período do sinal  $wad0$  será menor que o período do sinal  $rad0$ . Com isto, o conteúdo binário a ser avaliado à saída do contador, mostrar-se-á cada vez maior à medida em que o tempo passa. Em outras palavras, a ocupância do *buffer* aumenta sistematicamente. Quando esta ocupância atingir um valor maior que 520 bits, a saída *slip* do módulo de controle lógico torna-se ativa, sendo sinalizada esta condição ao módulo gerador de endereços

de leitura. Neste módulo, cujas saídas encontram-se todas em nível lógico *zero*, a saída  $b_8$  é setada. Com isto, o endereço de leitura que antes era **00 0000 0000**, passa a ser **01 0000 0000** ou seja, devido à ativação do sinal *slip* no módulo de controle lógico, o endereço de leitura sofreu efetivamente um salto de 256 posições à frente ou seja, um quadro básico *E1* inteiro deixa de ser lido, iniciando-se a leitura no quadro seguinte.

- Caso 2: *Slip* por *underflow*.

Supondo-se agora que a frequência da fase principal do sinal de relógio de escrita seja ligeiramente inferior à frequência da fase principal do sinal de relógio de leitura, o período do sinal *wad0* será maior que o período do sinal *rad0*. Assim, com o decorrer do tempo, o conteúdo binário avaliado na saída do contador, quando *rad0* estiver ativo diminuirá ou seja, a ocupância será sistematicamente reduzida a partir do valor inicial de 264 bits. Quando seu valor tornar-se menor que 8 bits, as saídas *slip* e *av* do módulo de controle lógico tornar-se-ão ativas, sendo esta condição sinalizada ao gerador de endereços de leitura. Neste módulo, cujas saídas encontram-se todas em nível lógico *zero*, as saídas  $b_8$  e  $b_4$  são setadas. Com isto, o endereço de leitura que antes era **00 0000 0000**, passará a ser **01 0001 0000** ou seja, o endereço de leitura sofrerá um salto de 272 posições à frente. Um salto de 272 posições à frente é equivalente a um salto de 256 posições para trás. Assim, o endereço de leitura sofreu efetivamente um retardo equivalente a 256 bits ou seja, um quadro básico *E1* inteiro será lido novamente do *buffer*.

Em qualquer um dos casos descritos acima, pode-se notar que o afastamento entre os endereços de escrita e de leitura nunca poderá ser menor que 8 bits. A manutenção deste afastamento mínimo é necessária devido ao fato de que o sinal proveniente do equipamento remoto pode estar (como geralmente ocorre) contaminado por *jitter*. Com o *jitter*, existem variações instantâneas na diferença entre os endereços de escrita e leitura e, se não fosse mantido um afastamento mínimo entre estes endereços, poderia ocorrer escrita e leitura simultâneas sobre uma mesma

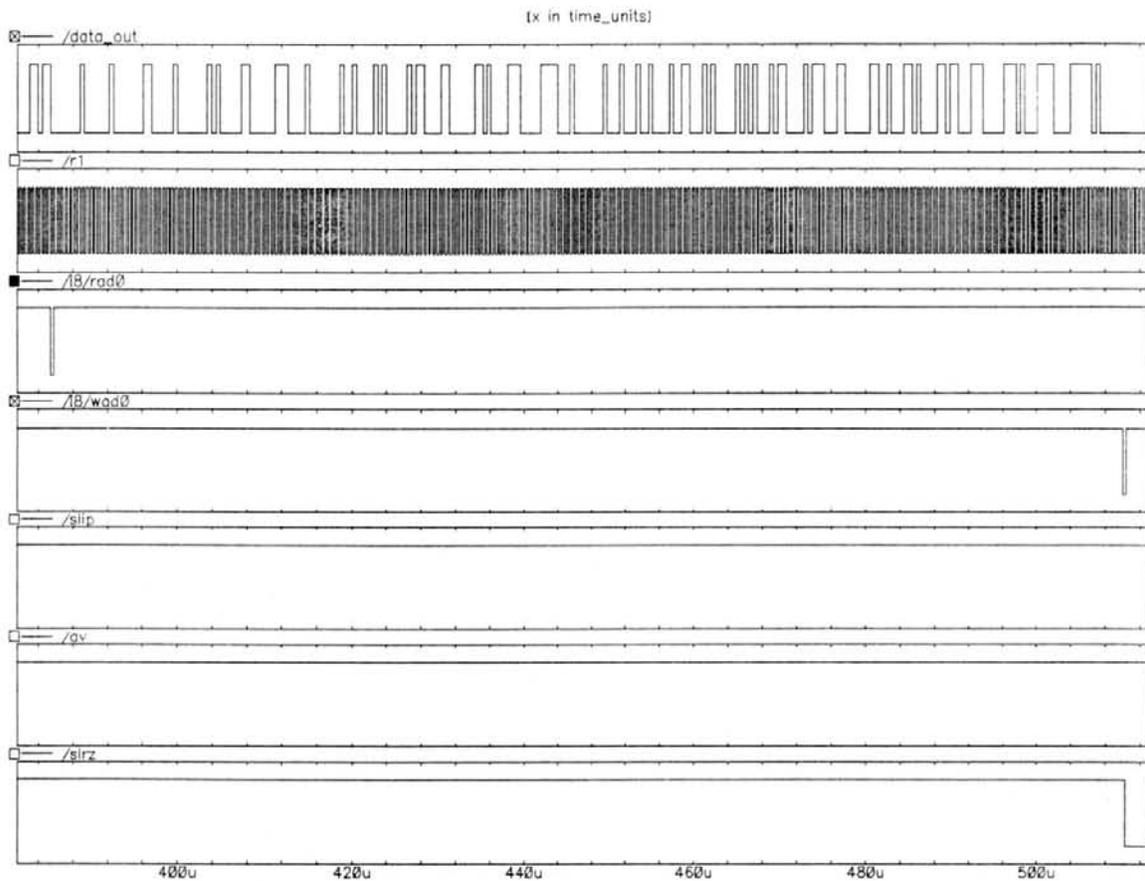


Figura 5.35 - Funcionamento com temporização mesócrona

posição (célula) de armazenamento, o que seria inadmissível. Assim, quando a diferença entre os endereços de escrita e leitura atinge o limite de 8 bits, um *slip* é forçado, deslocando-se o endereço de leitura de um quadro para frente ou de um quadro para trás. Este tipo de *slip* é denominado *slip controlado*, acerca do qual tecer-se-á alguns comentários a seguir.

No *slip* controlado não há necessidade de proceder-se ao re-alinhamento de quadro, uma vez que a supressão ou repetição na leitura de bits, decorrentes do *slip*, é realizada na forma de um quadro completo (leitura suprimida ou repetida de um quadro inteiro). De qualquer modo, a ocorrência de um *slip* deve ser levada em consideração, quando da avaliação da manutenção do sincronismo de quadro. Esta avaliação é realizada através da comprovação do padrão de alinhamento de quadro em quadros alternados, durante o *time-slot* correspondente ao canal de serviço (*time-slot* 0). Ora, se devido à ocorrência de um *slip* houve o deslocamento de um quadro

completo, a verificação do padrão de alinhamento de quadro passaria impropriamente a ser feita justamente nos quadros que não contêm o padrão de alinhamento de quadro, levando o sistema a forçar o procedimento de alinhamento de quadro, caracterizando assim um *slip não-controlado*. Desta forma, sempre que ocorrer um *slip*, deve ser realizada a correção acerca da qualificação dos quadros a partir dos quais serão buscados e testados os bits referentes ao padrão de alinhamento de quadro, de forma a evitar *slips* não-controlados.

Ainda que seja possível a ocorrência de *slips* sem que seja necessário proceder o re-alinhamento de quadro, o mesmo não é possível com relação ao sincronismo de multiquadro. A ocorrência de um *slip* sempre irá forçar a re-sincronização de multiquadro, sendo esta situação contabilizada como uma condição de erro. Entretanto, os erros associados ao multiquadro (perda de sincronismo de multiquadro e alta taxa de bits *CRC-4* recebidos com erro) somente provocam o re-alinhamento de quadro, quando contabilizados em quantidades expressivas, o que seguramente não é o caso da ocorrência de um simples *slip* em um enlace plesiócrono.

Tendo sido brevemente descrito o funcionamento do *buffer elástico* de agregado proposto neste trabalho, passa-se a seguir a apresentação dos resultados das simulações desenvolvidas.

A figura 5.35 mostra a simulação do funcionamento do sistema para temporização mesócrona. A ocupância do *buffer*, determinada pelo intervalo de tempo entre a ativação (em nível lógico *zero*) dos sinais *wad0* e *rad0*, está em torno de  $125 \mu\text{s}$  ou seja, praticamente a metade da capacidade total de armazenagem do dispositivo.

Para temporização plesiócrona, a simulação foi realizada a partir da definição de freqüências de relógio de escrita e leitura ligeiramente diferentes, quando da especificação dos vetores de teste. Cabe salientar entretanto, que o *offset* entre os relógios de escrita e leitura foi sobre-dimensionado, de forma a possibilitar a verificação da ocorrência de *slips* em um tempo exeqüível de processamento de simulação. Assim, definiu-se um *offset* da ordem de  $10^{-2}$ , a partir do qual espera-se a

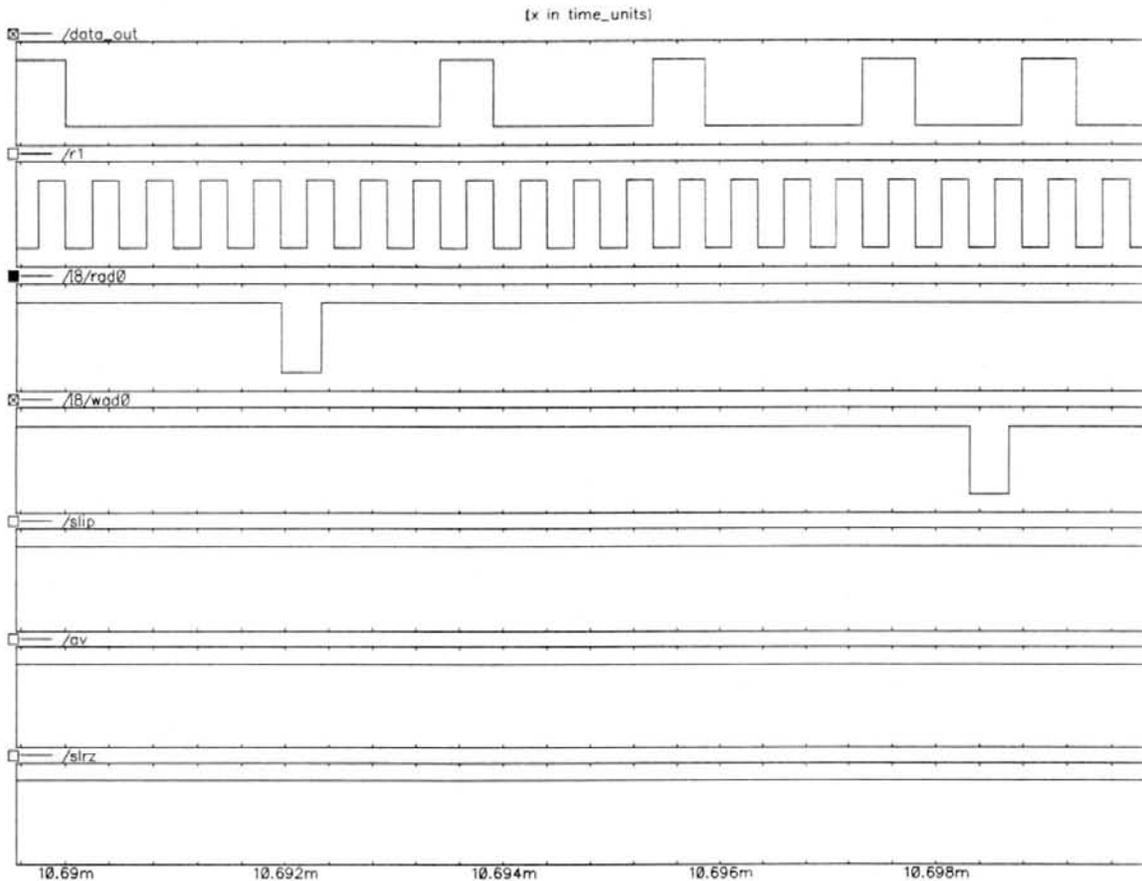


Figura 5.36 - Proximidade dos endereços de escrita e leitura.

ocorrência de um *slip* em torno de 10 ms após a habilitação do gerador de endereços de leitura (ativação do sinal *rd en*). Contudo, saliente-se ainda uma vez, que para operação real do sistema é necessário um *offset* de  $1,73 \cdot 10^{-9}$ , resultando em um *slip* a cada 20 horas.

A fim de verificar o correto funcionamento do *buffer elástico*, os octetos do agregado de entrada foram numerados, utilizando-se os cinco últimos bits de cada octeto, mantendo-se os três primeiros em nível lógico zero. Desta forma, os quadros iniciam com o conteúdo binário **00000001** para o octeto correspondente ao *time-slot* 1 e terminam com o conteúdo binário **00011111**, para o octeto correspondente ao *time-slot* 31. O conteúdo binário dos octetos para o *time-slot* 0 (canal de serviço) foi fixado em **00011011** (padrão de alinhamento de quadro) nos quadros que contém o padrão de alinhamento de quadro e **01000000** (bit *b1* setado) nos quadros que não contém o padrão de alinhamento de quadro. Desta maneira, os octetos corres-

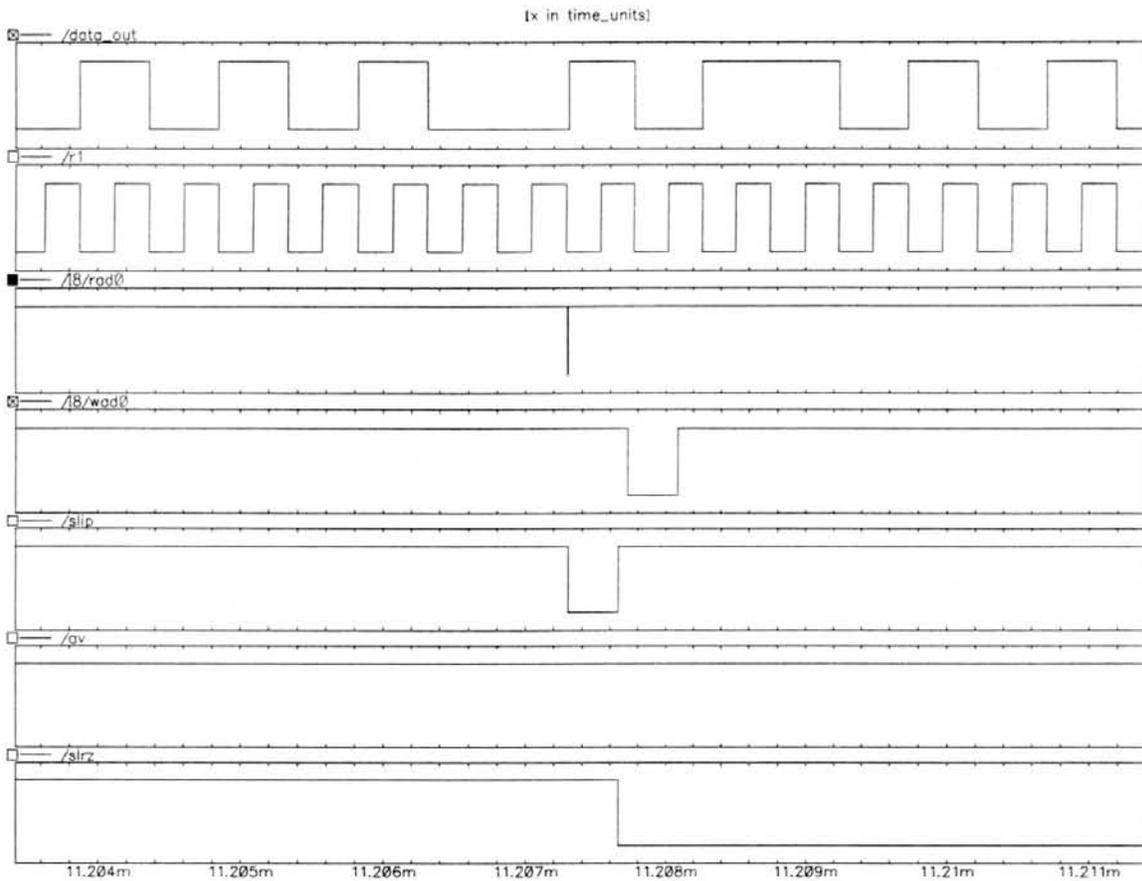


Figura 5.37 - *Slip por overflow.*

pondentes aos *time-slots* 20 e 21 por exemplo, de um quadro qualquer, apresentam conteúdos binários respectivamente de **000 10100** e **000 10101**.

Para a simulação da ocorrência de *slips*, foi utilizada a estratégia descrita a seguir. No último *time-slot* antes da ocorrência do *slip* (*time-slot* 20 do quadro 88), os três primeiros bits do octeto foram fixados em **010**; no *time-slot* 21 do quadro seguinte (quadro 89), os três primeiros bits do octeto foram fixados em **101**. Por fim, no quadro de número 87 (quadro anterior àquele em que ocorreu o *slip*), os três primeiros bits do octeto correspondente ao *time-slot* 21 foram fixados em **001**. Assim, como todos os demais octetos de todos os quadros que compõem o agregado de entrada estão com os três primeiros bits de seus octetos fixados em **000**, torna-se fácil a identificação dos octetos acima mencionados, quando da alteração do endereço de leitura provocada por um *slip*. Deste modo, se um *slip por overflow* ocorrer, tem-se como resultado na simulação, a seqüência de octetos **010 10100** e **101 10101**.

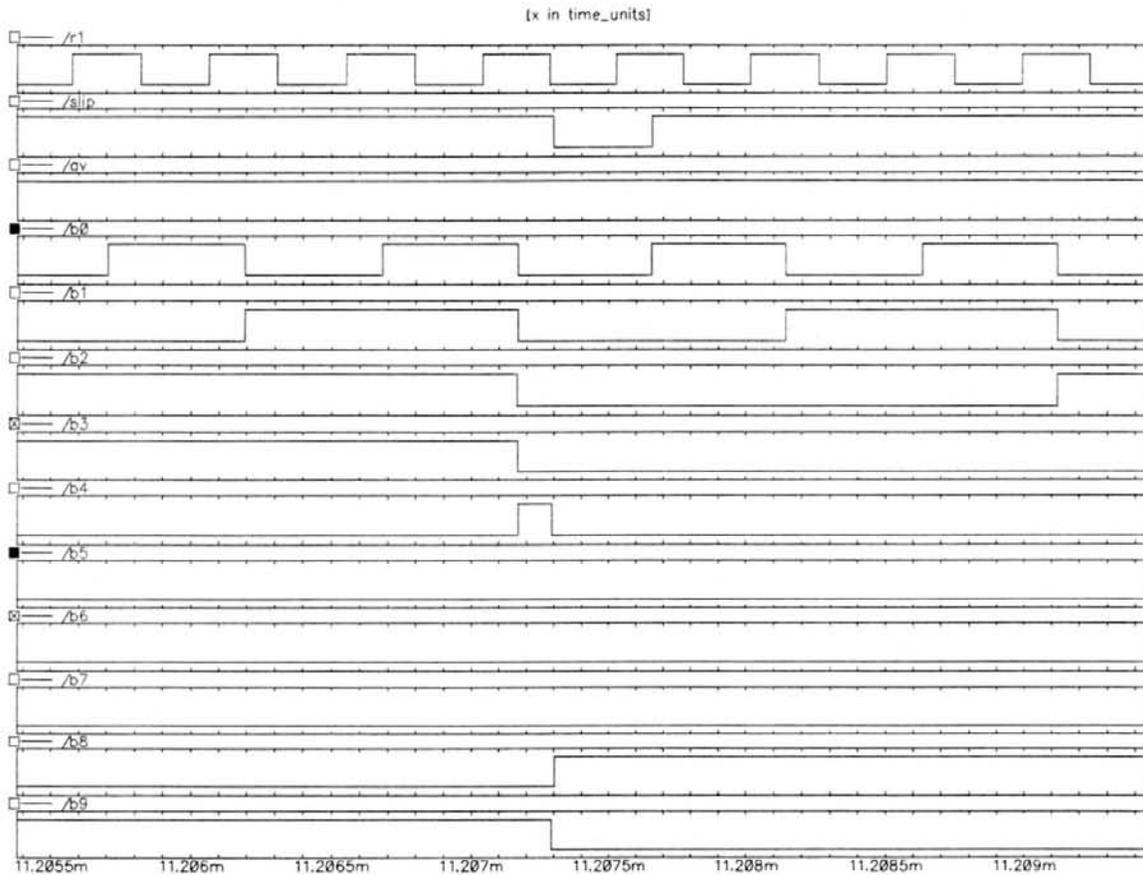


Figura 5.38 - Alteração do endereço de leitura (*slip* por *overflow*).

Analogamente, se um *slip* por *underflow* ocorrer, a simulação deve apresentar a seqüência **010 10100** e **001 10101**.

A figura 5.36 mostra a situação dos sinais *rad0* e *wad0*, sendo possível notar a existência de apenas 12 bits de diferença entre estes, sendo iminente a ocorrência de um *slip* por *overflow* nos instantes que seguem o período de tempo mostrado na figura. Esta figura contrasta com aquela para temporização mesócrona, onde pode ser visto um grande afastamento temporal entre os sinais *rad0* e *wad0*.

A figura 5.37 mostra o instante de ocorrência de um *slip* por *overflow*. Os oito primeiros bits da esquerda para a direita formam o octeto correspondente ao *time-slot* 20 do quadro 88, sendo que os oito bits restantes formam o octeto correspondente ao *time-slot* 21 do quadro 89, demonstrando portanto o correto funcionamento do dispositivo.

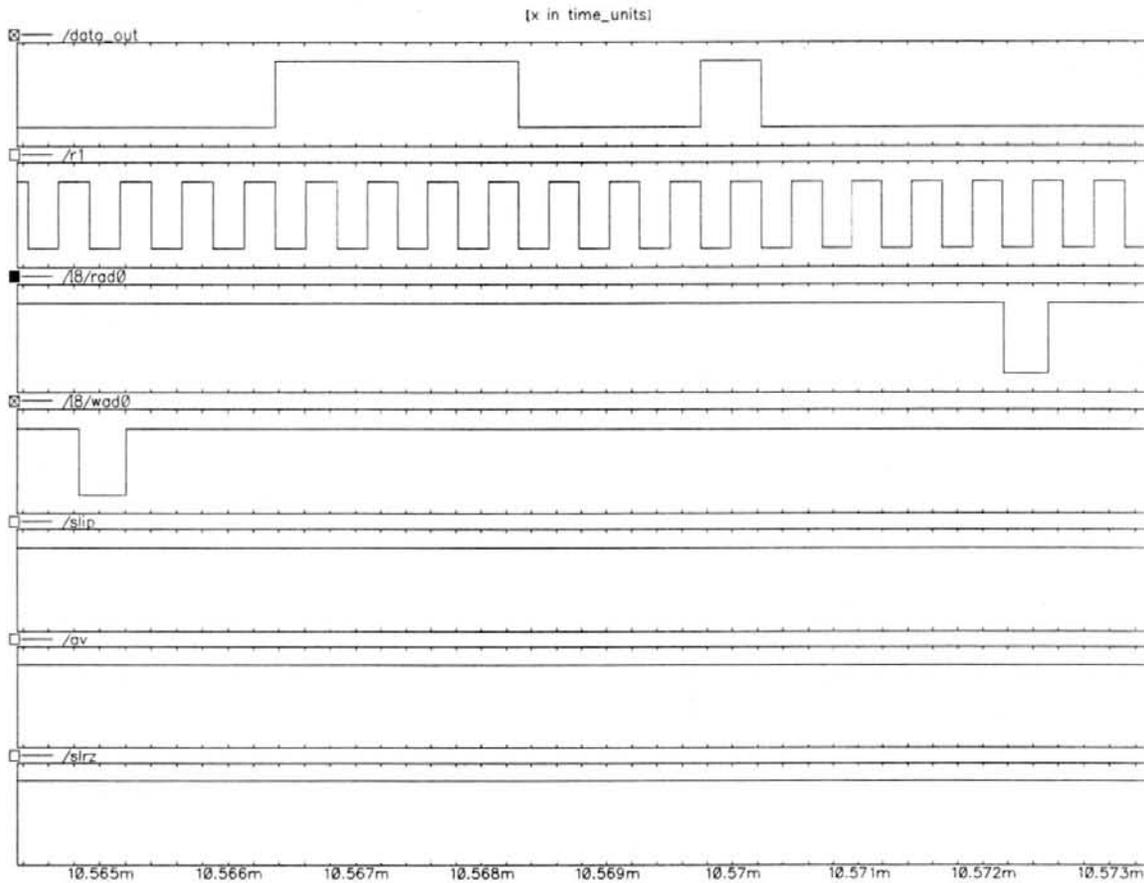


Figura 5.39 - Proximidade dos endereços de escrita e leitura.

A figura 5.38 mostra a saída (bits  $b_0$  a  $b_9$ ) do gerador de endereços de leitura, no momento em que o *slip* ocorre. O bit  $b_8$  é setado no flanco descendente do sinal *slip*, transferindo o endereço de leitura 256 bits à frente. Em condições normais, caso não ocorresse o *slip*, todos os bits  $b_0$  a  $b_9$  estariam resetados.

A figura 5.39 mostra os sinais *rad0* e *wad0*, nos instantes que precedem um *slip* por *underflow*. Aqui, apenas 14 bits separam os endereços de escrita dos de leitura, sendo que um *slip* por *underflow* é iminente nos instantes seguintes aos mostrados nesta figura.

Na figura 5.40, pode ser visto o *slip* por *underflow*. Os oito primeiros bits da esquerda para a direita formam o octeto correspondente ao *time-slot* 20 do quadro 88, sendo que os oito bits restantes formam o octeto correspondente ao *time-slot* 21 do quadro 87, mostrando portanto o retardo de 256 bits imposto ao gerador

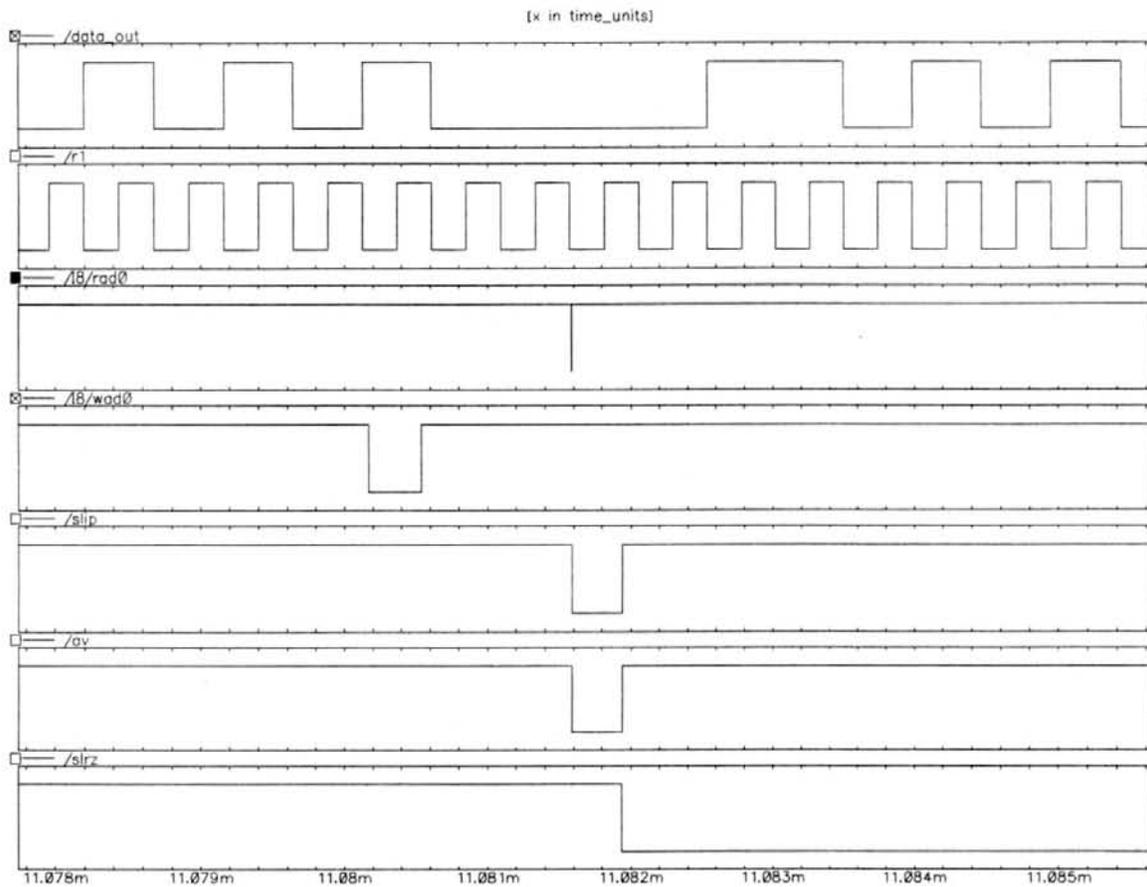


Figura 5.40 - Slip por *underflow*.

de endereços de leitura. Na figura 5.41 observa-se a alteração na saída do gerador de endereços de leitura, com os bits  $b_4$  e  $b_8$  setados no flanco descendente do sinal *slip*, transferindo o endereço de leitura 256 bits para trás. Em condições normais, caso não ocorresse o *slip*, todos os bits  $b_0$  a  $b_9$  estariam resetados.

Tendo sido apresentados os resultados da simulação do *buffer elástico* de agregado, é interessante tecer alguns comentários acerca da implementação do circuito e do processo de simulação propriamente dito.

O circuito completo deste *buffer elástico* para o agregado totalizou 24.415 transistores na forma de 6.103 *gates*. Na simulação de *slip* por *overflow*, o período dos sinais de relógio de escrita ( $w0$  e  $w1$ ) foi fixado em 482,4 ns, sendo que o período dos sinais de relógio de leitura foi fixado em 488 ns. Para simulação de *slip* por

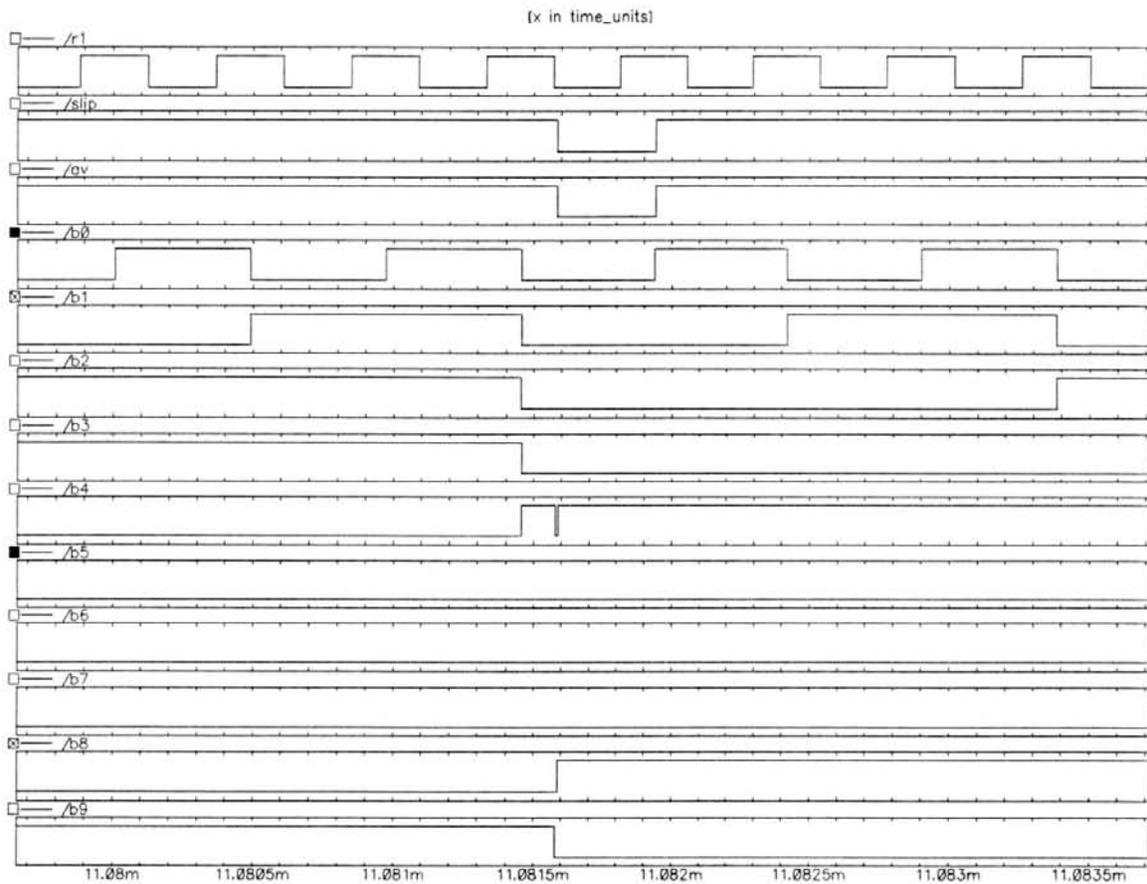


Figura 5.41 - Alteração do endereço de leitura (*slip* por *underflow*).

*underflow*, o período dos sinais de relógio de escrita ( $w0$  e  $w1$  foi fixado em 488 ns, sendo que o período dos sinais de relógio de leitura foi fixado em 482,4 ns.

Foram simulados 108 quadros básicos *E1* ou aproximadamente 13,5 ms em termos de operação real do dispositivo. Para tanto, foram gerados 27.650 vetores de teste. Quanto aos custos desta simulação em termos de recursos de *CAD*, a geração dos vetores de teste consumiu aproximadamente 5 minutos, ao passo que a simulação do circuito com os vetores de teste já gerados, consumiu aproximadamente 35 minutos. Tanto a geração dos vetores de teste quanto a simulação propriamente dita do circuito foram realizadas em *workstation SUN*, modelo *SPARC station 1+*.

### 5.3.2 Alinhamento de Quadro

Após serem submetidos ao armazenamento elástico, os quadros *E1* provenientes do equipamento multiplexador remoto estão já sincronizados à temporização do equipamento local. Com isto, é possível iniciar a demultiplexação propriamente dita, a partir da identificação dos quadros, o que é feito através da função de alinhamento de quadro.

Desta forma, a função de alinhamento de quadro é necessária durante o *start-up* do sistema *mux/demux*, ou por ocorrência de determinadas falhas que forcem o procedimento de re-alinhamento de quadro. O alinhamento de quadro é fundamental para o funcionamento do sistema, uma vez que é a partir deste que toda a temporização do sistema é inicializada.

De acordo com a recomendação *G.706*, a estratégia para alinhamento de quadro consiste em testar um conjunto de três condições em seqüência quais sejam, a ocorrência do padrão de alinhamento de quadro em um suposto quadro básico *E1*, seguindo-se a comprovação de um bit em nível lógico *um* na posição correspondente ao segundo bit (bit  $b_1$ ), no quadro seguinte e, como terceira condição de teste, verificar novamente a ocorrência do padrão de alinhamento de quadro, no próximo quadro. O circuito mostrado na figura 5.42 executa a função de alinhamento de quadro. Este circuito está dividido em cinco módulos, cujos diagramas esquemáticos podem ser vistos em detalhe, no apêndice deste trabalho.

Os módulos *step1*, *step2* e *step3* são sub-circuitos que testam as três condições de teste descritas acima. O módulo *counter* é utilizado para temporizar os instantes em que as condições de teste serão verificadas. O módulo *st.m* ativa a saída *in sinc*, quando as três condições necessárias ao alinhamento de quadro forem comprovadas. A seguir, o funcionamento do circuito de alinhamento de quadro será descrito.

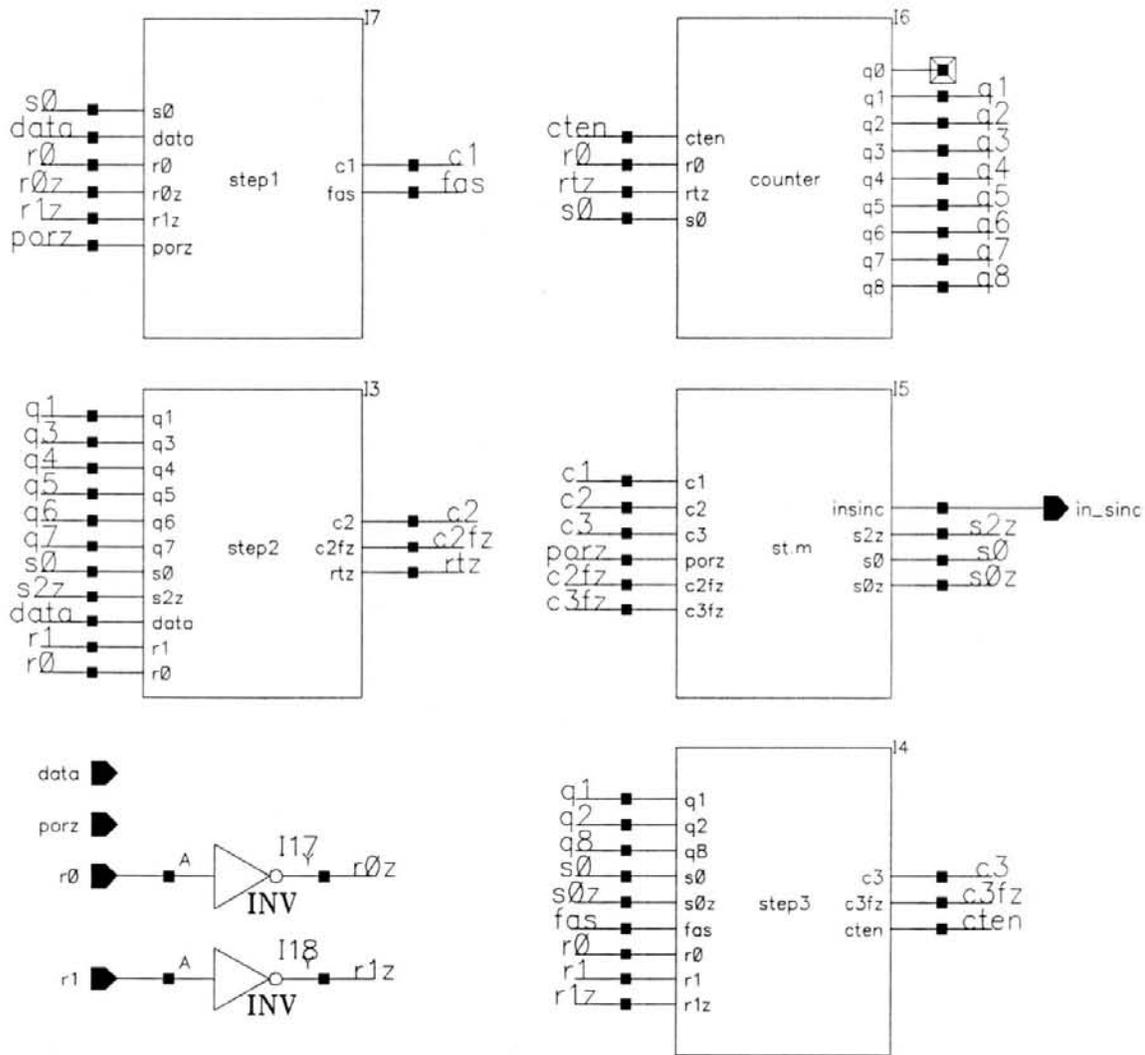


Figura 5.42 - Circuito para obtenção do alinhamento de quadro.

O módulo *st.m* é um contador *johnson* de quatro bits, apresentando portanto quatro estados possíveis:  $s0$ ,  $s1$ ,  $s2$  e  $s3$ . A ativação da entrada *porz* faz com que o estado  $s0$  seja o estado ativo. Uma transição ascendente em qualquer uma das entradas  $c1$ ,  $c2$  ou  $c3$ , faz com que seja ativado o estado seguinte. Transições ascendentes em qualquer uma das entradas  $c2fz$  ou  $c3fz$ , fazem com que o estado do módulo *st.m* passe a ser  $s0$ . Entretanto, a ativação de qualquer uma das entradas  $c1$ ,  $c2$ ,  $c3$ ,  $c2fz$  ou  $c3fz$  será inócua, caso  $s3$  seja o estado ativo do módulo *st.m*.

As transições ascendentes nas entradas  $c1$ ,  $c2$ ,  $c3$ ,  $c2fz$  e  $c3fz$  do módulo *st.m* são fornecidas pelos módulos *step1*, *step2* e *step3*, da maneira descrita a seguir.

Durante a ativação do estado  $s0$ , o módulo  $step1$  é habilitado e os dados a 2048 kbit/s são testados. Quando for detectada a seqüência correspondente ao padrão de alinhamento de quadro, a saída  $c1$  deste módulo é ativada (transição ascendente), provocando um avanço no estado do módulo  $st.m$ , que passa a ser o estado  $s1$ . Durante este estado, a segunda condição será testada, da maneira descrita abaixo.

O padrão de alinhamento de quadro é detectado no final do período de bit do oitavo bit ( $b_7$ ) de um suposto quadro que contém o padrão de alinhamento de quadro. Assim, para testar o segundo bit ( $b_1$ ) no quadro seguinte, o módulo  $step2$  é habilitado após 250 períodos de bit (contabilizados pelo módulo  $counter$ ), a partir da detecção do padrão de alinhamento de quadro. Assim que é habilitado, o módulo  $step2$  testa o bit presente na entrada  $data$  (dados a 2048 kbit/s). Se o bit testado estiver em nível lógico  $um$ , a saída  $c2$  é ativada (transição ascendente) e o estado ativo do módulo  $st.m$  passa a ser  $s2$ . Se o bit testado estiver em nível lógico  $zero$ , a saída  $c2fz$  do módulo  $step2$  é ativada (transição ascendente), fazendo com que o estado ativo do módulo  $st.m$  volte a ser  $s0$ .

Sendo  $s2$  o estado ativo do módulo  $st.m$ , o módulo  $step3$  é habilitado para testar a ocorrência do padrão de alinhamento de quadro, ao final do período do oitavo bit ( $b_7$ ), no quadro seguinte. Para tanto são contabilizados 262 períodos de bit a partir do segundo bit ( $b_1$ ) do quadro anterior, quando a entrada  $fas$  do módulo  $step3$  é avaliada. O sinal  $fas$ , proveniente do módulo  $step1$ , é ativado sempre que for detectada a seqüência de bits formadores do padrão de alinhamento de quadro. Assim, caso tenha sido detectado o padrão de alinhamento de quadro nos bits  $b_1$  a  $b_7$  do quadro atual, a saída  $c3$  do módulo  $step3$  é ativada (transição ascendente), fazendo com que o estado ativo do módulo  $st.m$  passe a ser  $s3$ . Com isto, a saída  $insinc$  deste módulo é ativada, sinalizando a obtenção do sincronismo de quadro. Para o caso em que o padrão de alinhamento de quadro não tenha sido confirmado (não ativação do sinal  $fas$ ), a saída  $c3fz$  do módulo  $step3$  é ativada

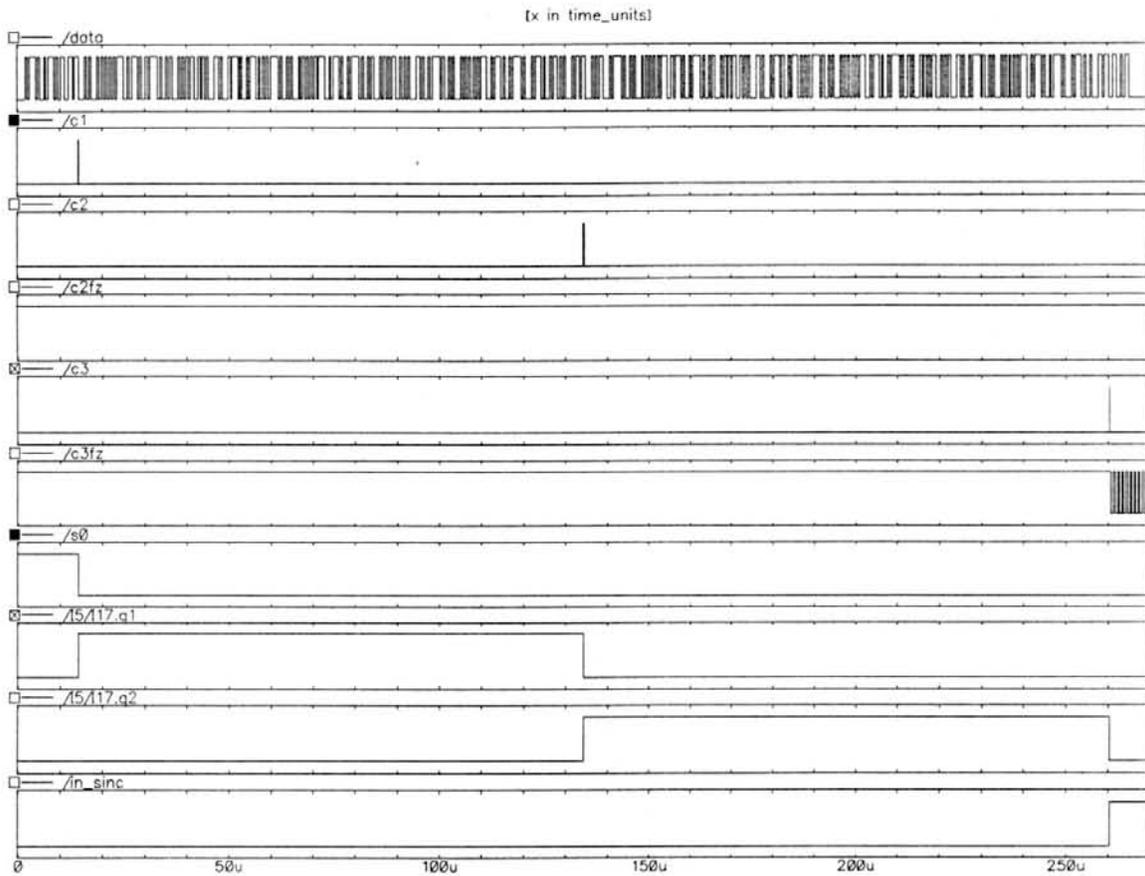


Figura 5.43 - Simulação do circuito de alinhamento de quadro.

(transição ascendente), fazendo com que o estado ativo do módulo *st.m* passe a ser *s0*.

Uma vez que tenha sido obtido o sincronismo de quadro pela ativação do estado *s3* do módulo *st.m*, somente a ativação do sinal *porz* poderá inicializar o circuito de alinhamento de quadro.

A seguir serão apresentados os resultados das simulações do circuito de alinhamento de quadro. Para este circuito, foram realizadas três simulações. A primeira simulação realizada, que pode ser vista na figura 5.43, mostra a ativação do sinal *in\_sinc* após a ocorrência em seqüência das três condições de teste, *c1*, *c2* e *c3*.

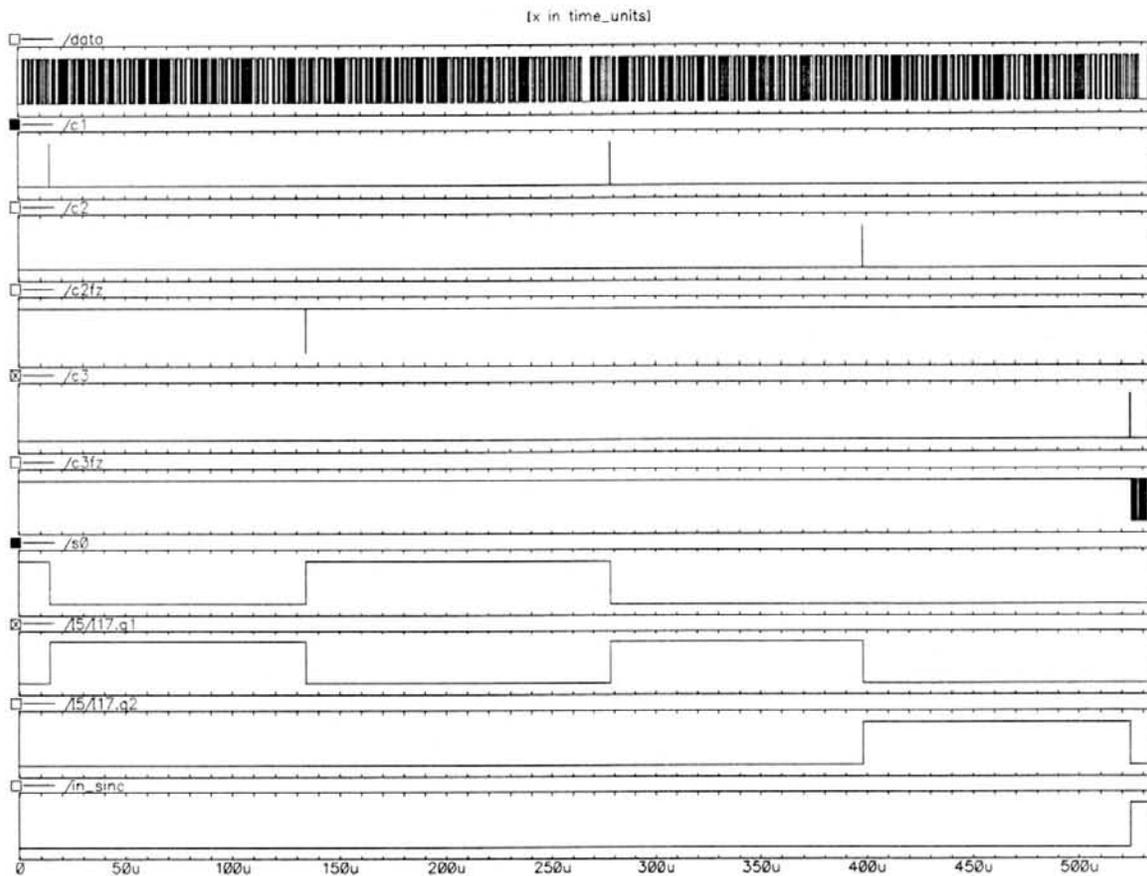


Figura 5.44 - Simulação de falha para a segunda condição.

Na figura 5.44 é mostrada a simulação da falha da condição  $c2$ . É interessante notar a ativação do sinal  $c2fz$  e do estado  $s0$ , em decorrência da ativação do primeiro. Pode ser notado também na figura, que após a falha no teste da condição  $c2$ , a ocorrência em seqüência das três condições de teste, faz com que a saída *in\_sinc* seja ativada.

Por fim, na figura 5.45 pode ser vista a simulação da falha da condição de teste  $c3$ , em que é mostrada a ativação do sinal  $c3fz$  e do estado  $s0$ , decorrente da ativação de  $c3fz$ . Tal como nas simulações anteriores, a ativação do sinal *in\_sinc* acontece após a ocorrência em seqüência das três condições de teste,  $c1$ ,  $c2$  e  $c3$ .

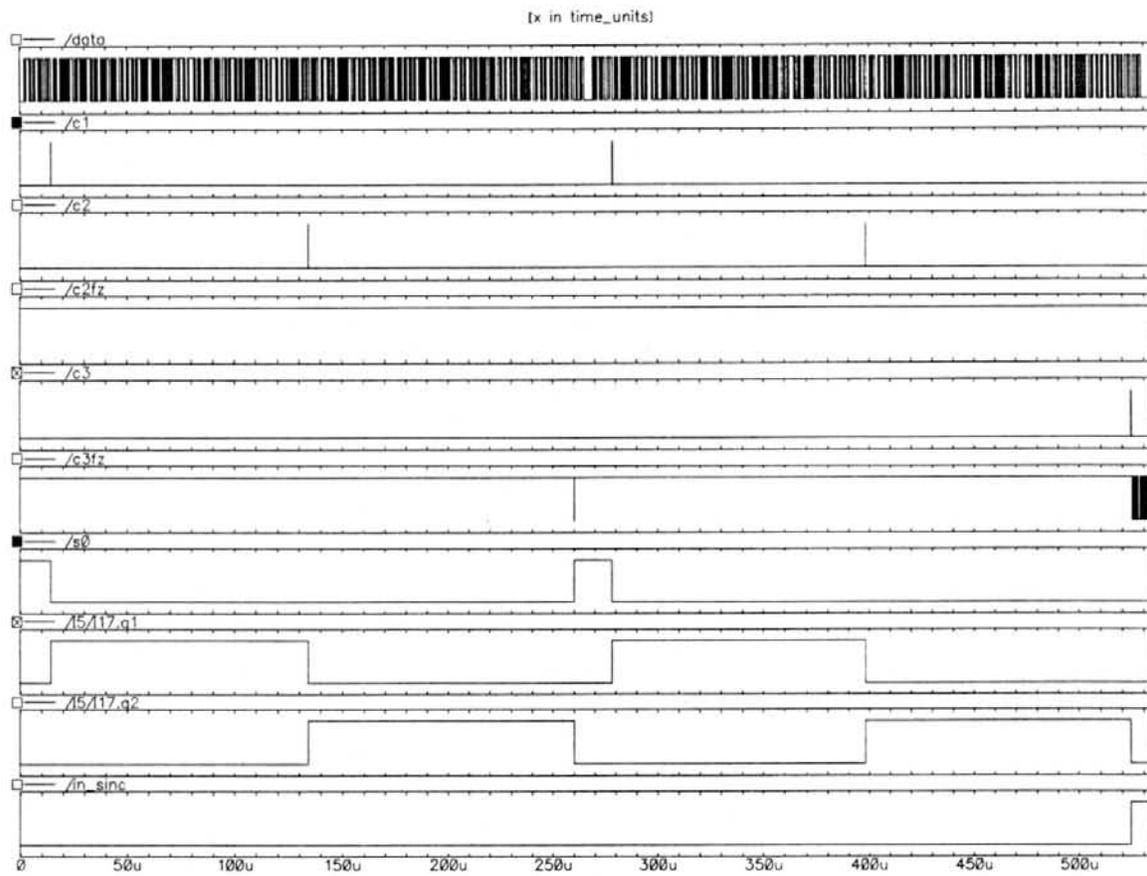


Figura 5.45 - Simulação de falha para a terceira condição.

### 5.3.3 Geração de *Time-Slots*

Com a obtenção do alinhamento de quadro, é possível iniciar a geração dos *time-slots* para a separação dos octetos recebidos a 2048 kbit/s (agregado). Esta função é desenvolvida pelo circuito gerador de *time-slots*.

O circuito gerador de *time-slots*, cujo diagrama de blocos pode ser visto na figura 5.46, é implementado a partir de 3 sub-circuitos quais sejam, um contador em anel (módulo *ring.counter* na figura), um contador síncrono de 3 bits (módulo *3.bit.sync.count*) e um gerador de sinais de temporização (módulo *tmg.gen*).

A geração dos 32 *time-slots* do quadro *E1* é efetivamente realizada pelo módulo *ring.counter*, que disponibiliza 32 saídas, das quais apenas uma pode estar ativa em um determinado instante de tempo. Este módulo é empregado também

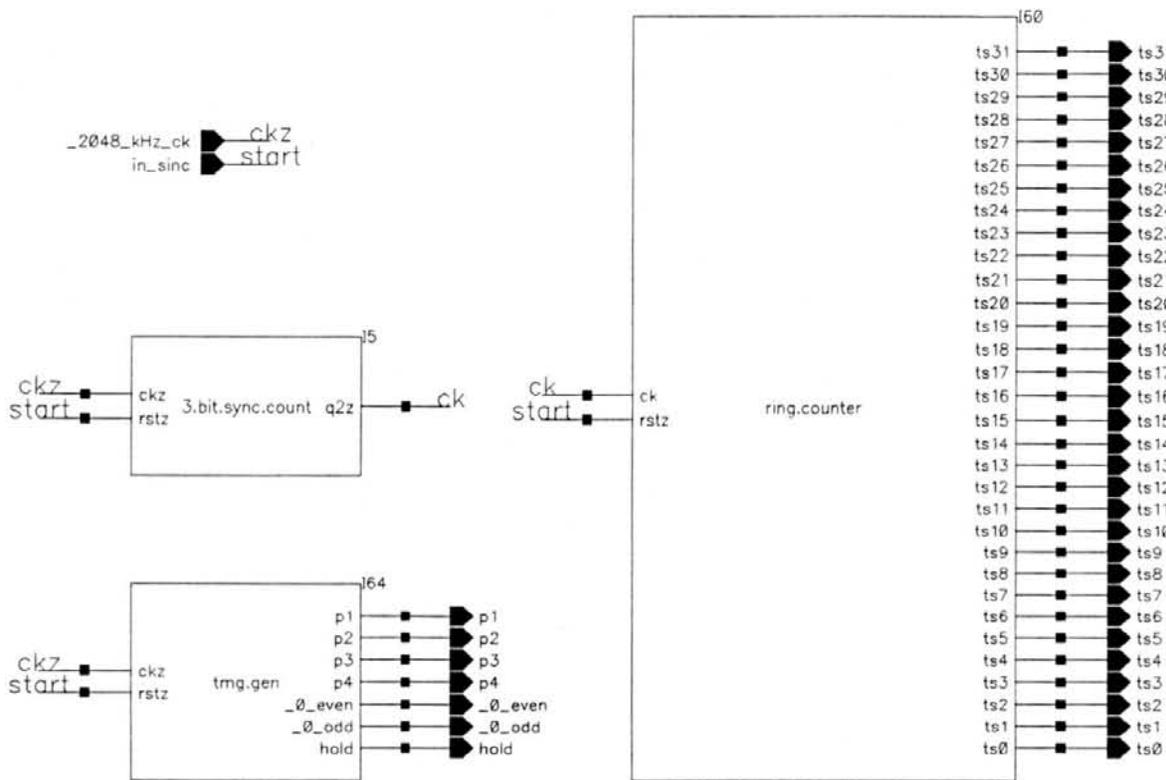


Figura 5.46 - Módulo gerador de *time-slots*.

para a geração dos *time-slots* para a multiplexação. No circuito *ring.counter*, a ativação da entrada *rstz* (inicialização do módulo) desativa todas as saídas, com exceção da saída *ts1*, que passa a ser a saída ativa, independentemente do estado anterior das demais saídas do módulo *ring.counter*. O estado das saídas pode ser alterado através de uma transição ascendente na entrada de relógio (entrada *ck*). Assim, se  $ts[n]$  é a saída ativa em um determinado instante de tempo, uma transição ascendente na entrada *ck* faz com que a saída seguinte ( $ts[n+1]$ ) seja ativada.

O módulo *3.bit.sync.count* fornece as transições ascendentes necessárias à mudança de estado das saídas do módulo *ring.counter*. Para tanto, o módulo *3.bit.sync.counter* recebe o sinal de relógio de 2048 kHz, sendo que em sua saída tem-se uma transição ascendente a cada oito ciclos deste sinal de relógio.

A inicialização do gerador de *time-slots* ocorre a partir da ativação do sinal *in sinc*. Esta por sua vez, ocorre no início do *time-slot* correspondente ao canal 1. Assim, a ativação dos *time-slots* coincide com os octetos de 2048 kbit/s,

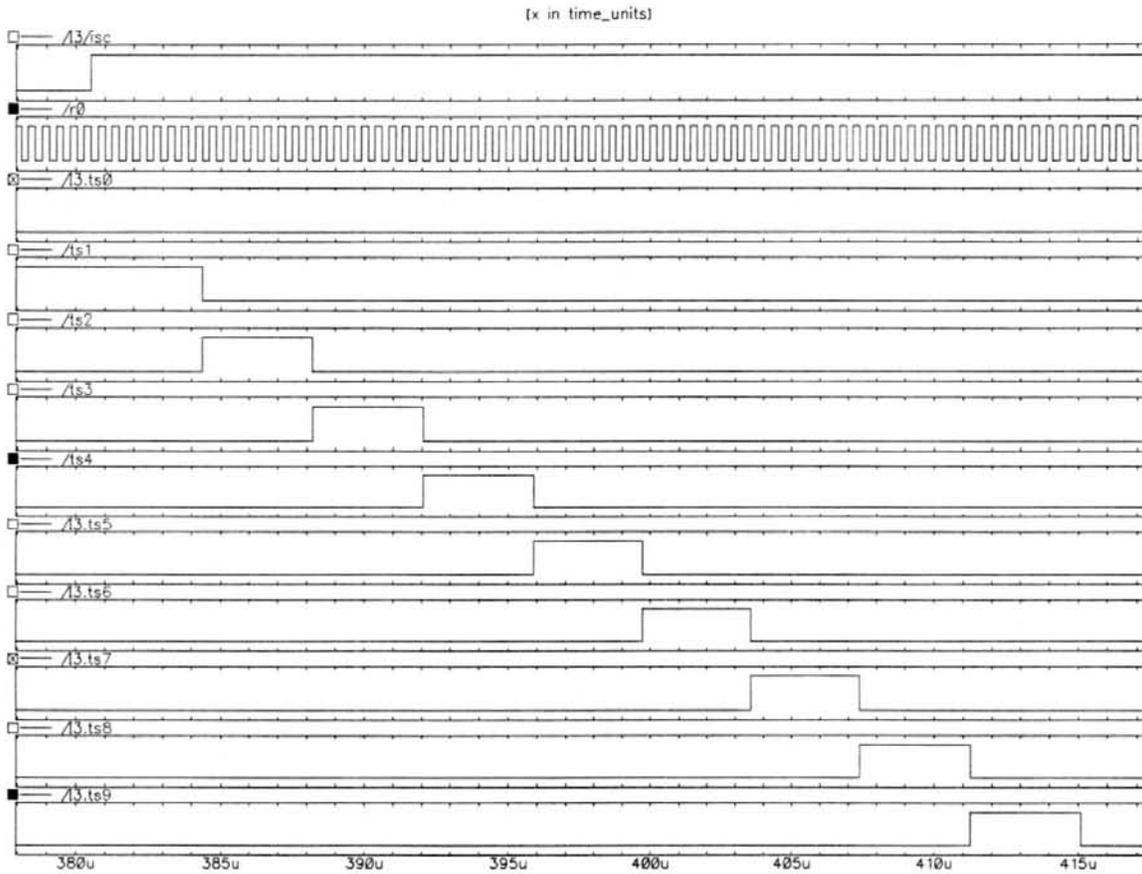


Figura 5.47 - Simulação do circuito gerador de *time-slots*.

podendo estes serem separados, estando assim próprios para serem transmitidos a seus respectivos canais tributários.

O circuito gerador de sinais de temporização (módulo *tmg.gen*) fornece sinais de relógio para os módulos de transmissão aos canais tributários. Existe um relacionamento estreito entre a temporização necessária à transmissão dos octetos demultiplexados aos tributários e a geração dos *time-slots*. Por isto, o módulo *tmg.gen* inclui-se no módulo gerador de *time-slots*. A relação entre os sinais *ts0* a *ts31* e os sinais *p1*, *p2*, *p3*, *p4*, *0 even*, *0 odd* e *hold* será detalhada na seção que descreve os módulos de transmissão de tributários.

Na figura 5.47 pode ser vista a ativação em seqüência dos *time-slots* *ts0* a *ts9*. A partir da ativação do sinal *in sinc* pode ser comprovada a ativação seqüencial dos *time-slots*, a cada oito ciclos do sinal de relógio de 2048 kHz.

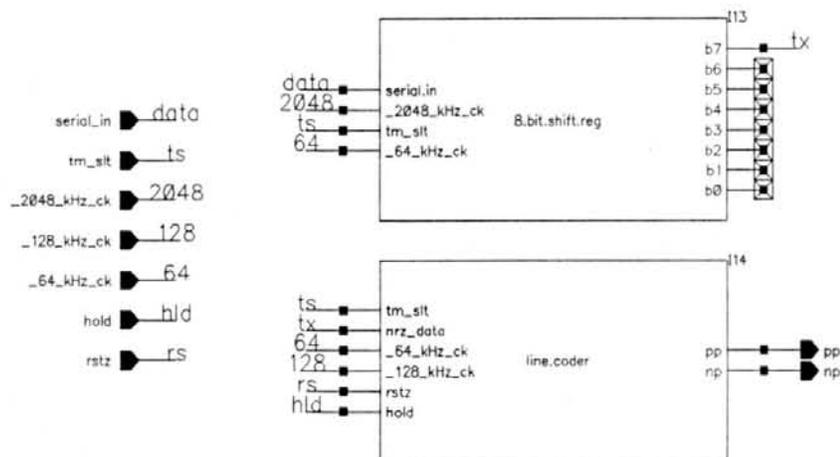


Figura 5.48 - Módulo de transmissão para canais tributários.

### 5.3.4 Módulos de Transmissão de Tributários

Tendo sido recebido o agregado proveniente do *buffer elástico*, é necessário realizar a demultiplexação dos octetos e transmissão dos mesmos aos seus respectivos canais tributários. Esta função é executada pelos módulos de transmissão dos canais tributários. Na figura 5.48 é mostrado o diagrama de blocos funcionais para um destes canais de transmissão. Para um equipamento *mux/demux* completo são necessários 31 destes módulos, um para cada interface de transmissão de tributário.

Um módulo de transmissão para um canal tributário é formado pelos circuitos *8.bit.shift.reg* e *line.coder*, cujos diagramas esquemáticos podem ser vistos no apêndice deste trabalho. O circuito *8.bit.shift.reg* é um registrador de deslocamento sobre o qual são escritos oito bits de dados a 2048 kbit/s. Os flancos ascendentes do sinal de relógio de 2048 kHz transferem o octeto presente à entrada *serial in*, durante a ativação do *time-slot*. A ativação do *time-slot* ocorre a meio período do nível lógico *zero* do sinal de relógio de 64 kHz. Desta forma, um octeto na forma serial é convertido para a forma paralela e disponibilizado à saída do módulo *8.bit.shift.reg* (terminais  $b_0$  a  $b_7$ ).

O octeto na forma paralela à saída do módulo *8.bit.shift.reg*, necessita a seguir ser convertido para a forma serial a 64 kbit/s. Para tanto, os flancos

descendentes do sinal de relógio de 64 kHz fazem com que os bits do octeto em paralelo na saída do módulo *8.bit.shift.reg* sejam rotacionados e, como resultado, na saída  $b_7$  deste módulo, os octetos são disponibilizados na forma serial a 64 kbit/s e levados ao circuito codificador de linha (módulo *line.coder*). Desta forma, o módulo *8.bit.shift.reg* recebe um sinal na forma de rajadas de 8 bits seriais a 2048 kbit/s a cada 125  $\mu$ s e converte este sinal em um feixe contínuo de dados seriais a 64 kbit/s, constituindo-se assim em uma pequena estrutura de armazenamento elástico.

A codificação do sinal serial *NRZ* proveniente do módulo *8.bit.shift.reg* é realizada pelo módulo *line.coder*. Um bit de dados presente à entrada *nrz data* deste módulo é avaliado sincronamente à transição ascendente do sinal de relógio de 64 kHz e, sendo um nível lógico *zero* é gerado um bloco-código **1010** para transmissão, ao passo que se o bit avaliado for um nível lógico *um*, é gerado um bloco-código **1100** para transmissão. Os blocos-código correspondentes aos níveis lógicos *zero* e *um* são gerados a partir de sinais com freqüências de 128 kHz e 64 kHz respectivamente, sendo este último o próprio sinal de relógio de 64 kHz.

O módulo codificador de linha, viola a regra de alternância de polaridade dos sinais de linha *pp* e *np* a cada oito ciclos do sinal de relógio de 64 kHz, contados a partir da ativação do *time-slot* do canal. Para tanto, é fornecido também a este módulo, o sinal de temporização de *time-slot*.

O sinal *hld*, na entrada *hold* do módulo *line.coder*, é um sinal de relógio com freqüência de 256 kHz, utilizado para eliminar espículas do sinal codificado gerado, através de sobre-amostragem do sinal com espículas, nos instantes em que este está estável.

Para a carga de um octeto sobre o módulo *8.bit.shift.reg* é necessário que o sinal de relógio de 64 kHz esteja em nível lógico *zero*, quando da ativação do *time-slot* do canal. Sabe-se que o defasamento entre *time-slots* adjacentes é de 3,90625  $\mu$ s. Assim, o sinal de relógio de 64 kHz fornecido a cada módulo de transmissão de tributário deve também acompanhar este defasamento, para canais adjacentes. A

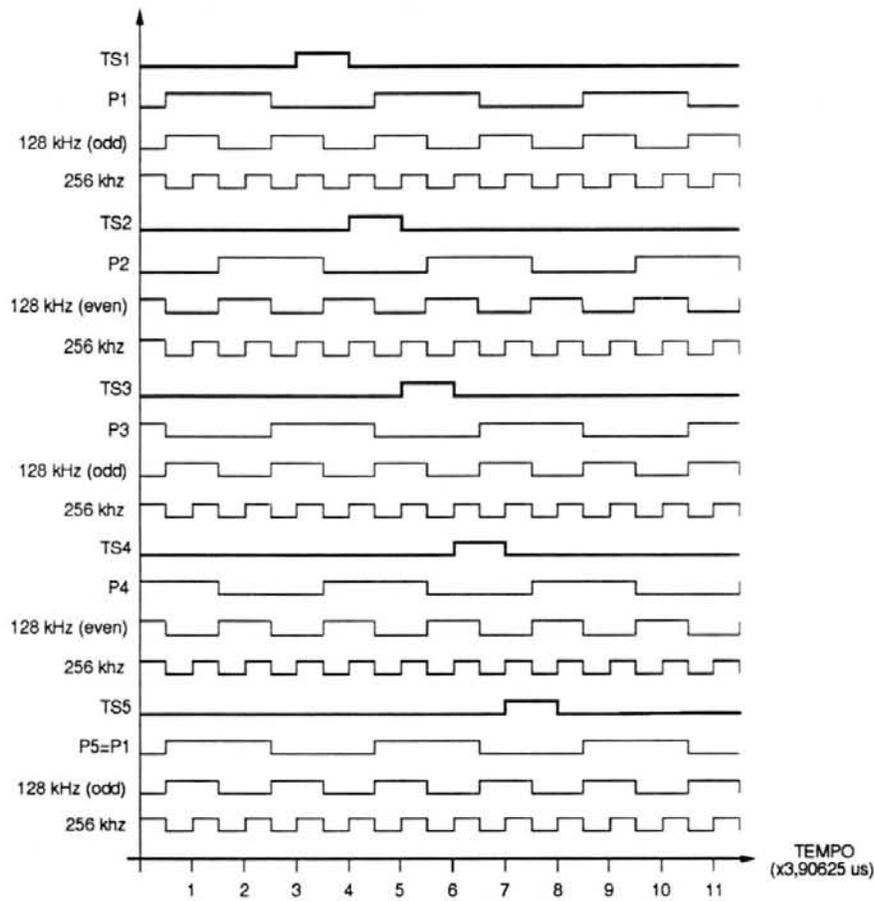


Figura 5.49 - Diagrama de temporização para os módulos de transmissão de tributários.

figura 5.49 ilustra diagramas de temporização para os canais 1 a 5, onde podem ser vistos também os sinais de 128 kHz (formadores dos blocos-código *zero*) e de 256 kHz (sinal *hold*). Um exame dos sinais mostrados na figura, revela que para o sinal de relógio de 64 kHz são necessárias quatro fases ( $p1$ ,  $p2$ ,  $p3$  e  $p4$ ), uma vez que, a partir da fase  $p4$ , um atraso de  $3,90625 \mu\text{s}$  produz uma fase  $p5$  a qual é equivalente à fase  $p1$ . Assim, a fase  $p1$  constitui-se no sinal de relógio de 64 kHz para os canais 1, 5, 9, 13, 17, 21, 25 e 29. A fase  $p2$  é o sinal de relógio de 64 kHz para os canais 2, 6, 10, 14, 18, 22, 26 e 30. Para os canais 3, 7, 11, 15, 19, 23, 27 e 31, o sinal de relógio de 64 kHz é  $p3$ . Por fim,  $p4$  é o sinal de relógio para os canais 4, 8, 12, 16, 20, 24 e 28.

Para a geração dos blocos-código representativos dos bits *zero*, é necessária a existência de apenas duas fases do sinal de 128 kHz, uma vez que um

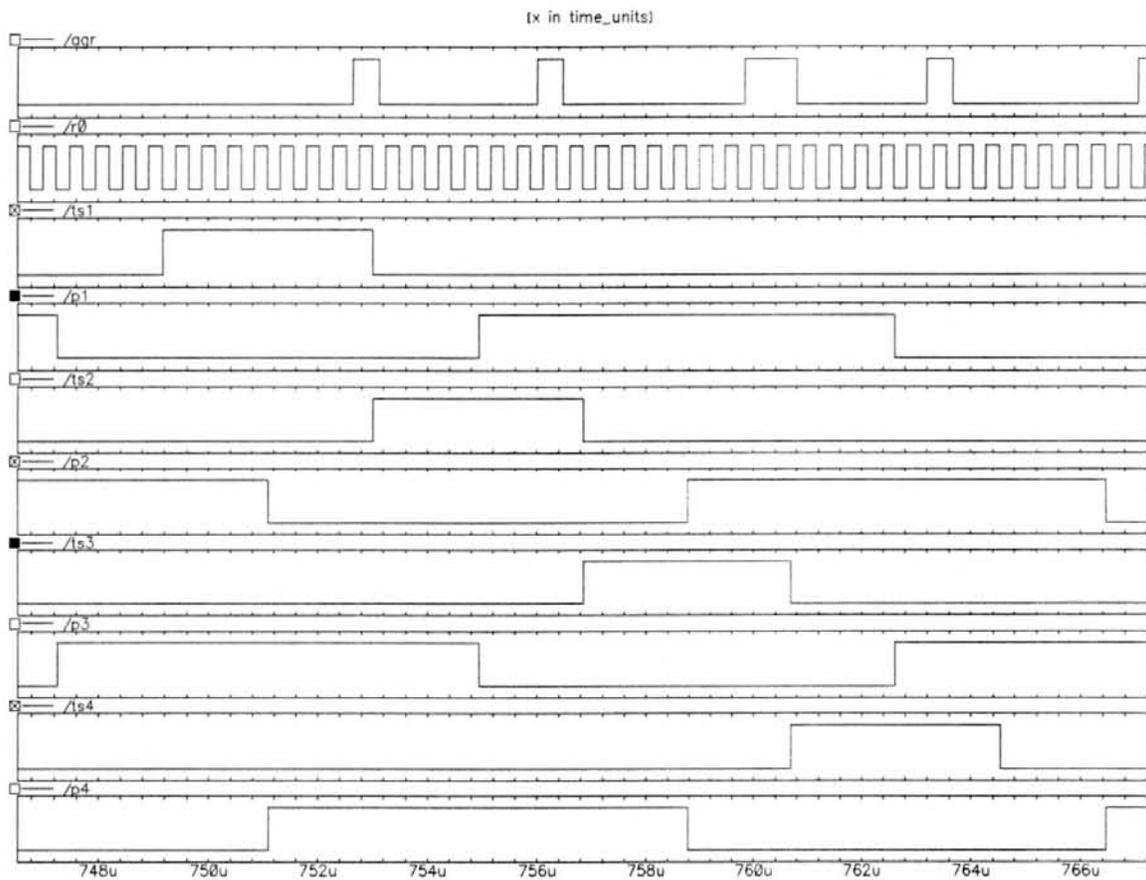


Figura 5.50 - Carga de octetos a 2048 kbit/s nos módulos *8.bit.shift.reg*.

atraso de  $3,90625 \mu\text{s}$  em um sinal com esta frequência, é equivalente a invertê-lo. Assim, para os canais de numeração par (2, 4, 6, ..., 30), o sinal de 128 kHz é denominado *even*, ao passo que para os canais de numeração ímpar (1, 3, 5, ..., 31), o sinal de 128 kHz denomina-se *odd*.

O sinal de 256 kHz é o mesmo para todos os módulos de transmissão de 64 kbit/s, uma vez que um atraso de  $3,90625 \mu\text{s}$  não modifica sua fase em relação aos sinais de 64 kHz e 128 kHz.

Os sinais de relógio de 64 kHz, 128 kHz e 256 kHz são produzidos pelo circuito gerador de temporização, que constitui-se em um sub-circuito do módulo gerador de *time-slots*, descrito na seção 5.3.3.

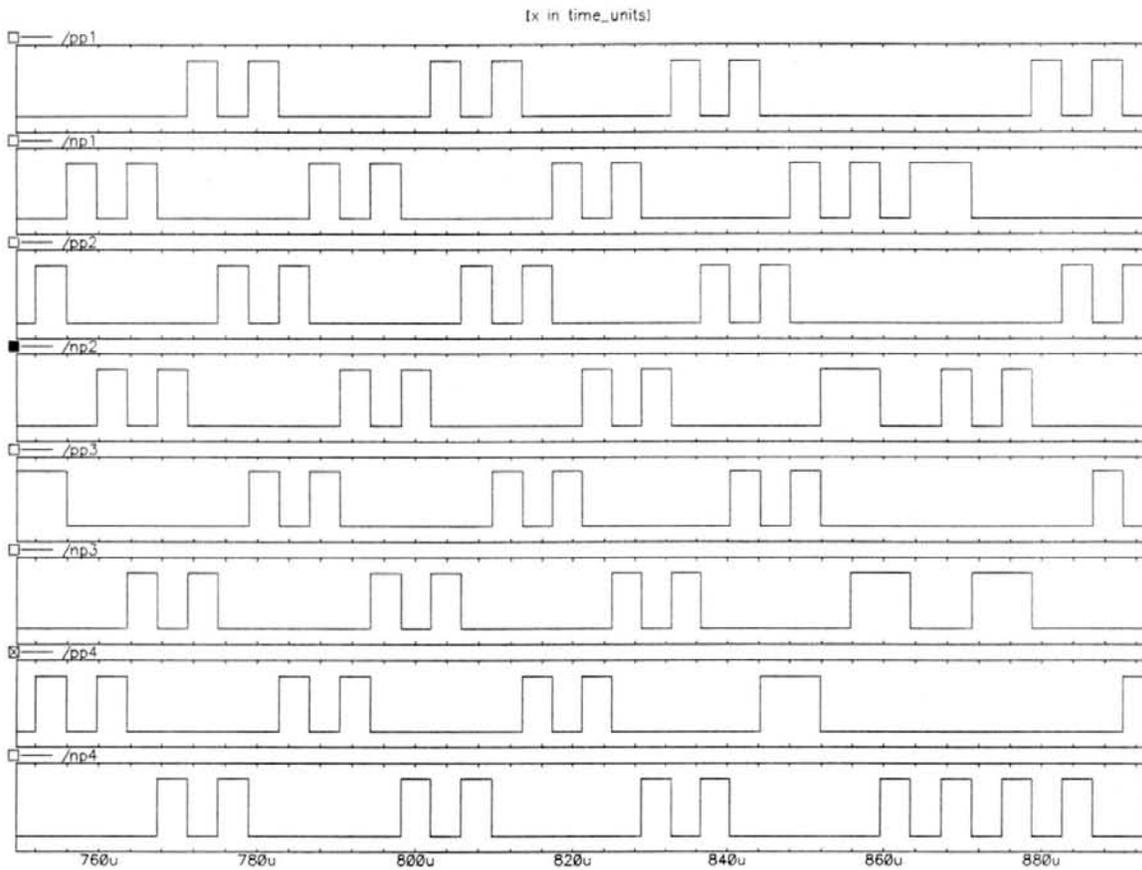


Figura 5.51 - Saídas *G.703* codificadas a 64 kbit/s para os canais 1, 2, 3 e 4.

A seguir serão mostrados os resultados da simulação do funcionamento do módulo de transmissão para os canais tributários.

Na figura 5.50 pode ser vista a simulação do circuito de transmissão para canais tributários, no momento de ativação dos *time-slots* dos canais 1, 2, 3 e 4. O sinal *agr* é o agregado, proveniente da saída do *buffer elástico* e *r0* é a fase principal do sinal de relógio de 2048 kHz. Os sinais *ts1*, *ts2*, *ts3* e *ts4* são os *time-slots* dos canais 1, 2, 3 e 4, sendo que *p1*, *p2*, *p3* e *p4* são os sinais de relógio de 64 kHz para estes canais.

Na simulação mostrada na figura 5.50 pode ser comprovada a carga dos octetos **0000 0001**, **0000 0010**, **0000 0011** e **0000 0100** respectivamente para os canais 1, 2, 3 e 4. Note-se que a ativação dos *time-slots* *ts1*, *ts2*, *ts3* e *ts4* ocorre respectivamente a meio período dos sinais de relógio de 64 kHz *p1*, *p2*, *p3* e *p4*.

A figura 5.51 mostra as saídas  $pp$  e  $np$  do circuito codificador, para os canais 1, 2, 3, e 4. Os octetos codificados para cada um destes canais, correspondem àqueles carregados no módulo *8.bit.shift.reg*, durante a ativação de seus respectivos *time-slots*, tal como mostrado na figura 5.50. Também é possível verificar nesta simulação, a violação da regra de alternância de polaridade do sinal codificado. Aqui, o bit  $b_7$  repete a polaridade do bit  $b_6$ . Os sinais  $pp$  e  $np$  de cada um dos canais são a seguir levados a circuitos *driver* de linha e transmitidos aos tributários remotos.

## 6 CONCLUSÕES

Neste trabalho foram estudadas as funções de multiplexação para a hierarquia digital de transmissão *ITU-T E1*. As funções de multiplexação compreendem os procedimentos necessários à implementação de equipamentos *mux/demux*, largamente empregados na estruturação de redes digitais integradas. Assim, inicialmente foram estudadas as recomendações técnicas que norteiam a estrutura dos equipamentos *mux/demux* a nível de interfaces, formatos de quadro e multiquadro, taxas de bit, códigos de linha, recursos para supervisão de enlace e controle de erros.

Em seqüência, foi realizado um estudo complementar com a finalidade de identificar o ambiente de operação dos equipamentos *mux/demux* digitais, em aplicações reais.

Partindo dos estudos iniciais realizados, foi possível a especificação e projeto de módulos de circuito para execução das funções de multiplexação. Estes módulos foram a seguir simulados com o fito de validar as arquiteturas propostas. Para tanto, foi utilizado o *framework SOLO/CADENCE* para a edição esquemática dos circuitos projetados, seguindo-se a simulação dos mesmos através do simulador lógico *SILOS*.

Nesta fase do trabalho, os circuitos foram simulados para as mais severas condições de operação, especificamente no que diz respeito aos atrasos inerentes às estruturas lógicas utilizadas na implementação dos módulos funcionais, em vista da utilização de sinais de relógio com freqüências relativamente elevadas. Também foi simulada a capacidade de operação do sistema frente ao desalinhamento temporal dos sinais recebidos dos canais tributários e, de maneira enfática, seu funcionamento a partir de sinais degradados (temporização mesócrona e plesiócrona).

No total, foram especificados, projetados e simulados, os seguintes módulos funcionais para multiplexação *E1*: circuito para recuperação de relógio de 64 kHz,

codificador e decodificador *G.703*, detector de violações à regra de codificação *G.703*, *buffers elásticos* para tributários e agregado, geradores de *time-slots*, conversor de taxa serial/serial (2048 kbit/s / 64 kbit/s), módulo sinalizador de alinhamento de quadro, bem como diversos sub-circuitos utilizados na implementação dos módulos funcionais. Os módulos codificador e decodificador *HDB-3*, já previamente especificados e projetados, foram apenas apresentados e re-simulados neste trabalho.

Outros módulos, de implementação analógica, foram apenas especificados funcionalmente. Estes são: gerador de base de tempo para o sistema *mux/demux* e circuito para recuperação de relógio de 2048 kHz.

Por terem sido especificados e projetados independentemente, os módulos funcionais validados, oferecem ainda a possibilidade de serem utilizados em aplicações diversas àquela proposta neste trabalho. Como exemplo, os módulos funcionais para recepção e transmissão aos canais tributários, podem ser utilizados para a implementação de *codecs* para enlaces digitais de 64 kbit/s. Unidades para teste de sistemas de transmissão *E1* podem também ser implementados a partir destes módulos. A partir do redimensionamento do gerador de *time-slots*, é possível também criar sub-sistemas multiplexadores, com menor número de canais e velocidades de agregado próprias à aplicações específicas. Assim, ainda que não padronizados, tais sistemas podem ser empregados na estruturação de redes locais, concentradores de terminais públicos de consulta, entre outras aplicações. Por exemplo, um sub-sistema de multiplexação com 7 entradas tributárias de 64 kbit/s e um canal de serviço *E1*, produziria um agregado a 512 kbit/s, capaz de atingir maiores distâncias, sem necessidade de uso de regeneradores de linha. Tal aplicação implementaria um *PABX* digital, em um típico sistema de ganho de pares.

Como trabalhos futuros pretende-se, em uma primeira fase completar o conjunto de módulos funcionais para multiplexação *E1*, através do projeto e simulação dos seguintes circuitos, alguns destes já em andamento:

- Módulo para alinhamento de multiquadro *CRC-4*;

- verificador de redundância cíclica *CRC-4*;
- módulos para montagem e desmontagem do canal de serviço;
- portas de entrada/saída do canal interno de 20 kbit/s;
- contadores de taxa de erro para o verificador *CRC-4* e para os detectores dos padrões de alinhamento de quadro e multiquadro.

Além dos módulos acima citados, pretende-se ainda incorporar facilidades para o estabelecimento de laços de testes locais e remotos (*network test loopbacks*) e um sistema de controle capaz de coordenar a operação conjunta dos módulos, a nível de sistema.

Paralelamente a isto, será realizado um estudo visando o particionamento racional das funções implementadas, na forma de um *chip-set* para aplicação em redes de comunicação *E1*. Finalmente, os circuitos projetados, validados e particionados serão submetidos à fabricação na *foundry* francesa *ES2*.

É importante ressaltar que durante a elaboração deste trabalho, o processo *CMOS 1.2 $\mu$*  da *foundry ES2*, para o qual os circuitos especificados foram projetados, deixou de ser oferecido no âmbito do Programa Multi-usuário. Desta forma, é necessário portar o projeto desenvolvido para o processo atualmente disponível no Programa Multi-usuário, qual seja o de *1.0 $\mu$* .

Em uma segunda fase, pretende-se projetar os módulos gerador de base de tempo, recuperador de relógio de 2048 kHz, regeneradores e *drivers* de linha para agregado e tributários, todos de implementação analógica. A partir deste ponto e, utilizando-se o *chip-set* digital, será possível o projeto e construção de um sistema *mux/demux E1*, para aplicação em redes digitais de comunicação. Nesta fase, poder-se-á também desenvolver regeneradores e *drivers* de linha para cabos coaxiais, ao invés de pares de fios trançados. Esta alternativa, ainda que não prevista em norma, possibilitaria um aumento no alcance de utilização do sistema multiplexador.

A incorporação de recursos tais como acesso analógico (*PCM* telefônico, p.ex.), canais fracionários e *sub-rate* às entradas tributárias e capacidade de comutação temporal e espacial, bem como facilidades para dotar o sistema com funções de gerenciamento e controle de rede, tornariam o sistema *mux/demux* um poderoso nó de rede digital de comunicações. Todas estas são possibilidades que abrem um extenso leque de tópicos de interesse à trabalhos acadêmicos ou de pesquisa.

Enfim, a validade do trabalho realizado está intimamente relacionada com sua aplicabilidade. Com efeito, a partir do desenvolvimento da microeletrônica enquanto suporte a diferentes áreas tecnológicas, tornou-se possível também à área de comunicações, a implementação de estruturas digitais de comunicação, mais eficientes, econômicas e confiáveis, adjetivos estes que, amalgamados, resultam à comunidade de usuários, em serviços com qualidade crescentemente melhorada.

**ANEXO 1    SUBCIRCUITOS PARA OS  
MÓDULOS DE RECEPÇÃO DE  
TRIBUTÁRIOS**

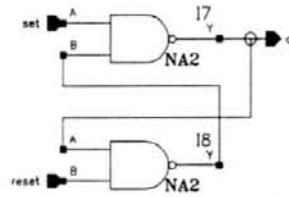


Figura 1.1 - Módulo *sr.ff*; (flip-flop set-reset).

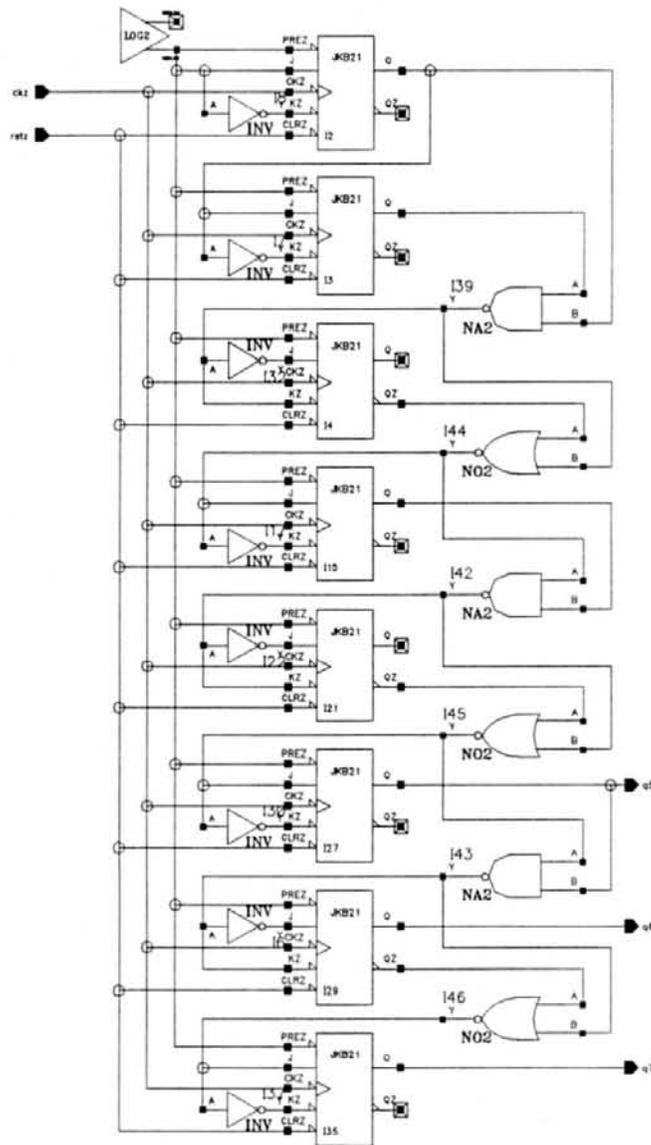


Figura 1.2 - Módulo *8.bit.sync.count*; (contador síncrono de oito bits).



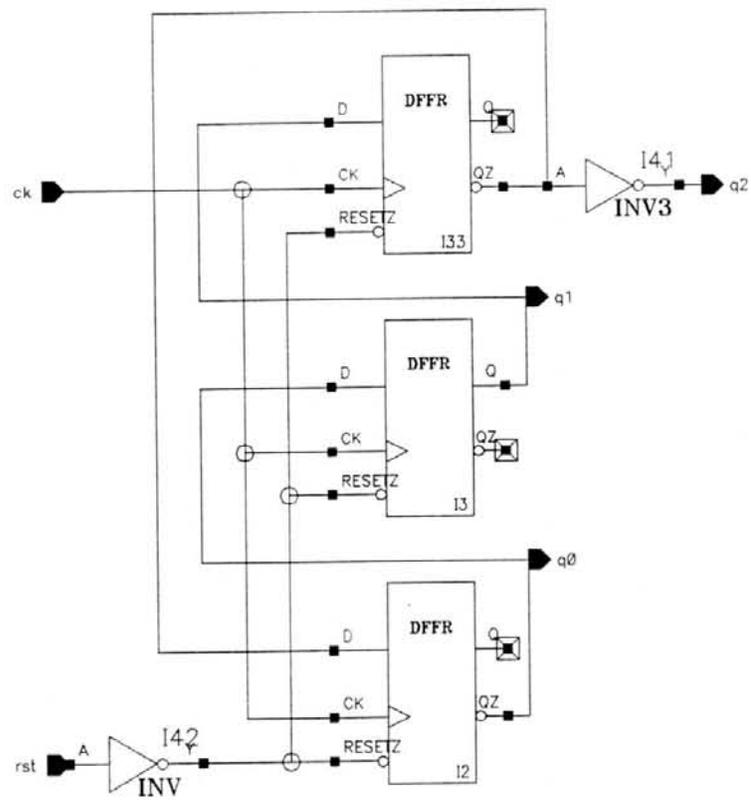


Figura 1.7 - Módulo *3.bit.jcount*; (contador *johnson* de 3 bits).

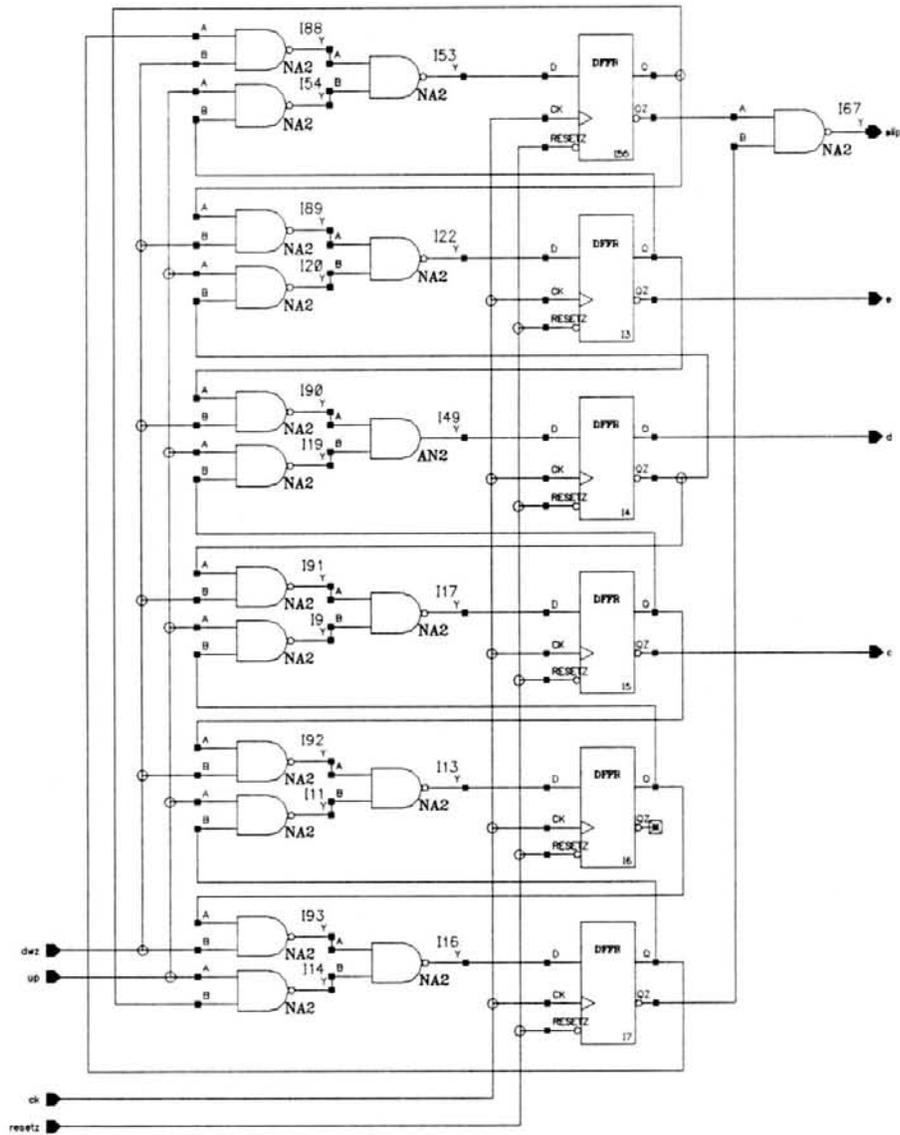


Figura 1.8 - Módulo *u.d.counter*; (contador *up/down*).

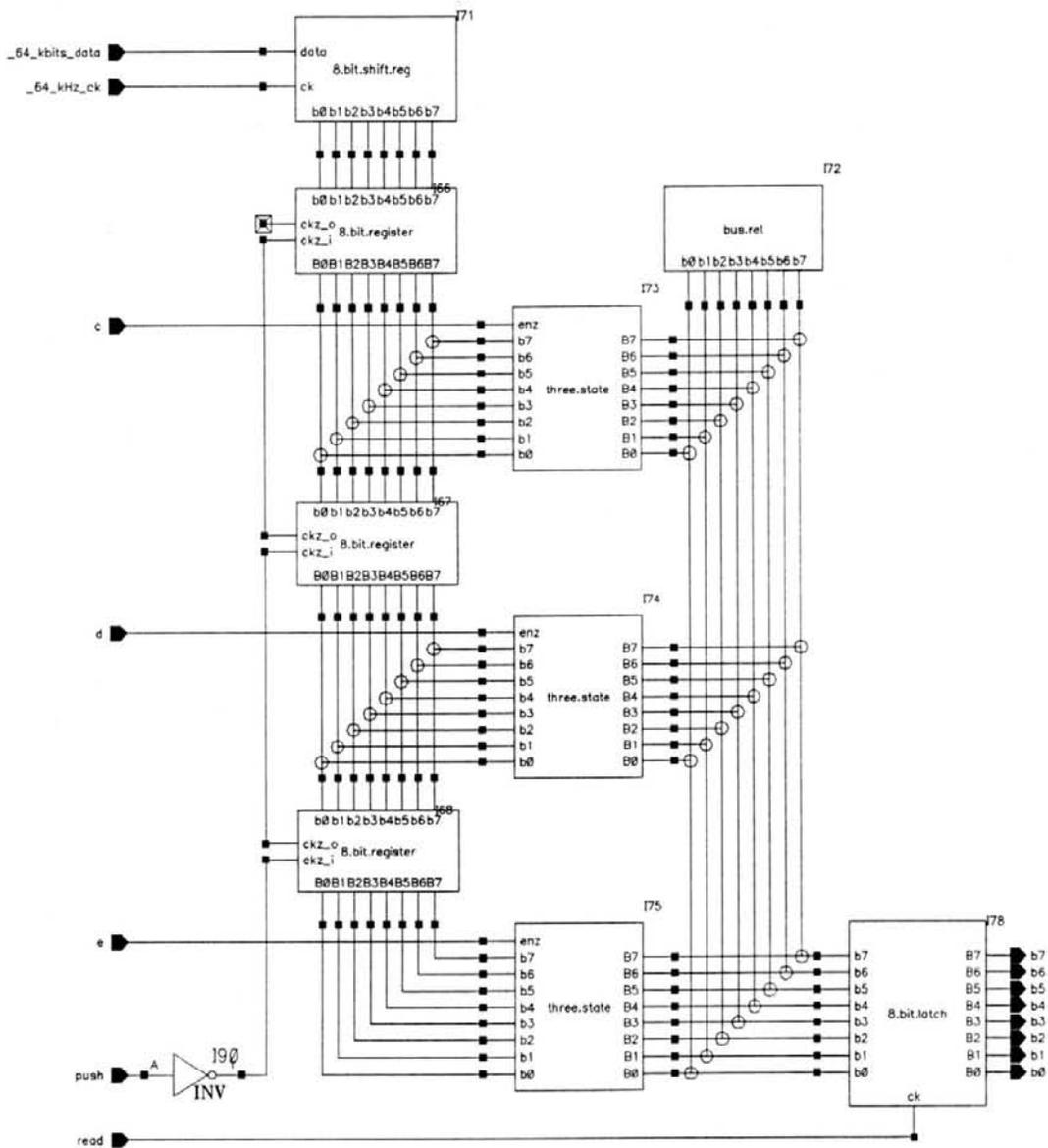


Figura 1.9 - Módulo *stack*; (registrador de deslocamento, registradores *R1*, *R2* e *R3*, tristates *T1*, *T2* e *T3*, retentor de barramento e *latch* de saída).

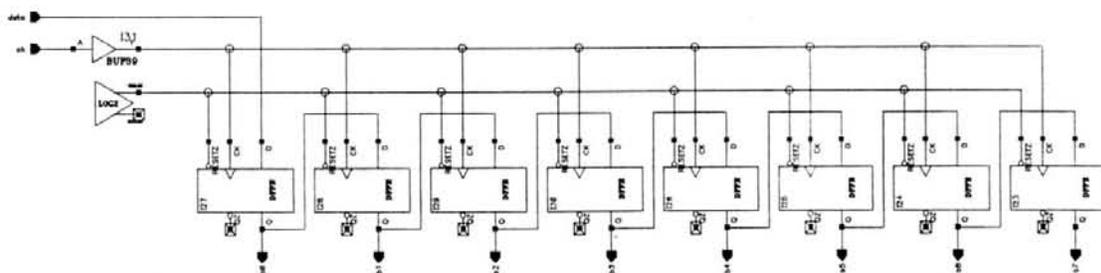


Figura 1.10 - Módulo *8.bit.shift.reg*; (registrador de deslocamento de oito bits).

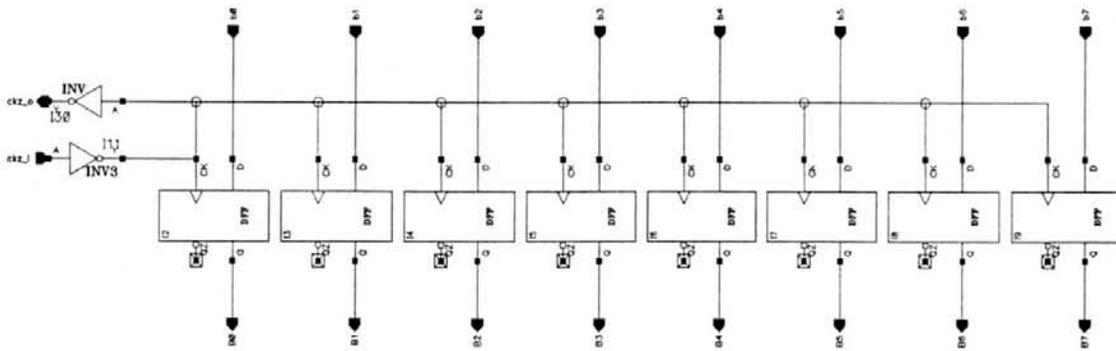


Figura 1.11 - Módulo *8.bit.register*; (registrador de carga paralela de oito bits).

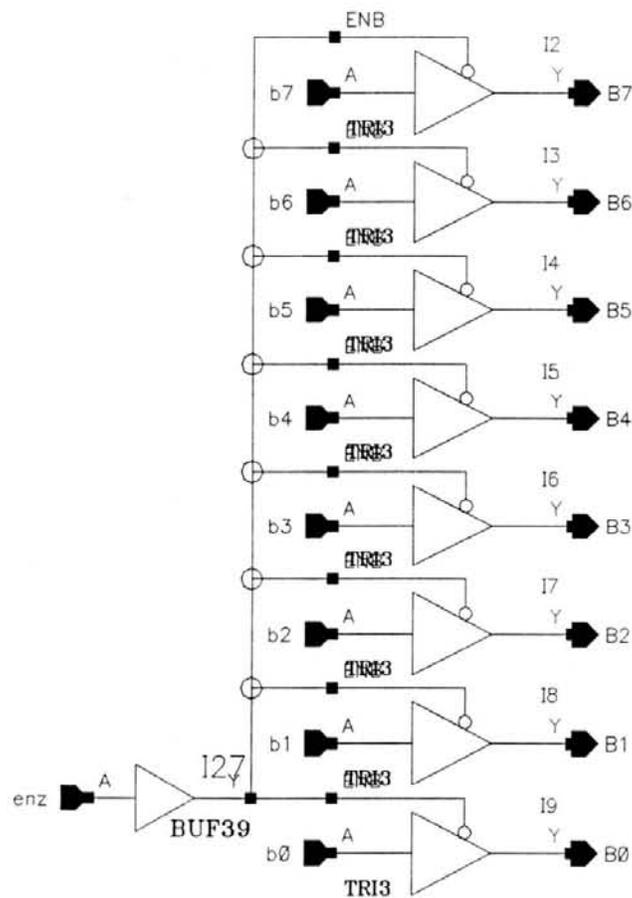


Figura 1.12 - Módulo *three.state*; (tristate).

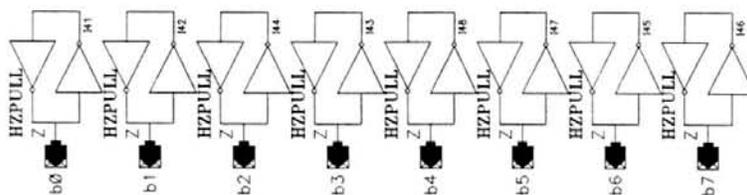


Figura 1.13 - Módulo *bus.ret*; (retentor de barramento).

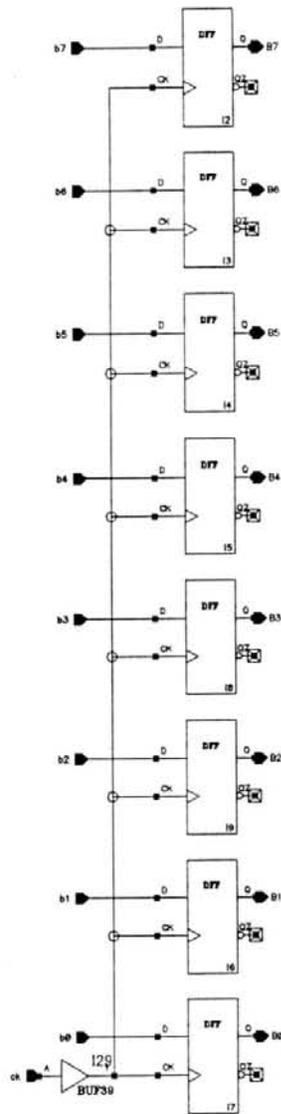


Figura 1.14 - Módulo *8.bit.latch*; (*latch* de oito bits).

**ANEXO 2    SUBCIRCUITOS PARA OS  
MÓDULOS CODIFICADOR E  
DECODIFICADOR *HDB-3***

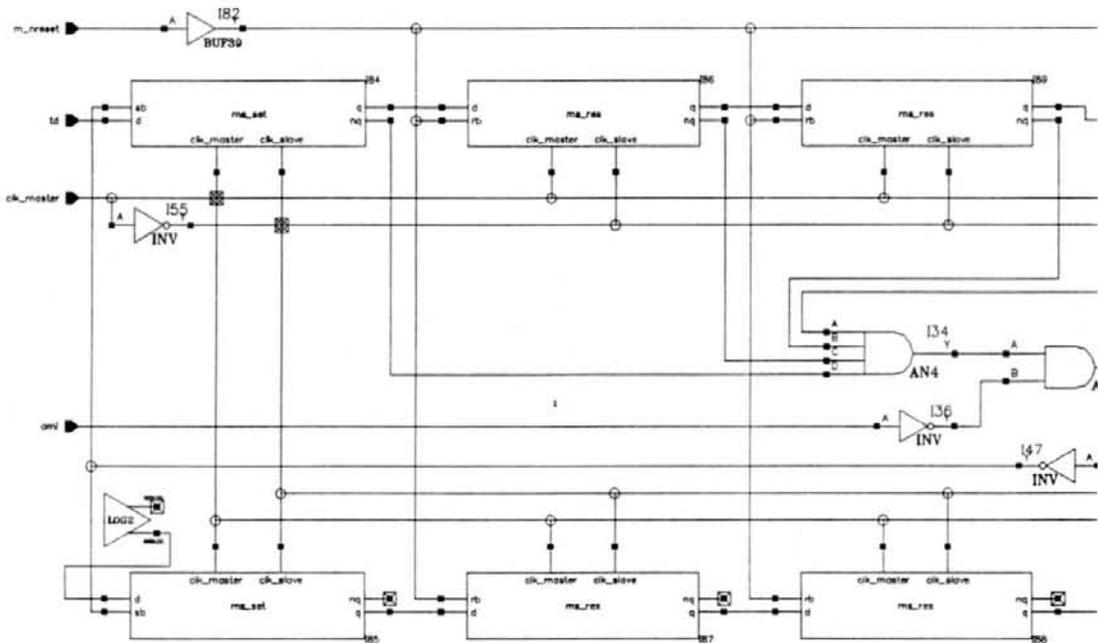


Figura 2.1 - Módulo *hdb cod*; (codificador *HDB-3*; esquema parcial 1 de 3).

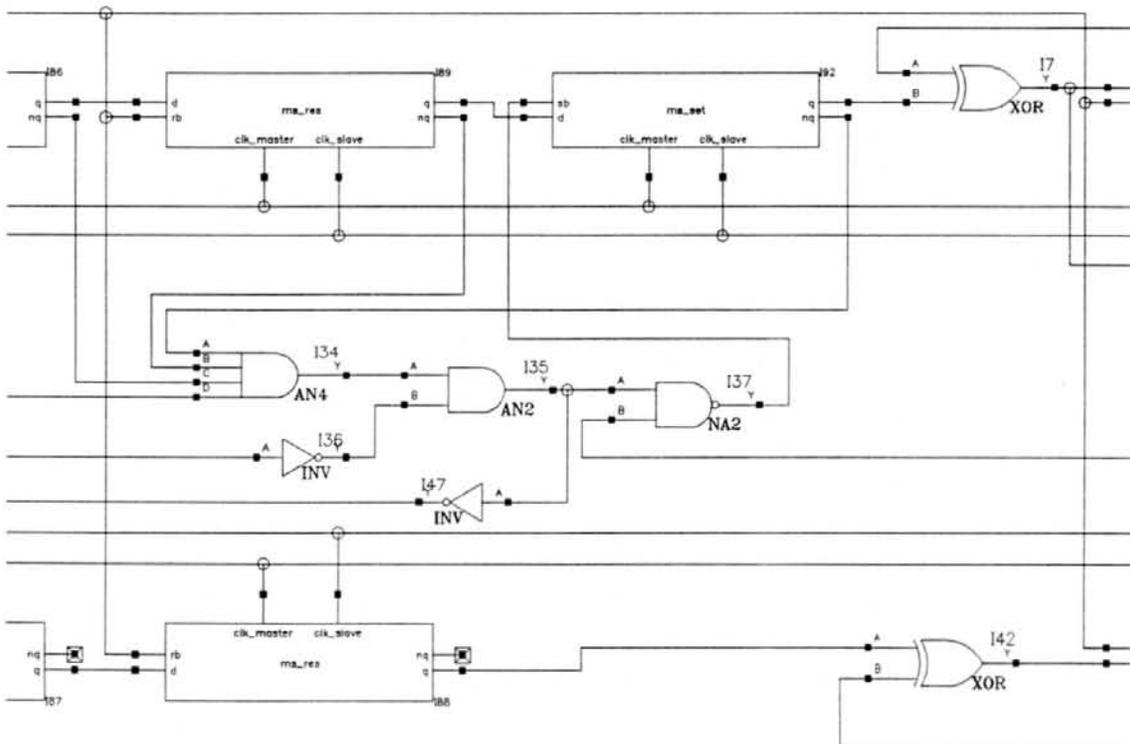


Figura 2.2 - Módulo *hdb cod*; (codificador *HDB-3*; esquema parcial 2 de 3).

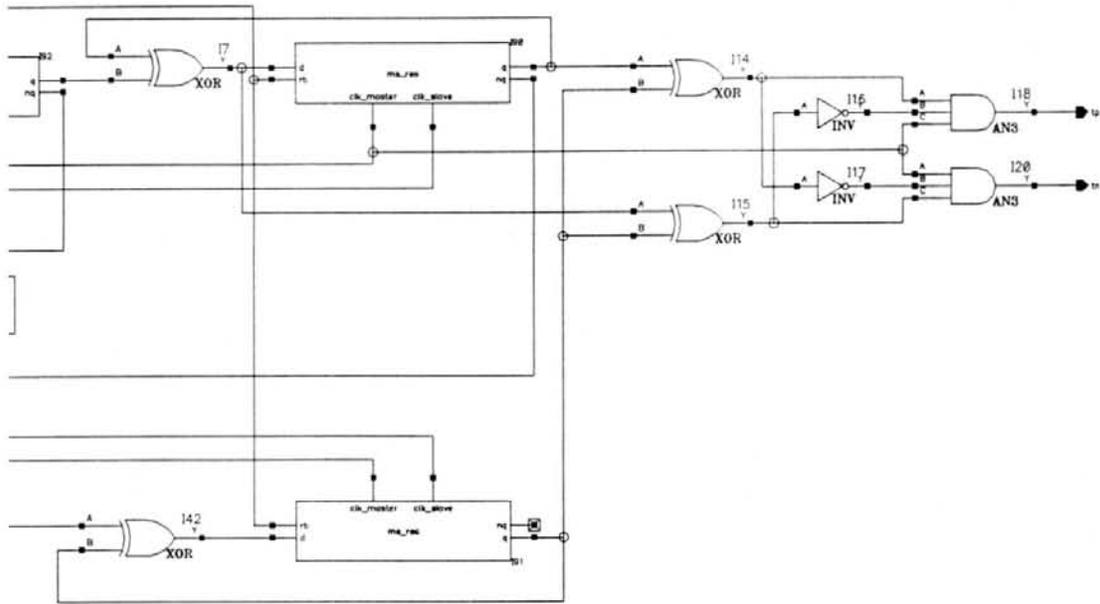


Figura 2.3 - Módulo *hdb cod*; (codificador *HDB-3*; esquema parcial 3 de 3).

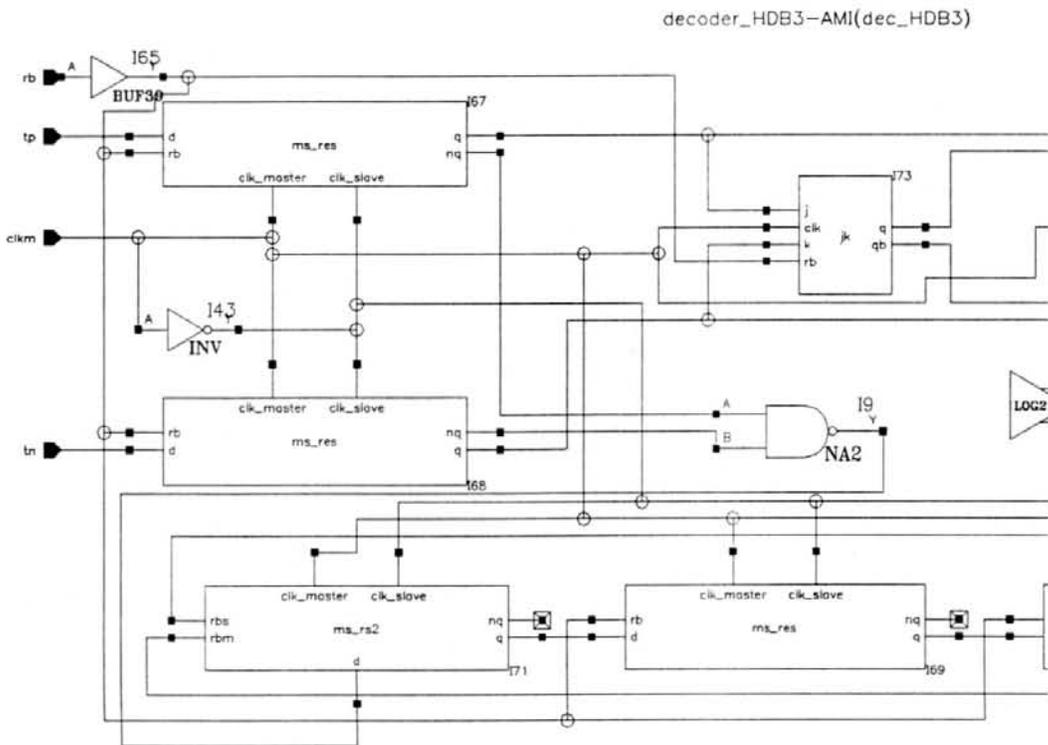
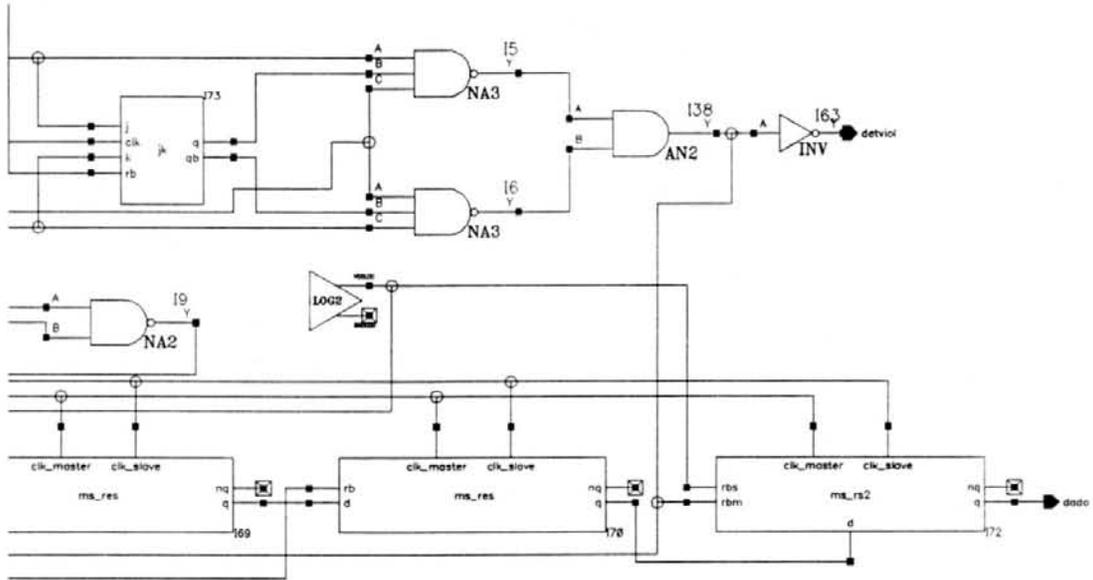
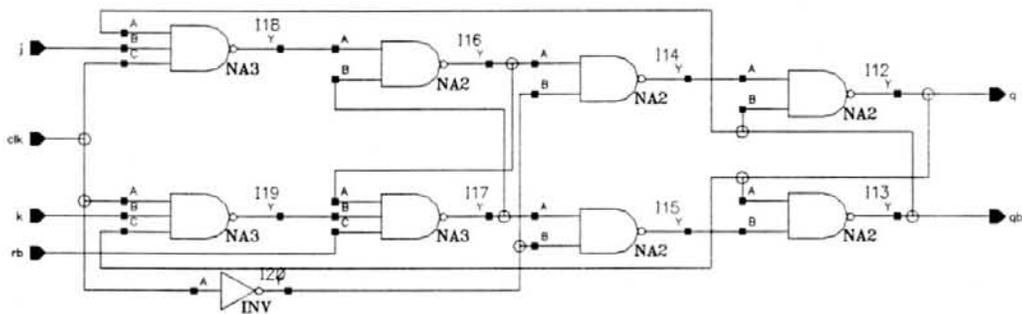


Figura 2.4 - Módulo *hdb dec*; (decodificador *HDB-3*; esquema parcial 1 de 2).

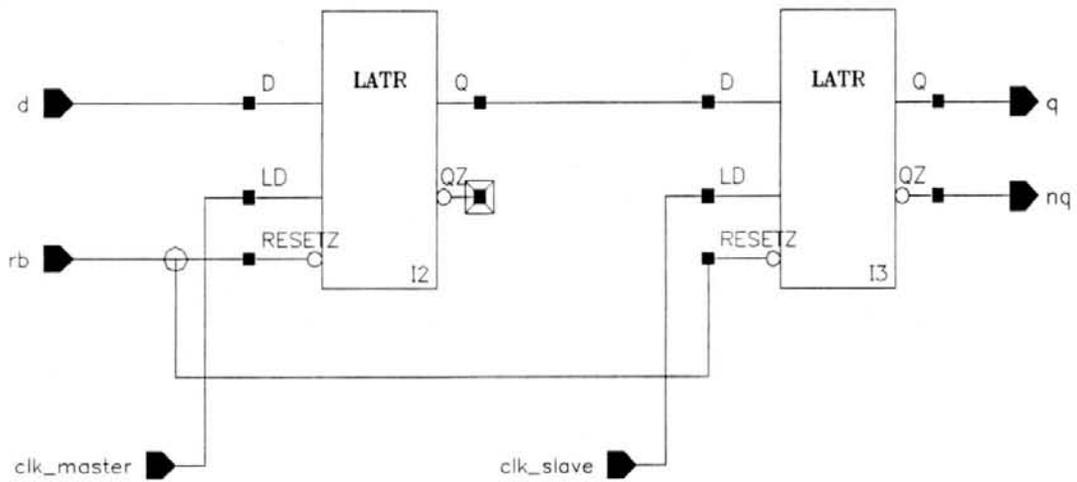
ecoder\_HDB3-AMI(dec\_HDB3)

Figura 2.5 - Módulo *hdb dec*; (decodificador *HDB-3*; esquema parcial 2 de 2).

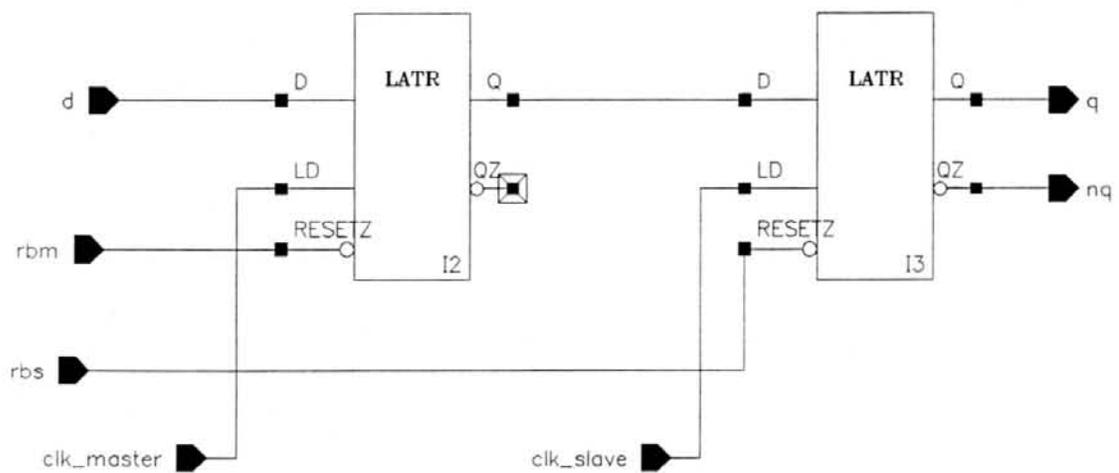
Flip-Flop\_JK\_com\_reset\_master(jk)

Figura 2.6 - Módulo *jk*; (flip-flop *jk*).

Flip-Flop\_Master-Slave(ms\_res)

Figura 2.7 - Módulo *ms res*; (flip-flop tipo *D master/slave* com reset).

Flip-Flop\_Master-Slave(ms\_rs2)

Figura 2.8 - Módulo *ms rs2*; (flip-flop tipo *D master/slave* com resets independentes).

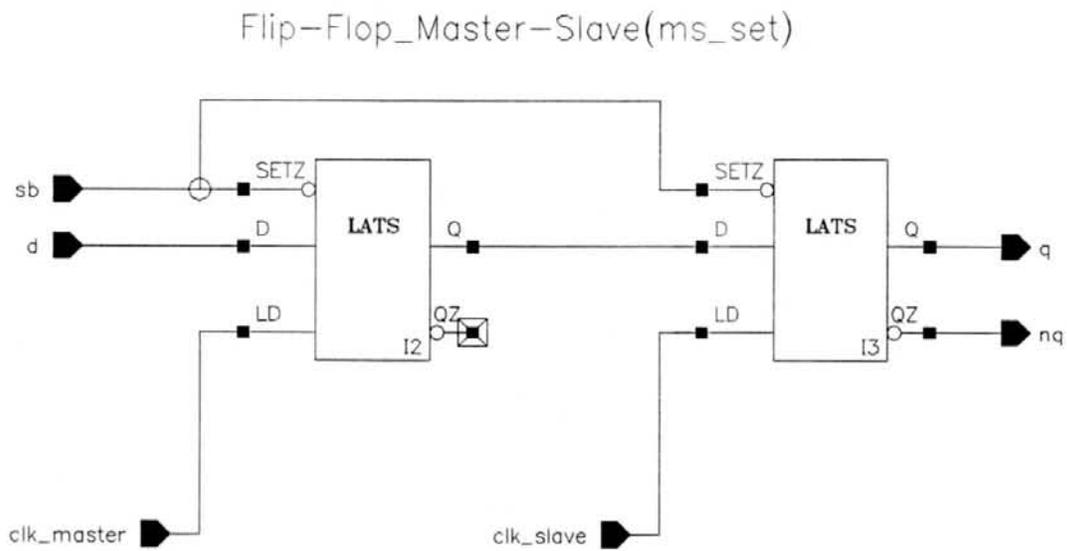


Figura 2.9 - Módulo *ms set*; (flip-flop tipo *D master/slave* com set).

**ANEXO 3    SUBCIRCUITOS PARA O**  
*BUFFER ELÁSTICO DE*  
**AGREGADO**

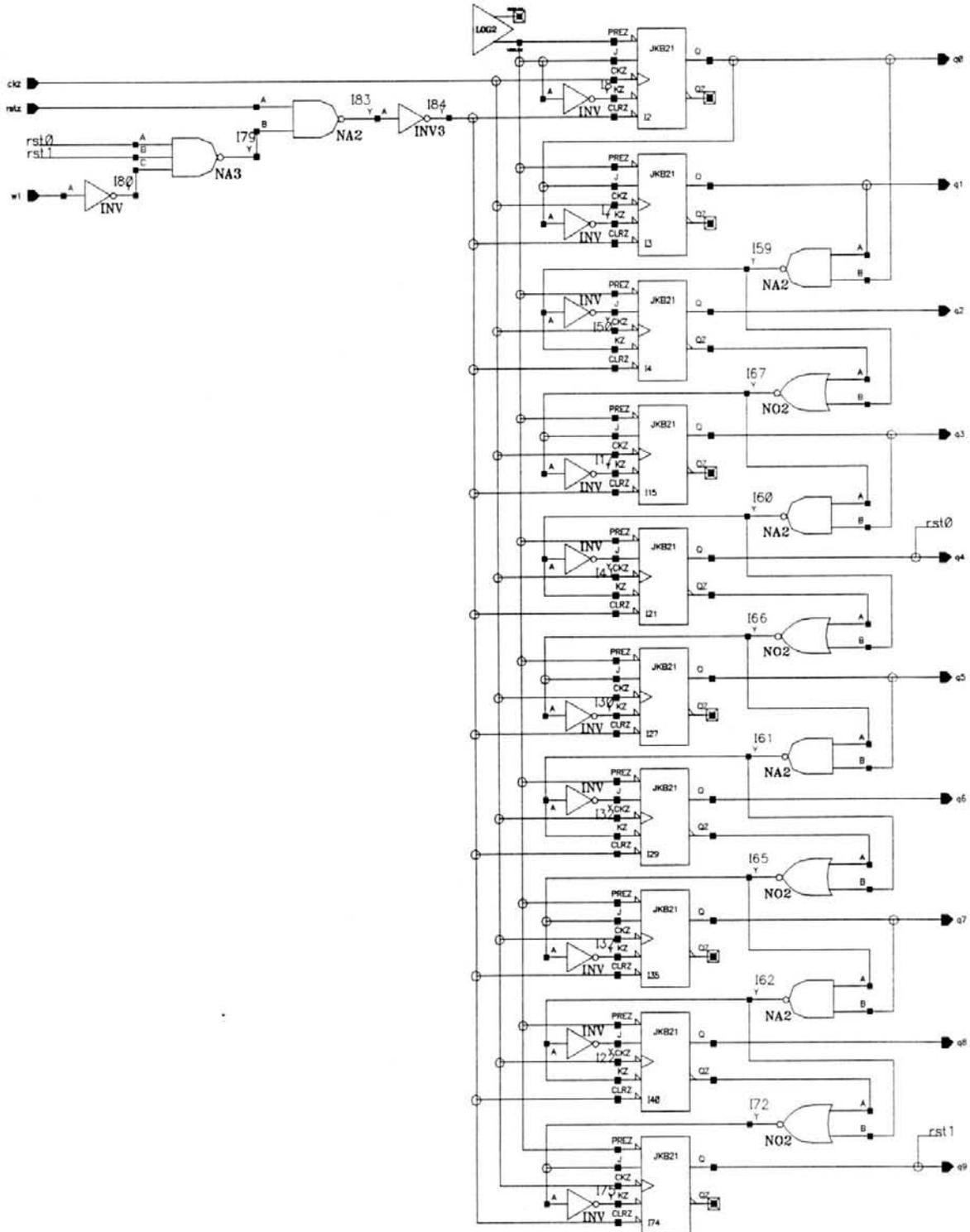


Figura 3.1 - Módulo *wr.add.gen*; (gerador de endereços de escrita).

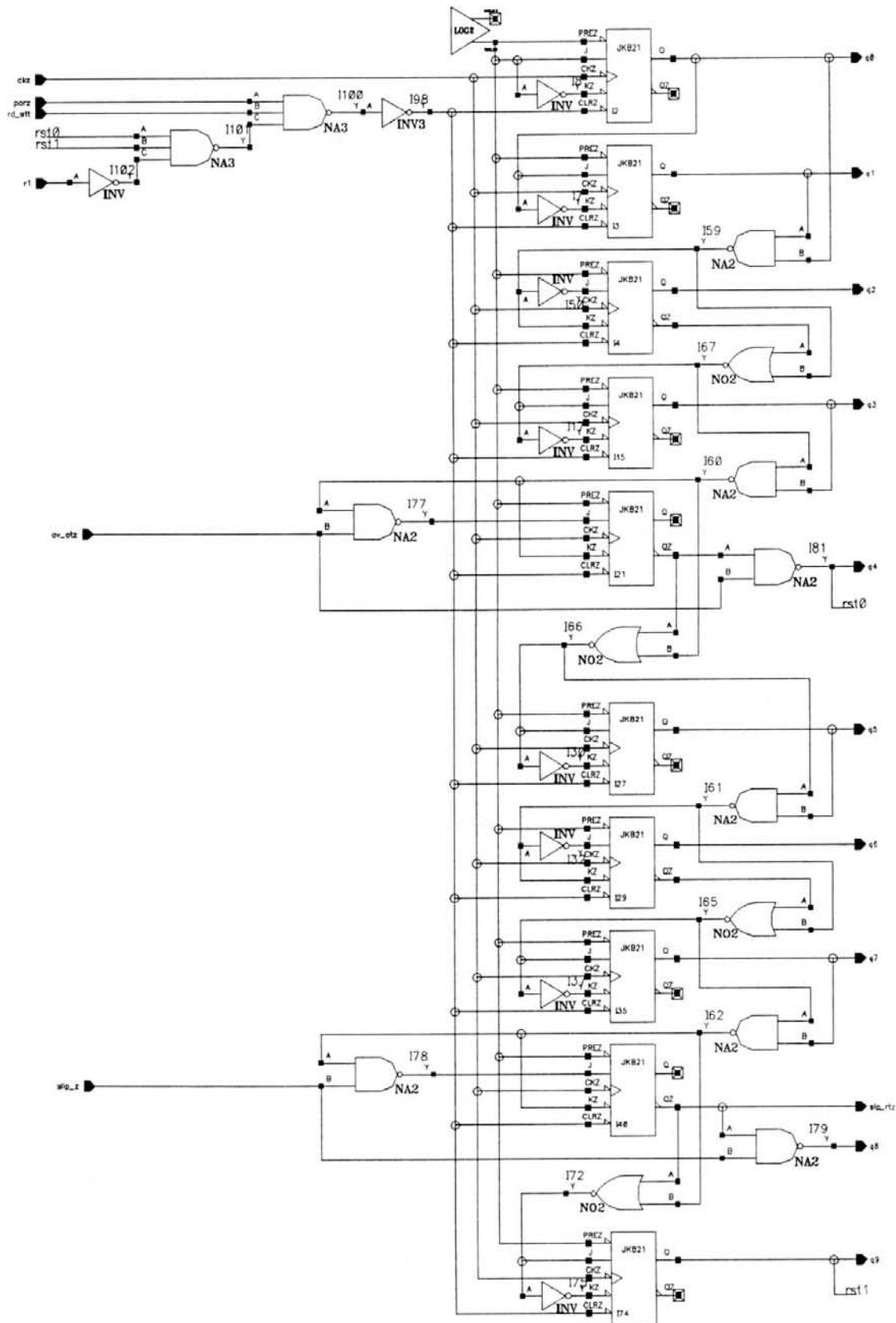


Figura 3.2 - Módulo *rd.add.gen*; (gerador de endereços de leitura).

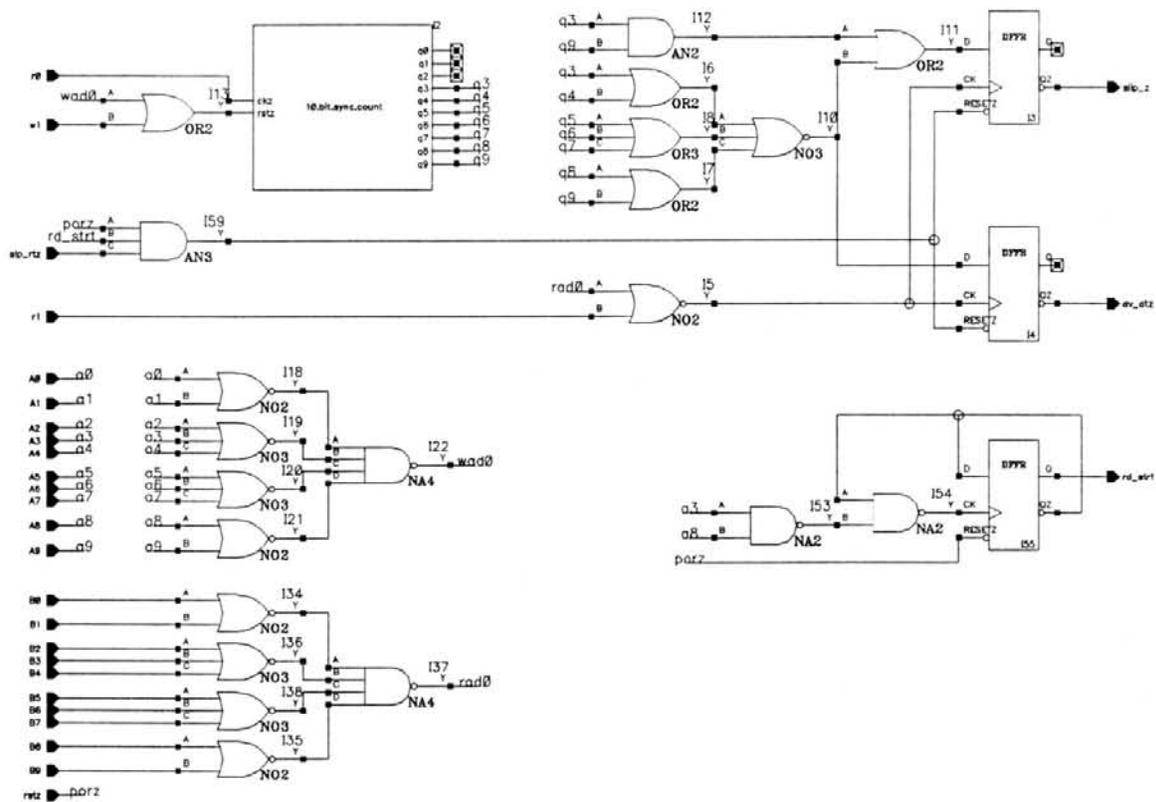


Figura 3.3 - Módulo *logic*; (lógica de controle do módulo *rd.add.gen*).

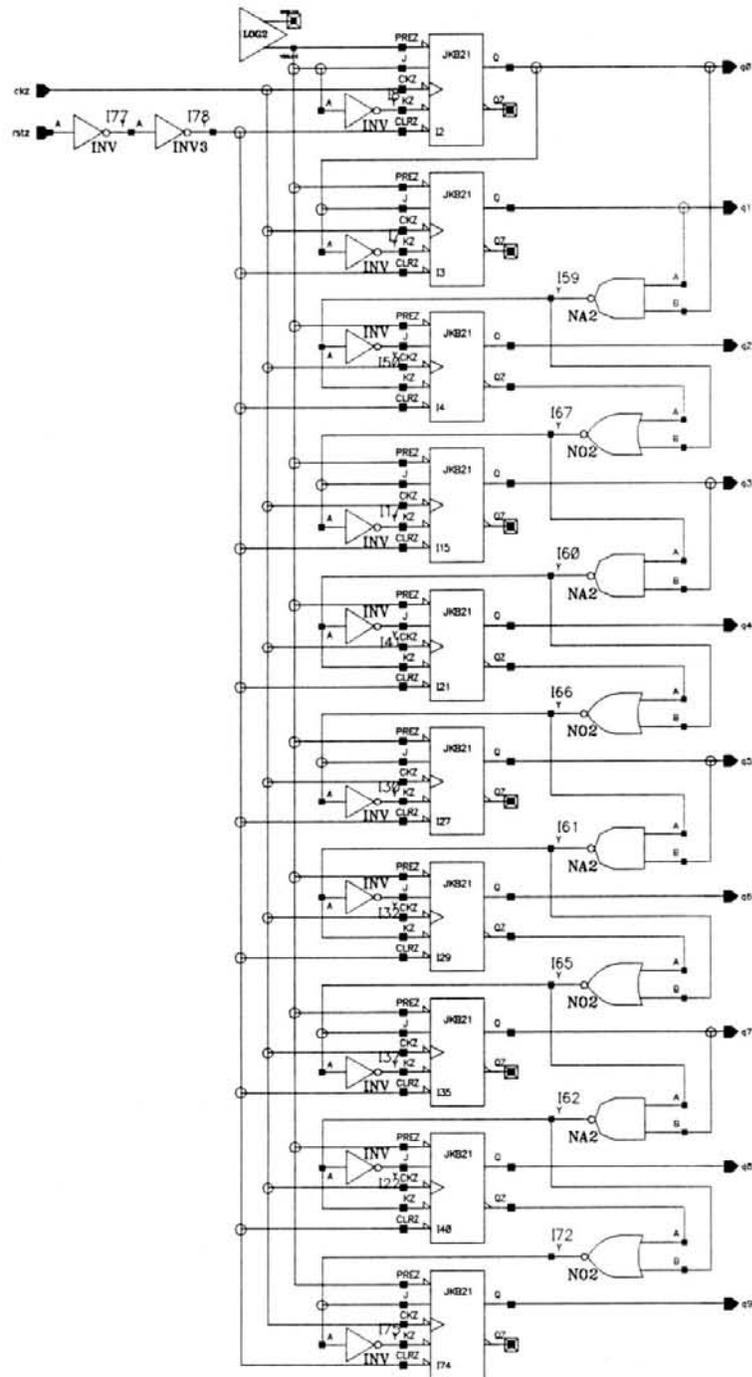


Figura 3.4 - Módulo *10.bit.sync.count*; (contador síncrono de 10 bits).

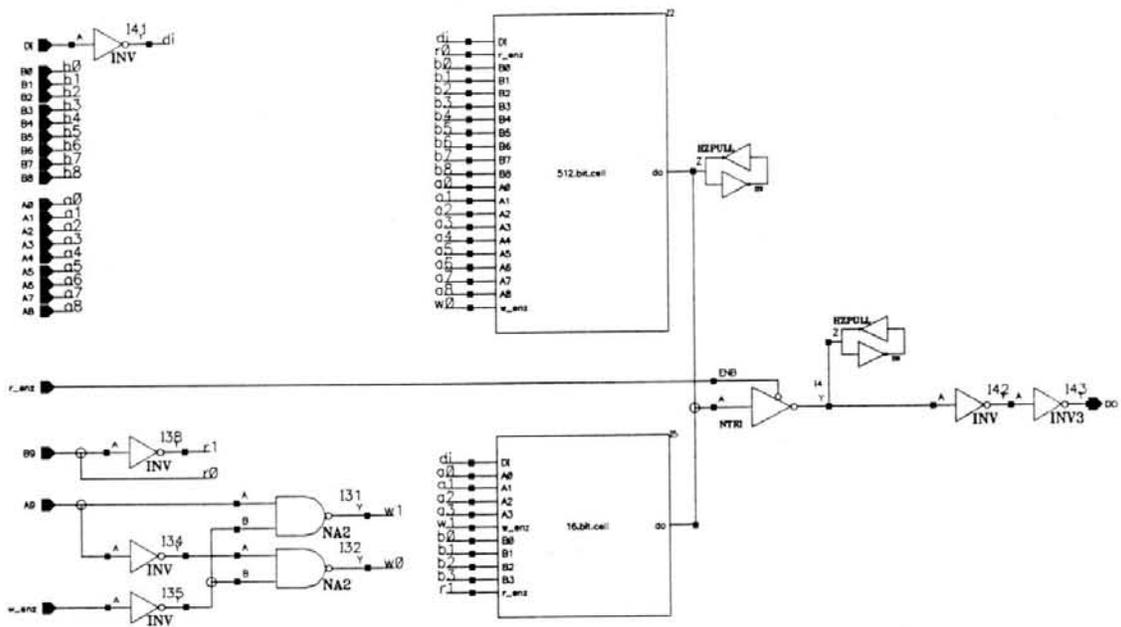


Figura 3.5 - Módulo 528.bit.cell; (célula de armazenagem; capacidade: 528 bits).

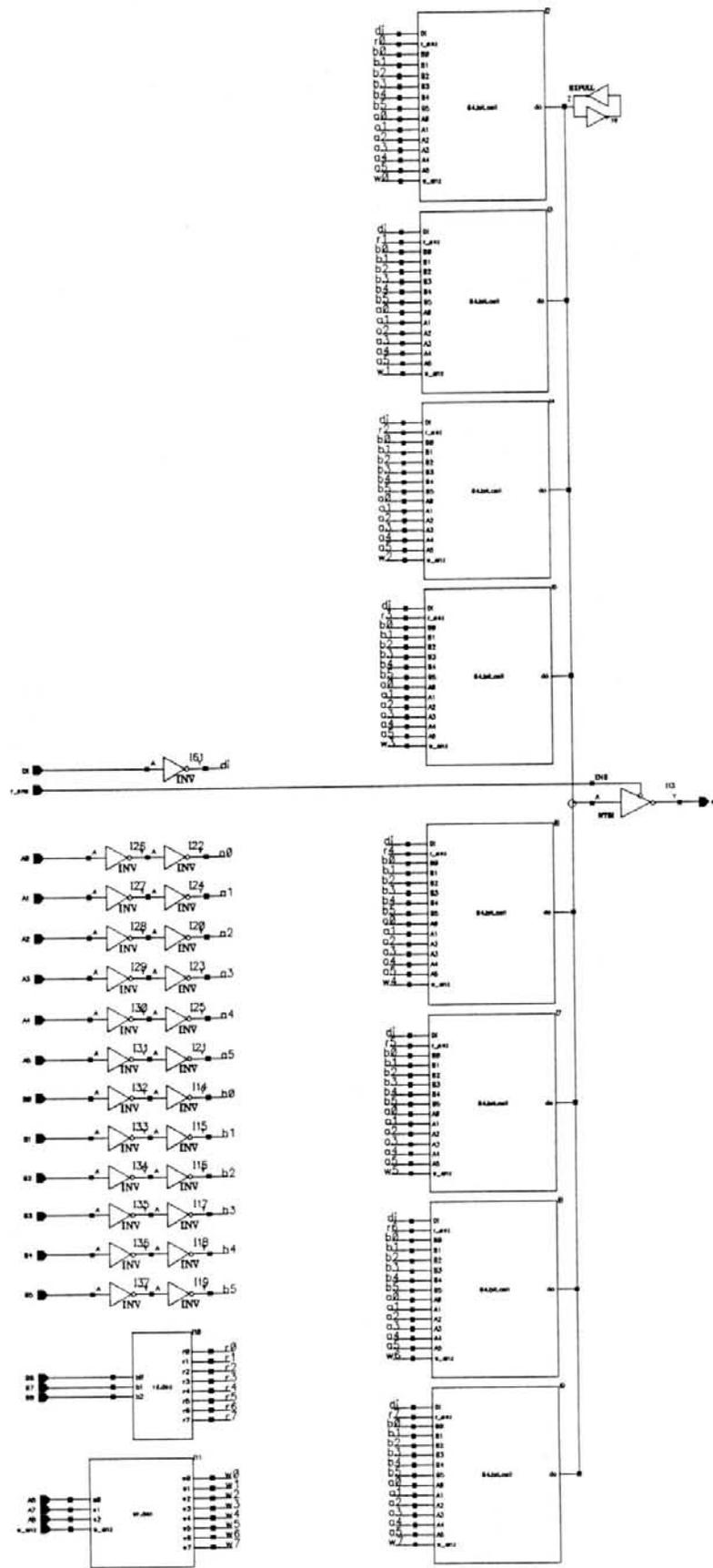


Figura 3.6 - Módulo 512.bit.cell; (célula de armazenagem; capacidade: 512 bits).

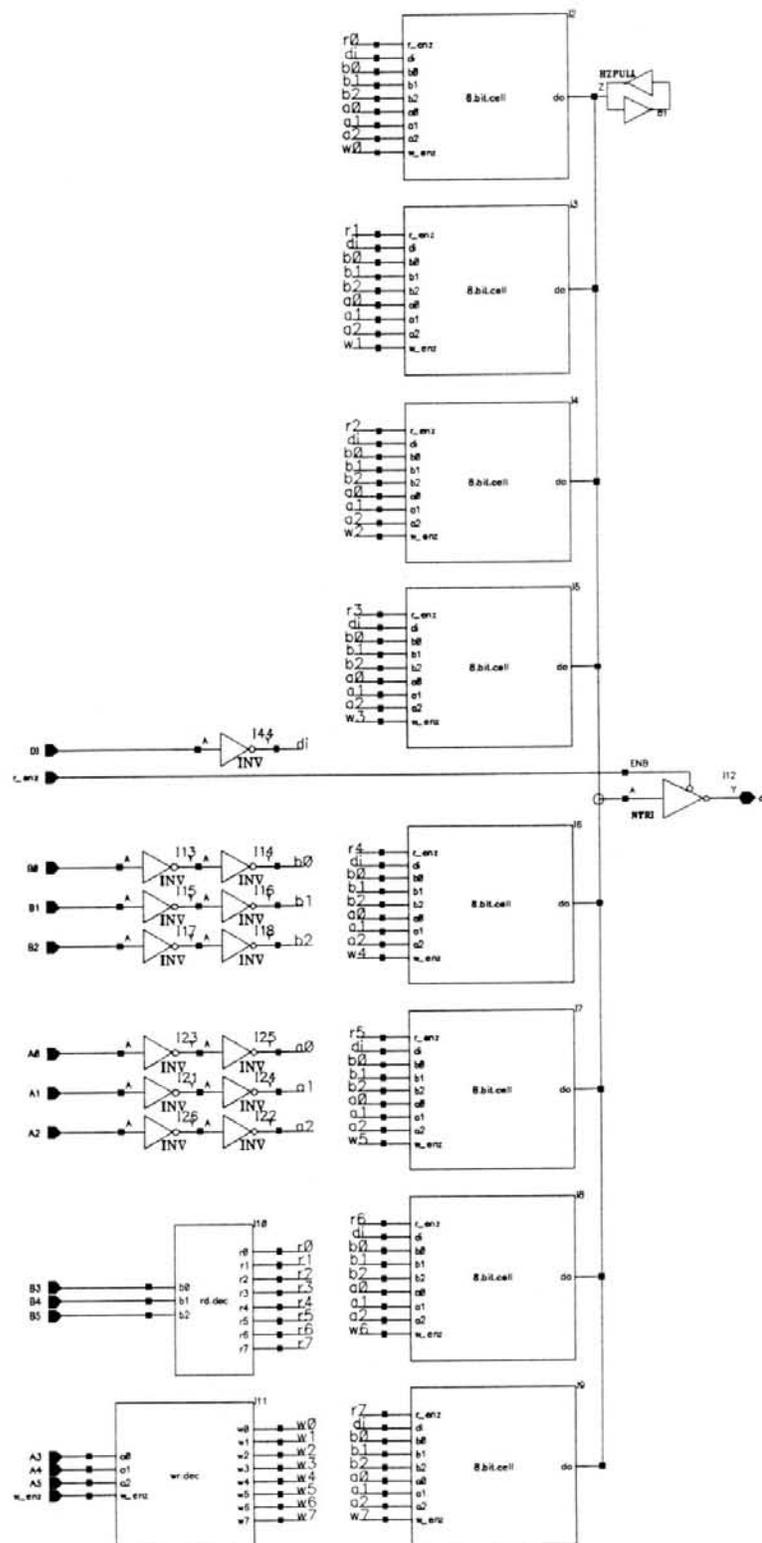


Figura 3.7 - Módulo *64.bit.cell*; (célula de armazenagem; capacidade: 64 bits).

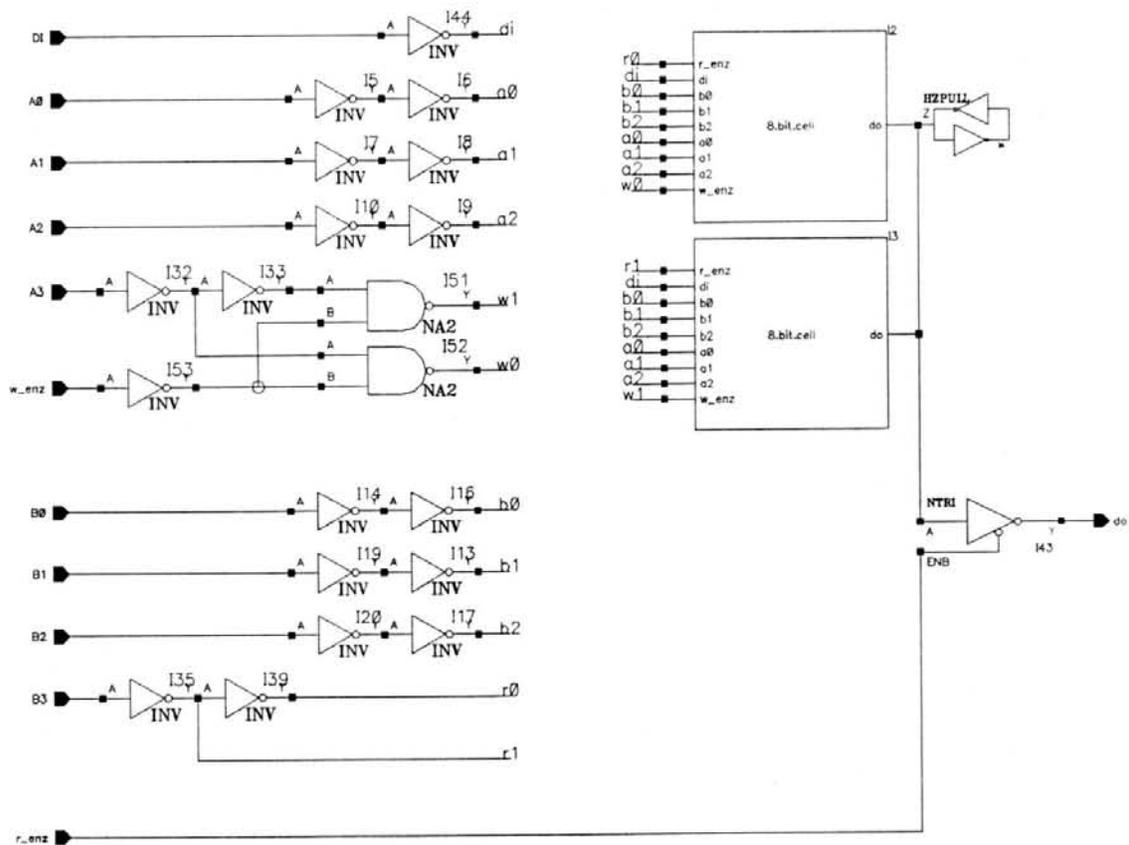


Figura 3.8 - Módulo *16.bit.cell*; (célula de armazenagem; capacidade: 16 bits).

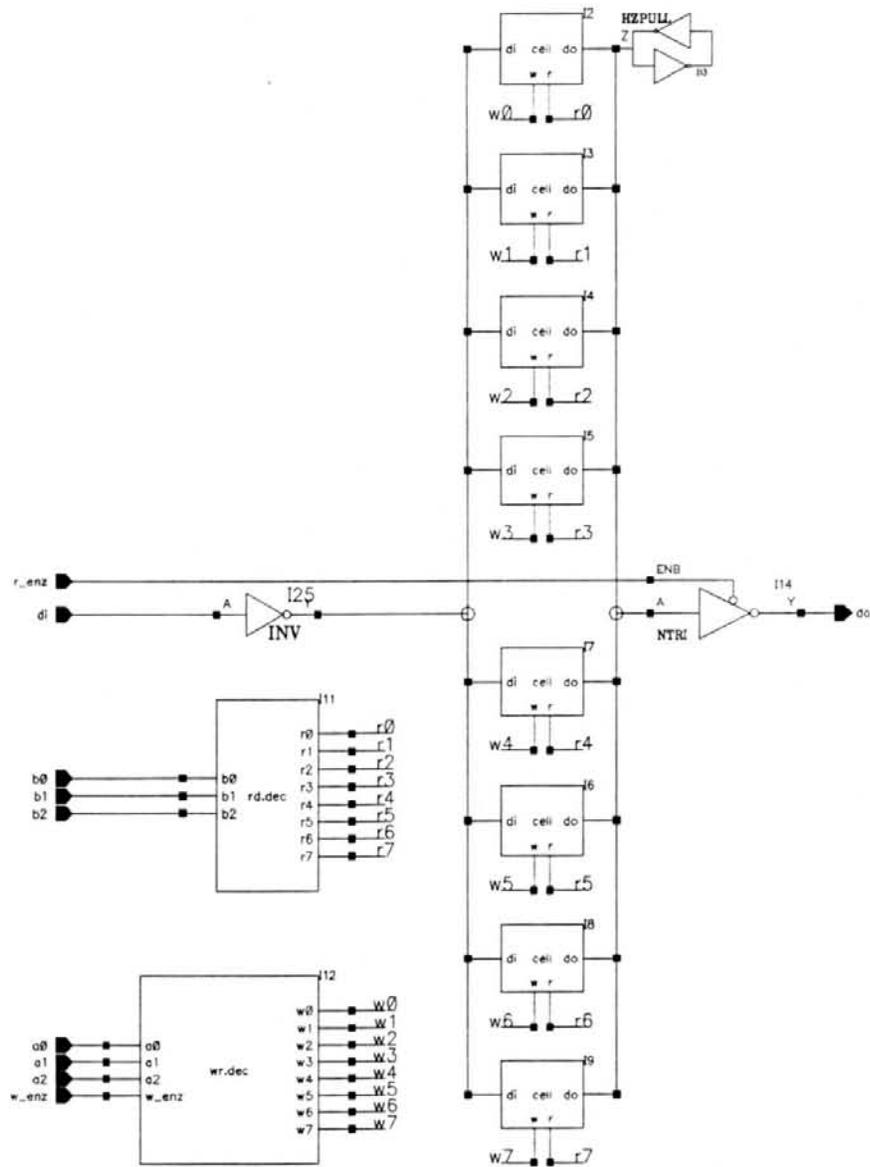


Figura 3.9 - Módulo *8.bit.cell*; (célula de armazenagem; capacidade: 8 bits).

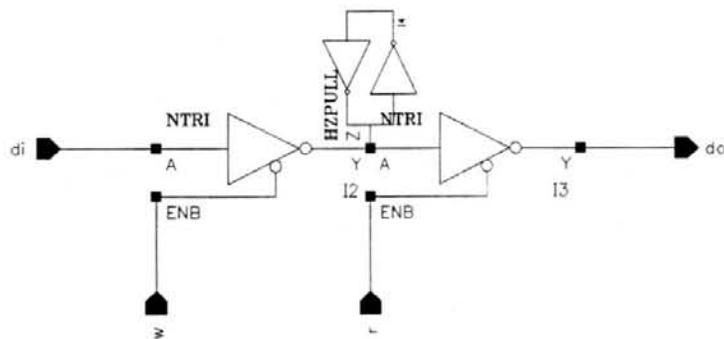


Figura 3.10 - Módulo *cell*; (célula de armazenagem; capacidade: 1 bit).

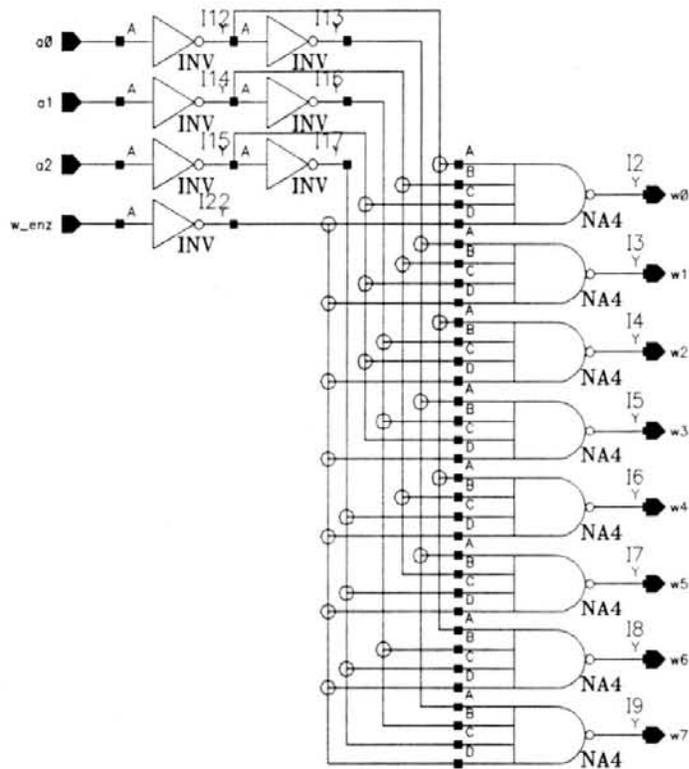


Figura 3.11 - Módulo *wr.dec*; (decodificador de escrita).

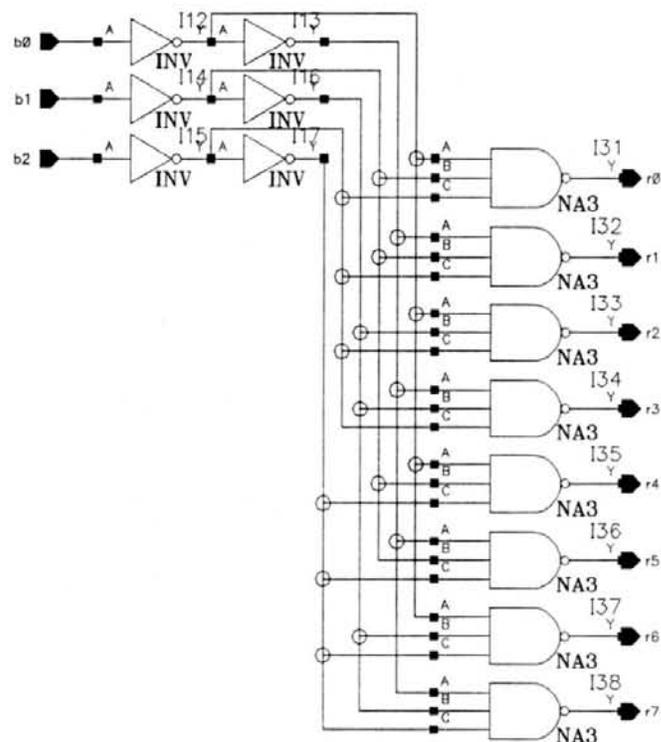


Figura 3.12 - Módulo *rd.dec*; (decodificador de leitura).

**ANEXO 4    SUBCIRCUITOS PARA O  
MÓDULO DE ALINHAMENTO  
DE QUADRO**

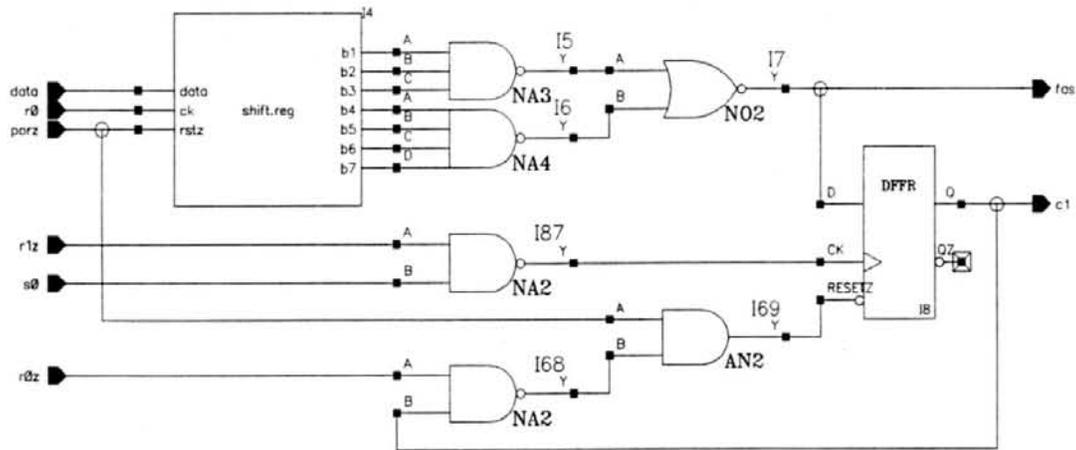


Figura 4.1 - Módulo *step1*; (verificador da primeira condição do procedimento de alinhamento de quadro).

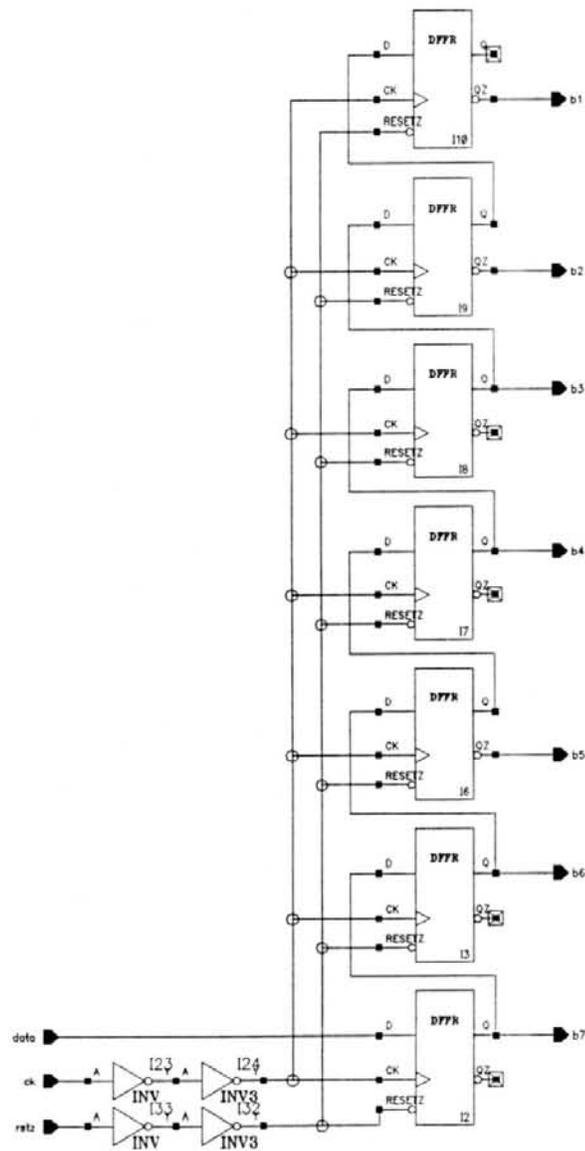


Figura 4.2 - Módulo *shift.reg*; (registrador de deslocamento).

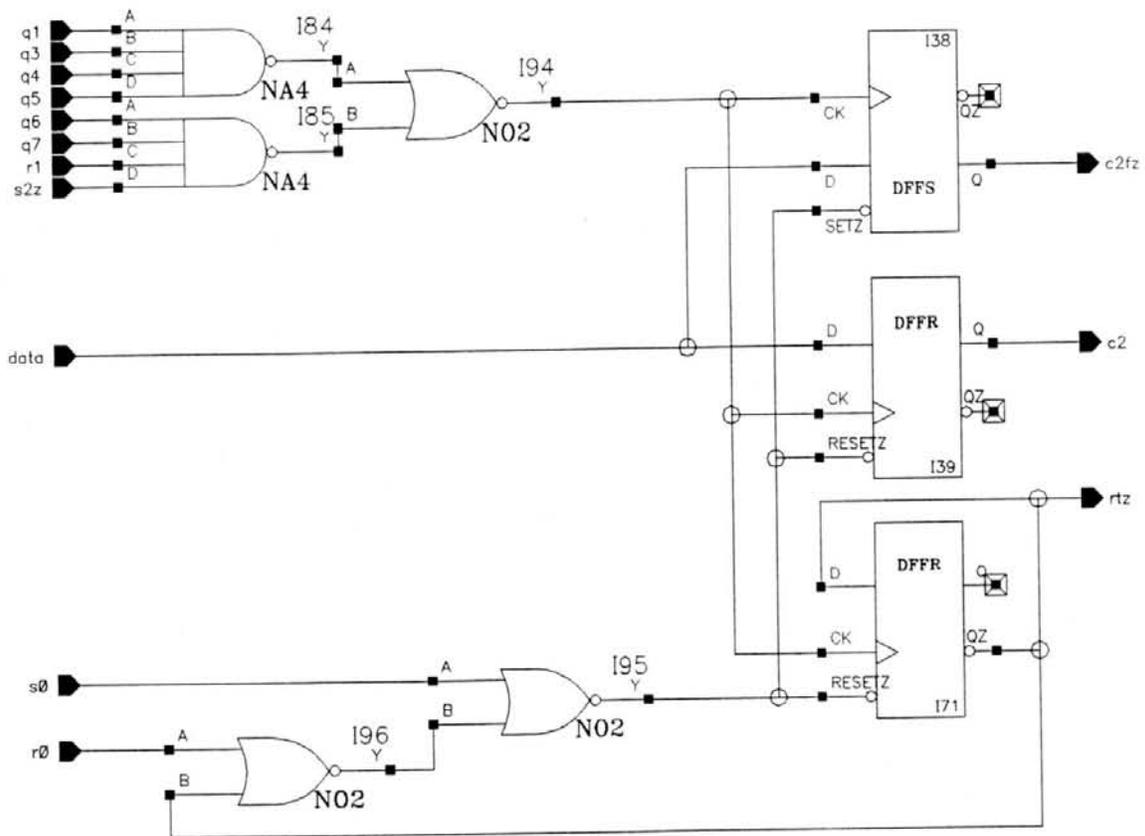


Figura 4.3 - Módulo *step2*; (verificador da segunda condição do procedimento de alinhamento de quadro).

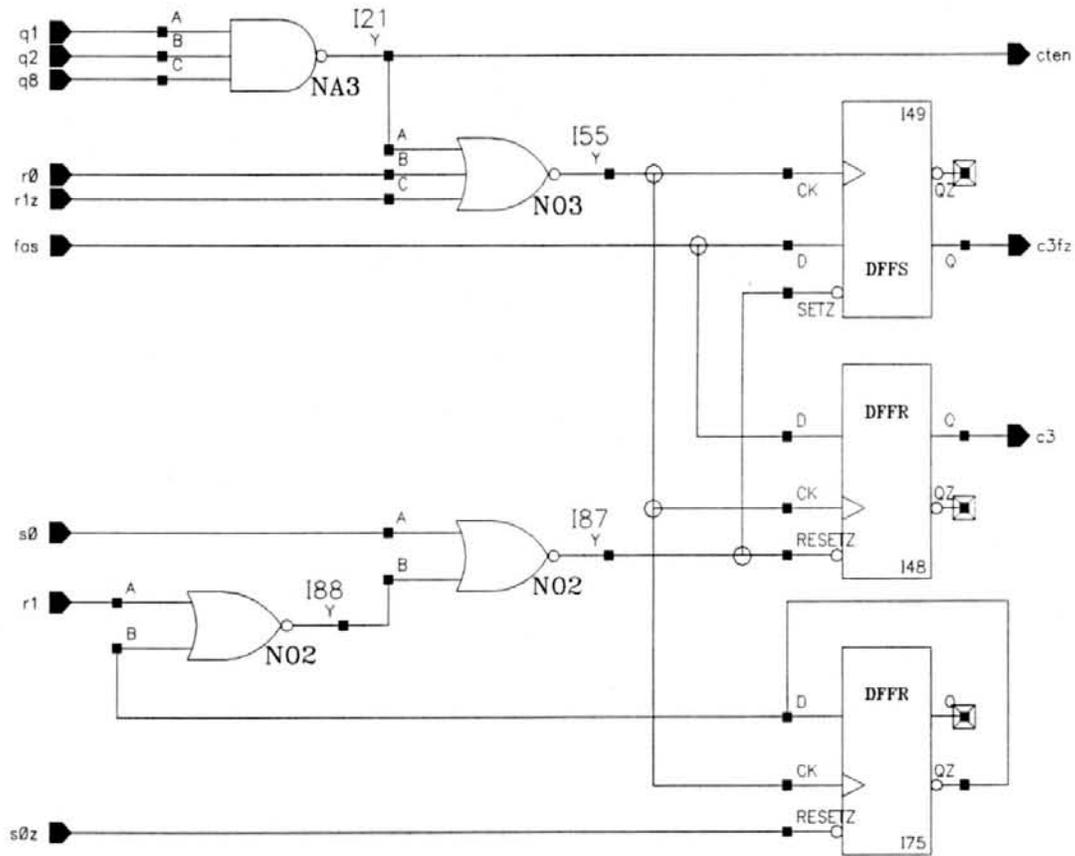


Figura 4.4 - Módulo *step3*; (verificador da terceira condição do procedimento de alinhamento de quadro).

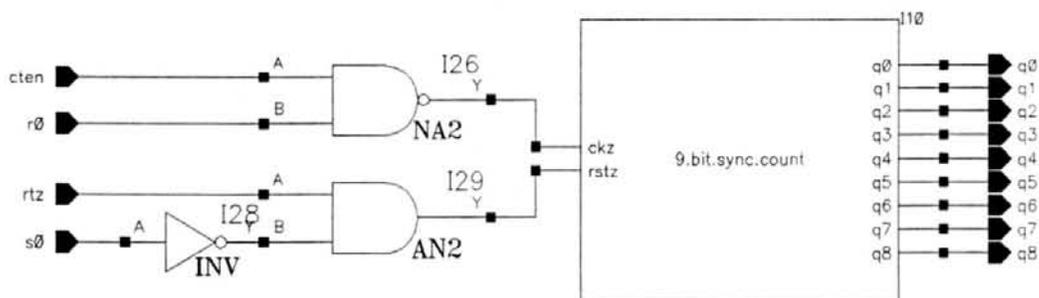


Figura 4.5 - Módulo *counter*; (contador temporizador).



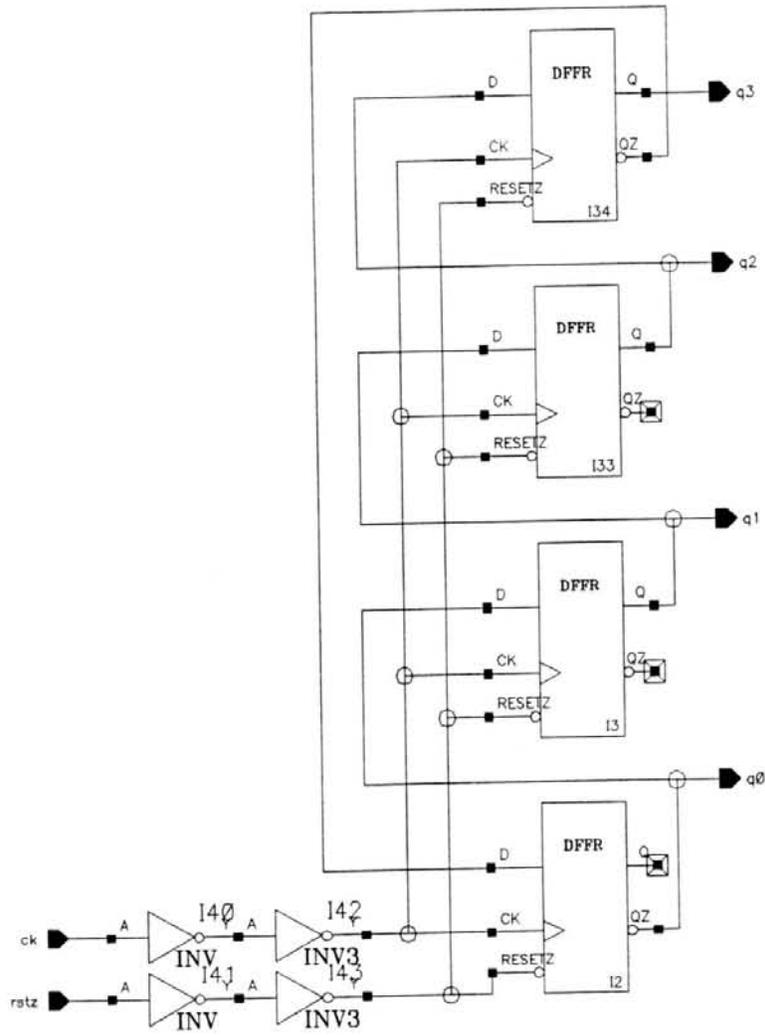


Figura 4.8 - Módulo *j.count*; (contador *johnson*).

**ANEXO 5**    **SUBCIRCUITOS PARA O**  
**MÓDULO GERADOR DE**  
*TIME-SLOTS*

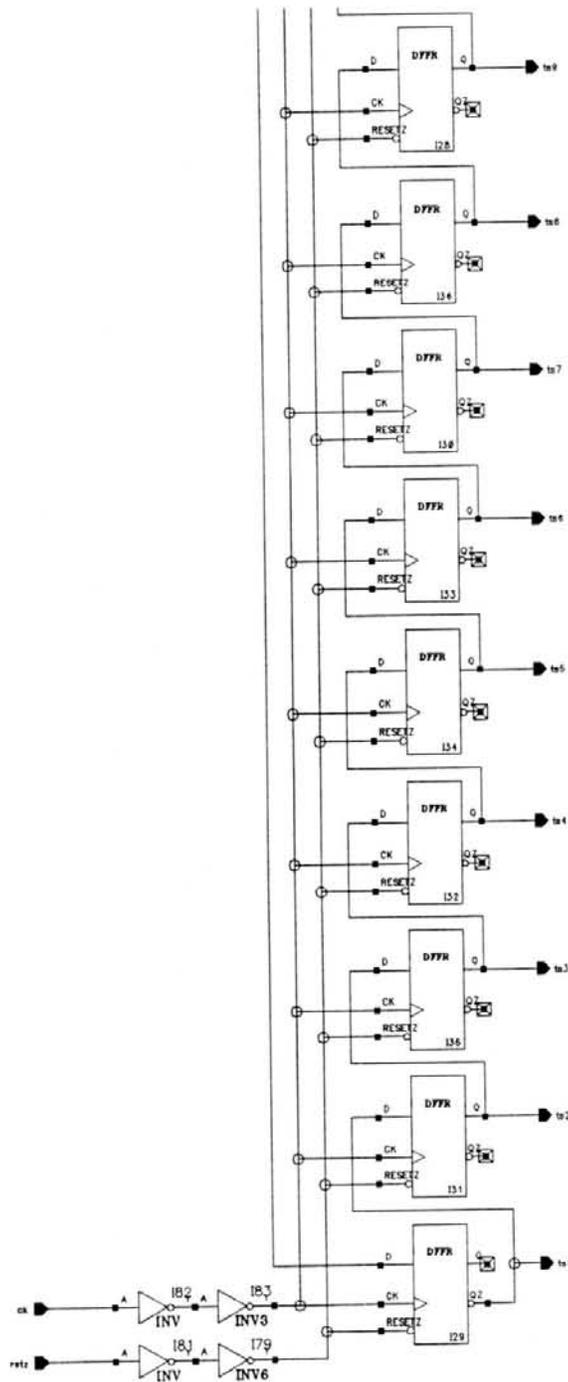


Figura 5.1 - Módulo *ring.counter*; (contador em anel; esquema parcial 1 de 4).

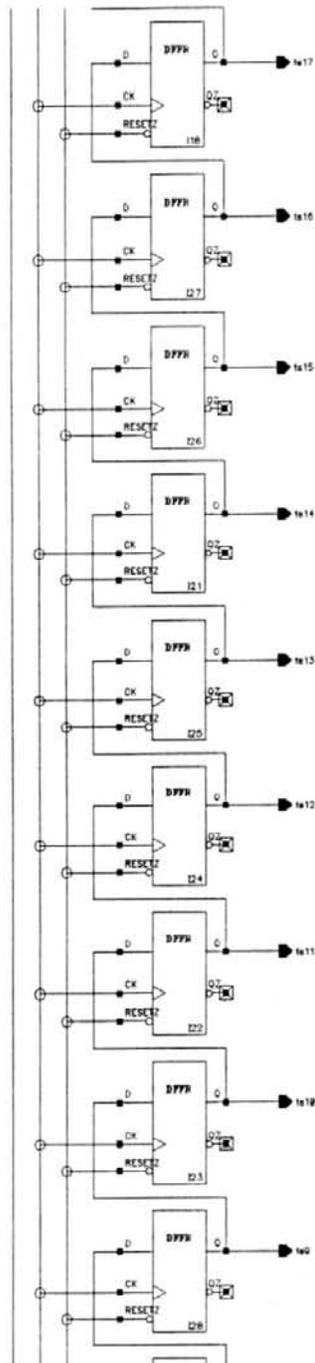


Figura 5.2 - Módulo *ring.counter*; (contador em anel; esquema parcial 2 de 4).

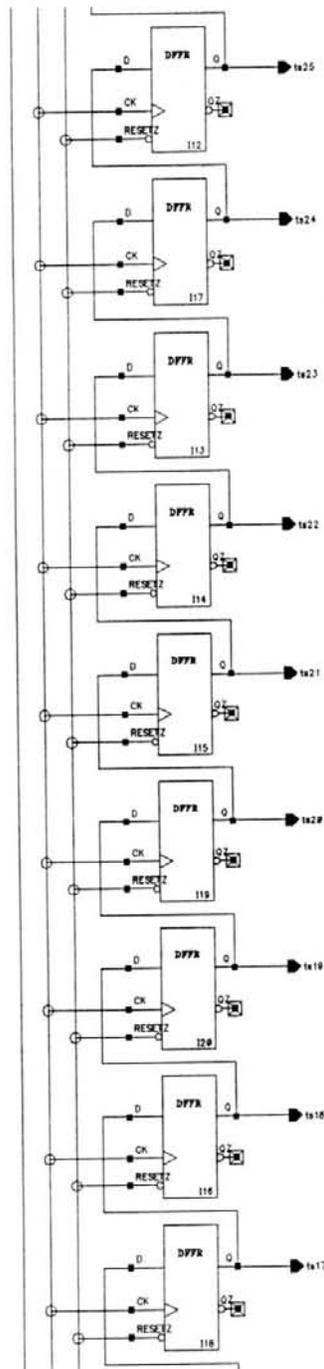


Figura 5.3 - Módulo *ring.counter*; (contador em anel; esquema parcial 3 de 4).

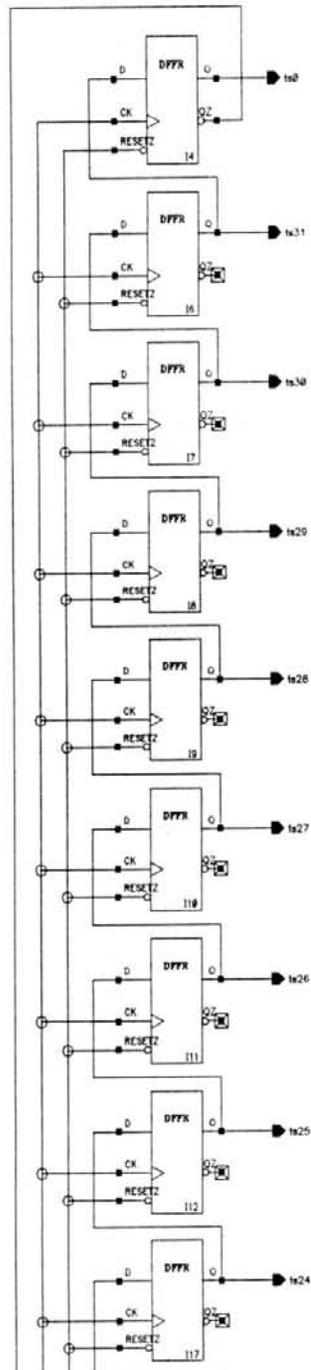


Figura 5.4 - Módulo *ring.counter*; (contador em anel; esquema parcial 4 de 4).

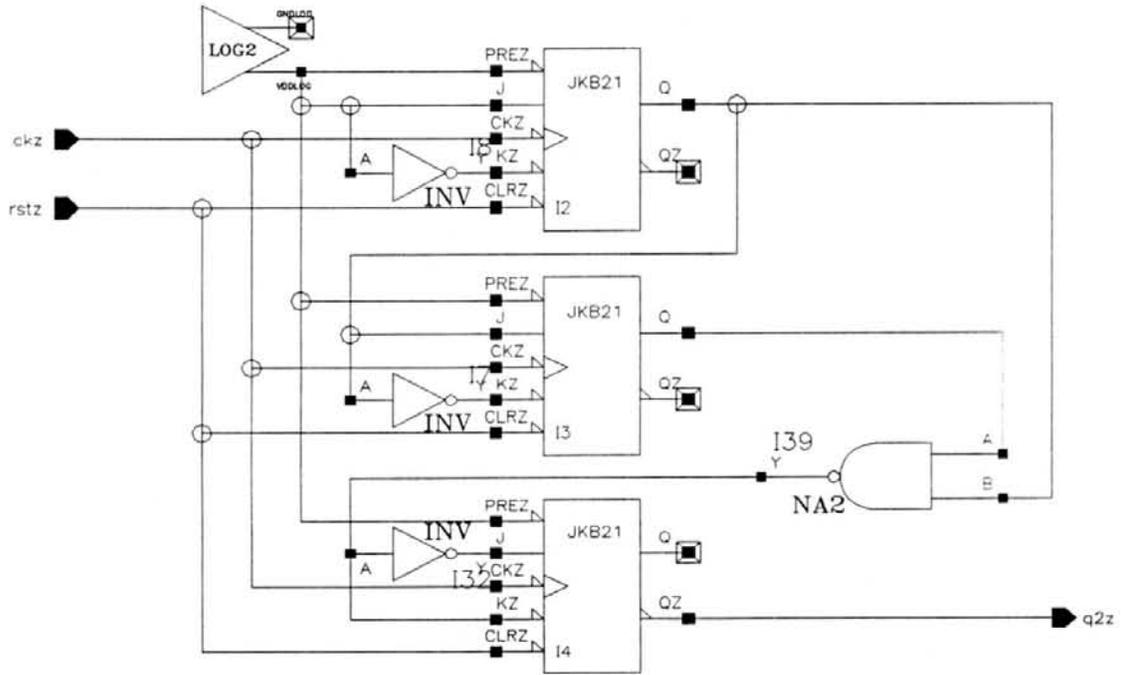


Figura 5.5 - Módulo *3.bit.sync.count*; (contador síncrono de 3 bits).

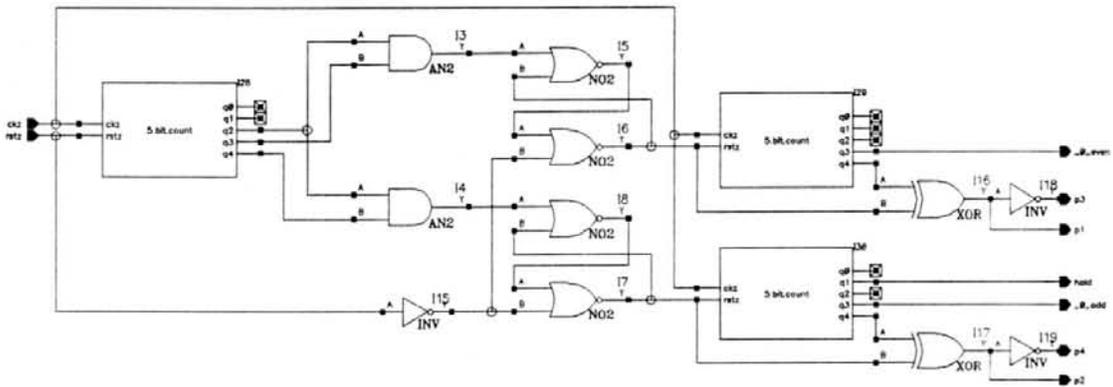


Figura 5.6 - Módulo *tmg.gen*; (gerador de temporização para a transmissão de 64 kbit/s).

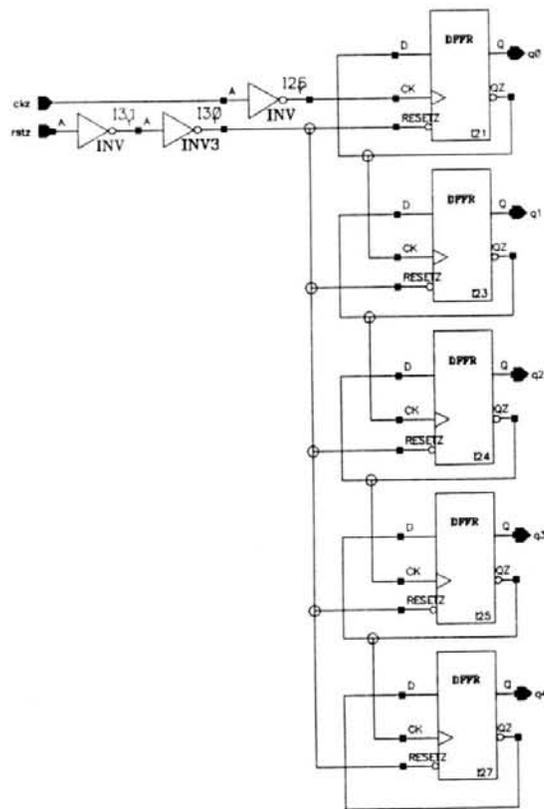


Figura 5.7 - Módulo *5.bit.count*; (contador assíncrono de 5 bits).

**ANEXO 6    SUBCIRCUITOS PARA OS  
MÓDULOS DE TRANSMISSÃO  
DE TRIBUTÁRIOS**

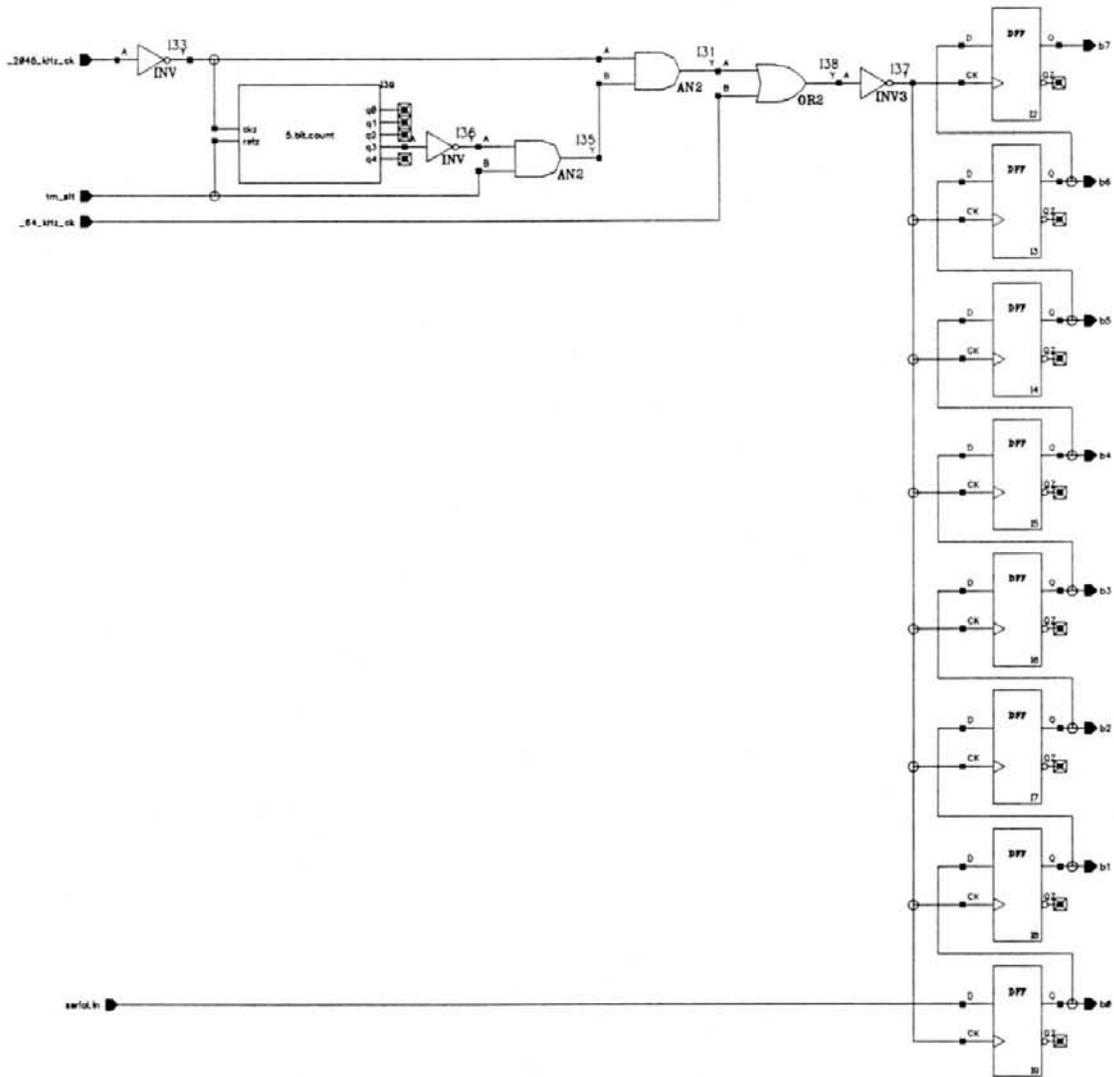


Figura 6.1 - Módulo *8.bit.shift.reg*; (registrador de deslocamento de 8 bits).

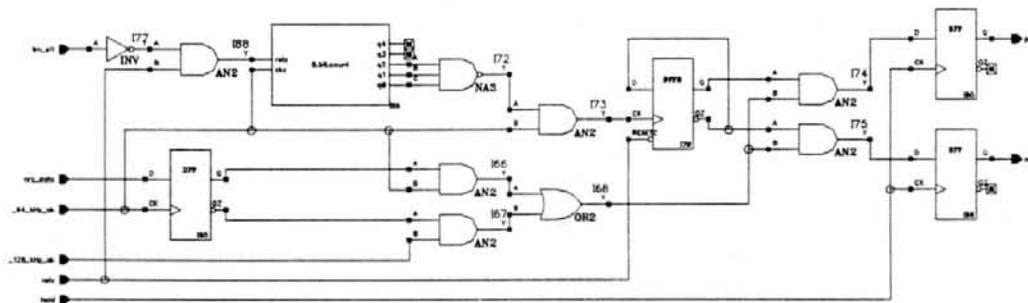


Figura 6.2 - Módulo *line.coder*; (codificador *G.703*).

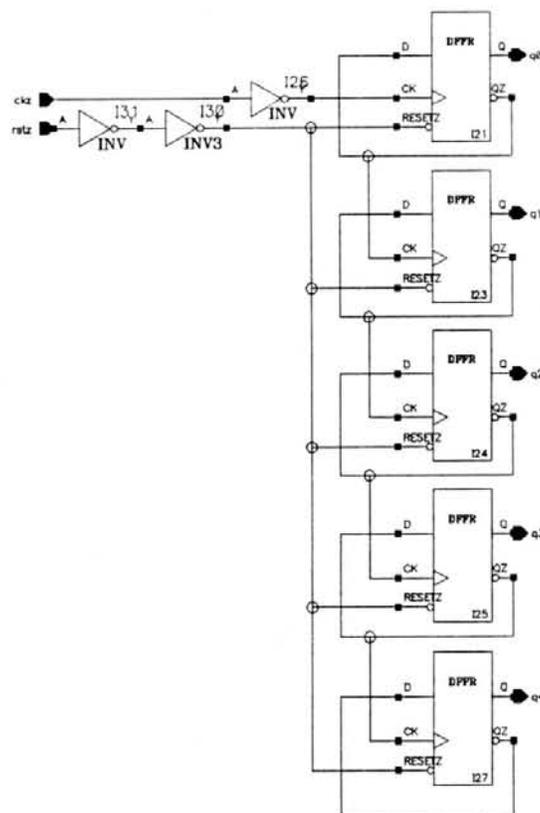


Figura 6.3 - Módulo *5.bit.count*; (contador assíncrono de 5 bits).

## BIBLIOGRAFIA

- [AIT 94] AITA, A.L. et al. CHIP-SET para Modens Banda Base Transmitindo em Par Trançado de Linha de Assinante. In: SBMICRO, 9., 1994, Rio de Janeiro. **Anais....** Rio de Janeiro: Lis, 1994. p.74-82.
- [ANN 86] ANNARATONE, M. **Digital CMOS Circuit Design**. Massachusetts: Kluwer Academic, 1986. 344p.
- [BAP 90] BAPTISTELLA, L.F.B.; LOBO, M.P.L. **Rede Digital de Serviços Integrados**. São Paulo: McGraw-Hill, 1990. 313p.
- [BAT 92] BATES, B. **Introduction to T1/T3 Networking**. Boston: Artech House, 1992. 192p.
- [BCR 90] BELL COMMUNICATION RESEARCH TECHNICAL PERSONNEL. **Telecommunications Transmission Engineering vol.1**. USA: Bellcore, 1990. 787p.
- [BCR 90a] BELL COMMUNICATION RESEARCH TECHNICAL PERSONNEL. **Telecommunications Transmission Engineering vol.2**. USA: Bellcore, 1990. 732p.
- [BEL 91] BELLAMY, J. **Digital Telephony**. New York: John Willey & Sons, 1991. 572p.
- [BER 93] BERENDONK, C.H.C. **A Rede Digital da Embratel com Equipamentos E1**. Rio de Janeiro: Embratel-DTC41, 1993. 19p.
- [BUR 92] BURES, K.J. Understanding Timing Recovery and Jitter in Digital Transmission Systems - Part 1. **RF Design**, Atlanta, GA, p.42-53, Oct.1992.

- [BUR 92a] BURES, K.J. Understanding Timing Recovery and Jitter in Digital Transmission Systems - Part 2. **RF Design**, Atlanta, GA, p.27-36, Nov.1992.
- [BYR 63] BYRNE, C.J.; KARAFIN B.J.; ROBINSON D.B. Systematic Jitter in a Chain of Digital Repeaters. **Bell System Technical Journal**, New York, v.42, p.2679-2714, Nov.1963.
- [CAD 88] CADENCE SYSTEMS INC. **Design Framework Manual**. USA: SDA Systems, Inc., 1988.
- [COR 93] CORREIA, M.; BERNAL, P. **Rede Corporativa Integrada**. São Paulo: Érica, 1993. 150p.
- [DAV 90] DAVIDSON, R.; MULLER, N.J. **LANs to WANs: Network Management in the 1990's**. Boston: Artech House, 1990. 541p.
- [EMB 92] EMBRATEL. **Especificações Gerais de Equipamentos Multiplex de 2048 kbit/s Sistema de Práticas Telebrás**. Brasília: Embratel, 1992. 235p.
- [EMB 94] EMBRATEL. **Rede de Serviços Digitais E1: Apostila de Treinamento**. Curitiba: Embratel, 1994. 115p.
- [EQU 94] EQUITEL SA. **Multiplexação Plesiócrons: Apostila de Treinamento**. Curitiba: Equitel, 1994. 58p.
- [ES2 89] EUROPEAN SILICON STRUCTURES. **Solo 2000 Standard Cells Family Libraries Manual**. France: European Silicon Structures, 1989. 207p.
- [GIO 86] GIOZZA, W.; ARAÚJO, J.; SAUVÉ, J. **Redes Locais de Computadores**. São Paulo: Mc Graw Hill, 1986. 399p.
- [HEL 86] HELD, G. **Data Communications Networking Devices**. England: John Wiley and Sons, 1986. 372p.

- [ITU 88] INTERNATIONAL TELECOMMUNICATION UNION. **General Aspects of Digital Transmission Systems; Terminal Equipments; Recommendations G.700-G.795**. Melbourne: 614p. IXth Plenary Assembly, 1988.
- [ITU 88a] INTERNATIONAL TELECOMMUNICATION UNION. The Control of Jitter and Wander within Digital Networks which are Based on the 2048 kbit/s Hierarchy. **Recommendation G.823**. Melbourne: ITU-T, 1988. p.38-50. Fascicle III.5.
- [ITU 88b] INTERNATIONAL TELECOMMUNICATION UNION. Considerations On Timing And Synchronization Issues. **Recommendation G.810**. Melbourne: ITU-T, 1988. p.13-16. Fascicle III.5.
- [ITU 88c] INTERNATIONAL TELECOMMUNICATION UNION. Timing Requirements at the Outputs of Primary Reference Clocks Suitable for Plesiochronous Operation of International Digital Links. **Recommendation G.811**. Melbourne: ITU-T, 1988. p.16-21. Fascicle III.5.
- [KAI 94] KAISER, G.W. **Multiplexação de Dados em Alta Velocidade**. Porto Alegre: CPGCC da UFRGS, Fev.1994. 47p. (Trabalho Individual).
- [KAI 95] KAISER, G.W.; WAGNER, T.V.; ROCHOL, J. Time Alignment of Plesiochronous and Mesochronous Signals in Digital Data Communication Systems. In: SBMICRO, 10., 1995, Canela, RS. **Anais....** Porto Alegre: Palotti, 1995. v.1, p.223-231.
- [LEE 90] LEE, E.A.; MESSERSCHIMITT, D.G. **Digital Communication**. Boston: Kluwer Academic, 1990. 713p.
- [MEN 91] MENG, T.H. **Synchronization Design for Digital Systems**. Boston: Kluwer Academic, 1991. 175p.

- [OWE 82] OWEN, F.F.E. **PCM and Digital Transmission Systems**. New York: Mc Graw Hill, 1982. 295p.
- [PAN 72] PAN J.W. Synchronizing and Multiplexing in a Digital Communications Network. **Proceedings of the IEEE**, New York, v.60, p.594-601, May 1972.
- [RIB 80] RIBEIRO, Marcello P.; BARRADAS, Ovídeo C.M. **Telecomunicações - Sistemas Analógico-digitais**. Rio de Janeiro: LTC, 1980. 1174p.
- [SHO 88] SHOJI, M. **CMOS Digital Circuits Technology**. New Jersey: Prentice Hall, 1988. 434p.
- [SIL 88] SIMUCAD INC. **SILOS User's guide**. California: Simucad, Inc., 1988.
- [SOA 95] SOARES, L.F.G.; LEMOS, G.; COLCHER, S. **Redes de Computadores**. Rio de Janeiro: Campus, 1995. 576p.
- [SPO 93] SPOHN, D.L. **Data Network Design**. New York: McGraw-Hill, 1993. 778p.
- [TAN 88] TANENBAUM, A. **Computer Networks**. New Jersey: Prentice Hall, 1988. 658p.
- [WAK 90] WAKERLY, J. **Digital Design-Principles and Practices**. New Jersey: Prentice Hall, 1990. 716p.



Funções de Multiplexação E1

por

Gustavo Weymar Kaiser

Dissertação apresentada aos Senhores:

Dr. Fernando Gehm/Moraes

Prof. Dr. José Roberto Boisson de Marca (PUC/RJ)

Prof. Dr. Carlos Eduardo Pereira (DELET/UFRGS)

Prof. Juergen Rochol

Vista e permitida a impressão.

Porto Alegre, 27 / 08 / 96.

Prof. Tiaraju Vasconcellos Wagner,  
Orientador.