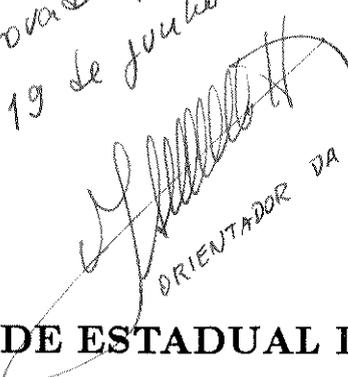


Este exemplar corresponde à  
redação final da tese defendida por  
Luís Roberto Ferreira e aprovada pela  
Comissão Julgadora em 19 de junho de 1989

  
ORIENTADOR DA TESE

**UNIVERSIDADE ESTADUAL DE CAMPINAS**

**Faculdade de Engenharia Elétrica**

**Dissertação de Mestrado:**

*Uma Proposta de Integração VLSI do Protocolo da*

*Camada de Enlace na Sinalização da RDSI*

**Orientador : Prof. Dr. Márcio Luiz de Andrade Netto**

**Orientado : Luís Roberto Ferreira**

Campinas, junho de 1989



# Conteúdo

|          |                                                                         |           |
|----------|-------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Introdução</b>                                                       | <b>1</b>  |
| 1.1      | Considerações sobre a RDSI . . . . .                                    | 1         |
| 1.1.1    | Objetivos e princípios da RDSI . . . . .                                | 2         |
| 1.1.2    | A Série I de Recomendações . . . . .                                    | 3         |
| 1.1.3    | Serviços de Assinante . . . . .                                         | 3         |
| 1.1.4    | Acesso do usuário à rede . . . . .                                      | 6         |
| 1.1.5    | Configuração de referência CCITT . . . . .                              | 8         |
| 1.2      | Resumo dos Capítulos Seguintes . . . . .                                | 10        |
| <b>2</b> | <b>Noções de Interesse para o Trabalho</b>                              | <b>14</b> |
| 2.1      | O modelo ISO e o modelo RDSI-CCITT . . . . .                            | 14        |
| 2.1.1    | O Modelo de Referência para Sistemas Abertos da ISO . . . . .           | 14        |
| 2.1.2    | Adequação dos modelos OSI e CCITT-RDSI . . . . .                        | 17        |
| 2.2      | Protocolos de Sinalização Via Canal D . . . . .                         | 23        |
| 2.2.1    | Especificação de Camada 1 . . . . .                                     | 23        |
| 2.2.2    | Especificação de Camada 2 . . . . .                                     | 24        |
| 2.2.3    | Especificação de Camada 3 . . . . .                                     | 24        |
| 2.3      | Representação de Protocolos . . . . .                                   | 25        |
| 2.4      | O Protocolo de Enlace no Canal D . . . . .                              | 26        |
| 2.4.1    | A camada de enlace do modelo OSI/ISO . . . . .                          | 26        |
| 2.4.2    | O protocolo LAPD . . . . .                                              | 28        |
| 2.5      | Microeletrônica e RDSI . . . . .                                        | 33        |
| 2.6      | O Particionamento Adotado e Outros Particionamentos Possíveis . . . . . | 34        |
| <b>3</b> | <b>Uma Proposta de Especificação</b>                                    | <b>37</b> |
| 3.1      | Introdução . . . . .                                                    | 37        |
| 3.1.1    | Visão Global . . . . .                                                  | 37        |
| 3.1.2    | Arquitetura Funcional . . . . .                                         | 38        |
| 3.1.3    | As Portas Seriais de Comunicação . . . . .                              | 39        |
| 3.1.4    | Aplicações Previstas para o CH2 . . . . .                               | 41        |
| 3.1.5    | Pinagem Sugerida . . . . .                                              | 43        |
| 3.2      | Descrição do Bloco de Interfaces . . . . .                              | 45        |
| 3.2.1    | Estudo das Interfaces . . . . .                                         | 45        |
| 3.2.2    | Chaveamento Interno dos Canais B . . . . .                              | 60        |
| 3.2.3    | Acesso aos Canais B pelo Processador . . . . .                          | 63        |
| 3.2.4    | Lógica de Interrupções . . . . .                                        | 66        |
| 3.3      | Descrição do Bloco Tratador de HDLC . . . . .                           | 68        |
| 3.3.1    | Ativação dos Modos de Operação do Controlador HDLC . . . . .            | 68        |
| 3.3.2    | Funções de HDLC . . . . .                                               | 70        |
| 3.3.3    | Transferência de Pacotes de Dados . . . . .                             | 75        |
| 3.3.4    | Tratamento do Campo de Endereço . . . . .                               | 76        |
| 3.3.5    | Temporização . . . . .                                                  | 78        |

|          |                                                                    |            |
|----------|--------------------------------------------------------------------|------------|
| 3.3.6    | Condições Iniciais do THDLC após Reset Hardware . . .              | 78         |
| 3.4      | Descrição do Bloco Tratador de LAPD (TLD) . . . . .                | 79         |
| 3.4.1    | Funções e Procedimentos . . . . .                                  | 79         |
| 3.4.2    | Comunicação com Níveis Superiores . . . . .                        | 80         |
| 3.4.3    | Comunicação com Entidade Par . . . . .                             | 81         |
| 3.4.4    | Parâmetros e Variáveis do LAPD . . . . .                           | 82         |
| 3.4.5    | O LAPD/TLD em LEDES . . . . .                                      | 84         |
| 3.5      | Descrição do Bloco de FIFOS . . . . .                              | 86         |
| 3.5.1    | Controle das R-FIFOs . . . . .                                     | 86         |
| 3.5.2    | Controle das X-FIFOs . . . . .                                     | 90         |
| 3.5.3    | Deteção de Colisão e Procedimentos . . . . .                       | 92         |
| 3.6      | Descrição do Conjunto de Registradores . . . . .                   | 93         |
| 3.6.1    | Registradores de Condições e Eventos Internos . . . . .            | 98         |
| 3.6.2    | Registradores de Controle de Operação do LAPD/HDLC . . . . .       | 103        |
| 3.6.3    | Registradores com Endereços do HDLC/LAPD . . . . .                 | 111        |
| 3.6.4    | Registradores com Campo de Controle do LAPD . . . . .              | 114        |
| 3.6.5    | Registradores do Campo de Dados do HDLC/LAPD . . . . .             | 114        |
| 3.6.6    | Registradores de Controle e Dados das Interfaces Seriais . . . . . | 115        |
| <b>4</b> | <b>Sugestões de Implementação</b>                                  | <b>119</b> |
| 4.1      | Projeto Hierárquico . . . . .                                      | 119        |
| 4.2      | Projeto Full Custom VLSI . . . . .                                 | 121        |
| 4.3      | Projeto Lógico Hierárquico do CH2 . . . . .                        | 123        |
| 4.3.1    | Bloco de Interface . . . . .                                       | 124        |
| 4.3.2    | Bloco THDLC . . . . .                                              | 125        |
| 4.3.3    | Bloco de FIFOs . . . . .                                           | 127        |
| 4.3.4    | Bloco TLD . . . . .                                                | 127        |
| 4.4      | Projeto do Bloco TLD . . . . .                                     | 127        |
| 4.4.1    | Metodologia Básica . . . . .                                       | 128        |
| 4.4.2    | Arquitetura Proposta ao TLD . . . . .                              | 131        |
| 4.4.3    | Programação da PLA que implementa o LAPD . . . . .                 | 139        |
| <b>5</b> | <b>Considerações de Software</b>                                   | <b>144</b> |
| 5.1      | Programação do CH2 . . . . .                                       | 144        |
| 5.1.1    | Porta B como interface IOM no Terminal . . . . .                   | 144        |
| 5.1.2    | Porta B como interface IOM na Central . . . . .                    | 147        |
| 5.1.3    | Porta B como interface HDLC . . . . .                              | 147        |
| 5.2      | Um Processo Driver de Enlace de Dados . . . . .                    | 149        |
| 5.2.1    | Modelo para Buffers de Dados . . . . .                             | 149        |
| 5.2.2    | A Interface CH2 com o Handler . . . . .                            | 152        |
| 5.2.3    | Inicialização do CH2 . . . . .                                     | 152        |
| 5.2.4    | Procedimentos de Designação e Teste de TEI . . . . .               | 153        |
| <b>6</b> | <b>Conclusões e Considerações Finais</b>                           | <b>154</b> |

|             |     |
|-------------|-----|
| 7 Anexo I   | 157 |
| 8 Anexo II  | 158 |
| 9 Anexo III | 159 |

## Lista de Figuras

|    |                                                                                           |     |
|----|-------------------------------------------------------------------------------------------|-----|
| 1  | Serviços de Suporte e Telesserviços . . . . .                                             | 7   |
| 2  | Configuração de Referência para Conexão do usuário à RDSI . . . . .                       | 9   |
| 3  | Implementações mais comuns no Acesso Básico . . . . .                                     | 11  |
| 4  | Implementações mais comuns no Acesso Primário . . . . .                                   | 12  |
| 5  | Exemplo da Arquitetura Funcional por Camadas de um Processo de Telecomunicações . . . . . | 16  |
| 6  | O novo Modelo de Referência por Camadas . . . . .                                         | 18  |
| 7  | Camadas tratadas em cada Grupo Funcional . . . . .                                        | 19  |
| 8  | Canal B Comutado por Circuito . . . . .                                                   | 20  |
| 9  | Canal D em Fase de Sinalização com a Central Local . . . . .                              | 21  |
| 10 | Canal B ou Canal D, Comutados por Pacotes, transportando Dados . . . . .                  | 22  |
| 11 | Símbolos da LEDES . . . . .                                                               | 27  |
| 12 | Estrutura Geral de Quadros LAPD . . . . .                                                 | 30  |
| 13 | Utilização dos Quadros LAPD . . . . .                                                     | 32  |
| 14 | Circuitos Integrados num Acesso Básico RDSI . . . . .                                     | 36  |
| 15 | Arquitetura Funcional . . . . .                                                           | 40  |
| 16 | Pinagem Sugerida . . . . .                                                                | 44  |
| 17 | Estrutura de Quadro da Interface IOM . . . . .                                            | 47  |
| 18 | Estrutura de Quadros IOM Multiplexada . . . . .                                           | 49  |
| 19 | Interface HDLC - Aquisição contínua de dados . . . . .                                    | 52  |
| 20 | Interface HDLC - Aquisição com <i>strobe</i> . . . . .                                    | 53  |
| 21 | Estrutura de Quadro da Interface SLD . . . . .                                            | 55  |
| 22 | Configuração de Timing de Terminal . . . . .                                              | 57  |
| 23 | Configuração de Timing de Central . . . . .                                               | 58  |
| 24 | Estrutura de Quadro da Interface SSI . . . . .                                            | 59  |
| 25 | Princípios de Chaveamento de Canais B . . . . .                                           | 60  |
| 26 | Loop na Interface SLD . . . . .                                                           | 61  |
| 27 | Chaveamento da Interface SLD . . . . .                                                    | 62  |
| 28 | Chaveamento da Interface SSI . . . . .                                                    | 63  |
| 29 | Loop da Porta B . . . . .                                                                 | 64  |
| 30 | Modos de Operação do HDLC . . . . .                                                       | 69  |
| 31 | Arquitetura da R-FIFO . . . . .                                                           | 87  |
| 32 | Algoritmo de Funcionamento da R-FIFO . . . . .                                            | 89  |
| 33 | Arquitetura da X-FIFO . . . . .                                                           | 91  |
| 34 | Projeto Hierárquico . . . . .                                                             | 120 |
| 35 | Ciclo de Realização de um Circuito Integrado . . . . .                                    | 122 |
| 36 | Particionamento do Circuito . . . . .                                                     | 129 |
| 37 | Idéia Geral do Bloco TLD . . . . .                                                        | 132 |
| 38 | Arquitetura do TLD . . . . .                                                              | 140 |
| 39 | Modelo de Software de Sistema Distribuído . . . . .                                       | 150 |
| 40 | Software de Sinalização via Canal D na RDSI . . . . .                                     | 151 |
| 41 | Modelo de Gerenciamento da Memória do Handler . . . . .                                   | 151 |

## Capítulo Um

### Introdução

# 1 Introdução

Este capítulo tem dois objetivos básicos. Primeiro, pretende trazer alguma informação do que é a Rede Digital de Serviços Integrados (RDSI), de forma bastante sumária, como subsídio e motivação para acompanhar o trabalho; isto é feito na seção 1.1 - *Considerações sobre a RDSI*. Segundo, trazer um resumo de como o trabalho está dividido nos capítulos seguintes; isto é feito na seção 1.2 - *Resumo dos capítulos seguintes*.

## 1.1 Considerações sobre a RDSI

Na última década as disciplinas de *processamento computacional e comunicações*, anteriormente separadas, se misturaram, resultando numa disciplina única - *processamento da informação*. O próximo passo é prover uma rede para processamento da informação que combina a inteligência oferecida pela disciplina de processamento com os recursos de transmissão da disciplina de comunicações. Tal rede atenderia as necessidades de todos os usuários, tratando todas as fontes de informação (por exemplo, voz, dados, imagens, gráficos e vídeo) como dados e utilizando as vantagens da tecnologia mais recente disponível. Esta nova rede de processamento de informação vem sendo conhecida como RDSI (Rede Digital de Serviços Integrados).

O corpo de recomendações internacionais para a RDSI define-a como uma rede, em geral evoluindo a partir da rede telefônica digital atual, que provê conexão digital de extremo-a-extremo, suportando uma larga faixa de serviços de voz e não-voz para os quais os usuários têm acesso através de um conjunto limitado de interfaces usuário-rede padronizadas. A RDSI terá a habilidade para usar um canal de comunicação único para todas as formas de transferência de informação, independente da fonte.[CBT83]

Para conseguir atingir esta meta, a RDSI requer uma tremenda cooperação e coordenação entre os fornecedores de serviços (operadoras do sistema de telecomunicações), de semicondutores, de equipamentos e usuários. O esforço necessário para levar à maturidade a RDSI está sendo conduzido sob os auspícios do CCITT (Comitê Consultivo Internacional para Telefonia e Telegrafia), cujo objetivo é estabelecer padrões para telefonia assim como para transmissão de dados. O CCITT tem 6 grupos de trabalho envolvidos com a RDSI:

- SG I - Serviços Telemáticos - Recomendações da Série F.
- SG VII - Comunicação de Dados em Redes Públicas de Dados - Recomendações da Série X.
- SG VIII - Terminais Telemáticos - Recomendações da Série T.
- SG XI - Comutação e Sinalização Digital - Recomendações da Série Q.

- SG XVII - Comunicação de Dados na Rede Telefônica - Recomendações da Série V.
- SG XVIII - Redes Digitais e RDSI - Recomendações da Série I.

Em termos de número de pessoas envolvidas, este esforço não tem precedentes.

### 1.1.1 Objetivos e princípios da RDSI

O conceito de RDSI combina dois objetivos: primeiro, a indústria telefônica esforçar-se-á em direção a um único serviço de transmissão capaz de integrar todos os serviços de comunicações em uma rede única, e segundo, a indústria adaptar-se-á à tendência irreversível das facilidades totalmente digitais. Estes objetivos levam aos seguintes princípios que norteiam o desenvolvimento em RDSI:

- o desenvolvimento será evolutivo e baseado na melhoria tecnológica da rede de transmissão existente;
- a RDSI irá suportar, na mesma rede, serviços de voz e não-voz combinados, que estarão disponíveis através de um único acesso de usuário;
- a RDSI será uma rede inteligente, funcionalmente dividida em níveis de acordo com os princípios do modelo de referência para sistemas abertos da ISO (veja seção 2.1);
- um conjunto limitado de padrões para as interfaces do acesso usuário-rede estarão disponíveis;
- a RDSI será uma rede comutada por circuitos onde os nós serão centrais com controle de programa armazenado (CPAs);
- portabilidade é um objetivo primário - os assinantes devem ser capazes de conectar e usar seus terminais em qualquer lugar do mundo.[JWC87]

À medida que a RDSI tornar-se operacional, vai-se passar a usufruir de benefícios como redução de custos, possibilidade de novos serviços, habilidade de misturar serviços diferentes na mesma linha de acesso, interface simplificada, monitoração e controle facilitados e vasto potencial para novas oportunidades de processamento de informações.

### 1.1.2 A Série I de Recomendações

A Tabela 1 mostra toda a estrutura da Série I de Recomendações do CCITT, que é uma série de recomendações específicas de RDSI (não são mostradas todas as Recomendações já existentes). Esta série trata de seis grandes temas, objetivando cobrir todos os aspectos de relevância para esta rede [CCT85].

| <b>Recomendações</b>             | <b>Assuntos Tratados</b>                  |
|----------------------------------|-------------------------------------------|
| I.100s                           | Conceito de RDSI (I.120)                  |
|                                  | Estrutura da Série I (I.110)              |
|                                  | Terminologia (I.112)                      |
|                                  | Método para Descrição de Serviço (I.130)  |
|                                  | Evolução da RDSI (I.140)                  |
| I.200s                           | Princípios de Serviços RDSI (I.210)       |
|                                  | Serviços de Suporte (I.230s)              |
|                                  | Telesserviços (I.240s)                    |
|                                  | Serviços Suplementares (I.250s)           |
| I.300s                           | Princípios de Capacitação de Rede (I.310) |
|                                  | Modelos de Referência (I.320s)            |
|                                  | Numeração e Roteamento na RDSI (I.330s)   |
|                                  | Tipos de Conexão (I.340s)                 |
|                                  | Objetivos de Desempenho (I.350s)          |
| I.400s                           | Tarifação                                 |
|                                  | Interfaces Usuário-Rede                   |
|                                  | Camada 1 do Acesso Básico (I.430)         |
|                                  | Camada 1 do Acesso Primário (I.431)       |
|                                  | Camada 2 - Princípios Gerais (I.440)      |
|                                  | Camada 2 - Especificação (I.441)          |
|                                  | Camada 3 - Princípios Gerais (I.450)      |
| Camada 3 - Especificação (I.451) |                                           |
| I.500s                           | Interfuncionamento de Redes               |
|                                  | Princípios de Interfuncionamento (I.510s) |
|                                  | Interfuncionamento RDSI-RDSI (I.520)      |
|                                  | Interfuncionamento RDSI-RTPC (I.530)      |
| I.600s                           | Princípios de Manutenção                  |

Tabela 1: Recomendações da Série I

Na Série I.100 encontram-se os conceitos gerais de RDSI, as definições e a terminologia, e são introduzidos os métodos de representação de serviços e rede.

Na Série I.200 são definidos os serviços de telecomunicações padrões para o usuário, incluindo a descrição dos serviços principais, de forma a criar uma RDSI de padronização internacional.

Na Série I.300 são cobertos os aspectos relativos à *rede* propriamente dita, isto é, sua arquitetura, os princípios de numeração, tarifação, encaminhamento de chamadas e as capacitações necessárias ao fornecimento dos serviços.

Na Série I.400 são descritos os detalhes das *interfaces de acesso usuário-rede*, incluindo os protocolos de comunicação. Nesta série, o nível de detalhe é bastante grande, pois desta forma é possível o projeto de circuitos integrados dedicados, importante para o barateamento da RDSI.

A Série I.500 traz as *interfaces internas* à RDSI, destinadas ao interfuncionamento de redes, sejam as existentes (telefônica, rede de comutação de pacotes de dados, etc) ou a própria RDSI.

A Série I.600 dedica-se integralmente aos aspectos de *manutenção* na RDSI.

### 1.1.3 Serviços de Assinante

A RDSI irá prover aos usuários acesso direto capaz de manipular muitas formas e tipos diferentes de informação. Estão incluídos:

- sinais dos telefones analógicos convencionais;
- voz digital a 64 Kbps (ou submúltiplos);
- dados por pacotes;
- dados a taxas correntes de 300 a 56K bps;
- informações digitais a taxas superiores a 64 Kbps (por exemplo a 2,048 Mbps);
- telemetria a taxas muito baixas;
- sinalização e controle de rede.

A unidade básica de serviço da RDSI é o canal ; ele representa uma porção específica da capacidade de transporte de informação da interface. Os canais são descritos e classificados por tipo, sendo definidos atualmente os seguintes:

- canal B - 64 Kbps
- canal D - 16 ou 64 Kbps
- canal H6 - 384 Kbps (igual a 6 canais B)
- canal H11 - 1536 Kbps (igual a 23 canais B)

- canal H12 - 1920 Kbps (igual a 30 canais B)

Tais canais são combinados em estruturas de interface que definem a capacidade máxima de informação em um dado ponto.

A maioria dos serviços é baseada em um ou mais canais B. O canal B, que é o canal mais importante numa RDSI, tem taxa de 64 Kbps e não carrega informação de sinalização de rede. Toda sua capacidade fica disponível para informação de usuário, de qualquer tipo, desde que possa ser digitalizada e suportada por uma taxa de 64 Kbps. O canal B pode ser usado tanto para acessar serviços transparentes comutados por circuito quanto os baseados em telefonia, assim como serviços comutados por pacotes.

O canal D, outra parte importante do repertório de canais da RDSI, tem como função prioritária o transporte de informação de sinalização de rede. Ele opera a 16 Kbps ou 64 Kbps dependendo da estrutura da interface e usa protocolos definidos pela RDSI (ao contrário dos canais B, que usam protocolos definidos pelo usuário, dependendo do serviço que carrega). O canal D pode ser usado também para transportar dados de usuário comutados por pacotes ou informações de baixa velocidade tais como aplicações em medidas meteorológicas. Se nenhuma sinalização existe para ser transportada, toda a capacidade do canal D pode ser usada para dados do usuário.

A separação do canal de sinalização em um canal exclusivo (canal D) é uma característica de grande importância na RDSI. Esta sinalização *fora da faixa* não perturba os canais de transporte de informação de usuário e pode ser solicitada durante o andamento de uma comunicação, seja para sinalizar ao usuário de destino ou à própria rede. Possibilita, ainda, a implementação simples de configuração ponto a multiponto.

Os canais H são usados para carregar informações de usuário a taxas mais altas que aquelas conseguidas por um canal B, tais como fac-símile rápido, áudio de alta qualidade, vídeo para teleconferências, etc. Na Experiência Piloto Brasileira em RDSI não serão utilizados esses canais e por isso não nos estenderemos a respeito de sua utilização.

De acordo com a Recomendação I.200, uma RDSI provê ao usuário dois tipos de serviços de telecomunicações - os serviços de suporte e os telesserviços. Em termos do Modelo de Referência OSI da ISO (veja Capítulo Dois), os serviços de suporte cobrem basicamente as camadas 1 a 3 e deixam o usuário se preocupar com os níveis 4 a 7; os telesserviços cobrem todos os 7 níveis. A figura 1 ilustra essas diferenças.

A Recomendação I.211 define os serviços de suporte mais importantes. Estão incluídos:

- Serviço em Modo Circuito a 64 Kbit/s irrestrito;
- Serviço em Modo Circuito para transferência de informação de usuário a 3.1 KHz (voz ou dados na faixa de voz);

- Serviço para Conversação (modo circuito para transferência de voz);
- Serviço em Modo Pacote, chamada virtual, nos canais B e D;
- Serviço em Modo Pacote, datagrama, no canal D.

Os telesserviços transformam a RDSI numa rede de transporte, processamento e carregamento de informações. A Recomendação I.210 fala de telefonia, teletex, fac-símile, videotexto, modo misto e MHS. Os telesserviços cobrem tanto comunicações terminal a terminal, quanto aplicações terminal-hospedeiro (Host).

A Recomendação I.210 também define os *Serviços Suplementares*. Estes serviços não são oferecidos por si mesmo, mas como melhoramentos aos serviços de suporte e telesserviços. Exemplos incluem chamada em espera, suspensão de chamada, transferência de chamada, apresentação do número de origem, grupo fechado de usuários, etc.

#### 1.1.4 Acesso do usuário à rede

A combinação de canais B e D define a *estrutura* das interfaces usuário-rede. O CCITT padronizou o acesso RDSI através de dois tipos de interface: acesso básico e acesso primário. Em ambos os casos tem-se um único canal D e um certo número de canais B. A taxa de transmissão no canal B é de 64 Kbps tanto para o acesso básico quanto para o primário, enquanto que no canal D a taxa de transmissão para o acesso básico é de 16 Kbps e para o acesso primário é de 64 Kbps.

O acesso básico, que é o mínimo que o usuário poderá contratar junto à operadora do sistema, é utilizado para transmissões de baixo tráfego com um número reduzido de terminais. Ele é composto do arranjo 2B+D, ou seja, dois canais B e um canal D. Assumindo que o canal D não está carregando sinalização de rede, o usuário tem à sua disposição 144 Kbps para transporte de dados.

O acesso primário é utilizado para alto tráfego de informações, tais como em PABX, tendo, portanto, uma capacidade muito maior que aquela oferecida pelo acesso básico. No Brasil adotamos a estrutura composta de 30 canais B mais um canal D, totalizando uma taxa de dados igual a 2,048 Mbps.[STB]

#### 1.1.5 Configuração de referência CCITT

A configuração de referência para a conexão do usuário é baseada em duas recomendações do CCITT:

- a recomendação I.411 descreve as interfaces do lado do usuário;
- a recomendação Q.512 descreve as interfaces do lado da central.

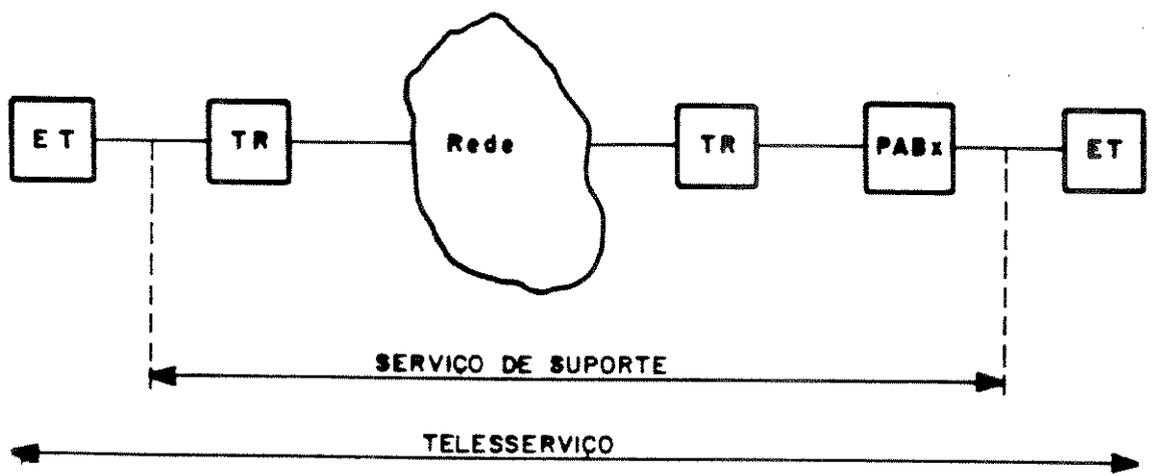


Figura 1: Serviços de Suporte e Telesserviços

A figura 2 mostra tal configuração de uma forma simplificada que assume conexão direta a uma central pública, ignorando qualquer mux/demux ou repetidor intermediários.

A configuração de referência, um elemento chave da interface de usuário RDSI, inclui dois conceitos: grupo funcional e ponto de referência. Grupos funcionais são conjuntos de funções que podem ser necessárias numa interface. Na figura 2 os grupos funcionais são representados pelos retângulos. Pontos de referência são usados para separar os grupos funcionais e podem corresponder a interfaces físicas reais. A presença ou não de um grupo funcional ou ponto de referência depende do tipo de serviço oferecido pela rede ao usuário ou da natureza do equipamento do usuário. A seguir são apresentados, de forma resumida, as funções de cada grupo e algumas informações básicas a respeito dos pontos de referência.[SOT87]

- *Terminação de Rede 1 (TR1)* : este grupo inclui a terminação física e elétrica da rede nas instalações do usuário. Monitoração de desempenho, transferência de potência, facilidades de manutenção e adequação de interfaces diferentes são funções deste grupo. Tais funções pertencem ao nível físico do modelo OSI (veja capítulo 2).
- *Terminação de Rede 2 (TR2)* : este grupo constitui funções mais avançadas da RDSI, relacionadas aos níveis de enlace e de rede do modelo OSI. Tratamento de protocolos, comutação, concentração e multiplexação são funções típicas executadas pelos equipamentos TR2, que incluem PABXs, redes locais, controladores de *clusters* e multiplexadores.
- *Equipamento Terminal 1 (ET1)* : este equipamento terminal é perfeitamente compatível com os requisitos RDSI. Um ET1 pode se conectar diretamente a um TR1 e pode ser um telefone digital, terminal de dados ou qualquer outro equipamento que sirva para tais aplicações.
- *Equipamento Terminal 2 (ET2)* : este grupo engloba os terminais que não cumprem os requisitos de interface RDSI, mas são compatíveis com interfaces existentes na atualidade, tais como RS232C, V24/V28 e X.21. Para se conectar à RDSI, um ET2 precisa de um adaptador de terminais.
- *Adaptador de Terminais (AT)* : este equipamento provê a interface e conversão de protocolos necessárias para um ET2 se interfacear com uma RDSI.
- *Terminal de Linha (TL)* : este grupo faz o acoplamento entre a linha de transmissão e o terminal da central.
- *Terminal de Central* : este grupo faz o tratamento da sinalização, mantém e supervisiona a conexão do lado da central (níveis 2 e 3 da OSI). Inclui funções como tratamento de protocolo, multiplexação e gerência, operação e manutenção da rede.

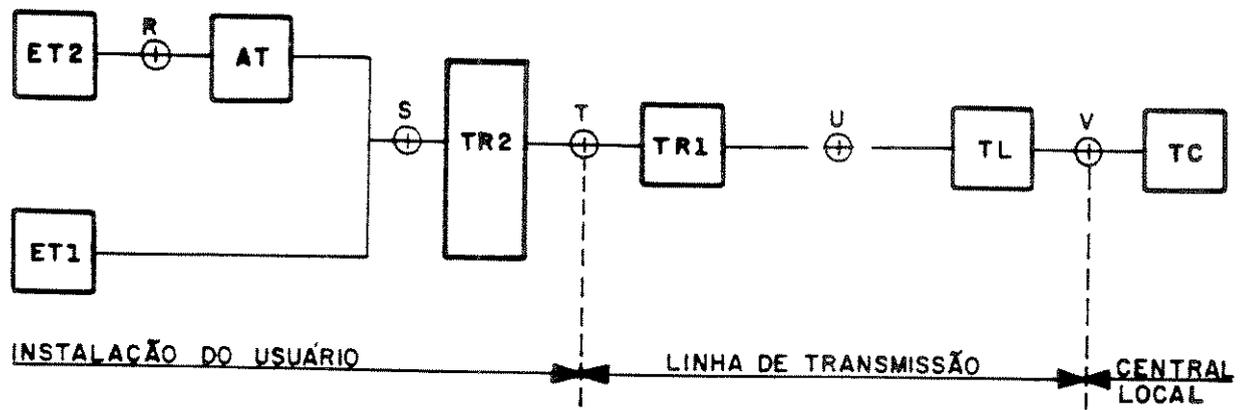


Figura 2: Configuração de Referência para Conexão do usuário à RDSI

Um ponto de referência é um ponto teórico separando dois grupos funcionais e pode ou não corresponder a uma interface física. A interface usuário-rede na RDSI é uma interface física correspondente aos pontos de referência S e T. O ponto de referência R representa uma interface terminal diferente da interface RDSI (por exemplo, uma interface X.25, um par a/b, etc). A linha de transmissão é representada pelo ponto de referência U e a interface de entrada numa central pública pelo ponto de referência V.[SOT87]

Na prática os usuários irão realizar a configuração de referência em uma quantidade limitada de modos diferentes. A figura 3 apresenta algumas implementações mais comuns para o acesso básico. Um usuário pode implementar uma configuração ponto a ponto para conectar um terminal integrado voz/dados ou ainda dois terminais de dados a uma interface de acesso básico RDSI. Ele não precisa da função TR2 e os pontos S e T são coincidentes. Alternativamente, um usuário com até 8 terminais pode implementar uma configuração de barramento S passivo. Um canal B é alocado a um terminal específico por um período de tempo, de acordo com a sinalização trocada com a rede via canal D; o canal D é compartilhado por todos os equipamentos. A Recomendação I.430 prevê um procedimento para resolver a contenção quando vários terminais querem transmitir usando os canais B ao mesmo tempo.

A figura 4 ilustra algumas das configurações possíveis num acesso primário. Um PABX prevê a função TR2 e um equipamento separado prevê a função TR1 (ou poderiam ser um único equipamento).

## 1.2 Resumo dos Capítulos Seguintes

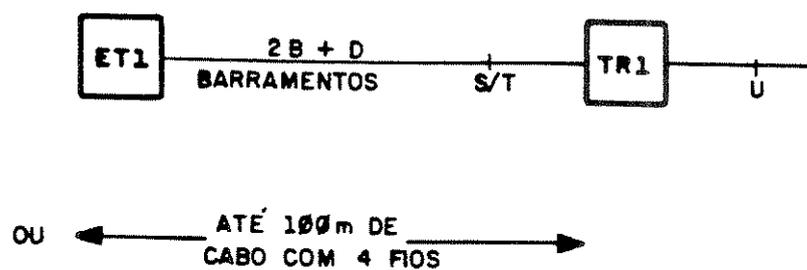
O *Capítulo Dois* apresenta um conjunto de conceitos que visam buscar um melhor entendimento deste trabalho. Inicialmente foi feita uma breve revisão do modelo de protocolos da ISO e a adequação do Modelo de Referência do CCITT para a RDSI apresentado no capítulo presente ao Modelo de Referência de Sistemas Abertos, o Modelo OSI. Ainda neste capítulo são analisados, de forma sucinta, os protocolos de sinalização na RDSI e, de particular interesse para o restante do trabalho, o protocolo de enlace no canal D. Finalmente discutimos nossa visão a respeito do papel da microeletrônica na RDSI.

O *Capítulo Três* é o núcleo deste trabalho. É nele que apresentamos uma sugestão de especificação de um circuito integrado que, prioritariamente, executa, de forma autônoma, as funções da camada de enlace na sinalização da RDSI via canal D. Procuramos ser minuciosos nesta especificação, de forma a que aqueles que irão implementá-la não venham a ter dúvidas nos aspectos funcionais.

Ainda visando ajudar a implementação no que se refere aos aspectos estruturais, são apresentadas e discutidas algumas arquiteturas e circuitos que poderão servir de base aos implementadores. Isso é feito no *Capítulo Quatro*.

O *Capítulo Cinco* é um complemento ao Capítulo Três. Ele parte do princípio

a) Ponto a Ponto



b) Multiponto

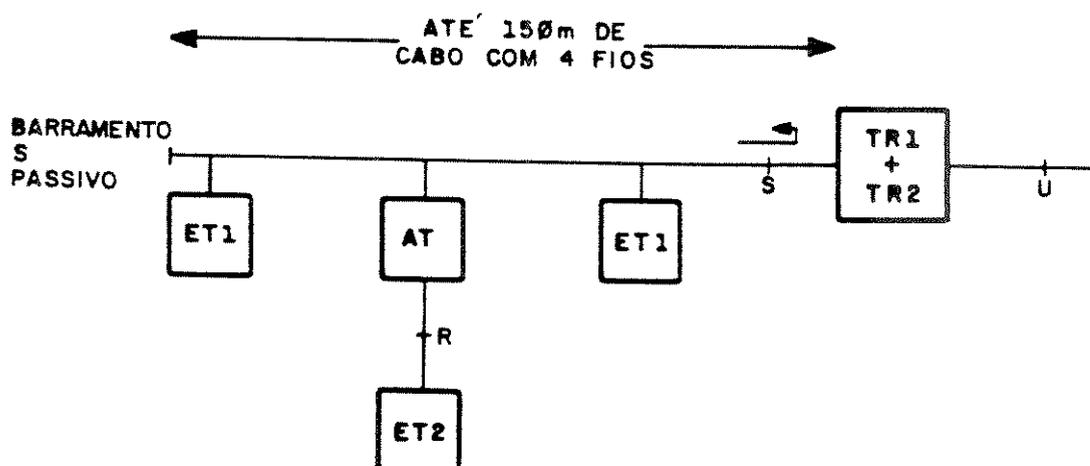


Figura 3: Implementações mais comuns no Acesso Básico

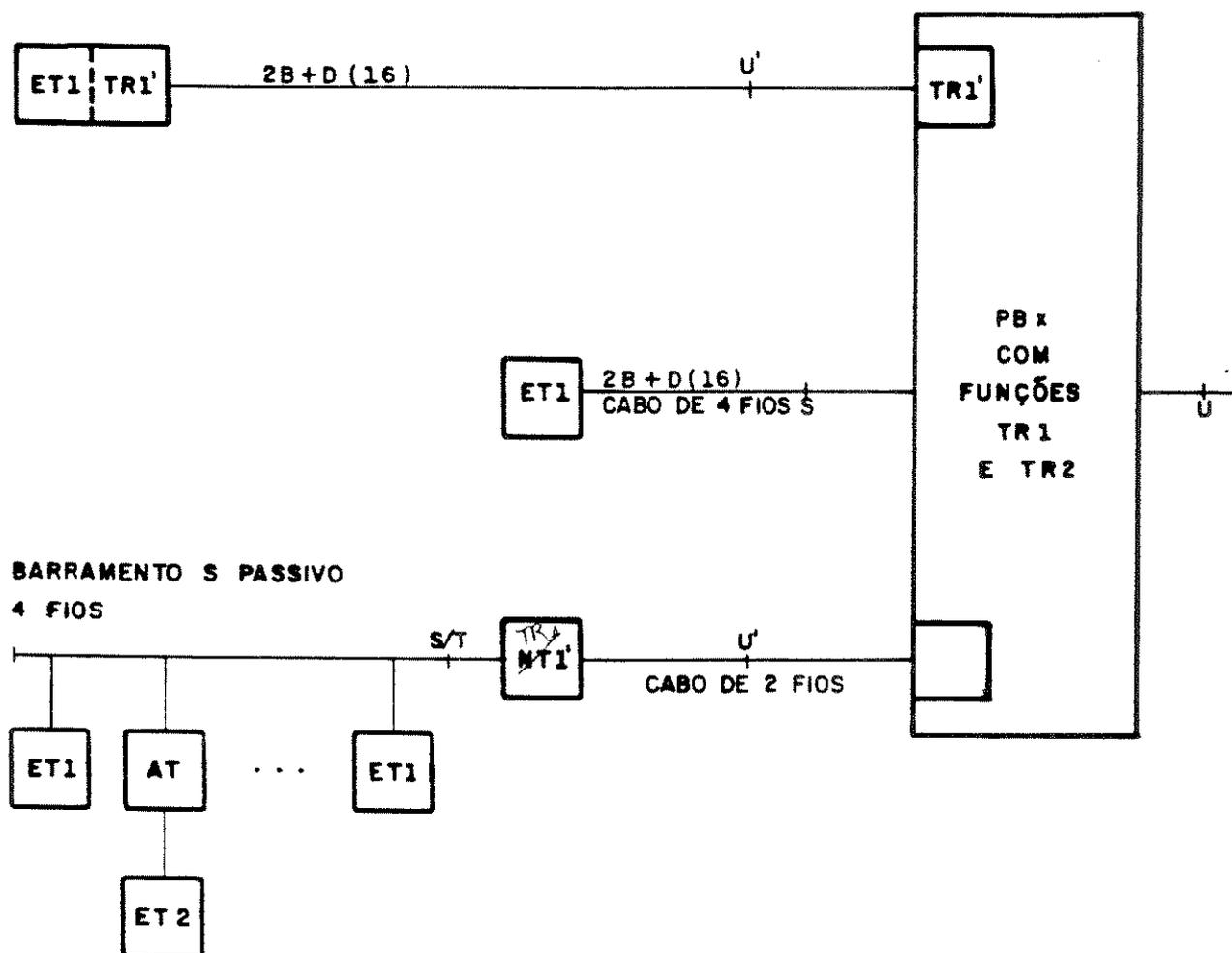


Figura 4: Implementações mais comuns no Acesso Primário

de existência do CI e analisa alguns aspectos de software relacionados ao mesmo.

O *Capítulo Seis* trata de apresentar algumas conclusões a respeito do trabalho.

Finalmente, o *Anezo I* apresenta o protocolo LAPD descrito na forma de *LEDS*, o *Anezo II* apresenta o *LEDS* da implementação do LAPD no CH2 e o *Anezo III* é uma lista de Referências Bibliográficas.

## Capítulo Dois

### Noções de Interesse para o Trabalho

## 2 Noções de Interesse para o Trabalho

Este capítulo procura apresentar alguns conceitos de importância para o entendimento do trabalho e a justificativa da linha adotada. Para tanto, é subdividido nas seguintes seções:

- 2.1. O modelo ISO e o modelo RDSI-CCITT
- 2.2. Protocolos de Sinalização Via Canal D
- 2.3. Representação de Protocolos
- 2.4. O protocolo de enlace no canal D
- 2.5. As interfaces padrões para interconexão de CIs
- 2.6. Microeletrônica e RDSI

### 2.1 O modelo ISO e o modelo RDSI-CCITT

A interconexão de equipamentos em rede não é um conceito novo, mas até recentemente as redes tinham uma natureza proprietária, isto é, cada fornecedor com sua solução, que era incompatível com a solução de outro. No passado, um usuário final tinha que comprar todos os seus equipamentos de um mesmo vendedor para poder interconectá-los. Este problema foi resolvido com o desenvolvimento do conceito de *sistema aberto*. Um sistema aberto é construído usando padrões largamente aceitos. Eles permitem ao usuário final adquirir equipamentos de vários vendedores de maneira a obter uma solução ótima para sua aplicação. Note-se que quando um padrão torna-se largamente aceito, também torna-se factível sua implementação em VLSI, e com isso diminui o número de conexões ( e o preço ! ) nas placas dos equipamentos.

#### 2.1.1 O Modelo de Referência para Sistemas Abertos da ISO

A *International Standard Organization* (ISO), num esforço para encorajar o surgimento de redes *abertas* desenvolveu o modelo de referência *Open System Interconnection* (OSI). Em linguagem simples, o modelo agrupa logicamente as funções e conjuntos de regras, chamadas protocolos, necessários para estabelecer e conduzir comunicação entre duas ou mais partes. O modelo consiste de 7 funções, normalmente referenciadas como camadas ou níveis, descrevendo os objetivos de cada nível em termos gerais, não em termos de uma implementação específica.

Esta solução de modelo em camadas traz consigo duas vantagens. Primeira, níveis permitem uma divisão clara da tarefa de projeto via modularidade, tornando as especificações bem limpas. Segunda, os sistemas baseados em uma arquitetura dividida em níveis são bastante flexíveis. A flexibilidade é conseguida porque cada nível funciona independentemente do nível superior ou inferior a ele. Assim, implementações específicas de um dado nível podem ser mudadas facilmente, por exemplo, para alcançar uma melhoria no desempenho.

As funções de cada nível do modelo OSI são resumidas a seguir.

*Nível Físico* : descreve o meio físico sobre o qual a cadeia de bits será transmitida. Este nível especifica o tipo de cabo, conectores, níveis dos sinais, taxa de bits, técnica de codificação de dados, métodos de modulação e métodos de detecção de colisão em redes com contenção.

*Nível de Enlace* : descreve as regras para transmissão no canal; este nível especifica itens como o formato da informação (quadro) e os procedimentos para ganhar controle do canal (método de acesso), transmissão do quadro e desconexão.

*Nível de rede* : oferece os comandos necessários ao estabelecimento e liberação de chamadas através dos diferentes nós de comutação da rede.

Estes três níveis são chamados coletivamente de camadas de suporte ou de sub-rede.

*Nível de Transporte* : assegura a integridade dos dados numa base fim-a-fim e fornece a necessária qualidade do serviço na troca de informações.

*Nível de Sessão* : estabelece e termina conexões lógicas entre entidades da rede; é responsável ainda pelo mapeamento de nomes lógicos em endereços de rede.

*Nível de Apresentação* : provê qualquer conversão necessária nos formatos ou códigos, de maneira a colocar a informação numa forma reconhecível.

*Nível de Aplicação* : fornece seus serviços diretamente ao usuário final e atua como uma *janela* por onde passam as informações dos processos que entram e saem do ambiente OSI.

O gerenciamento da rede é responsável pela planificação das operações, o que inclui o tratamento de estatísticas operacionais tais como erros e tráfego. É responsável também pela inicialização da rede e manutenção. O gerenciador da rede faz interface com cada um dos 7 níveis.

Na figura 5 é mostrado que o nível de transporte e aqueles situados acima dele executam funções fim-a-fim (constituindo, juntos, a chamada *camada de aplicação ou de usuário*) e que as funções executadas do nível de rede para baixo podem ser executadas também pelos nós intermediários da rede. [TAR86]

### 2.1.2 Adequação dos modelos OSI e CCITT-RDSI

O modelo de referência para sistemas abertos foi desenvolvido visando apenas a construção de redes de computadores, para comunicação de dados. Desta forma, foi necessário adaptá-lo para que pudesse ser aplicado também numa rede que

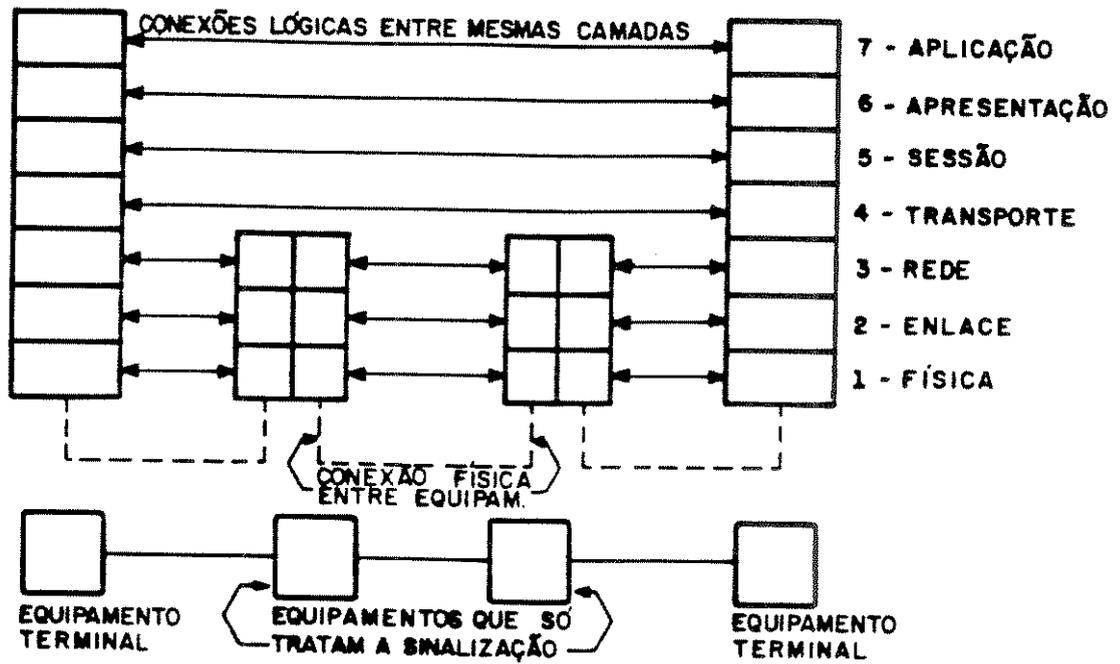


Figura 5: Exemplo da Arquitetura Funcional por Camadas de um Processo de Telecomunicações

inclui, além de comunicação de dados, comunicação de voz, textos e imagens e que possui o modelo apresentado no capítulo 1, isto é, a RDSI.

Na RDSI, o canal D deve utilizar um procedimento de acesso ao enlace orientado para transferência de pacotes, isto é, um acesso estatístico, ao passo que o canal B deve utilizar um procedimento de acesso fixo ao enlace, multiplexado no tempo.

A interação do usuário com a rede ocorre através dos canais B e/ou D, por onde são transportadas as informações de voz e/ou dados do usuário. A sinalização é feita sempre pelo canal D; este canal poderá, eventualmente, carregar dados por pacotes, dados de telemetria, alarme, etc.

A estrutura de protocolo que permite o transporte de informações usuário-rede, em ambos os canais, é aberta e dividida por camadas. As camadas, por sua vez, são divididas em dois grupos: camadas de suporte e camadas de aplicação. As camadas de suporte são divididas em 3 níveis e servem basicamente para sinalização entre o usuário e a rede, proporcionando os meios físicos e lógicos para que as conexões sejam estabelecidas, mantidas e tarifadas. As camadas de aplicação são divididas em 4 níveis e servem basicamente às funções de comunicação extremo a extremo entre usuários ou entre usuário e recursos da rede.

O que se conclui dessas observações, quando se olha o Modelo OSI, é que na RDSI não existe uma única estrutura de protocolo de 7 camadas implementadas, mas 3 estruturas simultâneas, uma voltada ao usuário, uma voltada à sinalização e uma terceira voltada à gerência da rede. Veja Figura 6.

A figura 7 mostra quais são os níveis tratados em cada grupo funcional definido no acesso do usuário.

A figura 5 é um exemplo da arquitetura funcional por camadas de um processo de telecomunicação. Nem todas as camadas devem estar presentes em determinadas fases da comunicação e em determinados canais. Desta forma, a figura 8 apresenta a divisão por camadas de um canal B comutado por circuitos, a figura 9 mostra o canal D em fase de sinalização com a central local e a figura 10 mostra os canais B ou D, comutados por pacotes, transportando dados.

Na figura 5 são representadas as camadas do protocolo para uma utilização geral do canal B, quando ET1 é um equipamento terminal de dados. No caso de ET1 ser um telefone digital, a camada 2 desaparece e as camadas 3 e superiores se resumem à conversão A/D e D/A do sinal de voz.

A representação do serviço telefônico atual através da estrutura de protocolo descrita anteriormente não é direta, já que várias camadas inexistem neste serviço, em função de suas definições. Assim, uma conexão telefônica por comutação de circuito, já estabelecida através da rede, implica numa alocação contínua e exclusiva do canal respectivo (B). Desta forma se tornam necessárias a camada 1, que garante a conexão física, a camada 3, que garante a supervisão e tarifação, e a camada 6, que garante a conversão A/D e D/A da voz. As demais não se aplicam.[CBT83]

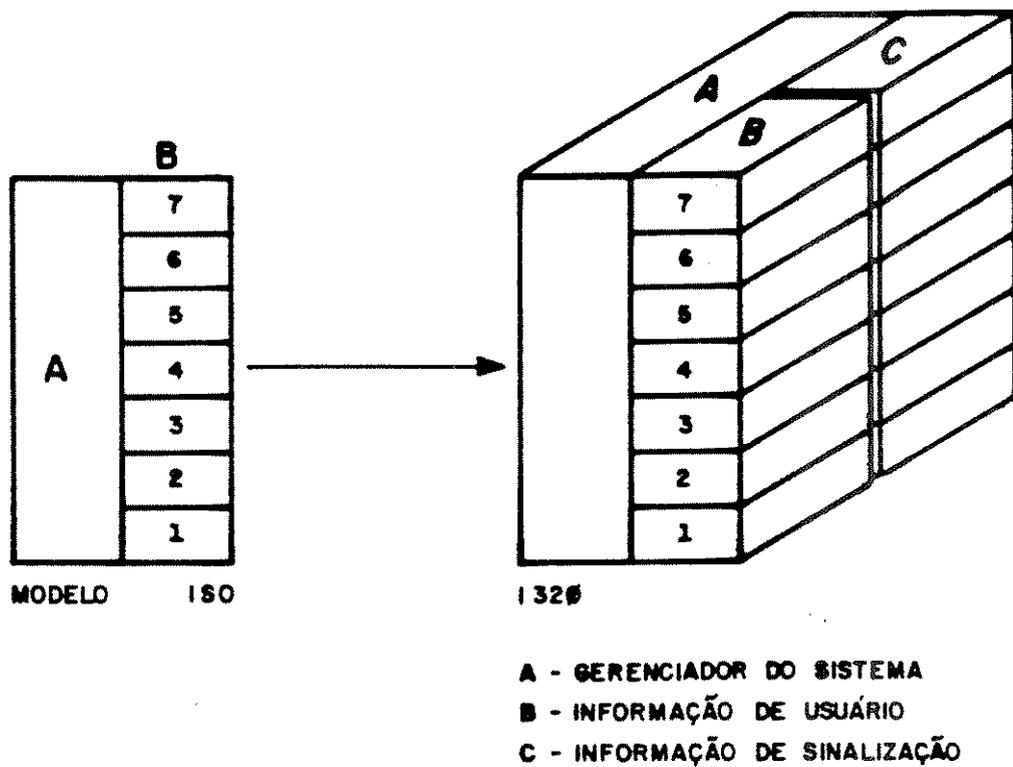


Figura 6: O novo Modelo de Referência por Camadas

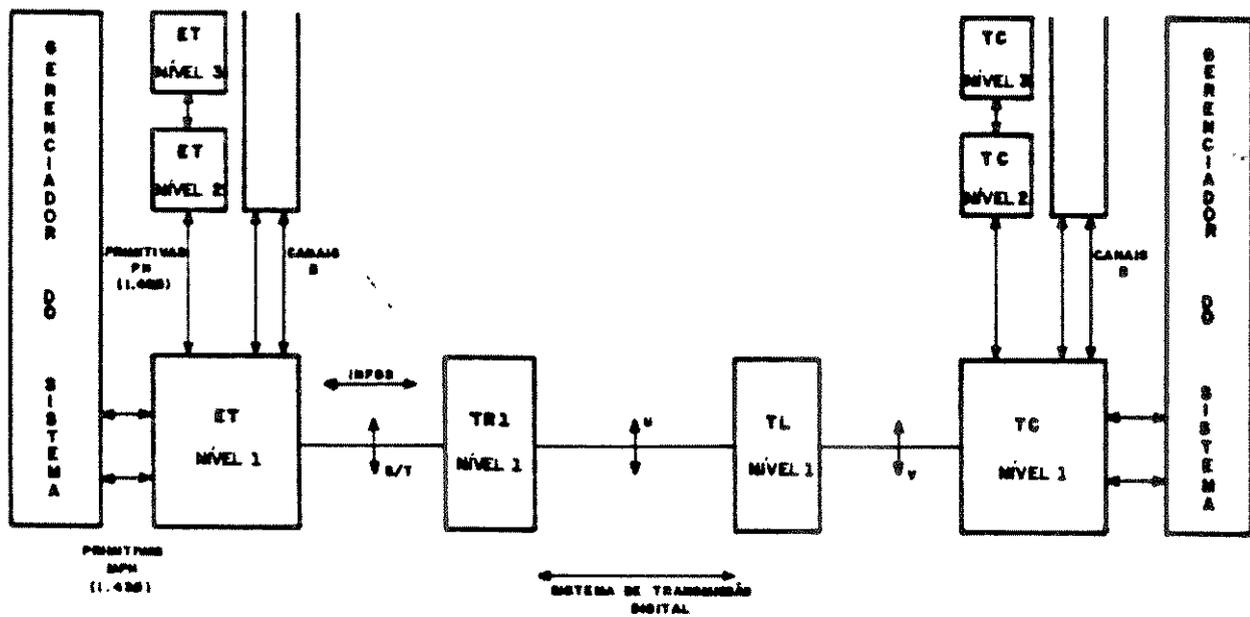


Figura 7: Camadas tratadas em cada Grupo Funcional

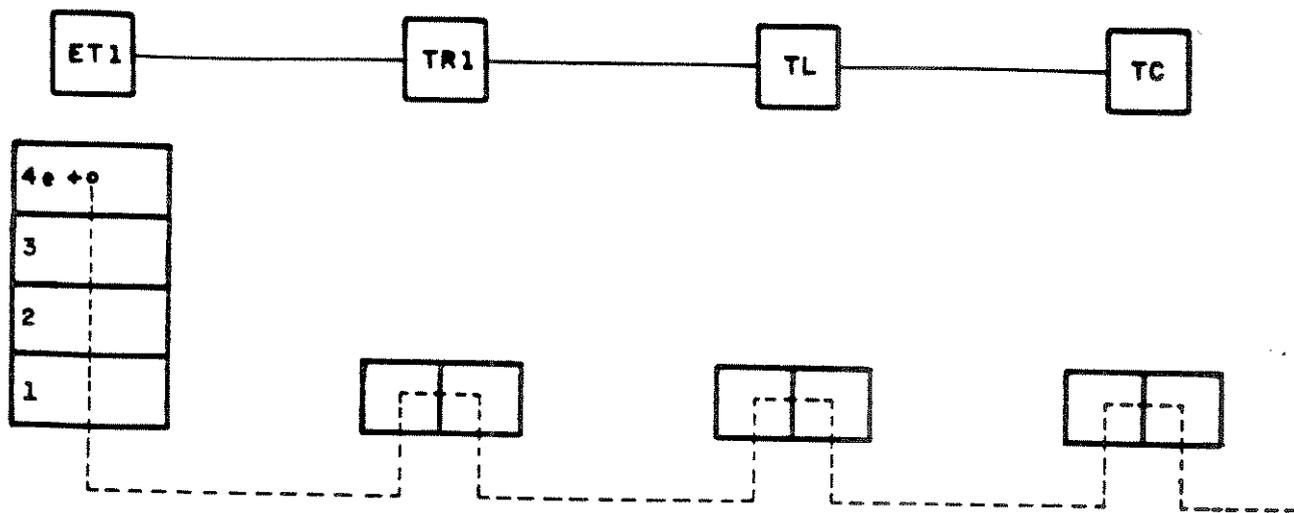


Figura 8: Canal B Comutado por Circuito

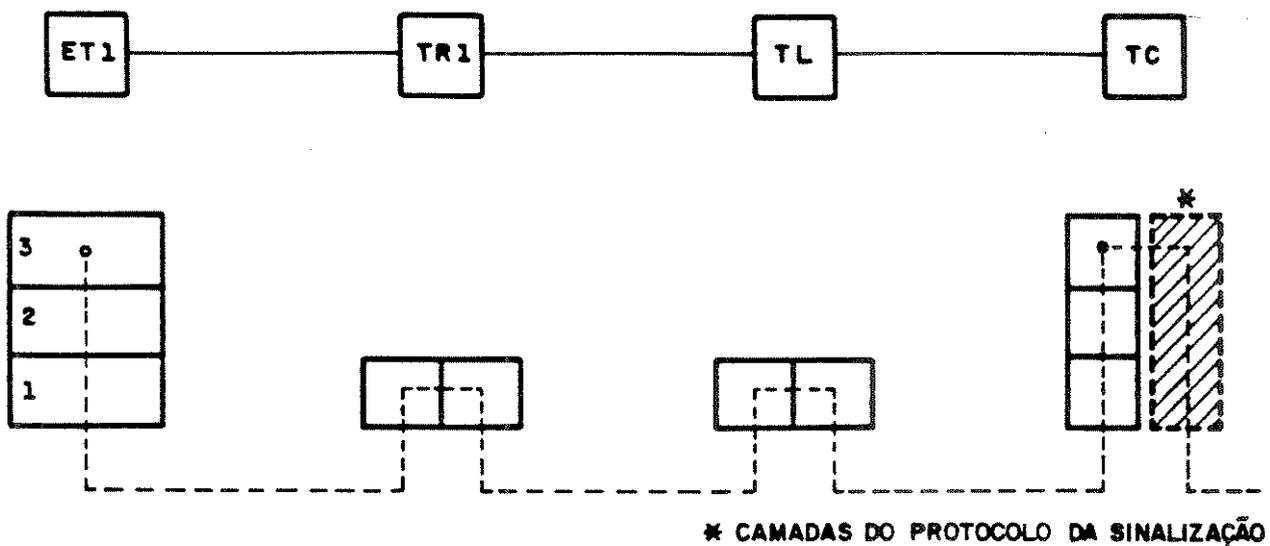


Figura 9: Canal D em Fase de Sinalização com a Central Local

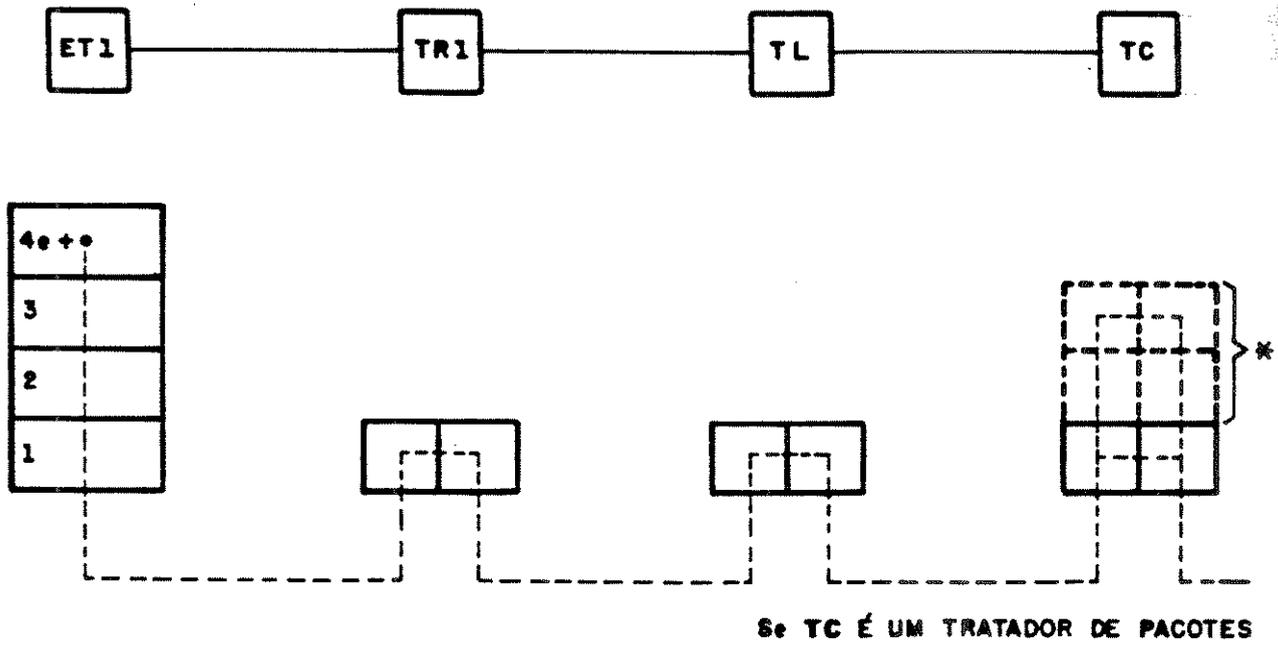


Figura 10: Canal B ou Canal D, Comutados por Pacotes, transportando Dados

No caso da figura 8, o protocolo do canal D é tratado em suas 3 camadas tanto pelo ET1 como pelo TC, pois está associado à fase de sinalização entre o terminal e a central local. Se a sinalização deve continuar até a próxima central, o tratamento envolve as camadas respectivas da sinalização número 7, representadas na região sombreada da figura. Não necessariamente o tratamento do protocolo da sinalização número 7 encontra-se no TC.

## 2.2 Protocolos de Sinalização Via Canal D

O canal D de uma linha de acesso usuário-rede carrega a sinalização para os canais B na mesma linha. Entre os nós da rede, os enlaces de sinalização são estruturados de acordo com as regras definidas pelo CCITT no seu Sistema de Sinalização por Canal Comum, também chamado Sinalização Número 7.[STB]

A Recomendação I.320 identifica os fluxos de informação dentro de uma RDSI e contém um modelo de referência indicando os caminhos percorridos pela informação de sinalização e pela informação de usuário-a-usuário. Os fluxos principais da informação de sinalização dizem respeito ao controle das chamadas e dos serviços suplementares, enquanto os fluxos de informação de usuário-a-usuário dependem da aplicação particular. A rede também carrega fluxos de informação para sinalização do sistema de gerência e para operação e manutenção dos equipamentos.

A sinalização usuário-rede precisa suportar o estabelecimento de todos os tipos de conexão requeridos para transferência de informação entre usuários. Ele precisa suportar também os serviços suplementares, incluindo sinalização usuário-a-usuário como um meio adicional de transferência de informação entre usuários. O protocolo de canal D é, então, estruturado em níveis:

- O nível físico (camada 1) da interface usuário-rede é definido nas interfaces S, T e U nos acessos básico e primário.
- O nível de enlace (camada 2) provê transmissão de quadros livres de erros entre duas entidades.
- O nível de rede (camada 3) assume o controle e supervisão das chamadas.

### 2.2.1 Especificação de Camada 1

A Recomendação I.430 especifica o nível 1 do acesso básico na interface usuário-rede. Esta Recomendação detalha:

- as características elétricas e físicas das ligações entre os componentes do Modelo de Referência;
- as configurações de fiação permitidas aos terminais;
- como a taxa de 192 Kbit/s é usada para prover a capacitação 2B+D;

- como os ETs podem ter acesso ao canal D comum, de forma organizada;
- as primitivas que o nível 1 passa ao nível 2 e vice-versa;
- os procedimentos de ativação e desativação da interface;
- etc

A Recomendação I.431 especifica o nível 1 do acesso primário, que é baseado num acesso PCM.

### 2.2.2 Especificação de Camada 2

Veja seção 2.4 neste capítulo.

### 2.2.3 Especificação de Camada 3

A Recomendação I.450 traz os *Princípios Gerais* e a Recomendação I.451 traz as *Especificações Detalhadas* dos procedimentos de sinalização para estabelecimento, modificação e término de chamadas. As funções da camada 3 para controle de chamada nos modos circuito, pacote e sem conexão, envolvem: [LFB88]

- tratamento das mensagens de camada 3;
- gerenciamento dos recursos alocados a uma chamada;
- gerenciamento da temporização;
- detecção de falhas.

O diálogo entre o usuário e a rede com propósitos de controle de chamada é feito através de mensagem de tamanho variável. Para uma chamada no modo circuito ou no modo pacote em um canal B, estas mensagens são trocadas antes que a conexão de canal B seja estabelecida. Para uma chamada modo pacote no canal D ou para uma chamada sem conexão, o diálogo pode ser concomitante com o início da transferência de informação.

Vários tipos de mensagens são especificadas, cada uma determinando uma ação no lado do usuário ou no lado da rede. Cada mensagem contém elementos de informação, usualmente de tamanho variável, que define o tipo de pedido, os recursos a serem reservados ou o resultado de uma operação de processamento.

Uma chamada pode estar em uma de três fases distintas: estabelecimento, fase ativa ou liberação. Alguns tipos de mensagens são específicas de uma dada fase, enquanto outras podem ocorrer em qualquer uma delas. A seguir são mostradas algumas mensagens possíveis:

- Fase de Estabelecimento da Chamada:
  - Estabelecimento

- Confirmação de Estabelecimento
- Alerta : usuário chamado está sendo alertado
- Fase Ativa:
  - Suspensão : suspensão temporária de uma chamada
  - Religação : retomada da chamada suspensa
  - Notificação : aviso de suspensão e religação do usuário remoto
- Fase de Liberação da Chamada
  - Desconexão
  - Liberação Completa
- Todas as Fases:
  - Informação : envio de número de destino, pedido de serviços suplementares, etc
  - Progresso : informa estado da chamada nos casos de interfuncionamento
  - Reiniciação : para recuperação de erros

## 2.3 Representação de Protocolos

Uma representação textual de protocolos, ainda que necessária e mais interessante ao usuário, pode levar a ambiguidades de interpretação dos procedimentos do mesmo. Com o objetivo de evitar tais ambiguidades, o CCITT definiu uma linguagem que provê uma especificação e descrição do comportamento dos sistemas de telecomunicações, chamada *SDL Specification and Description Language*, que no Brasil é conhecida como *LEDS Linguagem de Especificação e Descrição de Sistemas*. As especificações e descrições usando LEDS pretendem ser formais, no sentido de que é possível analisá-las e interpretá-las de forma não ambígua.

Uma especificação de sistema consiste de uma especificação do comportamento funcional e um conjunto de parâmetros gerais do sistema. A linguagem procura descrever somente os aspectos comportamentais de um sistema; os parâmetros gerais se referem a certas propriedades, tais como capacidade do sistema, que têm que ser descritas usando técnicas diferentes.

O comportamento de um sistema é descrito num padrão estímulo-resposta, assumindo que tanto um quanto outro são entidades discretas. Assim, o comportamento do sistema descrito em LEDS é a sequência de respostas a uma dada sequência de estímulos, visto do lado de fora do sistema.

A Linguagem de Especificação e Descrição de Sistemas provê, ainda, conceitos de estruturação que permitem que um sistema seja particionado, de forma que ele pode ser definido, desenvolvido e compreendido, uma parte de cada vez.

Os símbolos mais utilizados nesta linguagem, e que aparecerão no restante deste trabalho, estão mostrados na figura 11 e explicados a seguir:

- Estado: representa o estado em que se encontra o protocolo ou o estado para o qual o protocolo está indo a partir daquele momento (próximo estado)
- Tarefa: representa uma ação do protocolo naquele momento;
- Decisão: contém um parâmetro a partir do qual, em função de seu valor, um caminho diferente será seguido;
- Recepção de Sinal Interno <sup>1</sup>: indica a chegada de mensagens/primitivas geradas dentro de um mesmo processo. Na aplicação deste trabalho, indica as mensagens recebidas do processo LAPD da estação remota;
- Recepção de Sinal Externo: indica a chegada de mensagens /primitivas geradas por um processo externo. Na aplicação deste trabalho, indica as primitivas recebidas dos processos de camadas 1 e 3;
- Geração de Sinal Interno: indica uma mensagem/primitiva enviada para o mesmo processo. Na aplicação deste trabalho, indica as mensagens enviadas ao processo LAPD da estação remota;
- Geração de Sinal Externo: indica uma mensagem/primitiva enviada para um processo externo. Na aplicação deste trabalho, indica as primitivas enviadas aos processos de camadas 1 e 3;
- Descrição de uma Macro: indica que a descrição a seguir corresponde a uma macro cujo nome de referência encontra-se no símbolo ou que naquele ponto é feita uma chamada de macro.

Além disso, em nossa aplicação (anexo II), utilizamos também o símbolo *Geração de Sinal Externo* para indicar *interrupções* ao processador.

## 2.4 O Protocolo de Enlace no Canal D

### 2.4.1 A camada de enlace do modelo OSI/ISO

A tarefa do nível de enlace de dados é transformar uma facilidade de transmissão com possibilidade de erros em uma linha livre de tais erros e entregá-la para o nível 3. Isto é feito através da quebra da cadeia de dados de entrada em quadros, da transmissão dos quadros sequencialmente e do processamento dos quadros reconhecidos enviados de volta ao receptor. Uma vez que o nível físico meramente aceita e transmite uma cadeia de bits sem qualquer atenção a seu significado ou estrutura, é o nível de enlace quem deve criar e reconhecer as

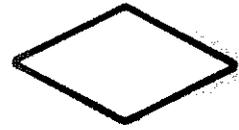
<sup>1</sup>A distinção entre sinal interno e externo é nossa; não faz parte da SDL.



**ESTADO**



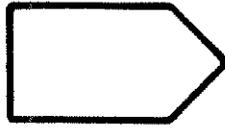
**TAREFA**



**DECISÃO**



**REC. SIN. INT.**



**GER. SIN. INT.**



**MACRO**



**REC. SIN. EXT.**



**GER. SIN. EXT.**

Figura 11: Símbolos da LEDS

fronteiras do quadro. Isto pode ser feito pela adição de padrões especiais de bits no início e no final do quadro. Estes padrões podem acidentalmente ocorrer na cadeia de dados e assim cuidado especial precisa ser tomado de modo a evitar confusão.

Um ruído na linha de transmissão pode destruir um quadro completamente. Neste caso, o software (ou o hardware) de nível 2 que enviou o quadro precisa retransmiti-lo. O que pode ocorrer, no entanto, é que transmissões múltiplas de um mesmo quadro introduzem a possibilidade de quadros duplicados. Por exemplo, isto pode ocorrer se o quadro de reconhecimento for destruído. É função do nível 2 resolver todos os problemas relacionados com alteração, perda ou duplicação de quadros, de forma que o nível 3 pode assumir que está trabalhando com uma linha livre de erros. Outra característica do nível 2 é que ele executa controle de fluxo, de modo que o transmissor não envia dados em quantidade maior do que o receptor pode suportar em dado período.[TAR86]

#### 2.4.2 O protocolo LAPD

Na RDSI, a sinalização usuário-rede para todas as conexões da comutação de circuitos é feita em um canal independente chamado canal D, diferentemente do que ocorre em redes convencionais, tais como a rede telefônica, onde a sinalização é feita no mesmo canal em que o circuito é comutado. Desta forma o canal D é compartilhado pelos diversos terminais da interface usuário-rede, que utilizam enlaces de dados de camada 2 independentes, para a troca de mensagens de sinalização.[KANO86]

O protocolo de acesso da RDSI, seja num acesso básico, seja num primário, possui as camadas de suporte (1 a 3) do modelo OSI, sendo que o Protocolo de Nível de Enlace é o LAPD (Link Access Procedure on the D Channel).

**Objetivos do Protocolo** - Sendo um protocolo de nível 2, o LAPD objetiva transferir informações entre entidades de nível 3 na interface usuário-rede da RDSI, livre de erros que eventualmente tenham ocorrido durante a transmissão, compreendendo, ainda, um controle de fluxo e os procedimentos de estabelecimento e liberação do enlace.

Conforme definido nas recomendações I.440 e I.441 do CCITT, o LAPD suporta instalações com múltiplos terminais na casa do usuário, além de possuir pontos de acesso para múltiplas entidades de nível 3.

Da mesma forma que o protocolo X.25 - LAPB, usado para acesso a uma Rede de Comutação de Pacotes, o LAPD é baseado no Protocolo HDLC (High Level Data Link Control). No entanto, enquanto o primeiro pode suportar apenas um enlace lógico sobre uma linha física, o segundo pode suportar vários terminais na interface usuário-rede, permitindo ainda a comunicação de extremo-a-extremo de um grande número de entidades de nível 3 através de seus Pontos de Acesso ao Serviço (SAPs).

**Funções do LAPD** - As funções do protocolo podem ser divididas em funções a nível de bit e funções de comunicação.

Estão classificadas na primeira categoria:

- Transparência de dados: inserção e retirada de bits 0;
- Delimitação de quadros através de flags;
- Detecção e recuperação de erros de transmissão através de CRC;
- Reconhecimento de endereço;

Estão classificadas na segunda categoria:

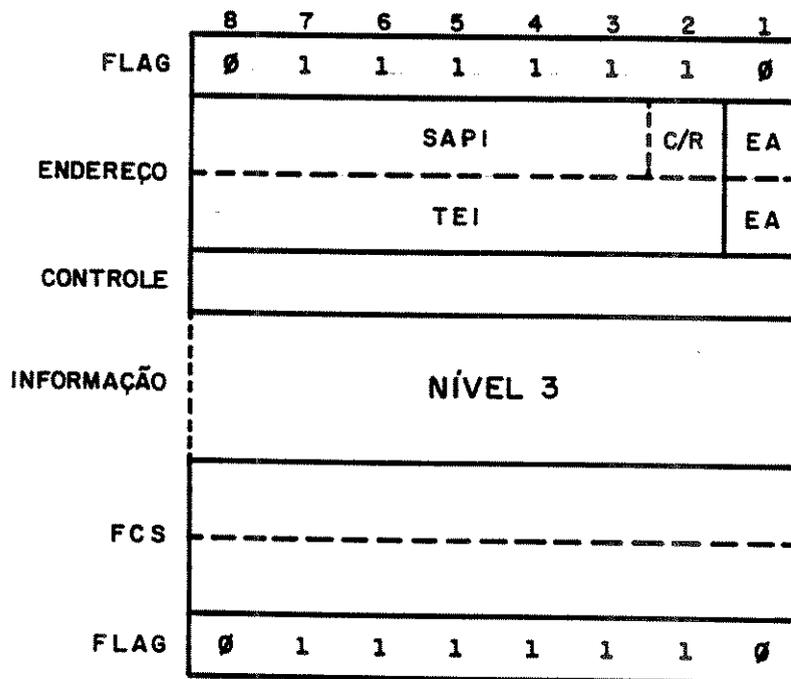
- Detecção e recuperação de erros de operação na conexão de enlace;
- Notificação à entidade de gerenciamento da ocorrência de erros não recuperáveis;
- Controle de fluxo de quadros de informação;
- Controle de sequência de quadros de informação numerados em módulo até 128;
- Rejeição de quadros de informação quando for o caso;
- Rejeição de quadros inválidos;
- Conexão e desconexão de enlaces;
- Assinalamento da identificação de um equipamento terminal (TEI);
- Transmissão e recepção de informação não numerada.

**Estrutura do Quadro LAPD** - O formato geral do quadro de informação está mostrado na figura 12.

O campo de endereço utiliza dois octetos que carregam a informação de Identificação do Equipamento Terminal (TEI) que irá receber o quadro e a Identificação do Ponto de Acesso ao Serviço (SAPI) oferecido pela rede.

O SAPI identifica, com o seu valor, os procedimentos aplicados a um dado quadro (de sinalização, de comunicação por pacotes, funções de manutenção, etc). Ele poderá ter valores que vão de 0 a 63, sendo este último conhecido como SAPI de grupo, que é sempre usado para difusão *broadcasting*.

O TEI pode assumir 128 diferentes possíveis valores, identificando uma conexão particular dentro de um Ponto de Acesso ao Serviço. Inicialmente o valor de TEI de um terminal não é fixo. Somente quando o dispositivo é conectado fisicamente à rede é que irá ocorrer a designação do valor de TEI. Este assinalamento é pedido pelo terminal e ocorre automaticamente. O terminal



EA: Bit de extensão do campo de endereço

C/R: Bit comando/resposta

Figura 12: Estrutura Geral de Quadros LAPD

pode sugerir um valor particular de TEI, mas é a rede quem irá determinar se o valor sugerido é aceito ou não. Depois dos procedimentos de designação do TEI, cada terminal terá um valor diferente para o mesmo, que será usado em toda transferência de informação. Este valor fica válido até que o terminal seja desconectado da rede, ou quando a rede decide que aquele valor deixou de ser válido. O valor TEI=128 é usado para difusão.

O bit C/R no octeto SAPI identifica um quadro como sendo um comando ou uma resposta. O terminal deve transmitir um comando sempre com C/R=0 ou uma resposta com C/R=1. A central transmite um comando com C/R=1 ou uma resposta com C/R=0.

O campo de controle, que identifica o tipo de quadro que está sendo transmitido, poderá utilizar um (quadros não numerados) ou dois (quadros de informação e supervisão) octetos. Contém o bit P/F indicador de solicitação de comando ou resposta e, quando tiver dois octetos, os números de sequência de transmissão  $N(S)$  e de recepção  $N(R)$ , que permitem um controle de numeração.

O campo de informação, cujo tamanho é um múltiplo de 8 bits, é utilizado pelos quadros I (informação), UI (informação não numerada) e XID (identificação) para enviar dados e pelo quadro FMRJ para rejeição de quadros não recuperáveis pela retransmissão.

O campo CRC (FCS) é utilizado como sequência de verificação de quadro através de um código polinomial gerado na transmissão e conferido na recepção,

permitindo a detecção de erros ocorridos devido a problemas no meio físico de transmissão.

**Tipos de Quadros** - Os quadros no protocolo LAPD podem ser divididos em 3 categorias:

1. de *informação (I)* com controle de sequência N(S)

Carregam no campo de informação os dados provenientes da camada 3

2. de *supervisão*

Utilizados para a execução de funções de controle e supervisão, tais como reconhecimento de quadros de informação, pedidos de retransmissão de quadros I e pedidos de suspensão temporária de transmissão de quadros I.

Estão compreendidos nesta categoria:

- RR (Receiver Ready) : confirmação de quadros I
- RNR (Receiver Not Ready) : indicação de ocupação
- REJ (Reject) : rejeição de quadros I

3. *não-numerados*

Provêm funções adicionais de controle e transferência de informação não-numerada.

Figuram nesta categoria:

- SABME (Set Assynchronous Balanced Mode Extended) : pedido de inicialização de enlace
- DISC(Disconnect) : pedido de desconexão de enlace
- UA (Unnumbered Acknowledge) : confirmação de recepção de comando SABME ou DISC
- UI (Unnumbered Information) : informação não numerada e sem confirmação
- DM (Disconnect Mode) : indicação de enlace desconectado
- FRMR (Frame Reject) : indicação de quadro inválido
- XID (Exchange Identification) : indicação de informações a serem trocadas entre as entidades

**Procedimentos do Protocolo** - A figura 13 ilustra a utilização de cada um desses quadros.

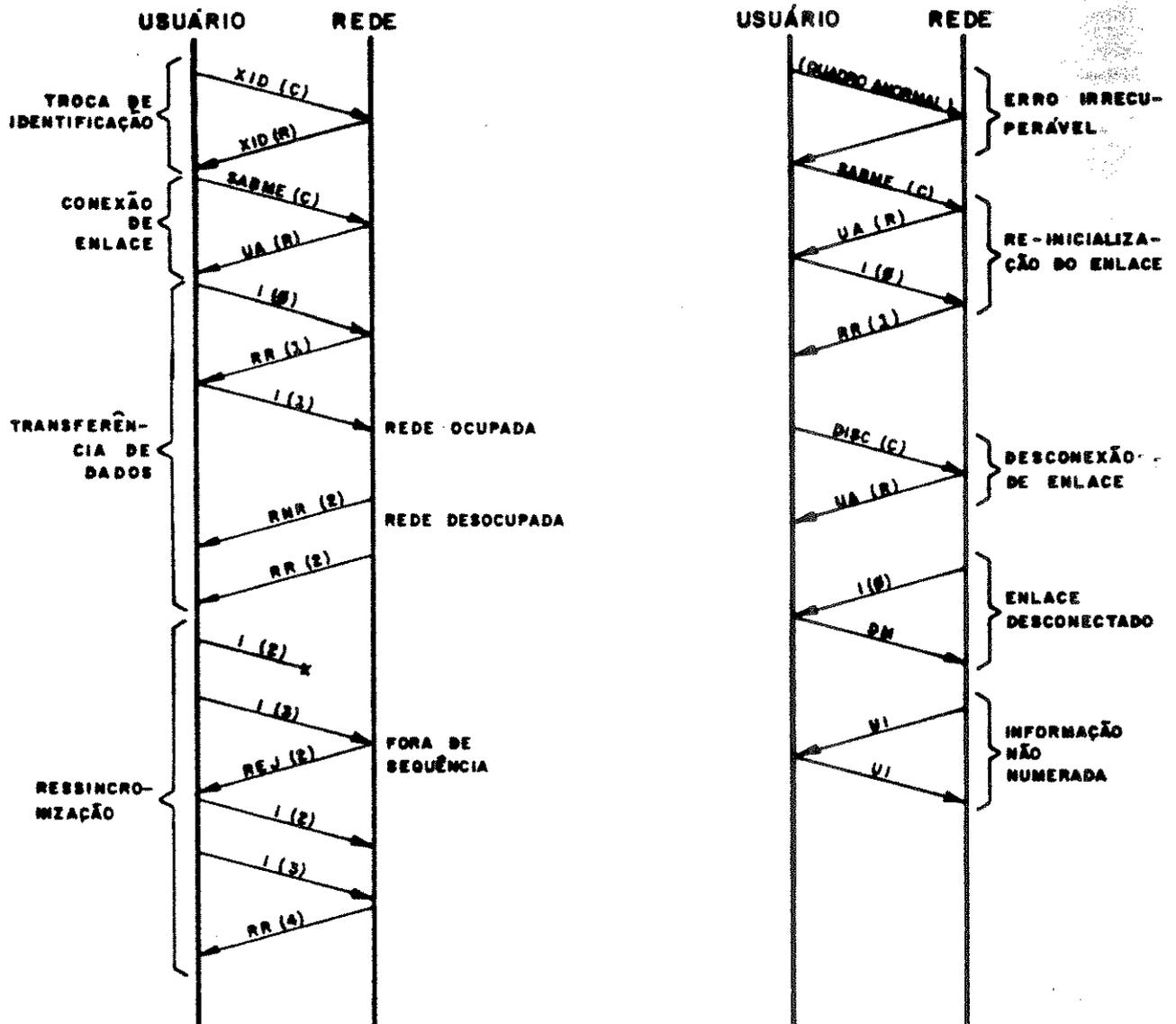


Figura 13: Utilização dos Quadros LAPD

**Descrição via LEDS** - O anexo I traz, na forma da Linguagem de Especificação e Descrição de Sistemas (recomendação CCITT, série Z), todo o protocolo LAPD.

## 2.5 Microeletrônica e RDSI

O sucesso de um projeto de RDSI reside na tecnologia e nos produtos de muitas indústrias, sendo uma das mais significativas a indústria de semicondutores. A quantidade de potência de processamento requerida para implementar os protocolos propostos não é desprezível e as facilidades de transmissão de dados necessárias no nível físico são complexas. Para que a RDSI seja viável é imperativo que tais funções sejam oferecidas a baixos custos, e isso é um desafio para a microeletrônica e a tecnologia VLSI.

O fornecimento de chips baratos para a RDSI é a chave para o crescimento do mercado. Da mesma forma, a RDSI é importante para a indústria de microeletrônica como um enorme mercado potencial.[EBLF87b]

À medida que os sistemas se tornam mais complexos, a tendência dos projetos tem sido a de particionamento de conjunto de circuitos em blocos funcionais bem definidos. Tal particionamento localiza os circuitos que executam uma função particular em posições específicas no sistema, promovendo, então, modularidade. Modularidade é uma consideração de grande importância, particularmente para atingir flexibilidade na configuração dos sistemas. Sistemas com possibilidade de configuração para diferentes usos ou situações são econômicos para projetar e atrativos ao usuário. Modularidade é também uma excelente defesa contra a obsolescência. A quantidade de investimento num dado projeto é muito alta para ser jogada fora, especialmente quando apenas algumas porções devem ser melhoradas ou atualizadas. Ambas as filosofias de projeto citadas só podem ser implementadas se o sistema é projetado de forma modular desde o início.[COSTA87]

É necessário ter um efetivo e bem definido meio de comunicação e transferência de informação entre módulos. A definição de uma interface comum entre blocos funcionais facilita a adição de novos circuitos a um projeto; esta interface deve ser simples de forma a permitir que os módulos tenham uma faixa razoável de complexidade. Algumas interfaces estão se tornando virtuais padrões para interconexão de chips para telecomunicações, em particular, para a RDSI e contam com um número cada vez maior de fabricantes que as adotam. Entre estas interfaces figuram a SLD (Subscriber Line Datalink) e a IOM (ISDN Oriented Modular) Geração 1<sup>2</sup>, a primeira definida por grandes fabricantes como Intel, Siemens e SGS e a segunda definida pela Siemens.

A interface IOM trata-se, na verdade, não de uma simples interface, mas de uma arquitetura. Um objetivo da arquitetura IOM é interconexão flexível de

<sup>2</sup>No início de 1988 foi definida uma interface IOM Geração 2, que apresenta grande número de inovações em relação à IOM de Primeira Geração. Esta nova interface parece que se tornou padrão entre os fabricantes europeus de chips para RDSI.

dispositivos para diferentes aplicações, tal como o uso de dispositivos idênticos para aplicações diferentes através de chaveamento do modo de operação. Outro objetivo é dar possibilidade de um particionamento interessante para a fase inicial de desenvolvimento da RDSI em unidades de complexidade facilmente gerenciáveis e que poderiam ser integradas posteriormente. Provavelmente a parte mais visível do conceito IOM é o barramento IOM e o seu predecessor, o barramento SLD.

A interface SLD consiste de 3 linhas: a linha de dados bidirecional, uma linha de sinal de relógio de 512 KHz e uma linha para definição do direcionamento dos dados, a 8 KHz. Em qualquer transferência entre dispositivos, um atua como mestre e outro como escravo. O dispositivo mestre gera o sinal de relógio e a linha de direção. Quando a linha de direção está em nível alto, a transferência de dados ocorre do mestre para o escravo; quando está em nível baixo, a transferência ocorre na direção contrária. Assim, 32 bits são transferidos em cada direção durante o período de 125 micro-segundos. Os 32 bits compõem 4 octetos - os octetos de dados/voz B1 e B2, um octeto de controle e um octeto de estado/sinalização.

O barramento IOM é uma conexão a quatro fios que inclui 2 linhas de dados - uma para transmissão e uma para recepção. A direção dos dados é do ponto de vista do usuário. Duas outras linhas definem relógio de bit e sincronização de quadro. O relógio de bit é controlado por um sinal de 512 KHz; o sincronismo de quadro é controlado por um sinal de 8 KHz.[LERACH86]

## 2.6 O Particionamento Adotado e Outros Particionamentos Possíveis

Dos 7 níveis definidos numa RDSI, os 3 primeiros são de importância fundamental para os projetistas de CIs. Para eles, há a crucial necessidade de se determinar quantos níveis, e de que forma, suportar num único chip, ou ainda, como fazer a distribuição de funções por um conjunto de chips de forma inteligente, onde a modularidade fica assegurada.

Limitar-nos-emos a apresentar algumas possíveis divisões, sem entrar no mérito de suas vantagens e desvantagens, e apresentaremos, mais detalhadamente, a arquitetura que sugerimos.

1. Primeira Possibilidade : CI dedicado ao tratamento da camada física apenas, ficando os níveis 2 e 3 resolvidos por software em processador. Esta é uma solução tradicional.
2. Segunda Possibilidade : CI dedicado ao tratamento da camada física mais as funções de nível de bit (funções básicas) da camada de enlace; o restante da camada de enlace e o nível 3 são resolvidos por software em processador. Esta é uma solução já bastante encontrada, principalmente com chips que tratam parte do protocolo HDLC.

3. Terceira Possibilidade : Funções de camadas 1 e 2 completas num único chip. Esta solução é algo a ser atingido.
4. Quarta Possibilidade : Um chip para tratamento de camada 1 e outro que, além de parte de funções de sinalização, traz incluídas funções de codecs e filtros ou outras funções de áudio para tratar um canal B.

E, assim, a divisão pode atender a uma série ilimitada de possibilidades, melhores ou piores dependendo da aplicação imaginada. A visão que adotamos para componentes VLSI que podem ser utilizados num acesso básico de assinante ponto a multiponto da RDSI é apresentada na figura 14 . Note-se nesta concepção que os protocolos com as funções de camadas 1 e 2 ficam resolvidas a nível de hardware (nos circuitos integrados dedicados) e as funções de camada 3 serão realizadas pelo microprocessador. Também a cargo de CIs dedicados estão o controle de funções de transferência e manipulação dos sinais de dados e voz. Nesta figura, e no restante deste trabalho, usamos a seguinte nomenclatura:

- CH1-S : circuito integrado que executa as funções da camada 1 da interface S.
- CH1-U : circuito integrado que executa as funções da camada 1 da interface U.
- CH2 : circuito integrado que executa as funções da camada 2 do acesso do usuário à rede.

Note-se que nesta arquitetura os seguintes critérios são atendidos:

- alta modularidade, garantindo interconexão flexível dos dispositivos para diferentes aplicações.
- o uso de dispositivos idênticos em diferentes aplicações, via chaveamento do modo de operação, resulta em um número mínimo de CIs individuais. Esta estratégia leva a uma redução nos custos da fase inicial de implantação da RDSI.
- tal particionamento na fase inicial permite a implementação de dispositivos com nível de complexidade gerenciável em relação à segurança de desenvolvimento, recursos e tempo.
- uma interface bem definida permite aos projetistas de sistemas e equipamentos adaptar suas soluções intermediárias a este conceito.

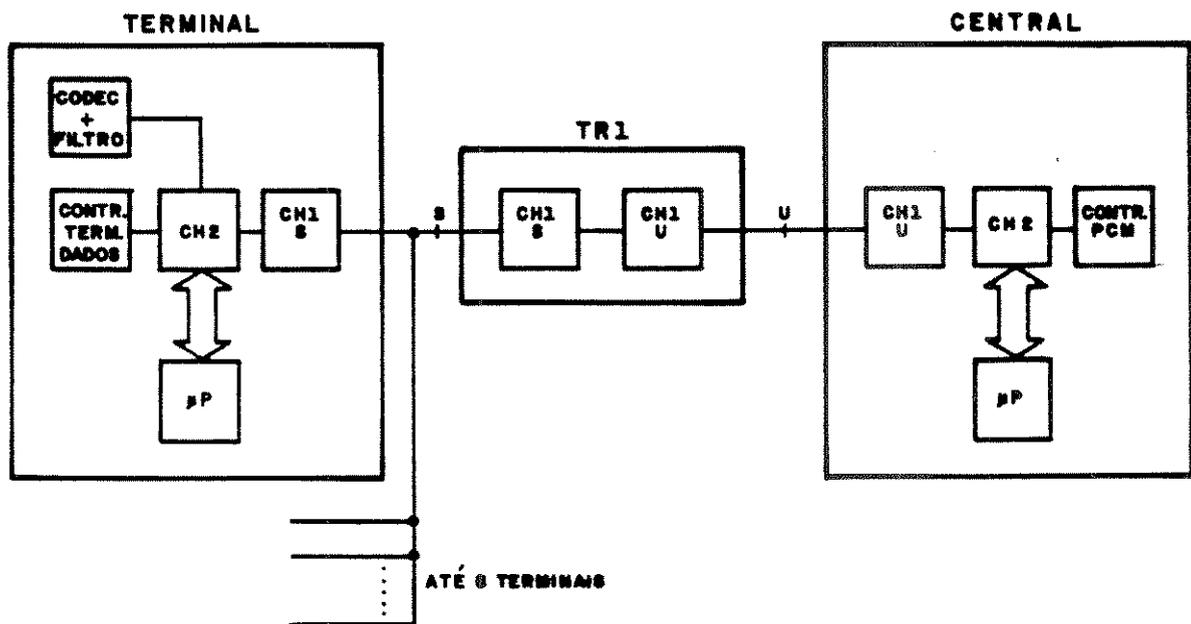


Figura 14: Circuitos Integrados num Acesso Básico RDSI

## **Capítulo Três**

### **Uma Proposta de Especificação**

## 3 Uma Proposta de Especificação

Este capítulo, que é o núcleo do trabalho, apresenta uma descrição detalhada de uma proposta de implementação de um circuito integrado cuja função principal é resolver, com um mínimo de atuação do processador, todo o protocolo de nível de enlace do acesso do usuário à rede numa RDSI. É objetivo também deste capítulo a discussão do interrelacionamento do chip, ora chamado CH2, com outros chips de interesse numa RDSI.

Para tanto, é dividido nas seguintes seções:

- 3.1. Introdução
- 3.2. Descrição do Bloco de Interfaces
- 3.3. Descrição do Bloco Tratador de HDLC
- 3.4. Descrição do Bloco Tratador de LAPD
- 3.5. Descrição do Bloco de FIFOs
- 3.6. Descrição do Conjunto de Registradores

### 3.1 Introdução

#### 3.1.1 Visão Global

É um fato cuja verificação não é nada difícil, que a tecnologia VLSI tem caminhado no sentido de integrar cada vez mais funções que tradicionalmente eram executadas por software [SLM87]. O mercado internacional apresenta, de diferentes fontes, circuitos integrados capazes de executar funções de camada física e parte da camada de enlace (funções de nível de bit), aliviando, assim, o processador de tais operações. O CH2 se apresenta como um participante da nova geração de circuitos integrados que deverão surgir, a curto prazo, em que também as funções de comunicação da camada de enlace serão integradas.

O nível de enlace é tratado na RDSI em quatro grupos funcionais: ET1, AT, TR2 e TC. Para que possa operar em qualquer desses equipamentos, o CH2 deve ter uma configuração flexível. Assim, quando operando no terminal (ET1, AT), deve poder se ligar a chips que executem funções de camada 1 da interface S (CH1-S) de um lado e com módulos inteligentes para tratamento de voz (codecs + filtros) e dados (USARTs) de outro, além de se comunicar com o processador. Quando operando na central (TC) ou no TR2 (PABX), deve estar configurado de forma a se interligar com chips tratadores de funções de camada 1 da linha de transmissão (CH1-U) de um lado e de outro com módulos que se destinam a executar funções típicas de centrais (comutação e multiplexação), do tipo de *Controladores de Placas de Linhas de Assinante* que tem aparecido no mercado de microeletrônica, além da comunicação com o processador.

Tal flexibilidade só é atingida se forem muito bem definidas as vias de comunicação do chip, seguindo a tendência internacional de padronização, buscando, inclusive, um meio de permitir aos projetistas de sistemas e equipamentos a sua utilização junto a circuitos integrados fornecidos por outros fabricantes que adotem tais interfaces.

A interface IOM é a utilizada para fazer a comunicação com os chips tratadores de camada física. A utilização de componentes de camada 1 implica na eliminação de todo um conjunto de sinais de controle que informariam o processador do estado das interfaces S e U, de forma que tais informações sejam transferidas para ele através do CH2; em alguns casos, o próprio CH2 faz o tratamento desta informação, aliviando ainda mais o processador. A interface utilizada para possibilitar a interação do processador com codecs/filtros e USARTs nos terminais e com controladores de placas de assinantes nas centrais é denominada SLD. A nova geração de módulos de voz e dados tem aparecido com tal interface, cuja característica principal é o uso de apenas três sinais, minimizando os barramentos de dados e controle para comunicação com o processador, simplificando o layout das placas e diminuindo o número de pinos necessários nos CIs. No entanto, como existe uma grande quantidade de chips que executam as funções de codecs e USARTs, alguns já bastante consagrados, e que não têm interface SLD, o CH2 provê ainda uma terceira interface, a SSI.

Num acesso básico da RDSI, quando operando no terminal, o CH2 irá executar autonomamente os procedimentos do protocolo do nível de enlace (LAPD) que circula pelo canal D a 16 Kbit/s (ou, em algumas aplicações, pode circular também no canal B a 64 Kbit/s), fazendo ainda a separação dos canais B e transferência dos mesmos para os devidos módulos tratadores de voz e dados. Este é o modo de operação principal previsto para o chip, e o mais complexo. Neste caso, as mensagens da camada 3, que são transportadas no campo de informação do quadro LAPD, são transferidas ao processador para tratamento através de memórias FIFO internas, que funcionam como armazenadores temporários de pacotes de dados. Não existe, a nível de CH2, limitação em relação ao comprimento máximo dos quadros que circulam pelo canal D; esses buffers com 64 octetos por direção (recepção e transmissão) determinam apenas o tempo máximo permitido para o processador reagir antes da ocorrência de transbordamento (overflow) ou falta (underflow) de dados.

É possível ainda ao CH2 operar com 3 outros modos, de forma a poder ser utilizado em outras situações tanto na própria RDSI, como fora dela.

A interface com o processador prevê possibilidades de interligação direta com várias famílias de microprocessadores (MCS48, MCS51, 8085) e, com pequenas mudanças no hardware externo, é possível a interligação com outras famílias.

### 3.1.2 Arquitetura Funcional

A figura 15 apresenta a arquitetura interna do chip, onde pode-se distinguir quatro grandes blocos: Bloco de Interfaces, Bloco Tratador de HDLC, Bloco Tratador de LAPD e Bloco de FIFOs.[EBLF87a]<sup>1</sup>

<sup>1</sup>Esta divisão não é apenas didática; ela permite a distribuição do esforço de desenvolvimento em grupos estanques de projetistas, desde que as interfaces entre os blocos estejam bem definidas

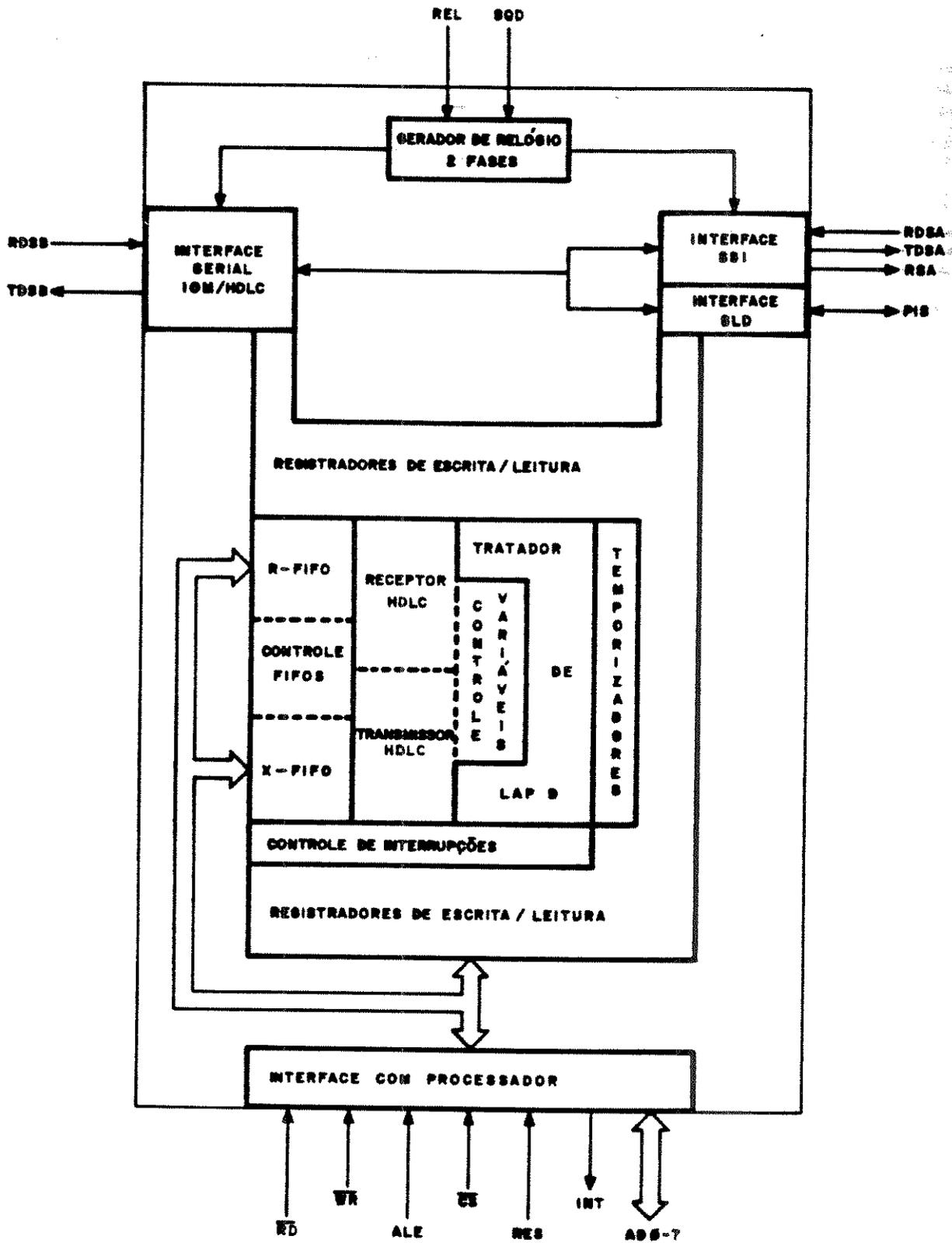


Figura 15: Arquitetura Funcional

O Bloco de Interfaces compreende a interface com o processador, as portas seriais A (SSI) , B (IOM/HDLC) e C (SLD) e a unidade geradora de fases de relógio.

O Bloco Tratador de HDLC (THDLC) é responsável pelo tratamento das funções de nível de bit do protocolo HDLC, tanto na transmissão quanto na recepção, além da execução de outras funções de apoio ao Bloco TLD.

O Bloco Tratador de LAPD é responsável pelo tratamento dos procedimentos de comunicação do protocolo LAPD.

O Bloco de FIFOS inclui tanto as memórias em si, quanto todo o circuito que as controla.

O Conjunto de Registradores de comando e estado faz parte dos 4 grandes blocos citados, de acordo com a função executada por eles. O controlador de interrupções é o responsável pelo gerenciamento dos diversos tipos de interrupções provenientes dos 4 blocos.

### 3.1.3 As Portas Seriais de Comunicação

Conforme já foi dito, o CH2 possui três portas seriais com diferentes características, através das quais o chip pode se comunicar com o meio externo. São elas : a porta serial A, a porta serial B e a porta serial C. Ainda que fisicamente tais portas sejam em número de três, através de configurações, um número maior de interfaces é conseguido.

O CH2 possui uma entrada de sinal de relógio (REL) e uma entrada para um sinal de sincronismo, gerado a partir de REL no dispositivo fornecedor dos mesmos, chamado Sincronismo de Quadro (SQD). Tais sinais são compartilhados por todas as interfaces; a partir deles, no módulo *gerador de fases de relógio* do Bloco de Interfaces , são gerados todos os outros sinais internos necessários às interfaces.

É parte integrante de todas as portas seriais uma ou mais vias de comunicação de dados (por onde circulam os quadros); no caso específico da porta A existe um sinal de relógio (RSA) de saída que pode ser utilizado pelo sistema onde o chip é aplicado.

A porta A, onde está a interface SSI, possui duas vias para comunicação de dados; uma para transmissão (TDSA) e outra para recepção (RDSA). Esta interface é destinada a suportar Codecs ou Controladores de Dados a uma taxa de 64 Kbps ou 128 kbps que não possuem interface SLD.

A porta B é a mais complexa. Numa configuração ela comporta uma interface IOM e noutra ela é uma interface para dados em formato HDLC. Duas vias de dados estão na porta B: uma para transmissão (TDSB) e outra para recepção (RDSB).

Finalmente a porta C funciona como interface padrão SLD; porisso apenas uma via de comunicação existe, chamada PIS.

No item 3.2, Descrição do Bloco de Interfaces, são apresentados maiores detalhes.

### 3.1.4 Aplicações Previstas para o CH2

Neste item são mostradas algumas aplicações que se visualizam para o CH2 da forma como está sendo especificado. Aplicações outras poderão, no entanto, aparecer, com pequenas alterações a serem feitas nesta sugestão.

Devido à grande gama de diferentes modos de operação e aplicações previstas para o CH2, é interessante deixar claro, já nesta fase, quais são as decisões a serem tomadas na programação do mesmo (Registrador de Modo).

- Primeira Decisão: Como operará a Porta B (bits HMD2-0)

A porta B pode ser programada como interface IOM ou como interface HDLC. No primeiro caso é uma aplicação necessariamente na RDSI, onde existem os canais B1, B2 e D, além dos campos de controle. No segundo caso trata-se de usar o CH2 como um controlador HDLC inteligente de propósitos gerais, onde a taxa de relógio de dados é a mesma taxa do sinal REL e um sinal de strobe (entrada SQD) determina a taxa de dados.

- Segunda Decisão: Modo de Operação do Controlador HDLC Interno (bits SMD1-SMD0)

O grau de tratamento do quadro recebido pela Porta B é definido neste momento. Se a Porta B está programada como interface HDLC, não é possível programar o chip no modo automático. Se, no entanto, a Porta B está programada como interface IOM, qualquer um dos quatro modos de operação é possível.

- Terceira Decisão: Seleção do Canal a ser Tratado (bits SCL1-SCL0)

Se a Porta B está programada como interface IOM, é possível selecionar qual canal (B1, B2 ou D) vai ser tratado pelo Tratador de LAPD Interno (modo automático) ou vai ser enviado à R-FIFO (outros modos). Vale salientar que os canais B1 ou B2 somente poderão ser tratados se carregarem um protocolo do tipo HDLC.

**Chip no Lado do Terminal com Interface IOM** Quando usado no lado do terminal, seja em ET1 ou em Adaptadores de Terminais (AT), as aplicações para o CH2 abrangem tanto voz quanto dados. Este terminal pode estar ligado a uma interface S (tanto a um TR1 quanto a um PABX), a uma interface U diretamente ou ainda a uma interface U' utilizada para terminais remotos se interligarem à PABXs.<sup>2</sup>

<sup>2</sup>Para a interface U, que é a linha de transmissão, a tendência mundial (e a decisão nacional brasileira) é a de utilização de híbrida adaptativa com cancelamento de eco, enquanto que para terminais remotos de PABX, a interface U' deverá ser a adotada, onde uma técnica mais simples chamada TCM (Time Compressed Multiplexed) ou ping-pong é usada.

**Tratando Canal D** Neste caso, o canal D da Interface IOM será enviado ao Tratador HDLC interno e os canais B1 e B2 serão repassados, via portas A e C, a chips que executam funções típicas de terminais (tratamento de voz ou dados), ou seja, Codecs para voz e Controladores de Protocolos do tipo das USARTs para dados (por exemplo, o próprio CH2 quando tem a Porta B programada como interface HDLC).

A utilização simultânea ou não das funções associadas à porta A e à porta C pode ocorrer desde que o chaveamento dos canais B seja devidamente programado.

Nesta aplicação o canal D carrega o protocolo LAPD e o chip deve ser programado para operar no modo automático, quando então ele resolve autonomamente toda a camada de enlace de sinalização no canal D. No entanto, existe a opção de programação nos outros modos menos completos (não automático, mboxtransparente e transparente estendido) se se preferir um tratamento do protocolo externamente ao chip (isto é, no processador).

### **Tratando Canais B**

- **Retirados Diretamente da IOM**

O canal B1 ou o canal B2 da interface IOM pode ser enviado ao Tratador HDLC interno, desde que carregue um protocolo tipo HDLC. Se este protocolo for o LAPD (caso possível em aplicações de Terminais Telemáticos), o modo automático pode ser programado e o tratamento do protocolo é feito pelo CH2. Caso o protocolo seja outro (por exemplo, LAPB, LLC, etc), outros modos de operação (não automático, transparente, transparente estendido) devem ser programados.

Nesta aplicação as Portas A e C ficam sem função.

- **Recebendo os Canais de outro CH2 operando como Tratador de Canal D**

Esta configuração é chamada Mestre-Escravo. O CH2 escravo tem a Porta B programada como interface HDLC. Uma vez que neste modo o TLD é desativado, não é possível programar o CH2 no modo automático.

**Chip no Lado da Central com Interface IOM** O CH2 tem aplicações nos grupos funcionais que empregam as funções de comutação, isto é, TR2 (PABX) e TC(nó RDSI). De um lado, aquele que se interliga com a linha da interface S (TR2) ou U (TC), o CH2 se interfaceia com componentes tratadores de camada 1 através da interface IOM (porta serial B), que deverá estar operando no modo multiplexado ( $REL = n \times 512 \text{ KHz}$ , onde  $n$  varia de 1 a 16). Neste caso tem-se 8 quadros IOM multiplexados no tempo em 125 micro-segundos, onde cada quadro IOM completo é definido dentro de cada *time-slot*; o CH2 trata apenas um *time-slot*, que é programável de forma semelhante à aplicação em Terminal.

**Chip como Controlador HDLC de Propósitos Gerais** A Porta B pode ser programada como interface HDLC, quando então o CH2 passa a funcionar como um Controlador HDLC inteligente de apenas um canal onde qualquer protocolo do tipo HDLC possa estar circulando (neste modo de operação o Controlador de LAPD fica desativado), a uma frequência máxima de 4 MHz.

Nos bits HMD2-0 do Registrador de Modo pode-se programar diferentes modos de operação para o Controlador HDLC : aquisição contínua de dados, aquisição com sinal de *strobe* ativo em nível lógico 1 ou, ainda, aquisição com sinal de *strobe* ativo em nível lógico 0.

### 3.1.5 Pinagem Sugerida

A figura 16 mostra uma pinagem sugerida para o CH2. A tabela 2 descreve todos os sinais de entrada/saída.

| Pino  | Sinal     | Tipo | Descrição                                                                         |
|-------|-----------|------|-----------------------------------------------------------------------------------|
| 1 a 8 | AD0 a AD7 | E/S  | Dados e Endereços multiplexados                                                   |
| 9     | REL       | E    | Sinal de Relógio para o CH2                                                       |
| 10    | CS        | E    | Sinal de seleção do CH2 para uma operação de escrita/leitura                      |
| 11    | ALE       | E    | Indicação de endereço presente no barramento de dados/endereços externos          |
| 13    | RES       | E    | Sinal de Reset                                                                    |
| 14    | WR        | E    | Sinal de Leitura pelo processador                                                 |
| 15    | RD        | E    | Sinal de Escrita pelo processador                                                 |
| 16    | SQD       | E    | Sinal de sincronização (modo IOM) ou de <i>strobe</i> (modo HDLC)                 |
| 17    | PIS       | E/S  | Dados seriais da interface SLD                                                    |
| 18    | RDSB      | E    | Recepção de dados seriais na porta B                                              |
| 19    | TDSB      | S    | Transmissão de dados seriais na porta B                                           |
| 20    | RDSA      | E    | Recepção de dados seriais na porta A                                              |
| 21    | TDSA      | S    | Transmissão de dados seriais na porta A                                           |
| 22    | RSA/SQA   | S    | Sinal de Relógio de 128 KHz para porta A sincronismo de quadro atrasado (central) |
| 23    | INT       | S    | Sinal ativado quando o CH2 pede uma interrupção ao processador                    |

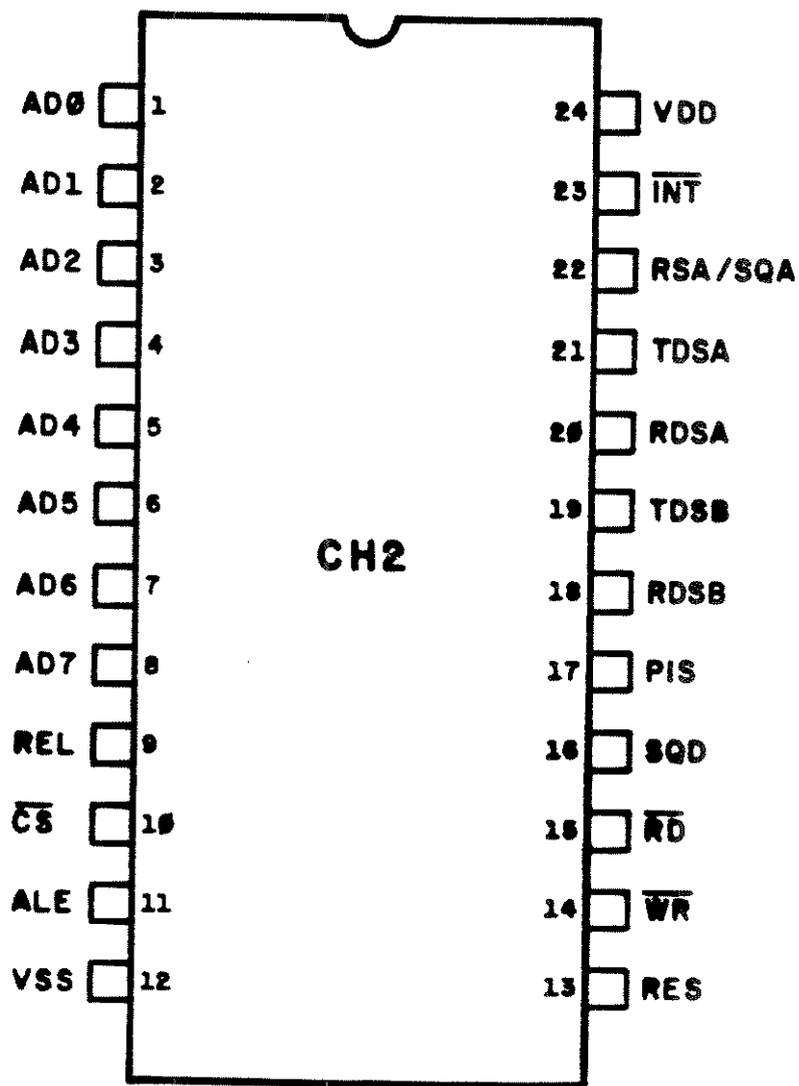


Figura 16: Pinagem Sugerida

## 3.2 Descrição do Bloco de Interfaces

O CH2 apresenta quatro diferentes tipos de interfaces orientadas para o usuário:

- *Interface com o Processador* - paralela, permite que o processador se comunique com o CH2; com isso, as funções de nível mais alto do modelo OSI poderão ser executadas por software no primeiro;
- *Interfaces para Canais B* - são as interfaces seriais SSI e SLD, que funcionam como fontes/destinos de informações que circulam nos canais B1 e B2;
- *Interface IOM* - serial, permite a comunicação com componentes de camada 1 na RDSI <sup>1</sup>;
- *Interface HDLC* - serial, permite que o CH2 opere como controlador de protocolos do tipo HDLC em aplicações na própria RDSI ou fora ela.

Estas interfaces, todas padronizadas, são as mesmas adotadas no circuito integrado PEB 2070 - ICC, da Siemens.

### 3.2.1 Estudo das Interfaces

Além da interface paralela com o processador, o CH2 possui três portas seriais de comunicação: porta A, que suporta a interface SSI, porta B que ora suporta a interface IOM, ora a interface HDLC e a porta C que suporta a interface SLD.

**Interface com o Processador** A interface com o processador consiste de transceptores de barramentos (dados e endereços), registradores de comunicação e lógica para controle desses barramentos. Através dessa interface, o CH2 se conecta a barramentos de dados/ endereços multiplexados de um sistema baseado em microprocessadores que adotem tal filosofia, tais como as famílias MCS-51, MCS-48 (microcontroladores) e MCS-85, MCS-86 E MCS-88.

Através de operações de leitura e escrita em registradores especiais do CH2 (vide item 3.6), as seguintes funções podem ser executadas:

- transferência de pacotes de dados no canal D (sejam de sinalização provenientes dos processos de camada 3 ou de gerência, de envio de dados via pacotes com propósitos diferentes de sinalização, ou de telemetria);
- controle das funções de camada 2 da RDSI (através de registradores que controlam o campo de endereço e parâmetros do protocolo);

---

<sup>1</sup>Aqui é adotada a versão 1 desta interface, conforme foi definida pela Siemens A. G.. Uma nova geração desta interface, chamada *IOM Revision 2*, com estrutura e procedimentos modificados, está em uso a partir do final de 1988. Esta nova versão é suportada por empresas européias de grande porte (contrato de adesão assinado pelas seguintes empresas: Plessey da Inglaterra, Siemens da Alemanha, Alcatel da França e Italtel da Itália). Existe compatibilidade entre componentes que adotam as versões 1 e 2.

- acesso aos canais B (registradores permitem que os canais B que circulam pelas portas A, B e C sejam monitorados pelo processador);
- chaveamento de canais B ( o processador pode comandar a maneira como deve ocorrer as comutações internas entre as portas seriais);
- controle das funções de camada 1 ( o processador se comunica com os componentes de camada 1 via registradores do CH2 e interface IOM);
- suporte de funções de diagnóstico (os erros e notificações para estatísticas são passadas ao processador via registradores de estado);
- notificação de eventos especiais (isso ocorre através de registradores de interrupção).

Fazem parte desta interface a via de 8 bits de dados/endereços multiplexados, o sinal de escrita ( $\overline{WR}$ ), o sinal de leitura ( $\overline{RD}$ ), o sinal de seleção do chip ( $\overline{CS}$ ), um sinal habilitador de latch de endereço (ALE) usado na multiplexação de endereços e dados, um sinal de saída de interrupção ( $\overline{INT}$ ) e um sinal de reset (RES).

**Porta B programada como Interface IOM** Neste item são fornecidas informações mais completas a respeito desta interface, no que concerne a seus sinais, sua estrutura de quadro e seus procedimentos.[SIEM86]

**Estrutura de Quadro** Em todas as aplicações a taxa de dados da RDSI de 144 Kbit/s (acesso básico -2B + D) precisa ser transferida através da interface IOM de forma transparente. Além disso, há a necessidade da troca de informações de controle para ativação/desativação da camada 1 e para chaveamento de *loops* de teste. Algumas aplicações requerem também capacidade para transferir informações de manutenção através desta interface. Toda essa informação é transferida num modo multiplexado por divisão no tempo baseado na estrutura de quadro mostrada na figura 17. Cada quadro, em cada direção, carrega os quatro octetos a seguir:

- os canais B1 e B2 de 64 Kbit/s, transportados nos dois primeiros octetos;
- o terceiro octeto (canal Monitor) é usado para transferir informações de manutenção entre o dispositivo de camada 2 e o componente de camada 1;
- os dois bits de canal D são transferidos no quarto octeto, assim como os quatro bits de comando/indicação (bits A) que controlam a ativação e a desativação e os bits T e E. O bit E serve para dar suporte ao manuseio do canal Monitor e o bit T é reservado para um canal transparente de 8 Kbit/s.

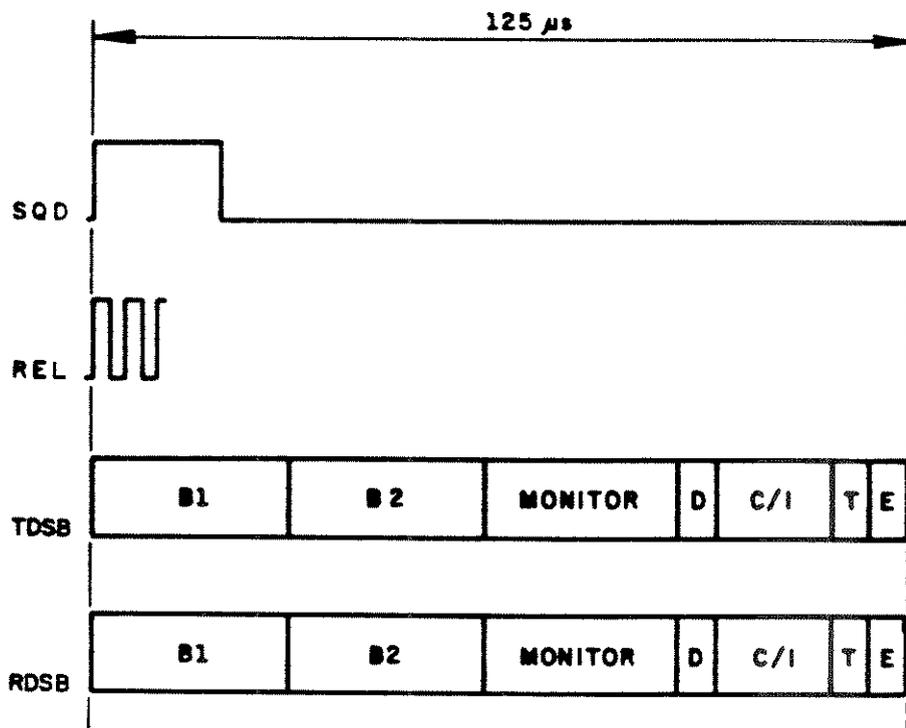


Figura 17: Estrutura de Quadro da Interface IOM

Esses quatro octetos por quadro exigem uma taxa de bits de 256 Kbit/s. A taxa de transmissão depende do modo:

- **Modo Normal (contínuo)**
  - taxa de dados: 256 Kbit/s
  - taxa de bits: 256 Kbit/s
  - frequência do sinal de relógio REL: 512 KHz
- **Modo Multiplexado**
  - taxa de dados: 256 Kbit/s
  - taxa de bits: 4 Mbit/s
  - frequência do sinal de relógio REL: 8 MHz

O diagrama de tempos da interface IOM no modo multiplexado é mostrado na figura 18.

**Sinais da Interface** A figura 17 apresenta os sinais da interface. RDSB e TDSB são as duas linhas de dados. REL e SDQ definem respectivamente o relógio de dados e o sinal de sincronismo de quadro nesta interface. No caso da central, estes dois sinais são supridos pelo relógio do sistema da central, enquanto que em outras aplicações o CI mestre gera tais sinais a fim de fornecê-los para o CI escravo.

**Procedimentos da Interface** São apresentados os procedimentos a serem executados em cada um dos canais da interface.

### **Canais B**

No estado ativo, os dois canais B são transmitidos transparentemente através da interface IOM. O octeto B1 é utilizado para carregar o canal B1 a uma taxa de 64 Kbit/s, que pode estar transportando voz ou dados. O mesmo vale para o octeto B2 em relação ao canal B2.

### **Canal de Monitoração**

O canal de monitoração é utilizado tanto no grupo funcional TL (Terminação de Linha) quanto nos terminais. No primeiro caso é usado para expandir a capacidade de transmissão; nesta aplicação é necessário ler informações a respeito da taxa de erros, coeficientes do compensador de eco e situação da fonte de potência provenientes do dispositivo de tratamento de camada 1 da interface U. Por razões de pinagem e com a finalidade de diminuir o tamanho das placas de circuito impresso onde serão colocados, o dispositivo de camada 1 não tem

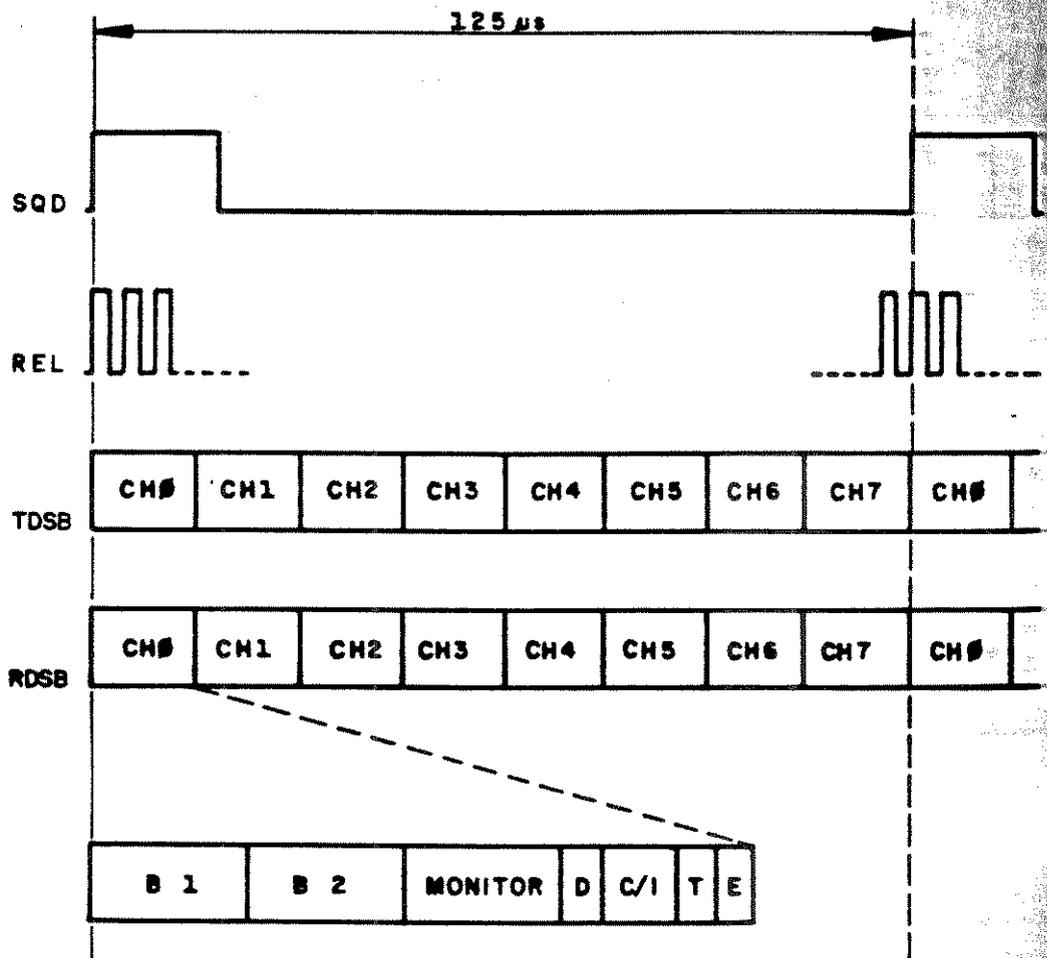


Figura 18: Estrutura de Quadros IOM Multiplexada

interface paralela para comunicação com o processador. A informação é transferida para este último via a interface IOM e o dispositivo de camada 2 (CH2) com o procedimento a seguir:

O processador escreve um comando de um único octeto num registrador especial do dispositivo de camada 2 via o barramento paralelo. O dispositivo de camada 2 envia esse comando uma única vez para o dispositivo de camada 1, seguidos de octetos de NOP. Os códigos na faixa de valores F0H até FFH são definidos como octetos de NOP. Como reação ao comando recebido, o dispositivo de camada 1 envia dados de volta para o dispositivo de camada 2 via a interface IOM.

A validade dos dados é indicada pelo bit E (canal C/I); se este bit é colocado em nível lógico 0 significa que o canal Monitor do próximo quadro contém dados válidos. O dispositivo de camada 2 armazena, então, este octeto em um registrador e gera uma interrupção ao processador.

No registrador de extensão de interrupções (REX1), o bit MOR ativado em nível 1 indica octeto do canal Monitor válido no registrador RMON.

Uma outra aplicação para o octeto de monitoração é quando o CH2 está sendo utilizado no terminal numa interface S, com um chip apropriado para tratamento de camada 1. Este chip deve transferir para o CH2 informações sobre o estado da interface S (se está livre ou ocupada) através do bit 3 do canal Monitor. Se este bit estiver em nível lógico 0, a interface está livre e o CH2 inicia a transmissão da mensagem no mesmo quadro. Se este bit estiver em nível lógico 1, a interface está ocupada e/ou houve colisão. Ao verificar esse evento, o CH2 termina a mensagem corrente pelo canal D e transmite uma sequência de *1s* (Idle state) no mesmo quadro. O CH2 somente irá transmitir a mensagem novamente se a interface S estiver livre, ou seja, somente quando o bit de controle do canal Monitor vier ativado para nível lógico 0 novamente.

### Canal Comando/Indicação

No acesso básico da RDSI a linha de transmissão, assim como o equipamento terminal, podem ficar num estado de repouso quando nenhuma transmissão está em curso. Quando é iniciada uma transmissão entre o ET e a TL, a linha de transmissão - consistindo de uma ou mais seções - precisa ser ativada. Na interface IOM os procedimentos para ativação/desativação estão contidos no canal C/I.

A troca de informações no canal C/I ocorre da forma a seguir: os dois dispositivos conectados via IOM enviam um para o outro permanentemente um código de quatro bits no campo C/I (bits A). Este código é retirado de um registrador especial do dispositivo emissor (no CH2, registrador RRCI) e depende do estado em que este se encontra. O mesmo código é enviado durante todo o tempo em que o conteúdo do registrador permanece o mesmo.

Uma mudança do comando a ser enviado é iniciado pelo processador num dos pontos extremos, escrevendo uma nova palavra de comando no registrador

### RXCI do CH2.

A mudança no código transferido é reconhecida pelo dispositivo receptor. O dispositivo de camada 1 irá interpretar o novo código e enviar informação de controle na linha de transmissão ou chavear um *loop* de teste. O dispositivo de camada 2, por outro lado, irá responder a uma mudança no código através de um pedido de interrupção ao processador.

Neste canal ainda estão contidos os bits T, E e D. O bit T é um canal reservado para comunicações de qualquer espécie a uma taxa de 8 Kbit/seg. Os dois bits D transportam o canal D a uma taxa de 16 Kbit/seg, carregando prioritariamente sinalização, mas podendo também carregar dados. O bit E faz o controle das informações que o componente de camada 1 usado na interface U (na central) passa para o processador através do canal Monitor. Quando o componente de camada 1 passa para o processador uma informação sobre coeficientes ou erros do compensador de ecos, primeiro é transmitido para o CH2 o octeto de sinalização com o bit E ativado para nível lógico 0 significando que ele reconheceu o comando anterior enviado para o componente de camada 1 e que a resposta será dada no octeto Monitor do próximo quadro a ser transmitido para o CH2. Isto irá fazer com que o CH2 interrompa o processador assim que o octeto Monitor do próximo quadro seja armazenado; a interrupção significa que o octeto existente no registrador RMON é válido e pode ser lido pelo processador.

**Porta B programada como Interface HDLC** Esta interface permite ao CH2 funcionar como um chip auxiliar na implementação de protocolos do tipo HDLC, dando margem a dois possíveis usos: aquisição contínua de dados ou aquisição de dados quando sinal de *strobe* (pino SQD) está ativo. Os bits HMD2-0 do Registrador de Modo definem o modo de aquisição de dados.

1. Aquisição contínua de dados (HMD2-0 = 111)  
A frequência do sinal de relógio no pino REL e a taxa das linhas de dados RDSB e TDSB devem ser iguais. O pino SQD é desconsiderado. Veja figura 19.
2. Aquisição segundo sinal de *strobe* ativo em nível lógico 1 (HMD2-0 = 110)  
Captura/Transmissão de dados ocorre somente durante o período de tempo em que o sinal de *strobe* (pino SDQ) estiver em nível lógico 1.
3. Aquisição segundo sinal de *Strobe* ativo em nível lógico 0 (HMD2-0 = 101)  
Captura/Transmissão de dados ocorre somente durante o período de tempo em que o sinal de *strobe* estiver em nível lógico 0.

Uma aplicação imediata dos casos 2 e 3 é a chamada *Configuração Mestre-Escravo*.

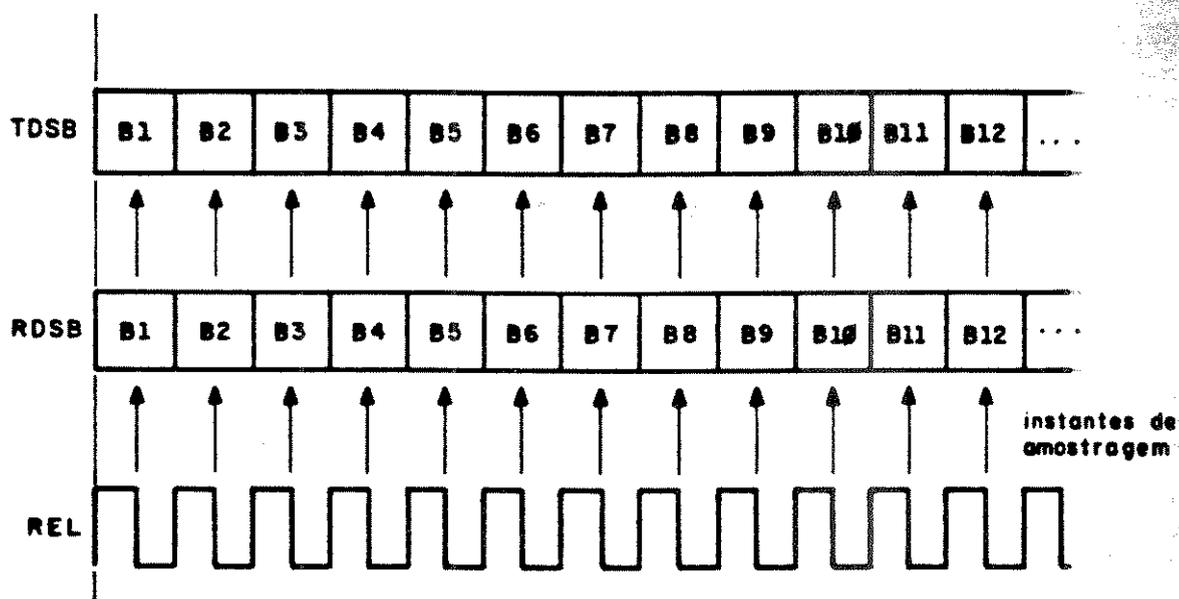


Figura 19: Interface HDLC - Aquisição contínua de dados

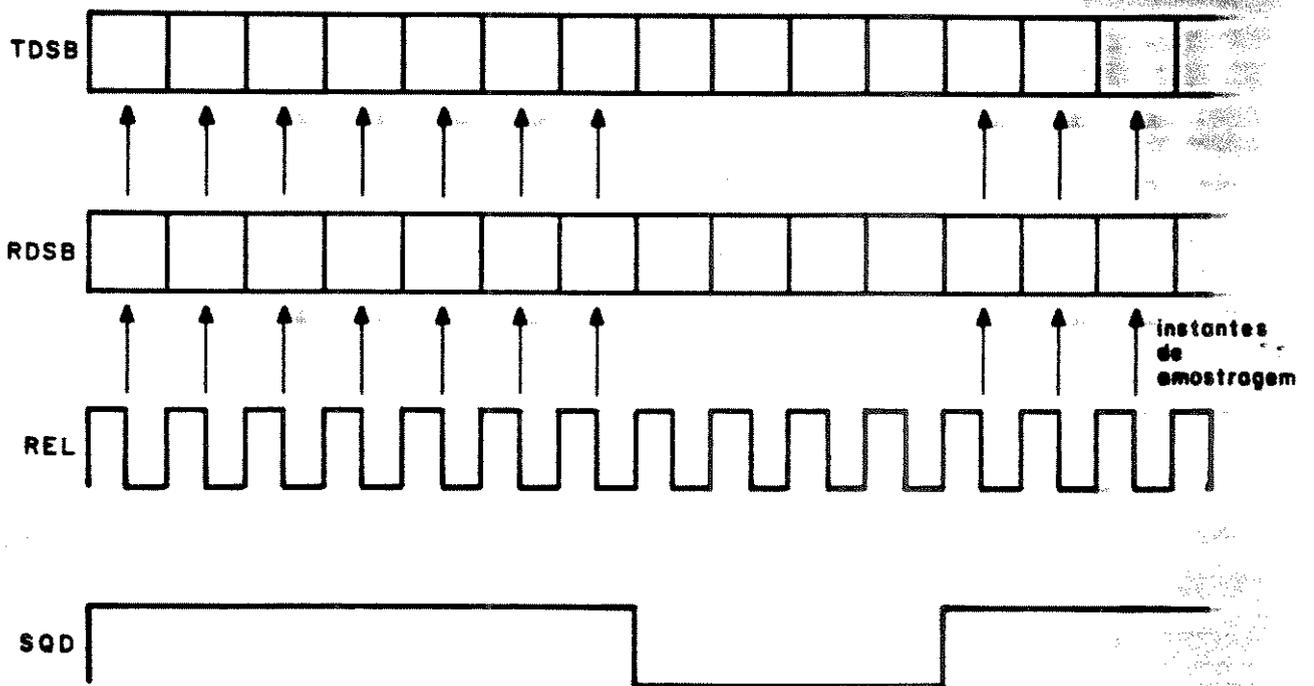


Figura 20: Interface HDLC - Aquisição com *strobe*

Nesta configuração um CH2 mestre tem a porta B programada como interface IOM, faz o tratamento do canal D e tem um CH2 escravo ligado à sua porta A. O CH2 escravo tem a porta B programada como interface HDLC. Neste caso, a informação a ser tratada pelo CI escravo pode estar circulando tanto pelo canal B1 quanto pelo canal B2 da interface da porta A do CH2 mestre. A entrada SQD do CH2 escravo define qual canal carrega a informação, segundo a programação dos bits HMD2-0 (modos 2 e 3) e a entrada REL recebe o sinal RSA de saída do chip mestre. Assim, nesta configuração, a taxa de bit é de 128 Kbit/s e a taxa de comunicação de dados (carregada em um dos canais B) é de 64 Kbit/s.

**Sinais da Interface** São em número de quatro os sinais que compõem esta interface: aqueles para transmissão e recepção de quadros (pinos TDSB e RDSB), um sinal de relógio (pino REL) e um sinal de sincronismo de quadro (pino SQD).

**Interface SLD** A SLD (Subscriber Line Datalink) é uma interface que possui três fios para transferência síncrona de dados entre dispositivos mestre e escravo, suportando um enlace serial tipo ping-pong com quatro canais de 64Kbit/seg multiplexados no tempo.

A interface SLD foi desenvolvida visando aplicações em telecomunicações, quando a norma é transferência síncrona de dados a 8 KHz. Ela provê uma interface física padronizada para a transferência de canais de dados e voz comutados por circuito e canais de controle e sinalização, permitindo que uma grande gama de diferentes dispositivos de assinantes sejam produzidos economicamente (com interface padrão). Com a sua adoção nem todos os CIs necessitarão de uma interface direta (paralela) com o processador, diminuindo o barramento nas placas de circuito impresso e permitindo que a troca de comandos e controle entre os chips corram pelos octetos FC e SIG do quadro.

**Estrutura de Quadro** A estrutura de quadro é semelhante, sob muitos aspectos, àquela da interface IOM. Os dois primeiros octetos carregam os canais B, o terceiro octeto é um canal FC (Feature Control) e o último octeto é denominado SIG (Signaling). A figura 21 apresenta a estrutura do quadro SLD.

Assim como na interface IOM, os octetos B1 e B2 são utilizados para transporte de informações de dados/voz a uma taxa de 64 Kbit/seg. Os octetos FC e SIG possibilitam o transporte de informações de controle e sinalização entre componentes que necessitam se comunicar com o processador ou entre si. Atualmente os componentes existentes que possuem interface SLD são os conjuntos *codecs + filtros*, com aplicações nos terminais e os chips utilizados para repassar canais de voz e dados para a matriz de comutação na central (controladores de placas de linha). O uso a ser dado aos canais FC e SIG é dependente

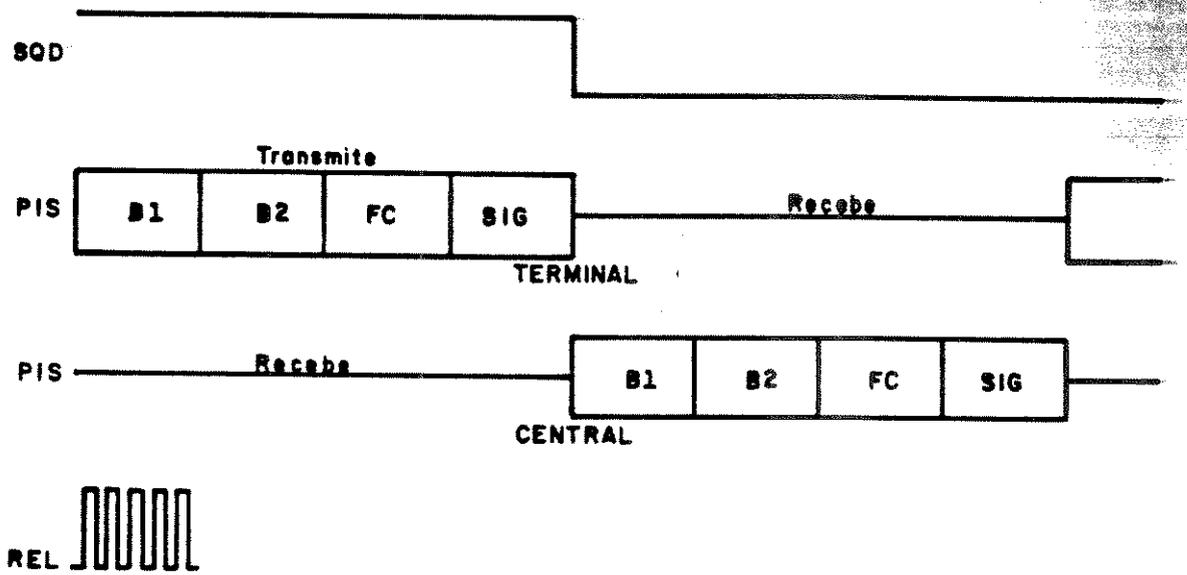


Figura 21: Estrutura de Quadro da Interface SLD

de cada fabricante, não sendo padronizado qualquer procedimento a nível de especificação de interface.

**Sinais da Interface** Os três fios da interface SLD consistem de um sinal de relógio usado como taxa de bits (REL), um sinal de direção de dados (SQD) e uma via de dados (PIS). A transmissão na interface SLD pode ocorrer quando o sinal SQD está em nível lógico 1 ou quando em nível lógico 0; no CH2 isso pode ser programado.

**Configurações de Timing** Dependendo do grupo funcional em que está operando o CH2, terminal ou central de comutação, deve ser programada uma configuração específica de timing.

O timing de terminal é mostrado na figura 22 e serve para repassar canais B a chips controladores de voz/dados com interface padrão SLD. O timing de central é mostrado na figura 23. Este modo é usado para conectar o CH2 a chips controladores de placas de assinantes via interface padrão SLD. Dados (somente canais B1 e B2) e sinal de relógio são supridos pelo CH2, que transmite durante a fase negativa do sinal SQD. A interface IOM é sincronizada com o sinal SQA, gerado pelo CH2 a partir de atraso no sinal SQD suprido pela rede.

**Interface SSI - Serial Synchronous Interface** A porta serial SSI (porta A) serve como uma conexão full-duplex às fontes/destinos de canais B em equipamentos terminais com uma taxa de dados de 128 Kbit/seg.

**Estrutura de Quadro** A estrutura de quadro da interface, mostrada na figura 24, transporta apenas os canais de voz ou dados a 64 Kbit/seg, não possuindo inteligência como nas interfaces expostas anteriormente. Nesta interface o sinal de sincronismo SQD tem a função de separar os canais B, isto é, pode ser utilizado pelo hardware externo ao chip para fornecer as habilitações distintas para os componentes que estão recebendo os canais.

**Sinais da Interface** A interface SSI consiste de uma linha de dados para cada direção (TDSA e RDSA), o sinal de sincronização de quadro a 8 KHz (SQD) e o sinal de relógio a 128 KHz (RSA), que é uma saída do CH2.

### 3.2.2 Chaveamento Interno dos Canais B

Através de programação no registrador RCPS1 é possível direcionar os canais B que chegam/saem das várias interfaces, B1 e B2 independentemente, para as demais interfaces do CH2. Isso é feito de acordo com a utilização desejada pelo chip, tais como loop de teste da linha, loop de teste de equipamento na central, geração de voz no terminal, passagem de voz para codecs, passagem de dados

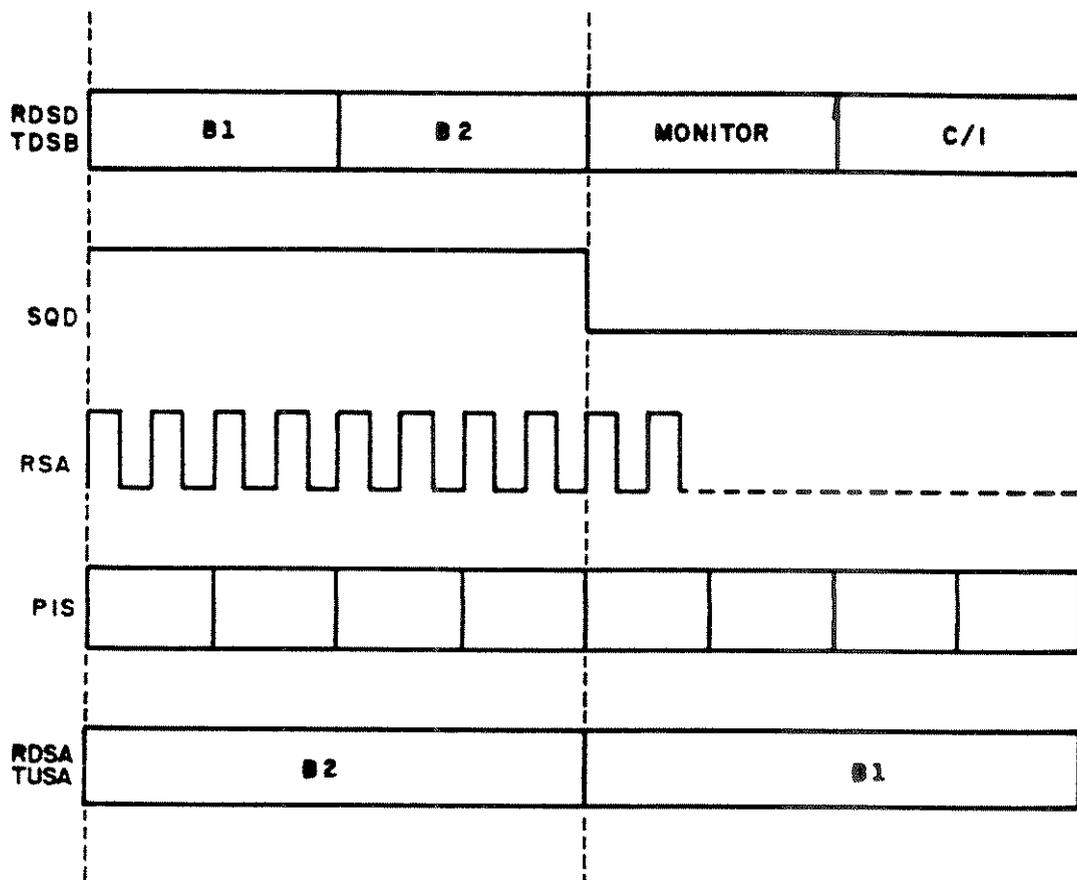


Figura 22: Configuração de Timing de Terminal

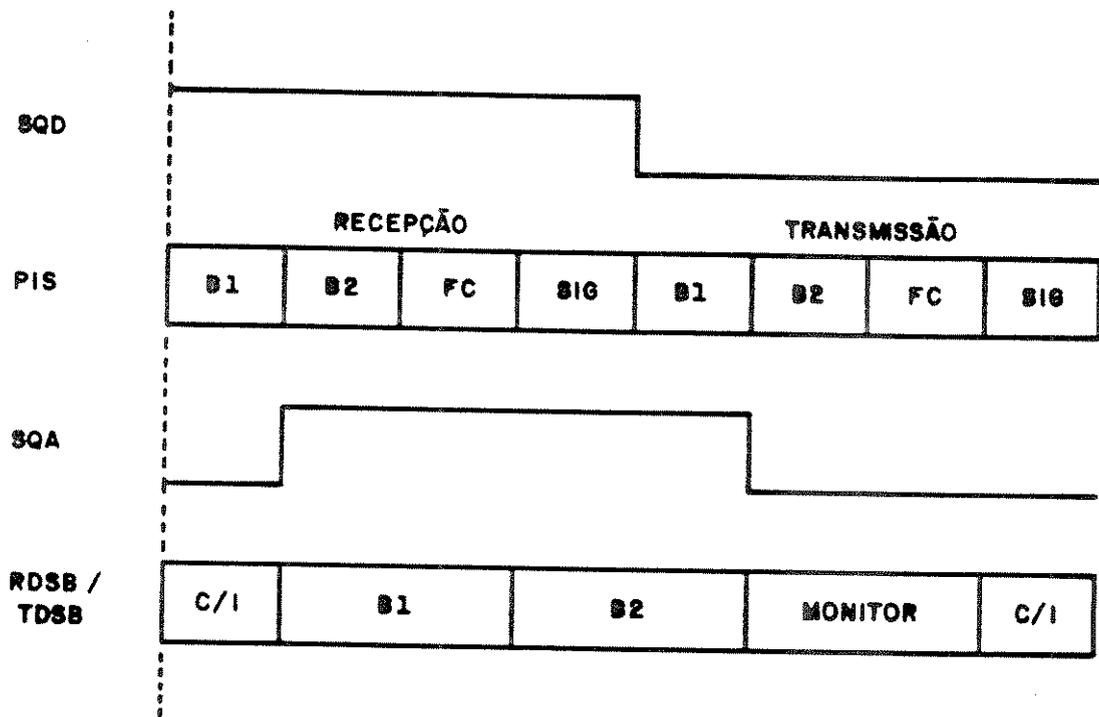


Figura 23: Configuração de Timing de Central

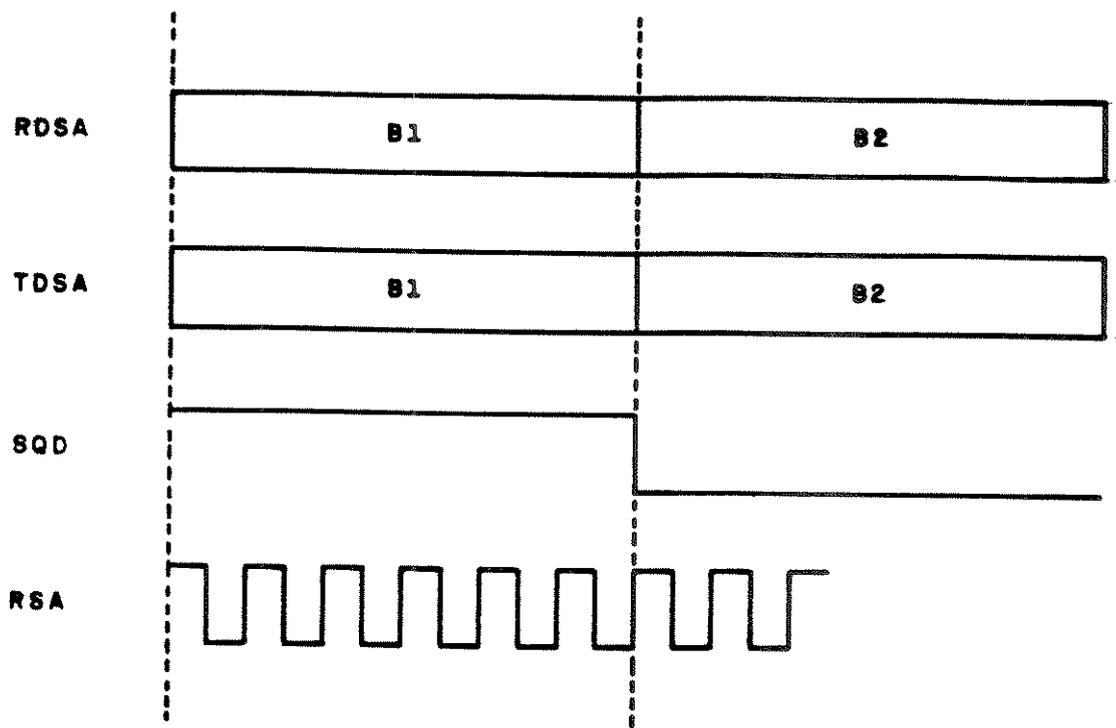


Figura 24: Estrutura de Quadro da Interface SSI

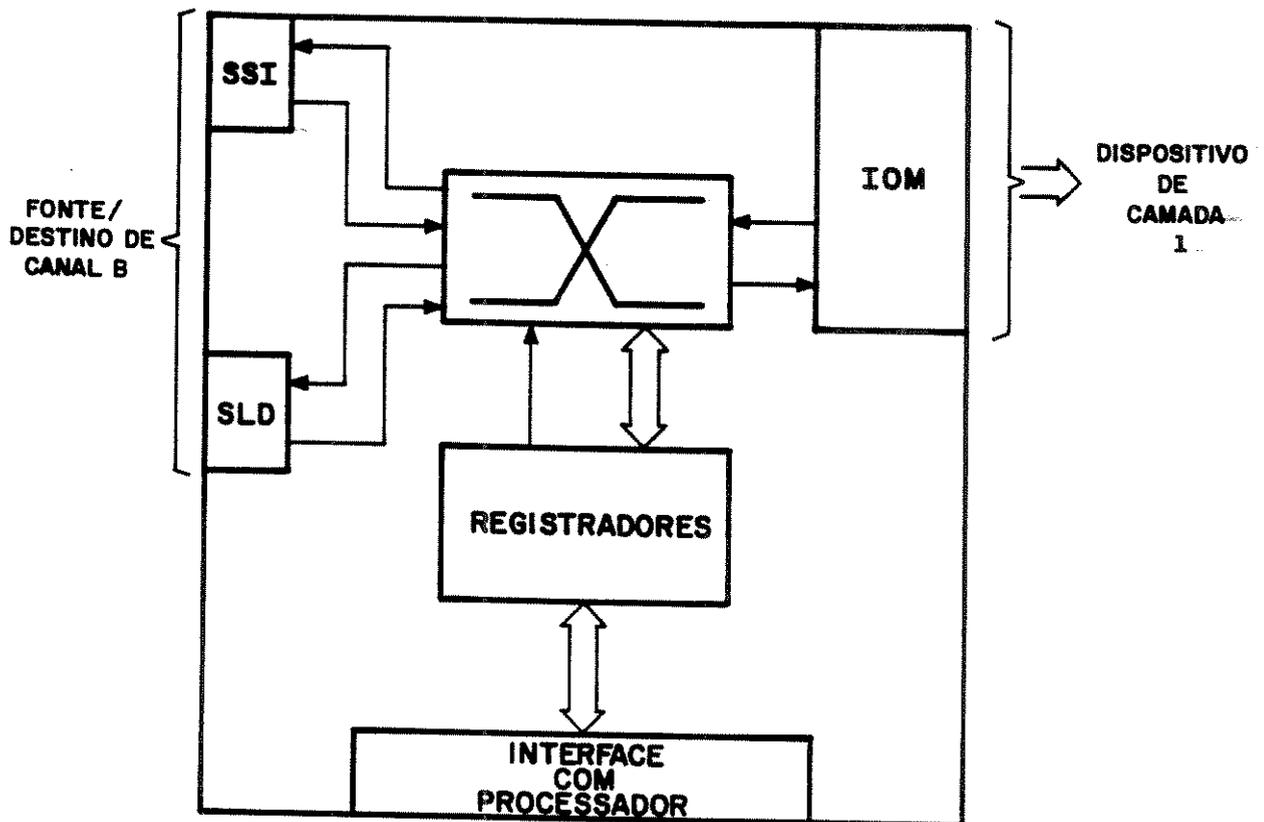


Figura 25: Princípios de Chaveamento de Canais B

para terminais e outros. A figura 25 mostra os princípios de chaveamento de canais B.

**Configuração 1 - Loop na Interface SLD** Nesta configuração, cada canal B, independentemente, pode ser chaveado diretamente dos registradores internos do chip (BCX1/BCX2) para a saída SIP, isto é, o processador define o conteúdo dos canais B que saem pela porta C ou, então, o processador apenas monitora os dados de canal B que neste caso estão espelhados na porta citada (figura 26).

**Configuração 2 - Chaveamento da Interface SLD** Nesta configuração o chip permite que os canais B que entram/saem da porta B sejam chaveados para a saída SIP onde deverão ser tratados por algum dispositivo com interface compatível com a SLD (figura 27).

Os canais B poderão ser monitorados pelo processador enquanto são repassados para a porta C ou vice-versa, de maneira que nos registradores BCX1/BCX2 estarão os dados que são recebidos pela porta B e nos registradores BCR1/BCR2

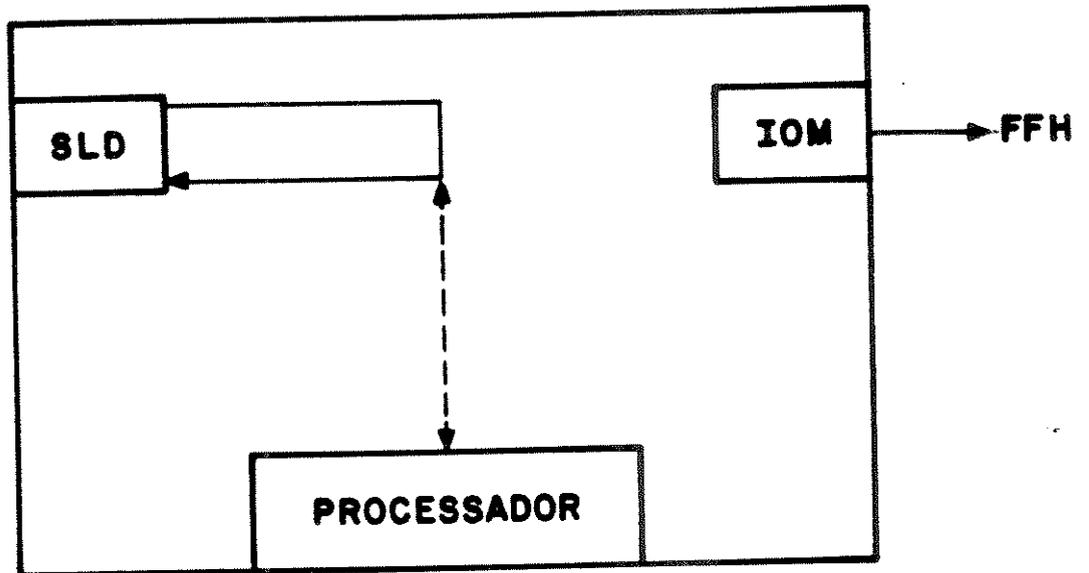


Figura 26: Loop na Interface SLD

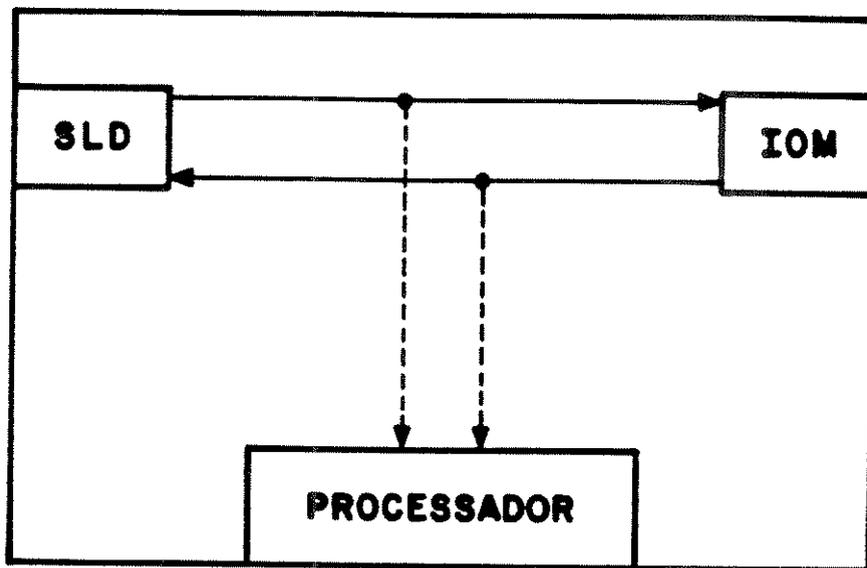


Figura 27: Chaveamento da Interface SLD

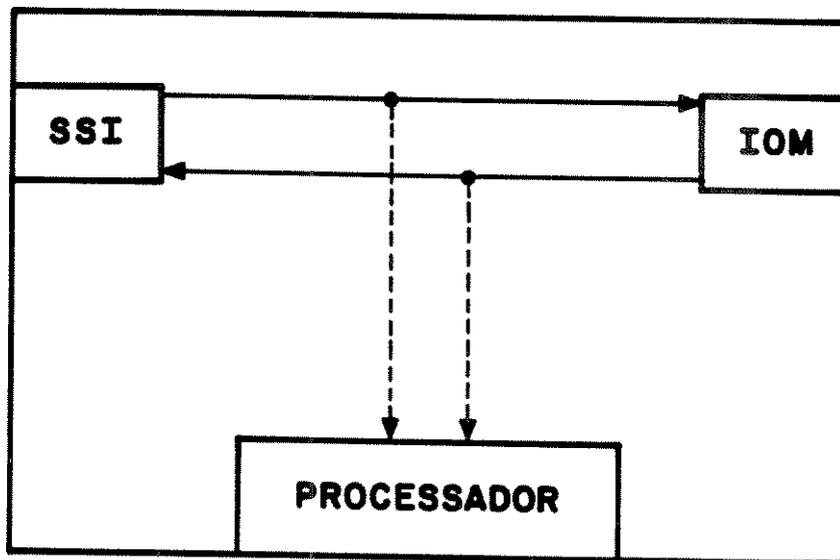


Figura 28: Chaveamento da Interface SSI

estão os valores recebidos pela porta C. A validade dos dados recebidos na porta C e armazenados nos registradores BCR1/BCR2 deve ser confirmada através do bit BVS do registrador RRCI antes de proceder à leitura.

**Configuração 3 - Chaveamento da Interface SSI** Esta configuração faz com que os canais B vindos da porta B sejam repassados para a porta A (e vice-versa), podendo o processador monitorá-los através da leitura dos registradores de dados (figura 28). Os dados recebidos na porta B podem ser lidos nos registradores BCX1/BCX2 e os dados recebidos na porta A podem ser lidos nos registradores BCR1/BCR2.

**Configuração 4 - Loop da Porta B** Nesta configuração os canais B podem ser espelhados na porta B e monitorados pelo processador, ou então se permite que seja feita uma troca de dados com a porta B para se realizar loops de teste (figura 29). Tal configuração tem aplicações relacionadas com loops de teste entre central e terminal, bem como loops de teste no terminal.

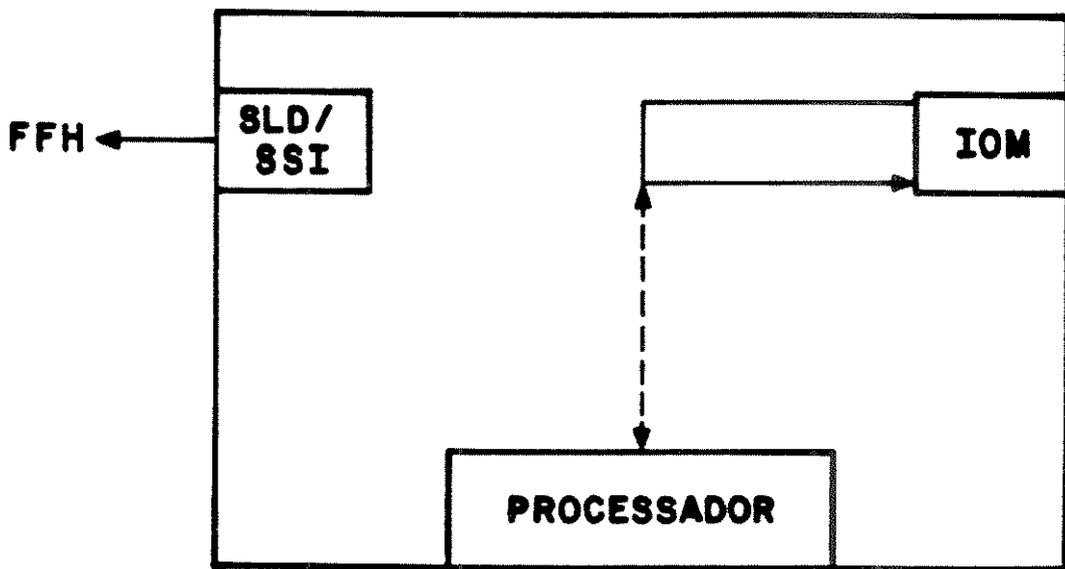


Figura 29: Loop da Porta B

### **3.2.3 Acesso aos Canais B pelo Processador**

O processador acessa os canais B1 e B2 pela leitura do Registrador de Recepção de canais B (BCR1/BCR2) ou pela leitura/ escrita do Registrador de Transmissão de canais B (BCX1/BCX2). A sincronização dos acessos é feita por programação do Registrador de Controle das Portas Seriais 2 (RCPS2).

**Loop na SLD, com monitoração dos canais B** No registrador RCPS2 programar transferência síncrona 1 quando na configuração de timing de central e transferência síncrona 0 quando na configuração de timing de terminal. Os dados espelhados na porta C podem ser lidos de BCX1 ou BCX2.

**Loop na SLD, com controle sobre os canais B** No registrador RCPS2 programar transferência síncrona 1 quando na configuração de timing de central e transferência síncrona 0 quando na configuração de timing de terminal. Após a ocorrência da interrupção, a validade dos dados tem que ser verificada no registrador RRCI (bit BVS). Subsequentemente o dado recebido na porta C pode ser lido de BCX1/BCX2, e o dado a ser transmitido pode ser escrito, em seguida, em BCX1/BCX2, após o qual a interrupção síncrona tem que ser reconhecida, de forma a evitar a geração de overflow.

**Monitoração dos Canais B com Chaveamento da Porta C** Programar transferência síncrona 0 no registrador RCPS2. O dado recebido da porta B pode ser lido de BCX1/BCX2 e os dados recebidos da porta C de BCR1/BCR2. Antes da leitura de BCR1/BCR2, a validade do dado tem que ser confirmada (RRCI, bit BVS).

**Monitoração dos Canais B com Chaveamento da Porta A** Programar transferência síncrona 0 no registrador RCPS2. Os dados recebidos na porta B podem ser lidos de BCX1/BCX2 e os dados recebidos na porta A de BCR1/BCR2.

**Loop da Porta B, com Monitoração dos Canais B** Programar transferência síncrona 0 no registrador RCPS2. Os dados recebidos na porta B podem ser lidos de BCX1/BCX2.

**Loop da Porta B, com Controle sobre os Canais B** Programar transferência síncrona 0 no registrador RCPS2. Os dados recebidos na porta B podem ser lidos de BCX1/BCX2 e os dados a serem transmitidos podem ser subsequentemente colocados em BCX1/BCX2.

**Transmissão no Canal Feature Control** Programar transferência síncrona 0 no registrador RCPS2. Subsequentemente a cada interrupção síncrona (SIN - reg. REIN), um octeto deve ser escrito em RFCS. No caso de apenas um octeto tiver que ser transmitido, não é necessário programar transferência síncrona; o CH2 vai transmitir esse octeto apenas uma vez, e em seguida FFH é transmitido automaticamente.

**Recepção no Canal Feature Control** Programar transferência síncrona 1 no registrador RCPS2. Subsequentemente a cada interrupção síncrona um octeto é lido de RFCS.

**Transmissão no Canal de Sinalização** Todo octeto escrito pelo processador no registrador XSGS é transmitido continuamente no seu *time-slot* até que um novo octeto seja escrito.

**Recepção no Canal de Sinalização** Pode operar de duas formas, detecção de transição ou não, dependendo do bit DTS do registrador RXCI. Se DTS estiver ativado, a cada mudança do octeto recebido no canal SIG é gerada uma interrupção TCS e o conteúdo do canal pode ser lido no registrador RSGS. Caso contrário, o processador terá que fazer o acesso ao octeto através de varredura (pooling) no registrador.

#### 3.2.4 Lógica de Interrupções

Todos os blocos do CH2 podem gerar causas de interrupções em algumas situações específicas, conforme é descrito ao longo deste capítulo. No entanto, é o Bloco de Interfaces o responsável pela geração do sinal de interrupção ao processador, tratamento das máscaras e desativação dos pedidos quando já atendidos.

Quando é apresentada uma causa de interrupção (ou várias simultaneamente) por algum bloco constituinte do chip, é ativado o bit correspondente nos registradores de estado de interrupções, que são em número de três. Em função disso será ativado, em nível lógico 1, o sinal INT de pedido de interrupção ao processador. Este, ao detetar tal solicitação, deverá inicialmente ler o registrador REIN. Caso a(s) causa(s) da interrupção esteja(m) em um dos registradores de extensão (REX1 e REX2), isso é notificado pela ativação dos bits EI1 e/ou EI2 em REIN, e o processador deverá ler também esses registradores.

Ao atender uma solicitação de interrupção o processador poderá verificar que as causas são múltiplas. Na verdade interrupções separadas foram geradas a partir de cada causa, mas podem estar aparecendo simultaneamente nos registradores de estado, e o processador deverá atender a cada uma delas adequadamente.

Os bits EI1 e EI2 são desativados logo após a leitura dos registradores REX1 e REX2 respectivamente e todos os outros bits são desativados após a leitura do

registrador REIN. Desta forma, o processador deve se preocupar no sentido de guardar as informações obtidas numa leitura, já que esta operação irá apagá-las.

Cada interrupção poderá ser seletivamente mascarada através da ativação do bit respectivo no registrador MASC. No caso de interrupções dos tipos contidos em REX1 e REX2, nenhuma interrupção é gerada com um bit EI1 ou EI2 mascarado. No entanto, este bit é ativado em REIN.

### 3.3 Descrição do Bloco Tratador de HDLC

As funções a nível de bit do protocolo LAPD, isto é, sequência de flags, transparência de dados, sequência de verificação de CRC, aborto de quadro e outros são iguais (ou similares) às funções de nível de bit de outros protocolos de nível de enlace orientados a bit <sup>1</sup>, que por sua vez foram baseados no protocolo da ISO chamado HDLC (High Level Data Link Control). O que chamaremos de Bloco Tratador de HDLC neste chip é um conjunto de circuitos que, além de executar todas as funções a nível de bit existentes no protocolo LAPD, é um auxiliar do Bloco Tratador de LAPD (TLD), resolvendo para este toda a parte operativa. Em outras palavras, o protocolo LAPD é resolvido autonomamente no chip por duas entidades: o THDLC, responsável pela parte operativa, e o TLD, responsável pela parte de controle.

Esta seção apresenta a descrição das funções do THDLC; a seção 3.8 traz algumas sugestões para a implementação.

#### 3.3.1 Ativação dos Modos de Operação do Controlador HDLC

O CH2 dá suporte ao tratamento do protocolo LAPD com diferentes níveis de capacidade, dependendo do modo selecionado para transferência de mensagens (figura 30). Esses modos são ativados através de programação no Registrador de Modo. São eles:

- Modo Automático
- Modo Não-Automático
- Modo Transparente
- Modo Transparente Estendido

**Modo Automático** O CH2 processa autonomamente todos os quadros de um enlace lógico, de acordo com a recomendação I.441 do CCITT e o descrito na seção 3.4. Neste modo, o CH2 trata o protocolo com janela variando de 1 até 128 (modo estendido).

O chip analisa o campo de endereço, reconhecendo um endereço válido, analisa o campo de controle, verificando a sequência correta dos quadros e responde ao lado que transmitiu a trama com a resposta apropriada.

Durante os *procedimentos de comunicação*, não é necessário um diálogo entre o CH2 e o processador; o chip irá informar a este, sempre que necessário, o estado do procedimento. O software de nível 2 remanescente no processador do sistema é usado para inicialização, recuperação de erros e troca de informações de nível 3. Para este último caso, os dados do campo de informação de quadro

---

<sup>1</sup>Entre eles encontramos o protocolo X.25 para comunicação de dados via pacotes em redes públicas e o protocolo LLC (IEEE 802.2) para redes locais.

QUADRO LAPD

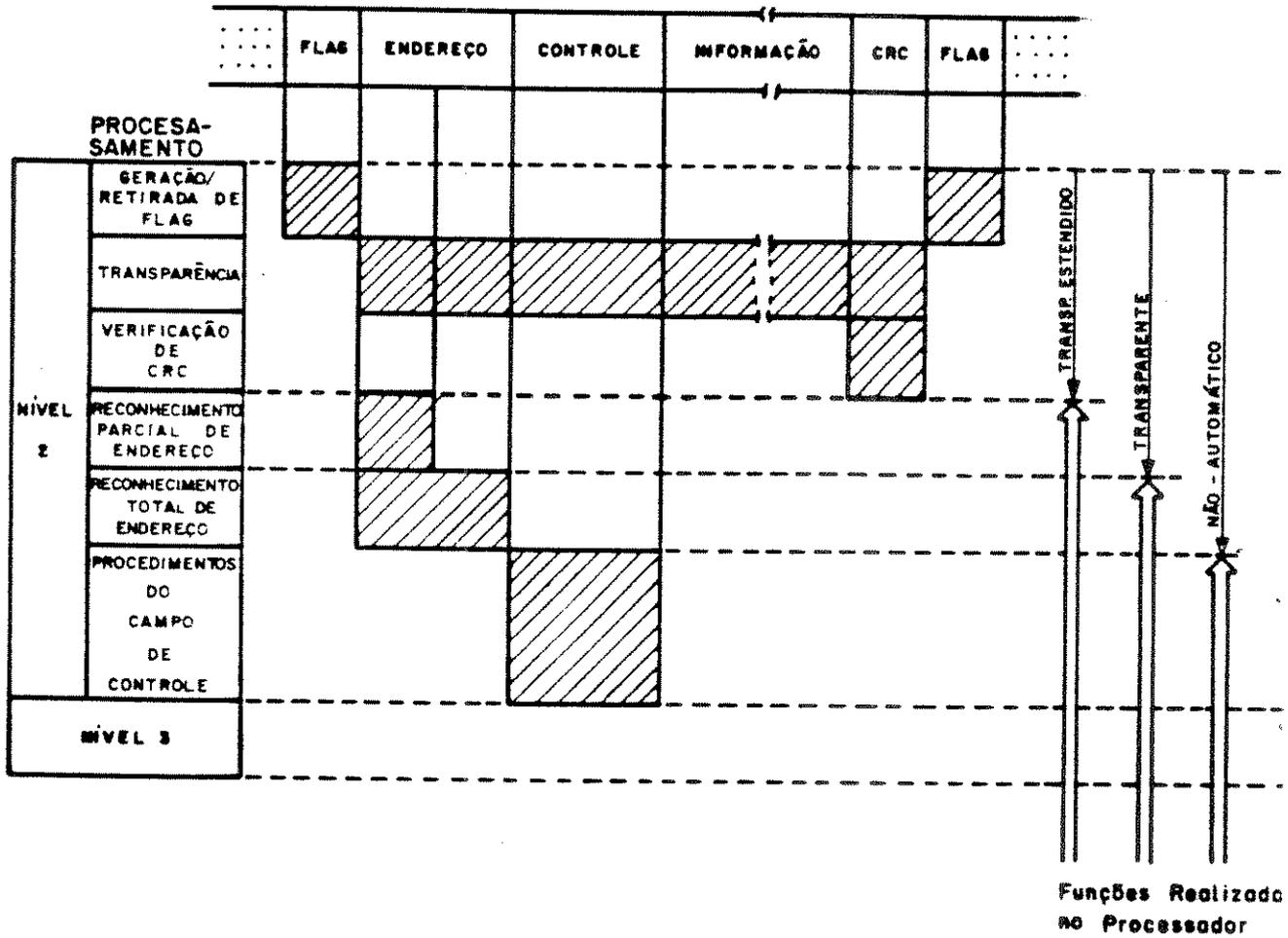


Figura 30: Modos de Operação do HDLC

HDLC recebido devem ser temporariamente armazenados em uma memória FIFO de recepção (R-FIFO), enquanto que dados a serem enviados devem ser carregados em uma memória FIFO de transmissão (X-FIFO). Ainda que tratado automaticamente pelo CH2 neste modo, o campo de controle pode ser lido dos Registradores RHCR1 e RHCR2.

Quando programado no modo automático, o chip fica inicialmente numa fase não-operacional, onde o comportamento deste modo é idêntico ao comportamento do modo transparente estendido. Esta fase permite que o processador troque informações com a entidade par, de modo a definir um valor para o campo TEI (Terminal Equipment Identifier). Uma vez definido este valor, através da ativação do bit *Entra no Estado TEI Designado* no Registrador de Comando 2, o chip passa à chamada fase operacional.

**Modo Não-Automático** Neste modo, os procedimentos a serem tomados em função do conteúdo no campo de controle do quadro, não são executados pelo CH2; o campo de controle fica disponível ao processador para leitura nos registradores RHCR1 e RHCR2 para que este faça a análise do conteúdo e a geração de resposta adequada.

Todo o quadro recebido, entre o flag inicial e o CRC, é guardado na R-FIFO, mas existe o tratamento interno do campo de endereço. Na transmissão de quadros, o CH2 irá apenas colocar os flags e os octetos de CRC envelopando o conteúdo da X-FIFO, que deverá conter os campos de endereço, controle e informação.

**Modo Transparente** Similar ao modo automático; a diferença está no tratamento do campo de endereço, onde somente o primeiro octeto é analisado.

**Modo Transparente Estendido** Como no modo transparente, sem qualquer análise do campo de endereço.

### 3.3.2 Funções de HDLC

O Bloco THDLC é responsável pelas tarefas de montagem e desmontagem de quadros, integridade dos dados, análise de erros de protocolo e gerenciamento das variáveis do protocolo LAPD (quando no modo automático).

Desta forma, alguns circuitos podem ser visualizados como parte deste bloco. Aqui eles serão apresentados divididos (com certa margem de tolerância) de acordo com a função a que se relacionam.

**Circuitos Relacionados com Montagem e Desmontagem de Quadros**  
São os seguintes os circuitos que assim se classificam:

- Circuito de Inserção/Retirada de Flags

- Circuito de Transparência de Dados
- Circuito Montador e Transmissor de Quadros no Modo Automático
- Circuito Tratador de Quadros Recebidos
- Circuito Tratador de Comandos Provenientes do Processador para Transmissão de Quadros

**Circuito de Inserção/Retirada de Flags** Todos os quadros devem iniciar e terminar com uma sequência de flags consistindo de um bit "0" seguido por seis bits "1" contínuos e um bit "0". Em alguns casos o flag de fechamento serve também como flag de abertura do próximo quadro.

Na recepção de quadros este circuito deteta e retira os octetos de flags e os coloca antes da transmissão.

**Circuito de Transparência de Dados** Na transmissão este circuito deve inserir um bit "0" depois de toda sequência de cinco bits "1" consecutivos (incluindo os cinco últimos bits do campo CRC) e na recepção deve retirar um bit "0" que segue uma sequência de cinco bits "1" consecutivos, de forma a garantir que um flag (7EH) ou uma sequência de aborto não seja simulada dentro de um quadro.

**Circuito Montador e Transmissor de Quadros no Modo Automático** Na fase operacional do modo automático, a decisão de quando um quadro não-numerado (exceto quadros UI e XID) ou de supervisão deve ser enviado é uma função do Bloco Tratador de LAPD. Uma vez tomada a decisão, o TLD avisa o circuito Montador de Quadros do THDLC, através de um código, para que este monte o quadro e o passe para o circuito transmissor.

O transmissor HDLC tem uma fila de espera que guarda os comandos de transmissão de quadros provenientes do TLD e ainda não executados, possibilitando a este último o tratamento adequado dos quadros recebidos, independente de o transmissor estar ocupado ou não. Ela possibilita ainda que ao TLD sejam transparentes os casos de colisão ou de envio de quadros pedido pelo processador diretamente ao THDLC (XQI, XQUI, XQT).

**Circuito Tratador de Quadros Recebidos** Circuitos que executam uma sequência de operações durante a recepção de quadros. Estas operações estão mostradas nos Leds do anexo II.

A solução do problema de recepção depende do modo como o chip está programado. Na fase não-operacional do modo automático e nos outros modos, ou ainda no caso de recebimento de quadro UI quando na fase operacional do modo automático, o TLD não está em funcionamento e, assim, ao THDLC basta passar o quadro para o processador via a R-FIFO.

Na fase operacional do modo automático, temos três casos distintos: recebimento de quadro UI, cujo tratamento foi mostrado acima, recebimento de quadros de supervisão ou não-numerados, cujo tratamento é feito pelo Circuito Analisador do Campo de Controle no Modo Automático e, finalmente, o recebimento de quadro de informação (I). O tratamento deste último caso é dividido entre o THDLC e o TLD. Se o processador local estiver ocupado, o THDLC ignora o campo de informação e avisa ao TLD da chegada do quadro e sua situação, isto é, não recebido por problemas do receptor. Caso o processador local esteja livre, o TLD é avisado do recebimento de quadro I, mesmo que haja problema de numeração (neste caso o campo de informação também será ignorado). O TLD não será avisado, no entanto, caso tenha ocorrido algum erro na recepção.

**Circuito Tratador de Comandos Provenientes do Processador para Transmissão de Quadros** Circuito que executa uma sequência de operações após o recebimento de comandos de transmissão de quadros, conforme mostrado nos LEDES do anexo II.

O tratamento do comando XQT, independente do modo de operação do chip, é tal que tudo o que estiver na X-FIFO é colocado entre os campos de flag de abertura e o CRC. O comando XQUI no modo automático faz com que um quadro UI seja transmitido, onde o campo de informação é retirado da X-FIFO.

No início da transmissão de quadro do tipo I alguns testes devem ser executados. Para evitar a repetição de tais testes no mesmo quadro, quando este for maior que 32 octetos, é criada a variável *Quadro I Iniciado*. O TLD só é avisado da transmissão de quadro I (QIT - Quadro I Transmitido) se não houve problemas durante esta fase.

**Circuitos Relacionados com Integridade de Dados** São os seguintes os circuitos que executam tal função:

- Circuito de Geração e Verificação de CRC
- Circuito Gerador e Testador de Sequência de Aborto
- Circuito para Detecção de Transbordamento (Overflow)

**Circuito de Geração e Verificação de CRC** O CRC é um código para detecção de erros introduzidos pelo meio físico de transmissão. Este campo deve ser uma sequência de dezesseis bits calculada como o complemento de um da soma (módulo 2) de:

- o resto de  $(x \text{ elevado a potência } K) \cdot (x^{15} + x^{14} + x^{13} + x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x^1 + 1)$  dividido (módulo 2) pelo polinômio gerador  $x^{16} + x^{12} + x^5 + 1$ , onde  $K$  é o número de bits existente

no quadro entre, mas não incluindo, o bit final do flag de abertura e o primeiro bit do CRC, excluindo os bits inseridos para transparência, e

- o resto da divisão (módulo 2) pelo polinômio gerador  $x^{16} + x^{12} + x^5 + 1$ , do produto de  $x^{16}$  pelo conteúdo do quadro existente entre, mas não incluindo, o bit final do flag de abertura e o primeiro bit do CRC, excluindo os bits inseridos para transparência.

Na implementação do transmissor, o conteúdo inicial do registrador do dispositivo que computa o resto da divisão é colocado com todos os bits em "1" e então é modificado pela divisão pelo polinômio gerador (como descrito acima) nos campos de endereço, controle e informação; o complemento de um do resto resultante é transmitido como uma sequência CRC de dezesseis bits.

Na implementação do receptor, o conteúdo inicial do registrador do dispositivo que irá computar o resto é colocado com todos os bits em "1". O resto final depois da multiplicação por  $x^{16}$  e então divisão (módulo 2) pelo polinômio gerador  $x^{16} + x^{12} + x^5 + 1$  da sequência serial de bits recebidos, incluindo o campo CRC, será "0001 1101 0000 1111" ( $x^{15}$  até  $x^0$  respectivamente) na ausência de erros de transmissão.

Se ocorreu erro detectado na recepção do campo CRC, o bit CRC do Registrador de Estado de Recepção RERC é ativado e gerada uma interrupção MRC.

**Circuito Gerador e Testador de Sequência de Aborto** A recepção de sete ou mais bits "1" consecutivos deve ser interpretada como um aborto de quadro e o nível de enlace deve ignorar o quadro que está sendo recebido.

São duas as situações em que uma sequência de aborto é gerada e transmitida para avisar o receptor de que o quadro é inválido:

1. a partir de um comando XRES (Reset do Transmissor) do processador;
2. no caso de não existência de dados disponíveis na X-FIFO, após um comando de transmissão de quadro do processador e sem que o bit de mensagem completa (FMX) esteja ativado. Com isso, o caracter de aborto protege o receptor dos erros causados pelo próprio transmissor.

Durante a recepção, a detecção de uma sequência de aborto é avisada ao processador através da ativação do bit MRA do Registrador de Estado da Recepção (RERC) e de uma interrupção MRC.

**Circuito para Detecção de Transbordamento (Overflow)** Durante a recepção de quadros, duas situações diferentes de transbordamento podem ocorrer:

1. o processador não leu os octetos da R-FIFO, liberando-a para receber os dados que completam a mensagem que está sendo recebida. Neste caso, o bit TDR (Transbordamento de Dados Recebidos) é ativado e uma interrupção TRC é gerada.

2. um novo quadro recebido não pode ser armazenado porque a R-FIFO estava cheia. Neste caso, uma interrupção TRC é gerada. O fluxograma da figura 32 mostra tais interrupções.

**Circuitos Relacionados com Análise de Erros de Protocolo** Os erros de protocolo são tratados parte pelo THDLC e parte pelo TLD. O THDLC detecta erros que irão resultar em respostas do tipo FRMR. Mesmo na fase operacional do modo automático o CH2 não gera quadros FRMR (apesar de recebê-los e interpretá-los); eles deverão ser montados e enviados via comando XQT pelo processador, a partir do aviso de existência de tais erros, que é feito por interrupções codificadas nos bits ER4-ER1.

São três os circuitos que executam tal função:

- Circuito Analisador do Campo de Controle no Modo Automático
- Circuito Detetor do Campo de Informação onde não Permitido
- Circuito Detetor de Quadro com Comprimento Incorreto

**Circuito Analisador do Cpo de Controle no Modo Automático** No modo automático o THDLC analisa o campo de controle do quadro recebido e verifica se é um código correto. Se for, o TLD recebe a informação do tipo de quadro recebido, desde que não tenha ocorrido outro tipo de erro durante a recepção, como por exemplo aborto do quadro, erro de CRC, etc. No caso de ocorrência de algum erro, ou no caso de recebimento de algum quadro válido mas não tratado internamente pelo CH2 (como, por exemplo, quadro XID), é gerada uma interrupção do tipo *Indicação de Erro L*. Neste caso o processador deve proceder à análise, buscando o campo de controle nos Registradores RHCR1 e RHCR2.

**Circuito Detetor de Campo de Informação onde não Permitido** No modo automático, se for recebido algum quadro dos tipos U ou S contendo campo de informação, ele será descartado e uma interrupção do tipo *Indicação de Erro M* é gerada.

**Circuito Detetor de Quadro com Comprimento Incorreto** O bit MEM do Registrador de Modo determina o tamanho do campo de endereço do quadro. Se MEM=0, o menor quadro possível é de 4 octetos (2 octetos para CRC, 1 octeto para endereço e 1 octeto para controle) e se MEM=1, o menor quadro possível é de 5 octetos. Caso o quadro recebido apresente um número de octetos menor que o mínimo, o THDLC descarta o mesmo e gera interrupção do tipo *Indicação de Erro N*.

Não existe limitação, para o CH2, do tamanho máximo do quadro.

**Circuitos Relacionados com Gerência de Variáveis do LAPD** O tratamento do protocolo LAPD foi dividido no CH2 de tal forma que a parte de controle ficou com o TLD enquanto que a parte operativa ficou com o THDLC. Dessa forma, operações aritméticas e lógicas sobre as variáveis do protocolo serão executadas pelo THDLC, sob comando do TLD.

As variáveis aqui tratadas são: V(A), V(S), V(R), N(R), N(S), RC e as operações são de comparação lógica (igualdades e desigualdades), incremento de variáveis e teste de valores, encontradas nos LEDS do anexo II.

### 3.3.3 Transferência de Pacotes de Dados

A recepção pelo CH2 e posterior passagem pelo processador de pacotes de dados, assim como a passagem de dados do processador para ser transmitido pelo CH2, é feita através de interrupções (sem uso de Controlador externo de DMA) e utilizando-se quatro memórias FIFOs internas de 32 octetos cada, duas para transmissão e duas outras para recepção, que servirão como armazenadores temporários de dados em ambos os casos. Tem-se, assim, uma solução intermediária entre a lentidão da interrupção a cada octeto recebido e a eficiência de um canal de DMA.

O conteúdo da FIFO é dependente do modo em que o chip está programado. Na fase operacional do modo automático, elas armazenam apenas o campo de informação dos quadros I e UI, ou, no caso da X-FIFO, todo um quadro transparente. Nos outros modos elas conterão, além dos octetos do campo de informação, os campos de endereço e controle, conforme visto na seção *Ativação dos Modos de Operação do Controlador de HDLC*.

**Procedimentos na Recepção de Quadros** As mensagens recebidas são armazenadas na R-FIFO de acordo com os mecanismos descritos na seção 3.5, no item *Controle da R-FIFO*. A sinalização de que a R-FIFO contém dados recebidos é feita pela ativação do bit DAR do Registrador de Estado dos Controladores. Os procedimentos a serem seguidos pelo processador estão descritos na seção 3.5, no item *Leitura da R-FIFO*.

As operações executadas pelo THDLC no caso de recepção de quadro I e UI (modo automático) estão mostradas nos LEDS do anexo II.

**Procedimentos na Transmissão de Quadros** Quando deseja fazer a transferência de pacotes de dados, o processador poderá fazer uso de três comandos distintos, dois dos quais (XQI e XQUI) valem somente para a fase operacional do modo automático. Os comandos são:

- *XQT* - Transmite Quadro Transparente. Ao receber esse comando o CH2 irá buscar na X-FIFO todos os campos do quadro, à exceção dos flags e CRC.

- *XQI* - Transmite Quadro de Informação Numerado. O campo de endereço será buscado nos Registradores XAD1 e XAD2, o campo de controle é montado pelo THDLC e o campo de informação é retirado da X-FIFO.
- *XQUI* - Transmite Quadro de Informação Não-Numerado. A montagem é a mesma do caso acima, porém o campo de controle não contém numeração do quadro.

O procedimento a ser executado pelo processador para a transmissão de um quadro é descrito a seguir.

O bit XFE (Permissão de Escrita na X-FIFO) do Registrador de Estado dos Controladores (REST) deve ser testado para saber se há X-FIFO disponível para carregamento.

Em caso positivo, se a mensagem tiver 32 octetos ou menos, o processador deverá carregar os dados em posições consecutivas da X-FIFO e gerar um dos comandos de transmissão com o bit FMT (Fim da Mensagem Transmitida) do Registrador de Comando 1 ativado.

Se a mensagem for maior que 32 octetos, o processador deve carregá-los em conjuntos de 32 e após cada carregamento deve gerar um dos comandos de transmissão com o bit FMT desativado. Um novo carregamento só pode ocorrer depois que uma interrupção BXP (Buffer de Transmissão Pronto) tenha sido gerada pelo CH2. Quando finalmente os últimos octetos da mensagem forem colocados na X-FIFO, o comando gerado pelo processador deve ter o bit FMT ativado para evitar que novas interrupções BXP venham a ocorrer e para que o THDLC feche o quadro que está sendo transmitido.

As operações executadas pelo THDLC ao receber os comandos de transmissão de quadro estão mostradas nos Leds do anexo II.

### 3.3.4 Tratamento do Campo de Endereço

O tratamento do campo de endereço, tanto na transmissão quanto na recepção de quadros é executado pelo THDLC. Esse tratamento depende do modo de operação do chip e do valor do bit MEM do Registrador de Modo. Este bit define se o campo de endereço do quadro HDLC tem um ou dois octetos. Ele é irrelevante, no entanto, para a fase operacional do modo automático, quando o quadro terá sempre dois octetos.

**Recepção de Quadros** Dois registradores programáveis pelo processador são usados para a análise do primeiro octeto do campo de endereço dos quadros recebidos: SAPI1 e SAPI2. Dois outros são usados para análise do segundo octeto: TEI1 e TEI2. Além deles, o chip compara os endereços com valores fixos de endereço de difusão, definidos no protocolo LAPD.

**Fase Operacional do Modo Automático** O CH2, ao receber um quadro, irá comparar o primeiro octeto do campo de endereço com os valores fixos FEH e FCH (SAPI de difusão), assim como com os valores individualmente programados nos Registradores SAPI1 e SAPI2 (neste caso, somente os 6 bits mais significativos). Em relação ao segundo octeto, a comparação é feita com o valor FFH (TEI de difusão) ou com os 7 bits mais significativos dos Registradores TEI1 e TEI2.

O CH2 irá considerar um endereço válido quando ambos os octetos correspondem a um dos valores programados; no entanto, o processamento interno do campo de controle será feito somente se ocorrer na combinação dos valores constantes nos Registradores SAPI1 e TEI1 nos quadros recebidos. Todos os outros quadros, desde que com endereço válido, serão transferidos diretamente ao processador.

O quadro recebido será interpretado como comando ou resposta de acordo com as regras apresentadas na descrição do Registrador SAPI1.

Os bits SA1, SA0, C/R, TE1 e TE0 do Registrador de Estado de Recepção são atualizados.

**Modo Não Automático** Neste caso, campos de endereço com um ou dois octetos são aceitos. O tratamento para campo de endereço com dois octetos é idêntico aquele da fase operacional do modo automático. Quando o campo de endereço tem apenas um octeto, a comparação será feita com os valores em TEI1 e TEI2. Os bits SA0, SA1, e C/R do Registrador de Estado de Recepção são irrelevantes e os bits TE0 e TE1 são atualizados.

**Modo Transparente** Quando o campo de endereço do quadro possui dois octetos, o primeiro é comparado com os valores fixos FEH e FCH, assim como com os valores individualmente programados nos Registradores SAPI1 e SAPI2 (6 bits mais significativos). São atualizados os bits SA0, SA1 e C/R do Registrador de Estado de Recepção e os bits TE0 e TE1 são irrelevantes.

**Modo Transparente Estendido e Fase Não Operacional do Modo Automático** Não há análise do campo de endereço.

**Transmissão de Quadros** Com exceção da fase operacional do modo automático, em todos os outros modos o campo de endereço deve vir na X-FIFO, junto com os campos de controle e informação.

Na fase operacional do modo automático, o CH2 irá montar o primeiro octeto do campo de endereço com os 6 bits mais significativos do Registrador XAD1 e o segundo octeto com os 7 bits mais significativos do Registrador XAD2. O bit 1 do primeiro octeto (C/R) será 1 ou 0, conforme seja um quadro de comando ou de resposta e o chip esteja do lado da central ou do lado do usuário.

### 3.3.5 Temporização

A lógica de temporização é responsável pela geração e controle de temporizações que existem durante a transmissão de quadros ou ainda para aplicações de propósitos gerais externas ao CH2. Em função do modo de operação do chip, definem-se dois tipos de temporizações: interna, para a fase operacional do modo automático e externa, para os demais.

No registrador programável RTEM são definidos dois campos:

- Campo N3-N1, com validade apenas no modo automático.
- Campo *Valor*, que serve de base para um circuito contador, cujo sinal de relógio é tal que se pode temporizar entre 125 ms e 4 s.

No modo automático, fase operacional, as temporizações ocorrem sem intervenção do processador. No item *Temporizações de LAPD* são analisadas tais temporizações.

Nos outros modos de operação o temporizador pode ser usado para propósitos gerais. A interrupção TEM ocorre após cada expiração do tempo correspondente aquele programado no campo *Valor*. Após a programação do Registrador RTEM, o comando ITE (Inicia Temporizador) deve ser gerado pelo processador para habilitar a contagem.

### 3.3.6 Condições Iniciais do THDLC após Reset Hardware

Após o reset hardware, o chip fica em um estado em que o receptor HDLC está desativado. A R-FIFO 0, assim como a X-FIFO 0, passam a ser as FIFOs correntes (isto é, aquelas que serão acessadas pelo THDLC), estão descarregadas, e os circuitos fornecedores de endereço apontam para a posição 0 de ambas. Durante esse período o CH2 deverá ter suas interfaces e seu modo de operação programados. Em seguida, o comando RAC deve ser gerado de forma a habilitar a operação normal do chip.

### 3.4 Descrição do Bloco Tratador de LAPD (TLD)

O TLD é o bloco do CH2 responsável pelo controle sobre os parâmetros do protocolo LAPD, controle sobre os serviços oferecidos à camada de nível superior, controle sobre os serviços oferecidos à camada de gerência e controle sobre os elementos de procedimento da comunicação de entidades pares no nível 2.

Esta seção descreve o funcionamento deste bloco.

#### 3.4.1 Funções e Procedimentos

Uma entidade de enlace de dados ponto a ponto pode estar em um dos três estados básicos:

1. *Estado de TEI Não Designado*

Neste estado nenhum valor de TEI foi designado e a transferência de informação de camada 3 não é possível.

2. *Estado de TEI Designado*

Neste estado um valor de TEI já foi acertado entre o equipamento terminal e a central através de um *procedimento de designação de TEI*. Este procedimento não é suportado autonomamente pelo CH2; ele deve ser executado pelo processador fazendo uso do comando de transmissão de quadros transparentes (XQT). Neste estado é possível a transferência de informação não reconhecida (operação onde a transferência de informação de camada 3 é executada através de quadros do tipo UI).

3. *Estado de Quadros Múltiplos Estabelecido*

Neste estado, atingido através de um *procedimento de estabelecimento de quadros múltiplos*, é possível a transferência de informação reconhecida (transferência de informação de camada 3 executada através de quadros do tipo I) e não reconhecida.

Estes estados básicos podem ser expandidos, conforme se pode observar nos LEDES do anexo I.

O TLD do CH2 só opera no modo automático. O modo automático poderá estar em uma de duas fases: operacional e não operacional. Na inicialização do CH2, quando programado no modo automático, o chip entra diretamente na fase não operacional, que corresponde ao estado do protocolo LAPD onde o valor do TEI não está designado. Durante esta fase as transferências de dados entre terminal e central far-se-ão através de comando de transferência de quadros transparentes (XQT) e nenhuma análise é feita nos campos de endereço e controle dos quadros recebidos. Após o término dos procedimentos de designação do TEI, o processador deve fornecer ao CH2 (através da programação dos Registradores TEI1 e TEI2) o valor de TEI acertado com a central e um comando TED (RCMD2) deve ser gerado, fazendo com que o TLD entre na

fase operacional. A partir daí o controle sobre os procedimentos é totalmente do TLD; a interação com o processador se dará apenas para que este solicite conexão/desconexão ou nas transmissões e retransmissões do quadros I.<sup>1</sup>

Na fase operacional o TLD poderá estar em um de quatro estados: TEI Designado, Esperando Estabelecimento de Conexão, Esperando Desconexão ou Multi-Quadros Estabelecido. Os estados intermediários, Esperando Estabelecimento de Conexão e Esperando Desconexão, são situações temporárias que ocorrem entre o momento em que uma entidade faz um pedido (de conexão ou desconexão) e o momento em que a entidade par aceita e confirma tal pedido. O estado Multi-Quadros Estabelecido, conforme implementado, contempla tanto os procedimentos normais quanto aqueles procedimentos definidos numa situação de Recuperação por Temporização.

Tão logo ocorra a mudança de fase de não operacional para operacional através de comando do processador, o TLD entra automaticamente no estado TEI Designado. A partir deste ponto a operação segue os procedimentos apresentados no anexo II, que apresenta a adaptação para uma implementação hardware do protocolo mostrado no anexo I; eles contemplam a versão do Livro Azul (1988) da Recomendação I.441 do CCITT e a Prática Telebrás *Especificação de Protocolo para a Camada de Enlace de Dados na Interface Usuário Rede na RDSI*. A volta à fase não operacional ocorre através de comando específico no RCMD2 (FNO) ou através de Reset Hardware.

As informações que chegam ao TLD para tratamento podem ter três origens distintas: processador local (através dos Registradores de Comando), Entidade Par Remota (quadros recebidos pela Porta B e passados ao TLD via THDLC) ou então gerados pelo THDLC local (por exemplo, expiração de temporização).

### 3.4.2 Comunicação com Níveis Superiores

A camada de enlace oferece à camada 3 serviços de transferência de pacotes de dados em operações reconhecida ou não reconhecida, nos quadros dos tipos I e UI. Essas transferências são executadas pelo THDLC; o TLD apenas controla os parâmetros e variáveis da transmissão, os reconhecimentos através de quadros de supervisão e as situações anormais, tais como erros no sequenciamento dos quadros e conseqüente retransmissão, controle do estado *ocupado*, recuperação por temporização e estouro de janela. Essas informações são passadas à camada 3 através de interrupções em REX1. Ainda para a camada 3 são passadas informações de indicação e de confirmação de conexões e desconexões através de interrupções codificadas no registrador REX2.

Diferentes situações de erro são apresentadas nos procedimentos do protocolo e que devem ser passadas à entidade de gerenciamento da camada. Esses erros são codificados no Registrador REX2, através de indicações de erro A até N. A tabela abaixo relaciona o tipo de erro com o estado em que ocorre e o motivo da

<sup>1</sup>O processador limita-se a solicitar serviços através de um comando; o procedimento completo de conexão/desconexão/transmissão é executado pelo CH2.

ocorrência. O tratamento a ser dado a essas informações é objeto de definição de cada sistema.

| Tipo de Erro | Estado | Motivo                                 |
|--------------|--------|----------------------------------------|
| A            | MQ     | Quadros resposta de supervisão com F=1 |
| B            | MQ     | Quadro DM com F=1                      |
| C            | TD/MQ  | Recebimento de quadro UA               |
| D            | EE/ED  | Recebimento de quadro UA               |
| E            | MQ     | Quadro DM com F=0                      |
| F            | MQ     | Recebimento de quadro SABME            |
| G            | EE     | Expiração de T200                      |
| H            | ED     | Expiração de T200                      |
| I            | MQ     | Estouro de N200                        |
| J            | MQ     | Erro de N(R)                           |
| K            | MQ     | Recebimento de FRMR                    |
| L            | todos  | Erro no campo de controle              |
| M            | todos  | Cpo info onde não permitido            |
| N            | todos  | Erro de compr. de quadro               |

### 3.4.3 Comunicação com Entidade Par

A comunicação do TLD com a camada 2 da entidade par se dá através de troca de quadros definidos na Recomendação I441 e mostrados na tabela a seguir.

| Tipo         | Comando | Resposta | Nome                                 |
|--------------|---------|----------|--------------------------------------|
| Não Numerado | SABME   |          | Set Assynchronous Bal. Mode Extended |
|              |         | DM       | Disconnect Mode                      |
|              | DISC    |          | Disconnect                           |
|              |         | UA       | Unnumbered Acknowledge               |
|              |         | FRMR     | Frame Reject                         |
| Supervisão   | RR      | RR       | Receiver Ready                       |
|              | RNR     | RNR      | Receiver Not Ready                   |
|              | REJ     | REJ      | Reject                               |
| Informação   | I       |          | Information                          |

A tabela mostra apenas os quadros tratados automaticamente pelo TLD. Cabe notar que o TLD recebe e trata quadros resposta FRMR e comando REJ, mas não os envia.

O THDLC é o bloco responsável pela detecção da chegada de qualquer desses quadros e, quando não há erro na recepção, o TLD é avisado (com exceção de chegada de quadro UI). O envio de quadros também é executado pelo THDLC a partir de uma ordem proveniente do TLD, exceção aos quadros I, UI e FRMR que são comandados pelo processador. A transmissão de quadros transparentes pelo THDLC é feita sob ordem do processador (para enviar, por exemplo, quadros XID ou FRMR), não tendo, no entanto, qualquer ação sobre o TLD, que sequer será avisado.

**Quadro FRMR** Para montagem do campo de informação do quadro resposta FRMR, o processador terá que usar as informações contidas nos bits ER4 a ER1 do Registrador REX2. Assim, o bit W corresponde ao código 1100 (indicação de erro L), o bit X corresponde ao código 1101 (indicação de erro M) e o bit Z corresponde ao código 1010 (indicação de erro J). O bit Y que indica que o tamanho do campo de informação recebido excede o máximo estabelecido não é detectado pelo chip.

**Quadro XID** Quadros XID são recebidos no modo automático e passados diretamente para a R-FIFO para tratamento pelo processador. O envio de quadros XID deve ser feito através de comando XQT.

**Retransmissão de Quadros I** A retransmissão de quadros I a partir do recebimento de um comando REJECT da entidade par ou a partir do recebimento de qualquer comando de supervisão quando na condição de recuperação por temporização, deverá ser executada pelo processador após o recebimento da interrupção RMX.

O CH2 não guarda quadros transmitidos e não reconhecidos em seu buffer interno. Tais quadros devem ficar armazenados em memória externa e o processador deverá liberá-los à medida que forem sendo reconhecidos. A interrupção MVA é gerada toda vez que um novo quadro for reconhecido; através da leitura de RHCR2 o processador terá controle sobre quais quadros liberar.

#### 3.4.4 Parâmetros e Variáveis do LAPD

No modo automático todo quadro I é numerado sequencialmente e pode assumir valores na faixa de 0 a 127, isto é, o módulo adotado pelo chip é 128 (modo estendido).

O processador tem acesso aos seguintes parâmetros e variáveis do LAPD:

- **Temporizador T200**

É programável no Registrador RTEM, podendo variar de 125 micro segundos até 4 segundos, se  $CK = 512 \text{ KHz}$ .

- **Número Máximo de Retransmissões N200:**  
É programável no Registrador RTEM, podendo variar de 0 até 7.
- **Número Máximo de Quadros I Pendentes (K):**  
É programável no Registrador RTJA, podendo variar de 0 a 127.
- **Temporizador T203:**  
É fixo em 10 segundos. Esse parâmetro é opcional e sua habilitação é feita no bit 7 do Registrador RTJA. Ele define o tempo máximo permitido sem que ocorra troca de quadros na camada 2, servindo como uma monitoração.
- **Variável de Estado de Emissão V(S):**  
Poderá ser lida no Registrador XNUR.
- **Variável de Estado de Recepção V(R):**  
Poderá ser lida no Registrador RNUR.
- **Variável de Estado de Reconhecimento V(A):**  
Poderá ser lida no Registrador RHCR2. Toda vez que ocorrer mudança em V(A), isto é, toda vez que um novo quadro I transmitido for reconhecido pela entidade par, será gerada uma interrupção MVA no Registrador REX1. O processador fará uso desta interrupção para liberar seu buffer de quadros transmitidos e não reconhecidos.
- **Contador de Retransmissões (RC):**  
Poderá ser lido no registrador REST.

**Interrupção TEM** São duas as situações em que é gerada uma interrupção do tipo TEM:

1. Quando da primeira expiração de T200. Neste caso, o TLD ativou a variável RER (Estado de Recuperação de Relógio), passando a operar nesta nova condição.
2. Quando o TLD está na condição de Recuperação po Temporização e o processador pede uma transmissão de quadro de informação (XQI).

### 3.4.5 O LAPD/TLD em LEDS

A transcrição dos LEDS do protocolo LAPD para a a implementação hardware não é direta, sendo necessárias algumas adaptações e tomada de decisão em caso de opções.

Os anexos I e II trazem:

1. LEDES do protocolo, como se encontra descrito na Recomendação I.441 do CCITT, traduzido de acordo com a Prática Telebrás *Especificação de Requisitos Mínimos para a Camada de Enlace de Dados dos Acessos Básico e Primário no Canal D da RDSI*.
2. LEDES visando a implementação hardware, composto de duas partes:
  - (a) Parte do TLD;
  - (b) Parte do THDLC.

Os itens a seguir ajudam a acompanhar a tradução de um para o outro:

1. Na I.441, as primitivas do tipo *ED Solicitação* são aquelas recebidas pela camada 2 provenientes da camada 3. Na implementação do CH2 elas se traduziram em comandos do processador via Registrador de Comando, da forma a seguir:
  - ED Solicitação de Estabelecimento : comando ECO
  - ED Solicitação de Liberação : comando DCO
  - ED Solicitação de Envio de Dados : comandos XQI, XQUI, XQT.

O mesmo ocorre em relação aos sinais gerados pela entidade de gerenciamento da conexão:

- Coloca em receptor local ocupado : comando RNR
  - Elimina condição receptor local ocupado : comando RR
2. O evento *Desativação Persistente* na I.441 é gerenciado no CH2 através da transferência de informação codificada no canal C/I da interface IOM. Ao ser avisado de tal ocorrência, através da interrupção MCI, o processador deve tirar o chip da fase operacional do modo automático, tratar a condição e depois retornar à fase operacional, para que o TLD volte ao estado *TEI Designado*.
  3. As indicações e confirmações do tipo *ED* na I.441 são avisos gerados pela camada 2 endereçados à camada 3. No CH2 eles são passados via interrupções do tipo *CI3-1=XXX*.  
O mesmo ocorre em relação às primitivas do tipo *GED*. Elas são geradas pela camada 2 e endereçadas à camada de gerência e no CH2 são passadas via interrupções do tipo *ER4-1=YYYY*.
  4. As operações com a *fila I* na I.441, de maneira geral, não são necessárias na implementação do CH2, uma vez que o gerenciamento da mesma é externo. O processador saberá quando descartá-la após o recebimento de interrupções do tipo *ED indicação*.

5. Foram incluídas a geração de interrupções em casos não previstos na I.441, mas importantes na implementação hardware:
- *MER* - Mudança no Estado da Estação Remota  
Importante para que o processador evite comandos de transmissão de quadros quando a estação remota estiver ocupada.
  - *RMT* - Repita Mensagem Transmitida  
Invoca ao processador a retransmissão de quadros já transmitidos anteriormente mas que tiveram problemas de algum tipo até chegar ao destino.
  - *MVA* - Mudança na Variável V(A)  
Servem para avisar ao processador que alguns dos quadros já transmitidos e que estão no seu buffer de transmissão aguardando reconhecimento podem ser descartados.
6. No modo automático, o tratamento da recepção de quadros com campo de informação foi dividida entre o THDLC e o TLD; isto está claro nos Leds.
- Quadros do tipo UI e XID são tratados exclusivamente pelo THDLC. Quadros FRMJ tem seu campo de informação carregado na R-FIFO pelo THDLC e o tratamento do campo de controle feito pelo TLD.
- Quadros do tipo I tem tratamento mais complexo. Para o TLD, eles são divididos em *I Livre* e *I Ocupado*, dependendo do estado da estação local. Se a estação local estiver ocupada ou se estiver livre mas com problema no número de sequência do quadro, o THDLC irá ignorar o campo de informação. Caso contrário o campo de informação será armazenado na R-FIFO e o TLD será avisado da chegada de quadro *I Livre*, desde que não tenha ocorrido algum erro na recepção dos tipos transbordamento, aborto ou CRC. A parte tratada no TLD segue a I.441.
7. A transmissão de quadros com campo de informação do tipo XID, FRMJ e UI é feita exclusivamente pelo THDLC. No caso de quadro do tipo I, a transmissão é feita também pelo THDLC em sua maior parte. Após uma transmissão com sucesso, o TLD recebe a informação através de um sinal QIT (Quadro I Transmitido) e executa algumas operações de finalização do tratamento.

### 3.5 Descrição do Bloco de FIFOS

A estrutura de FIFO do CH2, que consiste de duas FIFOs de Recepção (R-FIFO) com 32 octetos cada, e duas FIFOs de Transmissão (X-FIFO) também com 32 octetos cada, assim como de um controlador inteligente de FIFO, constroi uma conexão flexível entre os controladores de HDLC e de LAPD e as funções de protocolo de nível superior executadas no processador do sistema.

#### 3.5.1 Controle das R-FIFOs

A arquitetura de controle das R-FIFOs está mostrada na figura 31.

Para cada conjunto de 32 posições (chamadas R-FIFO 0 e R-FIFO 1) existe um multiplexador que permite o acesso a uma das FIFOs pelo fornecimento de endereço, ora pelo THDLC (para carregamento dos dados) ora pelo processador (para leitura dos mesmos). A lógica de seleção dos multiplexadores é tal que, a cada momento, somente uma FIFO tem permissão de acesso pelo processador, enquanto à outra é permitido o acesso pelo THDLC.

À cada FIFO estão associados dois registradores com 15 bits cada, onde cinco bits são usados para guardar endereços de posições da R-FIFO que sinalizam um fim de mensagem armazenada, um é chamado *flag de ocupação* (bit FO, cujo uso é interno, e, portanto, não acessável pelo processador), um é chamado FMR, para indicação do fim de mensagem e oito formam um Registrador de Estado de Recepção (RERC). Devido à limitação do número de registradores, somente duas mensagens inteiras (ou uma inteira e parte de uma segunda) podem ser armazenadas em cada R-FIFO, mesmo havendo espaço disponível para uma terceira. Ainda à cada conjunto é associado um bit chamado BIF (bit de identificação de FIFO). O processador tem acesso às informações referentes a cada mensagem armazenada pela leitura dos Registradores RERC e RIRF.

**Escrita na R-FIFO** Após o reset hardware ou o comando RHR, ambas as R-FIFOs estarão vazias, a R-FIFO 0 passa a ser a R-FIFO corrente (isto é, aquela acessada pelo HDLC) e os registradores associados (RERC e RIRF) estão carregados com o conteúdo 0.

Quando um quadro é recebido, os octetos são armazenados na R-FIFO corrente, em posições consecutivas, pelo THDLC. Se o quadro contiver mais que 32 octetos, 32 deles são armazenados e uma interrupção MRC é colocada na fila de espera.

Se este quadro tiver até 32 octetos, ao fim da mensagem será colocada uma interrupção MRC na fila de espera de interrupções e o endereço da última posição carregada é armazenado no primeiro registrador com os bits FO e FMR ativados em nível lógico 1 e os bits dos Registradores de Estado adequadamente ativados.

Na ocorrência de uma interrupção do tipo MRC, o fluxograma da figura 32 será seguido.<sup>1</sup>

---

<sup>1</sup>O teste *outra FIFO livre* do fluxograma da figura 32 trata da verificação de se algum bit

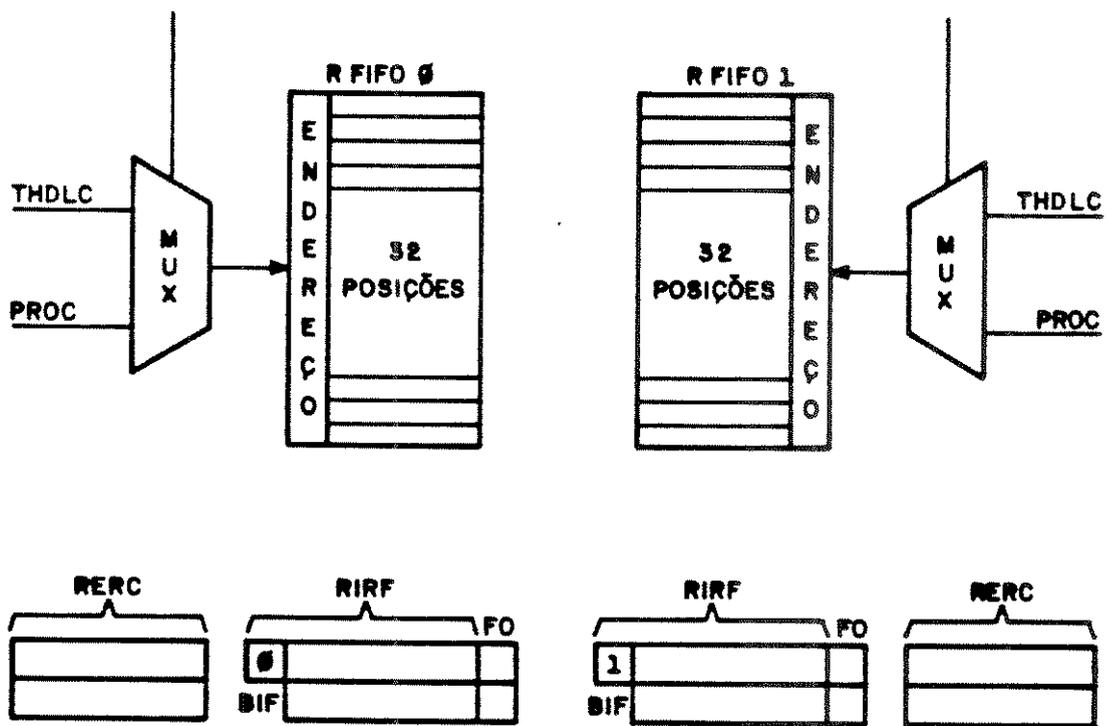


Figura 31: Arquitetura da R-FIFO

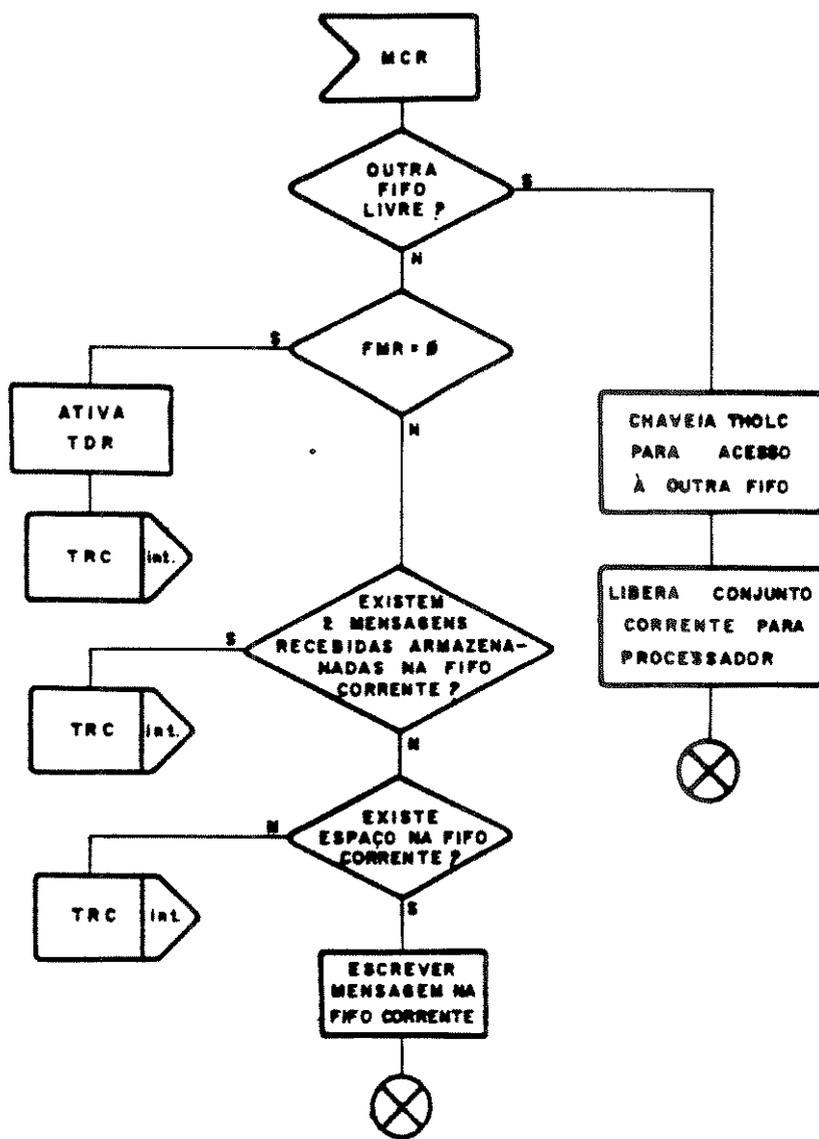


Figura 32: Algoritmo de Funcionamento da R-FIFO

Quando a segunda mensagem estiver para ser carregada na FIFO corrente, duas situações alternativas poderão ocorrer:

1. a mensagem cabe inteiramente nos octetos ainda disponíveis. Neste caso, uma nova interrupção MRC é colocada na fila de espera de interrupções e o endereço do último octeto carregado da mensagem é colocado no registrador RIRF que estiver vazio, onde é feito  $FO=FMR=1$ . A partir de então vale o fluxograma da figura 32.
2. a mensagem não cabe inteiramente nos octetos disponíveis. Neste caso são armazenados os octetos que couberem, uma interrupção MRC é colocada na fila de espera de interrupções e o endereço 31 é colocado no Registrador RIRF, onde é feito  $FO=1$ .

O armazenamento do endereço nos registradores é feito da forma descrita a seguir. O THDLC testa o bit FO do primeiro registrador. Se  $FO=0$ , o endereço é escrito neste. Se  $FO=1$  o mesmo teste é feito no segundo registrador. Neste caso, se  $FO=1$ , então não há registrador disponível, significando que uma nova mensagem não pode ser armazenada nesta FIFO.

**Leitura na R-FIFO** Ao receber uma interrupção do tipo MRC o processador deverá proceder à leitura do registrador RIRF. Com isso ele descobrirá, nos bits ADR1-5, qual foi a última posição carregada na R-FIFO, no bit BIF qual é a R-FIFO correntemente acessada e no bit FMR se foi fim de mensagem ou não. Para obter informações a respeito do quadro recebido, o processador deverá proceder à leitura do Registrador RERC.

O número de acessos de leitura que o processador deverá fazer na R-FIFO precisa ser igual à diferença entre o último valor em ADR1-5 lido e o valor corrente lido, se a R-FIFO for a mesma (isto é, o valor do bit BIF é o mesmo valor da leitura anterior) ou, no caso de ser outra R-FIFO acessada, deverão ser lidos tantos octetos quantos aqueles que vão da posição 0 até a indicada por ADR1-5.

**Transbordamento (Overflow)** Veja item *Circuito para Detecção de Transbordamento* da seção 3.3.2.

### 3.5.2 Controle das X-FIFOs

A arquitetura de controle das X-FIFOs está mostrada na figura 33.

Para cada conjunto de 32 posições (chamadas X-FIFO0 e X-FIFO1) existe um multiplexador que permite o acesso a uma dada FIFO, pelo fornecimento de endereço, ora pelo processador (para carregamento de dados) ora pelo THDLC

---

FO é igual a 1; em caso positivo, a FIFO não está livre.

(para leitura dos mesmos e envio pela linha). A lógica de seleção dos multiplexadores é tal que, a cada momento, somente uma X-FIFO tem permissão de acesso pelo processador, enquanto à outra é permitido o acesso pelo THDLC.

A cada FIFO está associado um contador crescente/decrecente. Durante o carregamento de dados pelo processador, o contador é incrementado a cada novo octeto. Durante a leitura pelo THDLC, o contador é decrementado a cada octeto lido.

**Escrita na X-FIFO** Após o reset, as X-FIFOs estão vazias, os contadores apontam a posição 0 e os multiplexadores estão selecionados de forma que o processador acessa as posições da X-FIFO0 e o THDLC acessa as posições da X-FIFO1.

O processador, quando deseja transmitir pacotes de dados, deve inicialmente carregá-los na X-FIFO que puder acessar, depois de verificar, pela leitura do bit XFE do Registrador de Estados (REST), se poderá fazê-lo. O bit XFE estará ativado sempre que pelo menos um dos contadores associados às X-FIFOs estiver zerado. A cada octeto carregado o contador é incrementado. Se o pacote de dados contiver até 32 octetos, após o carregamento completo na X-FIFO, o processador deve gerar o comando XQI se for quadro de informação numerado no modo automático (fase operacional), XQUI se for quadro de informação não numerado no modo automático (fase operacional) ou ainda se for quadro transparente (em qualquer modo), com a indicação de fim de quadro ativada (bit FMX). Se o pacote de dados for maior que 32 octetos, 32 deles são carregados e os comandos XQI, XQUI ou XQT são gerados, sem a ativação do bit FMX.

Ao receber algum desses comandos o CH2 irá verificar se a outra X-FIFO está livre (isto é, se o contador associado a ela está zerado) e, em caso positivo, os multiplexadores serão selecionados de forma que o processador e o THDLC troquem a X-FIFO que acessam e, se o bit FMX não está ativado, o THDLC gera uma interrupção BXP para avisar ao processador que um bloco de dados pode entrar na X-FIFO. Caso existam tais dados, o processador deve carregar os outros octetos na X-FIFO e dois comandos poderão ocorrer então:

1. se o pacote de dados acabou, o processador gera XQT, XQUI ou XQI com a ativação do bit FMX;
2. caso contrário, o processador deve gerar XQT, XQUI ou XQI sem ativação do bit FMX.

**Leitura na X-FIFO** O THDLC envia os dados contidos nas posições da X-FIFO a que tem acesso, iniciadas da posição 0. A cada dado enviado o contador é decrementado, até que atinja o conteúdo 0, deixando a X-FIFO em estado livre.

No caso do THDLC não encontrar mais dados disponíveis para enviar antes da ativação do indicador de fim de quadro (bit FMX), a mensagem transmitida

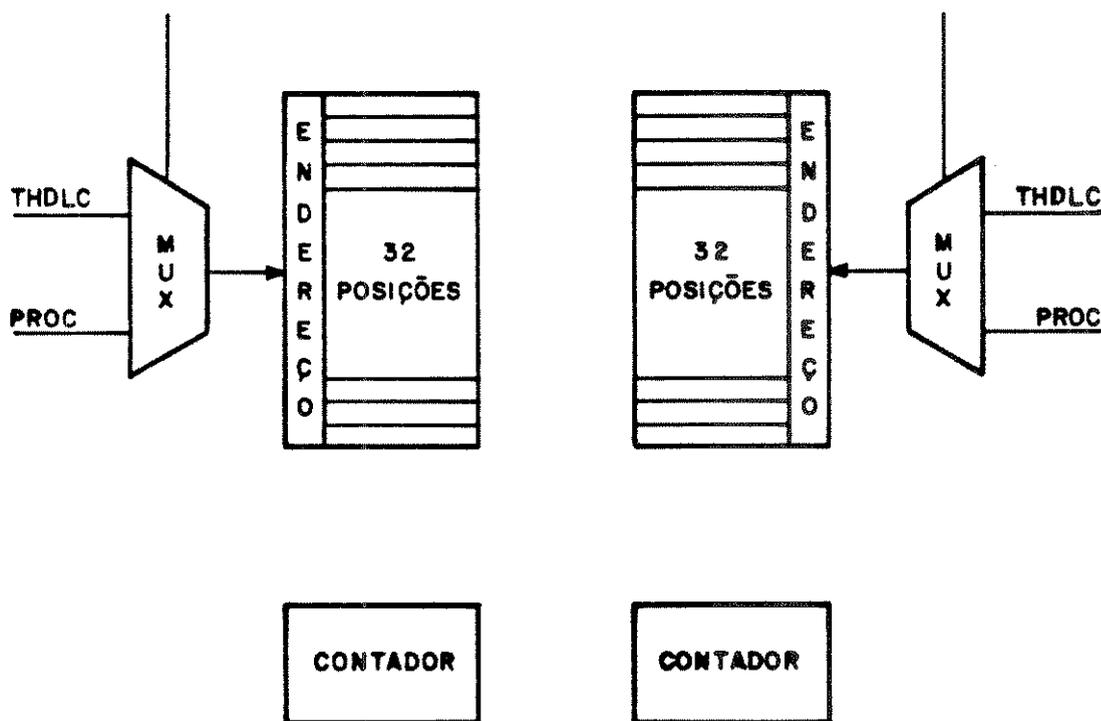


Figura 33: Arquitetura da X-FIFO

é abortada e o processador é avisado via interrupção UDX. A mensagem poderá ser abortada por desejo do processador através do comando XRES (Reset de Transmissão), quando então a X-FIFO será zerada e uma sequência de bits *uns* é transmitida.

### 3.5.3 Detecção de Colisão e Procedimentos

O componente de camada 1 acoplado ao CH2 via interface IOM irá fazer a detecção dos casos de interface ocupada ou ocorrência de colisão na linha. Na ocorrência do evento o CH2 será informado através da ativação do bit 5 do Canal Monitor. Ao receber a informação, o CH2 irá terminar a mensagem que porventura esteja sendo transmitida, enviando uma sequência de bits *uns* (*idle state*) e ficará aguardando o retorno do bit 5 ao nível lógico 0 (interface livre) para retomar a transmissão. Se a mensagem corrente continha mais que 32 octetos em X-FIFO, uma interrupção do tipo RMT é gerada para que o processador possa recarregar a mensagem.

### 3.6 Descrição do Conjunto de Registradores

A parametrização do CH2 assim como a transferência de dados e informações de controle entre o processador e o CH2 é executada via as FIFOs (de transmissão e recepção) e dois Conjuntos de Registradores.

As duas FIFOs são acessadas com os endereços 00 a 1FH, de forma a atingir todas as 32 posições de cada uma delas.

Os registradores da faixa de endereços 20H a 2FH são relacionados com os Tratadores de HDLC e de LAPD. As interfaces seriais são controladas e monitoradas com os registradores cujos endereço estão na faixa de 30H a 3FH.

As tabelas abaixo apresentam os registradores com seus endereços associados, divididos em Registradores de Leitura (pelo processador) e Registradores de Escrita.

| Endereço dos Registradores de Leitura |        |                                           |
|---------------------------------------|--------|-------------------------------------------|
| Endereço                              | Nome   | Descrição                                 |
| 00 a 1F                               | R-FIFO | FIFO de Recepção                          |
| 20                                    | REIN   | Reg. de Estado de Interrupções            |
| 21                                    | REST   | Reg. de Estado dos Controladores          |
| 22                                    | RERC   | Reg. de Estados da Recepção               |
| 23                                    | MOD0   | Reg. de Modo de Operação                  |
| 24                                    | RNUR   | Reg. de Numeração na Recepção             |
| 25                                    | XNUR   | Reg. de Numeração na Transmissão          |
| 26                                    | REX1   | Reg. de Extensão de Interrupções 1        |
| 27                                    | REX2   | Reg. de Extensão de Interrupções 2        |
| 28                                    | RIRF   | Reg. de Informações da R-FIFO             |
| 29                                    | TEI1   | Reg. do Valor 1 do Campo TEI              |
| 2A                                    | RHCR1  | Reg. do Campo de Controle HDLC Recebido 1 |
| 2B                                    | RHCR2  | Reg. do Campo de Controle HDLC Recebido 2 |
| 30                                    | RCPS1  | Reg. de Controle das Portas Seriais 1     |
| 31                                    | RCPS2  | Reg. de Controle das Portas Seriais 2     |
| 32                                    | RRCI   | Reg. de Recepção do Canal C/I             |
| 33                                    | RMON   | Reg. de Canal Monitor da IOM              |
| 34                                    | RSGS   | Reg. de Rx do Octeto SIG da SLD           |
| 35                                    | RFCS   | Reg. de Tx e Rx do Octeto FC da SLD       |
| 36                                    | XCB1   | Reg. de Dados p/ Tx de Canal B1           |
| 37                                    | XCB2   | Reg. de Dados p/ Tx de Canal B2           |
| 38                                    | RCB1   | Reg. de Dados p/ Rx de Canal B1           |
| 39                                    | RCB2   | Reg. de Dados p/ Rx de Canal B2           |

| Endereço dos Registradores de Escrita |        |                                       |
|---------------------------------------|--------|---------------------------------------|
| Endereço                              | Nome   | Descrição                             |
| 00 a 1F                               | X-FIFO | FIFO de Transmissão                   |
| 20                                    | MASC   | Reg. de Máscara de Interrupções       |
| 21                                    | RCMD1  | Reg. de Comando 1                     |
| 22                                    | RCMD2  | Reg. de Comando 2                     |
| 23                                    | MODO   | Reg. de Modo de Operação              |
| 24                                    | RTEM   | Reg. de Temporização                  |
| 25                                    | ENX1   | Reg. de Endereço de Transmissão 1     |
| 26                                    | ENX2   | Reg. de Endereço de Transmissão 2     |
| 27                                    | SAPI1  | Reg. do Valor 1 do Campo SAPI         |
| 28                                    | SAPI2  | Reg. do Valor 2 do Campo SAPI         |
| 29                                    | TEI1   | Reg. do Valor 1 do Campo TEI          |
| 2A                                    | TEI2   | Reg. do Valor 2 do Campo TEI          |
| 2B                                    | RTJA   | Reg. de Tamanho da Janela             |
|                                       |        |                                       |
| 30                                    | RCPS1  | Reg. de Controle das Portas Seriais 1 |
| 31                                    | RCPS2  | Reg. de Controle das Portas Seriais 2 |
| 32                                    | RXCI   | Reg. de Tx do Canal C/I               |
| 33                                    | RMON   | Reg. do Canal Monitor da IOM          |
| 34                                    | XSGS   | Reg. de Tx do Octeto SIG da SLD       |
| 35                                    | RFCS   | Reg. de Tx e Rx do Octeto FC da SLD   |
| 36                                    | XCB1   | Reg. de Dados p/ Tx de Canal B1       |
| 37                                    | XCB2   | Reg. de Dados p/ Tx de Canal B2       |

Para uma referência rápida, os registradores são apresentados abaixo divididos por funções:

- Condições e eventos internos

|      |                                           |
|------|-------------------------------------------|
| REIN | Registrador de Estado de Interrupções     |
| REX1 | Registrador de Extensão de Interrupções 1 |
| REX2 | Registrador de Extensão de Interrupções 2 |
| MASC | Registrador de Máscara de Interrupções    |
| REST | Registrador de Estado dos Controladores   |

- Campo de Dados do HDLC/LAPD

|       |                     |
|-------|---------------------|
| RFIFO | FIFO de Recepção    |
| XFIFO | FIFO de Transmissão |

- Controle e operação do HDLC/LAPD

|       |                                         |
|-------|-----------------------------------------|
| RCMD1 | Registrador de Comandos 1               |
| RCMD2 | Registrador de Comandos 2               |
| RTEM  | Registrador de Temporização             |
| RERC  | Registrador de Estados de Recepção      |
| MODO  | Registrador de Modo de Operação         |
| RFIR  | Registrador com Informações da R-FIFO   |
| RNUR  | Registrador de Numeração na Recepção    |
| XNUR  | Registrador de Numeração na Transmissão |
| RTJA  | Registrador de Tamanho da Janela        |

- Endereços do HDLC/LAPD

|       |                                          |
|-------|------------------------------------------|
| ENX1  | Registrador de Endereço de Transmissão 1 |
| ENX2  | Registrador de Endereço de Transmissão 2 |
| SAPI1 | Registrador do Valor 1 do Campo SAPI     |
| SAPI2 | Registrador do Valor 2 do Campo SAPI     |
| TEI1  | Registrador do Valor 1 do Campo TEI      |
| TEI2  | Registrador do Valor 2 do Campo TEI      |

- Campo de Controle do LAPD

|       |                                                  |
|-------|--------------------------------------------------|
| RHCR1 | Registrador do Campo de Controle HDLC Recebido 1 |
| RHCR2 | Registrador do Campo de Controle HDLC Recebido 2 |

- Controle e Dados de Interfaces Seriais

|       |                                                   |
|-------|---------------------------------------------------|
| RCPS1 | Registrador de Controle de Portas Seriais 1       |
| RCPS2 | Registrador de Controle de Portas Seriais 2       |
| XCB1  | Registrador de Dados para Transmissão de Canal B1 |
| RCB1  | Registrador de Dados para Recepção de Canal B1    |
| XCB2  | Registrador de Dados para Transmissão de Canal B2 |
| RCB2  | Registrador de Dados para Recepção de Canal B2    |
| RXCI  | Registrador de Transmissão do Canal C/I           |
| RRCI  | Registrador de Recepção do Canal C/I              |
| XSGS  | Registrador de Transmissão do Octeto SIG da SLD   |
| RSGS  | Registrador de Recepção do Octeto SIG da SLD      |
| RFCS  | Registrador de Tx e Rx do Octeto FC da SLD        |
| RMON  | Registrador de Canal Monitor da IOM               |

As tabelas a seguir apresentam cada registrador do CH2 com seus bits; a primeira se refere aos registradores relacionados com os Tratadores de HDLC e a segunda se refere aos registradores relacionados com o Bloco de Interfaces.

|       |      |      |      |      |      |      |      |      |
|-------|------|------|------|------|------|------|------|------|
| REIN  | MRC  | SIN  | MER  | BXP  | TEM  | MCI  | EI1  | EI2  |
| REX1  | RMT  | UDX  | TRC  | TTS  | MOR  | TCS  | EJA  | MVA  |
| REX1  | ER4  | ER3  | ER2  | ER1  | PRO  | CI3  | CI2  | CI1  |
| MASC  | MI7  | MI6  | MI5  | MI4  | MI3  | MI2  | MI1  | MI0  |
| REST  | DAR  | XFE  | XRNR | RRNR | RER  | RC3  | RC2  | RC1  |
| RCMD1 | XQUI | RHR  | RNR  | RR   | XQT  | XQI  | FMX  | XRES |
| RCMD2 | —    | —    | ITE  | RAC  | FNO  | TED  | ECO  | DCO  |
| MODO  | SMD1 | SMD0 | MEM  | SCL1 | SCL0 | HMD2 | HMD1 | HMD0 |
| RTEM  | N23  | N22  | N21  | TR5  | TR4  | TR3  | TR2  | TR1  |
| RIRF  | BIF  | FMR  |      | END5 | END4 | END3 | END2 | END1 |
| RERC  | TDR  | CRC  | MRA  | SA1  | SA0  | C/R  | TE1  | TE0  |
| RTJA  | T3H  | K6   | K5   | K4   | K3   | K2   | K1   | K0   |
| ENX1  |      |      |      |      |      |      | 0    | 0    |
| ENX2  |      |      |      |      |      |      |      | 0    |
| SAPI1 |      |      |      |      |      |      | CRI  | 0    |
| SAPI2 |      |      |      |      |      |      | 0    | 0    |
| TEI1  |      |      |      |      |      |      |      | 1    |
| TEI2  |      |      |      |      |      |      |      | 1    |
| RHCR1 |      |      |      |      |      |      |      |      |
| RHCR2 |      |      |      |      |      |      |      |      |
| XNUR  | XU7  | XU6  | XU5  | XU4  | XU3  | XU2  | XU1  | XU0  |
| RNUR  | RU7  | RU6  | RU5  | RU4  | RU3  | RU2  | RU1  | RU0  |

|       |      |      |      |      |      |      |      |      |
|-------|------|------|------|------|------|------|------|------|
| RCPS1 | PBZ  | SAC  | MPS  | IOM  | B1C1 | B1C0 | B2C1 | B2C0 |
| RCPS2 | STS3 | STS2 | STS1 | STS0 | ST1  | ST0  | SC1  | SC0  |
| RXC1  | C    | O    | D    | E    | DTS  | TLP  | XCT  | XCE  |
| RRCI  | C    | O    | D    | E    | —    | —    | RCT  | BVS  |
| RMON  |      |      |      |      |      |      |      |      |
| XCB1  |      |      |      |      |      |      |      |      |
| XCB2  |      |      |      |      |      |      |      |      |
| RCB1  |      |      |      |      |      |      |      |      |
| RCB2  |      |      |      |      |      |      |      |      |
| XSGS  |      |      |      |      |      |      |      |      |
| RSGS  |      |      |      |      |      |      |      |      |
| RFCS  |      |      |      |      |      |      |      |      |

### 3.6.1 Registradores de Condições e Eventos Internos

#### REIN - Reg. de Estado de Interrupções

- Tipo : leitura
- Endereço : 20H
- Valor após Reset : 00H
- Formato:

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| MRC   | SIN   | MER   | BXP   | TEM   | MCI   | EI1   | EI2   |

– *MRC - Mensagem Recebida*

Indica que a mensagem recebida está completa. Indica, assim, que foi recebida ou uma mensagem inteira (se menor que 32 octetos) ou a parte restante de uma mensagem mais longa. O conteúdo está disponível na R-FIFO e informações adicionais podem ser obtidas dos registradores RIRF e RERC.

– *SIN - Interrupção de Transferência Síncrona*

Indica a ocorrência de uma transferência síncrona conforme programado em RCPS2, isto é, no meio ou no início de um quadro IOM.

– *MER - Mudança de Estado do Receptor*

Válido somente no modo automático. Um quadro do tipo RR/RNR foi recebido da estação remota, indicando uma mudança no estado de recepção da mesma. O estado atual pode ser lido no bit RRNR do registrador REST.

– *BXP - Buffer de Transmissão Pronto*

Um bloco de dados de até 32 octetos pode ser escrito na XFIFO. A interrupção BXP será gerada depois de um comando de transmissão de quadro, quando uma das XFIFOs estiver vazia e o quadro ainda não foi completamente carregado ( comando FMT - Fim de Mensagem a Transmitir).

– *TEM - Interrupção de Temporização*

Interrupção de estouro de temporização. Veja seções *Temporizações*, *Temporizações de LAPD* e registrador RTEM.

- *MCI - Mudança no Código C/I*  
Foi reconhecida uma mudança no canal C/I. O mesmo código C/I, que difere daquele recebido anteriormente, foi recebido no mínimo duas vezes. O registrador RRCI armazena o código C/I.
- *EI1 - Extensão de Interrupções 1*  
Uma vez ativado, o processador deve verificar o registrador REX1 que contém informações sobre outras interrupções que ocorreram.
- *EI2 - Extensão de Interrupções 2*  
Uma vez ativado, o processador deve verificar o registrador REX2 que contém informações sobre outras interrupções que ocorreram.

### REX1 - Registrador de Extensão de Interrupções 1

- Tipo : leitura
- Endereço : 26H
- Valor após Reset : 00
- Formato:

|       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|
| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
| RMT   | UDX   | TRC   | TTS   | MOR   | TCS   | EJA   | MVA   |

#### - *RMT - Repetição de Mensagem Transmitida*

O último quadro transmitido deve ser novamente carregado na X-FIFO. Os quadros devem ser retransmitidos ou porque ocorreu colisão no barramento S após 32 octetos de dados já terem sido transmitidos, quando então a X-FIFO já foi comutada, ou, na fase operacional do modo automático, devido a problemas na recepção da unidade remota. Este segundo caso fica melhor explicado nos *LEDS* do estado *Multiquadros*.

Para janelas diferentes de 1, o processador deverá ler os registradores RHCR2 e XNUR para determinar quais quadros deverão ser retransmitidos.

#### - *UDX - Não Disponibilidade dos Dados a Transmitir*

O processador não forneceu em tempo os dados que completam um quadro que está sendo transmitido. O quadro foi terminado com uma sequência de aborto.

**Nota:** Se as interrupções RMT ou UDX estiverem ativadas, não é possível transmitir quadros de qualquer tipo; a leitura do registrador REX1 é suficiente para desativá-las.

- *TRC - Transbordamento na Recepção*  
Duas situações distintas provocam esta interrupção. Na primeira, todo um quadro recebido não pode ser armazenado pois a R-FIFO já estava lotada. Na segunda, ocorreu transbordamento de dados na trama corrente, e, neste caso, o bit TDR do registrador RERC também é ativado. Veja fluxograma da figura 32.
- *TTS - Transbordamento na Transferência Síncrona*  
Uma nova interrupção de sincronismo é ativada antes que o processador tenha reconhecido a interrupção anterior.
- *MOR - Octeto Monitor Recebido*  
Um octeto monitor recebido pela porta B quando programada como interface IOM está disponível no registrador RMON. Isso ocorre somente quando, no quadro IOM recebido anteriormente, o bit E foi ativado (nível lógico 0) pelo componente de camada 1, indicando ao CH2 a validade do octeto monitor.
- *TCS - Transição no Canal de Sinalização*  
Foi detetada uma transição na recepção do canal de sinalização SIG da interface SLD.
- *EJA - Estouro de Janela*  
Este bit é válido somente na fase operacional do modo automático. Indica que o número máximo de quadros transmitidos e não reconhecidos, número esse programado no registrador RTJA, foi atingido.
- *MVA - Mudança na Variável V(A)*  
Ocorreu mudança no valor da variável interna V(A), isto é, novos quadros I transmitidos foram reconhecidos. É um bit válido somente na fase operacional do modo automático.

## REX2 - Registrador de Extensão de Interupções 2

- Tipo : leitura
- Endereço : 27H
- Valor após Reset : 00H
- Nota : todo o registrador só tem validade na fase operacional do modo automático.
- Formato :

|       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|
| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
| ER4   | ER3   | ER2   | ER1   | PRO   | CI3   | CI2   | CI1   |

– *ER4 : ER1 - Indicadores de Erro de Protocolo*

Codificam o tipo de erro que porventura venha a ocorrer no processamento do protocolo LAPD, de acordo com a tabela abaixo. A ocorrência de cada tipo de erro está apresentada nos *LEDS* do anexo I. A seção 3.4 traz maiores detalhes.

| Código | Significado              |
|--------|--------------------------|
| 0000   | Não há indicação de erro |
| 0001   | Indicação de erro tipo A |
| 0010   | Indicação de erro tipo B |
| 0011   | Indicação de erro tipo C |
| 0100   | Indicação de erro tipo D |
| 0101   | Indicação de erro tipo E |
| 0110   | Indicação de erro tipo F |
| 0111   | Indicação de erro tipo G |
| 1000   | Indicação de erro tipo H |
| 1001   | Indicação de erro tipo I |
| 1010   | Indicação de erro tipo J |
| 1011   | Indicação de erro tipo K |
| 1100   | Indicação de erro tipo L |
| 1101   | Indicação de erro tipo M |
| 1110   | Indicação de erro tipo N |

– *PRO - Processador Remoto Ocupado*

Durante a transmissão de um quadro do tipo I, após comando XQI (registrador RCMD1) foi detetada a condição de processador da estação remota ocupado.

– *CI3 : CII - Indicadores e Confirmadores de Conexão/Desconexão*

Codificam informações de conexão e desconexão no processamento do protocolo LAPD, de acordo com a tabela a seguir.

| Código | Significado               |
|--------|---------------------------|
| 0XX    | Nada a informar           |
| 100    | Confirmação de desconexão |
| 101    | Confirmação de conexão    |
| 110    | Indicação de desconexão   |
| 111    | Indicação de conexão      |

### MASC - Registrador de Máscaras de Interrupções

- Tipo : escrita
- Endereço : 20H
- Valor após Reset : 00H
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| MI7   | MI6   | MI5   | MI4   | MI3   | MI2   | MI1   | MI0   |

Cada interrupção do registrador REIN pode ser mascarada individualmente ativando o bit respectivo no registrador MASC. As interrupções mascaradas não são indicadas ao ler o registrador REIN; todavia, elas permanecem armazenadas e são liberadas ao processador após o cancelamento do processo de mascaramento.

As interrupções dos registradores REX1 e REX2 não podem ser mascaradas individualmente. A máscara é feita *por registrador*, através do bit MI1 para o registrador REX1 e MI0 para o registrador REX2. Quando tais bits estiverem mascarados, nenhuma interrupção estendida é gerada, porém os bits permanecem ativados em REIN na ocorrência das mesmas. Qualquer leitura em REIN irá desativar esses bits, porém somente a leitura em REX1 ou REX2 irá desativar os bits nesses registradores.

### REST - Registrador de Estado dos Controladores

- Tipo : leitura
- Endereço : 21H
- Valor após Reset : 40H
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| DAR   | XFE   | XRNR  | RRNR  | RER   | RC3   | RC2   | RC1   |

#### – *Dados Recebidos*

Indica que existem dados recebidos pelo CH2 armazenados na R-FIFO e que ainda não foram lidos pelo processador. Pode ser usado, por exemplo, em operações tipo *pooling*.

- *XFE - Permissão de Escrita na X-FIFO*  
Indica que pelo menos um dos conjuntos que compõem a X-FIFO está vazio e que, portanto, dados a serem transmitidos podem ser carregados pelo processador.
- *XRNR - Indicador de Condição RNR Local*  
Indica a condição atual do CH2 em relação à recepção de dados; quando em nível 1 significa que o CH2 não está pronto para receber dados. Seu valor é determinado através de comandos do processador. Válido somente no modo automático.
- *RRNR - Indicador de Condição RNR Remota*  
Indica a condição atual da estação remota em relação à recepção de dados; quando em nível 1, significa que a estação remota enviou um comando RNR, indicando que não está pronta para receber dados. Válido somente no modo automático.
- *RER - Estado de Recuperação de Relógio*  
Indica se o protocolo LAPD no TLD está na condição de recuperação de relógio (nível 1) ou não (nível 0). Válido no modo automático.
- *RC3:1 - Contador de Retransmissão*  
Indica o número de vezes que um mesmo quadro foi transmitido. Válido no modo automático.

### 3.6.2 Registradores de Controle de Operação do LAPD/HDLC

#### RCMD1 - Registrador de Comando 1

- Tipo : escrita
- Endereço : 21H
- Valor após Reset : 00H
- Formato :

|       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|
| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
| XQUI  | RHR   | RNR   | RR    | XQT   | XQI   | FMX   | XRES  |

- *XQUI - Transmite Quadro de Informação Não- Numerado*  
Habilita a transmissão dos dados carregados anteriormente na X-FIFO em um quadro tipo UI. Válido no modo automático.
- *RHR - Reset Receptor HDLC*  
O receptor de HDLC é desativado. Todos os dados, assim como o conteúdo do *buffer* da mensagem, são apagados. Deve ser usado, por exemplo, após o aviso de ocorrência de transbordamento (*overflow*). A reativação do receptor HDLC é feita por comando RAC (RCMD2).
- *RNR - Receptor Não-Pronto*  
Quando em nível lógico 1, o CH2 é colocado na condição *não pronto para receber* e o bit XRNR do registrador REST é ativado. Válido somente no modo automático.
- *RR - Receptor Pronto*  
Quando em nível lógico 1, o CH2 é colocado na condição *pronto para receber* e o bit XRNR do registrador REST é zerado. Válido somente no modo automático.
- *XQT - Transmite Quadro Transparente*  
Habilita a transmissão de um quadro transparente, onde o THDLC limita-se a envelopar os dados armazenados na X-FIFO, adicionando os campos de Flag e CRC.
- *XQI - Transmite Quadro de Informação*  
Habilita a transmissão de dados carregados anteriormente na X-FIFO em um quadro numerado tipo I. Válido no modo automático.
- *FMX - Fim de Mensagem Transmitida*  
Este bit deve ser ativado junto a um dos comandos de transmissão de quadro (XQUI, XQT ou XQI) para avisar o CH2 que a parte remanescente de um quadro (se maior que 32 octetos) ou todo ele (se menor ou igual a 32 octetos) está presente na X-FIFO.
- *XRES - Reset de Transmissão*  
O transmissor de HDLC é desativado, são apagados os dados que porventura estiverem na X-FIFO e são transmitidos bits *1s* (estado de *idle*). O comando é usado para abortar um quadro.

## RCMD2 - Registrador de Comando 2

- Tipo : escrita
- Endereço : 22H
- Valor após Reset : 00H

- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
|       |       | ITE   | RAC   | FNO   | TED   | ECO   | DCO   |

- *ITE - Inicia Temporizador*  
Este bit não tem função no modo automático. Nos outros modos, esse comando habilita o início da contagem da temporização.
- *RAC - Receptor Ativo*  
Os circuitos de recepção de dados são ativados.
- *FNO - Fase Não Operacional*  
Este comando só é válido no modo automático e serve para transferir o TLD da fase operacional para a fase não operacional.
- *TED - TEI Designado*  
Este comando, válido somente no modo automático, deve ser usado para transferir o TLD da fase não operacional para a fase operacional. O TLD vai entrar diretamente no estado *TEI Designado*.
- *ECO - Estabelecimento de Conexão*  
Este comando, válido somente no modo automático, é usado para solicitar ao TLD que inicie o estabelecimento de uma conexão de enlace.
- *DCO - Estabelecimento de Desconexão*  
Este comando, válido somente no modo automático, é usado para solicitar ao TLD que inicie o estabelecimento de uma desconexão de enlace.

#### MODO - Registrador de Modo de Operação

- Tipo : leitura/escrita
- Endereço : 23H
- Valor após Reset : 00H
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| SMD1  | SMD0  | MEM   | SCL1  | SCL0  | HMD2  | HMD1  | HMD0  |

- *SMD1:SMD0 - Seleção de Modo*  
Seleciona o modo de operação do THDLC, de acordo com a tabela a seguir.

| Código | Significado                 |
|--------|-----------------------------|
| 00     | Modo automático             |
| 01     | Modo não automático         |
| 10     | Modo transparente           |
| 11     | Modo transparente estendido |

– *MEM - Modo de Endereçamento*

Define o número de octetos que compõem o campo de endereço do quadro HDLC, de acordo com a tabela abaixo.

| Nível | Significado     |
|-------|-----------------|
| 0     | Um único octeto |
| 1     | Dois octetos    |

– *SCL1:SCL0 - Seleção de Canal*

Seleciona o canal a ser tratado pelo TLD na fase operacional do modo automático ou a ser enviado às FIFOs de recepção nos outros modos, de acordo com a tabela abaixo.

| Código | Significado        |
|--------|--------------------|
| 00     | Transfere canal D  |
| 01     | Transfere canal B1 |
| 10     | Transfere canal B2 |
| 11     | Reservado          |

– *HMD2:HMD0 - Modos da Porta B*

Seleciona o modo de operação da Porta B, de acordo com a tabela a seguir.

| Código | Significado                                                                                                                                                                                                                                                                                                                                                                                         |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 000    | Interface IOM com aplicações onde o Canal Monitor não é usado.<br>Esta condição pode ocorrer em duas situações distintas:<br>A primeira é numa configuração ponto a multiponto (barramento S), desde que apenas um terminal esteja conectado (sem disputa de canal D) e a outra é numa configuração ponto a ponto (CH2 ligado a um chip de camada 1 que se liga diretamente à linha de transmissão) |
| 001    | Interface IOM, com utilização do octeto Monitor para controle de acesso ao canal D (ponto a multiponto em um barramento S).                                                                                                                                                                                                                                                                         |
| 010    | Interface IOM ponto a ponto, sendo o octeto Monitor utilizado para transmitir dados ou comandos.                                                                                                                                                                                                                                                                                                    |
| 011    | Função não definida                                                                                                                                                                                                                                                                                                                                                                                 |
| 100    | Função não definida                                                                                                                                                                                                                                                                                                                                                                                 |
| 101    | Interface HDLC com sinal de habilitação (strobe) ativo em nível lógico 0.                                                                                                                                                                                                                                                                                                                           |
| 110    | Interface HDLC com sinal de habilitação (strobe) ativo em nível lógico 1.                                                                                                                                                                                                                                                                                                                           |
| 111    | Interface HDLC com sinal de dados contínuo.                                                                                                                                                                                                                                                                                                                                                         |

#### RTEM - Registrador de Temporização

- Tipo : escrita
- Endereço : 24H
- Valor após Reset : 00H
- Formato:

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| N23   | N22   | N21   | TR5   | TR4   | TR3   | TR2   | TR1   |

– *N23:N21 - Valor do Parâmetro N200*

Define o número máximo de vezes que um mesmo quadro poderá ser retransmitido. É um parâmetro do protocolo LAPD, válido somente no modo automático.

– *TR5:TR1 - Valor Base do Temporizador*

O valor base programado nestes bits serve para definir duas diferentes temporizações do circuito, dependendo do modo de operação. No modo automático, define o parâmetro T200 do protocolo LAPD e nos demais modos define o tempo entre o momento em que é gerado

o comando ITE até o momento em que aparece uma interrupção TEM.

A equação abaixo define a relação entre o valor programado e a temporização correspondente:

$$t = [125 * (\text{valor} + 1)] [ns]$$

### RIRF - Registrador com Informação da R-FIFO

- Tipo : leitura
- Endereço : 28H
- Valor após Reset : 00H
- Nota : Este registrador só contém valores válidos após a ocorrência de uma interrupção RCV. Ele pode ser lido várias vezes até que seja lido o registrador RERC. A partir da leitura de RERC os dados em RIRF se referem a outra mensagem.
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| BIF   | FMR   |       | END5  | END4  | END3  | END2  | END1  |

– *BIF - Bit Indicador de FIFO*

A R-FIFO é composta de dois conjuntos de 32 octetos cada. Este bit define qual desses conjuntos está em condições de ser acessado (lido) pelo processador naquele momento.

– *FMR - Fim de Mensagem Recebida*

Em nível lógico 1 significa que o quadro recebido terminou; caso contrário, significa que a R-FIFO está lotada, mas o quadro que está sendo recebido ainda não terminou.

– *END5:1 - Endereço na R-FIFO*

Indica o endereço na R-FIFO que contém o último octeto da mensagem recebida.

### RERC - Registrador de Estados da Recepção

- Tipo : leitura
- Endereço : 22H

- Valor Após Reset : 00H
- Formato :

|       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|
| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
| TDR   | CRC   | MRA   | SA1   | SA0   | C/R   | TE1   | TE0   |

- *TDR - Transbordamento de Dados Recebidos*  
Indica que ocorreu um transbordamento de dados no quadro que está correntemente sendo recebido. Desta forma, o quadro deve ser descartado.
- *CRC - Código de Redundância Cíclica*  
Indica a ocorrência de um erro de CRC no quadro recebido.
- *MRA - Mensagem Recebida Abortada*  
Indica que o quadro corrente foi abortado, isto é, foi terminado com uma sequência de aborto.
- *SA1, SA0 - Endereço SAPI*  
Identificam o valor do primeiro octeto do campo de endereço do quadro recebido (SAPI), de acordo com a tabela XX.

| Código | Significado                            |
|--------|----------------------------------------|
| 00     | octeto igual ao conteúdo do reg. SAPI1 |
| 01     | octeto igual ao conteúdo do reg. SAPI2 |
| 10     | octeto igual ao valor de SAPI de grupo |
| 11     | não definido                           |

**Nota :** se os valores programados em SAPI1 e SAPI2 forem iguais, o código presente será 00.

- *C/R - Bit Comando / Resposta*  
Contém o valor do bit 1 do primeiro octeto do campo de endereço do quadro recebido.
- *TE1, TE0 - Identificador de Terminal*  
Identificam o valor do segundo octeto do campo de endereço do quadro recebido (TEI), de acordo com a tabela YY.

| Código | Significado                           |
|--------|---------------------------------------|
| 00     | octeto igual ao conteúdo do reg. TEI1 |
| 01     | octeto igual ao conteúdo do reg. TEI2 |
| 10     | octeto igual ao valor do TEI de grupo |
| 11     | não definido                          |

### RTJA - Registrador do Tamanho da Janela

- Tipo : escrita
- Endereço : 2BH
- Valor Após Reset : 01H
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| T3H   | K6    | K5    | K4    | K3    | K2    | K1    | K0    |

- *K0:K6 - Parâmetro K do Protocolo LAPD*  
No modo automático, define o tamanho da janela, isto é, o número máximo de quadros pendentes sem reconhecimento.
- *TH3 - Habilita T203*  
Permite o funcionamento do temporizador T203 interno (T203 é um parâmetro opcional do protocolo LAPD).

### RNUR - Registrador de Numeração na Recepção

- Tipo : leitura
- Endereço : 24H
- Valor Após Reset : 01H
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| RU7   | RU6   | RU5   | RU4   | RU3   | RU2   | RU1   | RU0   |

- *RU7:RU0 - Parâmetro V(R) do Protocolo LAPD*  
Contém o número do próximo quadro I a ser recebido.

### **XNUR - Registrador de Numeração na Transmissão**

- Tipo : leitura
- Endereço : 25H
- Valor Após Reset : 01H
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| XU7   | XU6   | XU5   | XU4   | XU3   | XU2   | XU1   | XU0   |

– *XU7:XU0 - Parâmetro V(S) do Protocolo LAPD*

Válido no modo automático. Contém o número do próximo quadro I a ser transmitido.

### **3.6.3 Registradores com Endereços do HDLC/LAPD**

#### **ENX1 - Registrador de Endereço de Transmissão 1**

- Tipo : escrita
- Endereço : 25H

Válido somente no modo automático. Numa transmissão de quadro, o conteúdo desse registrador irá compor os 6 bits mais significativos do primeiro octeto (SAPI) do campo de endereço.

#### **ENX2- Registrador de Endereço de Transmissão 2**

- Tipo : escrita
- Endereço : 26H

Válido somente no modo automático. Numa transmissão de quadro, o conteúdo desse registrador irá compor os 7 bits mais significativos do segundo octeto (TEI) do campo de endereço.

## SAPII - Registrador do Valor 1 do Campo SAPI

- Tipo : escrita
- Endereço : 27H
- Formato :

|            |       |       |
|------------|-------|-------|
| bits 7 a 2 | bit 1 | bit 0 |
| S A P I    | CRI   | 0     |

– *SAPI - Identificador do Ponto de Acesso ao Serviço*

Esse valor é usado para análise do primeiro octeto do campo de endereço dos quadros recebidos.

– *CRI - Interpretação do bit Comando / Resposta*

Define a interpretação a ser dada ao bit C/R do quadro recebido ou a ser transmitido, de acordo com a tabela abaixo.

| CRI | C/R | Significado          |
|-----|-----|----------------------|
| 1   | 0   | comando recebido     |
| 1   | 1   | resposta recebida    |
| 1   | 1   | comando transmitido  |
| 1   | 0   | resposta transmitida |
| 0   | 1   | comando recebido     |
| 0   | 0   | resposta recebida    |
| 0   | 0   | comando transmitido  |
| 0   | 1   | resposta transmitida |

### Aplicação

1. Chip usado no lado da central : CRI = 1
  - \* Na recepção
    - C/R = 0 : quadro comando recebido
    - C/R = 1 : quadro resposta recebido
  - \* Na transmissão
    - C/R = 0 : quadro resposta transmitido
    - C/R = 1 : quadro comando transmitido
2. Chip usado no lado do terminal : CRI = 0
  - \* Na recepção
    - C/R = 0 : quadro resposta recebido

- C/R = 1 : quadro comando recebido
- \* Na transmissão
  - C/R = 0 : quadro comando transmitido
  - C/R = 1 : quadro resposta transmitido

### **SAPI2 - Registrador do Valor 2 do Campo SAPI**

- Tipo : escrita
- Endereço : 28H
- Formato :

|            |       |       |
|------------|-------|-------|
| bits 7 a 2 | bit 1 | bit 0 |
| S A P I    | 0     | 0     |

Numa recepção de quadro, o conteúdo desse registrador é usado para análise do primeiro octeto do campo de endereço.

### **TEI1 - Registrador do Valor 1 do Campo TEI**

- Tipo : escrita/leitura
- Endereço : 29H
- Formato :

|             |       |
|-------------|-------|
| bits 7 a 1  | bit 0 |
| T E I ( 1 ) | 1     |

O valor programado neste registrador é válido somente nos modos automático e não automático e é usado para análise do segundo octeto do campo de endereço dos quadros recebidos. No modo transparente o segundo octeto do campo de endereço do quadro recebido poderá ser lido neste registrador.

### **TEI2 - Registrador do Valor 2 do Campo TEI**

- Tipo : escrita
- Endereço : 2AH
- Formato :

|             |       |
|-------------|-------|
| bits 7 a 1  | bit 0 |
| T E I ( 2 ) | 1     |

Válido nos modos automático e não automático, o conteúdo desse registrador é usado para análise do segundo octeto do campo de endereço dos quadros recebidos.

#### **3.6.4 Registradores com Campo de Controle do LAPD**

##### **RHCR1 - Registrador do Campo de Controle HDLC Recebido 1**

- Tipo : leitura
- Endereço : 2AH
- Valor Após Reset : 00H

Contém o primeiro octeto do campo de controle do quadro recebido.

##### **RHCR2 - Registrador do Campo de Controle HDLC Recebido 2**

- Tipo : leitura
- Endereço : 2BH
- Valor Após Reset : 00H

Contém o segundo octeto do campo de controle do quadro recebido.

#### **3.6.5 Registradores do Campo de Dados do HDLC/LAPD**

##### **R-FIFO - FIFO de Recepção**

- Tipo : leitura
- Endereço : 00 a 1FH

Armazena os octetos dos campos que compõem o quadro recebido, de acordo com o modo de operação. A leitura deve ocorrer após uma interrupção MRC.

### X-FIFO - FIFO de Transmissão

- Tipo : escrita
- Endereço : 00 a 1FH

Os dados a serem transmitidos precisam ser escritos na X-FIFO depois de uma interrupção BXP (modo interrupção) e/ou se o bit XFE (reg. REST) está ativo (modo pooling).

### 3.6.6 Registradores de Controle e Dados das Interfaces Seriais

#### RCPS1 - Registrador de Controle das Portas Seriais 1

- Tipo : escrita
- Endereço : 30H
- Valor Após Reset : 00H
- Formato:

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| PBZ   | CPS   | MPS   | IOM   | B1C1  | B1C0  | B2C1  | B2C0  |

- *PBZ - Coloca Porta B em Zero*  
A ativação do componente de camada 1 é feita colocando-se nível 0 na saída TDSB da Porta B. A ativação deste bit (nível 1) coloca TDSB em nível 0.
- *CPS - Condição da Linha SIP*  
A linha SIP é ativada (nível 0) ou fica em alta impedância (nível 1).
- *MPS - Modo de Timing das Portas Seriais*  
Define se o CH2 vai operar na configuração de timing de terminal (nível 0) ou configuração de timing de central (nível 1).
- *IOM - Modo da Interface IOM*  
Define se a interface IOM na Porta B opera no modo normal ou no modo multiplexado.

– B1C1, B1C0 - Chaveamento de Canal B1

Nestes bits se define qual é o caminho interno percorrido pelos dados do canal B1, de acordo com a tabela XX. Veja item 3.2.2.

| Código | Significado        |
|--------|--------------------|
| 00     | Loop na SLD        |
| 01     | Chaveamento da SLD |
| 10     | Chaveamento da SSI |
| 11     | Loop da Porta B    |

– B2C1, B2C0 - Chaveamento de Canal B2

Nestes bits se define qual é o caminho interno percorrido pelos dados do canal B2, de acordo com a tabela XX. Veja item 3.2.2.

### RCPS2 - Registrador de Controle das Portas Seriais 2

- Tipo : escrita/leitura
- Endereço : 31H
- Valor Após Reset : 00H
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| STS3  | STS2  | STS1  | STS0  | ST1   | ST0   | SC1   | SC0   |

– STS3:STS0 - Seleção de Time-Slot

No caso de IOM modo multiplexado, define qual dos 15 canais do quadro será tratado.

– ST1 - Transferência Síncrona 1

Quando ativado, faz com que uma interrupção SIN seja gerada no começo do quadro que circula pela Porta B. Veja item 3.2.3.

– ST0 - Transferência Síncrona 0

Quando ativado, faz com que uma interrupção SIN seja gerada no meio do quadro que circula pela Porta B. Veja item 3.2.3.

- *SC1 - Transferência Síncrona 1 Completa*  
Quando ativado, faz reconhecimento da interrupção síncrona 1; isso deve ocorrer antes que o centro do quadro seja atingido, de forma a evitar geração de Overflow.
- *SC0 - Transferência Síncrona 0 Completa*  
Quando ativado, faz reconhecimento da interrupção síncrona 0; isso deve ocorrer antes do início do próximo quadro, de forma a evitar a geração de Overflow.

### RXCI - Registrador de Transmissão de Canal C/I

- Tipo : escrita
- Endereço : 32H
- Valor Após Reset : F3H
- Formato :

|       |       |       |       |      |       |       |       |
|-------|-------|-------|-------|------|-------|-------|-------|
| bit 7 | bit 6 | bit 5 | bit 4 | bt 3 | bit 2 | bit 1 | bit 0 |
| C     | O     | D     | E     | DTS  | TLP   | XCT   | XCE   |

- *CODE - Código de Canal C/I*  
Carrega os códigos para sinalização de ativação/desativação e manutenção a serem transmitidos para um componente tratador de camada 1 . Esses códigos são apresentados nos documentos de padronização da interface IOM.
- *DTS - Detecção de Transição do Canal de Sinalização*  
Este bit aciona os circuitos internos que executam a detecção de transição no conteúdo do canal SIG recebido da SLD.
- *TLP - Loop de Teste*  
Quando ativado, o CH2 entra no modo de teste. Neste modo é executado um curto circuito entre os sinais RDSB E TDSB, permitindo ao usuário verificar a correção da lógica de chaveamento dos canais B, assim como os procedimentos nos canais Monitor e C/I.
- *XCT - Transmissão do Canal T*  
O conteúdo deste bit é colocado na posição 0 do octeto C/I da interface IOM durante a transmissão.
- *XCE - Transmissão do Canal E*  
O conteúdo deste bit é colocado na posição 1 do octeto C/I da interface IOM durante a transmissão.

### **RRCI - Registrador de Recepção de Canal C/I**

- Tipo : leitura
- Endereço : 32H
- Valor Após Reset : F0H
- Formato :

| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| C     | O     | D     | E     |       |       | RCT   | BVS   |

- *CODE - Código de Canal C/I Recebido*  
Contém os códigos recebidos dos componentes de camada 1 através da interface IOM.
- *RCT - Recepção de Canal T*  
Contém o bit da posição 0 do canal C/I recebido.
- *BVS - Canal B Válido na SLD*  
Indica que o processador já pode ler o canal B que chegou pela Porta C nos registradores apropriados.

### **RMON - Registrador de Tx e Rx do Canal Monitor**

- Tipo : leitura/escrita
- Endereço : 33H

### **XCB1 - Registrador de Dados para Transmissão de Canal B1**

- Tipo : leitura/escrita
- Endereço : 36H

### **XCB2 - Registrador de Dados para Transmissão de Canal B2**

- Tipo : leitura/escrita
- Endereço : 37H

**RCB1 - Registrador de Dados para Recepção de Canal B1**

- Tipo : leitura
- Endereço : 38H

**RCB2 - Registrador de Dados para Recepção de Canal B2**

- Tipo : leitura
- Endereço : 39H

**XSGS - Registrador de Transmissão do Octeto SIG da SLD**

- Tipo : escrita
- Endereço : 34H

**RS GS - Registrador de Recepção do Octeto SIG da SLD**

- Tipo : leitura
- Endereço : 34H

**RFCS - Registrador de Tx e Rx do Canal FC da SLD**

- Tipo : leitura/escrita
- Endereço : 35H

## **Capítulo Quatro**

### **Sugestões de Implementação**

## 4 Sugestões de Implementação

Neste capítulo serão apresentadas algumas sugestões para implementação do CH2 no silício. Não se tem aqui a pretensão de entrar em detalhes (até porque isso seria trabalho para um bom número de pesquisadores), mas apenas de sugerir uma metodologia para o encaminhamento do projeto. Começaremos abordando de forma sucinta a metodologia já consagrada de projeto hierárquico de sistemas em geral e dos sistemas digitais em particular. Em seguida é resumido o assunto de projeto hierárquico de CIs em VLSI, em particular os projetos *full custom* e suas condições necessárias e implicações. Esta abordagem é baseada em [COSTA87].

Com base no exposto acima é, então, apresentada uma divisão hierárquica em blocos, módulos e submódulos do circuito CH2 e, finalmente, é apresentada uma solução de circuito para o TLD, conforme idéias presentes em [EBLF] e [EBLF88].

### 4.1 Projeto Hierárquico

Uma metodologia bastante difundida de projeto é a utilização da descrição hierárquica dos sistemas e circuitos; a maior razão para isso é que com ela se consegue esconder a grande quantidade de detalhes que existe nos mesmos. Pela redução dos detalhes menos importantes para um objeto mais simples que está mais alto na hierarquia, pode-se simplificar muitas das operações de entendimento, análise e projeto.

Uma vez que muitos circuitos são por demais complicados para ser facilmente considerado em sua totalidade, um projeto completo é geralmente visto como uma coleção de componentes agregados que são então divididos em sub-agregados em uma forma recursiva e hierárquica. Um agregado que está na base da hierarquia é chamado de *folha* e aquele que está no topo da mesma é chamado de *raiz*; todos os demais são chamados *agregados de composição* porque eles compõem o corpo da hierarquia.

Uma nomenclatura possível (e que usaremos neste trabalho) para hierarquizar um projeto é mostrado na figura 34.

O problema mais difícil enfrentado pelos projetistas é aquele de seleção de uma organização hierárquica para o seu circuito. Esta organização define a maneira como o projetista irá pensar a respeito do mesmo. Se a organização errada é escolhida, ela poderá confundir o projetista e obscurecer soluções simples para o problema de projeto. O problema pode se tornar tão convoluído, que uma má hierarquia é pior que a total ausência dela.

Por outro lado, uma organização hierárquica clara torna mais fáceis todas as fases de um projeto. Se cada nível da mesma tem funcionalidade óbvia e agrega somente aqueles componentes que pertençam a aquele dado nível hierárquico, então o circuito fica mais fácil de entender. Por exemplo, com uma boa hierarquia, a simulação pode ser feita efetivamente, testando completamente cada

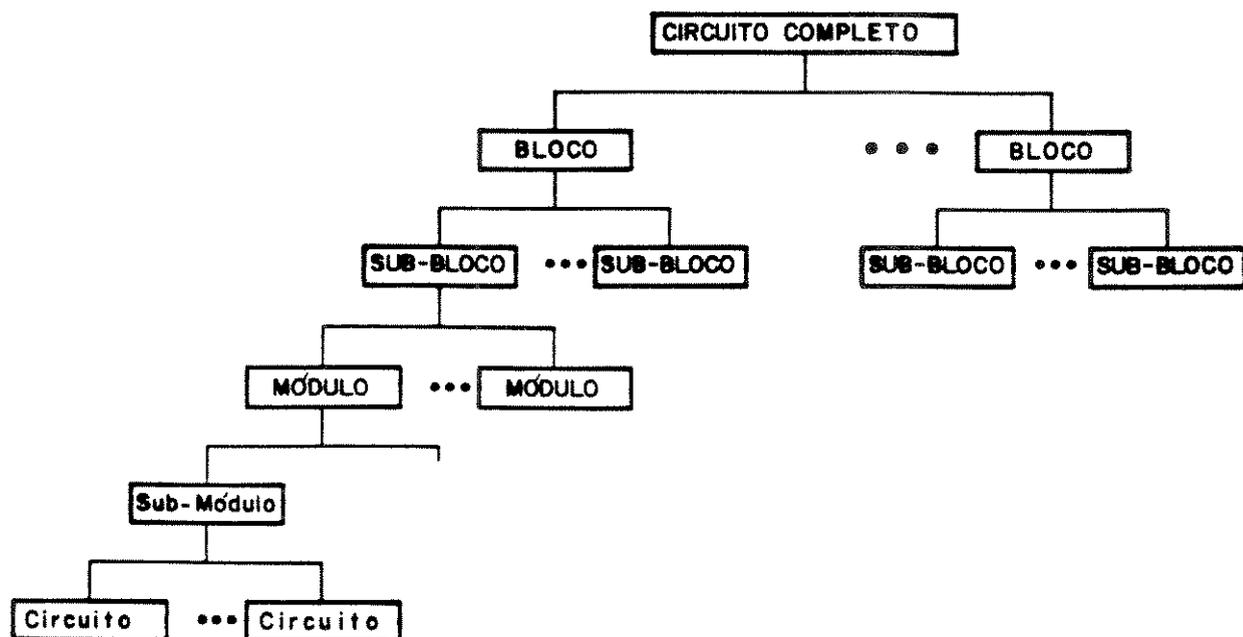


Figura 34: Projeto Hierárquico

nível hierárquico, a começar da base.

Infelizmente não há uma maneira de descrever precisamente como escolher uma boa organização hierárquica. O planejamento ideal de um projeto é uma arte mais que o próprio projeto dos circuitos.

Um aspecto a se considerar no planejamento é aquele do projeto *top-down* ou *bottom-up*. No primeiro, um circuito é projetado com detalhes sucessivamente maiores; no segundo, um circuito é visto com cada vez menos detalhes. A escolha de direção depende da natureza da tarefa, e as duas tarefas são quase sempre misturadas num projeto real. A distinção é ainda mais escurecida pelo fato de que a organização hierárquica pode ser criada com um método e então implementada na direção oposta.

Um outro importante atributo da organização hierárquica é o *fator de ramificação*. Quando a organização hierárquica é vista como uma árvore, o fator de ramificação, ao é o número médio de divisões que aparecem num dado nível da hierarquia. Um *fator de ramificação* razoavelmente baixo é importante numa boa organização hierárquica porque, se eles crescerem muito, o projetista não será mais capaz de considerar um simples nível hierárquico de uma só vez. Dado que a memória de trabalho humana é conhecido conter de cinco a nove objetos, esta faixa também é boa para o fator de ramificação.

## 4.2 Projeto Full Custom VLSI

No projeto de um circuito integrado três principais técnicas existem que diferem nos aspectos de custo, tempo de projeto, necessidade de especialização do projetista (ou, ainda, percentagem automatizada do ciclo de projeto) e desempenho final do circuito em termos de tamanho (e custo decorrente) e nível conseguido de integração. Essas técnicas são conhecidas como *gate array*, *standard cells* e *full custom*.

*Full custom* é a técnica tradicional de projeto. Cada bloco é projetado de forma a atender a especificação desejada no menor espaço físico possível. Esse projeto, dado a sua complexidade, exige um conjunto de recursos computacionais extremamente sofisticados, principalmente sistemas gráficos.

A técnica *full custom* é a que dá a maior flexibilidade ao projetista, tendo o mesmo liberdade na escolha do *placement*, roteamento, localização da pinagem, etc. É a técnica, das três, cujo resultado fornece a menor área e que permite atingir o melhor desempenho. Em muitos casos, é a única técnica recomendável. No entanto, o projeto *full custom* é caro e demorado, exigindo, ainda, projetistas especializados e competentes.

O ciclo de realização de um circuito integrado é mostrado na figura 35. a atividade é complexa, envolve várias iterações e tem realimentação em todas as etapas, da especificação ao chip final. A seguir será discutido, sucintamente, cada uma delas.

- Especificação

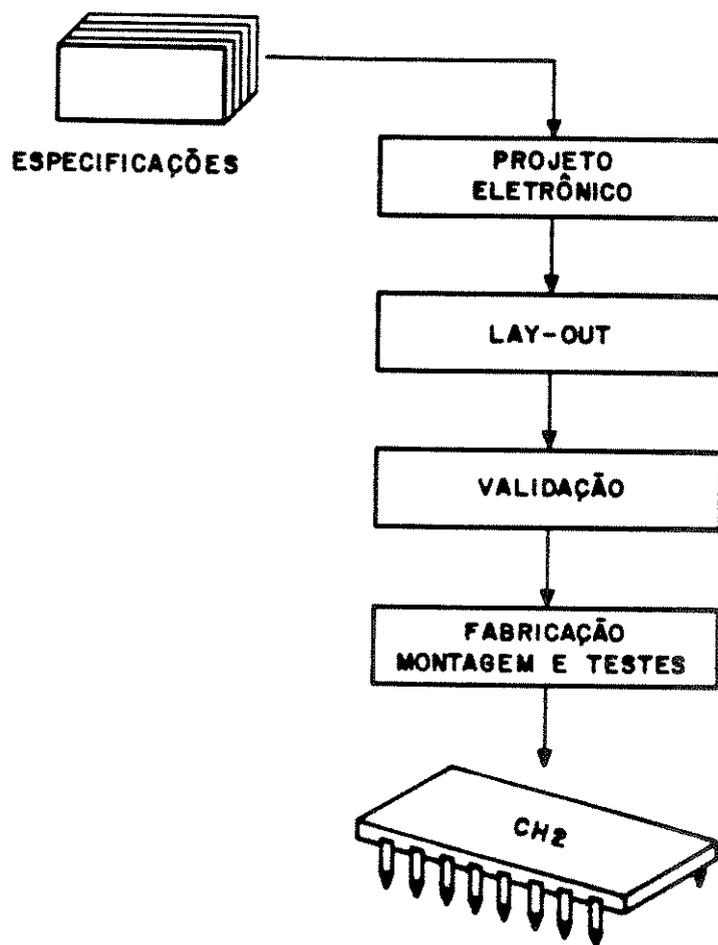


Figura 35: Ciclo de Realização de um Circuito Integrado

É, na verdade, uma fase anterior ao projeto. Diz o que se espera do circuito; traz, no mínimo, uma descrição funcional detalhada do mesmo, um diagrama em blocos e a descrição da pinagem.

- **Projeto Eletrônico**

Divide-se em três partes: projeto funcional, projeto lógico e projeto elétrico.

O projeto e simulação funcional descreve o circuito em nível de grandes blocos como registradores, ULAs, memórias, PLAs, decodificadores, etc.

O projeto e simulação lógica traz em detalhes as portas lógicas e flip flops que compõem o circuito. O projeto e simulação elétrica, normalmente feitos pra as células que foram utilizadas no projeto lógico, examina eletricamente o projeto a nível de transistores e faz a verificação dos caminhos críticos.

- **Lay Out**

Essa atividade é a abstração geométrica do projeto, onde são construídas as máscaras que serão utilizadas no processo de fabricação. É a fase mais demorada, e sujeita a muitos erros, exigindo uma fase subsequente de validação.

- **Validação**

É composta de várias etapas. A primeira delas, chamada de *extração*, verifica se a geometria corresponde ao projeto original através da extração do circuito elétrico a partir do lay out e com a comparação subsequente. São extraídos também os parâmetros reais (capacitância, dimensões finais, etc), cujos valores foram estimados durante a fase de simulações lógica e elétrica. A etapa seguinte, chamada de *Verificação de Regras de Projeto*, é onde é verificado se as regras de projeto, que são dependentes do processo de fabricação a ser utilizado, estão corretamente implementadas. A terceira etapa é a ressimulação do projeto eletrônico com os dados extraídos da etapa 1.

- **Fabricação, Montagem e Testes**

A partir das máscaras, um complexo processo de fabricação é desencadeado na chamada *foundry*, até chegar ao circuito no silício. Este circuito deverá ser montado (encapsulamento) e entrar numa outra fase, também de suma importância, que é o processo de teste de funcionamento e o processo de caracterização, cujo resultado é a folha de especificação (*data sheet*) final do circuito.

### 4.3 Projeto Lógico Hierárquico do CH2

A proposta de especificação apresentada neste capítulo já traz embutida uma divisão funcional para o circuito CH2 em quatro grandes blocos: Bloco de Interfaces, Bloco de FIFOs e seus controles, Bloco Tratador de HDLC e Bloco

Tratador de LAPD. Conforme já foi dito, além dos objetivos didáticos na apresentação, esta divisão permite a distribuição do esforço de projeto em grupos estanques de projetistas, desde que satisfeita a condição de se ter bem definidas as interfaces entre os blocos.

É claro que o conjunto de funções definidas para cada um desses blocos é ainda muito grande e, dessa forma, propomos uma divisão dos mesmos, quando for o caso, em sub-blocos, módulos, sub-módulos e até circuitos, numa estrutura de árvore em que as folhas representam a menor porção de circuito que se justifica simular separadamente. É claro que nem todos os blocos apresentam sub-blocos e nem todos os módulos apresentam sub-módulos cuja existência é justificável. Não se deve procurar divisões com o intuito único de preencher a estrutura da árvore, mas sim, quando necessário, para auxiliar a implementação. É bastante difícil chegar a uma divisão desse tipo antes de se enfrentar o desafio do projeto, isto é, antes de se *por a mão-na-massa*. No entanto, algumas considerações iniciais podem servir de premissas para o futuro implementador, e é isso que procuramos fazer neste item.

#### 4.3.1 Bloco de Interface

Parece natural, para este Bloco, que cada uma das portas seriais seja tratada separadamente e que o mesmo aconteça em relação ao contador de interrupções, à lógica de roteamento interno de canais B e à lógica de geração de sinais de relógio interno (sincronismo).

A divisão em sub-blocos parece desnecessária. Os seguintes módulos podem ser identificados [FSP]:

1. Módulo Tratador de Canal Monitor da IOM
2. Módulo Tratador do Canal C/I da IOM
3. Módulo Tratador do Canal Feature Control da SLD
4. Módulo Tratador de Canal SIG da SLD
5. Módulo de Montagem de Quadro
  - da IOM
  - da SLD
  - da Porta A
6. Módulo de Chaveamento de Canais B1 e B2
7. Módulo Tratador da Porta B como Interface HDLC
8. Módulo Tratador de Interface com o Processador
9. Módulo Tratador de Interrupções

10. Módulo Gerador de Sincronismo
11. Módulo de Comunicação com o THDLC

#### 4.3.2 Bloco THDLC

Em virtude da forma como foi definido este bloco, isto é, um auxiliar de grande porte do TLD, ele apresenta funções bastante distintas entre si. Por isso, é razoável que se definam alguns sub-blocos, agregando funções similares e/ou pertencentes a uma mesma estrutura funcional. São identificados quatro sub-blocos:

**Sub-Bloco de Recepção HDLC** Trata toda a problemática de recepção, incluindo a execução das funções a nível de bit do LAPD, desmontagem do quadro, tratamento do campo de endereço e passagem ao TLD do campo de controle de forma conveniente. Pode ser dividido em dois módulos e estes, por sua vez, em circuitos:

1. Tratador de funções de nível de bit
  - (a) Circuito de Ligação com o Bloco de Interfaces (recebimento do canal D);
  - (b) Circuito de Ligação com a R-FIFO;
  - (c) Circuito de Detecção e Extração de Flags;
  - (d) Circuito de Transparência de Bits (extração de bits 0);
  - (e) Circuito de Verificação de CRC;
  - (f) Circuito Detector de Sequência de Aborto.
2. Tratador de Quadro
  - (a) Circuito Detector de Quadro com Comprimento Incorreto;
  - (b) Circuito Detector de Campo Info onde não Permitido;
  - (c) Circuito Analisador do Campo de Controle;
  - (d) Circuito Tratador de Quadros Recebidos;
  - (e) Circuito Analisador do Campo de Endereço.

**Sub-Bloco de Transmissão HDLC** Trata toda a problemática de transmissão, incluindo a execução das funções a nível de bit do LAPD, montagem do quadro e passagem do canal D ao Bloco de Interfaces. Também pode ser dividido em dois módulos e respectivos circuitos:

1. Tratador das funções a nível de bit

- (a) Circuito de Ligação com o Bloco de Interfaces (transmissão do canal D);
- (b) Circuito de Ligação com a X-FIFO;
- (c) Circuito de Transparência de Bits (inserção de bits 0);
- (d) Circuito de Geração de CRC;
- (e) Circuito Gerador de Sequência de Aborto.

## 2. Tratador de Quadro

- (a) Circuito Tratador de Comandos do Processador p/ Tx de Quadros;
- (b) Circuito Tratador de Comandos do TLD p/ Tx de Quadros;
- (c) Circuito Montador de Quadros.

**Sub-Bloco de Controle de Variáveis** A partir de solicitações codificadas provenientes do TLD, esse circuito deve fornecer o valor de variáveis do protocolo LAPD ou fornecer o resultado de operações aritméticas e lógicas sobre suas variáveis. Esse circuito tem ainda a função de ativar, desativar, igualar ou incrementar variáveis. A tabela abaixo mostra os comandos sobre as variáveis geradas pelo TLD:

| Comando | Significado                                 |
|---------|---------------------------------------------|
| ZVVV    | zera variáveis V(A), V(S) e V(R)            |
| ZRC     | zera contador de retransmissão              |
| ZF      | zera bit F                                  |
| ZN3     | zera variável <i>iniciado pelo nível 3</i>  |
| ZER     | zera variável <i>exceção reject</i>         |
| ZCRS    | zera variável CRS                           |
| ZXRNR   | zera variável XRNR                          |
| ZRRNR   | zera variável RRNR                          |
| SN3     | ativa variável <i>iniciado pelo nível 3</i> |
| SER     | ativa variável <i>exceção reject</i>        |
| SCRS    | ativa variável CRS                          |
| SXRNR   | ativa variável XRNR                         |
| SRRNR   | ativa variável RRNR                         |
| SP      | ativa bit P                                 |
| SFP     | igualar bit F ao bit P                      |
| SVANR   | igualar variável V(A) à N(R)                |
| SVSNR   | igualar variável V(S) à N(R)                |
| IVS     | incrementa variável V(S)                    |
| IVR     | incrementa variável V(R)                    |
| IRC     | incrementa contador de retransmissão        |

A tabela a seguir apresenta as solicitações do TLD ao Sub-Bloco de controle:

|                              |
|------------------------------|
| $RC = N200$                  |
| $V(S) = V(A) ?$              |
| $N(R) = V(S) ?$              |
| $N(R) = V(A) ?$              |
| $N(S) = V(R) ?$              |
| $V(A) \leq N(R) \leq V(S) ?$ |

**Sub-Bloco de Temporizadores** Este Sub-Bloco cuida das temporizações executadas pelo CH2. No modo automático, os temporizadores T200 e T203; nos demais, o temporizador definido no Registrador RTEM e ativado pelo comando ITE. A ativação/desativação de T200 e T203 é feita através de sinais provenientes do TLD.

#### 4.3.3 Bloco de FIFOs

O Bloco de FIFOs pode ser dividido em quatro módulos:

1. Módulo de Memória R-FIFO;
2. Módulo de Memória X-FIFO;
3. Módulo de Controle R-FIFO;
4. Módulo de Controle X-FIFO.

Os módulos de controle podem ser divididos em circuitos que tratam, entre outros casos, o transbordamento (R-FIFO) ou não disponibilidade de dados (X-FIFO) e fila de espera de interrupções (R-FIFO).

#### 4.3.4 Bloco TLD

Este bloco será tratado na sub-seção seguinte e por isso não será feito aqui nenhum estudo do mesmo.

### 4.4 Projeto do Bloco TLD

Nesta seção é feita uma sugestão para a implementação do Bloco Tratador de LAPD. Ela é a mesma apresentada em [2] e [3] e foi desenvolvida por ambos os autores no Centro de Pesquisa e Desenvolvimento da Telebrás.

O que se busca é uma estrutura hardware que execute o que tradicionalmente é tarefa do software, isto é, o chip deve executar autonomamente todo o procedimento de tratamento do protocolo de camada de enlace da rede, de acordo com as recomendações do CCITT, descritas a partir da linguagem LEDES.

A solução natural para o problema seria a utilização de uma máquina de estados. Porém, em se explorando essa idéia, verificou-se que o circuito tomava proporções gigantescas, em virtude do grande número de ações e decisões que envolve cada estado do protocolo, o que desestimulou fortemente o progresso da mesma, seja pela infactibilidade dos circuitos dentro de dimensões físicas aceitáveis para o chip, seja pela complexidade da gerência e controle da máquina de estados.

Desta forma, o particionamento da solução, como será apresentada, torna mais gerenciável o problema e é aplicável a qualquer situação onde os procedimentos estejam descritos utilizando-se LEDES.

#### 4.4.1 Metodologia Básica

O método baseia-se no particionamento do circuito em três níveis hierárquicos, quais sejam:

- A - Gerenciamento do circuito,
- B - Resolução do estado em relação ao ramo e
- C - Resolução do ramo,

que podem ser visualizados na figura 36.

O circuito A deve ser capaz de armazenar em  $n$  bits, a cada instante, um dos  $P$  possíveis estados do protocolo ( $2^{n-1} \leq P \leq 2^n$ ). Os bits que determinarão cada estado são recebidos dos circuitos C que, entre outras atribuições, devem sempre fornecer o próximo estado do protocolo após a resolução do ramo. É função do circuito A a multiplexação desses sinais provenientes dos circuitos C, indicando uma possível mudança de estado. A partir desses  $n$  bits que codificam o estado, o circuito A é capaz de habilitar um único conjunto de circuitos BC, inibindo todos os demais.

Uma vez habilitado, o circuito B, que é uma máquina de estados, têm como função receber um conjunto de entradas (codificadas e geradas a partir da ocorrência de um evento) que definem dentro do estado selecionado em A qual o ramo da descrição LEDES que deve seguir. Esta máquina terá dois estados: um estado 0, para o qual ela vai após a ocorrência de um reset e onde fica esperando a chegada de entradas que designem algum ramo a seguir, e um estado 1, onde são ativados sinais que avisem que o endereço à saída da máquina está válido e que permitirão a operação de um contador na parte C. As saídas deste circuito são, portanto:

- uma via de endereço : END

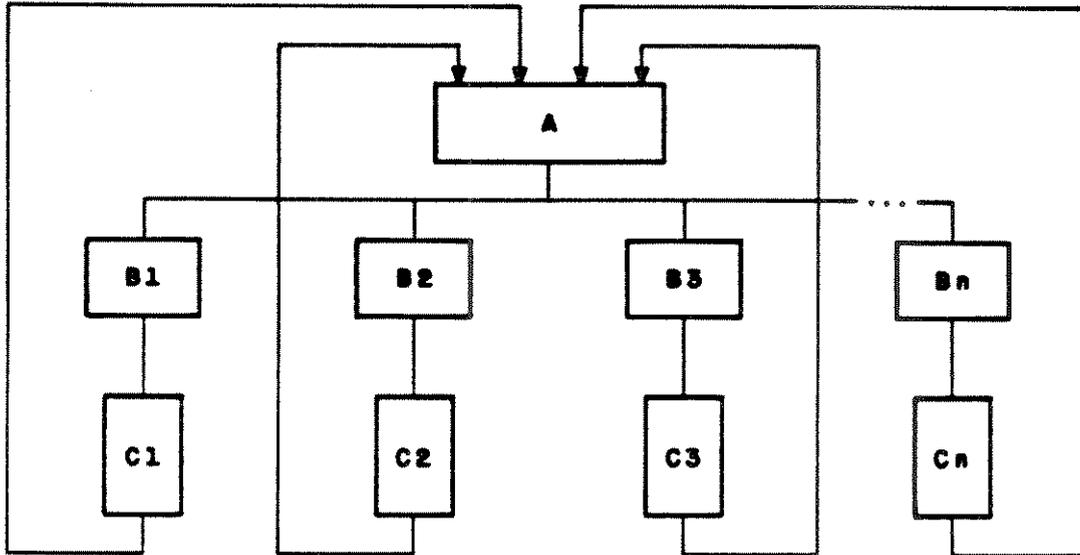


Figura 36: Particionamento do Circuito

- um sinal de *load* : L
- um sinal de habilitação de relógio : HR

Finalmente, o circuito C deve ser entendido como um contador, um registro de instruções e uma máquina de estados.

Este contador é carregado com a via END a partir do sinal L. Ao mesmo tempo, o sinal HR é ativado, habilitando, assim, o funcionamento normal do contador. A saída deste contador serve como um ponteiro para o registro de instruções que, por sua vez, deve executar as funções pertinentes ao ramo escolhido. A saída deste registro deve conter  $m + n + 1$  bits, dos quais  $m$  devem ativar circuitos que executam as tarefas descritas no ramo do LEDS,  $n$  bits irão realimentar o circuito A provendo a via de mudança de estado e a saída restante é um sinal de sincronismo para atualização de estado. A ocorrência deste sinal também determina o fim da execução do ramo LEDS.

Podemos dizer que, para cada estado diferente do protocolo, um conjunto definido de *eventos* pode ocorrer, que implicam em um conjunto ordenado de ações a serem executadas. Assim, são consideradas instruções :

- chegada de um determinado quadro na linha;
- aviso de que determinada condição foi atingida (por exemplo, erro);
- aviso proveniente de entidade externa (processador); e outros.

Note-se que um mesmo evento implica em procedimentos diferentes para serem executados, em função do estado em que se encontra o protocolo. O registro de instruções desta implementação é, na verdade, o reservatório de microações sequenciadas que formam a fase de execução dos diferentes eventos/estados.

Naturalmente que é muito comum a ocorrência de sub-ramos, gerados a partir de pontos de teste nas descrições LEDS. Esses casos são resolvidos fazendo-se uso de uma segunda máquina de estados, chamada ME de desvios, que, a partir das entradas a serem testadas e de um sinal habilitador desses testes (que é uma saída gerada no registro de instruções), deve fornecer o novo endereço a ser carregado no contador e a partir do qual está, no registro de instruções, a resolução do sub-ramo. Tal máquina deve fornecer, também, um sinal habilitador para carregamento do novo endereço no contador (LOAD) e um outro que será usado para selecionar um circuito multiplexador de modo que as linhas de endereço fornecidas por esta máquina sejam as que apareçam na entrada do contador.

A metodologia de se dividir em três partes a implementação, em hardware, de algoritmos descritos em forma de LEDS é particularmente interessante quando o número de operações e estados é grande, já que facilita o gerenciamento dos mesmos.

No Bloco TLD o número de grandes estados onde o circuito poderá se encontrar é quatro, além da fase não operacional. Desta forma, a codificação poderá ser feita de acordo com a tabela abaixo:

| Código | Estado                     |
|--------|----------------------------|
| 1 0 0  | TEI Designado              |
| 1 0 1  | Espera Estabelecimento     |
| 1 1 0  | Espera Desconexão          |
| 1 1 1  | Multi-Quadros Estabelecido |
| 0 x x  | Fase não Operacional       |

O esquema da figura 37 apresenta uma idéia geral do circuito nesta metodologia.

#### 4.4.2 Arquitetura Proposta ao TLD

Dentro da visão hierárquica de projeto, pode-se dividir o Bloco Tratador de LAPD em dois módulos:

1. Módulo Receptor (MR);
2. Módulo Processador (MP).

O Módulo Receptor é aquele que recebe mensagem das várias origens e as repassa ao Módulo Processador, que é o módulo que irá tratá-los.

**Interfaces** Os dois módulos MR e MP fazem interface entre si e com os demais blocos do CH2.

- Interface entre MR e MP
  1. Sentido MR  $\rightarrow$  MP
 

Um barramento de seis linhas : F1 F2 C1 C2 C3 C4.

Os bits F indicam qual é a origem dos dados e os bits C indicam qual mensagem foi recebida desta fonte.
  2. Sentido MP  $\rightarrow$  MR
    - ZRP - sinal para indicar a saída do TLD na condição Reconhecimento Pendente.
    - SRP - sinal para indicar a entrada do TLD na condição Reconhecimento Pendente.
    - RES - sinal de reset provido pelo Registro de Instruções do MP.
- Interface com THDLC



1. MR recebe informações de expiração dos temporizadores, de que quadro I foi transmitido e de códigos de mensagens recebidas da entidade par.
  2. MP recebe do THDLC informações a respeito do estado de variáveis do protocolo que precisam ser testadas e envia mensagem para que o transmissor HDLC monte e transmita quadros à entidade remota ou atualize variáveis internas.
- Interface com o Processador  
O MR recebe, armazena e repassa comandos enviados pelo processador.
  - Interface com o Bloco de Interfaces  
O MP envia ao BI dados referentes a pedidos de interrupção.

**Módulo Receptor** Os submódulos a seguir compõem o Módulo Receptor:

### 1. SVRD - Submódulo de varredura

Este sub-módulo é composto por um contador de 3 bits que excita um demultiplexador de 3 para 8. As 8 linhas de saída deste submódulo compõem o barramento de varredura denominado NHT. Um sinal gerado pelo circuito CIVA inibe o funcionamento deste submódulo.

#### (a) CIVA - Circuito inibição de varredura

Este circuito usa como entrada os bits C gerados por SREG e que fazem a interface MR - MP. Se estes bits estiverem todos desativados, significa que não houve nenhuma mensagem e o CIVA fica permitindo o prosseguimento da varredura. Caso uma mensagem válida seja recebida em SREG, CIVA inibe a varredura, que somente será reativada com o sinal RES proveniente do submódulo SRIN do MP. Se uma mensagem inválida vier a ser recebida em SREG, todo o processo descrito acima repetir-se-á. Assim, o Registro de Instruções em MP deverá executar uma instrução com a finalidade única de ativar o sinal RES.

### 2. SREC - Submódulo de recepção

O SREC é composto de registradores que devem armazenar mensagens oriundas de fontes distintas para o TLD.

- (a) LAPD-REC: O TLD envia mensagem para si mesmo. Essa mensagem tem o nome de Reconhecimento Pendente e é gerada pelo protocolo sempre que um quadro é recebido de tal forma a não exigir reconhecimento imediato (bit P = 0). O mnemônico para esta mensagem é SRP; quando, por algum dos motivos previstos no protocolo,

esse reconhecimento é provido, o módulo MP gera o comando ZRP que desativa esta condição.

Uma vez que este comando esteja ativado, ele tem a mais baixa prioridade de execução. Assim, ele só será executado se não houver mais nenhum comando ativado em qualquer dos outros registradores, de modo a garantir o correto funcionamento do protocolo nesta condição.

A leitura deste submódulo é habilitada por NHT0.

- (b) HDLC-REC: Recebe os seguintes comandos do Bloco THDLC:
- i. QIT - Quadro I Transmitido - leitura habilitada por NHT1.
  - ii. ET200 - Expiração do Temporizador T200 - leitura habilitada por NHT2.
  - iii. ET203 - Expiração do Temporizador T203 - leitura habilitada por NHT3.
- (c) PROC - REC : Recebe os seguintes comandos do processador:
- i. ECO - pedido de estabelecimento de conexão - leitura habilitada por NHT4.
  - ii. DCO - pedido de desativação de conexão - leitura habilitada por NHT5.
  - iii. RNR - aviso de processador local ocupado - leitura habilitada por NHT6.
  - iv. RR - aviso de processador local livre - leitura habilitada por NHT6.
- (d) PAR - REC : Recebe comandos do THDLC, referentes a quadros vindos da entidade par (canal D). Esses quadros são decodificados pelo Bloco THDLC e enviados de forma codificada, de acordo com a tabela abaixo:

| Quadro    | CC1 | CC2 | C3 | CC4 |
|-----------|-----|-----|----|-----|
| RR        | 0   | 0   | 0  | 0   |
| DISC      | 0   | 0   | 0  | 1   |
| RNR       | 0   | 0   | 1  | 0   |
| FRMR      | 0   | 0   | 1  | 1   |
| REJ       | 0   | 1   | 0  | 0   |
| DM        | 0   | 1   | 1  | 1   |
| UA        | 1   | 0   | 0  | 1   |
| I LIVRE   | 1   | 0   | 1  | 0   |
| I OCUPADO | 1   | 0   | 1  | 1   |
| SABME     | 1   | 1   | 1  | 0   |

Essa codificação procura aproveitar a ordem de bits no campo de controle dos quadros correspondentes.

Quando um quadro recebido vier com erro de protocolo (EP), o código enviado ao TLD será 1000.

O registrador PAR-REC tem quatro bits e armazena um novo dado sempre que houver transição no sinal FMLR (Fim de Mensagem de LAPD Recebida) gerado pelo bloco THDLC. A leitura deste registrador é habilitada pelo sinal NHT7.

Todos esses registradores estão ligados a um barramento CD de 4 bits que vai ser entregue ao submódulo SREG.

### 3. SIOR - Submódulo identificador de origem de mensagem

É um submódulo que com o barramento NHT à sua entrada, identifica qual a origem da mensagem. Como existem quatro origens distintas, dois bits (F1 e F2) são necessários para esta identificação.

### 4. SREG - Armazenador de dados para o MP

Este submódulo transfere dados do MR para o MP e espera um reset vindo do MP para voltar ao estado inicial e aguardar novos dados.

**Módulo Processador** Este módulo é um circuito microprogramado que usa como elemento armazenador um Registro de Instruções (submódulo SRIN) que nada mais é do que um PLA ou uma ROM. Esse submódulo, portanto, deve ser entendido como o núcleo do MP e que tem os outros submódulos como periféricos.

#### 1. SEND - Submódulo gerador de endereços - Circuito B

É uma máquina de estados que recebe o barramento do MR e dois bits do submódulo SESP. Esses dois bits determinam qual é o estado atual do protocolo LAPD e o barramento traz qual a mensagem que está sendo recebida. Deste modo, uma mensagem que é válida para, por exemplo, o estado Multiquadros Estabelecido, não necessariamente o será para o estado TEI Designado. À saída, essa máquina de estados apresenta o barramento END de 10 bits e uma linha de *load* - LA. Toda vez que uma mensagem válida é processada pelo SEND, um endereço aparece no barramento END e LA fica ativo.

Deste modo, após receber uma mensagem válida, o SEND aponta para uma determinada posição do SRIN a partir da qual está programado como essa mensagem deve ser processada.

#### 2. SESP - Submódulo armazenador do estado atual do protocolo - Circuito A

Este submódulo é um registrador de dois bits, pois são somente quatro os estados do protocolo. Sempre que é terminado o processamento de uma mensagem, o SRIN precisa fornecer qual é o próximo estado para o qual deve ir o sistema. O SRIN também fornece um pulso para que esse estado seja carregado em SESP. Assim, com o estado atualizado, o MP fica esperando uma nova mensagem válida.

### 3. SMXP - Submódulo Multiplex-Apontador

Recebe barramentos de 10 bits de vários submódulos:

- (a) SEND
- (b) SGER
- (c) SJBS
- (d) SSOM

Todos esses submódulos, portanto, podem endereçar o SRIN. O multiplex contido em SMXP deve, controlado por CCTR, prover este roteamento.

Uma vez escolhido qual é o barramento que vai endereçar o SRIN, é provida a sua carga num contador, que será o apontador de SRIN. O circuito de reset, CRES, limpa este contador, mantendo o SRIN no endereço 0 enquanto não vem uma nova mensagem.

#### (a) CCTR - Circuito controle de SMXP

É o circuito que controla o multiplex de SMXP e o carregamento do apontador. Para tanto ele recebe sinais dos seguintes submódulos periféricos:

- i. SEND - envia o sinal LA
- ii. SGPA - envia o sinal LC
- iii. SRIN - envia o sinal RET
- iv. SJSB - envia o sinal LB

O circuito combinacional que compõem CCTR utiliza para controle do multiplexador apenas os sinais enviados pelos três últimos submódulos citados acima, da seguinte forma:

| RET | LC | LB | Submódulo que acessa o apontador |
|-----|----|----|----------------------------------|
| 0   | 0  | 0  | SSOM                             |
| 0   | 0  | 1  | SSOM                             |
| 0   | 1  | 0  | SGER                             |
| 0   | 1  | 1  | SGER                             |
| 1   | 0  | 0  | SSOM                             |
| 1   | 0  | 1  | SSOM                             |
| 1   | 1  | 0  | SJSB                             |
| 1   | 1  | 1  | SEND                             |

#### 4. SRIN - Submódulo registro de instruções

É um arranjo PLA ou uma ROM que possui 10 entradas e 18 saídas. Os sinais de entrada endereçam o início de tratamento de um dado evento em um dado estado. As saídas tem a configuração mostrada abaixo:

| bit 17 | bit 16 | bit 15 | bit 14 | bit 13 | bit 12 | bit 11 | bit 10 |
|--------|--------|--------|--------|--------|--------|--------|--------|
| RES    | CI3    | CI2    | CI1    | ER4    | ER3    | ER2    | ER1    |

| bit 9 | bit 8 | bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| G6    | G5    | G4    | G3    | G2    | G1    | G0    | RET   | EF1   | EF2   |

onde

- (a) RES - é o sinal de reset gerado por instrução do RI;
- (b) CI3 a CI1 - sinais usados para indicar interrupção do tipo indicação/confirmação;
- (c) ER4 a ER1 - sinais usados para gerar interrupções de erro;
- (d) G6 a G0 - barramento que codifica instruções de operação do protocolo;
- (e) RET - é o sinal de *return* que finaliza cada subrotina gravada no RI;
- (f) EF1 e EF0 - determinam qual é o estado futuro do protocolo.

Uma descrição melhor deste submódulo está no próximo item desta parte.

##### (a) CRES - Circuito de Reset

Provê o reset geral do TLD. Recebe um sinal de sincronismo externo (SE) gerado pelo circuito CGSE, um sinal RES de SRIN e uma das variáveis de estado do submódulo SEND. Se SE for igual a 1, ficam zeradas as máquinas de estado do submódulo SEND e do submódulo

SJSB, além do apontador do SRIN. Com  $SE = 0$ , o apontador do SRIN fica em zero sob o comando de uma variável de estado da máquina do submódulo SEND ou sob comando do reset do SRIN. Ainda com  $SE = 0$ , as máquinas de estado ficam zeradas pelo reset de SRIN. A última função que este circuito implementa é o reset da máquina de estado do submódulo SJSB quando o SRIN executa uma instrução de RETURN.

(b) **CGSE - Circuito gerenciador de sincronismo externo**

Assim que o processador escreve o bit  $FNO = 1$  no Registrador de Comando 2, essa informação é transferida para este circuito, no sentido de tirar o TLD da fase operacional, isto é, fazer  $SE = 1$ . Assim que se desejar reiniciar o funcionamento do TLD, deve-se escrever  $TED = 1$  em CMDR2. Este procedimento faz com que SE volte a 0, posicionando o TLD no estado TEI Designado, que é o estado inicial do tratamento interno do protocolo.

(c) **CDEC - Circuito decodificador**

Recebe os bits G6 a G0 do SRIN e os decodifica.

(d) **CAIE - Circuito armazenador de interrupções de erro**

Este circuito recebe o barramento do SRIN, os sinais ER4 a ER1 e CI3 a CI1 para envio a Registrador EIR2 do Bloco de Interfaces.

O circuito CAIE ainda recebe os bits indicadores de erros L, M e N do THDLC. Esses bits precisam ser multiplexados com ER4-1 vindos do SRIN, com prioridade para esses últimos.

5. **SGER - Submódulo gerador de endereço de retorno**

O submódulo SRIN pode, sob certas circunstâncias, chamar uma subrotina para efetuar procedimentos padrões dentro do protocolo LAPD. Após a execução desta subrotina, o apontador do programa deve voltar ao endereço imediatamente subsequente àquele que a chamou. O SGER, então, provê todo esse procedimento. Para tanto ele armazena o valor do apontador no momento da chamada de subrotina, incrementa esse valor e espera a execução da instrução RETURN que está no fim de toda subrotina. Quando esta instrução é executada, o sinal RET é ativado, provendo a carga do novo apontador.

6. **SJSB - Submódulo gerenciador de desvios e subrotinas**

Toda vez que um desvio incondicional ou subrotina precisa ser ativado, esta máquina de estados entra em operação. A partir do momento que ela recebe dados do SRIN e identifica a situação de desvio incondicional ou subrotina, um endereço apropriado é colocado no seu barramento de saída concomitantemente à ativação do sinal de *load* - LB. O procedimento de carga desse novo endereço é semelhante ao descrito no submódulo

SEND. Ao receber uma instrução de retorno, indicando fim de subrotina, ou um reset, indicando fim de desvio, esta máquina de estados volta à sua condição original através de um reset do circuito CRES.

#### 7. SGPA - Submódulo gerador de passos

Algumas decisões são tomadas em função dos valores de variáveis do protocolo LAPD; este submódulo é responsável pela execução de testes sobre as mesmas. Como o LAPD testa variáveis externas enviadas pelo THDLC, assim que uma instrução de teste é executada, o circuito combinacional que está neste bloco gera um número binário menor que 32 que é o passo que deve ser adicionado ao apontador do SRIN, dependendo do resultado do teste. Este submódulo gera também um sinal de *load* - LC para a carga deste apontador em SMXP.

##### (a) CSOM - Circuito somador

Soma o passo gerado em SGPA ao ponteiro do SRIN que habilitou o teste. É, portanto, apenas um somador de uma palavra de 10 bits com uma palavra de 5 bits. A saída é um barramento de 10 bits que é entregue ao submódulo SMXP.

A figura 38 mostra a interligação entre esses módulos.

#### 4.4.3 Programação da PLA que implementa o LAPD

A partir dos LEDS que descrevem o protocolo a ser implementado, é definido um conjunto de instruções, permitindo escrever um programa que execute os procedimentos deste protocolo. A cada instrução é associado um mnemônico.

Este conjunto de instruções foi dividido em vários grupos, de acordo com a função básica de cada um. Tais grupos são:

- desativa bits/variáveis: mnemônico iniciado pela letra Z;
- inicia bits/variáveis: mnemônico iniciado pela letra S;
- envia comando: mnemônico iniciado pela letra E;
- habilita testes: mnemônico iniciado pela letra H;
- incrementa variável: mnemônico iniciado pela letra I;
- controle de temporizadores: mnemônico iniciado pela letra T;
- interrupções : mnemônico iniciado pela letra P;
- desvios: mnemônico iniciado pela letra J;
- subrotinas: mnemônico iniciado pelas letras LC;



- estados futuros : mnemônico iniciado pela letra F;
- sinal de reset;
- sinal de return.

O conjunto completo de instruções é mostrado na tabelas a seguir.

|       |                                  |
|-------|----------------------------------|
| ZRC   | zera contador RC                 |
| ZCRS  | zera variável CRS                |
| ZVVV  | zera variáveis V(A), V(S) e V(R) |
| ZN3   | zera variável N3I                |
| ZF    | zera bit F                       |
| ZRP   | zera variável RP                 |
| ZXRNR | zera bit XRNR                    |
| ZRRNR | zera bit RRNR                    |
| ZER   | zera variável ER                 |

|       |                                              |
|-------|----------------------------------------------|
| HN3   | habilita teste de variável N3I               |
| HVSVA | habilita teste de $V(S) = V(A)$              |
| HF    | habilita teste de bit F = 1                  |
| HCRS  | habilita teste de variável CRS = 1           |
| HRNR  | habilita teste de bit RRNR = 1               |
| HCMD  | habilita teste de verif. comando/resposta    |
| HP    | habilita teste de bit P = 1                  |
| HD3   | habilita teste de $V(A) \leq N(R) \leq V(S)$ |
| HNRVS | habilita teste de $N(R) = V(S)$              |
| HNRVA | habilita teste de $N(R) = V(A)$              |
| HNSVR | habilita teste de $N(S) = V(R)$              |
| HER   | habilita teste de variável ER                |
| HRC   | habilita teste de contador RC estourado      |
| HXRNR | habilita teste de variável XRNR = 1          |

|       |                                  |
|-------|----------------------------------|
| ESAB  | pedido de envio de comando SABME |
| EDISC | pedido de envio de comando DISC  |
| EUA   | pedido de envio de resposta UA   |
| ERRR  | pedido de envio de resposta RR   |
| ERNRR | pedido de envio de resposta RNR  |
| ERRC  | pedido de envio de comando RR    |
| ERNRC | pedido de envio de comando RNR   |
| EREJ  | pedido de envio de resposta REJ  |
| EDM   | pedido de envio de resposta DM   |

|       |                        |
|-------|------------------------|
| SN3   | ativa variável N3I = 1 |
| SFP   | força bit F = P        |
| SXRNR | ativa bit XRNR = 1     |
| SVANR | força V(A) = N(R)      |
| SP    | ativa bit P = 1        |
| SRRNR | ativa bit RRNR = 1     |
| SRP   | ativa variável RP      |
| SER   | ativa variável ER      |
| SCRS  | ativa variável CRS     |
| SVSNR | força V(S) = N(R)      |

|      |                          |
|------|--------------------------|
| TIT0 | inicia temporizador T200 |
| TPT0 | pára temporizador T200   |
| TIT3 | inicia temporizador T203 |
| TPT3 | pára temporizador T203   |

|     |                        |
|-----|------------------------|
| IVR | incrementa V(R)        |
| IVS | incrementa V(S)        |
| IRC | incrementa contador RC |

|             |                                          |
|-------------|------------------------------------------|
| PVAC        | pedido de interrupção VAC                |
| PRSC        | pedido de interrupção RSC                |
| PXMR        | pedido de interrupção XMR                |
| PTIN        | pedido de interrupção TIN                |
| PCI3-1=100  | pedido de int. confirmação de desconexão |
| PCI3-1=101  | pedido de int. confirmação de conexão    |
| PCI3-1=110  | pedido de int. indicação de desconexão   |
| PCI3-1=111  | pedido de int. indicação de conexão      |
| PER4-1=0001 | pedido de int. indicação de erro A       |
| PER4-1=0010 | pedido de int. indicação de erro B       |
| PER4-1=0011 | pedido de int. indicação de erro C       |
| PER4-1=0100 | pedido de int. indicação de erro D       |
| PER4-1=0101 | pedido de int. indicação de erro E       |
| PER4-1=0110 | pedido de int. indicação de erro F       |
| PER4-1=0111 | pedido de int. indicação de erro G       |
| PER4-1=1000 | pedido de int. indicação de erro H       |
| PER4-1=1001 | pedido de int. indicação de erro I       |
| PER4-1=1010 | pedido de int. indicação de erro J       |
| PER4-1=1011 | pedido de int. indicação de erro K       |

|     |                                 |
|-----|---------------------------------|
| FTD | ativa próximo estado igual a TD |
| FEE | ativa próximo estado igual a EE |
| FED | ativa próximo estado igual a ED |
| FMQ | ativa próximo estado igual a MQ |

|       |                                      |
|-------|--------------------------------------|
| JUMPx | tratamento de desvios incondicionais |
|-------|--------------------------------------|

*Nota:* Os desvios incondicionais existentes devem ser definidos no momento da programação.

|       |                                 |
|-------|---------------------------------|
| LCRxx | libera chamada de sub-rotina xx |
|-------|---------------------------------|

*Nota:* As sub-rotinas existentes devem ser definidas no momento da programação.

|     |                                |
|-----|--------------------------------|
| RET | sinal de retorno de sub-rotina |
| RES | sinal de reset                 |

Um programa pode ser escrito com tais instruções. É possível verificar que cerca de 600 posições são suficientes para que todos os ramos sejam resolvidos, de maneira que o SRIN precisa de uma entrada com 10 bits.

## Capítulo Cinco

### Considerações de Software

## 5 Considerações de Software

Conforme descrito na seção 3.1.4 deste trabalho, várias são as aplicações previstas para o CH2. Para cada uma dessas aplicações existem decisões de programação a serem tomadas, sendo as mais importantes aquelas contidas no Registrador de Modo.

Este capítulo tem por objetivo discutir dois aspectos de software relacionados ao uso do CH2:

- Quais registradores são importantes para uma dada aplicação e o que deve ser programado. Isso é feito na seção 5.1;
- A criação de um processo *driver* para enlace de dados usando o CH2. Isso é feito na seção 5.2.

### 5.1 Programação do CH2

A seção 3.1.4 deste trabalho já antecipa as principais decisões de programação do chip, todas contidas no Registrador de Modo: como operará a Porta B, qual é o modo de operação do controlador HDLC interno e, finalmente, caso a Porta B tenha sido programada como interface IOM, qual canal deve ser tratado automaticamente pelo TLD. Este registrador deve ser o primeiro a ser programado. Importante também é o fato de que, independente da programação, os registradores RIRF e RERC trazem informações a respeito da recepção de dados na R-FIFO e devem ser sempre considerados.

#### 5.1.1 Porta B como interface IOM no Terminal

A programação deste modo depende da configuração física do terminal na rede. Desta forma:

- Se apenas um terminal está conectado no barramento S, isto é, se não há disputa de canal D e o canal Monitor não for usado, HMD2:HMD0 = 000;
- Se o CH2 está ligado diretamente a um chip de camada 1 da interface U (isto é, não existe barramento S), HMD2:HMD0 = 000;
- Se o canal Monitor não for usado para disputa de canal D, isto é, configuração ponto a ponto, e se o mesmo for usado para transmitir dados ou comandos, HMD2:HMD0 = 010;
- Se o canal Monitor for usado para disputa de canal D numa configuração ponto a multiponto, HMD2:HMD0 = 001.

Qualquer um dos modos de operação poderá ser programado. Independente de que modo for escolhido, a programação dos registradores de controle e dados das interfaces seriais, deve ser a descrita a seguir.

- *RCPS1*: O bit PBZ deve ser ativado quando se quiser colocar em operação o componente de camada 1. O bit CPS define se a interface SLD vai ou não entrar em operação. O bit MPS deve ser programado em configuração de timing de terminal. O bit IOM deve ser programado para operação no modo normal. Os demais bits definem o caminho interno percorrido pelos dados dos canais B1 e B2 e sua programação depende da configuração física do chip na placa.
- *RCPS2*: Os bits STS3:STS0 são irrelevantes. Os demais bits devem ser programados de acordo com o tipo de acesso aos canais B pelo processador, conforme o item 3.2.3.
- *Demais*: Devem ou não ser programados de acordo com sua utilização.

**Programado no Modo Automático** Neste caso, o bit MEM não tem valor (sempre são usados dois octetos) e os bits SCL1 e SCL0 definem o canal a ser tratado pelo TLD. As informações fornecidas pelo registrador REX2 são, basicamente, as primitivas a serem passadas à camada de gerência e ao processo de nível 3. As demais interrupções exigem tratamento no próprio processo *handler*.

- *RCMD2*: O comando ITE não deve ser gerado; neste modo a temporização é interna e independe de comando do processador. O comando RAC deve sempre ser gerado após o comando RHR (*RCMD1*). O comando DCO deve ser gerado após o recebimento de uma primitiva de solicitação de desconexão de enlace enviado pelo processo de nível 3. O comando ECO deve ser gerado após o recebimento de uma primitiva de solicitação de estabelecimento de conexão enviada pelo processo de nível 3. O comando TED deve ser gerado a partir do momento em que os procedimentos de designação do TEI foram executados pelo processo *handler*. O comando FNO deve ser gerado em situações em que deseja-se a não operação do TLD, tais como, por exemplo, em alterações do valor do TEI; também um reset do TLD ocorre através do comando FNO seguido do comando TED.
- *RTEM*: Deve ser programado para definir os parâmetros T200 e N200 do LAPD.
- *RTJA*: Este registrador deve ser programado caso se deseje o funcionamento interno do parâmetro T203 ou no caso de se desejar janela maior que 1 (parâmetro K).
- *RNUR e XNUR*: Estes registradores carregam respectivamente os valores dos parâmetros V(R) e V(S) do protocolo LAPD. Toda vez que uma interrupção do tipo MVA vier a ocorrer, significando que novos quadros foram reconhecidos, o processo *handler* deve ler XNUR e RNUR pra decidir quantos novos quadros foram reconhecidos e liberar os buffers da fila de retransmissão.

- *ENX1 e ENX2*: Devem ser programados com os valores de SAPI e TEI a serem colocados no campo de endereço dos quadros transmitidos.
- *SAPI1*: Deve ser programado com o valor de SAPI que se deseja os quadros recebidos tenham, de forma a serem aceitos e tratados internamente pelo TLD. O bit CRI deve ser programado para definir se o CH2 está operando do lado da central ou do lado do terminal.
- *SAPI2*: Deve ser programado com o outro valor de SAPI que se permite aceitar quadros que chegam pela interface; note-se que os quadros que contiverem este valor no campo SAPI serão aceitos mas não tratados internamente no TLD, sendo repassados via R-FIFO ao processador.
- *TEI1*: Deve ser programado com o valor de TEI que se deseja os quadros recebidos tenham, de forma a serem aceitos e tratados internamente pelo TLD.
- *TEI2*: Deve ser programado com o outro valor de TEI que se permite aceitar quadros que chegam pela interface; note-se que, também neste caso, os quadros que contiverem este valor no campo TEI serão aceitos mas não tratados internamente no TLD, sendo repassados via R-FIFO ao processador.
- *RHCR1 e RHCR2*: São registradores auxiliares, que podem ser utilizados numa fase de teste; em operação normal pode ser irrelevante saber que quadros estão sendo recebidos, uma vez que eles são tratados internamente; no entanto, após uma interrupção RMT, RHCR2 e o registrador XNUR devem ser lidos para que o processo *handler* possa determinar quais quadros deverão ser retransmitidos.

**Programado nos demais modos** As interrupções MER(REIN), EJA e MVA (REX1) e todo o registrador REX2 não precisam se considerados pois não irão ocorrer.

- *REST*: Apenas os bits 7 e 6 contém informações relevantes.
- *RCMD1*: Os comandos XQUI, RNR, RR e XQI não podem ser gerados.
- *RCMD2*: Os comandos FNO, TED, ECO E DCO não podem ser gerados.
- *RTEM*: Deve ser programado se se fizer uso desta facilidade do CH2; os bits 7 a 5 (parâmetros N200) não são relevantes.
- *RTJA*: Irrelevante nestes modos.
- *RNUR*: Irrelevante nestes modos.
- *XNUR*: Irrelevante nestes modos.

- *ENX1*: Irrelevante nestes modos.
- *ENX2*: Irrelevante nestes modos.
- *RHCR1* e *RHCR2*: São registradores auxiliares; podem ser ignorados ou ter a informação utilizada para eventuais testes.
- *SAPI1*, *SAPI2*, *TEI1*, *TEI2*: Devem ser programados, dependendo do modo, se forem ou não de interesse; o bit *CRI* de *SAPI1* é irrelevante.

### 5.1.2 Porta B como interface IOM na Central

Da mesma forma que no caso do Equipamento Terminal, qualquer modo de operação pode ser programado. O *time-slot* do quadro IOM Multiplexado que será tratado internamente deve ser definido no registrador *RCPS2* e em *RCPS1* o modo IOM Multiplexado e a configuração de timing de central devem ser programados. Os registradores ligados aos Blocos *THDLC* e *TLD* têm programação idêntica ao modo terminal.

### 5.1.3 Porta B como interface HDLC

Uma vez definido que a Porta B irá operar como interface HDLC, os bits *SCL1* e *SCL0* do Registrador de Modo ficam irrelevantes, os bits *SMD1* e *SMD0* não podem assumir o valor 00 (modo automático) e o bit *MEM* deve ser programado adequadamente.

A seguir são feitas considerações a respeito dos demais registradores.

- *REIN*: As interrupções *SIN*, *MER*, *MC1* e *EI2* não devem ocorrer e não precisam ser tratadas;
- *REX1*: As interrupções *RMT*, *TTS*, *MOR*, *TCS*, *EJA* e *MVA* não devem ocorrer e não precisam ser tratadas;
- *REX2*: Não devem ocorrer interrupções deste registrador, que não precisa ser tratado;
- *REST*: Os bits *XRNR*, *RRNR*, *RER* e *RC3:1* não carregam informações de interesse neste modo e não devem ser considerados;
- *RCMD1*: Os comandos *XQUI*, *RNR*, *RR* e *XQI* não podem ser usados neste modo;
- *RCMD2*: Os comandos *FNO*, *TED*, *ECO* e *DCO* não podem ser usados neste modo;
- *RTEM*: Os bits *N23* a *N21* não tem interesse neste modo; os bits *TR5* a *TR1* devem ser programados com o valor do temporizador desejado, caso se faça uso desta facilidade do chip;

- *RTJA*: Este registrador não deve ser programado neste modo;
- *RNUR*: Este registrador não deve ser programado neste modo;
- *XNUR*: Este registrador não deve ser programado neste modo;
- *ENX1*: Este registrador não deve ser programado neste modo;
- *ENX2*: Este registrador não deve ser programado neste modo;
- *SAPI1*: Deve ser programado de acordo com o modo escolhido; o bit CRI é irrelevante;
- *SAPI2*: Deve ser programado de acordo com o modo escolhido;
- *TEI1*: Deve ser programado de acordo com o modo escolhido;
- *TEI2*: Deve ser programado de acordo com o modo escolhido;
- *RHCR1*: Não contém informações de interesse neste modo;
- *RHCR2*: Não contém informações de interesse neste modo;
- *RCPS1*: Não deve ser programado/lido neste modo;
- *RCPS2*: Não deve ser programado/lido neste modo;
- *RXCI*: Não deve ser programado neste modo;
- *RRCI*: Não deve ser lido neste modo;
- *XCBI*: Não deve ser programado/lido neste modo;
- *XCB2*: Não deve ser programado/lido neste modo;
- *RCB1*: Não deve ser lido neste modo;
- *RCB2*: Não deve ser lido neste modo;
- *XSGS*: Não deve ser programado neste modo;
- *RSGS*: Não deve ser lido neste modo;
- *RFCS*: Não deve ser programado/lido neste modo;
- *RMON*: Não deve ser programado/lido neste modo;

## 5.2 Um Processo Driver de Enlace de Dados

*Nota:* A idéia desta seção é centrada na aplicação de sinalização via canal D na RDSI.

Quando operando no modo automático, o CH2 executa a grande maioria das funções do nível de enlace do Modelo de Interconexão de Sistemas Abertos da ISO aplicado numa RDSI. Ele faz isso com um mínimo de supervisão por parte do processador e usualmente executa tarefas simultaneamente com este. O processador e o CH2 se comunicam através de uma estrutura de FIFOs de recepção e transmissão e através de registradores de leitura e escrita, ambos internos ao controlador.

O primeiro problema a ser resolvido para usar o CH2 é definir como o chip se adequa ao ambiente de software do sistema. O CH2 é muito mais potente que a grande maioria dos componentes periféricos conhecidos e por isso sua adequação pode apresentar alguns problemas particulares.

Consideremos inicialmente o modelo geral de como software de sistemas distribuídos poderia ser implementado, mostrado na figura 39. O caso específico de sinalização via canal D na RDSI pode ser visto na figura 40.

Entradas para o circuito CH2 incluem programação, comandos e conjunto de dados provenientes do processador. Resultados e saídas incluem comandos concluídos, quadros recebidos e interrupções.

O processo *Handler* deve consistir de procedimentos preliminares de *designação e teste de TEI* e de rotinas de software que realmente controlam o circuito. Tais rotinas incluem:

- Escrita e leitura nas estruturas de controle do CH2 (registradores);
- Geração de comandos para o CH2;
- Carregamento e leitura das FIFOs internas ao CH2 de/para buffers de memória;
- Tratamento das interrupções.

### 5.2.1 Modelo para Buffers de Dados

Para evitar gastos com espaço de memória usadas para buffer, uma estrutura é sugerida que elimina a cópia de um buffer de dados da camada 3 para outro gerenciado pelo processo *handler*; desta forma o processo de nível 3 acessa diretamente os buffers de transmissão e recepção, através de apontadores.

O *handler* do CH2 aloca buffers de dois *pools* de memória, conforme ilustrado na figura 41.

Quando o processo de camada 3 quer transmitir dados, ele pede um buffer de transmissão para o *handler*. Este retorna um apontador para um BT (Buffer de Transmissão) livre. O processo de camada 3 preenche o buffer e passa o apontador de volta ao *handler* para que ele execute a transmissão. Depois que o

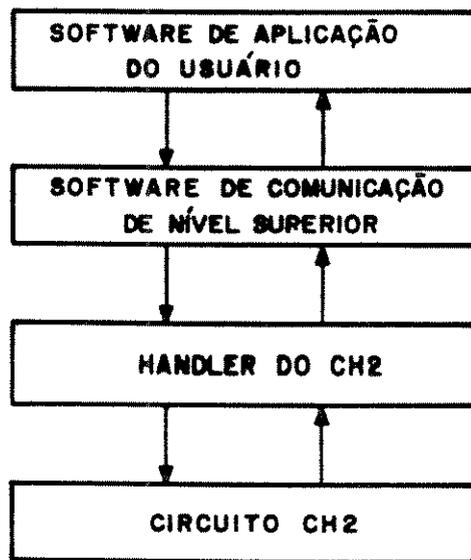


Figura 39: Modelo de Software de Sistema Distribuído

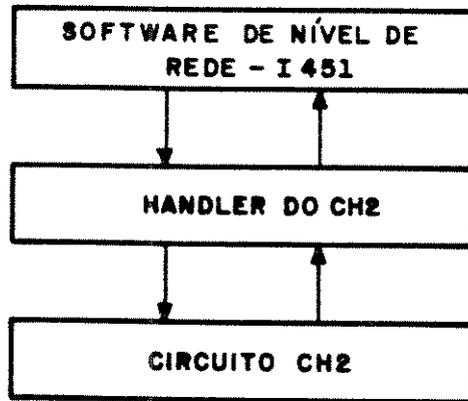


Figura 40: Software de Sinalização via Canal D na RDSI

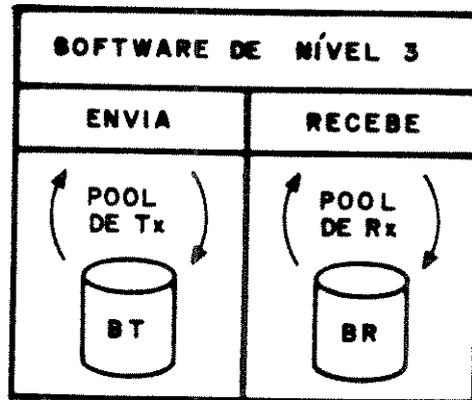


Figura 41: Modelo de Gerenciamento da Memória do Handler

quadro é transmitido, o *handler* coloca o buffer na fila de mensagens transmitidas e não reconhecidas (fila de retransmissão). Se o processo de camada 3 precisa de mais que um buffer por quadro, ele simplesmente pede outro BT para o *handler* e executa a ligação necessária com o BT anterior. Após o reconhecimento da mensagem o BT é liberado da fila de retransmissão.

Do lado da recepção, o *pool* de buffers é gerenciado pelo próprio *handler*. Quando um quadro é recebido, o CH2 interrompe o *handler*. Este passa um apontador para o BR (Buffer de Recepção) para o processo de camada 3. Quando este já tiver tratado os dados, o apontador é retornado ao *handler*, que coloca o BR na fila de buffers disponíveis.

### 5.2.2 A Interface CH2 com o Handler

A interface do CH2 com o *handler* provê as seguintes funções básicas:

1. inicialização do TB22;
2. inicialização/término do enlace;
3. gerência de fluxo de dados;
4. envio e recepção de quadros.

Ao ser ligado o sistema (power on), a função de inicialização é chamada. Esta função inicializa o CH2 e executa diagnósticos. Depois da inicialização, o *handler* está pronto para a execução das demais funções, dependendo do modo de operação programado.

Para enviar um quadro, o processo de camada de rede toma um ou mais buffers de transmissão do *handler*, preenche-os com dados, e chama a função de envio. Quando um quadro é recebido, o *handler* chama uma função de recepção no processo de nível de rede. Esta remove a informação que necessita e retorna os buffers de recepção para o *handler*.

As funções de Início de Enlace e Fim de Enlace assim como as funções de Gerência de Fluxo de Dados (ocupado/livre) são ativadas a partir de chamadas provenientes do processo de nível 3.

### 5.2.3 Inicialização do CH2

Compõe-se de três fases:

1. *Primeira Fase*: O processador precisa ressetar o CH2. Um reset software deve ser usado; ele fará com que os flags internos sejam colocados no estado inativo. Em particular, o TLD é colocado no estado *inativo* e os parâmetros de configuração assumem um valor *default*.

2. *Segunda Fase:* Trata-se da configuração do modo de operação do CH2. Neta fase são definidos como operação as interfaces e quais são os parâmetros do THDLC e do TLD.
3. *Terceira Fase:* O CH2 é colocado na fase operacional do modo automático. A partir daí ele executará autonomamente a maioria das funções do LAPD.

#### **5.2.4 Procedimentos de Designação e Teste de TEI**

Este procedimento deve ocorrer logo após a ativação do Equipamento Terminal e a execução das Primeira e Segunda Fases da inicialização do CH2. Ele visa chegar a um entendimento entre a central e o terminal sobre o número alocado de TEI, ou visa o teste do mesmo a qualquer momento quando solicitado pela central. O procedimento está definido e descrito na Recomendação I.441 do CCITT. Após o término do mesmo, o comando TED (CMDR2) deve ser enviado ao CH2 para a ativação da fase operacional do modo automático.

## **Capítulo Seis**

### **Conclusões e Considerações Finais**

## 6 Conclusões e Considerações Finais

Ao longo deste trabalho buscou-se atingir os seguintes objetivos:

- Mostrar como a microeletrônica é fundamental para o sucesso de uma Rede Digital de Serviços Integrados;
- Apresentar um conjunto possível de circuitos integrados VLSI dedicados que possibilitem a implementação da RDSI de forma viável;
- Descrever detalhadamente o funcionamento de um desses circuitos integrados que traz a inovação tecnológica da integração em hardware VLSI de todo o protocolo da camada de enlace na sinalização via canal D na RDSI; além disso o chip é versátil para várias outras aplicações e é compatível com interfaces padronizadas internacionalmente;
- Apresentar um pequeno número de sugestões em que um grupo de projetistas possam se basear para a tarefa da implementação;
- Apresentar algumas considerações iniciais de software para aqueles que se utilizariam do chip quando em funcionamento.

Todo esse conjunto pode ser amplamente melhorado, principalmente no tocante à atualização das interfaces, conforme inovações forem ocorrendo no mercado internacional (veja-se o caso da Segunda Geração da Interface IOM). As sugestões de implementação não pretendem ser as melhores possíveis, mas acredita-se possam ajudar a começar a pensar e discutir a grande tarefa de execução do projeto.

Interessante dizer que este circuito integrado, com grande número de modificações, principalmente no tocante ao Bloco de Interfaces, está sendo projetado por uma equipe de Pesquisadores de Telecomunicações na Área de Projeto de Circuitos Integrados Digitais do Centro de Pesquisa e Desenvolvimento da Telebrás, da qual o autor faz parte, inicialmente como especificador do chip e em seguida como projetista do Bloco TLD.

A decisão do CPqD em investir no desenvolvimento do CH2 levou em consideração algumas situações:

- a Telebrás designou o CPqD a instituição responsável pelo desenvolvimento dos equipamentos nacionais a serem utilizados na Experiência Piloto Brasileira, a ocorrer no início da próxima década. Dentro deste enfoque foram criados três projetos:
  1. Intelitel (Projeto Terminais Inteligentes para RDSI), que deverá desenvolver os adaptadores para terminais X.25, par a/b (telefone) e interface RS-232, além do telefone digital;
  2. Terminal da Central do Trópico RA, que deverá prover a CPA-T Trópico RA, desenvolvida no CPqD, com a capacitação RDSI;

3. TR1-Básico, que deverá desenvolver os equipamentos TR1 dos acessos básicos.

- Para cumprir os objetivos de capacitação em RDSI, é fundamental o empenho do CPqD na área de microeletrônica. Dois circuitos pareciam ser os mais interessantes para iniciarem o processo: o CH2 e o chip de camada 1 da interface U (este último seria desenvolvido num esforço conjunto com a empresa canadense MITEL);
- A Coordenação de Áreas de Circuitos Integrados já havia demonstrado sua capacitação em projetos de pequeno porte nas três técnicas conhecidas (gate array, standard cells e full custom). Havia, no entanto, chegado o momento de assumir o risco de projetos mais ambiciosos. Naturalmente isso implicaria em reformulação da estrutura (hardware e software) e da metodologia de projeto (a Coordenação se responsabilizaria por todo o processo de desenvolvimento: especificação, projeto lógico, projeto elétrico, simulações, layout, geração de padrões e teste) e no desenvolvimento da capacitação de gerência de projetos de alta tecnologia (havia que se coordenar o trabalho de, em época de pico, até 11 engenheiros e 3 técnicos).

Desta forma, o CH2, denominado internamente de TB22, contou com uma equipe de tamanho variável em função das etapas de desenvolvimento. A partir do momento em que a especificação foi terminada e (quase!) congelada, um grupo de 9 engenheiros de projeto digital e 2 engenheiros de estabilidade foram divididos em equipes de trabalho:

- Equipe do Bloco de Interfaces: 4 engenheiros
- Equipe do Bloco THDLC: 2 engenheiros, onde:
  - 1 engenheiro cuidando dos Sub-Blocos RHDLC e TEMP;
  - 1 engenheiro cuidando dos Sub-Blocos XHDLC e GERVAR.
- Equipe do Bloco TLD: 2 engenheiros
- Equipe do Bloco FIFO: 1 engenheiro

Os engenheiros de estabilidade dividiam seu tempo entre as quatro equipes.

A etapa inicial (e mais demorada!) constou do estudo detalhado das especificações requeridas (funcionais e elétricas/mecânicas) e a partir delas gerar um circuito digital que as executasse. Para tornar o projeto controlável, um número finito (e pequeno) de células específicas foram definidas e projetadas, de forma que cobrissem todas as necessidades previstas (naturalmente que ocorreram alguns ajustes nesta biblioteca ao longo do processo). Para cada célula da biblioteca foi criado um modelo lógico usando a linguagem do simulador HILO (simulador utilizado no CPqD, rodando em equipamento Digital VAX 785), de

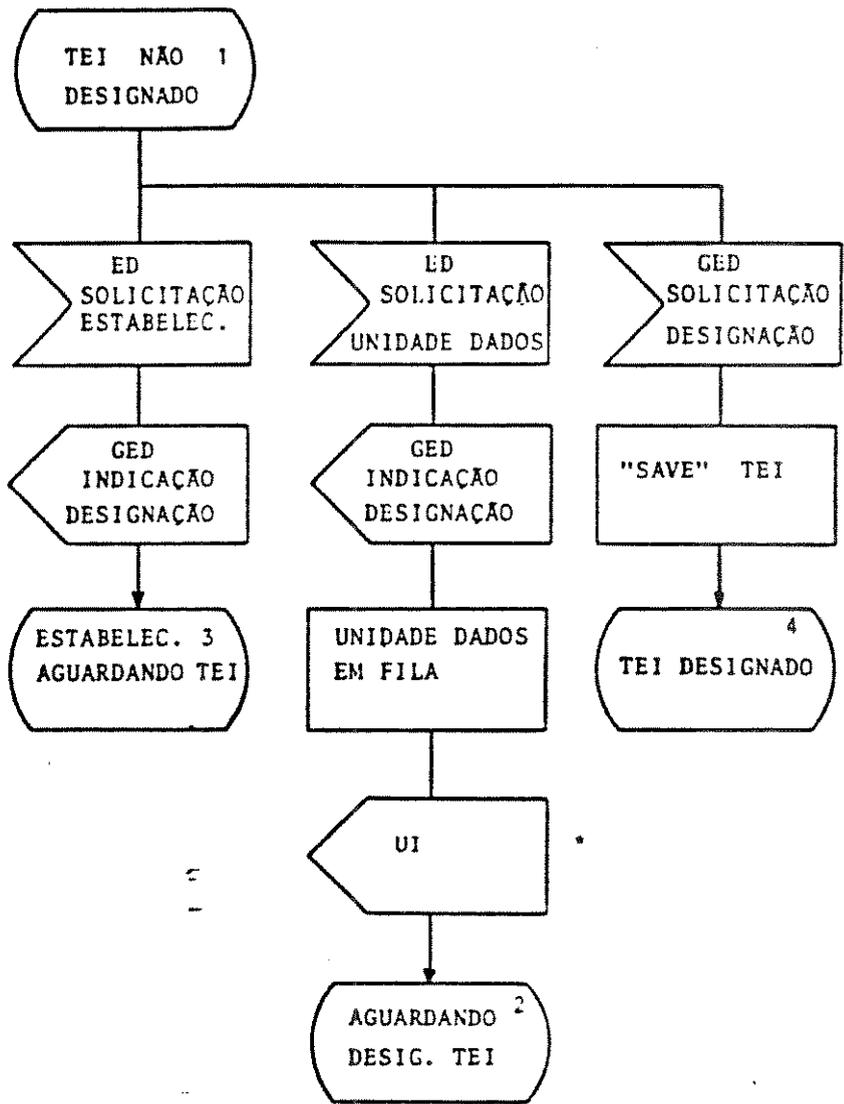
maneira que, uma vez definido um circuito, ele fosse descrito estruturalmente nesta linguagem (junto com uma forma de onda de estímulo) e simulado. O desenvolvimento do circuito foi totalmente executado levando-se em consideração técnicas modernas de projeto visando testabilidade.

No momento em que este capítulo está sendo escrito alguns dados do TB22 já estão disponíveis. A simulação lógica encontra-se em sua fase final, onde os grandes blocos estão sendo integrados. Por maior que tenha sido o cuidado no sentido de bem definir as interfaces entre os blocos e, mesmo que separadamente a simulação tenha dado resultados satisfatórios, integrá-los não é, sempre, uma tarefa trivial. A determinação da melhor forma de onda de estímulo também apresenta suas dificuldades, uma vez que o número possível de modos de operação do chip é muito grande, assim como o número de sinais internos.

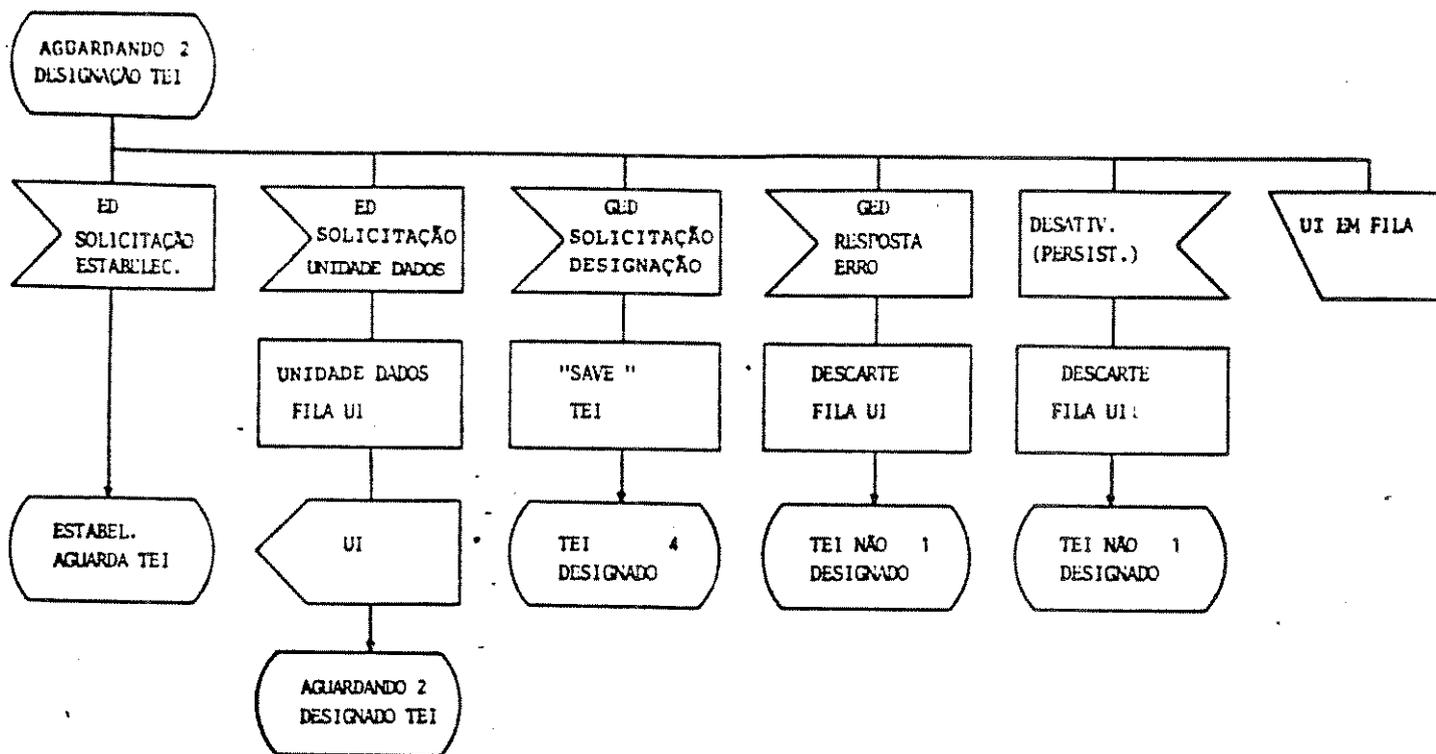
Os dados atuais nos permitem concluir que o TB22 terá da ordem de 50.000 transistores. Ele está sendo projetado na tecnologia CMOS com largura de canal de 1.25 micra, segundo as regras de projeto da AMI (American Microsystem Inc), *foundry* onde será fabricado. O tempo total de projeto (até fabricação), incluindo aquele necessário para treinamento, é de 2 anos; no entanto, com a experiência adquirida e com os conhecimentos e resultados documentados (circuitos, biblioteca de células, metodologia) prevê-se que um circuito de mesma complexidade neste mesmo campo de integração de protocolos poderá ser executado pelo grupo com a metade dos recursos ou com a metade do tempo.

## Anexo I

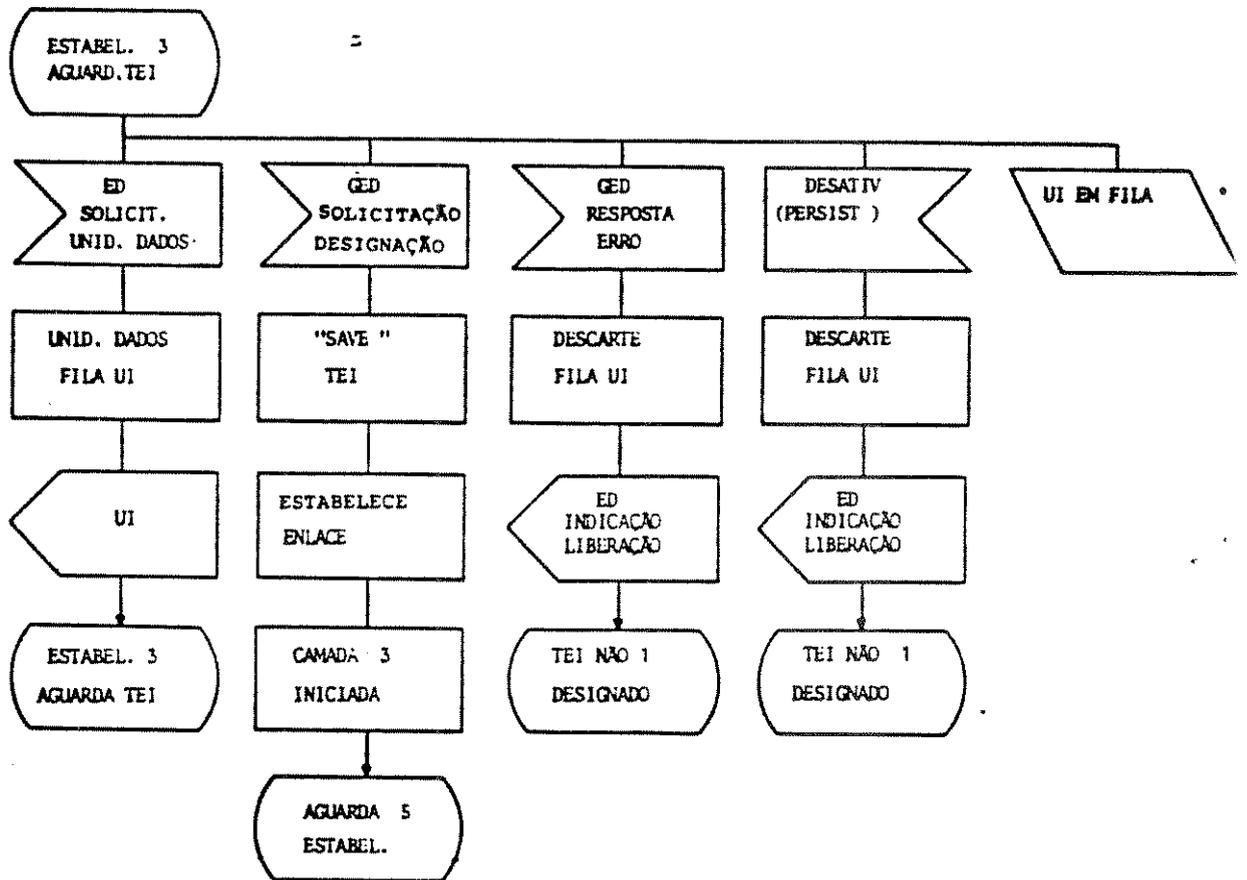
**Protocolo LAPD descrito na forma de LEDS**



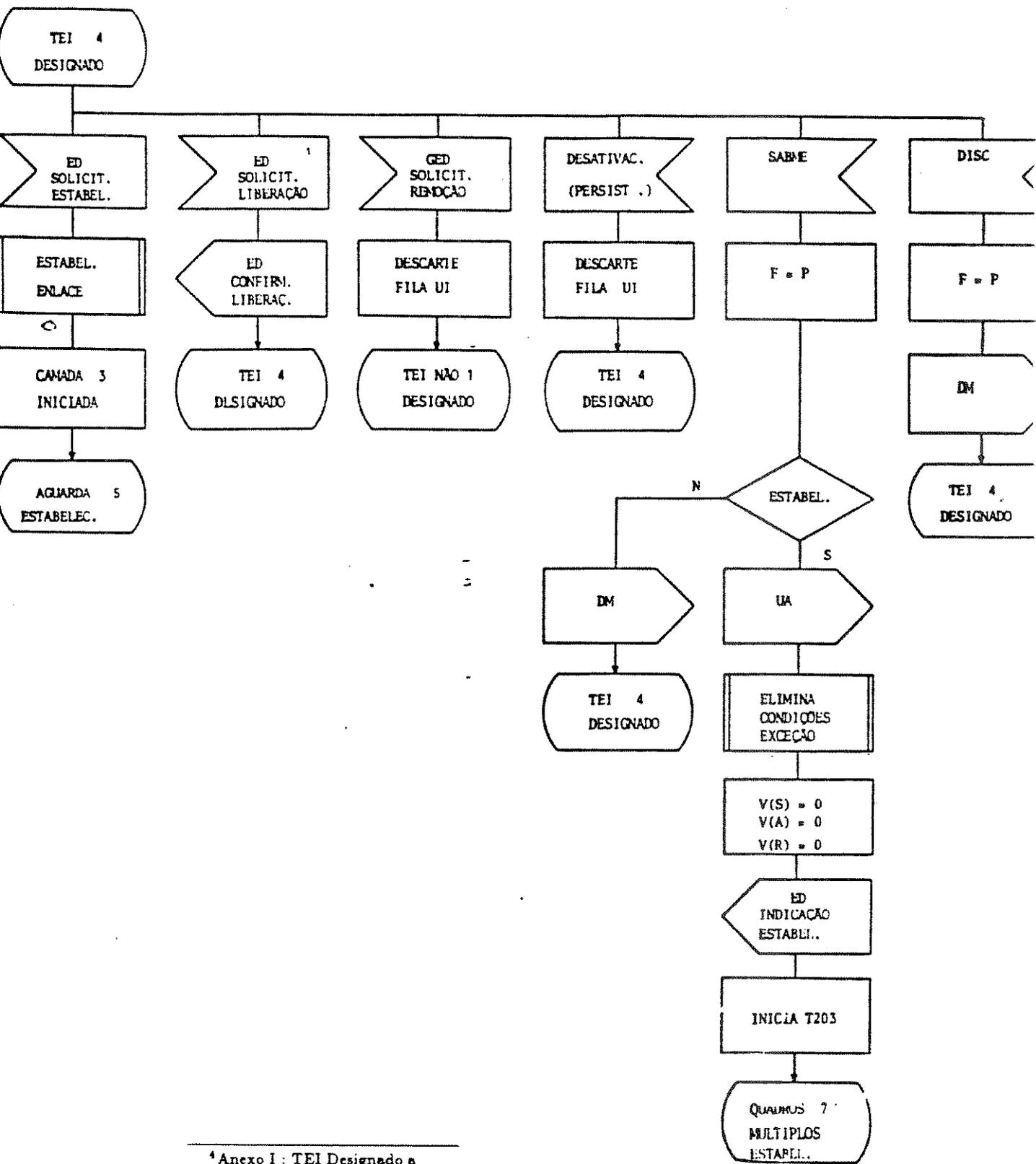
<sup>1</sup>Anexo I : TEI Não Designado



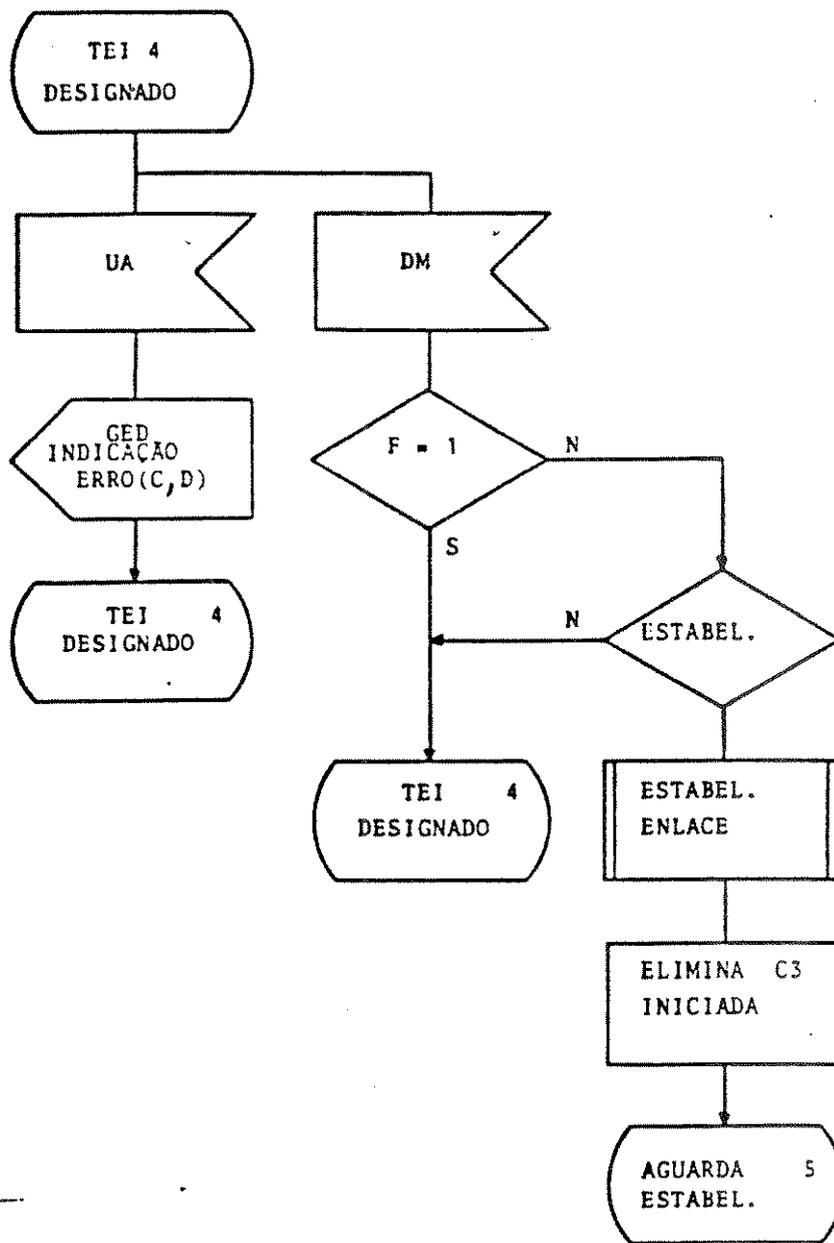
<sup>2</sup>Anexo I : Aguardando Designação de TEI



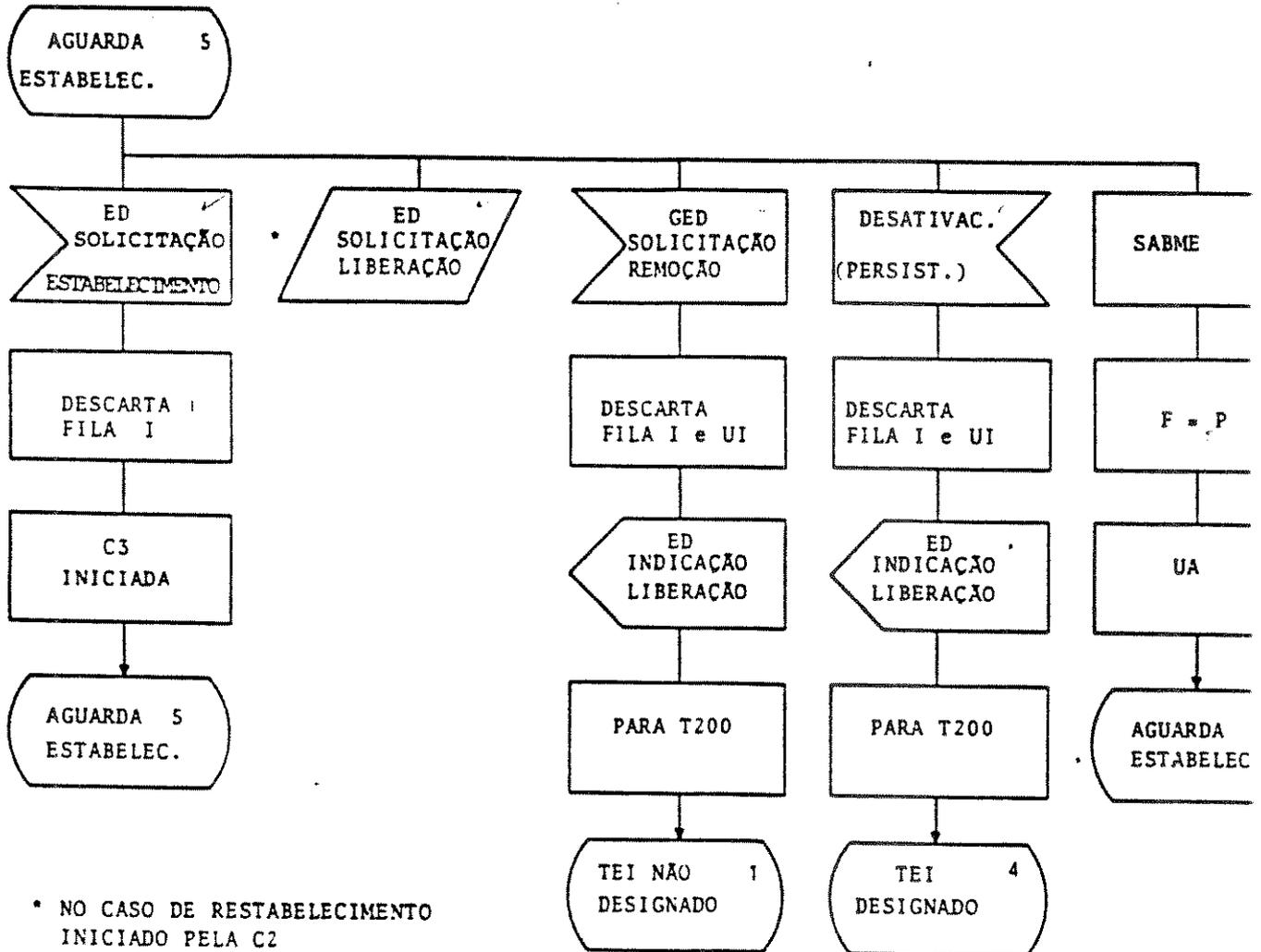
<sup>3</sup> Anexo I : Estabelecido Aguardando TEI



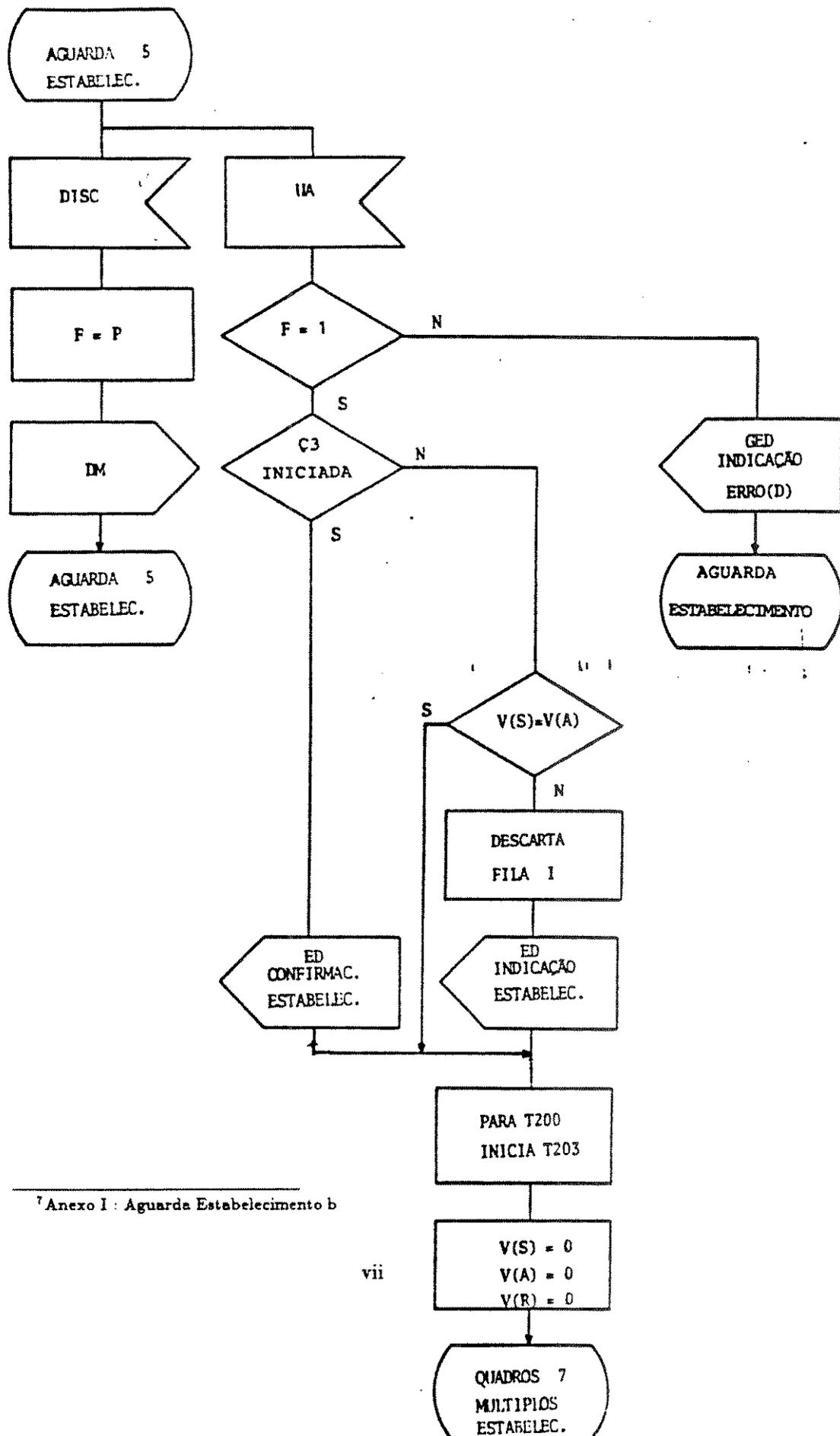
<sup>4</sup> Anexo I : TEI Designado a



<sup>5</sup> Anexo I : TEI Designado b

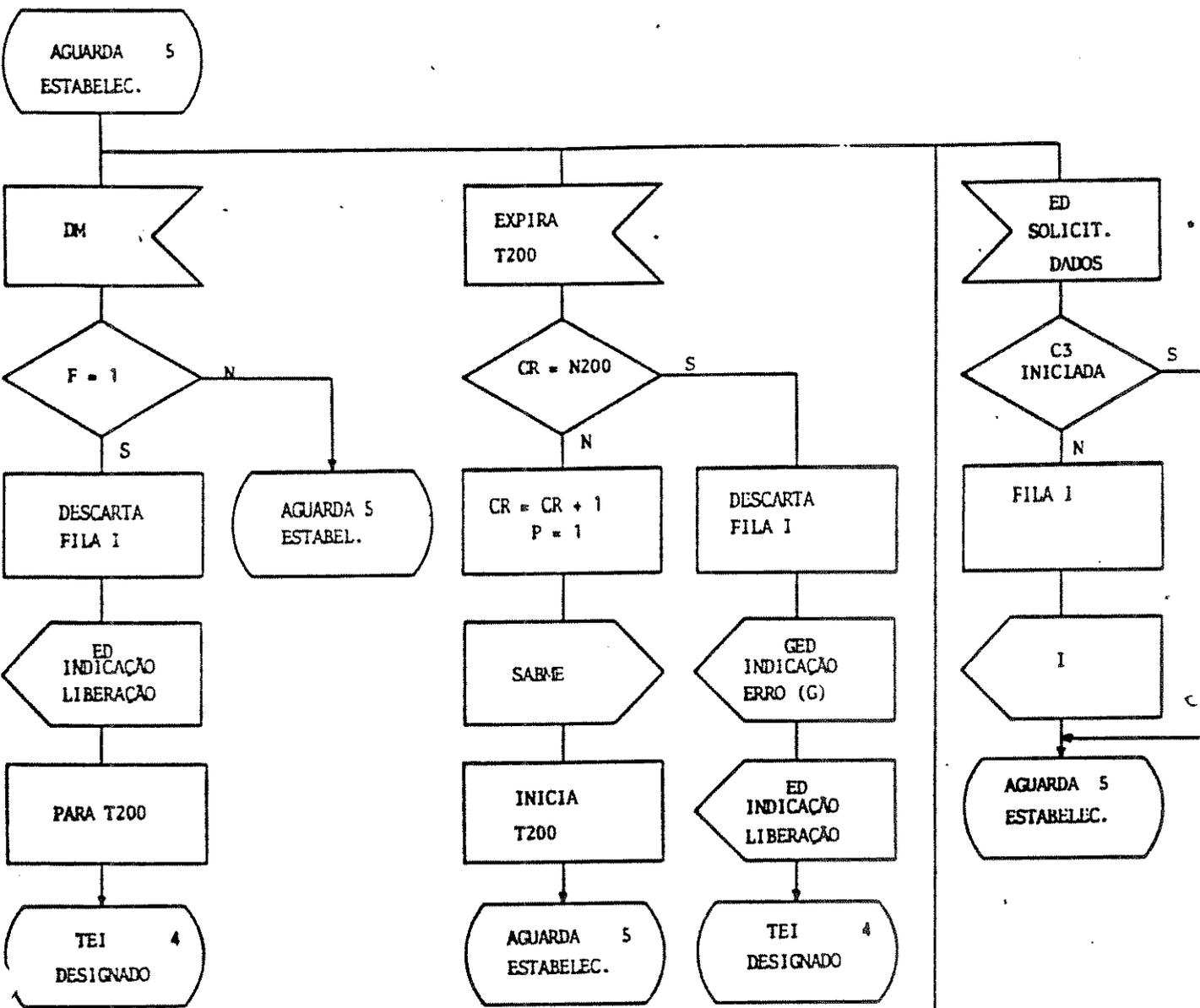


<sup>6</sup> Anexo I : Aguarda Estabelecimento a



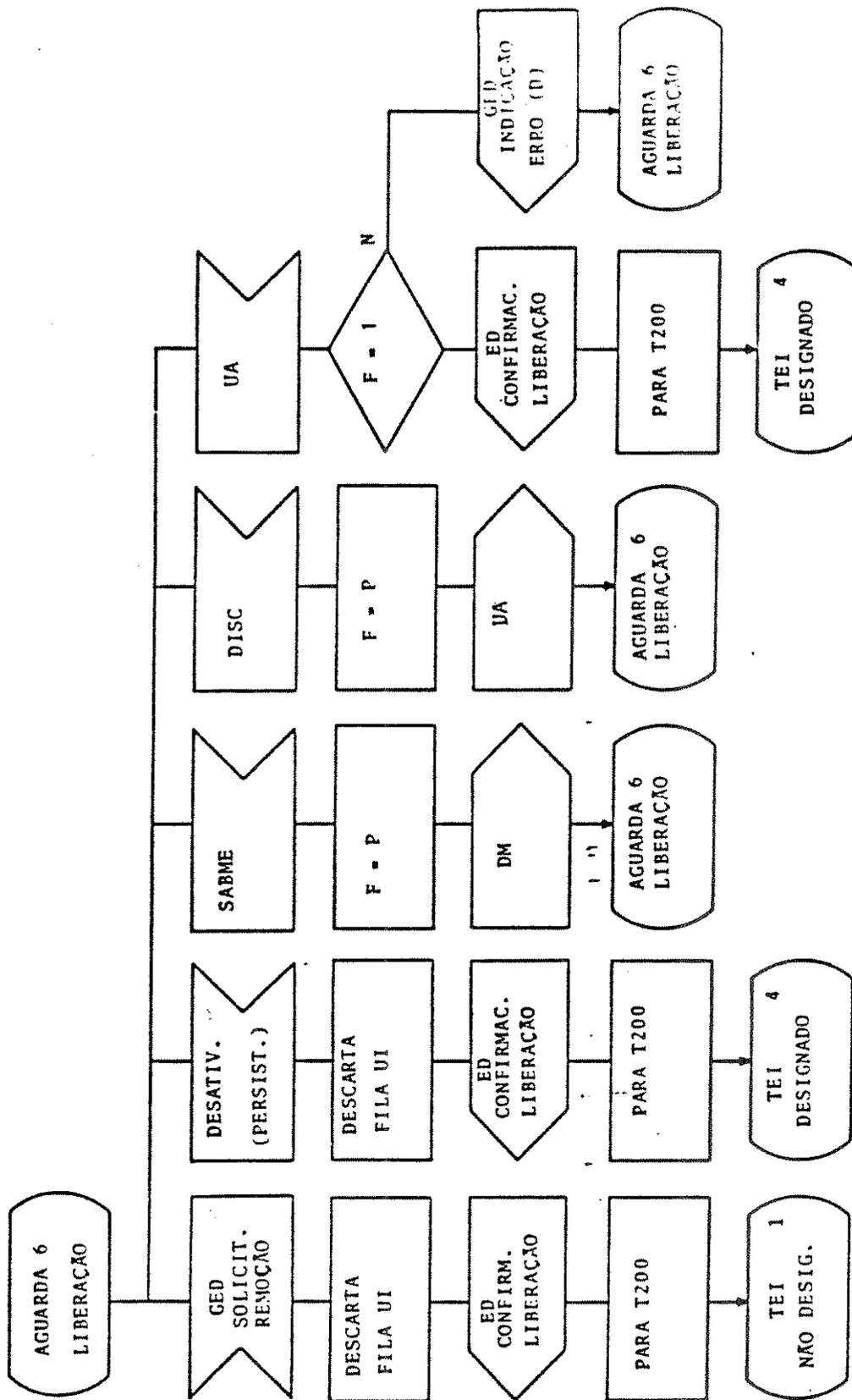
7 Anexo I: Aguarda Estabelecimento b

vii

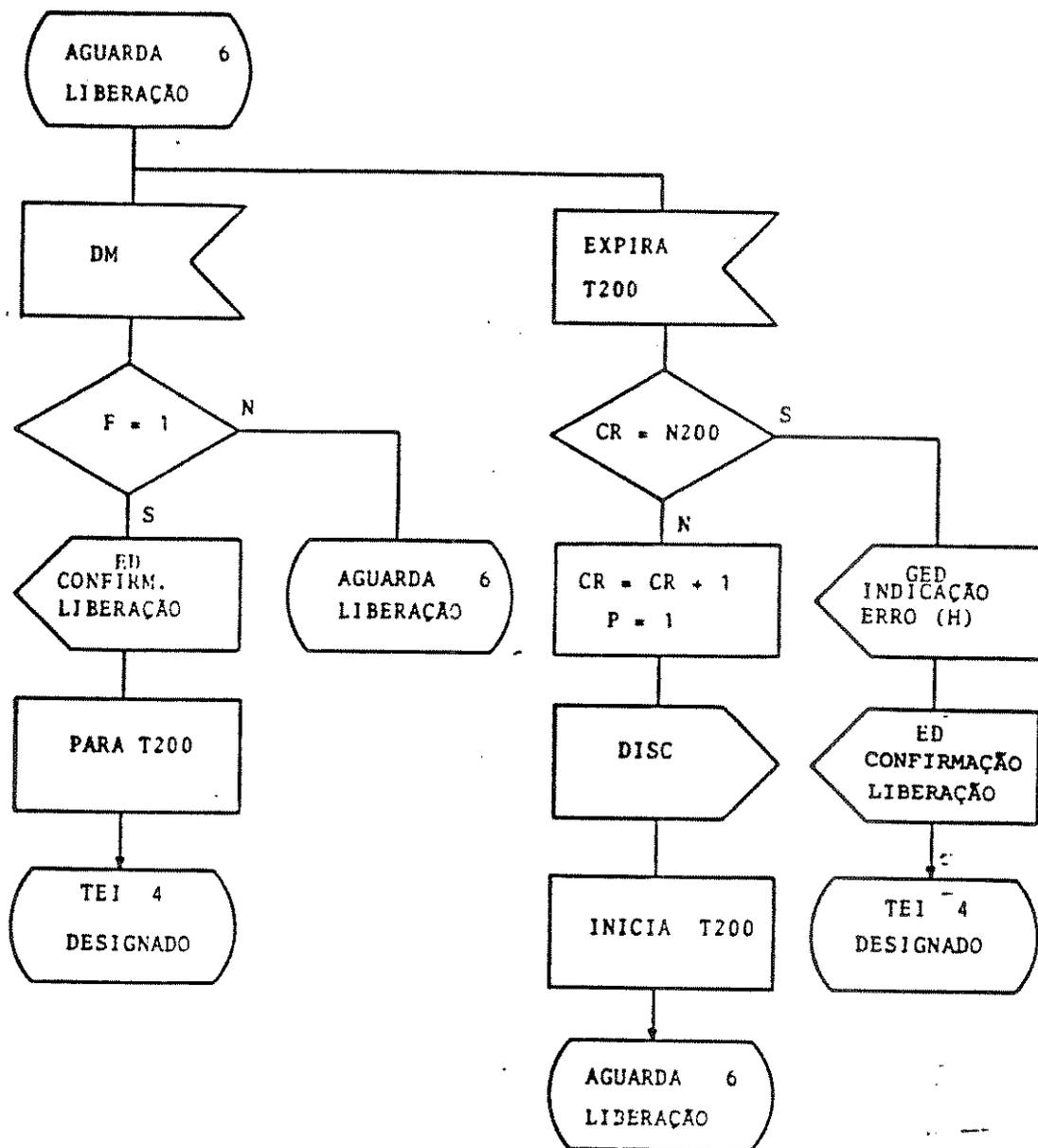


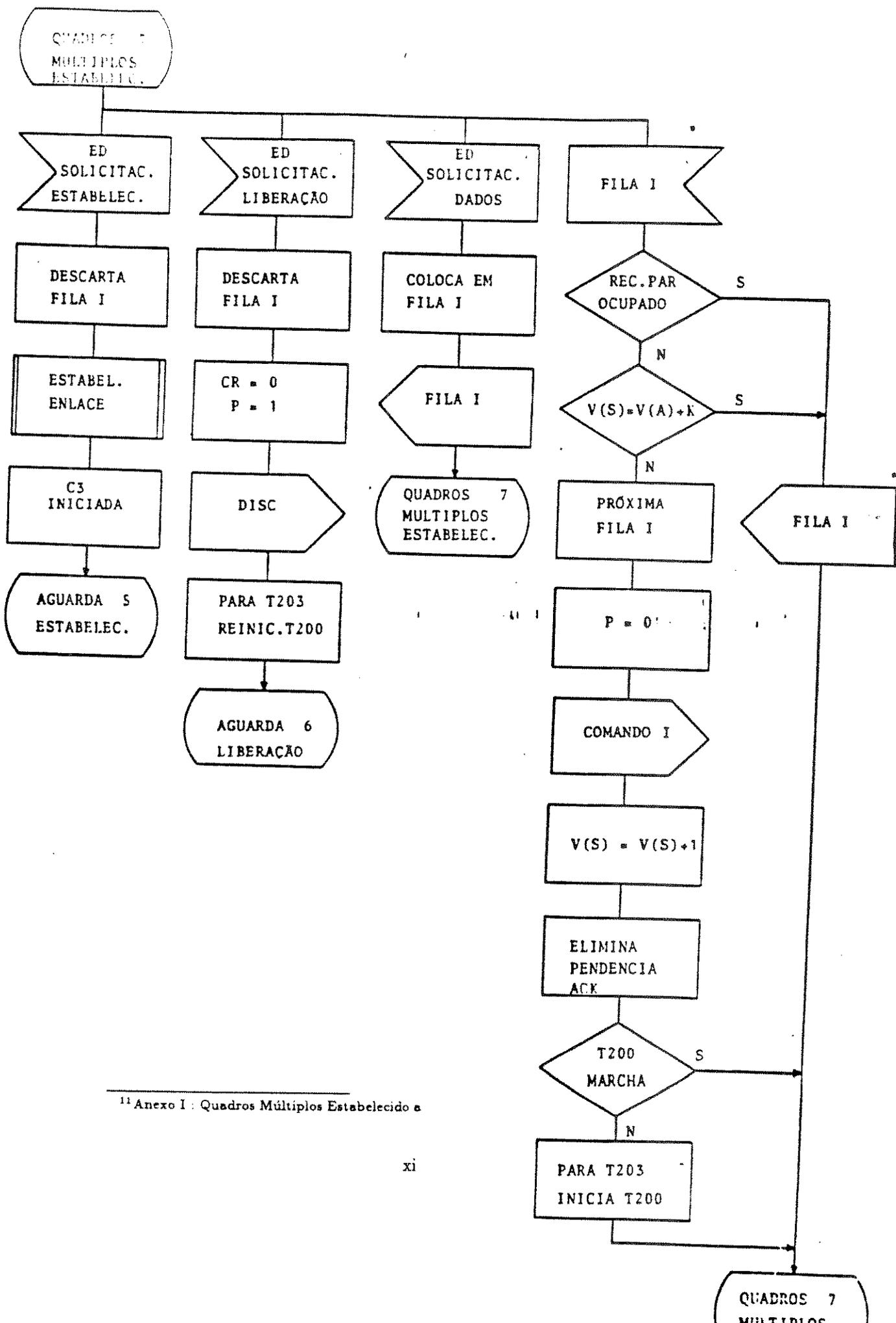
\* NO CASO DE RESTABELECIMENTO INICIADO PELA C2

\* Anexo I: Aguarda Estabelecimento c

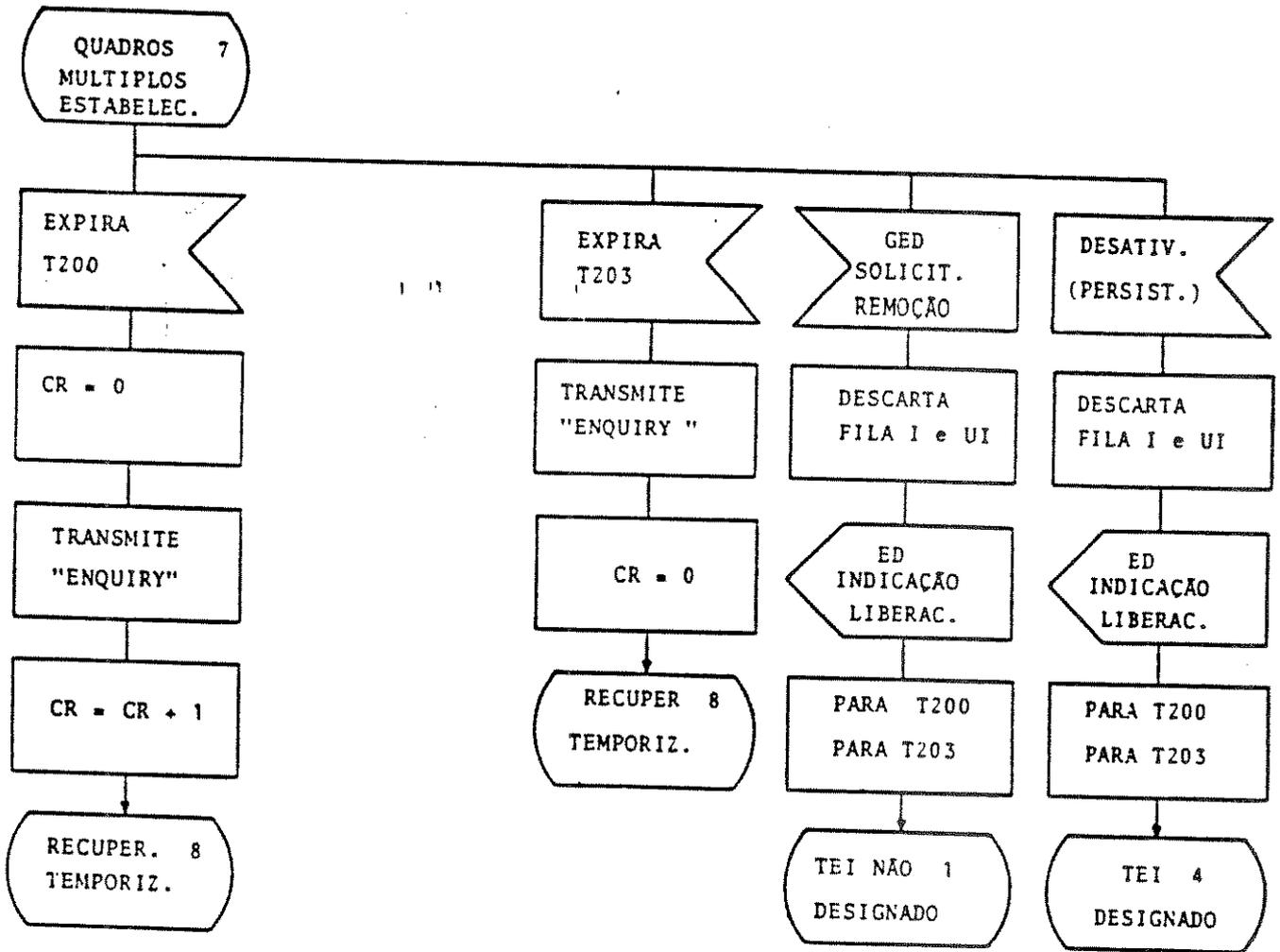


9 Anexo I : Aguarda Liberação a

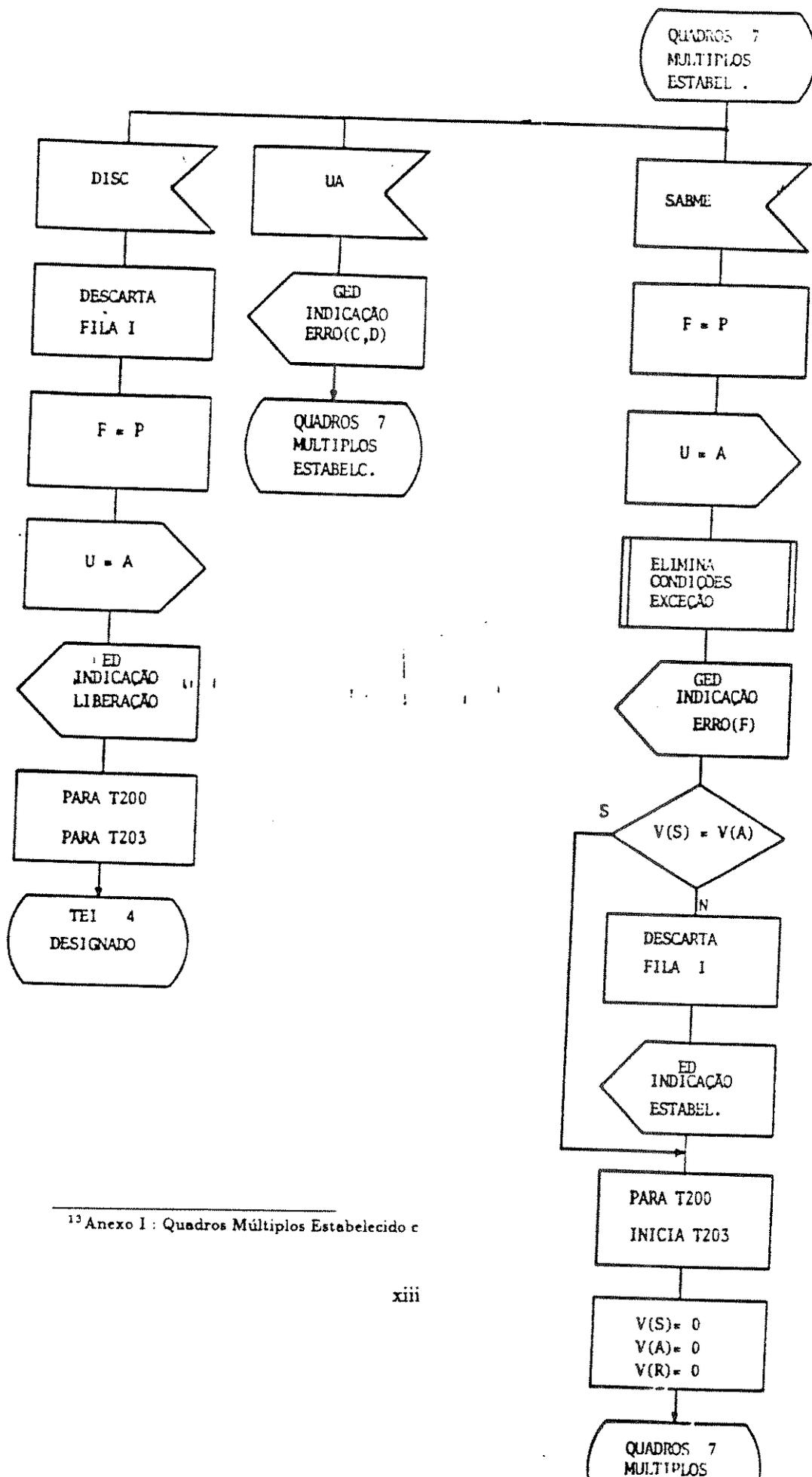




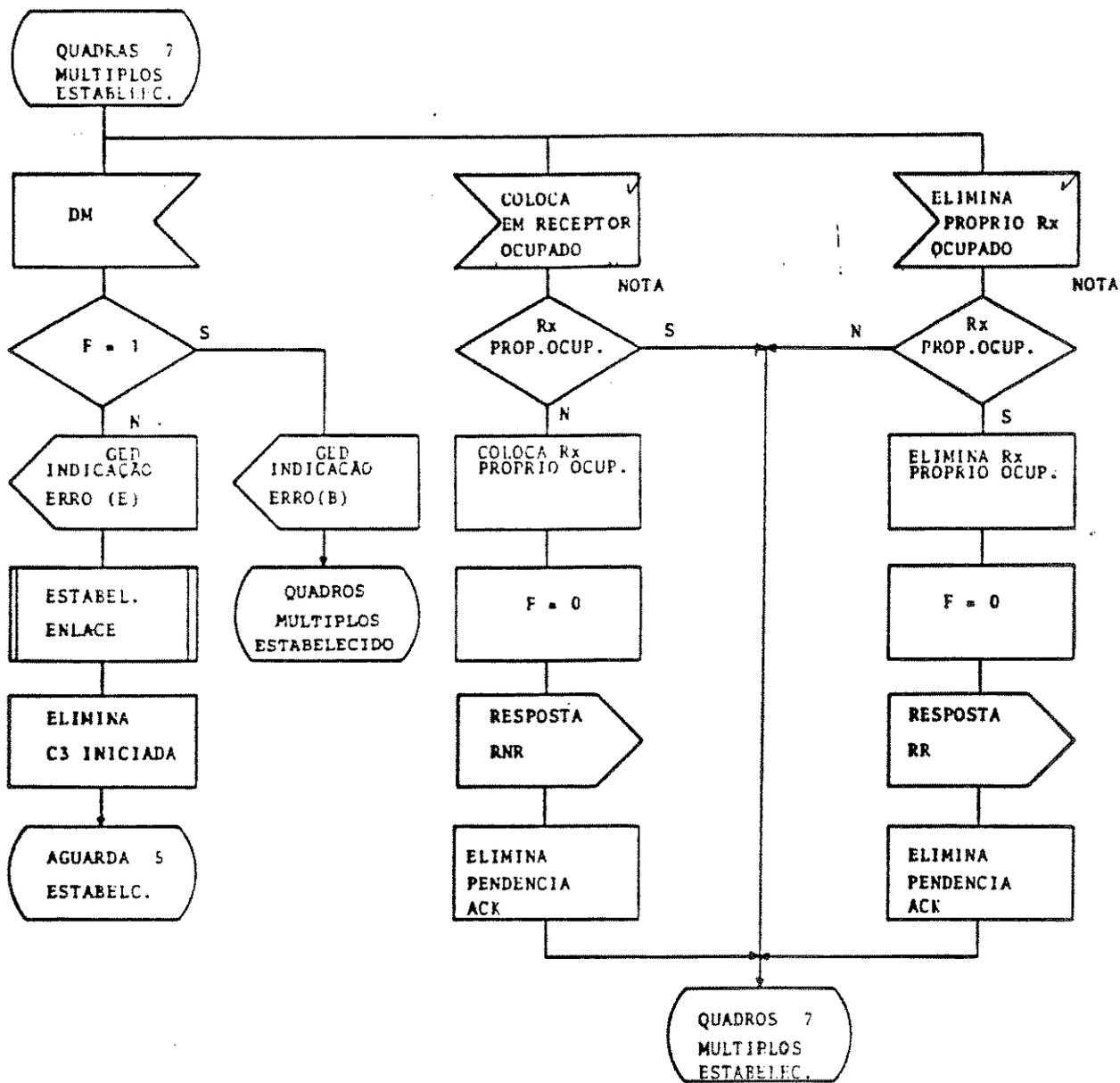
<sup>11</sup> Anexo I : Quadros Múltiplos Estabelecido a



<sup>12</sup>Anexo I : Quadros Múltiplos Estabelecido b

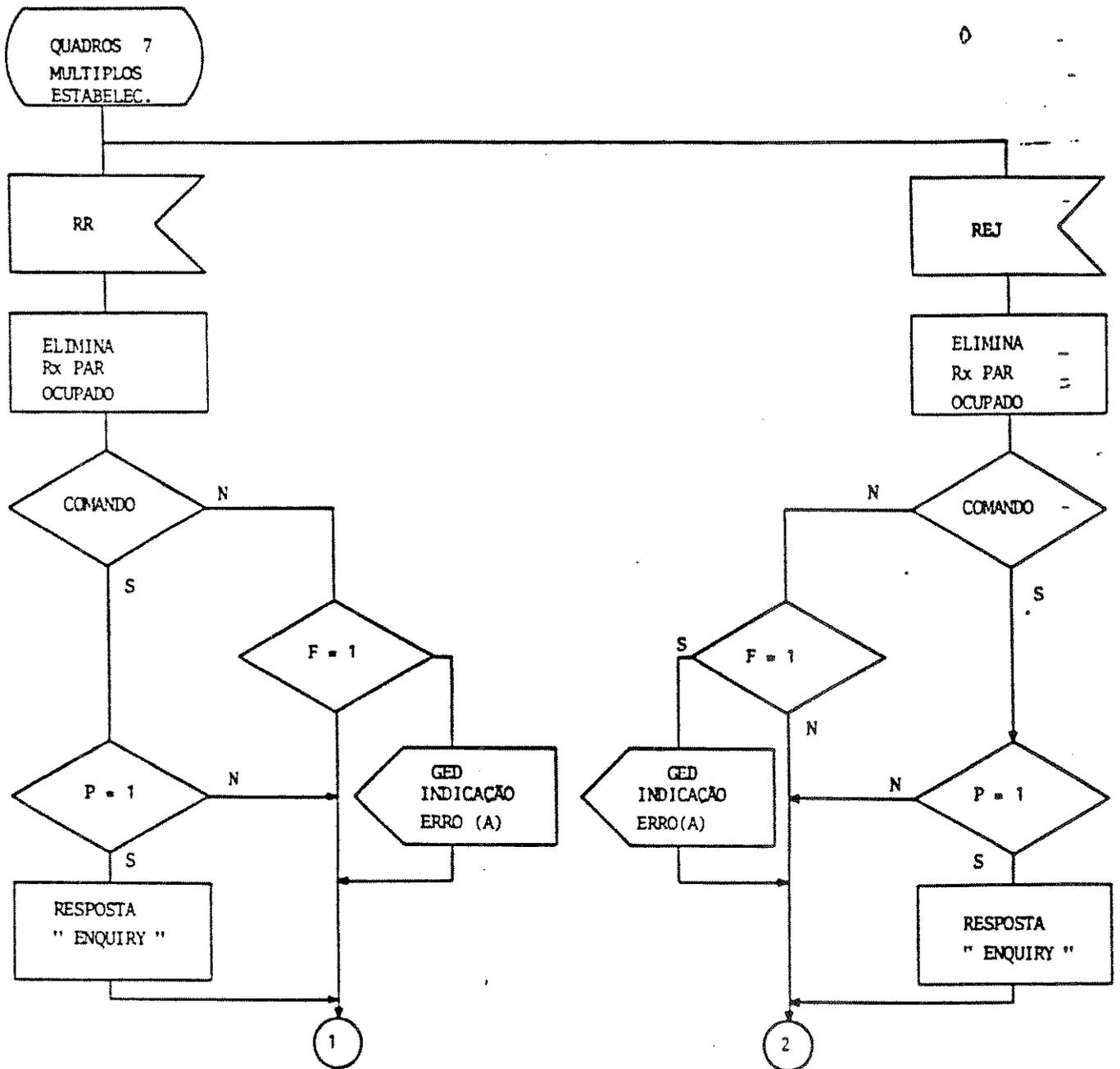


<sup>13</sup> Anexo I : Quadros Múltiplos Estabelecido c

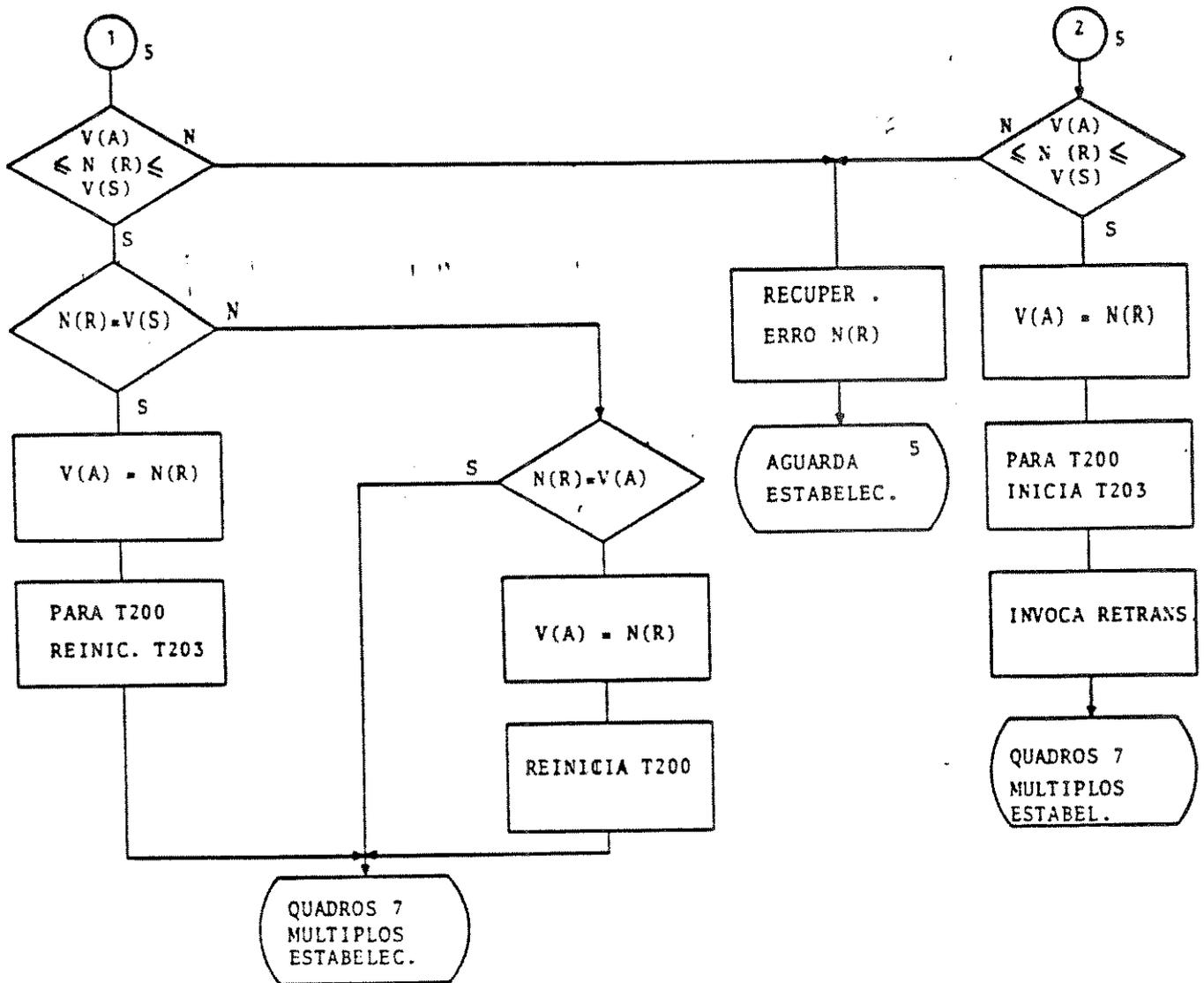


NOTA: SINAIS GERADOS PELA ENTIDADE DE GERENCIAMENTO DA CONEXÃO

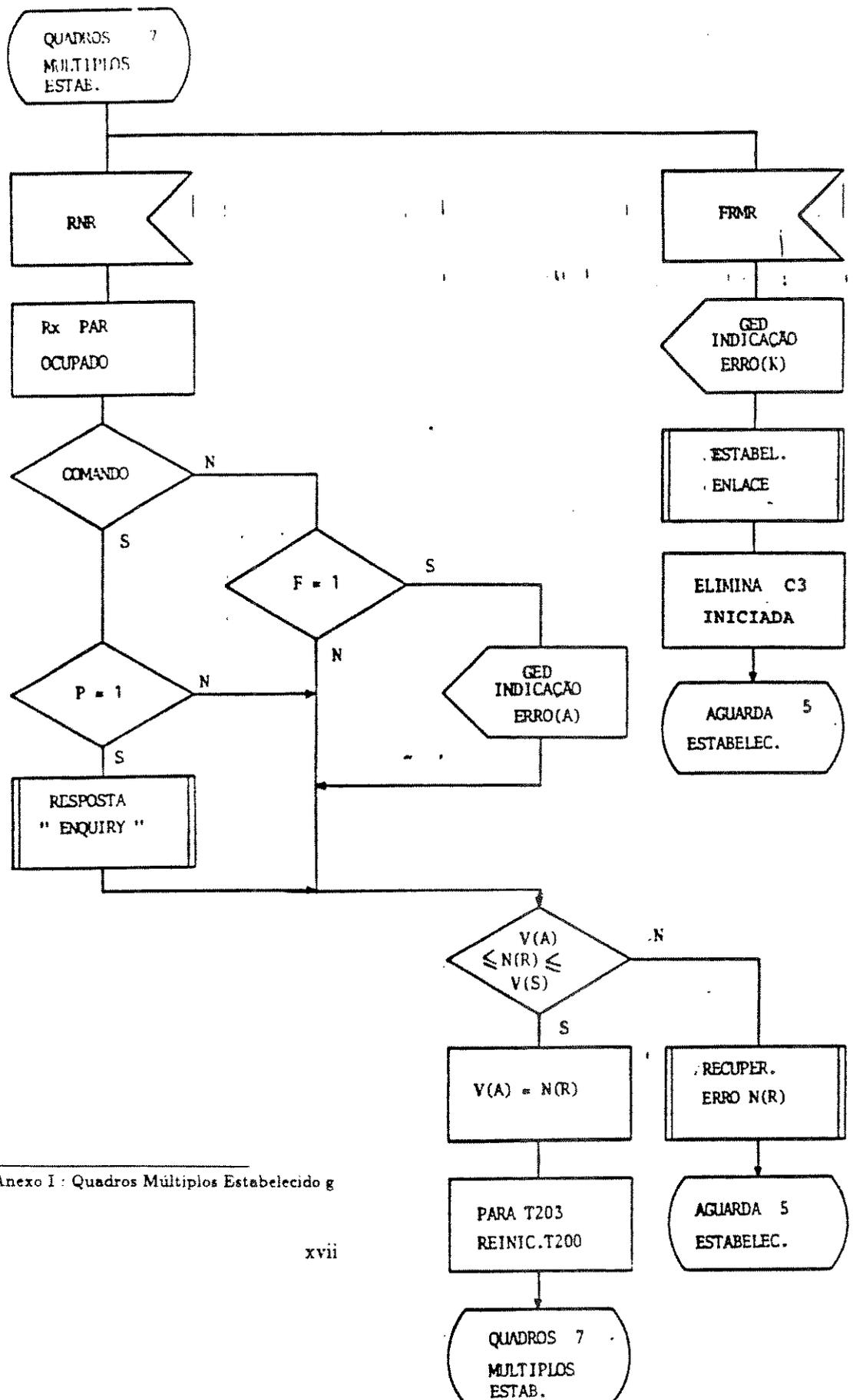
<sup>14</sup> Anexo I: Quadros Múltiplos Estabelecido d



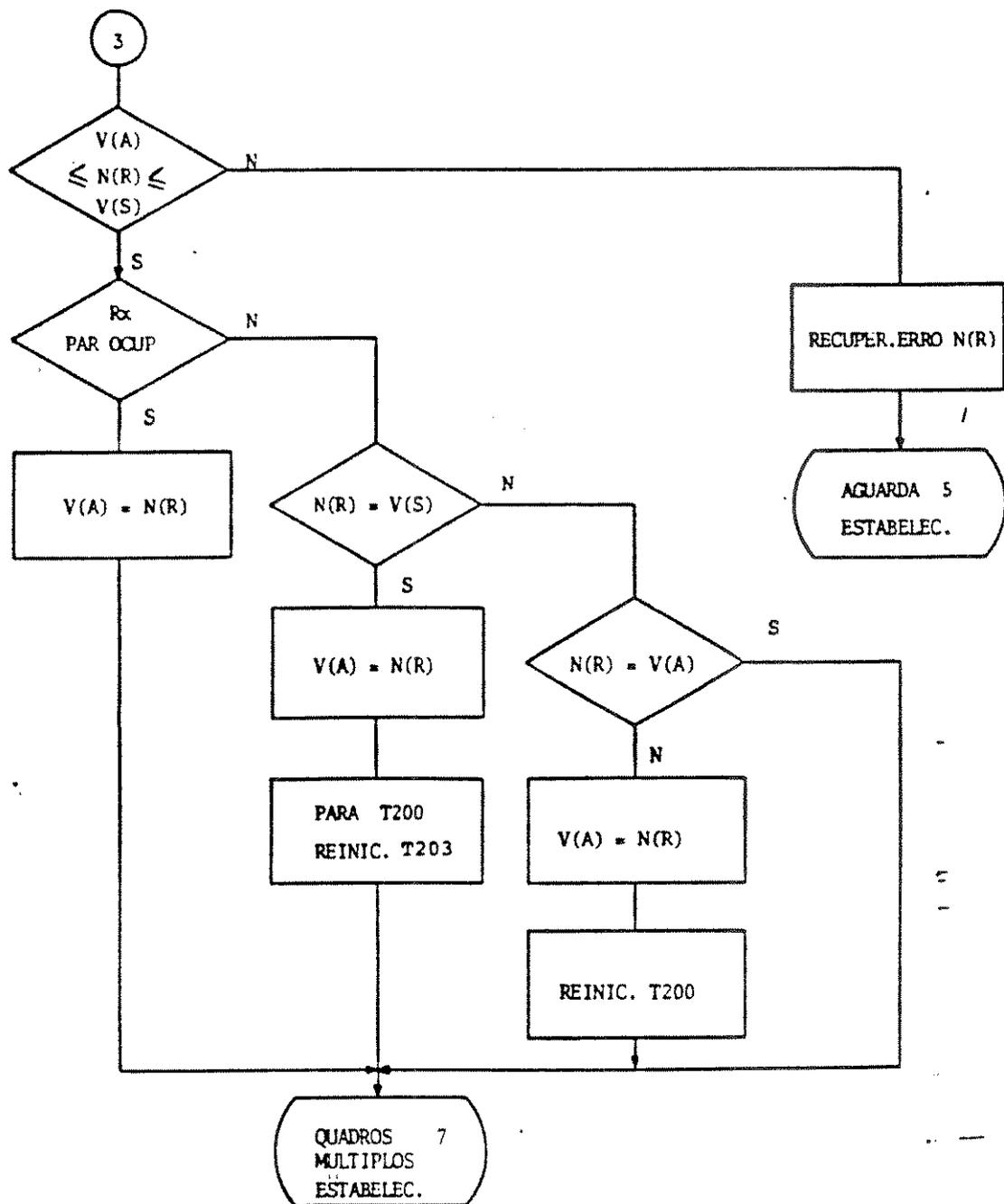
15 Anexo I : Quadros Múltiplos Estabelecido e



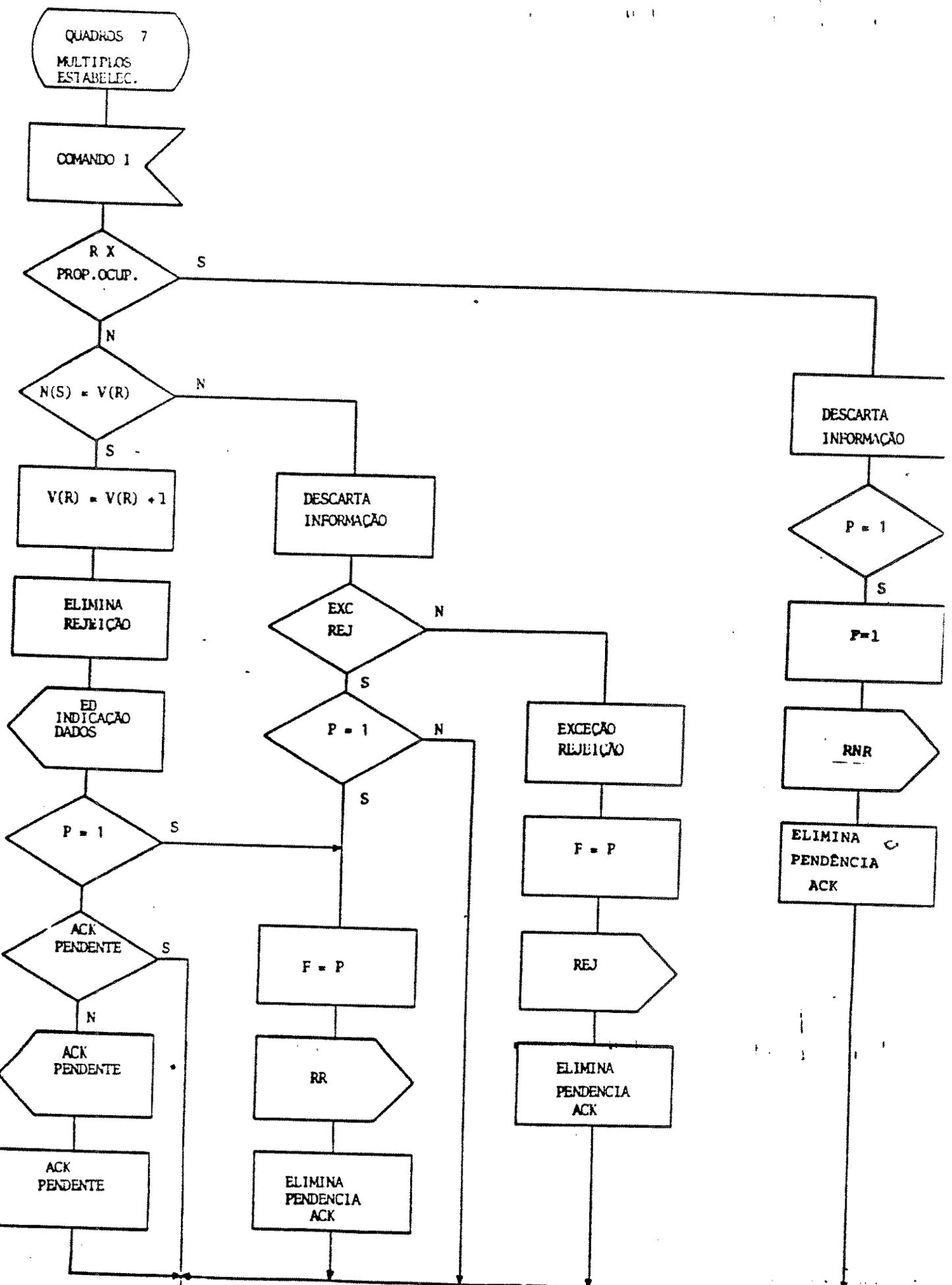
<sup>16</sup> Anexo I : Quadros Múltiplos Estabelecido f

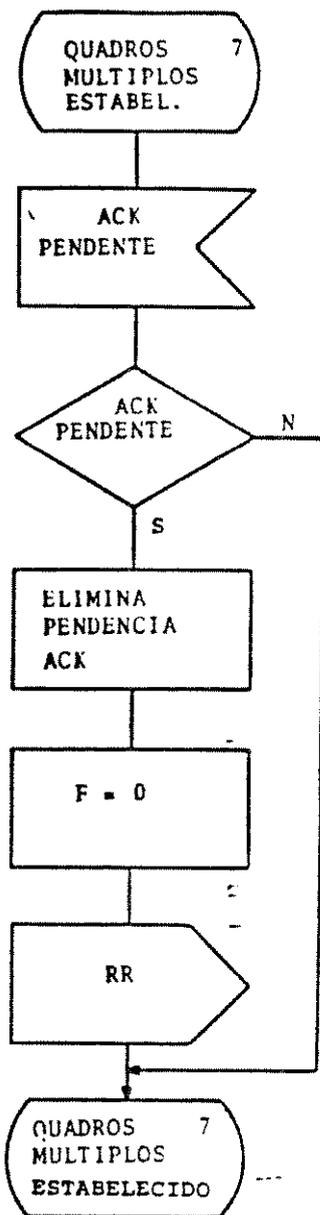


17 Anexo I : Quadros Múltiplos Estabelecido g

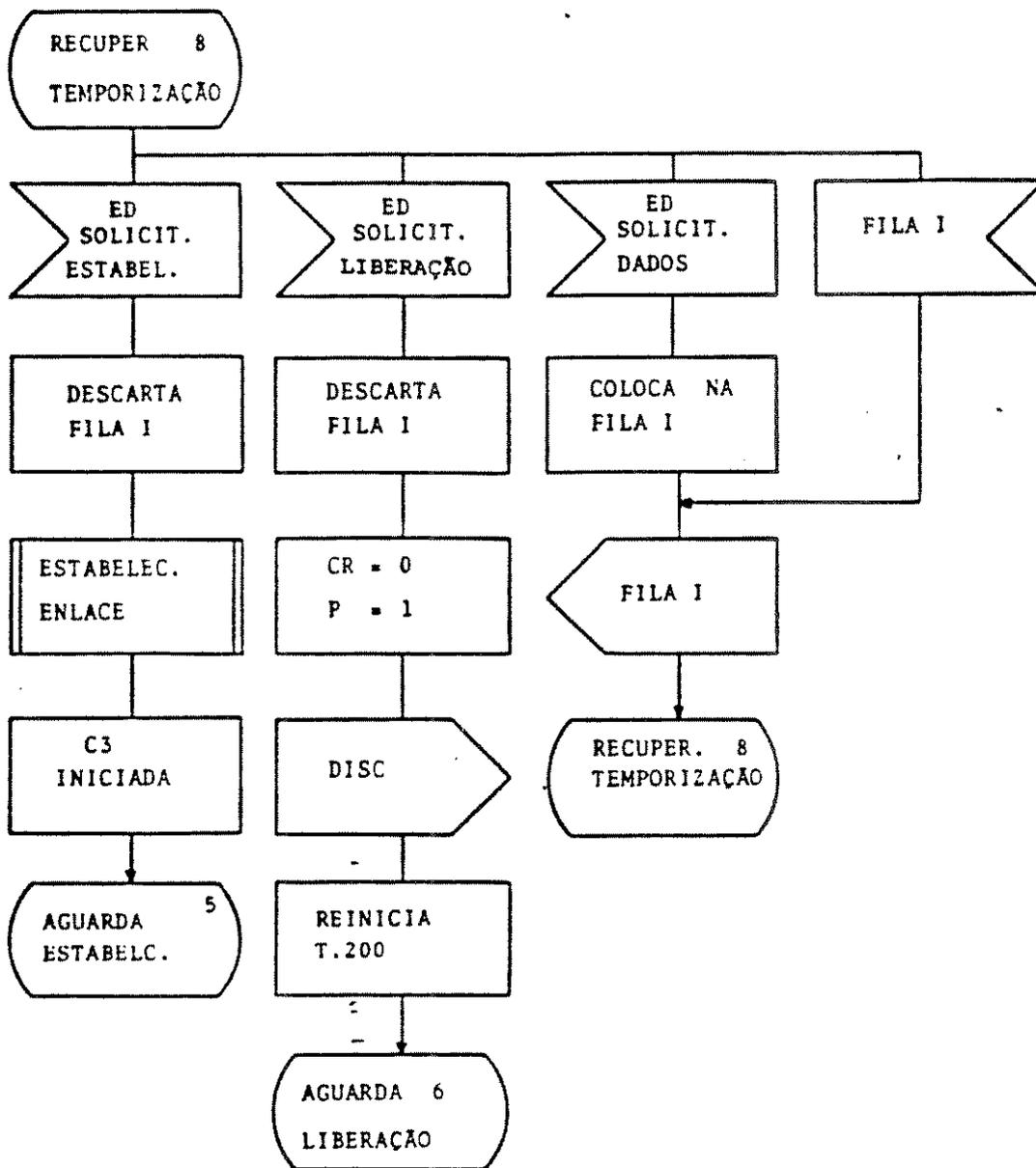


19 Anexo I : Quadros Múltiplos Estabelecido i

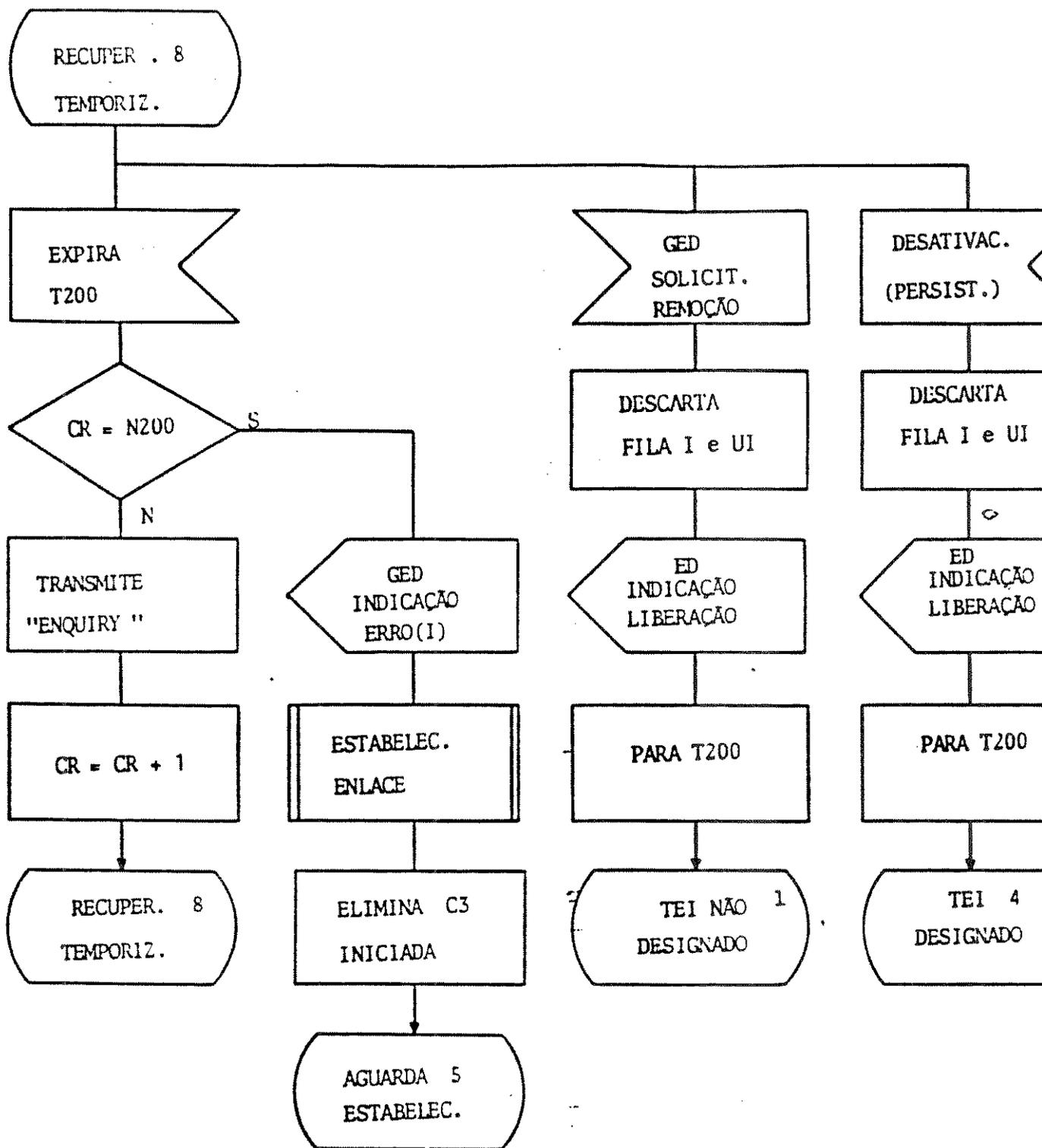




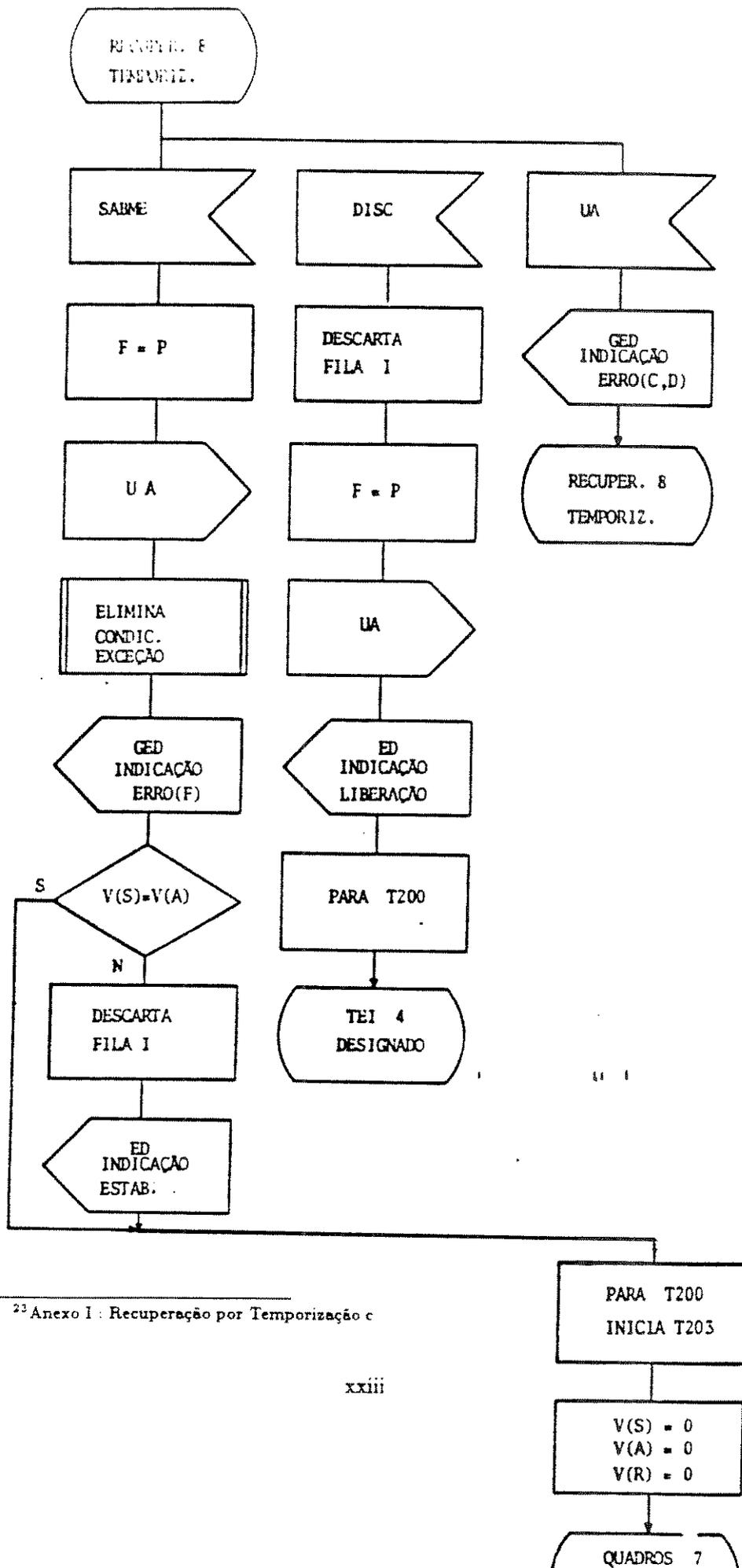
<sup>20</sup> Anexo I : Quadros Múltiplos Estabelecido j



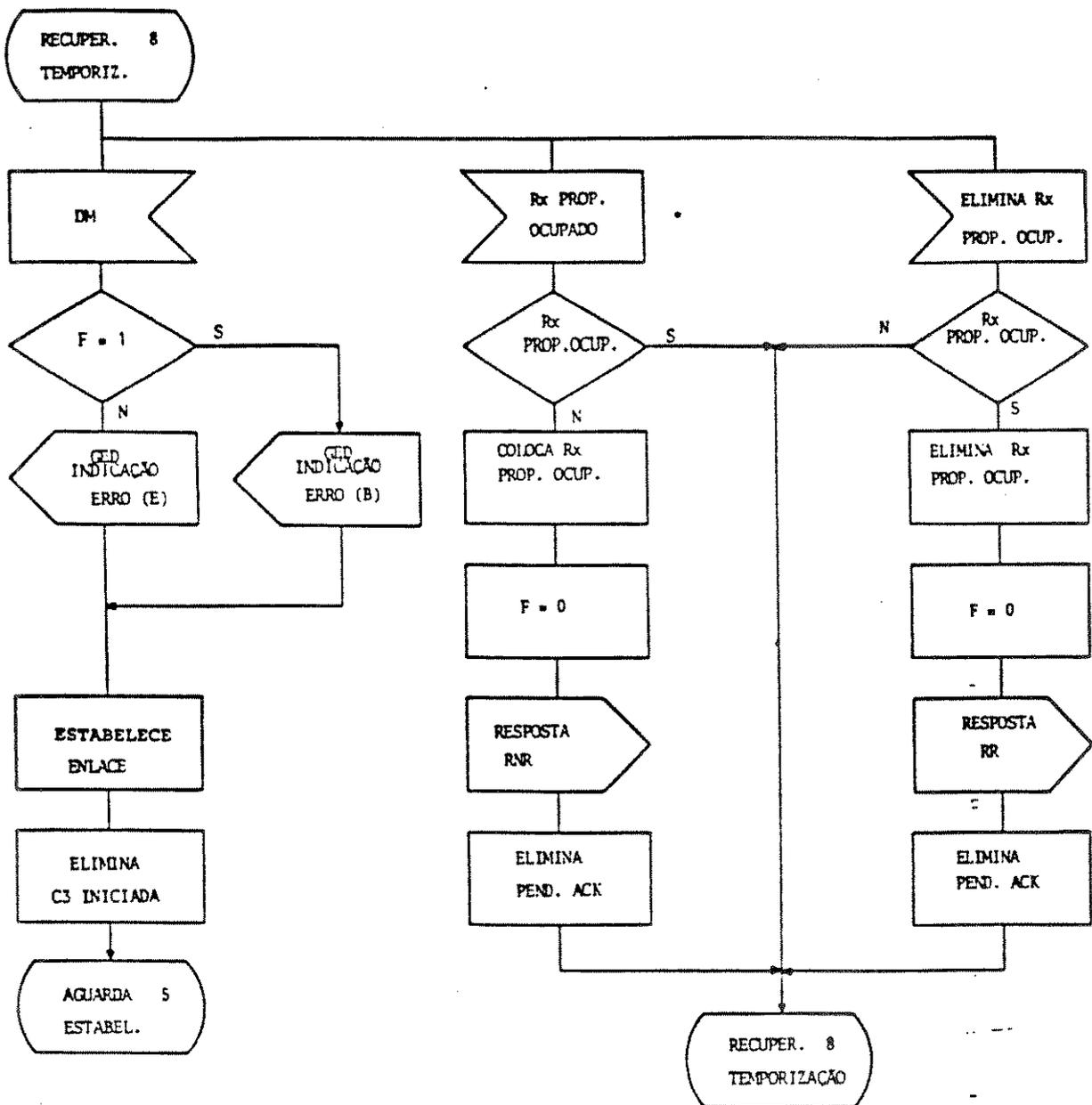
<sup>21</sup> Anexo I: Recuperação por Temporização a



22 Anexo I: Recuperação por Temporização b

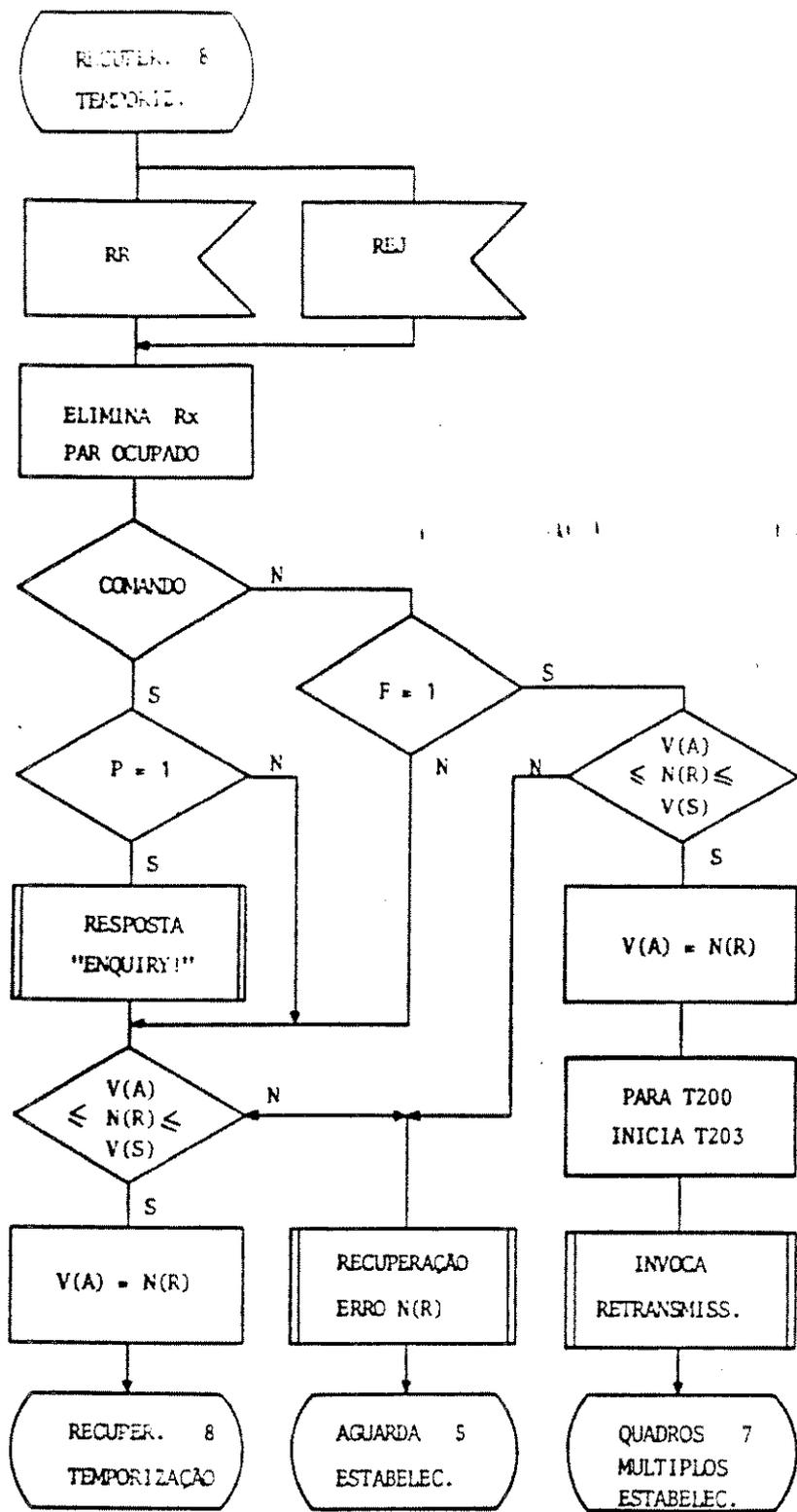


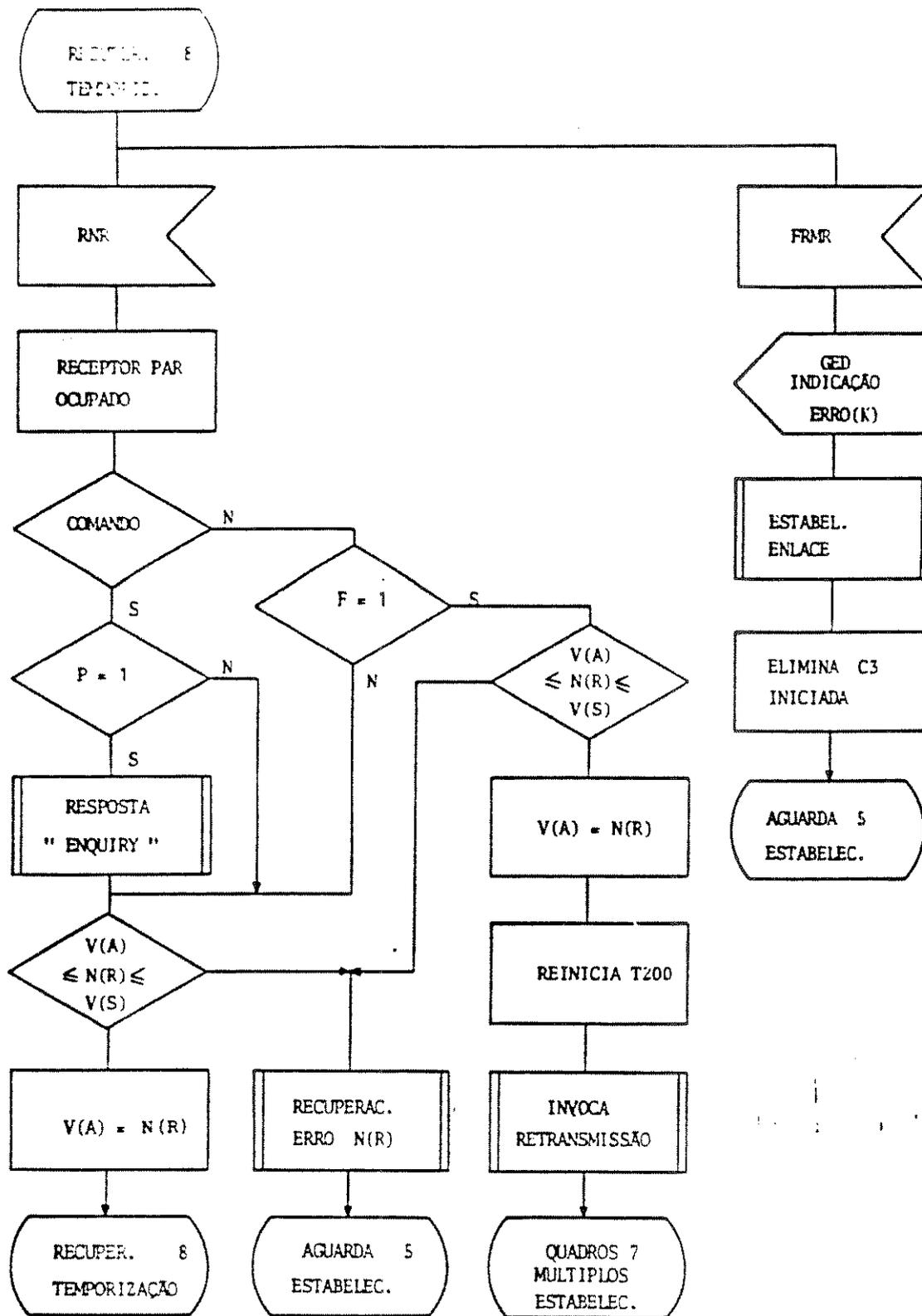
<sup>23</sup> Anexo I : Recuperação por Temporização c



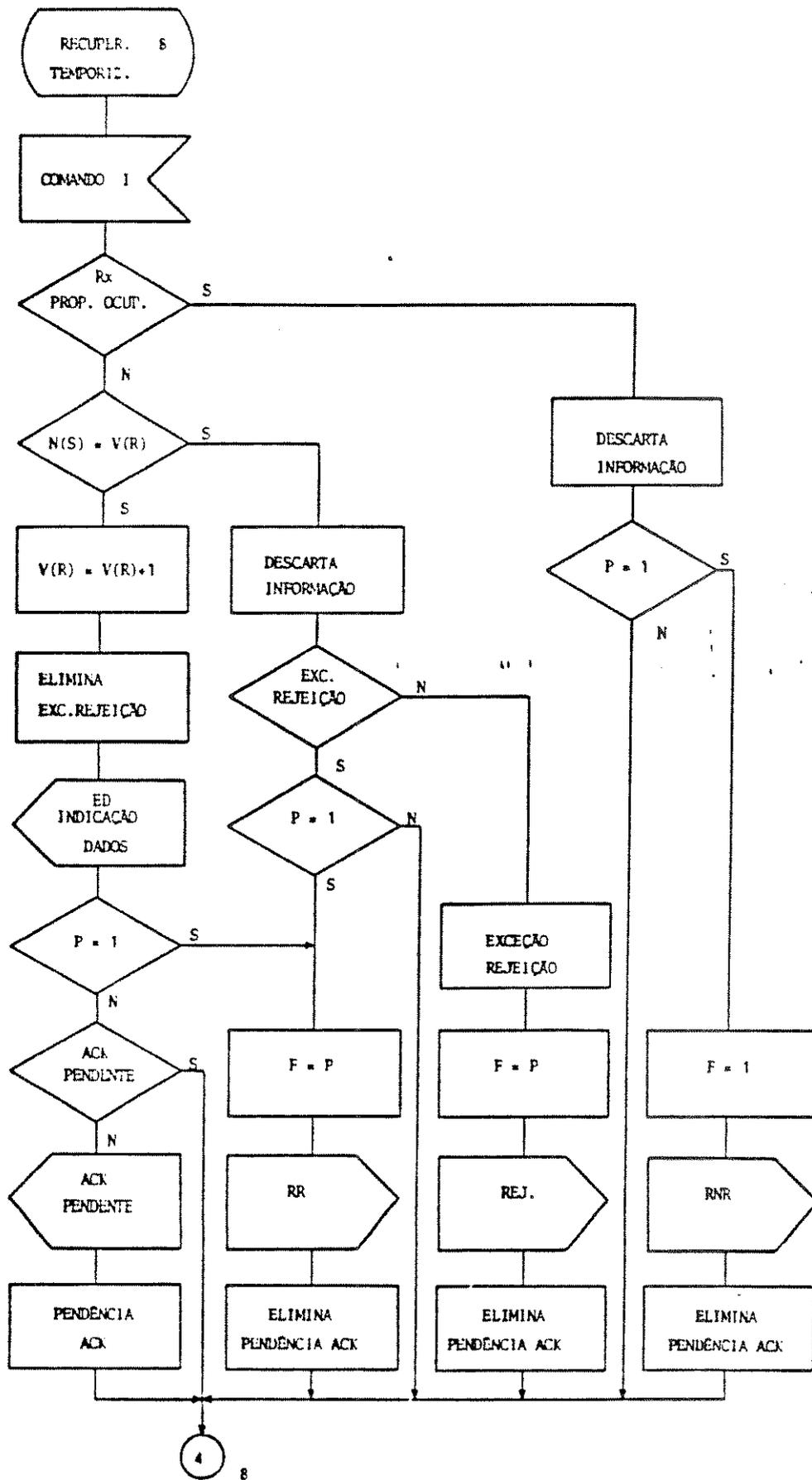
\* GERADO PELA ENTIDADE DE GERENCIAMENTO DA CONEXÃO

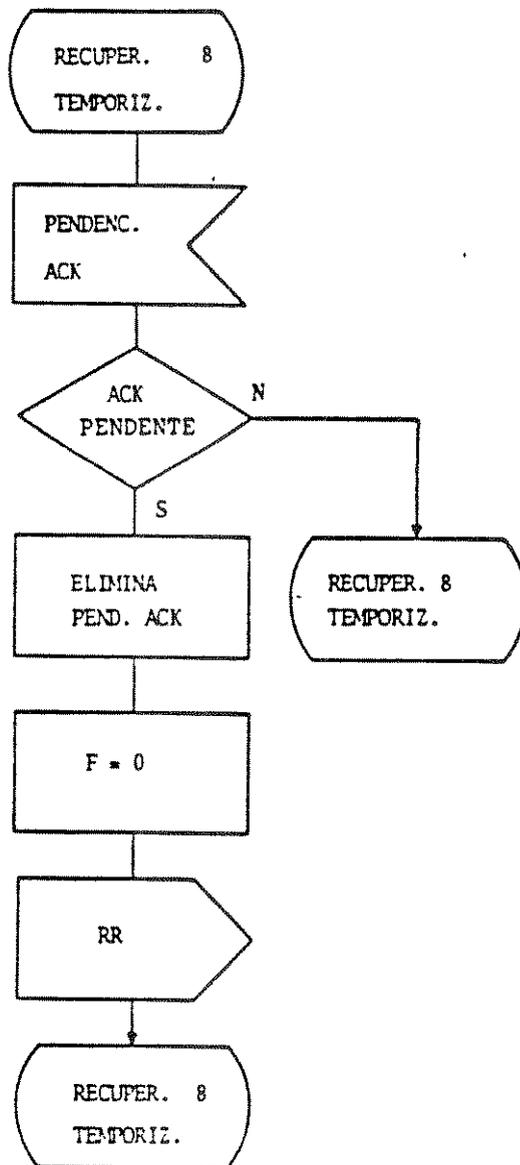
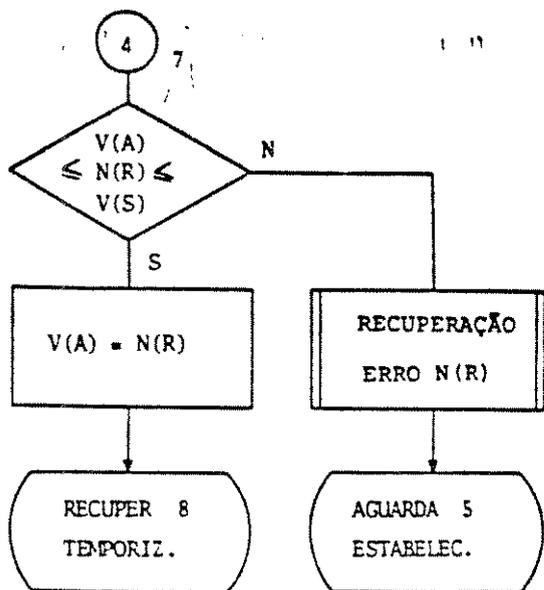
<sup>24</sup> Anexo I: Recuperação por Temporização d

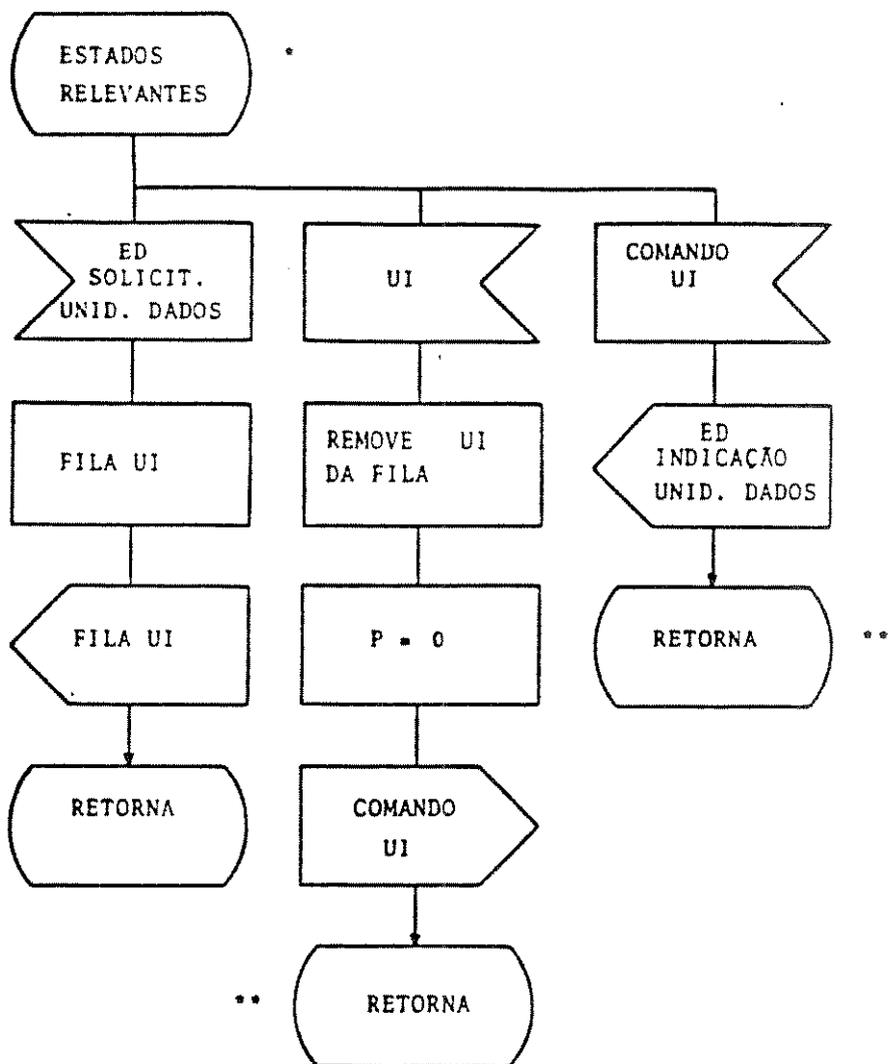




26 Anexo 1 : Recuperação por Temporização f







\* ESTADO RELEVANTE:

- 4 - TEI DESIGNADO;
- 5 - AGUARDANDO ESTABELECIMENTO
- 6 - AGUARDANDO LIBERAÇÃO
- 7 - QUADROS MÚLTIPLOS ESTABELECIDOS
- 8 - RECUPERAÇÃO POR TEMPORIZAÇÃO

\*\*\* RETORNA AO ESTADO INICIAL

<sup>2º</sup> Anexo I : Estados Relevantes e

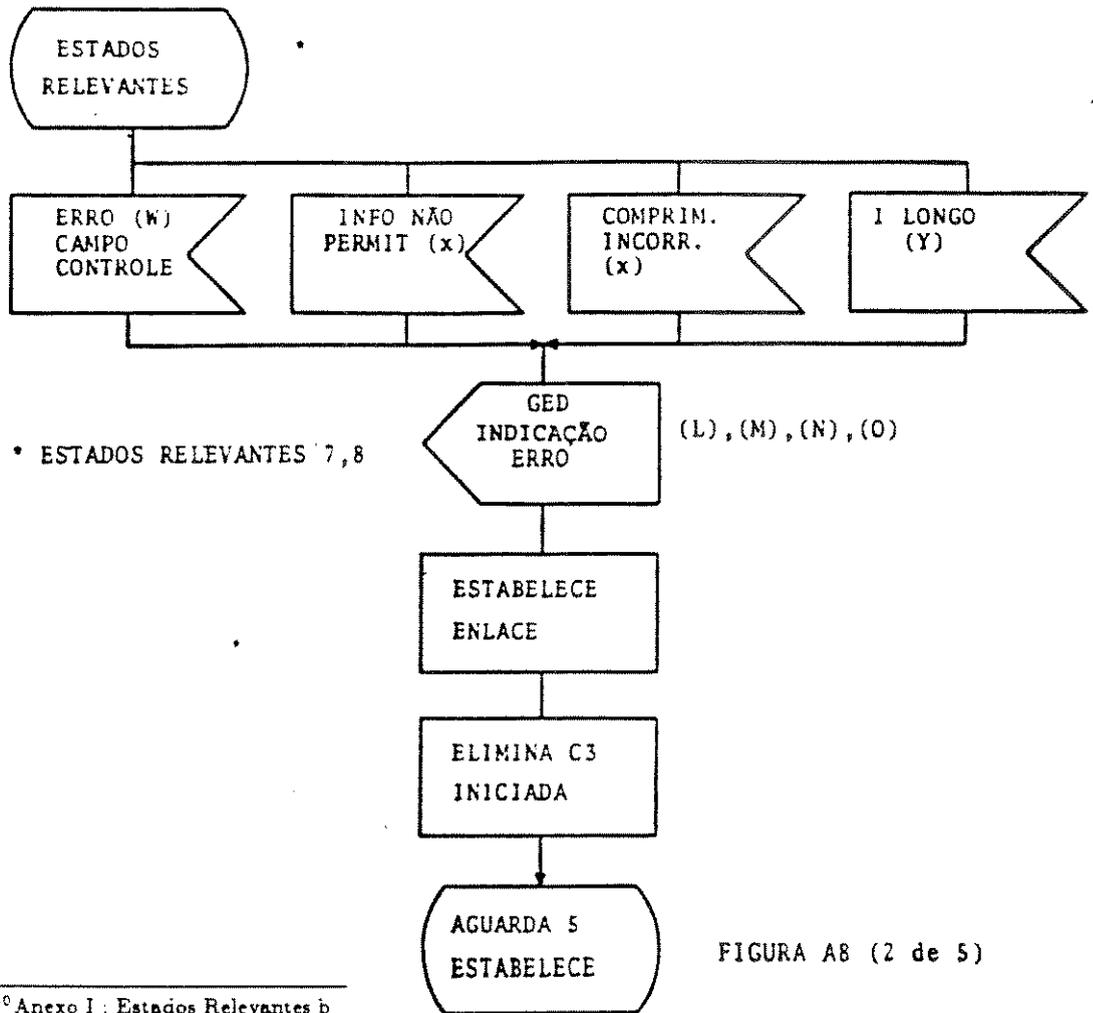
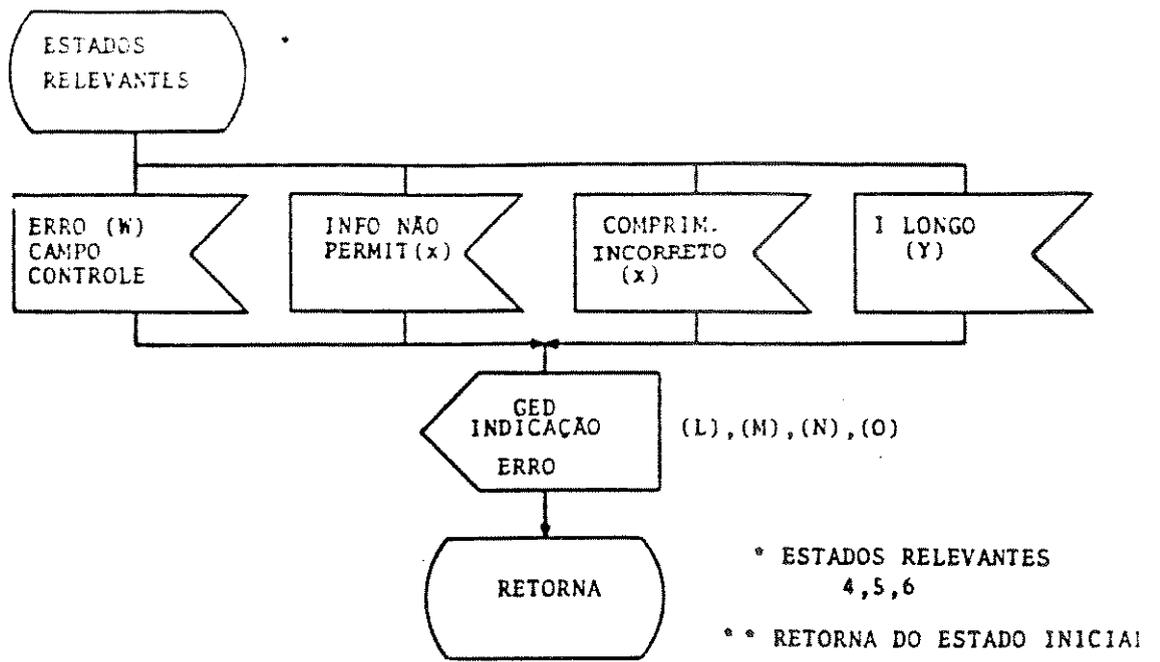
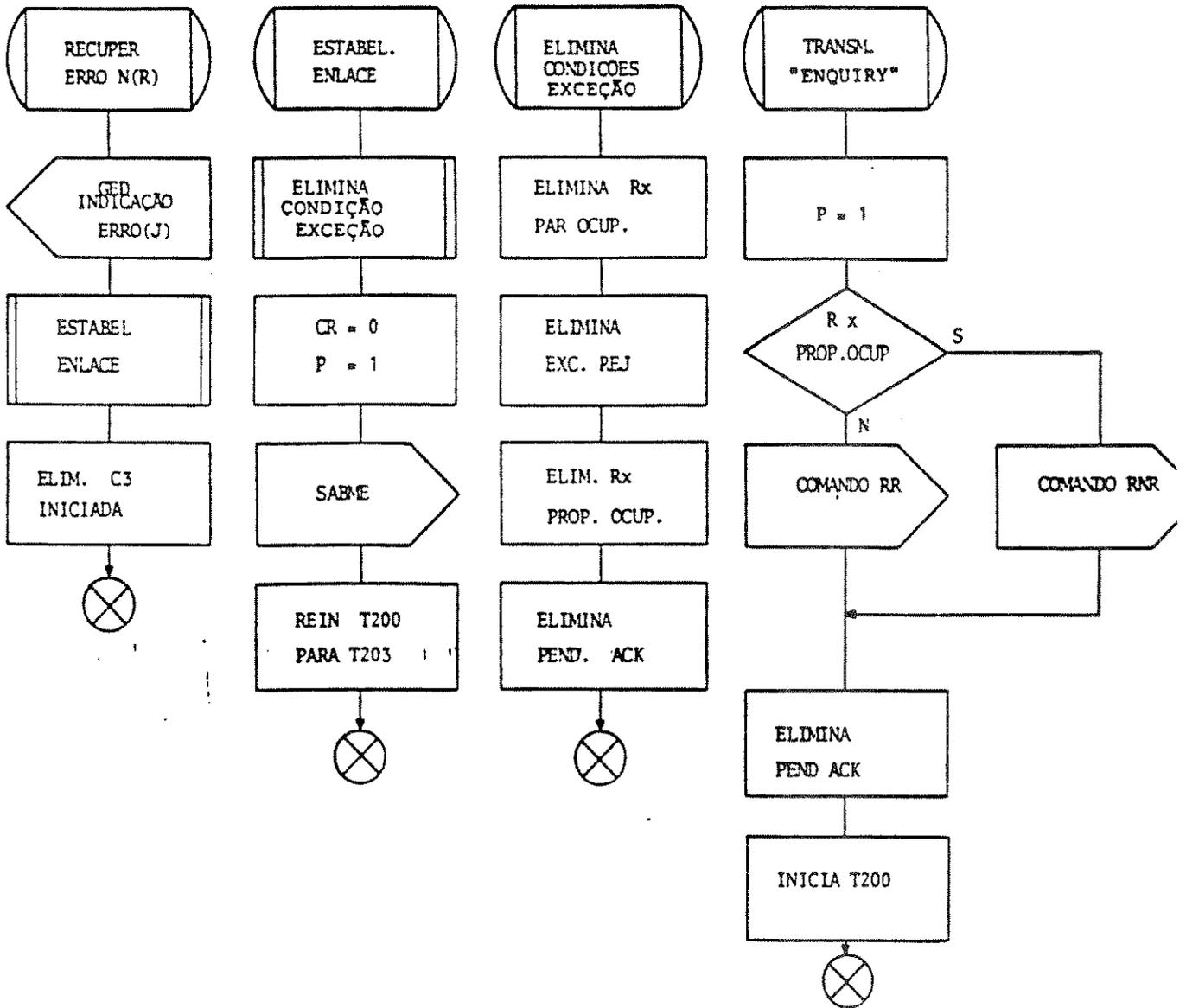
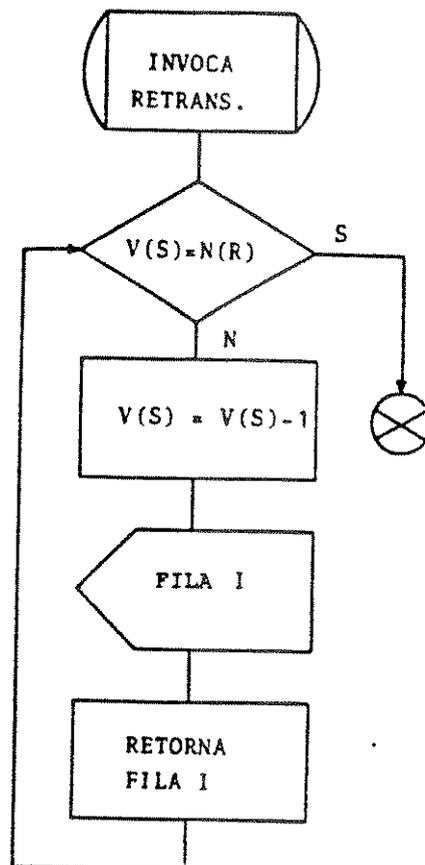
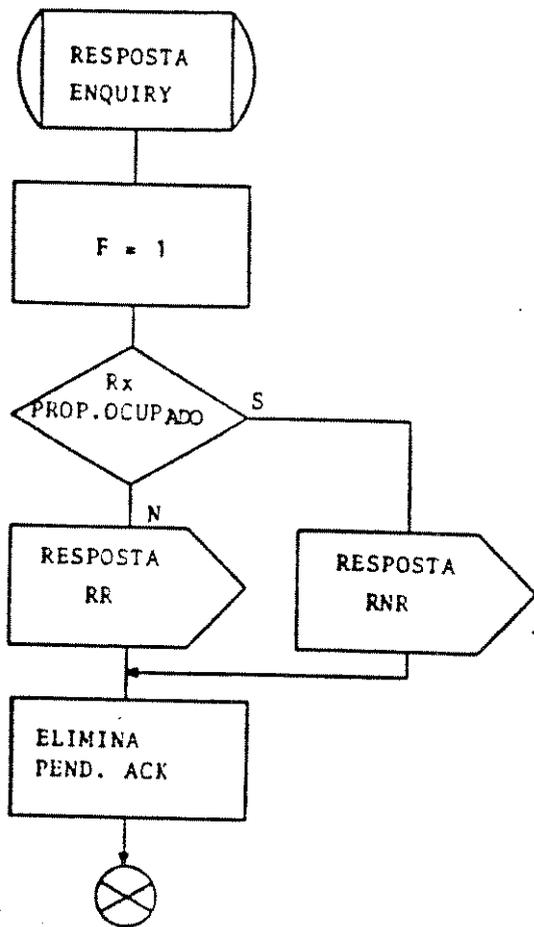


FIGURA A8 (2 de 5)

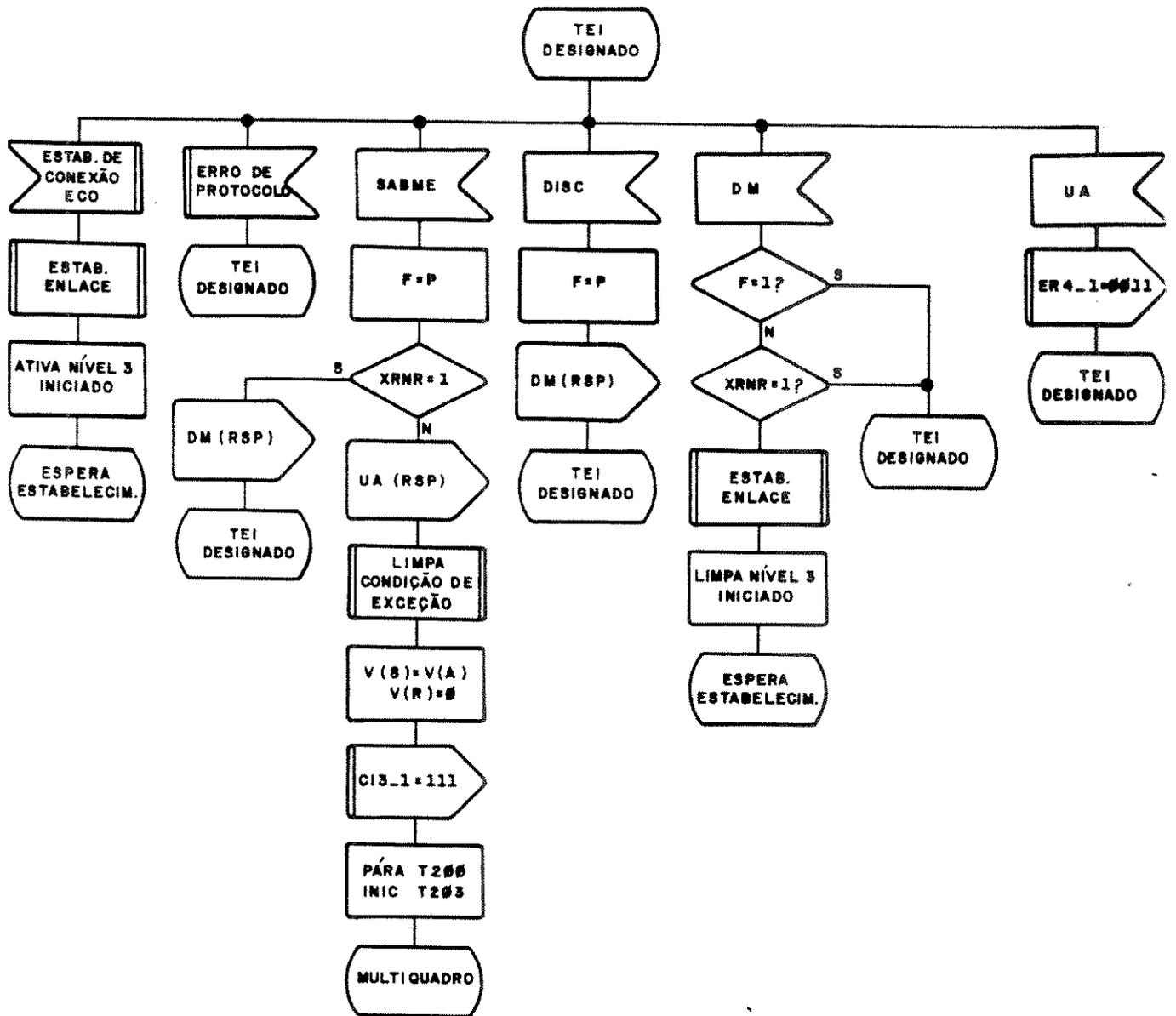
<sup>30</sup> Anexo I : Estados Relevantes b

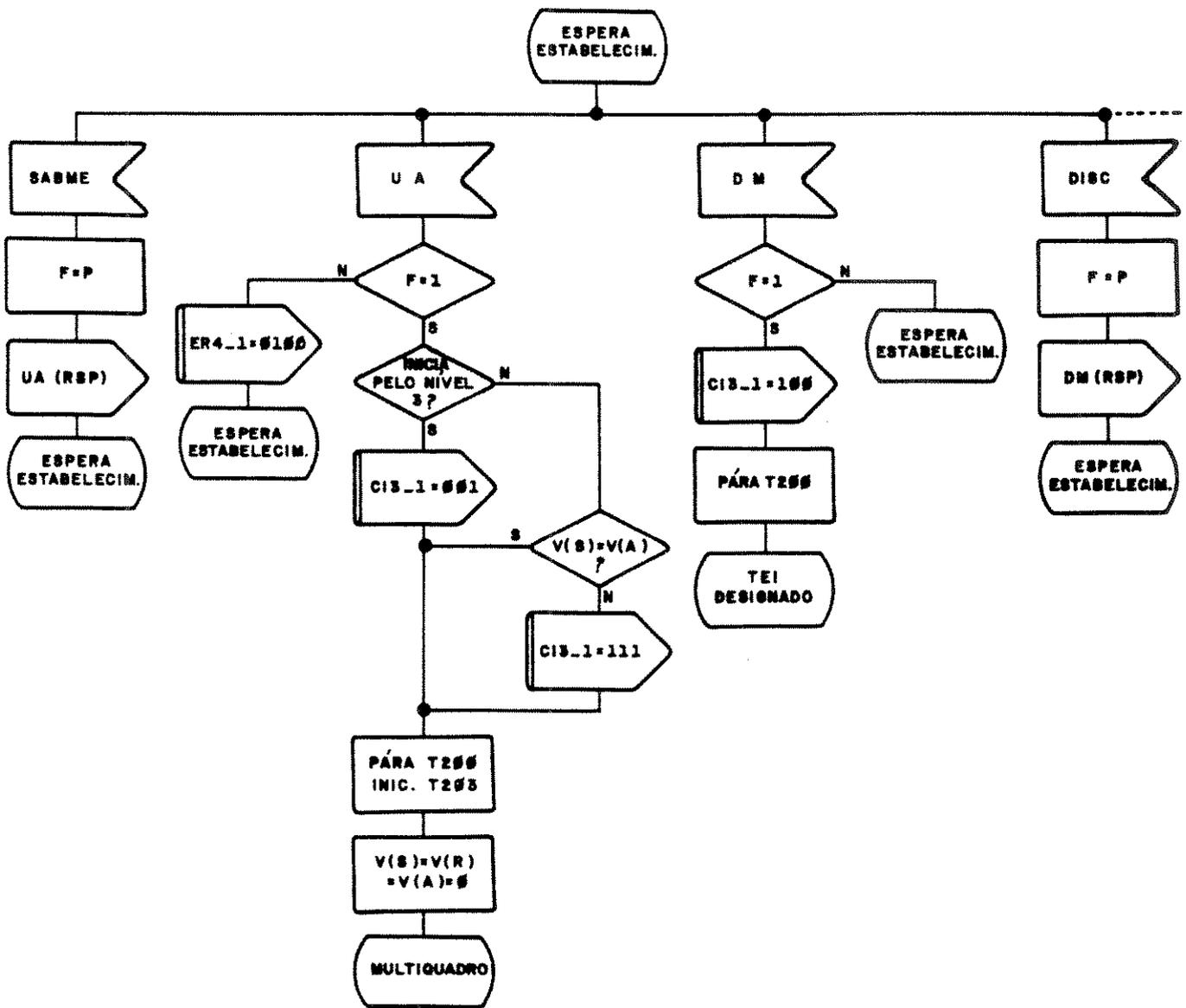


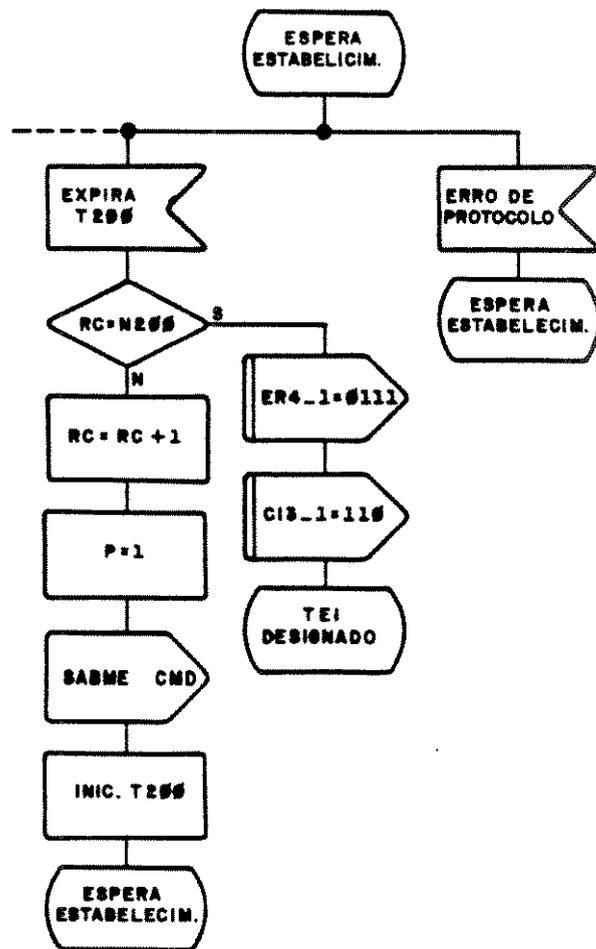


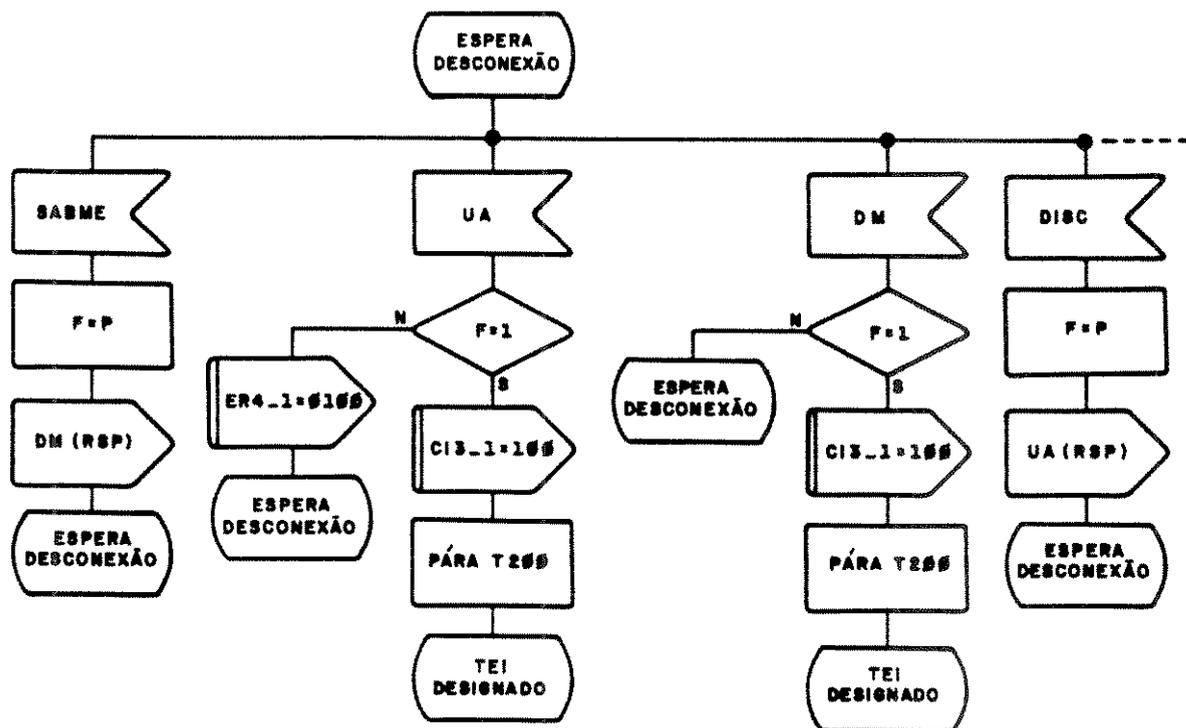
## Anexo II

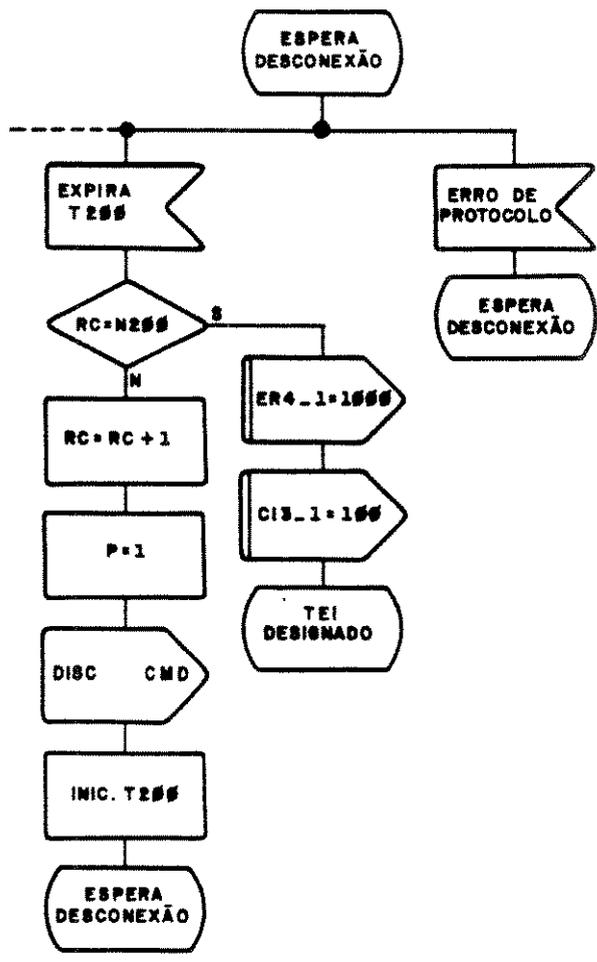
LEDS da implementação do LAPD no CH2

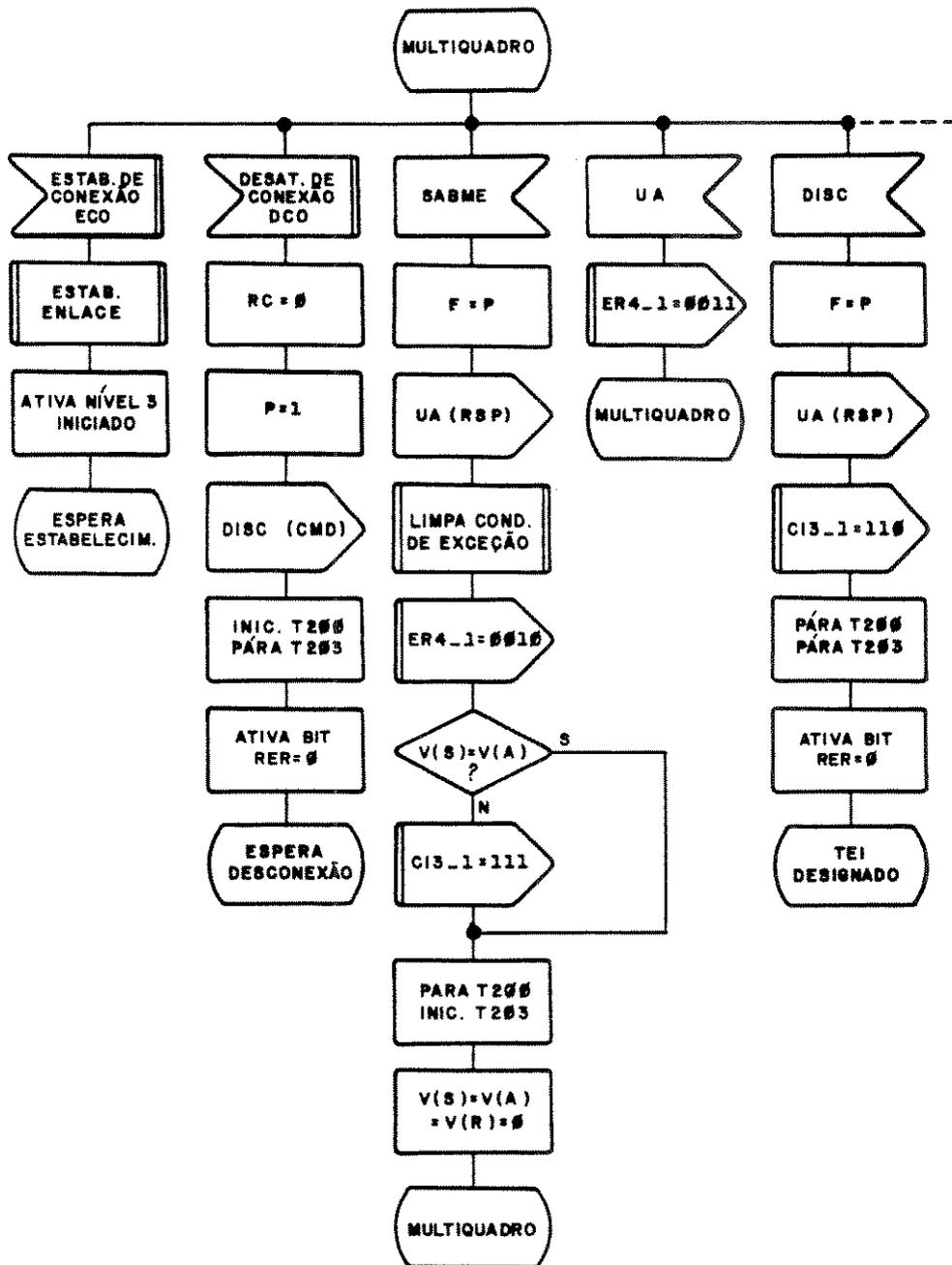


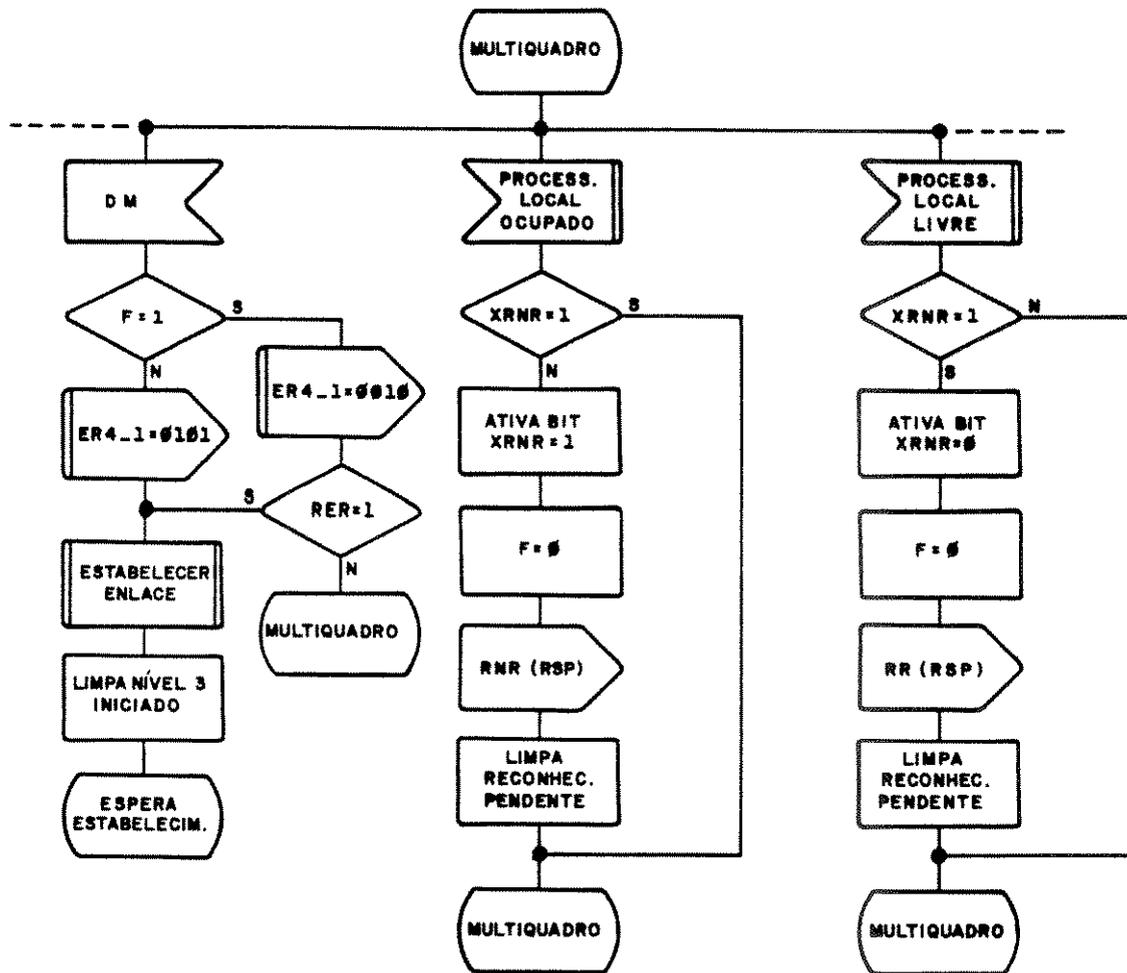




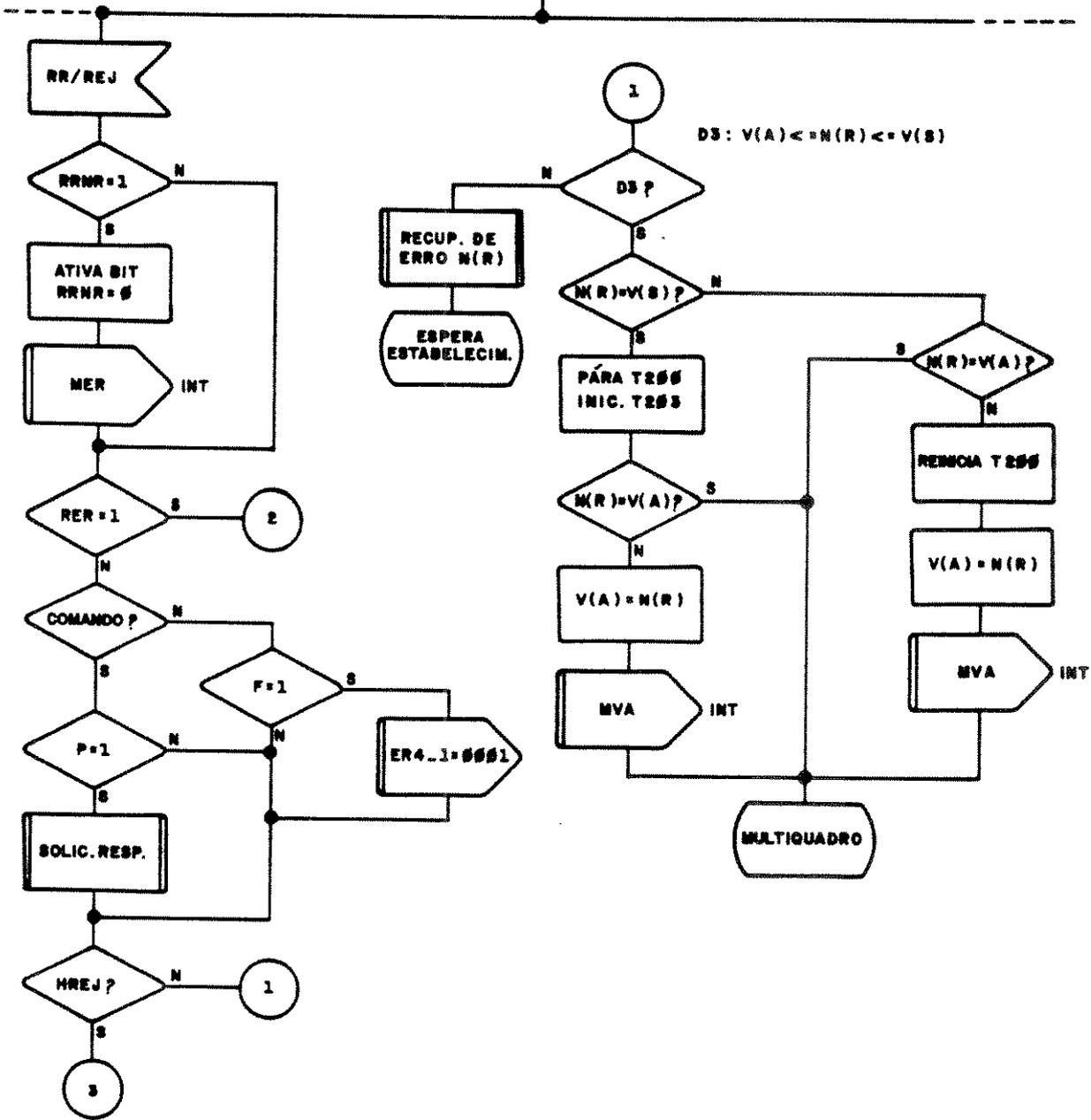




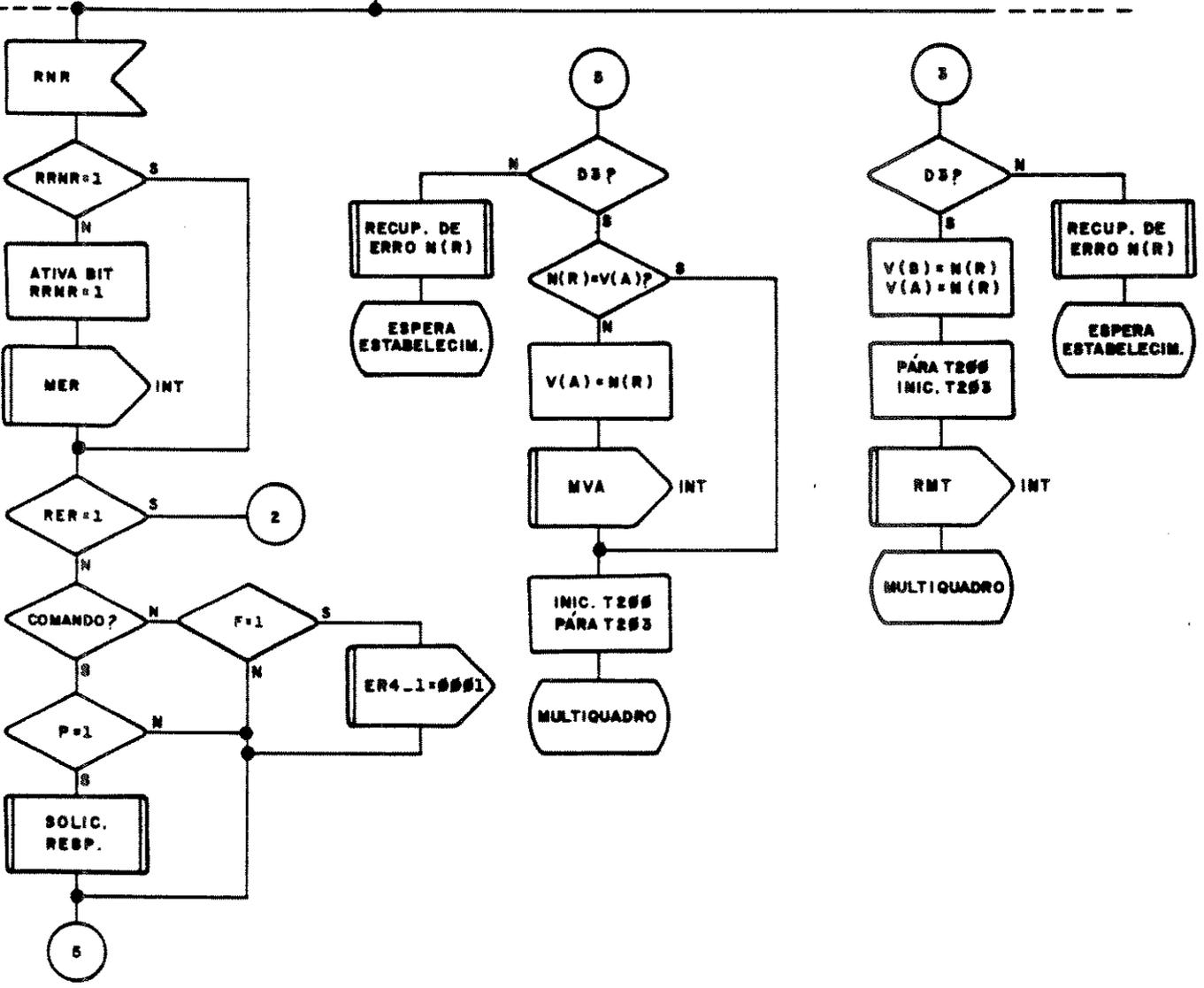




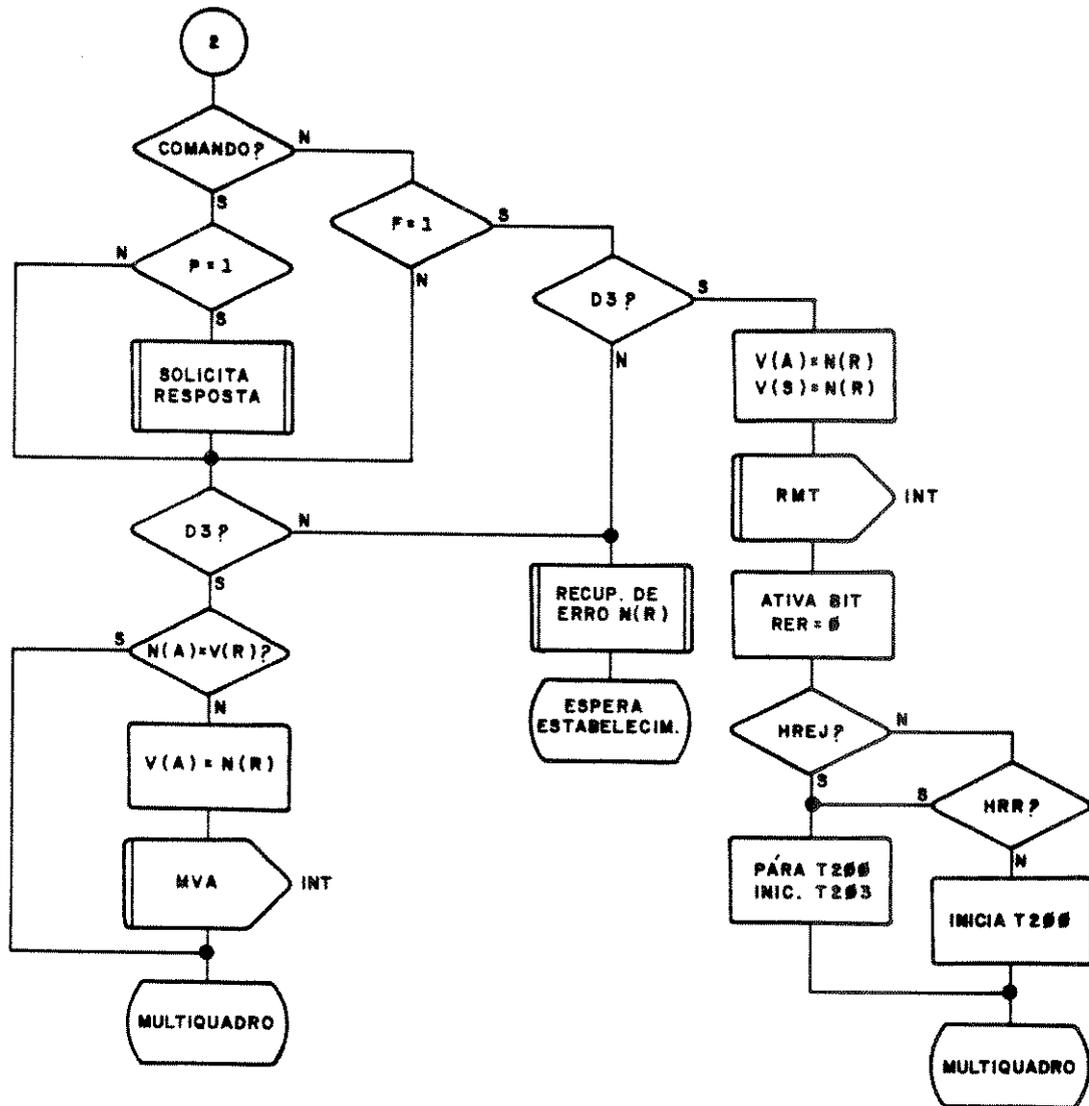
MULTIQUADRO



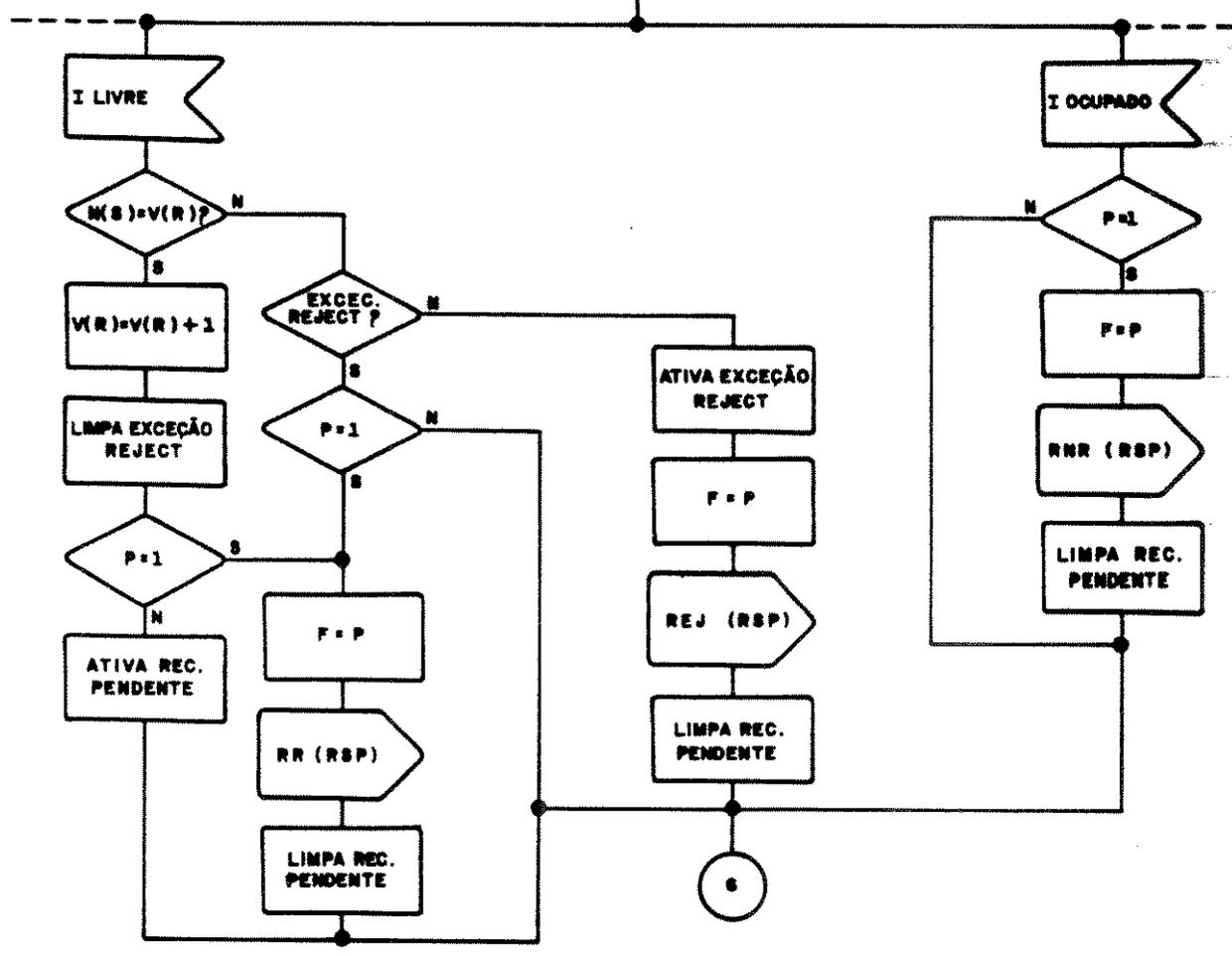
MULTIQUADRO



MULTIQUADRO



MULTIQUADRO



MULTIQUADRO

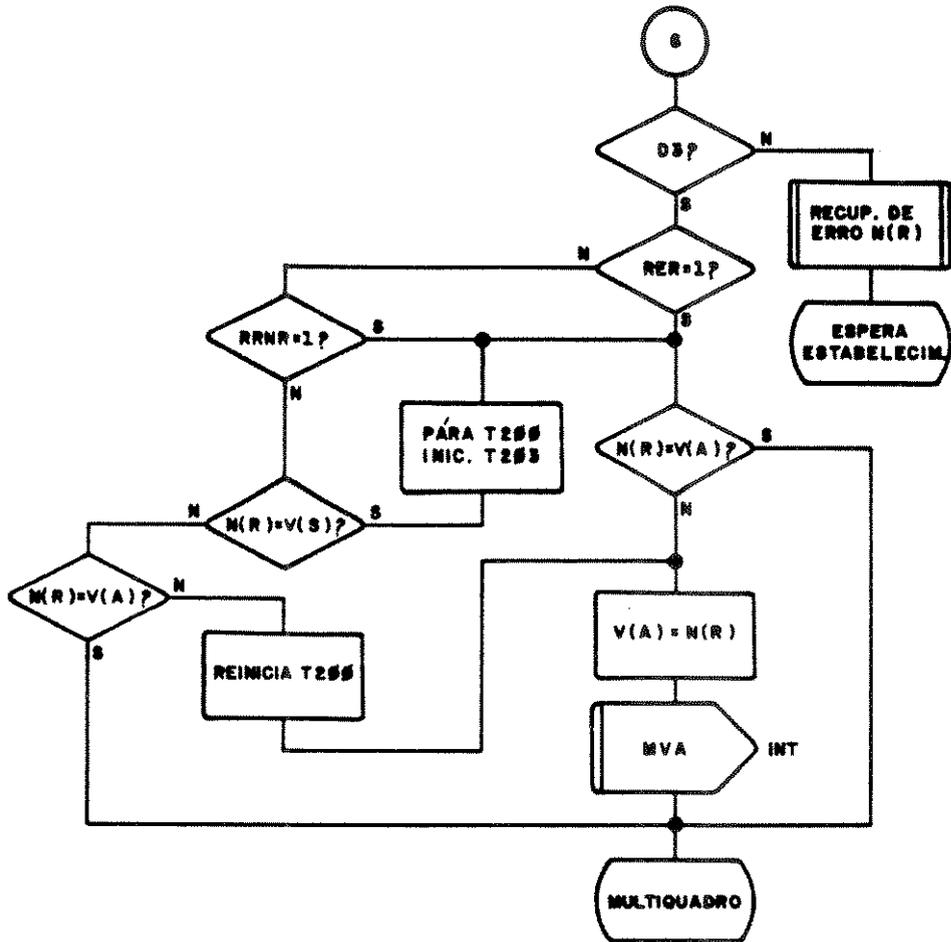
REC. PENDENTE

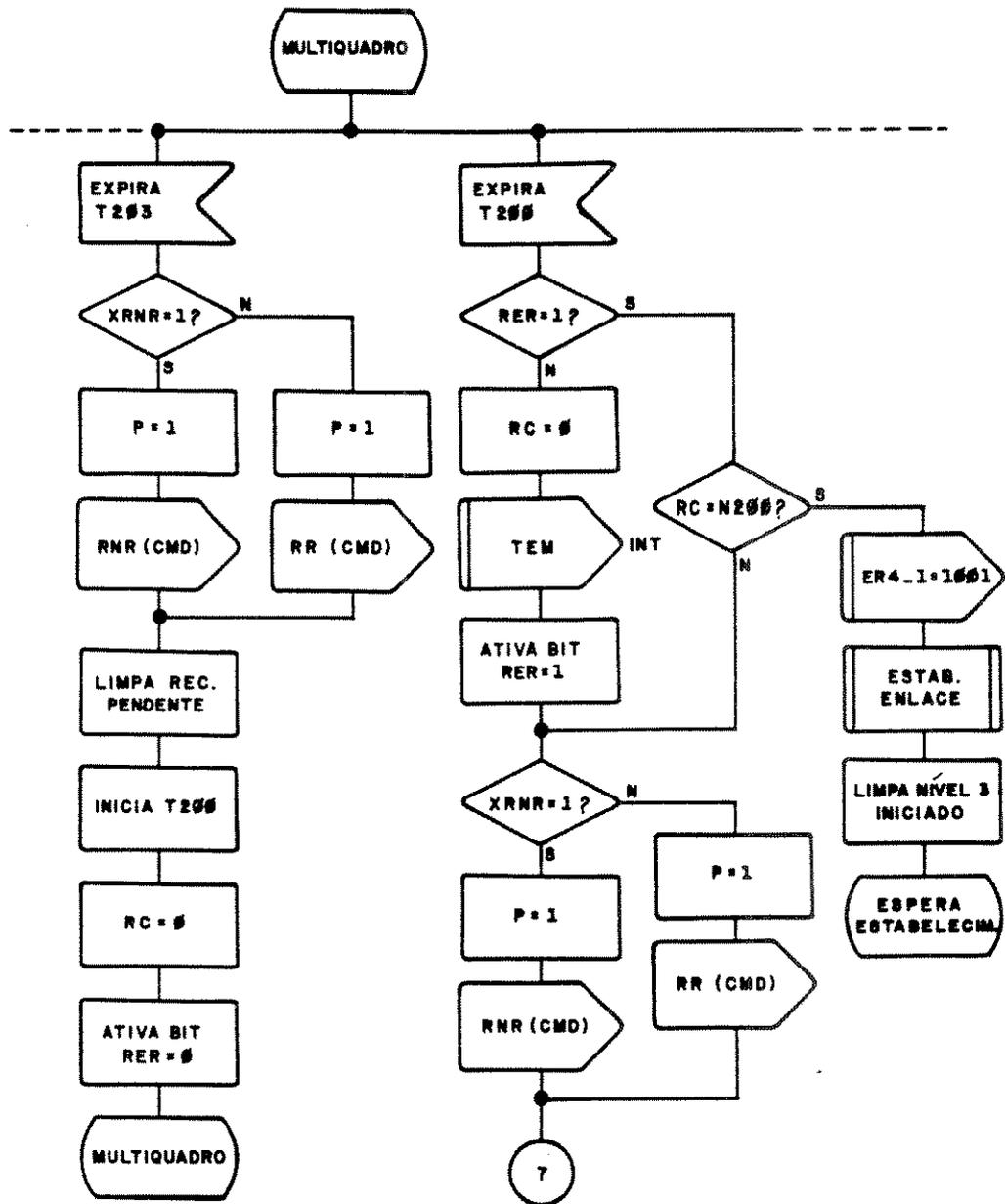
LIMPA REC. PENDENTE

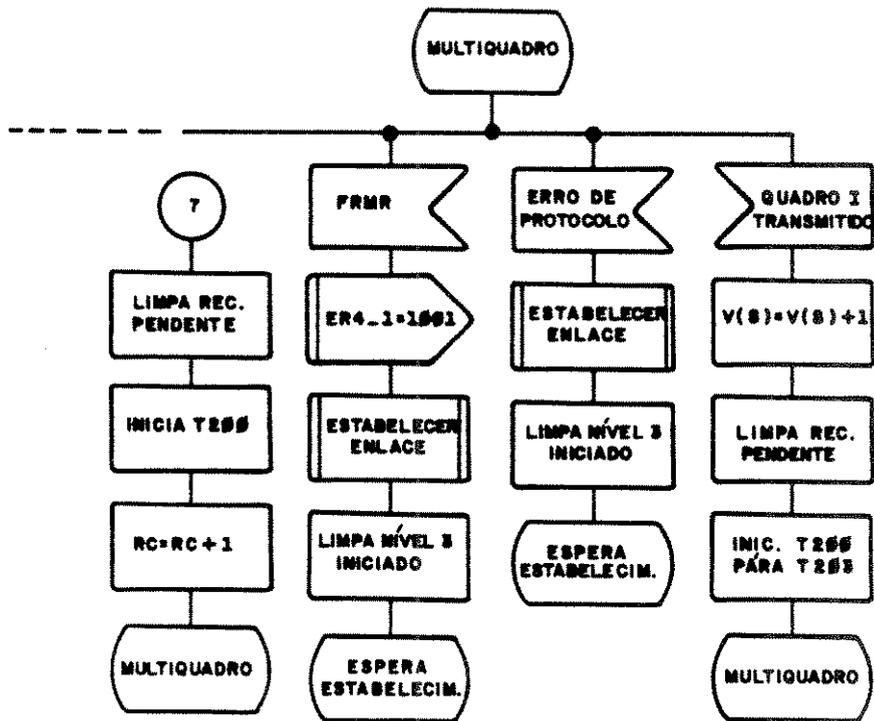
F = 0

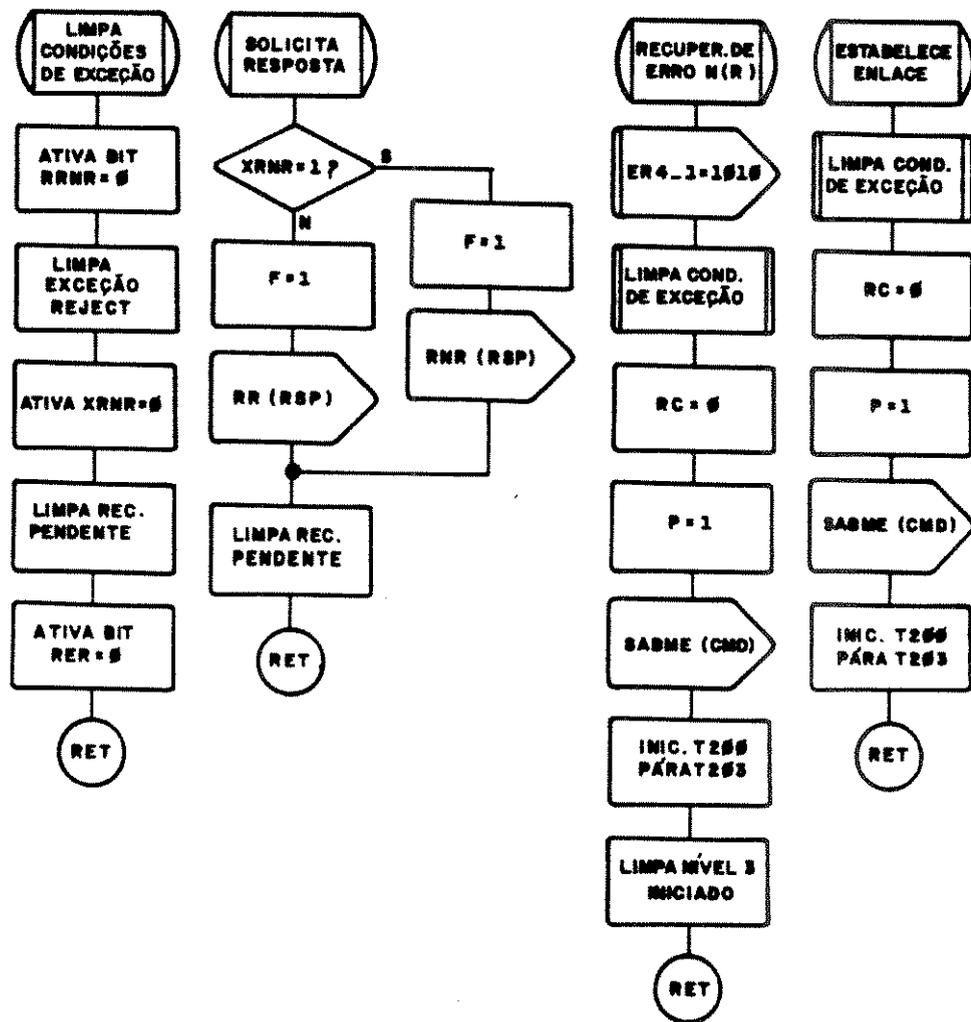
RR (RSP)

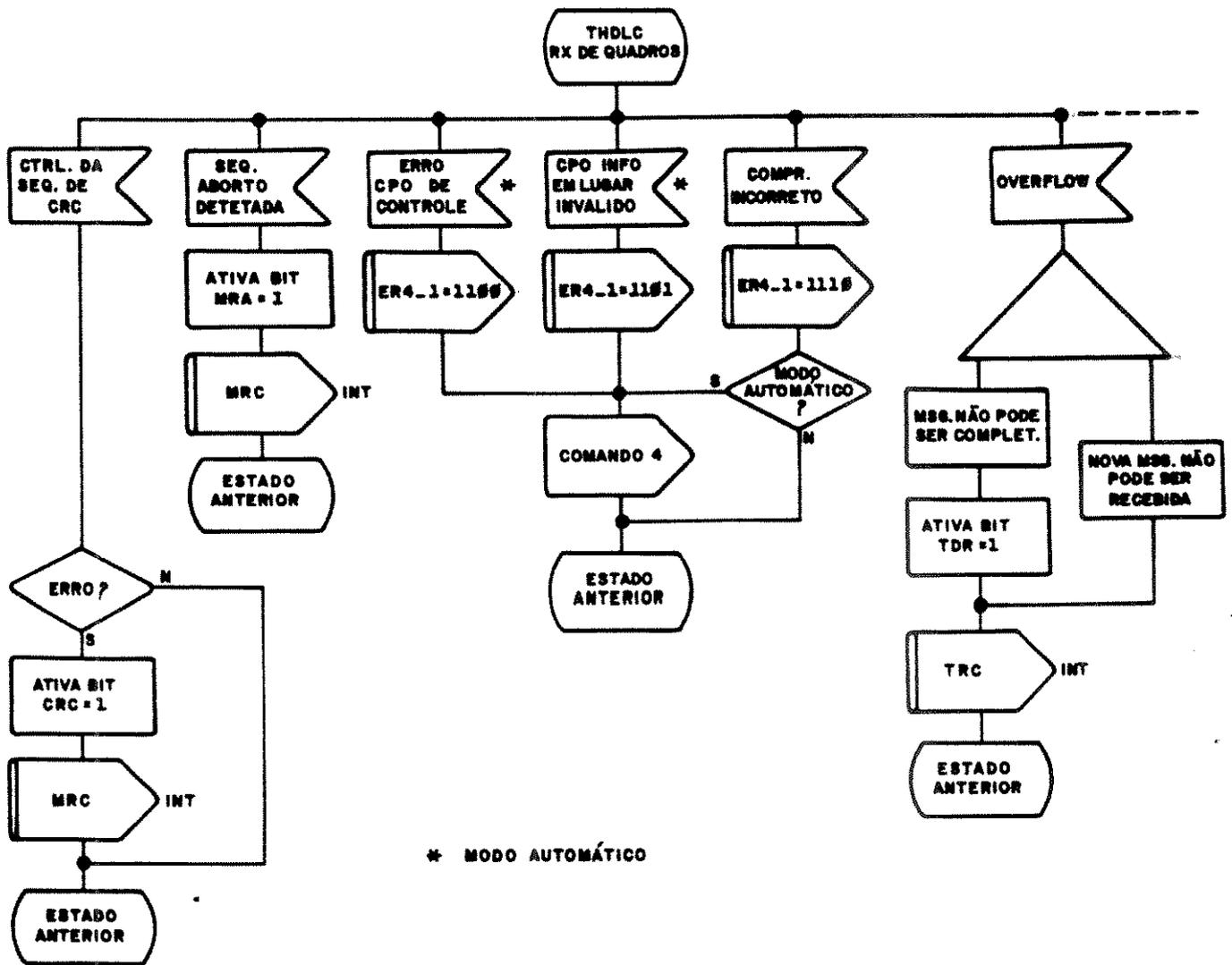
MULTIQUADRO

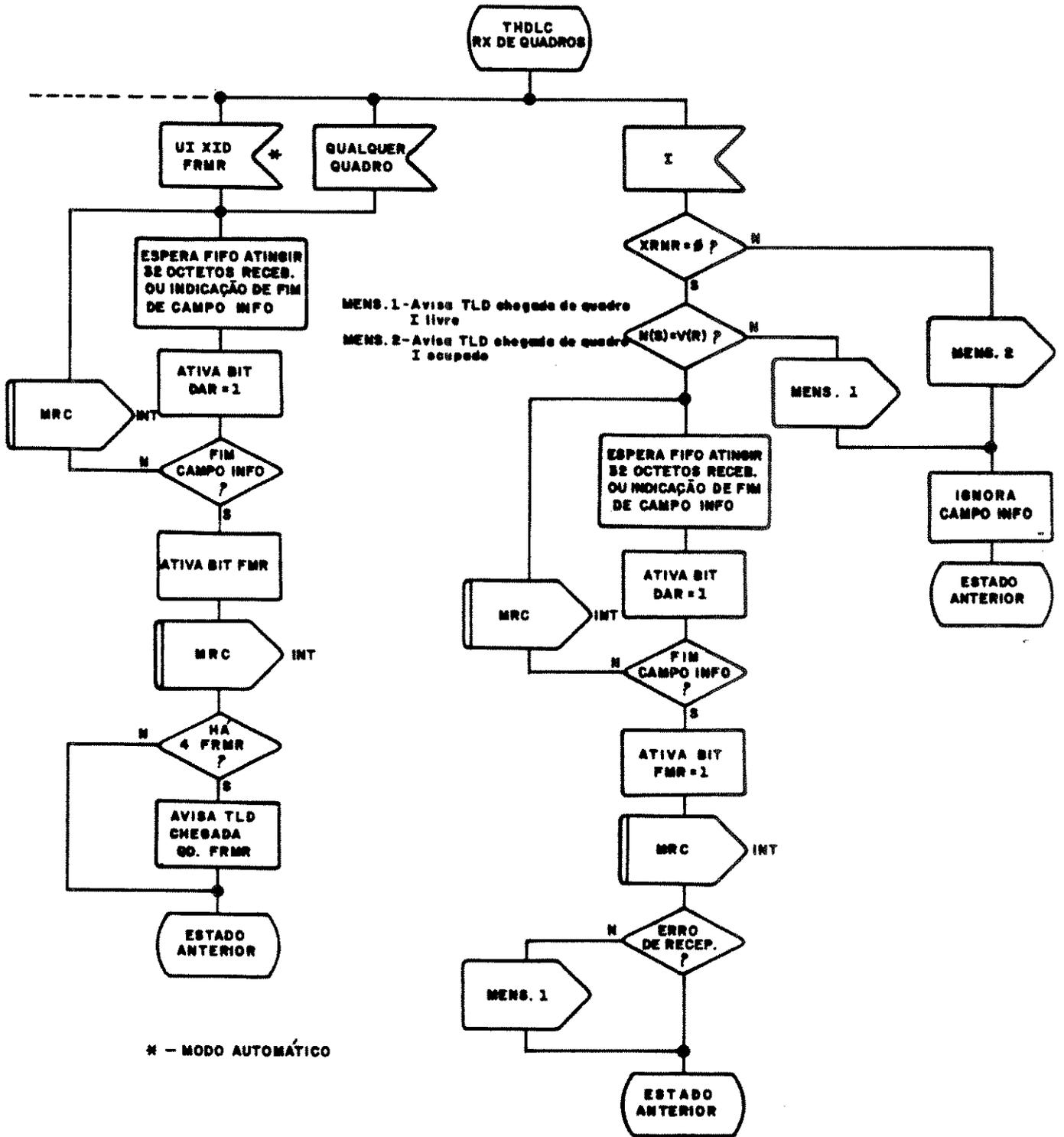


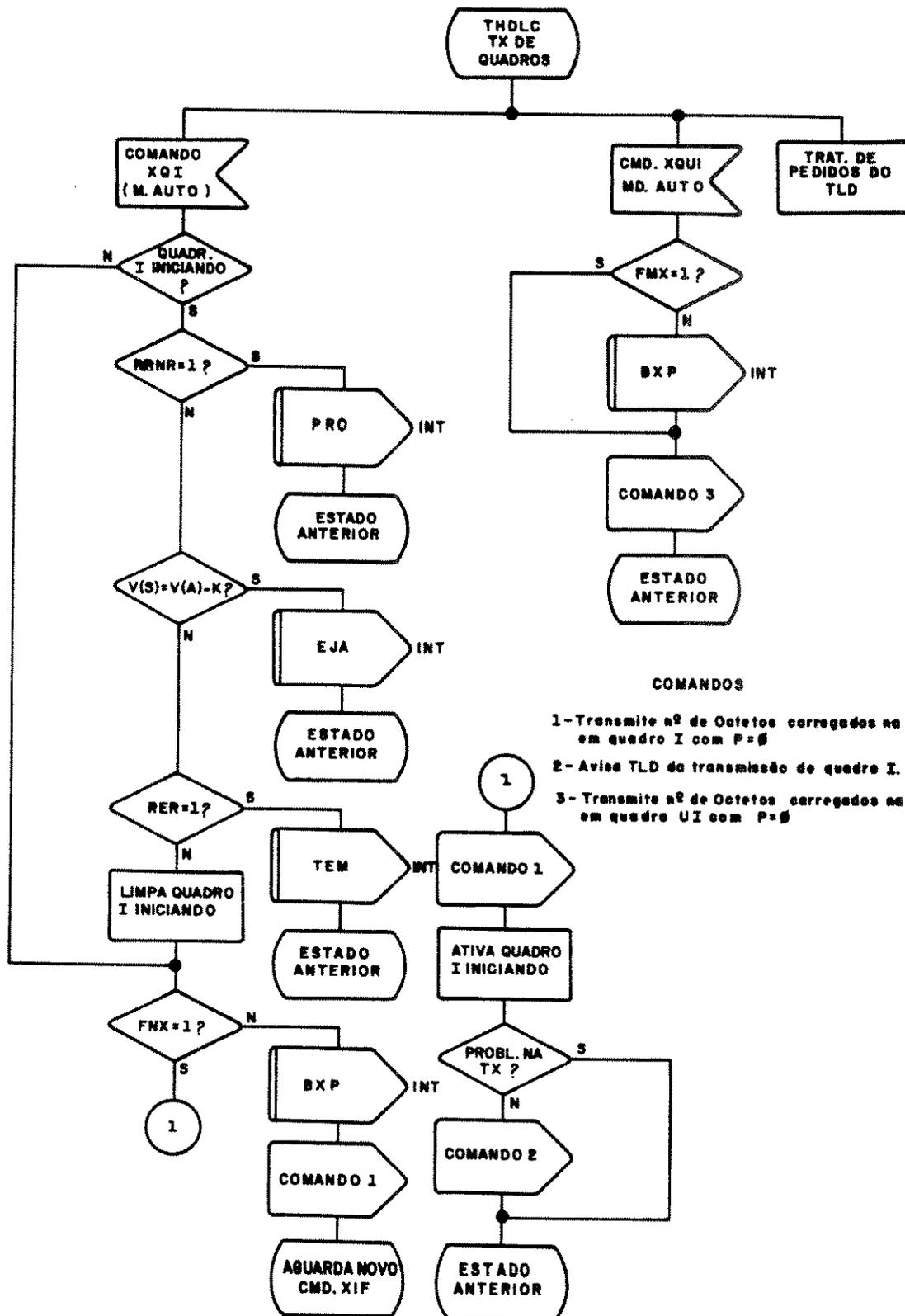












## Anexo III

### Referências Bibliográficas

## 9 Referências Bibliográficas

### **CBT83** CBTT XVIII

Comissão Brasileira de Telefonia e Telegrafia  
Fixação de Conceitos sobre a RDSI - 1983

### **CCT85** CCITT

Red Book - Volume III - Fascículo III.5  
Recommendations of the Series I - 1985

### **COSTA87** Eduardo Moreira da Costa

Introdução ao Projeto de Circuitos Integrados Dedicados  
II EBAI - 1987

### **EBLF** Eugênio Benito Jr. e Luís Roberto Ferreira

CPqD Telebrás - Relatório Interno de Projeto  
Descrição do Bloco TLD - Primeira Sugestão

### **EBLF87a** Eugênio Benito Jr. e Luís Roberto Ferreira

Camada de Enlace da RDSI - Uma Solução VLSI  
Anais do V Simpósio Brasileiro de Telecomunicações  
Campinas - setembro de 1987

### **EBLF87b** Eugênio Benito Jr. e Luís Roberto Ferreira

A Tecnologia VLSI e a Rede Digital de Serviços Integrados  
Anais do XX Congresso Nacional de Informática  
São Paulo - setembro de 1987

### **EBLF88** Eugênio Benito Jr. e Luís Roberto Ferreira

Uma Solução VLSI para o Protocolo LAPD  
Anais do XXI Congresso Nacional de Informática  
Rio de Janeiro - setembro de 1988

### **FSP** Luís Roberto Ferreira, J R G Santos e M A Peres

CPqD Telebrás - Relatório Interno  
Divisão do Bloco de Interfaces do TB22

**JWC87** James W. Conard

ISDN: A Global Solution to Transmission Needs  
Data Communications - winter 1987

**KANO86** Sadahiko Kano

Layer 2 and 3 ISDN Recommendations  
IEEE Journal on Select Areas in Communications  
Vol. sac-4, Núm. 3 - maio 1986

**LERACH86** Lothar Lerach

ISDN - Oriented Modular VLSI Chip Set for Central Office and PABX  
Applications  
IEEE Journal on Select Areas in Communications  
Vol. sac-4, Núm. 8 - novembro 1986

**LFB88** Luiz F. B. Baptistella

Rede Digital de Serviços Integrados  
Edição EBAI - 1988

**SIEM86** Siemens A. G.

IOM - ISDN Oriented Modular Interface  
Edição outubro de 1986

**SLM87** Steven L. Martin

ISDN ICs move toward single-chip multilayer support  
Computer Design - may 1987

**SOT87** SOTELEC

Commutation and Transmission  
ISDN Report - 1987

**STB** Telecomunicações Brasileiras S/A

Sistema de Práticas Telebrás - Série Engenharia  
Práticas para a Experiência Piloto Brasileira em RDSI

**TAR86** Liane M. R. Tarouco

Redes de Computadores - Locais e de Longa Distância  
Mc Graw-Hill - 1986

## Agradecimentos

Uma relação bastante extensa de pessoas contribuíram durante o processo de elaboração deste trabalho de forma que não seria possível citá-las todas. No entanto, sem me esquecer das demais, gostaria de deixar minha especial gratidão à chefia e a todos os componentes da Coordenação de Áreas de Circuitos Integrados e a elementos de outros grupos que conosco trabalham no CPqD Telebrás, pelo muito que me incentivaram. Em particular agradeço ao companheiro Eugênio Benito Júnior com quem trabalhei mais de perto. Agradeço ainda ao esforço e paciência de meu orientador, apesar dos meus atrasos de cronograma e incontáveis prorrogações na entrega de resultados, aos colegas professores do Departamento de Ciência da Computação da Unicamp que não permitiram que eu desistisse e aos amigos (em particular ao Júlio) que entenderam porque, em inúmeras ocasiões, não pude dar-lhes a devida atenção que certamente mereciam.

Finalmente agradeço à minha mãe, Arinda, a quem devo toda a minha formação.