

UNIVERSIDADE FEDERAL DA PARAIBA
CENTRO DE CIENCIA E TECNOLOGIA


CURSO DE MESTRADO EM ENGENHARIA ELETRICA

MODELAGEM E ESTUDO DE DESEMPENHO DO PROTOCOLO FDDI
VIA
FERRAMENTA RESQ2

Sérgio de Figueirêdo Brito

Campina Grande, Pb

Março/1992



MODELAGEM E ESTUDO DE DESEMPENHO DO PROTOCOLO FDDI

VIA

FERRAMENTA RESQ2

Sérgio de Figueirêdo Brito

*Dissertação apresentada ao
Curso de Mestrado em
Engenharia Elétrica da
Universidade Federal da
Paraíba, em cumprimento às
exigências para obtenção do
grau de Mestre.*

William Ferreira Giozza, Dr. Ing.

Orientador

Campina Grande, Pb

Março/1992

MODELAGEM E ESTUDO DE DESEMPENHO DO PROTOCOLO MAC FDDI
VIA FERRAMENTA RESQ2

SERGIO DE FIGUEIREDO BRITO

DISSERTAÇÃO APROVADA EM 20.03.92



WILLIAM FERREIRA GIOZZA, Dr.Ing., UFPB
Orientador



MARIA IZABEL CAVALCANTI CABRAL, D.Sc., UFPB
Componente da Banca



JOSE AUGUSTO SURUAGY MONTEIRO, Ph.D., UFPE
Componente da Banca

CAMPINA GRANDE - PB
MARÇO - 1992

DIGITALIZAÇÃO:
SISTEMOTECA - UFCG

A Deus pela oportunidade e apoio.

Ao caro orientador William Ferrreira Giozza pelo profissionalismo, estímulo e dedicação com que orientou todo o mestrado.

A minha querida esposa Mônica pelo apoio, carinho e compreensão nos momentos mais difíceis.

À equipe do NPD pela atenção e presteza dispensadas aos trabalhos de simulação.

Um agradecimento especial a todos os professores, funcionários, colegas e amigos que contribuíram direta ou indiretamente para este trabalho. Em particular, ao casal de amigos Luciano e Lita que efetivamente ofereceram os recursos e apoio sincero, necessários a conclusão deste trabalho.

SUMARIO

Neste trabalho é proposto um modelo utilizando o pacote de simulação RESQ2 da IBM para a avaliação de desempenho do protocolo de acesso ao meio de redes do padrão FDDI ("*Fiber Distributed Data Interface*").

Inicialmente é apresentada uma visão geral do padrão FDDI, enfocando aspectos de aplicação, arquitetura, topologia e operação do protocolo de controle de acesso ao meio do FDDI. Na sequência, apresenta-se a modelagem do mecanismo de acesso ao meio FDDI através da ferramenta RESQ2. Os resultados obtidos com a modelagem proposta são analisados e comparados com outros resultados existentes de desempenho deste protocolo, obtidos através de ferramentas analíticas e outros simuladores.

Finalizando este trabalho, são tecidas algumas considerações e conclusões acerca do uso da ferramenta RESQ2 e do desempenho do protocolo de acesso ao meio do padrão FDDI.

ABSTRACT

In this work is proposed a model based on IBM's RESQ2 simulation tool for performance evaluation of the FDDI Medium Access Control protocol. FDDI architecture and topology as well as operation of the FDDI MAC protocol are described. Results from the proposed modeling are analysed and compared to others results from performance evaluation of FDDI networks, obtained by analytical and other simulation tools.

INDICE

1. INTRODUÇÃO.....	9
2. REDES PADRAO FDDI	16
2.1 Introdução.....	17
2.2 Topologia das Redes FDDI.....	20
2.3 Sub-camadas PMD e PHY.....	23
2.4 Formatos do Quadro e da Ficha MAC FDDI.....	26
2.5 Sub-camada MAC.....	28
2.6 Otimização FDDI-II.....	32
3. MODELAGEM DO PROTOCOLO MAC FDDI	34
3.1 Introdução.....	35
3.2 A Ferramenta RESQ2	
3.2.1 Representação Gráfica do Modelo.....	44
3.2.2 Estrutura de um Programa RESQ2.....	49
3.3 Modelagem e Construção do Simulador FDDI	
3.3.1 Controle de Liberação de Quadros e Contabilização do Tempo de Resposta.....	61
3.3.2 Inicialização dos Quadros e Criação/Inicialização da Ficha.....	63
3.3.3 Mecanismo de Controle de Acesso ao Meio Baseado na Posse da Ficha.....	67
3.3.4 Controle de Transmissão e Propagação.....	78

3.3.5 Controle de Destino do Quadro/Ficha.....	81
3.3.6 Estação de Controle da Ficha.....	83
4. ESTUDO DE DESEMPENHO DO PROTOCOLO FDDI	85
4.1 Introdução.....	86
4.2 Elementos para Estudo de Desempenho do FDDI.....	87
4.3 Outros Estudos de Desempenho de Redes FDDI.....	92
4.4 Resultados Obtidos com o Simulador RESQ2.....	96
5. CONCLUSÕES.....	105
REFERENCIAS BIBLIOGRAFICAS.....	109
ANEXO I - DIAGRAMAS DO MODELO FDDI EM RESQ2	
I.1 Diagrama da Estação Síncrona.....	118
I.2 Diagrama da Estação Assíncrona.....	119
I.3 Diagrama da Estação de Controle da Ficha.....	120
ANEXO II - ARQUIVOS DE PROGRAMA PARA SIMULAÇÃO	
II.1 Arquivo de Sub-modelos.....	121
II.2 Arquivo de Rede para Tabela 4.1.....	127
II.3 Arquivo de Rede para Tabela 4.2.....	131
II.4 Arquivo de Rede para Tabela 4.3.....	135
ANEXO III - SIMULAÇÃO E OBTENÇÃO DE ESTATISTICAS	
III.1 Passos para Simulação de Uma Configuração.....	139
III.2 Memorial de Cálculo para Obtenção das Estatísticas Resumidas.....	140

LISTA DE FIGURAS E TABELAS

CAPITULO 1

Figura 1.1 - Topologia em anel duplo do padrão FDDI.....13

CAPITULO 2

Figura 2.1 - Mapeamento da arquitetura FDDI
no modelo OSI.....19

Figura 2.2 - Detalhamento da topologia FDDI.....20

Figura 2.3 - Topologia FDDI com possíveis configurações
das estações.....22

Figura 2.4 - Formato do QUADRO e da FICHA.....27

Figura 2.5 - Configuração interna da estação FDDI-II.....32

CAPITULO 3

Figura 3.1 - Funcionamento de um modelo
de rede de filas.....36

Figura 3.2 - Topologias de ACTIVE QUEUES
e tradução RESQ2.....45

Figura 3.3 - Topologias das PASSIVES QUEUES
e tradução RESQ2.....47

Figura 3.4 - Símbolos dos NODES e tradução RESQ2.....48

Figura 3.5 - Estrutura de programa RESQ2
e modo de execução.....50

Figura 3.6 - Modelagem genérica de uma estação
do anel FDDI.....56

Figura 3.7 - Diagrama de Blocos de uma estação MAC FDDI.....	56
Figura 3.8 - Diagrama completo da estação SINCRONA em RESQ2.....	57
Figura 3.9 - Estrutura do programa de simulação do protocolo FDDI/RESQ2.....	59
Figura 3.10 - Bloco Controle de Liberação de Quadros e Contabilização do Tempo de Resposta.....	62
Figura 3.11 - Seção do programa RESQ2 do Bloco Controle de Liberação de Quadros e Contabilização do Tempo de Resposta.....	62
Figura 3.12 - Bloco Inicialização dos Quadros e Criação /Inicialização da Ficha.....	64
Figura 3.13 - Seção do programa RESQ2 do Bloco Inicia- lização dos Quadros e Criação/ Inicialização da Ficha.....	64
Figura 3.14 - Bloco MAC da Estação SINCRONA.....	70
Figura 3.15 - Seção do programa RESQ2 do Bloco MAC da Estação SINCRONA.....	71
Figura 3.16 - Bloco MAC da Estação ASSINCRONA.....	75
Figura 3.17 - Seção do programa RESQ2 do Bloco Implementação da MAC Estação ASSINCRONA...	75
Figura 3.18 - Bloco Controle de Transmissão e de Propagação.....	78
Figura 3.19 - Seção do programa RESQ2 do Bloco Controle de Transmissão e de Propagação.....	79
Figura 3.20 - Bloco Controle de Destino do QUADRO/FICHA..	81

Figura 3.21 - Seção do programa RESQ2 do Bloco Controle de Destino do QUADRO/FICHA.....	81
Figura 3.22 - Estação de Controle da Ficha.....	84
Figura 3.23 - Seção do programa RESQ2 da Estação de Controle da Ficha.....	84

CAPITULO 4

Figura 4.1 - Atraso de Quadro Médio x Carga Oferecida....	98
Figura 4.2 - Atraso de Acesso x Carga Oferecida.....	98
Figura 4.3 - Utilização do Anel x Carga Oferecida.....	99
Figura 4.4 - Atraso de Acesso por estação para carga de 113% no exemplo 3.....	102
Figura 4.5 - Perdas de quadros percentual por estação para carga de 113% no exemplo 3.....	103
Tabela 4.1 - Parâmetros de simulação do exemplo 1.....	98
Tabela 4.2 - Parâmetros de simulação do exemplo 2.....	100
Tabela 4.3 - Parâmetros de simulação do exemplo 3.....	100

CAPITULO 1

INTRODUÇÃO

As **Redes Locais** de computadores [GIOZZA,1986] [MOURA,1986] surgiram como uma das soluções para os problemas de interconexão e compartilhamento de recursos computacionais (hardware e software) dispersos a nível local. Em seu estágio inicial, como toda nova tecnologia, a relação custo/benefício não era muito atraente, razão pela qual, os investimentos por parte dos usuários não foram muito entusiásticos. Entretanto, à medida em que houve uma evolução do parque instalado de equipamentos em termos de capacidade de armazenamento, especialização de aplicações, velocidade de processamento das máquinas e processamento distribuído, as Redes Locais puderam mostrar todo o seu potencial e a sua aplicabilidade.

Hoje em dia, dentro das organizações (empresas, instituições, etc), a aplicabilidade das Redes Locais está associada à demanda de serviços cada vez mais sofisticados e exigentes. Dentre as aplicações típicas, nesse contexto, tem-se a transferência de grandes quantidades de informações entre mainframes, ou entre mainframes e minicomputadores associados aos periféricos compartilhados das instalações, podendo existir quilômetros de separação entre os equipamentos a serem conectados. Outra aplicação típica é a interligação das redes locais de baixa velocidade instaladas nos departamentos da organização, dando origem às chamadas redes dorsais. As redes dorsais precisam ter velocidade suficiente para manipular toda carga agregada das redes locais de baixa velocidade interligadas. Além dessas aplicações, cresce a demanda por uma maior

capacidade para gerenciar e controlar as comunicações corporativas através de um serviço de integração de tráfegos em uma mesma rede (voz, dados e imagem). Todos esses requisitos de serviços podem, em princípio, ser atendidos através de uma rede de comunicação que integre a corporação como um todo, oferecendo taxas de transmissão e banda passante, compatíveis com as suas necessidades.

Para suprir esta demanda crescente dos usuários por um ambiente de comunicação corporativa integrada, várias arquiteturas de redes têm sido propostas [MCQUILLAN,1990], em sua maioria, utilizando como meio de transmissão a fibra óptica [GIOZZA,1991], [GALDINO,1991]. As **Redes Metropolitanas (MAN)** [MOLLENAUER,1988], [NEWMAN,1988], [BRITO,1991] e as **Redes Digitais de Serviços Integrados de Banda Larga (B-ISDN)** [ARMBRUSTER,1987], [MINZER,1989], [BAPTISTELLA,1990], por exemplo, são tecnologias emergentes que se propõem a solucionar os requisitos de banda e de velocidade exigidos pelas aplicações. No entanto, o espectro de utilização destas implementações envolve a rede pública, onde as definições e padronizações são bem mais lentas e complexas devido ao universo diversificado de equipamentos e instituições envolvidos.

Dentre as propostas de tecnologias de redes para integração corporativa pode-se destacar a arquitetura de rede de computadores **FDDI** ("*Fiber Distributed Data Interface*") [MCCOOL,1988], [ROSS,1990], aceita como padrão internacional [ISO/FDDI,...]. As redes FDDI operam em taxa

de transmissão de 100 Mbit/s com fibra óptica [ISO/FDDI,1990a], tem topologia em anel duplo [ISO/FDDI,1989a], e protocolo de controle de acesso ao meio baseado no controle de rotação da ficha com temporizadores [ISO/FDDI,1989b].

O esquema de gerenciamento de acesso ao meio das redes FDDI possibilita basicamente dois tipos de tráfego [ISO/FDDI,1989b], permitindo que serviços com diferentes requisitos de acesso e banda de transmissão, possam ser atendidos de forma eficiente pelo protocolo. O esquema para tráfego síncrono oferece banda garantida e tempo de acesso máximo para transmissão, podendo ser usado em aplicações de tempo real, tipo voz ou controle de processos. O esquema para tráfego assíncrono pode ser utilizado para serviços que compartilhem o canal de comunicação com características de banda e acesso menos restritivas. Visando incorporar características de redes metropolitanas (MAN), foi realizada uma otimização no padrão FDDI para implementação do serviço de comutação de circuitos. Essa otimização, denominada de **FDDI-II** [ROSS,1990], [ANSI/FDDI,1990b,1990e,1990f e 1990g], oferece a mais o tráfego isócrono [BRITO,1991], permitindo ao anel, inclusive, interligar-se com centrais telefônicas implementando o padrão SONET ("*Synchronous Optical Networks*") [ANSI/FDDI,1990d].

A arquitetura FDDI especifica até a primeira sub-camada do nível de **Enlace de Dados** do modelo OSI/ISO (sub-camada de Controle de Acesso ao Meio MAC), aceitando

como sub-camada superior o protocolo LLC ("Logical Link Control") padrão IEEE 802.2/ISO 8802.2 [TANEMBAUM,1981], [GIOZZA,1986]. Cada nodo FDDI é composto das sub-camadas PHY ("PHYSICAL") [ISO/FDDI,1989a], PMD ("Physical Medium Dependent") [ISO/FDDI,1990], MAC ("Media Access Control") [ISO/FDDI,1989b] e SMT ("Station Management") [ANSI/FDDI,1990a]. A topologia para redes FDDI é o anel e dois tipos de conexões estão previstas no padrão [MCCOOL,1988], [ROSS,1990]: classe A, a estação liga-se diretamente ao anel e classe B, a estação liga-se ao anel através de um concentrador. A Figura 1.1 ilustra a topologia padrão FDDI.

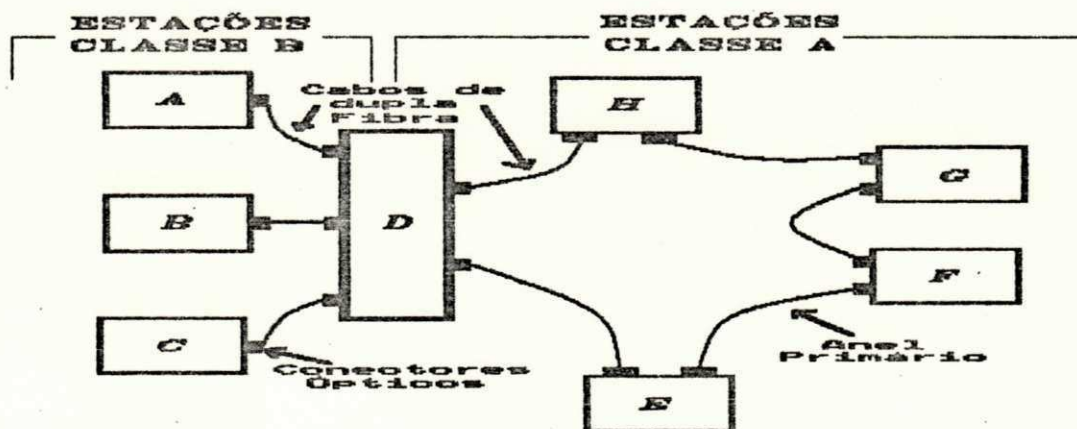


Figura 1.1 - Topologia em anel duplo do padrão FDDI

As redes FDDI estão atualmente em um nível de padronização avançado de modo que várias implementações já estão disponíveis no mercado para instalação [WILSON,1991]. Contudo, é uma tecnologia relativamente cara, cuja aquisição e implantação ainda pode ser bastante onerosa para instituições e organizações de porte médio. Assim, na fase de projeto de uma rede FDDI, é necessário que haja uma

análise criteriosa do contexto em que vai ser instalada a rede, para uma otimização de investimentos. Um estudo prévio de como se comportará uma configuração específica de rede FDDI sob determinadas circunstâncias de tráfego é normalmente necessário. Em redes FDDI, este estudo é bastante complexo tendo em vista o funcionamento do seu protocolo de acesso ao meio, o protocolo de ficha temporizada. Neste tipo de protocolo, o tempo médio de rotação da ficha no anel deve ser controlado, para que as características de tempo de espera máximo para acesso e de banda garantida possam ser asseguradas às estações pertencentes ao anel.

Em uma rede FDDI tem-se basicamente três configurações possíveis: todas estações gerando tráfego síncrono, todas estações gerando tráfego assíncrono, e um contexto misto onde tem-se tráfego síncrono e assíncrono no anel. Uma parametrização mais eficiente das variáveis de um sistema FDDI, necessita de recursos matemáticos que possibilitem analisar o comportamento da rede em operação. É necessário, portanto, modelar o seu comportamento em termos de suas variáveis e prever medidas tais como *vazão*, *tempo de acesso médio* e *atraso de quadros médio*, dentre outras medidas de desempenho de interesse. Dentre as possíveis ferramentas, através das quais pode-se efetuar um estudo prévio do comportamento de uma configuração específica de uma rede FDDI, pode-se destacar o desenvolvimento analítico e a simulação digital.

A ferramenta analítica provê resultados exatos

sobre os problemas que são a ela submetidos. Contudo, o nível de complexidade de desenvolvimento matemático envolvido na modelagem analítica pode ser muito grande, implicando geralmente em várias restrições no sistema analisado. Por outro lado, os resultados obtidos pelo processo de simulação digital, precisam passar por um processo de estimação de intervalo de confiança. Todavia, o método de simulação tem aplicação bem mais simples e o sistema não precisa ser tão restringido, em sua modelagem, para poder ser submetido à ferramenta.

Este trabalho, aborda um estudo, modelagem do protocolo de acesso ao meio FDDI através do pacote de simulação de redes de filas de propósito geral RESQ2 [SAUER,1982a] e obtenção de alguns resultados de desempenho do protocolo FDDI. No próximo Capítulo (Cap. 2) será apresentada a arquitetura de rede padrão FDDI abordando sua topologia e o funcionamento de todas as suas camadas, com um detalhamento específico da sub-camada de controle de acesso ao meio (MAC). No Capítulo 3, serão mostradas a modelagem do sistema FDDI e a construção do simulador de via ferramenta RESQ2. No Capítulo 4, serão apresentados os resultados de desempenho obtidos com a aplicação do simulador em RESQ2, em algumas configurações de redes FDDI, e comparados com resultados existentes sob as mesmas circunstâncias. Finalmente, no Capítulo 5, serão apresentadas as conclusões do trabalho desenvolvido, destacando-se vantagens/desvantagens de uso da ferramenta RESQ2 e o comportamento do protocolo MAC FDDI.

CAPITULO 2

REDES PADRAO FDDI

2.1 INTRODUÇÃO

O padrão FDDI ("*Fiber Distributed Data Interface*") pode ser visto como o resultado da evolução das redes locais de computadores, no sentido de atender às exigências dos seus usuários em termos de requisitos de transmissão (banda disponível e tempo de resposta). Atualmente, em todas as esferas de uma organização informatizada é crescente a pressão no sentido de mover e de processar de forma veloz e eficiente, uma quantidade crescente de informações que circula no ambiente, com a melhor relação custo/benefício, maior segurança e que permita um melhor mauiseio. As redes de alta velocidade [HANSON,1990] visam cobrir especificamente estas necessidades.

Em uma estrutura de processamento de dados baseada em computadores de grande porte, em um mesmo ambiente ou em região geográfica limitada, as redes FDDI podem ser utilizadas para conectar os computadores de grande porte e os demais tipos de processadores da rede com dispositivos periféricos de alta velocidade (por exemplo: discos), formando as redes "BACKEND".

A disseminação da cultura de redes locais em um ambiente com processamento descentralizado, começa a trazer problemas a nível de integração da empresa como um todo. A instalação de redes locais de baixa velocidade permite a formação de pequenos núcleos de processamento independentes. A formação de uma rede dorsal ("BACKBONE") [MANTELMANN,1989], [SHAPIRO,1989], conectando diversas

redes de baixa velocidade (padrão IEEE 802.3, 802.4 ou 802.5) via portas, é mais uma aplicação possível das redes FDDI que pode trazer ganhos significativos em termos de integração de recursos e informações da instituição com um todo. Devido às características de tempo máximo de acesso garantido e de meio de transmissão confiável (fibra óptica), a interligação de diversos equipamentos em um ambiente industrial apresenta-se também como uma aplicação de redes FDDI de grande interesse.

As estações de trabalho de alto desempenho, operando como servidoras de rede, atualmente, oferecem uma série de recursos para as estações clientes que requisitam canais de comunicação bem mais velozes que os atuais e tempos de resposta menores. Esta demanda, em conjunto com as aplicações distribuídas [STROHL,1991] representam uma das tendências mais importantes de uso do FDDI .

O padrão FDDI foi desenvolvido pela ANSI ("*American National Standardization Institute*") nos E.U.A., sendo o comitê responsável pela sua padronização o ASC X3T9.5. Boa parte dos documentos ANSI/FDDI já foram transformados em documentos ISO [ROSS,1990].

A arquitetura de rede FDDI é especificada até a camada de Enlace de Dados segundo o Modelo de Referência para Interconexão de Sistemas Abertos padrão OSI/ISO [ISO,1984]. Um mapeamento da arquitetura FDDI na arquitetura RM-OSI/ISO pode ser visto na Figura 2.1.

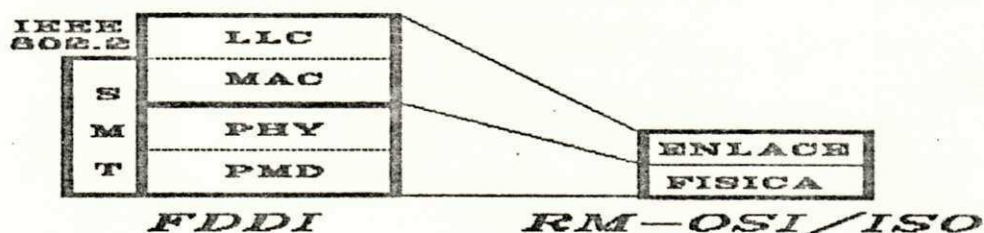


Figura 2.1 - Mapeamento da arquitetura FDDI no modelo OSI/ISO

A arquitetura FDDI tem a seguinte funcionalidade básica em termos de seus protocolos:

- . **PMD** (*Physical layer Medium Dependent*): Controla os níveis de potência óptica, caracteriza o transmissor e o receptor óptico, define os requisitos de sinal para transmissão óptica, padroniza os conectores e define a taxa de erros de bit (BER) [ISO/FDDI,1990a];
- . **PHY** (*PHYSical layer protocol*): Controla a codificação/decodificação, o relógio e a transmissão/recepção dos quadros [ISO/FDDI,1989a];
- . **MAC** (*Medium Access Control*): Controla o acesso ao meio, o endereçamento, a verificação de erros e a formatação dos quadros com as mensagens provenientes de níveis superiores [ISO/FDDI,1989b];
- . **SMT** (*Station Management*): Gerencia localmente o protocolo, efetua o controle da configuração interna da estação na inicialização do anel e da operação da estação [ANSI/FDDI,1990a].

Neste Capítulo, serão abordadas as características de funcionamento e de parametrização de um anel FDDI. Primeiramente, apresenta-se a topologia de funcionamento do anel FDDI, assim como as características

que lhe dão confiabilidade. Prosseguindo, descreve-se o funcionamento do protocolo referente à camada Física e os procedimentos para gerenciamento de uma estação FDDI. Por último, é detalhado o funcionamento da sub-camada MAC do FDDI, cuja modelagem e estudo de desempenho serão apresentados nos capítulos posteriores.

2.2 TOPOLOGIA DAS REDES FDDI

A topologia básica do padrão FDDI [ISO/FDDI,1989a], ilustrada na Figura 2.2, é um anel duplo de árvores com estações caracterizadas pelos seus tipos de conexão e número de sub-camadas MAC's. Uma estação FDDI conectada ao anel duplo, denominada de DAS ("*Dual Attachment Station*"), tem duas conexões, uma para cada anel da configuração dupla e pode ter uma ou duas MAC's ("*single or dual*" MAC). Estações de trabalho de alto desempenho, computadores de grande porte, servidores e roteadores são geralmente estações FDDI do tipo DAS [ROSS,1990], [STROHL,1991].

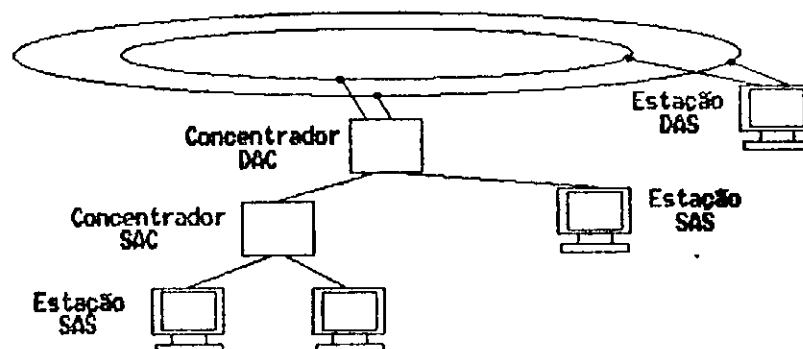


Figura 2.2 - Detalhamento da Topologia FDDI.

Estações que têm apenas uma conexão física (SAS - "*Single Attachment Station*") são conectadas a um dos anéis da configuração dupla do FDDI através de concentradores. Um concentrador pode estar ligado diretamente à configuração em anel duplo (DAC - "*Dual Attachment Concentrator*") ou conectado a outro concentrador (SAC - "*Single Attachment Concentrator*"). [RUIZ,1989], [CODEN,1990].

A configuração FDDI em anel duplo permite que haja uma otimização da banda passante disponível. O segundo anel que fica inativo durante a operação normal, pode ser utilizado pelas estações DAS para efetuarem transmissões no meio. Isso é possível desde que, além da MAC primária, as estações DAS tenham uma MAC secundária ("*Double MAC*") para gerenciar as transmissões no segundo anel, duplicando assim a banda padrão do FDDI para 200 Mbps. A Figura 2.3 ilustra os tipos de configurações internas das estações que o FDDI permite, apresentando também, um exemplo de anel com as estações conectadas.

Os concentradores DAC também podem ter duas MAC's, possibilitando a ativação do anel secundário para fins de transmissão, com a conseqüente ampliação da banda passante disponível. Quando utiliza-se um concentrador DAC com apenas uma MAC instalada não ocorre redundância.

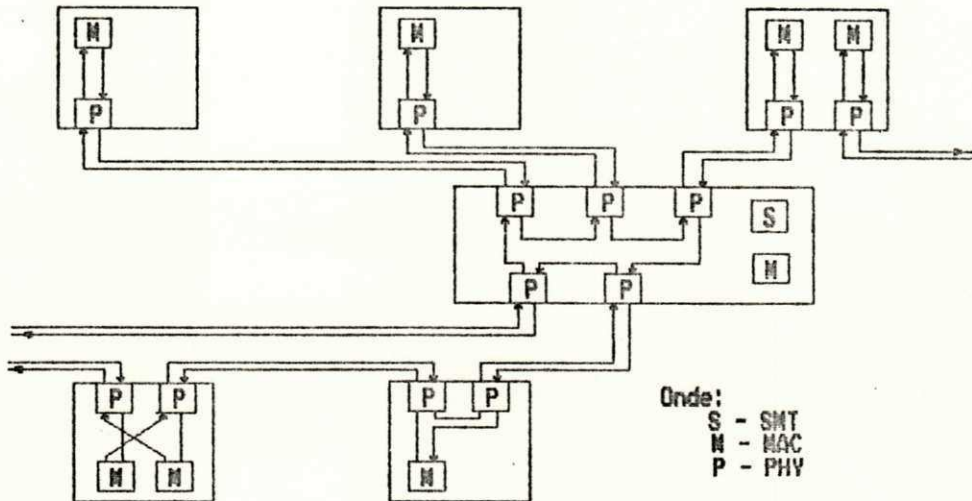
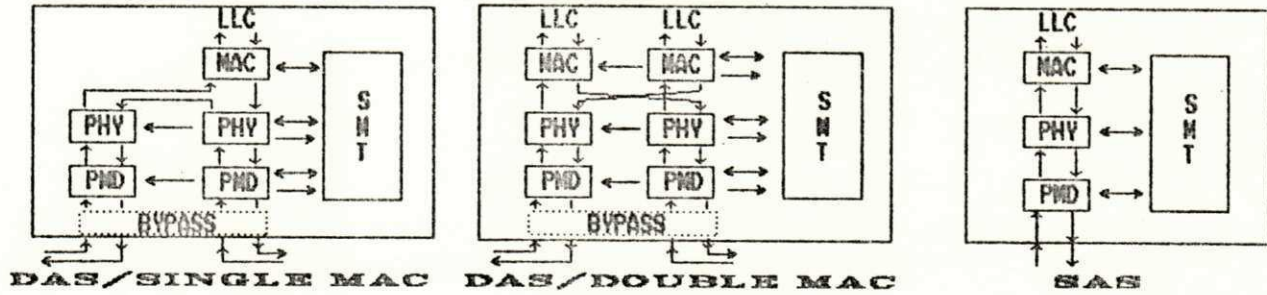


Figura 2.3 - Topologia FDDI com possíveis configurações das estações.

A topologia em anel redundante prevê um grau de tolerância a falhas no meio físico e/ou em estações pertencentes ao anel [JOHNSON,1986a]. Em caso de falha de qualquer um destes elementos, é possível isolar o problema e voltar a fechar o anel através do canal secundário. Em cada estação do tipo DAS, acoplado à sub-camada PHY, existe uma "chave comutadora" que, em caso de falha, possibilita o desvio (BYPASS na Figura 2.3) dos quadros, formando novas configurações de anel. A estrutura em árvore controlada pelos concentradores oferece a facilidade de desligar-se estações a eles conectadas sem prejudicar o funcionamento

da rede. O mecanismo de desvio (BYPASS) pode ser ativado pela própria estação, por estímulo da estação vizinha, por um operador da estação, automaticamente quando se desliga a energia ou por algumas funções de controle geral da rede.

Quando ocorre uma falha, a rede automaticamente se reconfigura, isolando qualquer elemento com problema e mantendo a operação do anel. Um protocolo de gerenciamento da rede [ANSI/FDDI,1990a] continua monitorando o enlace ou a estação com problema, permitindo à rede se reconfigurar automaticamente, e restaurar a sua operação normal, assim que o reparo tenha sido efetuado. Qualquer uma dessas situações pode resultar na perda de quadros individuais que, posteriormente, devem ser retransmitidos.

2.3 SUB-CAMADAS PMD E PHY

As sub-camadas PMD ("*Physical Medium Dependent*") [ISO/FDDI,1990] e PHY (PHYsical) [ISO/FDDI,1989a] provêm, respectivamente os componentes de hardware de comunicação por fibra óptica e os protocolos para suportar uma ligação física entre estações FDDI.

Mais especificamente, a sub-camada PMD define os níveis de potência do sinal óptico para transmissão, as características do transmissor/receptor óptico, os requisitos do sinal para transmissão óptica, as especificações dos conectores ópticos e a taxa de erro de bit BER [BURR,1986], [ISO/FDDI,1990].

A padronização da sub-camada PMD baseia-se em dois objetivos: primeiro, com a rede operando a 100 Mbps,

com 2 Km de fibra óptica entre estações adjacentes, deseje-se um BER de 1 bit a cada $2,5 \times 10^{10}$ bits transmitidos; segundo, busca-se a definição de um hardware de implementação de custo relativo baixo e componentes de fácil comercialização.

O meio de transmissão escolhido inicialmente para implementação do FDDI é a fibra óptica [HARTMANN,1989], [GIOZZA,1991], do tipo multimodo com índice gradual, diâmetros 62,5/125 microns, operando com uma banda de 400 MHz/km e atenuação de 2,5 dB/km na janela de 1300 nm. Exemplos de projetos de instalação de redes FDDI podem ser encontrados em [STEVENS,1990].

Os dispositivos ópticos ativos [GIOZZA,1991] recomendados para transmissão em redes FDDI são os LED's ("*Light Emitter Diodes*") que, apesar de menos potentes que os lasers, são mais baratos, mais seguros e danificam menos. Os dispositivos fotodetectores [GIOZZA,1991] sugeridos preferencialmente pelo padrão são os fotodiodos PIN ("*Positive - Intrinsic - Negative*").

Existem já alguns protótipos de conectores ópticos FDDI [OHLHABER,1989], [GIOZZA,1991], contudo, o padrão define apenas que as perdas nas conexões devem estar entre 0,2 e 1,0 dB. Com relação às chaves comutadoras (BYPASS) do sistema de tolerância a falhas do anel, não é especificada uma perda padrão. Todavia, define-se que a perda máxima, entre duas estações adjacentes deve ser de 11 dB e que a perda máxima entre a entrada e a saída de uma estação desviada, deve ser 3 dB.

Dentro da funcionalidade específica da sub-camada PHY, ela é responsável pela codificação/decodificação das informações procedentes/destinadas da/para sub-camada MAC e pelo controle do relógio e dos procedimentos de transmissão/recepção. As operações de transmissão e de recepção funcionam simultaneamente.

O FDDI usa o esquema de codificação de grupo denominado de 4B/5B. A técnica de codificação em grupo mapeia cada quatro bits de dados ou símbolos de controle provenientes da sub-camada MAC, em um código de cinco bits, denominado de código de grupo [ROSS,1989]. Cada sequência de quatro bits proveniente da sub-camada MAC é, então, codificada em um grupo cinco de bits NRZ que são, em seguida, codificados em cinco bits NRZI para transmissão no meio. O resultado deste método de codificação é um fluxo serial de bits, que contém no mínimo duas transições para cada símbolo transmitido, ou seja, cada símbolo carrega sua informação de relógio e tem, no máximo, três zeros consecutivos. Com o esquema de codificação 4B/5B, a taxa de 100 Mbps de dados é transformada em 125 Mbps efetivamente inseridos na rede. Usando uma codificação Manchester diferencial, por exemplo, seria necessária uma taxa efetiva de 200 Mbps para inserir na rede a mesma quantidade de informação.

O padrão FDDI, a nível físico, especifica a técnica de sincronização ao nível de bit, em modo síncrono não coerente [GIOZZA,1986]. O receptor emprega relógio de frequência variável, geralmente implementado por dispositivo

PLL, e o transmissor utiliza um relógio local com frequência fixa, enquanto que um tampão ("buffer") elástico em cada estação compensa eventuais diferenças entre os dois relógios. Define-se no padrão FDDI uma estabilidade percentual para o relógio de transmissão da ordem de 0,005%. O tampão ("buffer") mínimo que cada estação deve ter é de 10 bits. Cada QUADRO transmitido no FDDI deve ter um preâmbulo (PA) de 16 octetos, e define-se um tamanho máximo de QUADRO igual a 4500 octetos.

2.4 FORMATOS DO QUADRO E DA FICHA MAC FDDI

O padrão FDDI define para a sub-camada MAC dois formatos de UDP's (Unidades de Dados de Protocolo) para a operação do protocolo: o QUADRO e a FICHA. Os QUADROS são unidades que permitem a transferência de informações pelo protocolo, podendo ter seu tamanho variável. A FICHA é representada por um "QUADRO" especial de tamanho fixo, usado para dar o direito a uma estação de transmitir QUADROS no anel.

As informações de controle do protocolo FDDI são denominadas de símbolos [ROSS,1989]. O formato do QUADRO e da FICHA MAC FDDI, mostrados na Figura 2.4, consiste dos seguintes campos:

- . PA ("*Preamble*"): campo que contém 16 símbolos específicos para o estabelecimento e manutenção da sincronização do relógio das estações;
- . SD ("*Starting Delimiter*"): contém símbolos específicos que delimitam o início de um QUADRO;

- . FC ("*Frame Control*") : contém símbolos que identificam o tipo de QUADRO e suas características; os símbolos distinguem QUADROS síncronos de QUADROS assíncronos, o tamanho do campo de endereço (16 ou 48 bits) e o tipo de QUADRO (se dados do nível superior LLC ou dados de gerenciamento da SMT);
- . DA ("*Destination Address*") e SA ("*Source Address*") : ambos são compostos por 16 ou 48 bits, dependendo do valor do FC, e contêm, respectivamente, o endereço destino e o endereço fonte do QUADRO; o DA pode ser tanto um endereço individual como de grupo, sendo que o endereço de grupo pode ser reconhecido por mais de uma estação;
- . FCS ("*Frame Check Sequence*") : campo de 32 bits que contém um código de redundância cíclica usando o polinômio padrão dos protocolos IEEE 802 [GIOZZA,1986];
- . ED ("*Ending Delimiter*") : delimitador de fim de QUADRO;
- . FS ("*Frame Status*") : tem no mínimo três símbolos indicadores de controle que são modificados pela estação quando ela repete o QUADRO; os símbolos indicam o estado da recepção do QUADRO no destino.

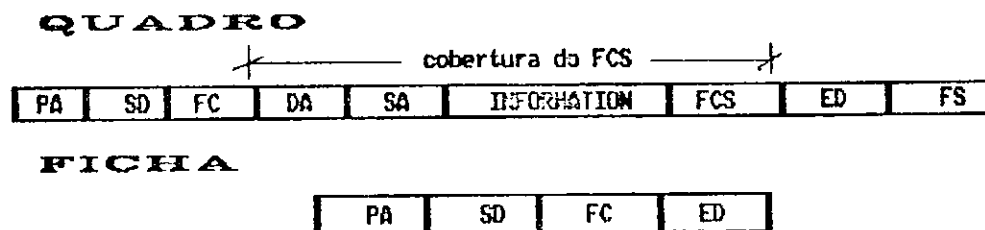


Figura 2.4 - Formatos do QUADRO e da FICHA MAC FDDI.

2.5 SUB-CAMADA MAC

A sub-camada MAC (*"Medium Access Control"*) do FDDI [ISO/FDDI,1989b] responsabiliza-se pelas tarefas de controle de acesso ao meio, endereçamento, verificação de erros e formatação de quadros. Esta sub-camada da arquitetura FDDI será apresentada mais detalhadamente pois é o objeto de análise principal deste trabalho.

Na operação do protocolo de passagem de FICHA do anel FDDI, a estação só pode transmitir QUADROS com a posse da FICHA. Independente da quantidade de QUADROS que a estação poderá transmitir, após a captura da FICHA, ela é retirada do anel. De posse da FICHA, a estação origem transmite o(s) seu(s) QUADRO(S) e cada estação intermediária repete o QUADRO até a estação destino. Uma característica importante do protocolo MAC FDDI é a reposição da FICHA ao anel logo após a transmissão dos QUADROS [ULM,1982]. Isso diferencia o mecanismo de controle de acesso do FDDI do utilizado nas redes locais padrão IEEE 802.5 [GIOZZA,1986].

A estação destino reconhece o QUADRO que lhe é endereçado através do DA, copia o quadro para o armazenador (*"buffer"*) de recepção local e a sua MAC notifica o LLC (via SMT) que chegou um QUADRO. Ao receber um QUADRO, a MAC destino marca o campo FS do QUADRO com uma das seguintes informações: reconhecimento do seu endereço, cópia do QUADRO e detecção de erro no QUADRO. Após esta operação, o QUADRO é recolocado no anel para que retorne à estação fonte.

Retornando à estação origem, o QUADRO é reconhecido pelo seu SA. A MAC origem examina o campo FS e observa o sucesso/insucesso da transmissão. Caso os dados tenham sido recebidos na MAC destino de forma correta, o QUADRO é retirado do anel, sendo esta retirada denominada pelo FDDI de "*striping*".

Em redes FDDI, o fato de uma estação possuir a FICHA, não significa que ela terá banda ilimitada para transmissão. O padrão FDDI especifica um protocolo de controle temporizado da rotação da FICHA (*trt* - "*token rotation timer*") que controla a velocidade com que a FICHA circula no anel. Este protocolo provê as características de tempo máximo de acesso para o tráfego síncrono e de banda passante necessária para a transmissão destes QUADROS de natureza crítica.

No processo de inicialização do anel [MCCOOL,1988] e [ISO/FDDI,1989b], cada estação FDDI sugere um tempo de rotação da FICHA no anel igual à metade de seu tempo de acesso síncrono desejado. Nesta fase, as estações entram em regime de competição e o menor tempo sugerido será atribuído à variável *ttrt* ("*target token rotation time*") comum a todas as estações. O *ttrt* equivale ao tempo médio de rotação da FICHA no anel, enquanto que o tempo máximo de rotação da FICHA no anel será $2 \times ttrt$. Desta forma, fica garantido que o tempo de acesso síncrono das estações ao anel estará sempre dentro da frequência desejada.

O controle do acesso de cada estação ao anel, é feito usando-se o `ttrt` em conjunto com mais dois temporizadores, chamados `trt` ("token rotation timer") e o `tht` ("token holding timer"), e uma variável de estado denominada de `late_ct`. Após a definição do `ttrt` no processo de inicialização, cada estação recebe um percentual deste `ttrt` como banda garantida para suas transmissões síncronas, dividindo assim a banda total de transmissão. Quando a FICHA chegar adiantada (com relação ao acesso síncrono) numa estação, o tempo remanescente poderá ser utilizado para transmissões assíncronas.

Para fins de descrição do funcionamento do protocolo MAC FDDI, assume-se que todos os temporizadores `trt` e variáveis `late_ct` são inicializados com zero e expiram quando chegam ao valor do `ttrt`. O `trt` é usado em cada estação para medir o tempo entre duas chegadas sucessivas da FICHA e, é zerado, cada vez que a FICHA é recebida dentro do `ttrt`. O `trt` irá expirar se ele for incrementado até `ttrt` antes que a FICHA tenha chegado à estação. Quando o `trt` expira, o `late_ct` é incrementado de um e o `trt` é recarregado com zero, continuando a temporização. Quando a FICHA chega tarde em uma estação (`late_ct=1`), o `trt` não é zerado, continuando a temporização e acumulando, então, o atraso de rotação atual para a temporização da próxima rotação de FICHA. O resultado do acúmulo do atraso, na rotação da FICHA, é o seguinte: como a rotação da FICHA excede `ttrt` por um tempo `A`, as transmissões assíncronas serão restringidas até que este atraso seja compensado por

rotações de FICHA menores que t_{trt} de um tempo A. Quando o anel recupera o atraso, o(s) $late_ct(s)$ são novamente zerados. Todo este procedimento assegura que a média de rotação da FICHA é t_{trt} . Se o $late_ct$ exceder 1, então, um procedimento de recuperação de erro é acionado.

O temporizador t_{ht} é usado por cada estação para controlar a quantidade de tempo que a FICHA fica presa para transmissões assíncronas. O t_{ht} é carregado com o valor do $t_{trt} - trt_{corrente}$ quando a FICHA é recebida dentro do tempo normal ($late_ct=0$). Quando o t_{ht} alcança o limite de tempo de transmissão (FICHA presa) para um nível de prioridade particular, a FICHA não pode ser mais usada para transmitir QUADROS daquele nível. Transmissões ainda em progresso quando o t_{ht} expira são completadas. O tempo residual que se segue à expiração do t_{ht} é denominado "asynchronous overrun". O valor máximo do "asynchronous overrun" é igual ao tempo máximo de transmissão de um QUADRO. O $t_{pri}(i)$ ($i=1$ a 8) define o t_{ht} limiar para o nível de prioridade assíncrona i . A convenção adotada usualmente é que a prioridade aumenta de 1 a 8. Um valor maior de limiar significa mais tempo para transmissão de QUADROS daquele nível de prioridade antes da FICHA ser passada adiante. Portanto, o nível de prioridade que tem associado uma janela de transmissão maior, tem, conseqüentemente, maior prioridade do que os níveis de prioridade com limiares de FICHA presa menores.

2.6 OTIMIZAÇÃO FDDI-II

Dentro do cenário de aplicações em Redes Metropolitanas [BRITO,1991] foi desenvolvida uma otimização do FDDI básico, denominada FDDI-II, com o objetivo de permitir numa rede FDDI, aplicações de voz digital e vídeo comprimido conjuntamente com o tráfego de dados por pacotes. Algumas modificações na estrutura interna da estação FDDI original foram feitas, conforme ilustrado na Figura 2.5.

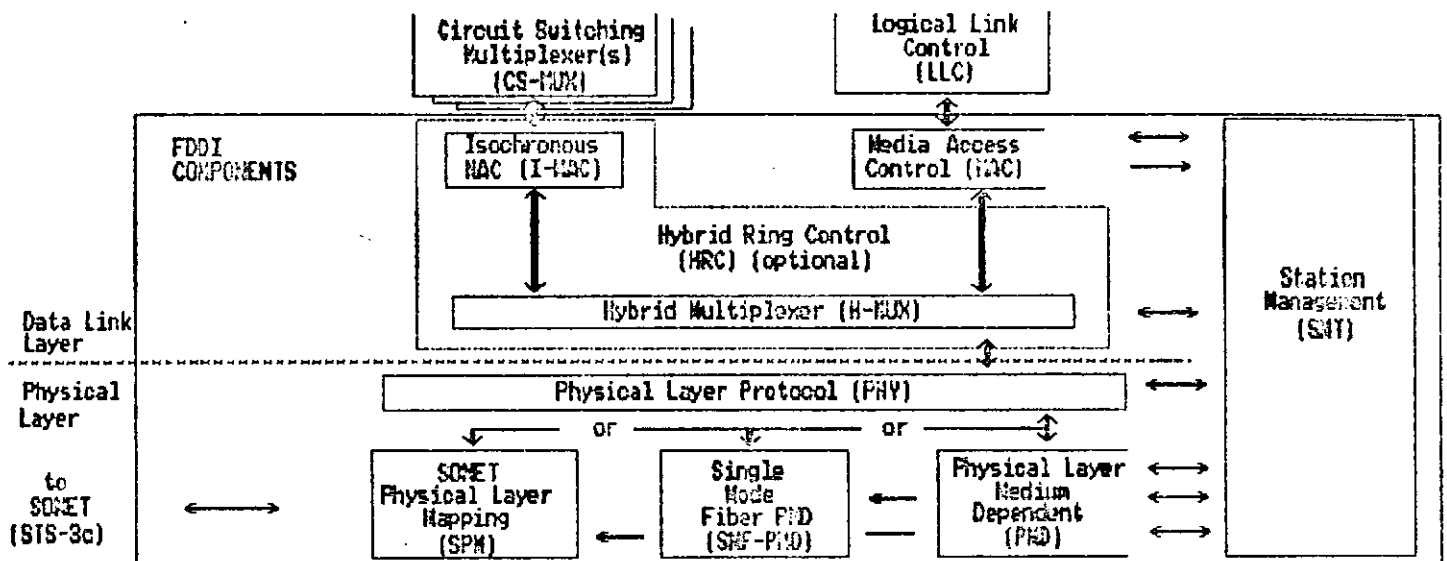


Figura 2.5 - Configuração interna de uma estação FDDI-II.

Basicamente, as sub-camadas PMD e PHY do FDDI-II permanecem iguais às do FDDI básico. A sub-camada MAC, formada pela otimização I-MAC [ANSI/FDDI,1990g] e pela HRC ("Hybrid Ring Control") [ANSI/FDDI,1990e], incumbe-se de, juntamente com o SMT, de gerenciar o tráfego de voz que exige a técnica de comutação de circuitos. No FDDI-II este tipo de tráfego é chamado de isócrono. A sub-camada SMF-PMD [ANSI/FDDI,1990f], uma versão do PMD para fibras monomodo,

permite que possam ser instalados enlaces ópticos com até 60 Km de distância entre estações adjacentes. A sub-camada SPM ("*SONET Physical Mapping*") [ANSI/FDDI,1990d] provê uma alternativa ao PMD básico para as redes usufruírem das facilidades de transmissão do padrão SONET [BALLART,1989].

Para o tráfego de voz podem ser alocados no FDDI-II, da banda passante de 100 Mbps, canais de até 6.144 Mbps cada. No sistema de telecomunicações americano ou japonês, esse canal elementar representam 4 subcanais de 1.536 Mbps e, no sistema europeu, também adotado no Brasil, 3 subcanais de 2.048 Mbps. Cada canal de 6.144 Mbps no FDDI-II é chamado de WBC ("*WideBand Channel*"). Pode-se ter até 16 WBC's em um anel FDDI-II.

No FDDI-II, o protocolo de FICHA temporizada convive naturalmente com os canais alocados, sendo que a banda que não estiver alocada para WBC's é utilizada pelo protocolo trt. O anel inicia sempre a operação no modo FDDI básico (ou seja só tráfego de comutação de pacotes), passando em seguida a operar no modo híbrido: comutação de pacotes + comutação de circuitos.

O protocolo FDDI-II opera em seus WBC's com uma frequência de um octeto a cada 125 μ s. Na fase de inicialização, é definida uma estação que irá marcar o ciclo mestre de 8 kHz (125 μ s). Com o FDDI-II, existe a possibilidade de interligação com a rede pública, sendo necessária apenas uma conexão com uma central telefônica pública de tal forma que sejam sincronizados os relógios.

CAPITULO 3

**MODELAGEM DO PROTOCOLO
MAC FDDI**

3.1 INTRODUÇÃO

A solução do modelo de um sistema permite analisar o seu comportamento dentro de determinadas situações específicas desejadas. Isso facilita sobremaneira a tarefa do projetista de sistemas que terá uma idéia do seu funcionamento antes mesmo de colocá-lo em prática, possibilitando ajustes que, além da otimização de recursos, diminuirão os riscos de problemas durante a sua operacionalização.

Na modelagem de um sistema, costuma-se extrair suas características mais importantes e com isto, normalmente, no próprio processo de modelagem, algumas restrições de funcionamento do sistema já são necessárias para sua representação. Em um protocolo de controle de acesso ao meio de redes locais de computadores, baseado em processos movidos a eventos discretos, pode-se utilizar a modelagem através de redes de filas.

Um sistema, ao ser modelado por redes de filas, permite que as suas entidades que provêm serviço sejam mapeadas em um conjunto estruturado de redes de filas, cada qual contendo os respectivos servidores. O modelo resultante é composto pelos clientes, entidades que percorrerão o modelo requisitando serviços de elementos do sistema; elementos, entidades que representam os recursos existentes no sistema e rotas que representam os possíveis encaminhamentos que os clientes podem tomar. As entidades que compõem um sistema modelado, normalmente, possuem atributos necessários à representação do seu funcionamento.

Estes atributos, de forma geral, são utilizados em decisões de roteamento e na prestação do serviço aos clientes.

Na modelagem de um sistema por através de rede de filas, o sistema funciona, basicamente, com clientes que desejam ser servidos pelos recursos da rede de filas, onde os elementos representam todo o processamento que o sistema deve efetuar. Particularmente neste trabalho, os clientes serão os QUADROS ("FRAMES") que circulam numa rede de computadores FDDI. Os recursos que os quadros requisitarem serão as atividades de processamento da estação e de transmissão no meio. As rotas serão os caminhos que os QUADROS terão que seguir, partindo da geração na estação origem até alcançar o seu destino. A Figura 3.1 ilustra o funcionamento descrito.

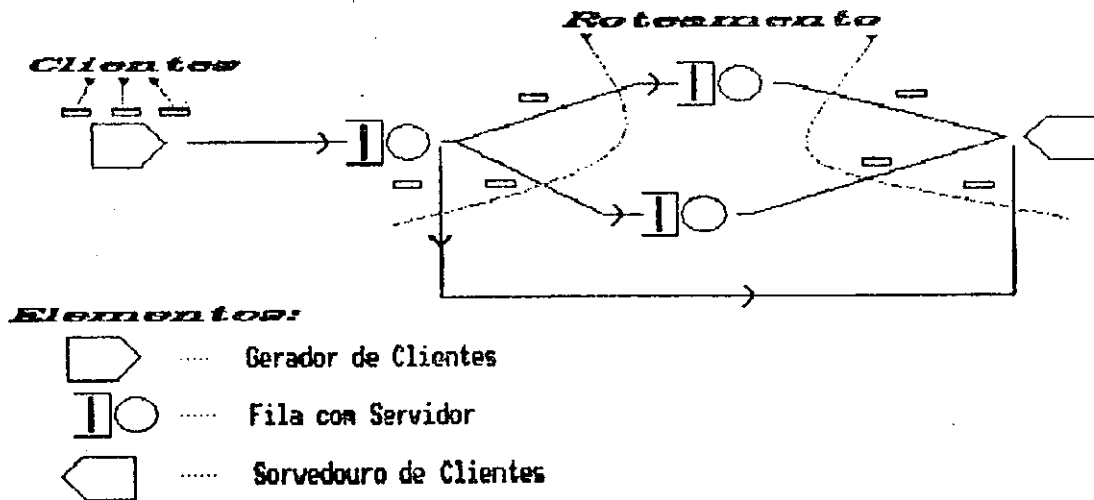


Figura 3.1 - Funcionamento de um modelo geral de rede de filas.

Pode-se modelar um sistema por redes de filas através da estrutura elementar ou através da estrutura com recursos estendidos [SAUER,1985].

A modelagem baseada na estrutura elementar consiste de clientes que visitam filas e requisitam serviços dos servidores daquelas filas. Neste caso, a rede deve ter uma fonte geradora de clientes externa e um sorvedouro para a saída dos clientes já servidos.

No caso da modelagem baseada em redes de filas estendidas é possível uma representação bem mais exata da realidade de funcionamento de um sistema. Além dos recursos elementares, facilidades tais como requisição de serviços simultâneos pelo mesmo cliente, sincronização e paralelismo (mensagens de controle), bem como efeitos de protocolos estão disponíveis neste tipo de representação.

Os recursos de redes de filas estendidas precisam ser oferecidos pela ferramenta que será utilizada para analisar o modelo. Assim, o tipo de solução adotada para a análise (analítica ou numérica) refletir-se-á diretamente sobre os recursos necessários para o analista representar o seu sistema.

A solução de modelos de redes de filas, através da ferramenta analítica [KLEINROCK,1975], é efetuada via equações matemáticas que relacionam os parâmetros do modelo com as medidas de desempenho desejadas, oferecendo, como resultado, valores exatos das medidas de avaliação de desempenho. Devido ao nível de complexidade relacionado a este tipo de solução, faz-se necessário, geralmente,

restrições na modelagem do sistema. Isso pode influir negativamente na obtenção de resultados de desempenho precisos do sistema em estudo.

A solução de modelos por métodos numéricos processa-se através de interações, métodos de convergência, interpolação, obtendo-se valores estimados, com um erro de aproximação, das medidas obtidas. O método numérico não é tão preciso quanto o método analítico, porém, normalmente permite uma modelagem mais real do sistema em estudo. Um método numérico para solução de modelos de sistemas, a técnica de simulação digital [SAUER,1981,1983,1985], utiliza recursos computacionais para efetuar as interações. A solução de modelos via simulação digital pode exigir o desenvolvimento de um programa específico (simulador) para refletir as características do modelo em estudo ou utilizar uma ferramenta de simulação de propósito geral. No primeiro caso, pode-se utilizar os recursos de filas estendidas para uma modelagem mais eficiente do sistema, isto dependerá apenas da disposição de investimento em termos de lógica de programação para alcançar o objetivo. Por outro lado, no uso de uma ferramenta de simulação de propósito geral, o usuário da ferramenta está preso aos recursos que o utilitário oferece.

O uso de ferramentas de simulação de redes de filas de propósito geral permite uma economia de tempo muito grande no conhecimento da ferramenta que será utilizada para efetuar o estudo do sistema. Neste sentido, a IBM desenvolveu a ferramenta RESQ2 [SAUER,1982a-1982b]

que permite a modelagem e a simulação discreta orientada a processo de redes de filas com características estendidas. Este trabalho, visando uma otimização do tempo de estudo do modelo de uma rede de computadores FDDI, baseia-se no uso da ferramenta RESQ2.

O processo de simulação de sistemas, inerentemente, envolve variação estatística devido ao uso de números aleatórios. Assim, na simulação de modelos, precisa-se homologar os resultados obtidos, através de uma análise estatística dos resultados de simulação e a da determinação do período de simulação. O método padrão para estimar a precisão dos resultados de uma simulação é a complementação do resultado de uma determinada medida de desempenho p desejada (por exemplo: vazão), com a obtenção do intervalo de confiança, $(p - y, p + y)$, dentro do qual o ponto p obtido estará com uma probabilidade de $x\%$ (por exemplo: 90%) [SAUER,1983].

Neste Capítulo, inicialmente será apresentada uma visão da ferramenta de simulação RESQ2 mostrando os seus recursos para modelagem e simulação. Em particular, serão abordadas as facilidades oferecidas em termos de obtenção de resultados e dos métodos de análise de intervalos de confiança já embutidos na própria ferramenta. Na sequência, será apresentada a modelagem detalhada do protocolo de controle de acesso ao meio do padrão FDDI através da ferramenta RESQ2.

3.2 A FERRAMENTA RESQ2

A ferramenta de simulação de redes de filas de propósito geral RESQ2, desenvolvida pela IBM, tem como objetivo permitir a solução de modelos de redes de filas através do processo de simulação digital, ou através do método de análise matemática, utilizando a Teoria das probabilidades e outras ferramentas de matemática aplicada.

A utilização da ferramenta RESQ2 para a solução de modelos de redes de filas via simulação passa pelos seguintes procedimentos:

- . Representação gráfica do modelo via simbologia própria e parametrização inicial;
- . Mapeamento desta linguagem gráfica para uma linguagem de programação própria de alto nível;
- . Compilação do programa;
- . Execução do programa (simulação do modelo) para obtenção das medidas de desempenho desejadas;
- . Otimização do modelo via parametrização mais eficiente, incluindo até modificações na própria representação gráfica inicial.

Os elementos que compõem o processamento de clientes numa rede de filas têm basicamente as seguintes funções:

- . Geração de clientes;
- . Parametrização dos clientes;

- . Destruição de clientes;
- . Encaminhamento de clientes;
- . Enfileiramento e serviço de clientes;
- . Processamento de clientes na rede do tipo armazenamento ("bufferização") e controle;
- . Duplicação de clientes na rede.

A parametrização do modelo de redes de filas com a ferramenta RESQ2 passa pela atribuição dos parâmetros globais referentes aos clientes, dos elementos da rede e das cadeias que formam a rede:

- . a parametrização genérica consta de variáveis que são utilizadas para a execução do processamento como um todo ou de valores que são utilizados na parametrização dos elementos da rede; por exemplo, pode-se definir como variável genérica, um valor médio de serviço de um elemento para que seja melhor visualizada e facilite a sua modificação;
- . a parametrização dos clientes define atributos que permitem a prestação de serviço e decisões de roteamento; valores tais como o tamanho do QUADRO, estação de origem e a estação de destino, são exemplos de atributos de clientes;
- . a parametrização das cadeias inclui mecanismos que permitem o controle do fluxo de geração de quadros por uma determinada cadeia, se desejado.

As Funções de Distribuição de Probabilidade (FDPs) são fundamentais para o processo de simulação de modelos. A ferramenta RESQ2 permite a utilização de várias Funções de Distribuição de Probabilidade em seus processamentos, dentre as quais: *Branching Erlang*, *Uniform*, *Standard* e *Discrete*. É possível, também, associar estas funções formando outras FDP's.

O encadeamento de uma estrutura de rede de filas permite o roteamento dos clientes dentro da rede. Para facilitar a representação de funcionamento de um sistema em seu modelo, a ferramenta RESQ2 permite que decisões de roteamento dentro do modelo possam ser tomadas a partir de parâmetros previamente definidos. Assim, por exemplo, pode-se encaminhar um QUADRO para uma estação, ou outra, a depender de um parâmetro especificado dentro deste QUADRO.

Um sistema normalmente tem partes que podem ser generalizadas em sua modelagem, dependendo apenas de recursos disponíveis na ferramenta utilizada. Na ferramenta RESQ2, a representação de um sistema pode ser subdividida em submodelos que terão parametrizações externas genéricas, tornando o programa mais claro e conciso. Pode-se comparar os submodelos, às subrotinas de representação do sistema que são encadeadas, conforme a necessidade de estruturação do modelo do sistema.

A ferramenta RESQ2 oferece já embutidos três tipos de análise de intervalo de confiança, permitindo inclusive a escolha do método que melhor se adequa àquele

modelo específico de sistema. Os métodos disponíveis na ferramenta RESQ2 são: replicações independentes, regenerativo e espectral.

Para controlar a execução da simulação, a ferramenta RESQ2 oferece dois recursos, que são a definição de estados iniciais do sistema e de limites de execução do sistema:

Após a modelagem do sistema e sua parametrização, a definição do método de análise do intervalo de confiança a ser utilizado e a especificação de controle de execução da simulação, todas as etapas necessitam então ser traduzidas em um programa, em linguagem estruturada e de alto nível, que será compilado pela ferramenta RESQ2 e depois, então, executado para obtenção dos resultados de desempenho desejados.

A estrutura do programa em RESQ2 começa pela definição do modelo principal e dos submodelos que o compõem, passa pelo encadeamento geral e pela definição do método de intervalo de confiança que será utilizado e termina com a definição dos estados iniciais e dos limites de execução da simulação.

Como o estudo de um sistema via simulação depende em grande parte da sua representação gráfica, a seguir será apresentado o conjunto de símbolos, elementares e estendidos, utilizados para representação do sistema via modelo de redes de filas.

3.2.1 Representação Gráfica do Modelo

Na descrição do modelo de um sistema, a ferramenta RESQ2 oferece três grupos de símbolos de representação, que são: **ACTIVE QUEUES**, **PASSIVE QUEUES** e os **NODES**. Para facilitar o entendimento do modelo e do programa, será usada apenas a designação do símbolo RESQ2, em inglês, nesta apresentação. É interessante observar que nestas estruturas de representação já se encontram os recursos de filas estendidas, mencionados anteriormente.

3.2.1.1 Active Queues

As **ACTIVE QUEUES** são as principais estruturas de representação de modelos dentro da ferramenta RESQ2; é a estrutura onde os clientes efetivamente são servidos. A estrutura de representação das **ACTIVE QUEUES** é composta por filas e servidores, sendo que uma fila pode ter um ou mais servidores, de tal forma que um servidor não pertença a mais de uma fila. A fila da **ACTIVE QUEUE** pode ser subdividida em classes. As classes permitem que se tenha em uma mesma **ACTIVE QUEUE** distribuições e prioridades de serviço distintas, além de permitir rotas alternativas.

O funcionamento de uma **ACTIVE QUEUE** baseia-se na chegada de um cliente em sua fila (independente da classe em que ele entra), na obtenção de recursos do servidor baseado em determinada distribuição de serviço vinculada àquela classe, e no prosseguimento do processamento normal do modelo. Dependendo da quantidade de clientes que chega à **ACTIVE QUEUE** e do processo de serviço estipulado, podem ser formadas as filas de clientes.

As ACTIVE QUEUES permitem várias topologias de representação. A Figura 3.2 apresenta essas topologias já associadas à sua tradução para o código de programa RESQ2 de forma simplificada.

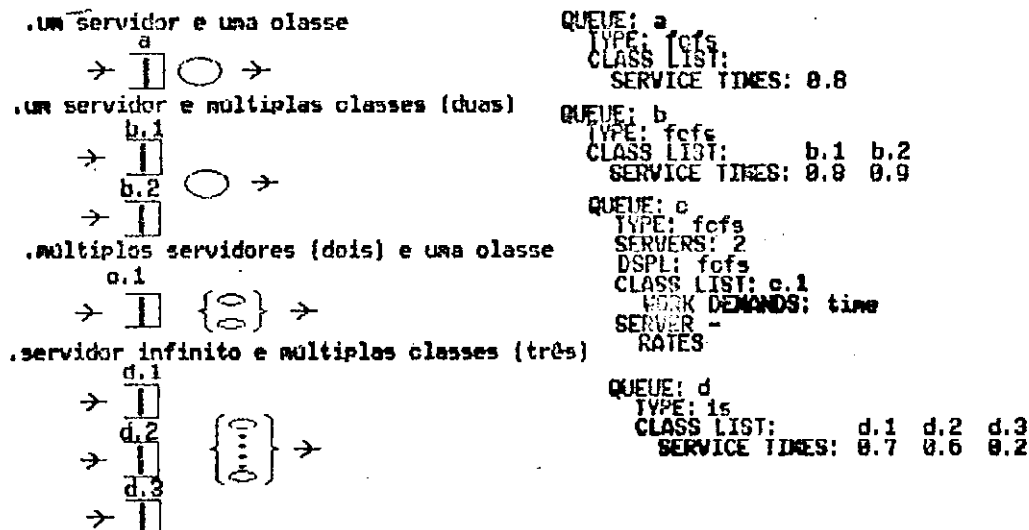


Figura 3.2 - Topologias de ACTIVE QUEUES e tradução RESQ2.

Nas estruturas apresentadas na Figura 3.2 tem-se o nome do servidor representado por uma letra e suas classes vinculadas a essas letras. O método de escalonamento de serviço de clientes, definido no campo TYPE ou no DSPL, exemplificado Figura na 3.2 é o FCFS ("First Come First Served"). No servidor D, um método de escalonamento especial IS ("Infinity Server") é definido, já que não são formadas filas. Neste método assim que o cliente chega, ele é servido. A ferramenta RESQ2 permite outros métodos de escalonamento: LCFS ("Last Come First Served"), RR ("Round Robin"), PS("Processor Sharing"), PRTY

("Nonpreemptive Priority") e PRTYPR ("Preemptive-Resume Priority").

A obtenção de recursos por um cliente pode ser feita de duas maneiras. Uma, especificando o SERVICE TIMES que define o tempo que o cliente gastará no servidor para obtenção de serviço e, a outra, através do WORK DEMANDS onde se especifica a demanda de recursos desejada, em conjunto com o campo RATE OF SERVICE que define a quantidade de trabalho que um servidor pode realizar por unidade de tempo. A função de distribuição de probabilidade padrão associada aos valores destes campos é a exponencial, podendo contudo ser definida qualquer outra FDP.

3.2.1.2 Passive Queues

As estruturas de representação do tipo PASSIVE QUEUE permitem a simulação do processo de tamponamento ("buffering"), recurso importantíssimo na representação de modelos com filas estendidas. Por exemplo, uma PASSIVE QUEUE pode representar uma determinada quantidade de "memória" disponível para uso, controlando a entrada de processos para serviço em uma CPU.

O funcionamento da PASSIVE QUEUE baseia-se na disponibilização de TOKEN's (fichas), que neste texto, daqui por diante, serão referenciadas por TOK's para não confundir com a ficha ("token") do protocolo de controle de acesso ao meio FDDI. Cada TOK representa uma unidade de recurso disponível a ser obtida por um cliente para prosseguir adiante no modelo. Voltando ao exemplo da

memória, seja uma PASSIVE QUEUE que possua 10 TOK's disponíveis; assim, tem-se 10 unidades de memória para os processos poderem entrar em serviço na CPU. Durante o processamento, um determinado cliente entra em fila para obtenção de 5 TOK's. Como a PASSIVE QUEUE tem 10, ela aloca 5 TOK's, o cliente passa adiante para o serviço, e a PASSIVE QUEUE permanece com 5 TOK's disponíveis. Caso, em seguida, chegue um novo cliente requisitando 7 TOK's, ele terá que parar, esperar que o cliente₁, que passou e levou 5 TOK's, termine o seu processamento liberando as TOK's. Com a liberação de TOK's pelo cliente₁, é completado o número de TOK's necessário na PASSIVE QUEUE para que o cliente₂ passe adiante. Exemplos de possíveis topologias que as PASSIVE QUEUES podem assumir estão apresentadas na Figura 3.3.

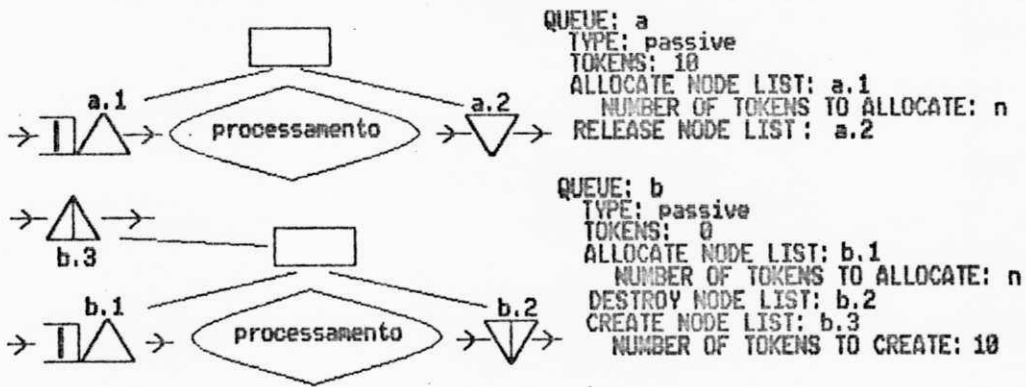


Figura 3.3 - Topologias das PASSIVES QUEUES e tradução RESQ2.

Na primeira opção de uso da PASSIVE QUEUE, ilustrada na Figura 3.3, os clientes param no nodo ALLOCATE NODE; caso existam TOK's suficientes armazenadas na PASSIVE

QUEUE, ele pega-as e segue adiante. Caso contrário, espera que TOK's sejam liberadas por outros clientes no RELEASE NODE, para então se movimentar. Nesta opção, o número de TOK's dentro do modelo é constante. Na segunda opção da Figura 3.3, tem-se a criação e destruição de TOK's devido a contingências do sistema em estudo. Nesta opção de topologia, as TOK's são alocadas no ALLOCATE NODE como no caso da primeira opção; porém, o número de TOK's disponíveis para obtenção dependerá de TOK's criadas no CREATE NODE e destruídas no DESTROY NODE. Assim, o número de TOK's no modelo, neste caso, é variável.

3.2.1.3 NODES

Os NODES são elementos da modelagem via RESQ2 que permitem operações com clientes do tipo criação, destruição, modificação de parâmetros, geração de "filhos" de clientes e roteamento. A Figura 3.4 apresenta os símbolos dos NODES que a ferramenta RESQ2 oferece para modelagem de sistemas.

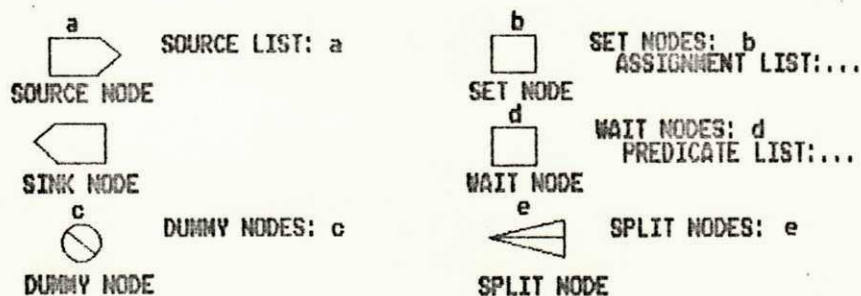


Figura 3.4 - Símbolos dos NODES e tradução RESQ2.

O SOURCE NODE gera novos clientes para processamento pelo modelo. O SINK NODE trabalha como sorvedouro de

clientes, destruindo-os após o processamento. O SINK NODE não é nomeado formalmente como os demais NODES, sendo definido, quando se deseja utilizá-lo, apenas como SINK e inserido no roteamento. O SET NODE efetua mudança de variáveis durante a simulação e mudança dos parâmetros dos clientes. No campo ASSIGNMENT LIST de um SET NODE são listadas todas as variáveis que se deseja mudar. O WAIT NODE segura o cliente até que uma determinada situação, definida no campo PREDICATE LIST, aconteça, liberando-o para seguir adiante no modelo. O SPLIT NODE permite a criação de um novo cliente independente, com os mesmos parâmetros do cliente origem, a partir da passagem de um cliente por dentro dele. O DUMMY NODE é utilizado apenas com o objetivo de roteamento quando o encaminhamento dos clientes se torna muito complexo ou quando deseja-se uma melhor visualização do modelo.

3.2.2 Estrutura de um Programa RESQ2

Um programa RESQ2 tem uma estrutura típica, basicamente constituída das seções apresentadas na Figura 3.5. Pode-se observar uma estrutura aninhada onde os níveis e as formas de aninhamento dependerão diretamente do diagrama proposto do modelo. Da Figura 3.5 pode-se deduzir também a sequência de passos para a execução de um programa RESQ2, ou seja, para a simulação do modelo.

```

MODELO: a
  Definição do método
  Declaração de Variáveis e Constantes
  Declaração de Filas
  Declaração dos Nodes
SUBMODELO: b
  Declaração de Variáveis e Constantes
  Declaração de Filas
  Declaração dos Nodes
SUBMODELO: c
  Declaração de Variáveis e Constantes
  Declaração de Filas
  Declaração dos Nodes
FIM SUBMODELO c
  Invocação do SUBMODELO c
  Encadeamento do SUBMODELO c
FIM SUBMODELO b
  Invocação SUBMODELO b
  Encadeamento do MODELO (final formal do modelo)
  Estatísticas
  Método para Intervalo de Confiança
  Definição dos Estados Iniciais
  Limites de Execução
FINAL DO PROGRAMA

PASSOS PARA A EXECUÇÃO DA SIMULAÇÃO

EVAL
MODEL: ...
  ---
  Parâmetros do modelo em aberto pedidos pelo RESQ
  ---
    
```

Figura 3.5 - Estrutura de programa RESQ2 e modo de execução.

Pode-se observar que as seções que constituem os diversos ninhos de um programa RESQ2 se repetem a nível de modelo e de submodelos, excetuando-se o método aplicado para a solução do modelo. Será apresentado a seguir de forma sucinta, o significado de cada uma das seções:

- . *Definição do Método:* A ferramenta RESQ2 permite a utilização de dois métodos para a solução do modelo: SIMULATION ou NUMERIC; SIMULATION é obtenção da solução

de um modelo através do processo de simulação propriamente dito e NUMERIC é a solução analítica do modelo através de equações matemáticas que relacionam os parâmetros do modelo com as medidas de desempenho desejadas, oferecendo como resultado valores exatos das medidas de avaliação de desempenho.

Declaração de Variáveis e Constantes: Existe basicamente três tipos de declarações: NUMERIC PARAMETERS, NUMERIC IDENTIFIERS e GLOBAL VARIABLES. A declaração NUMERIC PARAMETERS é seguida de nomes de variáveis cujos valores são definidos em tempo de encadeamento de submodelo ou de execução do modelo. Os NUMERIC IDENTIFIERS têm uma lista de nomes aos quais estão associados valores constantes; por exemplo, pode-se definir como NUMERIC IDENTIFIERS a distância em metros entre duas estações, `dist_est: 30` (metros), e depois, dentro da área de processamento do modelo, apenas usar o nome `dist_est`; quando se desejar mudar o valor da distância entre duas estações, não é necessário ter acesso a diversas partes do programa, apenas a esta área. GLOBAL VARIABLES são variáveis que podem ser modificadas no decorrer da execução de um modelo; podem ser aqui definidos contadores e armazenadores de propósito geral.

Declaração das Filas: Neste ponto do programa são definidas as ACTIVE QUEUES e as PASSIVE QUEUES.

- . *Declaração dos Nodos:* onde define-se os NODES do modelo.
- . *Encadeamento:* O encadeamento permite a interligação dos elementos dos modelos/submodelos. Neste ponto é definida toda a lógica de funcionamento do modelo, especificando todo o encaminhamento que o cliente terá dentro do sistema.
- . *Invocação:* A invocação fornece as condições para que o submodelo pronto possa ser encadeado com os outros elementos do submodelo/modelo que o envolve.
- . *Estatísticas:* A ferramenta RESQ2 permite, além das estatísticas padrões que ela oferece [SAUER,1982b], a obtenção de distribuições definidas pelo usuário.
- . *Método para Estimação de Intervalo de Confiança:* O processo de simulação via RESQ2 permite a utilização de um dos três seguintes métodos para estimação de intervalo de confiança: replicações independentes, regenerativo e espectral.
- . *Definição dos Estados Iniciais:* caso seja importante para o modelo que se está simulando, pode-se iniciar a simulação predefinindo condições iniciais.
- . *Limites de Execução:* pode-se impor alguns tipos de término de simulação ao modelo em simulação: por tempo de simulação ou por número de passagens de um cliente por

um determinado elemento do modelo; ou pode-se deixar que o modelo seja simulado até que os critérios de convergência do intervalo de confiança desejado.

A ferramenta RESQ2 denomina os clientes de um modelo de JOB's. Assim, no decorrer da explicação do funcionamento do programa de simulação do protocolo MAC FDDI, em alguns pontos, os QUADROS/FICHA podem ser denominados por JOB's. Associada aos JOB's (clientes do sistema), existe uma categoria de especificação, os JOB VALUES, que são parâmetros associados a cada JOB (cliente) e que podem ser modificados pelos SET NODES. Cada cliente pode ter vários parâmetros que ele carrega durante todo o seu percurso no modelo. Estes parâmetros podem ser utilizados durante o serviço, roteamento e na aquisição de TOK's para acesso às PASSIVE QUEUES. Exemplos de JOB VALUES, para os JOB'S do sistema FDDI, são: tipo do JOB, se FICHA, QUADRO SINCRONO ou QUADRO ASSINCRONO; tamanho do QUADRO ou da FICHA; endereço da estação que originou o QUADRO; endereço da estação a que se destina o QUADRO.

Para se transcrever o programa RESQ2 para o ambiente IBM [SAUER,1982b] existem dois métodos: através de um módulo iterativo chamado SETUP (executado digitando-se **setup** em uma linha de comando IBM) ou através da digitação do programa direto via editor de texto IBM. O SETUP, ao ser executado, inicia um processo de conversação com o usuário da ferramenta, perguntando todas as definições do modelo e seus elementos, e evoluindo, paulatinamente, dentro dos

aninhamentos naturais do programa. Paralelamente ao questionamento acima, o SETUP efetua uma pré-compilação onde já sinaliza erros de sintaxe. Ao término da digitação do programa em RESQ2, o SETUP gera um código compilado para ser executado através do módulo EVAL. Na transcrição via editor de comando (por exemplo: XEDIT), o usuário, ao terminar de digitar o programa, executa `setup` seguido do nome do arquivo que contém o programa. Com isto, o SETUP compila o programa e gera o código executável, fornecendo os possíveis erros de sintaxe. É importante chamar a atenção que na fase de correção de erros de sintaxe é melhor utilizar o comando SETUP em seu modo não conversacional.

3.3 MODELAGEM E CONSTRUÇÃO DO SIMULADOR FDDI

Para o estudo de desempenho do protocolo de controle de acesso ao meio FDDI, daqui por diante referenciado por MAC ("*Medium Access Control*") FDDI, montou-se um programa que simula o funcionamento de toda uma rede em anel, a partir do modelo lógico de uma estação. A modelagem do funcionamento da estação, operando no anel FDDI, baseou-se nas prerrogativas detalhadas no Capítulo 2. Na modelagem proposta o anel é composto de diversas estações do tipo DAS com uma MAC e não tem concentradores.

O modelo do funcionamento de um anel baseado no protocolo MAC FDDI, desenvolvido em RESQ2, permite o estudo dos diversos efeitos de diferentes padrões de configurações da rede e de carga de tráfego. Podem ser definidos pelos

usuários os seguintes parâmetros: taxa média de interchegada de quadros assíncronos baseada em distribuição exponencial, taxa constante de interchegada dos quadros síncronos e o tamanho fixo dos quadros. Os atrasos, tais como: latência das estações, atraso de propagação no cabo, passagem de quadros e ficha, são também parametrizados, neste modelo, pelo usuário. É feita uma implementação completa do mecanismo de acesso FDDI, incluindo detalhes referentes à captura da ficha e à condição de ficha presa para transmissão de quadros SINCRONOS e ASSINCRONOS. Este modelo não inclui detecção de erros e mecanismos de recuperação ou procedimentos de inicialização do anel. O `ttrt` para o anel é definido pelo usuário, pois não há este tipo de negociação entre as estações. A modelagem genérica de cada estação pode ser exemplificada como mostrado na Figura 3.6.

Segundo o padrão FDDI [ISO,1989b], cada estação deve ter o mecanismo de controle de acesso ao meio contemplando os modo de geração de quadros SINCRONOS e ASSINCRONOS. Contudo, a modelagem deste mecanismo para cada uma das estações do anel tornaria o programa muito complexo, implicando em tempos de simulação muito longos. Assim, sem perda de generalidade [JOHNSON,1988a,1988b], cada estação terá apenas um modo de geração de quadros: síncrono ou assíncrono.

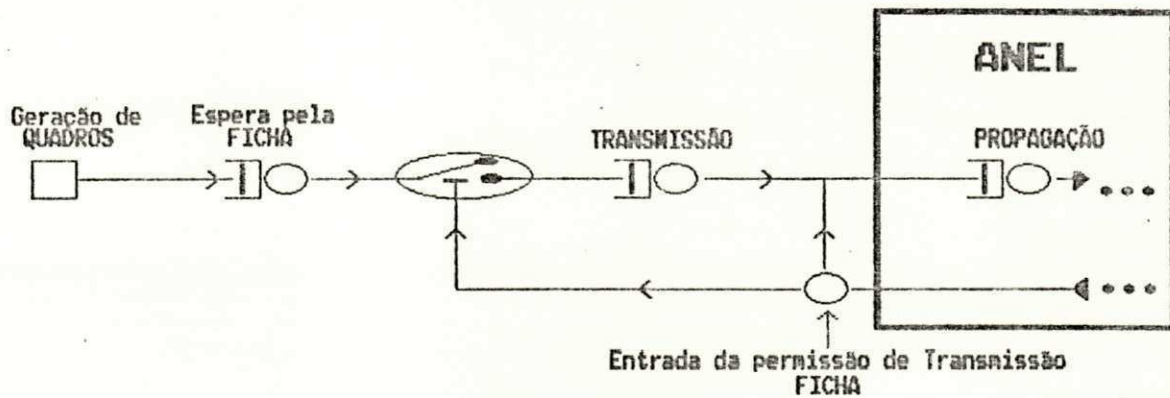


Figura 3.6 - Modelagem genérica de uma estação do anel FDDI.

O número e o tipo de estações no anel podem ser dimensionados conforme a carga síncrona, assíncrona ou mista, a que se deseje submeter o anel. O diagrama de blocos apresentado na Figura 3.7 descreve a filosofia para elaboração do modelo de uma estação MAC FDDI.

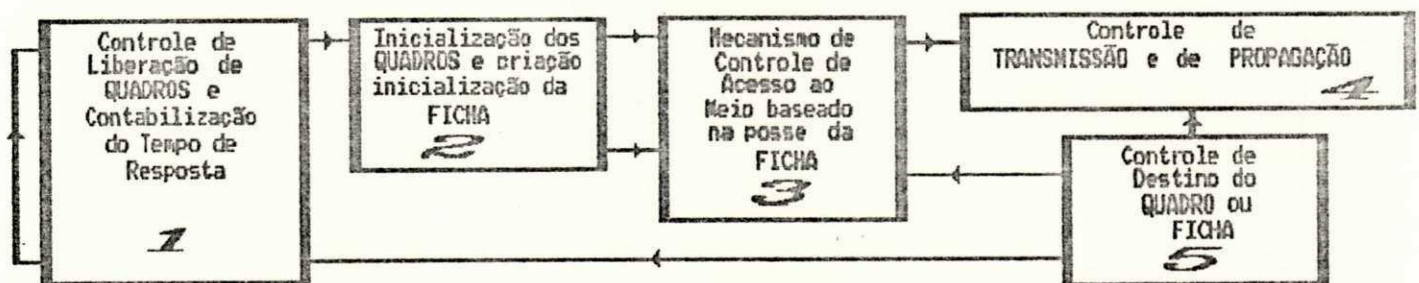


Figura 3.7 - Diagrama de blocos de uma estação MAC FDDI.

Dentro de cada um dos blocos ilustrados na Figura 3.7 estão embutidos elementos da linguagem RESQ2 para execução da simulação. Ilustrando a descrição de uma estação SINCRONA (que é muito similar à estação ASSINCRONA), a Figura 3.8 apresenta o seu diagrama completo com elementos RESQ2.

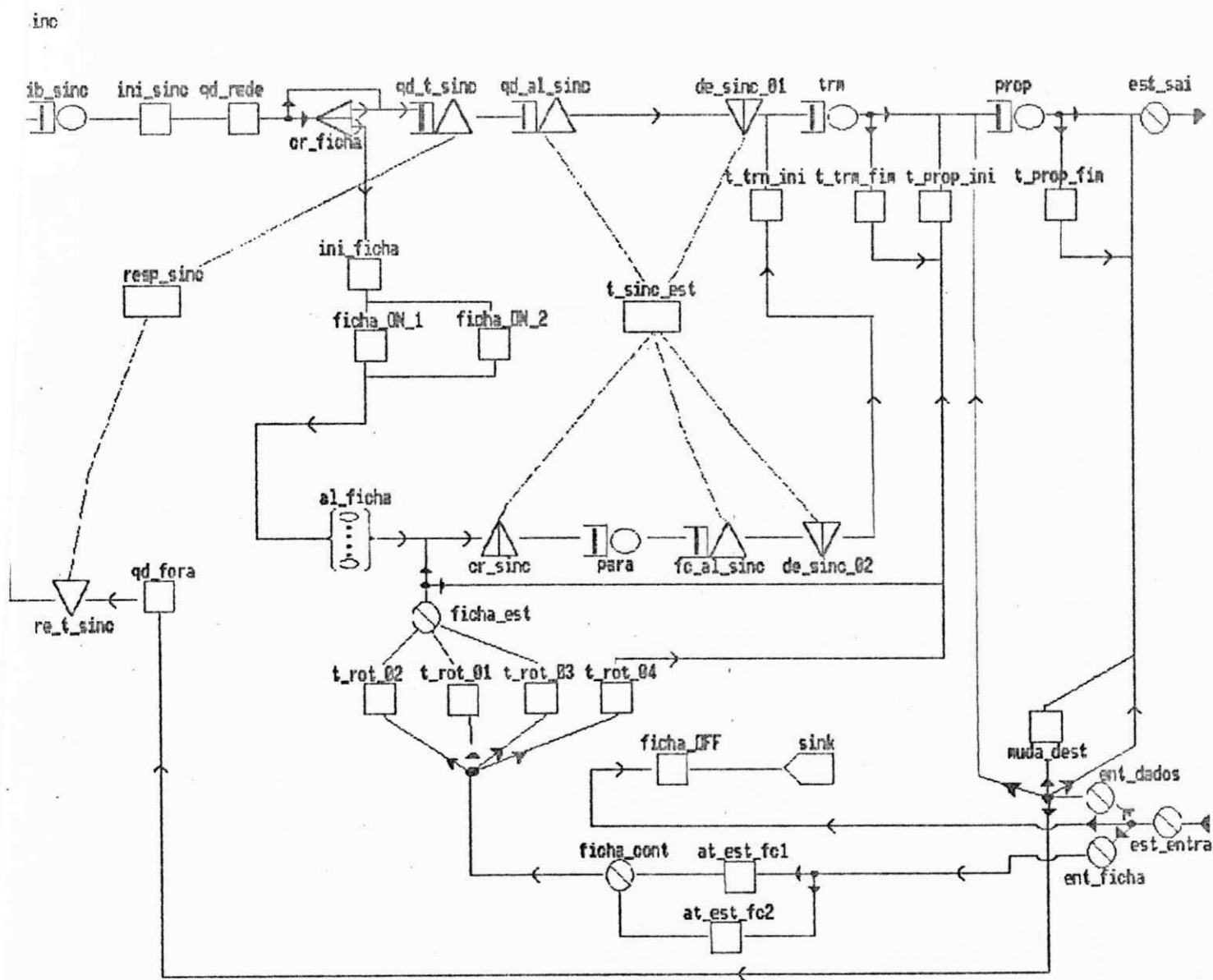


Figura 3.8 - Diagrama completo da estação SINCRONA em RESQ2.

Os diagramas completos dos dois tipos de estações e da estação para controle da FICHA podem ser encontrados no Anexo I. No programa final para simulação do anel FDDI, visando o estudo de desempenho do protocolo de controle de acesso ao meio, foram definidos os dois tipos de estações (SINCRONA e ASSINCRONA) como submodelos e um terceiro submodelo, representando a estação que efetua o controle de circulação da FICHA no anel.

Na simulação de uma determinada configuração de anel FDDI, o usuário da ferramenta RESQ2 necessita apenas concatenar os submodelos referidos, constando apenas de um controle de ficha e de quantas estações síncronas e assíncronas forem necessárias para oferecer a carga proposta ao anel (Anexo III). Na Figura 3.9 é apresentada a estrutura do programa usado para efetuar a simulação do sistema MAC FDDI, misturando a própria linguagem RESQ2 com observações explicativas.

No desenvolvimento do simulador FDDI/RESQ2 utilizou-se uma técnica de modularização do programa principal em sub-programas, visando uma estruturação do trabalho de avaliação de desempenho do protocolo MAC FDDI. Existem dois arquivos de programa que devem ser associados, via recursos de editor de programa, em tempo de compilação para, então, gerar um programa único que efetue as simulações desejadas (Anexo III). Um arquivo, denominado daqui por diante de **arquivo da rede** (Anexos II.2, II.3 e II.4), contém todas as variáveis de simulação do programa (NUMERIC IDENTIFIERS, NUMERIC PARAMETERS e GLOBAL

VARIABLES) e o encadeamento das estações do modelo. Em outro arquivo, denominado daqui por diante de **arquivo de submodelos** (Anexo II.1), estão definidos os submodelos para estação SINCRONA, estação ASSINCRONA e estação de controle da FICHA (com ACTIVE QUEUES, PASSIVE QUEUES, SET NODES e DUMMY NODES).

```

MODEL: FDDI
METHOD: SIMULATION
.Definição dos NUMERIC PARAMETERS, NUMERIC IDENTIFIERS e GLOBAL VARIABLES
.Nesta área estão definidas todas as variáveis e a maioria das constantes do programa.
SUBMODEL: ESTS_SINC
NUMERIC PARAMETERS: J /* Indica a numeração desta estação no anel*/
.DEFINIÇÃO DO FUNCIONAMENTO de uma estação do tipo SINCRONA através dos elementos RESQ2; ACTIVE
QUEUES, PASSIVE QUEUES, SET NODES e DUMMY NODES.
.ENCADEAMENTO dos elementos internos da estação SINCRONA.
END OF SUBMODEL ESTS_SINC
SUBMODEL: ESTS_ASSIN
NUMERIC PARAMETERS: J /* Indica a numeração desta estação no anel*/
.DEFINIÇÃO DO FUNCIONAMENTO de uma estação do tipo ASSINCRONA através dos elementos RESQ2;
ACTIVE
QUEUES, PASSIVE QUEUES, SET NODES e DUMMY NODES.
.ENCADEAMENTO dos elementos internos da estação ASSINCRONA.
END OF SUBMODEL ESTS_ASSIN
SUBMODEL: CT_FICHA
.CONTROLA o tempo médio e máximo de rotação de uma FICHA no anel através de PASSIVE QUEUE.
.ENCADEAMENTO dos elementos internos da estação de controle da FICHA.
END OF SUBMODEL CT_FICHA
.INVOCAÇÃO dos SUBMODELOS para ser feito o encadeamento de forma a construir o modelo do sistema
FDDI
com o número de estações desejadas.
.ENCADEAMENTO final do MODELO construindo os caminhos que os QUADROS deverão percorrer dentro de
todo o
sistema.
.MÉTODO PARA ANÁLISE DO INTERVALO DE CONFIANÇA
.DEFINIÇÃO DOS ESTADOS INICIAIS
.LIMITES DE EXECUÇÃO
END OF MODEL
    
```

Figura 3.9 - Estrutura do programa de simulação FDDI/RESQ2

O usuário da ferramenta de simulação FDDI/RESQ2, quando deseja formar a rede e modificar qualquer um dos diversos parâmetros desta, tais como distância entre

estações, tipos de estações no anel, número de estações no anel, `ttrt`, dentre outros, precisa apenas alterar o primeiro arquivo. O segundo arquivo, arquivo de submodelos, para fins de simulação de sistemas FDDI, em nenhum momento precisará ser modificado.

No arquivo de rede aborda-se, em um primeiro estágio, a especificação das variáveis de controle de funcionamento do protocolo e de parametrização do QUADRO/FICHA. A nomenclatura em termos de definição de variáveis é fundamental na explicação do funcionamento do programa de simulação desenvolvido.

A seguir, será descrito o funcionamento do modelo em RESQ2 de cada bloco identificado na Figura 3.7, explicando o seccionamento do diagrama da estação, através da parte do programa associada e seu encadeamento. Também será abordado o funcionamento da estação de controle da FICHA através da mesma metodologia. Como foi mencionado anteriormente, será apresentada basicamente a estação SINCRONA, incluindo as diferenças entre ela e a estação ASSINCRONA. Seguindo a própria sequência de funcionamento de uma estação, a apresentação do funcionamento do modelo MAC FDDI via RESQ2 passa pelos seguintes pontos: bloco de controle de liberação de quadros e contabilização do tempo de resposta, bloco de inicialização dos QUADROS e criação/inicialização da FICHA, bloco mecanismo de controle de acesso ao meio baseado na posse da FICHA, bloco de controle de transmissão e propagação de QUADROS e da FICHA, e bloco de controle de destinação do QUADRO/FICHA.

3.3.1 Controle de Liberação de Quadros e Contabilização do Tempo de Resposta

No modelo do sistema MAC FDDI em RESQ2, os QUADROS (JOB's) não são criados e destruídos, e sim inseridos inicialmente em uma quantidade calculada em um SET NODE, suficiente para que sempre haja QUADROS sendo oferecidos para a estação. Após serem enviados para a estação destino, os QUADROS retornam à estação origem. É computado o *tempo de resposta* e os quadros de retorno entram novamente em fila para serem retransmitidos pela mesma estação origem. As Figuras 3.10 e 3.11 representam a modelagem deste bloco graficamente e em termos de programa, respectivamente.

A cadeia de uma estação do anel é definida como fechada (Fig. 3.7), devendo-se estabelecer, no estado inicial de simulação, qual é a população de QUADROS (JOB's) a inserir na cadeia. O parâmetro INIT POP (Anexo II.2) define este valor.

Os QUADROS são inseridos no SET NODE *gersinc* onde para cada QUADRO é atribuído um parâmetro, denominado em RESQ2, JOB VALUE (ou resumidamente por JV tipo=sinc). Seguindo, cada QUADRO entra na ACTIVE QUEUE *lib_sinc*, simulando a geração de QUADROS pela estação baseada segundo uma taxa de interchegada com valor constante definida pelo produto dos NUMERIC IDENTIFIERS <tint_sinc> e <tmp_cv(j)>. Associado a este processamento é computado o *tempo de resposta* de cada QUADRO, considerado como o tempo que um QUADRO leva para sair da estação originária e chegar até a estação destino.

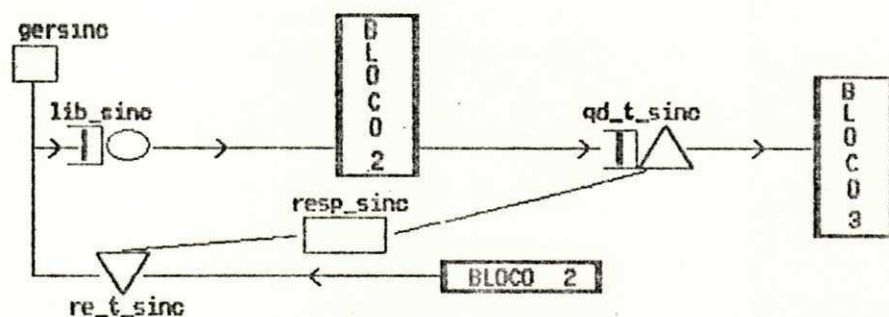


Figura 3.10 - Bloco Controle de Liberação de Quadros e Contabilização do Tempo de Resposta.

Submodelo

```

SUBMODEL: ests_sinc
  NUMERIC PARAMETERS: j /*Indica a estacao atual*/
  CHAIN PARAMETERS: cadeia
  QUEUE: lib_sincq /* Controla liberacao dos qds para o anel*/
    TYPE: fcfs
    CLASS LIST: lib_sinc
    SERVICE TIMES: constant(tint_sinc*tmp_cv(j)) /*(seg/qd)*/
  ...
  QUEUE: resp_sinc /*Controla o tempo de resposta sincrono*/
    TYPE: passive
    TOKENS: 2147483647 /*infinita*/
    DSPL: fcfs
    ALLOCATE NODE LIST: qd_t_sinc
    NUMBER OF TOKENS TO ALLOCATE: 1
    RELEASE NODE LIST: re_t_sinc
  ...
  SET NODES: gersinc /*Colocacao dos QUADROS sincronos*/
    ASSIGNMENT LIST:jv(tipo)=sinc
  
```

Encadeamento

```

CHAIN: cadeia
  TYPE: external
  INPUT: est_entra
  OUTPUT: est_sai
  :gersinc -> lib_sinc -> ini_sinc -> qd_rede
  ...
  :qd_fora -> re_t_sinc -> lib_sinc
  
```

Figura 3.11 - Seção do programa do Bloco Controle de Liberação de Quadros e Contabilização do Tempo de Resposta.

Nesta modelagem, *tempo de resposta* de um QUADRO é obtido através do recurso da PASSIVE QUEUE *resp_sinc*.

onde o QUADRO pega uma TOK no ALLOCATE NODE *gd_t_sinc* e, em seguida, é transmitido no anel até a estação destino, passando pelo processo de transmissão da estação origem e de propagação pelas estações do anel. Cada QUADRO, ao chegar na estação destino, retorna à estação origem sem efetuar nenhum processamento de propagação, assemelhando-se a um curto-circuito ligando as duas estações. No retorno de um QUADRO à estação de origem, é liberada a TOK no RELEASE NODE *re_t_sinc*. Assim, através de uma estatística própria da ferramenta RESQ2 que informa o tempo médio de uso de uma TOK, obtém-se o *tempo de resposta médio* de QUADROS.

Após um QUADRO retornar à estação de origem e liberar a respectiva TOK, ele é novamente enfileirado para entrar no processo de simulação de geração de QUADROS pela estação. O processamento deste bloco é similar nas estações ASSINCRONAS, apenas modificando-se os nomes dos elementos.

3.3.2 Bloco Inicialização dos Quadros e Criação/Inicialização da Ficha

Após o processo de sua "geração" um QUADRO passa pelo processo de inicialização, que é uma complementação dos seus parâmetros. O processo de inicialização de um QUADRO depende da "chegada" da FICHA para continuar o seu encaminhamento para a estação destino. O detalhamento do bloco 2 seccionado do diagrama principal (Fig. 3.7) e a programação RESQ2 associada a este bloco são apresentados nas Figuras 3.12 e 3.13, respectivamente.

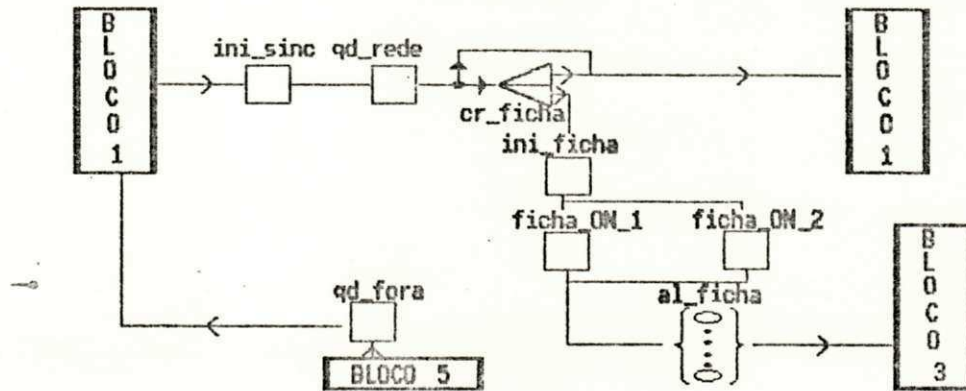


Figura 3.12 - Bloco Inicialização dos Quadros e Criação /Inicialização da Ficha.

Após ser servido pela ACTIVE QUEUE *lib_sinc* (Fig. 3.10), cada QUADRO passa pelo SET NODE *ini_sinc*, onde seus JV's (<tamanho>, <origem>, <destino> e <tmp_cv(j)>), são atualizados. O <tamanho> de um QUADRO é determinado pelo NUMERIC IDENTIFIER <tam_men>. A <origem> é o índice <j>, um NUMERIC PARAMETER do submodelo da estação que é definido em tempo de INVOCATION. A estação destino baseia-se no algoritmo descrito na equação $jv(\text{destino}) = \dots$ (Fig. 3.13), que possibilita que cada QUADRO seja destinado para qualquer estação com mesma probabilidade, excluindo a estação de origem. No SET NODE *ini_sinc* é controlada a GLOBAL VARIABLE <tmp_cv>, cujo objetivo é evitar a geração simultânea de QUADRO SINCRONO pelas estações SINCRONAS, no mesmo tempo simulado.

Submodelo

```

SUBMODEL: ests_sinc
  NUMERIC PARAMETERS: j /*Indica a estacao atual*/
  CHAIN PARAMETERS: cadeia
...
  QUEUE: al_ficha /* Controla liberacao dos qds para o anel*/
  TYPE: is
  CLASS LIST: al_ficha
  SERVICE TIMES: constant(alea_ficha)
...
  SET NODES: ini_sinc /*Inicializa JV's dos QUADROS sincronos*/
  
```

```

(cont. figura 3.13)
  ASSIGNMENT LIST:
jv(tamanho)=tam_men ++
jv(origem)=j ++
jv(destino)=ceil(uniform(1,(jv(origem)-1),(jv(origem)-1)/(mx_est-1);(j++
v(origem)+1),mx_est,(mx_est-jv(origem))/(mx_est-1))) ++
tmp_cv(j)=1
  SET NODES: qd_rede /*Incrementa quadros rodando no anel*/
  ASSIGNMENT LIST: n_qd_anel=n_qd_anel+1 ++
n_qd_sin=n_qd_sin+1
  SET NODES: qd_fora /*Decrementa quadros rodando no anel*/
  ASSIGNMENT LIST: n_qd_anel=n_qd_anel-1
  SET NODES: ini_ficha /*Inicializa JV's da nova FICHA criada*/
  ASSIGNMENT LIST:jv(tipo)=ficha ++
jv(tamanho)=tam_fic ++
ficha_anel=1 ++
n_vt_vaz=((CLOCK-fim_trab)/((at_metro*dist_est*mx_est)+(at_estacao*mx ++
est))) ++
fc_n_est=fc_n_est+1 ++
fc_rotac=fc_rotac+n_vt_vaz ++
fc_t_rotad=CLOCK/fc_rotac
  SET NODES: ficha_ON_1 /*Atualiza inf. rel. a criacao FICHA */
  ASSIGNMENT LIST: /* n_vt_vaz < 1 */ ++
alea_ficha=(CLOCK-fim_trab)
  SET NODES: ficha_ON_2 /*Atualiza inf. rel. a criacao FICHA */
  ASSIGNMENT LIST: /* n_vt_vaz >= 1 */ ++
alea_ficha=uniform(0,((at_metro*dist_est*mx_est)+(at_estacao*(mx_est-1++
))),1)
++
trt(*)=constant((at_metro*dist_est*mx_est)+(at_estacao*(mx_est-1)))

Encadeamento
  CHAIN: cadeia
  TYPE: external
  INPUT: est_entra
  OUTPUT: est_sai
...
:qd_rede -> qd_t_sinc cr_ficha; if(ficha_anel=1) if(t)
:cr_ficha -> qd_t_sinc ini_ficha; split
:ini_ficha -> ficha_ON_1 ficha_ON_2; if(n_vt_vaz<1) if(t)
:ficha_ON_1 ficha_ON_2 -> al_ficha -> cr_sinc

```

Figura 3.13 - Seção do programa do Bloco Inicialização dos Quadros e Criação/Inicialização da Ficha.

Do SET NODE *ini_sinc*, cada QUADRO passa para o SET NODE *qd_rede*, onde são modificadas as GLOBAL VARIABLES <n_qd_anel>, que controla o número de QUADROS tramitando no anel, e <n_qd_sin>, que atualiza o número de QUADROS SINCRONOS inseridos no anel. O SET NODE *qd_rede* opera em conjunto com o SET NODE *qd_fora*, atualizando exatamente o número de QUADROS transitando no anel através da GLOBAL

VARIABLE <n_qd_anel>. Este parâmetro é fundamental para o controle da FICHA no anel, como será visto adiante.

Após ser parametrizado, cada QUADRO espera a chegada da FICHA, podendo ocorrer dois tipos de eventos: o QUADRO efetivamente espera a chegada da FICHA, caso a GLOBAL VARIABLE <ficha_anel> = 1, ou cria uma nova FICHA, caso <ficha_anel> = 0. A FICHA, neste modelo, necessita ser destruída quando não houver tráfego a ser inserido na rede. Caso contrário, o número de QUADROS submetidos ao anel pelas estações será muito pequeno em relação ao tempo de simulação, implicando em medidas de desempenho pouco confiáveis. Dessa forma, principalmente sob baixa *carga oferecida*, a destruição da FICHA com o anel inativo é imprescindível.

A GLOBAL VARIABLE <ficha_anel>, que identifica a existência ou não de FICHA no anel, é atualizada no Bloco de Controle de Destino do QUADRO/FICHA (Bloco 5 da Fig. 3.7). Neste bloco (Bloco 5), testa-se a existência de QUADROS esperando para serem transmitidos através da GLOBAL VARIABLE <n_qd_anel>. Em caso negativo, a FICHA é destruída e o tempo simulado salta para o próximo evento, a geração de um novo QUADRO.

Caso uma estação necessite criar uma nova FICHA para o anel, cada QUADRO passa pelo SPLIT NODE *cr_ficha*, criando uma cópia do JOB QUADRO, que será uma FICHA. A FICHA passa pelo SET NODE *ini_ficha* para atualizar os JV's da FICHA e as GLOBAL VARIABLES para obtenção de estatísticas. No algoritmo para criação da FICHA prevê-se a

possibilidade dela ser destruída e recriada em um mesmo ciclo, através de um ajuste na GLOBAL VARIABLE <alea_ficha> nos SET NODES *ficha_on_1* e *ficha_on_2*.

Na sequência de processamento ilustrada na Figura 3.12, a FICHA passa pela ACTIVE QUEUE infinity server *al_ficha*, com serviço constante determinado pela GLOBAL VARIABLE <alea_ficha> [Anexo I], simulando o tempo que a FICHA levaria para chegar à estação. O processamento do bloco "Inicialização dos QUADROS e Criação/Inicialização da FICHA" para estações assíncronas é similar, modificando-se apenas os nomes dos elementos (Anexo II.1).

3.3.3 Mecanismo de Controle de Acesso ao Meio Baseado na Posse da Ficha

O mecanismo de controle de acesso ao meio do protocolo MAC FDDI, como visto no Capítulo 2, envolve um processamento de relativa complexidade. Sua implementação, em termos de modelo de simulação com a ferramenta RESQ2, necessita de vários "temporizadores" de simulação, paralelos ao próprio relógio do simulador, denominado na ferramenta RESQ2 pela variável CLOCK.

No protocolo MAC FDDI requer-se que o *tempo de rotação médio* da FICHA seja controlado pela variável *ttrt*. O *ttrt* representa, na verdade, a quantidade de banda, em termos de tempo disponível, para todas as transmissões das estações pertencentes ao anel, sejam síncronas ou assíncronas. Neste protocolo, além do *ttrt* tem-se as variáveis de protocolo *trt*, *late_ct* e *tht* cujas funcionalidades foram implementadas neste trabalho con-

forme a especificação do padrão [ISO/FDDI,1989b]. A variável de simulação <ttrt> não é negociável, sendo definida pelo usuário no arquivo de rede (Anexo II). As variáveis de simulação <trt>, <late_ct> e <tht> são definidas por cada estação do anel como GLOBAL VARIABLES do tipo vetor, e daqui por diante serão referenciadas vinculadas à variável de simulação <j>. Por exemplo, a variável <trt> pertencente à j-ésima estação será referenciada no texto por <trt(j)>.

Na modelagem proposta neste trabalho para o funcionamento do protocolo MAC FDDI, assume-se que todas as variáveis de simulação <trt(j)> e <late_ct(j)>, utilizadas como temporizadores, são inicializadas com zero e expiram quando chegam ao valor do <ttrt>. A variável <trt(j)> é usada em cada estação para medir o tempo entre duas chegadas sucessivas da FICHA, e é zerada cada vez que a ficha é recebida dentro do <ttrt>. O <trt(j)> irá expirar se ele for incrementado até <ttrt> antes que a FICHA tenha chegado à estação. Quando a <trt(j)> expira, a <late_ct(j)> é incrementada de um e a <trt(j)> é recarregada com zero, continuando a temporização. Quando a FICHA chega tarde em uma estação (<late_ct(j)=1>), a variável <trt(j)> não é zerada, continuando a temporização, acumulando, portanto, o atraso de rotação atual para a temporização da próxima rotação de FICHA. O resultado do acúmulo do atraso, na rotação da FICHA, é o seguinte: como a rotação da FICHA excede <ttrt> por um tempo A, as transmissões assíncronas serão restringidas até que este atraso seja compensado por

rotações da FICHA menores que $\langle ttrt \rangle$ de um tempo A. Quando o anel recupera o atraso, a(s) $\langle late_ct(j) \rangle$ (s) são naturalmente zeradas. Todo este procedimento assegura que a média de rotação da FICHA é $\langle ttrt \rangle$. Se a $\langle late_ct(j) \rangle$ exceder 1, uma variável de simulação denominada $\langle ab_rede(j) \rangle$, zerada no início da simulação, é incrementada de um, mostrando que houve uma extrapolação do *tempo de rotação máximo* da FICHA ($2 \times ttrt$). A variável $\langle ab_rede(j) \rangle$ permite obter-se dados de desempenho do protocolo sob a carga em estudo.

A implementação dos procedimentos descritos acima, para controlar o acesso ao meio de uma estação FDDI, exige alguns artifícios da linguagem RESQ2 que serão detalhados nas próximas seções. Como o processamento é distinto para as estações síncronas e assíncronas, a implementação de cada tipo de estação na linguagem RESQ2 será apresentada individualmente.

No simulador FDDI/RESQ2 desenvolvido, a variável $\langle tht \rangle$ foi substituída pela variável denominada $\langle assin_est(j) \rangle$, apenas por razões de sintaxe. Assim, daqui por diante, todas as funções atribuídas à variável $\langle tht \rangle$ serão desempenhadas pela variável $\langle assin_est(j) \rangle$. A variável $\langle assin_est(j) \rangle$ é usada, por cada estação, como temporizador, para controlar a quantidade de tempo que a FICHA fica presa em transmissões assíncronas. O temporizador $\langle assin_est(j) \rangle$ é carregado com o valor do $\langle ttrt \rangle - \langle trt(j) \rangle$ quando a FICHA é recebida dentro do tempo normal ($\langle late_ct \rangle = 0$). As transmissões ainda em progresso

são completadas quando o $\langle \text{assin_est}(j) \rangle$ expira. O tempo residual que segue-se à expiração do temporizador $\langle \text{assin_est}(j) \rangle$ é conhecido por "asynchronous overrun".

O método usado neste trabalho para a modelagem em RESQ2 do processamento das estações FDDI baseia-se no funcionamento das temporizações. Como no padrão FDDI, todas as variáveis dependem de temporização para controle do seu funcionamento, utilizou-se o tempo como referência para se definir parâmetros do tipo: banda disponível para transmissão por estação, quantidade de dados a serem transmitidos, perdas por estação e perdas de propagação. Ou seja, em nenhum momento da modelagem proposta utiliza-se diretamente unidades do tipo bps ou bits.

3.3.3.1 Implementação da Sub-Camada MAC de Estação SINCRONA

A sub-camada MAC de estação SINCRONA tem sua modelagem gráfica em RESQ2 apresentada na Figura 3.14. A seção do programa RESQ2 relacionada com esta descrição gráfica é encontrada na Figura 3.15.

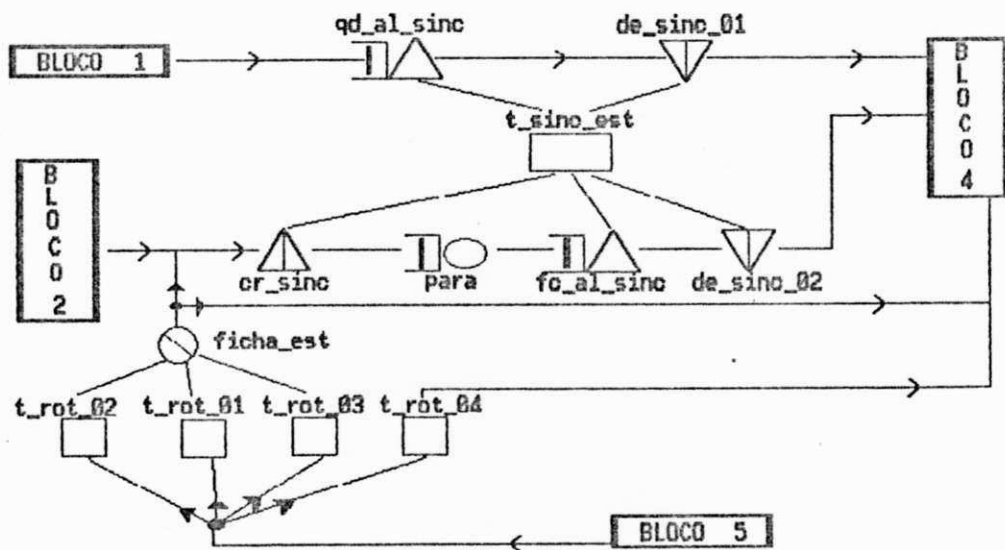


Figura 3.14 - Bloco MAC de Estação SINCRONA.

Submodelo

```

SUBMODEL: ests_sinc
  NUMERIC PARAMETERS: j /*Indica a estacao atual*/
  CHAIN PARAMETERS: cadeia
...
  QUEUE: t_sinc_est /*Controla banda de trm. sincrona*/
    TYPE: passive
    TOKENS: 0
    DSPL: fcfs
    ALLOCATE NODE LIST: qd_al_sinc fc_al_sinc
      NUMBERS OF TOKENS TO ALLOCATE: 1 ta(t_sinc_est)
    DESTROY NODE LIST: de_sinc_01 de_sinc_02
    CREATE NODE LIST: cr_sinc
      NUMBERS OF TOKENS TO CREATE: ceil(sinc_est/t_trm_qd)
  QUEUE: para_q /* Controla parada da ficha para liberacao dos qds*/
    TYPE: fcfs
    CLASS LIST: para
    SERVICE TIMES: 0
...
  SET NODES: t_rot_01 /*Modifica GV's p/ trt>ttrt e late_ct=0 */
    ASSIGNMENT LIST: /*Ficha atrasada e rotacao anterior OK */ ++
late_ct(j)=1 /* Avisa atraso rotacao atual */ ++
trt(j)=trt(j)-ttrt /* Acumula latencia */ ++
assin_est(j)=0 /* Nao transmite quadros assin. */
  SET NODES: t_rot_02 /*Modifica GV's p/ trt<ttrt e late_ct=0 */
    ASSIGNMENT LIST: /*Ficha em tempo e rotacao anterior OK */ ++
assin_est(j)=ttrt-trt(j) /* Transmite quadros assin. */ ++
trt(j)=0 /* trt atual e resetado*/
  SET NODES: t_rot_03 /*Modifica GV's p/ trt<ttrt e late_ct=1 */
    ASSIGNMENT LIST: /*Ficha em tempo e rotacao ant. atrasada*/ ++
late_ct(j)=0 /* Avisa rotacao atual OK */ ++
trt(j)=0 /* trt e resetado */ ++
assin_est(j)=0 /* Nao transmite quadros assin. */
  SET NODES: t_rot_04 /*Modifica GV's p/ trt>ttrt e late_ct=1 */
    ASSIGNMENT LIST: /*Ficha atrasada e rot. ant. atrasada */ ++
late_ct(j)=1 /* Avisa atraso rotacao atual */ ++
trt(j)=trt(j)-ttrt /* Acumula latencia */ ++
assin_est(j)=0 /* Nao transmite quadros assin. */ ++
ab_rede(j)=ab_rede(j)+1 /* E nao transmite quadros sincronos */

```

Encadeamento

```

CHAIN: cadeia
  TYPE: external
  INPUT: est_entra
  OUTPUT: est_sai
...

```

```

:qd_t_sinc -> qd_al_sinc -> de_sinc_01 -> trm
...

```

```

:ficha_cont -> t_rot_01 t_rot_02 t_rot_03 t_rot_04; ++
if(if(trt(j)>ttrt) and if(late_ct(j)=0)) /*F Atr. e ant. OK*/ ++
if(if(trt(j)<ttrt) and if(late_ct(j)=0))/*F OK e ant. OK*/ ++
if(if(trt(j)<ttrt) and if(late_ct(j)=1))/*F OK e ant. atr.*/ ++

```



```
(cont. figura 3.15)
if(if(trt(j))>ttrt) and if(late_ct(j)=1) /*Aborta Rede*/
:t_rot_01 t_rot_02 t_rot_03 -> ficha_est
:ficha_est -> t_prop_ini cr_sinc;
if(ql(qd_al_sinc)=0) if(t)
:cr_sinc -> para -> fc_al_sinc
:fc_al_sinc -> de_sinc_02 -> t_trm_ini -> trm
:t_rot_04 -> t_prop_ini
```

Figura 3.15. - Seção do programa do Bloco MAC de Estação SINCRONA.

O processamento do controle de liberação de QUADROS síncronos para transmissão no anel é especificamente pela PASSIVE QUEUE *t_sinc_est*, a partir da chegada da FICHA no CREATE NODE *cr_sinc*. Caso haja QUADROS para serem transmitidos pela *j*-ésima estação síncrona, estes devem estar enfileirados no ALLOCATE NODE *qd_al_sinc*, esperando uma TOK para passar adiante. Cada QUADRO síncrono utiliza uma TOK para continuar o processamento na rede. A FICHA pode vir de dois processamentos: criação, depois da destruição da mesma no anel por outra estação, ou passagem natural pela estação atual proveniente de outra estação. No primeiro caso, a FICHA entra direto no CREATE NODE *cr_sinc* e cria TOK's. No segundo caso, o encaminhamento é mais complexo, devendo ser feitos todos os testes para saber qual banda estará disponível para transmissão naquela estação.

A FICHA, ao chegar a uma estação, pode entrar na estação atual, caso a GLOBAL VARIABLE *<n_qd_anel>* > 0, ou entrar no SINK NODE para ser destruída, caso não haja QUADROS síncronos prontos para serem transmitidos pelas estações do anel. Entrando na estação atual, a FICHA

encontra quatro opções de processamento que são definidas no encadeamento do submodelo da estação síncrono:

```
. if((trt(j)>ttrt) and (late_ct(j)=0)) execute t_rot_01
  /*Ficha atual chega atrasada e anterior chegou em tempo*/
. if((trt(j)<=ttrt) and (late_ct(j)=0)) execute t_rot_02
  /*Ficha atual chega em tempo e anterior chegou em tempo*/
. if((trt(j)<=ttrt) and (late_ct(j)=1)) execute t_rot_03
  /*Ficha atual chega em tempo e anterior chegou atrasada*/
. if((trt(j)>ttrt) and (late_ct(j)=1)) execute t_rot_04
  /*Ficha atual chega atrasada e anterior chegou atrasada*/
  /*Aborta Rede*/
```

A FICHA, passando pelos respectivos SET NODES, executa os procedimentos pertinentes ao protocolo MAC do MAC FDDI. Conforme o padrão [ISO/FDDI,1989b], a estação SINCRONA sempre deverá ter uma banda fixa disponível para transmissão dos seus QUADROS síncronos, definida durante a inicialização do anel através do valor do <ttrt>. Assim, a variável <assin_est(j)> não tem nenhum efeito sobre o comportamento das transmissões síncronas. Após passar pelo SET NODE adequado, a FICHA pode seguir para outra estação caso não haja QUADROS síncronos a serem transmitidos ou, caso contrário, seguir para o CREATE NODE *cr_sinc* para criar as TOK's de transmissão.

A PASSIVE QUEUE *t_sinc_est* controla efetivamente o acesso ao meio dos QUADROS síncronos que desejam ser transmitidos. Inicialmente a PASSIVE QUEUE é criada sem TOK's disponíveis. Portanto, todo e qualquer QUADRO síncrono que deseje ser transmitido, deve parar no ALLOCATE NODE *qd_al_sinc* esperando TOK's para continuar o processamento. Quando a FICHA chega ao CREATE NODE *cr_sinc*, ela estimula a PASSIVE QUEUE *t_sinc_est* a criar um número

de TOK's igual ao arredondamento para mais da divisão dos NUMERIC IDENTIFIERS `<sinc_est>` por `<t_trm_qd>`. A variável `<sinc_est>` define a banda disponível, em segundos, para transmissões SINCRONAS por estação e a variável `<t_trm_qd>` define o tempo, em segundos, para transmissão de um QUADRO. Dessa forma, tem-se como subproduto deste processamento, a geração de um número `x` de TOK's que permite que um número `x` de QUADROS síncronos passem adiante no modelo, ou seja, uma TOK por QUADRO. A FICHA passa pela ACTIVE QUEUE para com tempo de serviço 0 (zero) apenas para permitir que os QUADROS síncronos possam continuar o seu encaminhamento, passando pelo DESTROY NODE `de_sinc_01` para destruir as TOK's adquiridas e entrar na ACTIVE QUEUE `trm` (bloco 4 da Fig.3.7) para serem servidos, ou seja transmitidos para o anel. Este artifício foi necessário, pois, do contrário, a FICHA passaria à frente dos QUADROS síncronos a serem transmitidos pela estação.

Após todos os QUADROS síncronos entrarem na ACTIVE QUEUE `trm` (bloco 4), a FICHA passa pelo ALLOCATE NODE `fc_al_sinc`, ali alocando quantas TOK's tenham sobrado apenas para poder seguir adiante, e as destrói no DESTROY NODE `de_sinc_02`. Com isto, a PASSIVE QUEUE `t_sinc_est` voltará a ficar sem TOK's. Quando novos QUADROS síncronos forem gerados, eles param no ALLOCATE NODE `qd_al_sinc`, esperando que a FICHA chegue na PASSIVE QUEUE `t_sinc_est` para criar TOK's.

3.3.3.2 Implementação da Sub-Camada MAC de Estação ASSINCRONA

A modelagem gráfica da sub-camada MAC de uma estação ASSINCRONA é representada na Figura 3.16 e a seção do programa RESQ2 relacionada com esta descrição gráfica é encontrada na Figura 3.17.

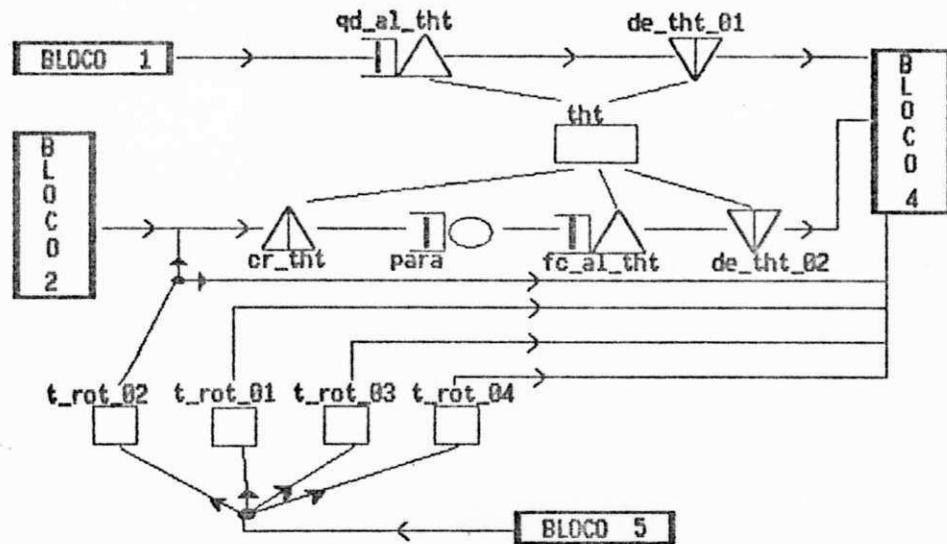


Figura 3.16 - Bloco MAC de Estação ASSINCRONA.

Submodelo

```

SUBMODEL: ests_assin
  NUMERIC PARAMETERS: j /*Indica a estacao atual*/
  CHAIN PARAMETERS: cadeia
  ...
  QUEUE: tht /*Controla banda de trm. assincrona*/
  TYPE: passive
  TOKENS: 0
  DSPL: fcfs
  ALLOCATE NODE LIST: qd_al_tht fc_al_tht
  NUMBERS OF TOKENS TO ALLOCATE: 1 ta(tht)
  DESTROY NODE LIST: de_tht_01 de_tht_02
  CREATE NODE LIST: cr_tht
  NUMBERS OF TOKENS TO CREATE: ceil(assin_est(j))/t_trm_qd
  QUEUE: para_q /* Controla parada da fic para liberacao dos qds*/
  TYPE: fcfs
  CLASS LIST: para
  SERVICE TIMES: 0
  ...
  SET NODES: t_rot_01 /*Modifica GV's p/ trt>ttrt e late_ct=0 */
  ASSIGNMENT LIST: /*Ficha atrasada e rotacao anterior OK */ **
  
```

```
(cont. fig. 3.17)
late_ct(j)=1          /* Avisa atraso rotacao atual */          ++
trt(j)=trt(j)-ttrt   /* Acumula latencia */                    ++
assin_est(j)=0       /* Nao transmite quadros assin. */
    SET NODES: t_rot_02 /*Modifica GV's p/ trt<=ttrt e late_ct=0 */
    ASSIGNMENT LIST: /*Ficha em tempo e rotacao anterior OK */ ++
assin_est(j)=ttrt-trt(j) /* Transmite quadros assin. */        ++
trt(j)=0             /* trt atual e resetado*/
    SET NODES: t_rot_03 /*Modifica GV's p/ trt<=ttrt e late_ct=1 */
    ASSIGNMENT LIST: /*Ficha em tempo e rotacao ant. atrasada*/ ++
late_ct(j)=0        /* Avisa rotacao atual OK */                ++
trt(j)=0            /* trt e resetado */                        ++
assin_est(j)=0      /* Nao transmite quadros assin. */
    SET NODES: t_rot_04 /*Modifica GV's p/ trt>ttrt e late_ct=1 */
    ASSIGNMENT LIST: /*Ficha atrasada e rot. ant. atrasada */  ++
late_ct(j)=1        /* Avisa atraso rotacao atual */            ++
trt(j)=trt(j)-ttrt /* Acumula latencia */                       ++
assin_est(j)=0      /* Nao transmite quadros assin. */          ++
ab_rede(j)=ab_rede(j)+1 /* E nao transmite quadros sincronos */
```

Encadeamento

```
CHAIN: cadeia
TYPE: external
INPUT: est_entra
OUTPUT: est_sai

...
:qd_t_assin -> qd_al_tht -> de_tht_01 -> trm
:trm -> t_trm_fim prop; if(jv(tipo)=ficha) if(t)
:t_trm_fim -> t_prop_ini-> prop
:prop -> t_prop_fim est_sai; if(jv(tipo)=ficha) if(t)
:t_prop_fim -> est_sai
:est_entra -> ent_ficha ficha_OFF ent_dados;

...
:ficha_cont -> t_rot_01 t_rot_02 t_rot_03 t_rot_04;          ++
if(if(trt(j)>ttrt) and if(late_ct(j)=0)) /*F Atr. e ant. OK*/   ++
if(if(trt(j)<=ttrt) and if(late_ct(j)=0))/*F OK e ant. OK*/    ++
if(if(trt(j)<=ttrt) and if(late_ct(j)=1))/*F OK e ant. atr.*/  ++
if(if(trt(j)>ttrt) and if(late_ct(j)=1)) /*Aborta Rede*/
:t_rot_02 -> t_prop_ini cr_tht;
    if(q1(qd_al_tht)=0) if(t)
:t_rot_01 t_rot_03 -> t_prop_ini
:t_rot_04 -> t_prop_ini
:cr_tht -> para -> fc_al_tht -> de_tht_02
:de_tht_02 -> t_trm_ini -> trm
```

Figura 3.17 - Seção do programa do bloco Implementação da MAC Estação ASSINCRONA.

O processamento do controle de liberação de QUADROS assíncronos para transmissão no anel é feito especificamente pela PASSIVE QUEUE *tht*, a partir da chegada da FICHA no CREATE NODE *cr_tht*. Caso haja QUADROS assíncronos

para serem transmitidos pela j -ésima estação assíncrona, estes devem estar enfileirados no ALLOCATE NODE *qd_al_tht*, esperando uma TOK para passar adiante. Cada QUADRO assíncrono utiliza uma TOK para continuar o processamento na rede. Quando a FICHA entra no CREATE NODE *cr_tht* para transmitir QUADROS assíncronos da estação atual, encontra quatro opções de processamento que são definidas no encadeamento do submodelo da estação ASSINCRONA:

```
. if((trt(j)>ttrt) and (late_ct(j)=0)) execute t_rot_01
  /*Ficha atual chega atrasada e anterior chegou em tempo*/
. if((trt(j)<=ttrt) and (late_ct(j)=0)) execute t_rot_02
  /*Ficha atual chega em tempo e anterior chegou em tempo*/
. if((trt(j)<=ttrt) and (late_ct(j)=1)) execute t_rot_03
  /*Ficha atual chega em tempo e anterior chegou atrasada*/
. if((trt(j)>ttrt) and (late_ct(j)=1)) execute t_rot_04
  /*Ficha atual chega atrasada e anterior chegou atrasada*/
  /*Aborta Rede*/
```

Ao passar pelos respectivos SET NODES *t_rot_xx*'s, a FICHA executa os procedimentos pertinentes ao protocolo MAC do FDDI. Para uma estação ASSINCRONA, a banda disponível para transmissão dos QUADROS assíncronos depende da velocidade de rotação da FICHA e do algoritmo associado aos SET NODES *t_rot_xx*'s. Após a passagem da FICHA pelo SET NODE adequado, caso exista banda disponível para transmissão, esta estará acumulada na variável *<assin_est(j)>*. Assim, no CREATE NODE *cr_tht* cria-se um número de TOK's, permitindo a passagem dos QUADROS assíncronos adiante.

Uma observação interessante quanto ao método de simulação utilizado está no fato que todo o processamento, desde a chegada da FICHA no CREATE NODE *cr_sinc* até o enfi-

leiramento dos QUADROS/FICHA para transmissão, ocorre no mesmo tempo simulado. Todas as temporizações a nível de transmissão e de propagação são inseridas pelo bloco (bloco 4 da Fig. 3.7) descrito a seguir.

3.3.4 Controle de Transmissão e de Propagação

Durante todo o processamento da estação, os valores dos temporizadores do anel só são modificados a partir da ação da FICHA. Existem três pontos principais no processamento do modelo, nos quais é feita uma manutenção nos valores dos $\langle trt(j) \rangle$'s que controlam a banda disponível de transmissão em uma dada estação: nos SET NODES t_rot_xx , onde define-se efetivamente a banda disponível e se a FICHA está atrasada ou não; nos SET NODES ligados à ACTIVE QUEUE trm que simula a transmissão de QUADROS e FICHA no meio, e à ACTIVE QUEUE $prop$ que simula a propagação de QUADROS e FICHA no anel. No bloco "Controle de Transmissão e Propagação" é simulada a transmissão e propagação de QUADROS e da FICHA, acumulando nos temporizadores da rede os respectivos tempos de processamento. As Figuras 3.18 e 3.19 ilustram o funcionamento deste bloco.

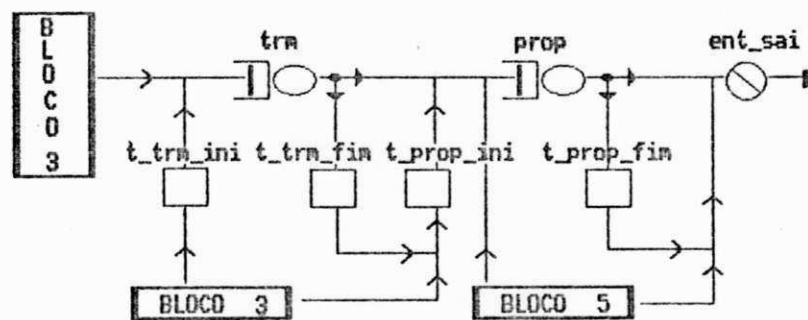


Figura 3.18 - Bloco Controle de Transmissão e de Propagação.

```

Submodelo
  SUBMODEL: ests_sinc
  NUMERIC PARAMETERS: j /*Indica a estacao atual*/
  CHAIN PARAMETERS: cadeia
  ...
  QUEUE: trm_q /*Controla atraso de trm. de quadros/fichas*/
  TYPE: fcfs
  CLASS LIST: trm
  SERVICE TIMES: constant(jv(tamanho)/cap_trm)/*(bits/bps)*/
  QUEUE: prop_q /*Controla atraso de prop. e est. quadros/fichas*/
  TYPE: fcfs
  CLASS LIST: prop
  SERVICE TIMES: constant((at_metro*dist_est)+at_estacao)
  ...
  SET NODES: t_trm_ini /*Inicia proc. atraso de trm. de qds e fc*/
  ASSIGNMENT LIST: ini_trab=CLOCK
  SET NODES: t_trm_fim /*Final. proc. atraso de trm. de qds e fc*/
  ASSIGNMENT LIST: fim_trab=CLOCK ++
  trt(*)=trt(*)+(fim_trab-ini_trab)
  SET NODES: t_prop_ini /*Inicia proc. atraso de prop. de qds e fc*/
  ASSIGNMENT LIST: ini_trab=CLOCK
  SET NODES: t_prop_fim /*Final. proc. atraso de prop. de qds e fc*/
  ASSIGNMENT LIST: fim_trab=CLOCK ++
  trt(*)=trt(*)+(fim_trab-ini_trab)

Encadeamento

  CHAIN: cadeia
  TYPE: external
  INPUT: est_entra
  OUTPUT: est_sai
  ...
  :trm -> t_trm_fim prop; if(jv(tipo)=ficha) if(t)
  :t_trm_fim -> t_prop_ini-> prop
  :prop -> t_prop_fim est_sai; if(jv(tipo)=ficha) if(t)
  :t_prop_fim -> est_sai

```

Figura 3.19 - Seção do programa do Bloco Controle de Transmissão e de Propagação.

O Bloco Controle de Transmissão e de Propagação realiza basicamente dois tipos de processamento: o primeiro relaciona-se com o acúmulo dos tempos de transmissão dos QUADROS e da FICHA em cada estação individualmente nos $\langle trt(j) \rangle$'s e, a segunda, relaciona-se com o acúmulo dos tempos de propagação do QUADRO/FICHA no meio e do tempo de processamento de cada QUADRO/FICHA pelas estações no per-

curso do anel. A transmissão dos QUADROS está ligada apenas aos dados que uma determinada estação deseja inserir no meio, enquanto que a propagação relaciona-se com todos os QUADROS que estão transitando no anel.

Para fazer a contabilização dos referidos tempos utiliza-se a variável do RESQ2 denominada CLOCK, que contém o tempo simulado atual. A variável CLOCK é armazenada pela FICHA, antes do processamento de transmissão, na GLOBAL VARIABLE <ini_trab> e subtraída do valor atualizado, após o processamento. Assim, por exemplo, caso tenha-se três QUADROS na ACTIVE QUEUE *trm*, inicialmente a FICHA passa pelo SET NODE *t_trm_ini* onde armazena o valor do CLOCK atual na GLOBAL VARIABLE <ini_trab> e entra na ACTIVE QUEUE *trm* atrás dos QUADROS a serem servidos. Após ser servida por último, a FICHA passa pelo SET NODE *t_trm_fim* onde subtrai o valor do CLOCK atual da GLOBAL VARIABLE <ini_trab>. Com este procedimento, obtém-se o tempo gasto para a transmissão dos QUADROS, incluindo a FICHA, e atualiza-se os <trt(j)>'s de todas as estações do anel.

Procedimento similar é efetuado na ACTIVE QUEUE *prop*. Porém, é importante observar que, no processamento natural do anel, os atrasos de processamento e de propagação só necessitam serem computados nos <trt(j)>'s se a FICHA estiver entrando no nodo de propagação, pois na análise de um sistema FDDI o que interessa é o atraso da FICHA.

3.3.5 Controle de Destino do QUADRO/FICHA

Um QUADRO pode, ao entrar numa estação, ter dois destinos: a estação atual ou uma outra estação. A FICHA deverá sempre passar pela estação mesmo que não haja QUADROS a serem transmitidos. A Figura 3.20 mostra os percursos possíveis para QUADROS e a FICHA ao chegarem em uma estação, enquanto que a Figura 3.2.1 apresenta a seção do programa RESQ2 associada.

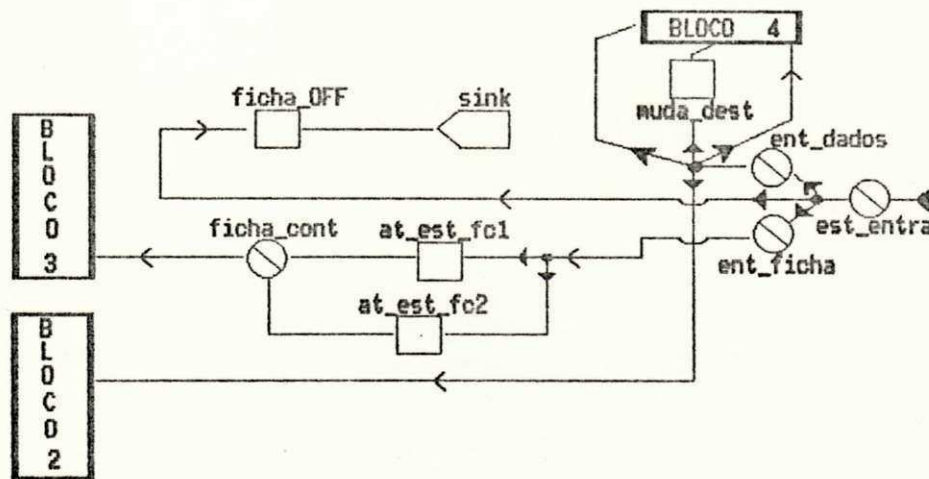


Figura 3.20 - Bloco Controle de Destino do QUADRO/FICHA.

Submodelo

```

SUBMODEL: ests_sinc
  NUMERIC PARAMETERS: j /*Indica a estacao atual*/
  CHAIN PARAMETERS: cadeia
  ...
  SET NODES: ficha_OFF /*Atualiza inf. rel. a destruicao da FICHA*/
  ASSIGNMENT LIST: ficha_anel=0
  ...
  SET NODES: muda_dest /*Muda dest. do QUADRO p/ est. de origem*/
  ASSIGNMENT LIST: jv(destino)=jv(origem)
  SET NODES: at_est_fc1 /* Atualiza Estats da Ficha < uma volta */
  ASSIGNMENT LIST: fc_n_est=fc_n_est+1
  SET NODES: at_est_fc2 /* Atualiza Estats da Ficha = uma volta */
  ASSIGNMENT LIST: fc_n_est=1
  fc_rotac=fc_rotac+1
  fc_t_rotmd=CLOCK/fc_rotac
  
```


(cont. fig. 3.21)

Encadeamento

```

CHAIN: cadeia
  TYPE: external
  INPUT: est_entra
  OUTPUT: est_sai
...
  :est_entra -> ent_ficha ficha_OFF ent_dados;          ++
if(if(jv(tipo)=ficha) and if(n_qd_anel>0))              ++
if(if(jv(tipo)=ficha) and if(n_qd_anel<1)) if(t)
  :ent_dados -> qd_fora muda_dest prop est_sai;        ++
if(if(jv(origem)=jv(destino)) and if(jv(destino)=j))    ++
if(if(jv(origem)^=jv(destino)) and if(jv(destino)=j))  ++
if(if(jv(origem)^=jv(destino)) and if(jv(destino)^=j)) ++
if(if(jv(origem)=jv(destino)) and if(jv(destino)^=j))
...
  :muda_dest -> est_sai
  :ficha_OFF -> sink
  :ent_ficha -> at_est_fc2 at_est_fc1;                  ++
if((fc_n_est)>(mx_est-1)) if(t)
  :at_est_fc2 at_est_fc1 -> ficha_cont

```

Figura 3.21 - Seção do programa do bloco Controle de Destino do QUADRO/FICHA.

Um QUADRO, ao chegar a uma estação, passa pelo DUMMY NODE *ent_dados* e pode ter quatro caminhos:

- . Entrar no SET NODE *qd_fora*, caso esta seja a sua estação origem e os JOB VALUES *jv(origem)* e *jv(destino)* sejam iguais;
- . Entrar no SET NODE *muda_dest*, caso a estação atual seja o destino, efetuar o *jv(destino)=jv(origem)* e iniciar o processo de curto-circuito encaminhando o QUADRO para o DUMMY NODE *est_sai*;
- . Seguir para outra estação, entrando no DUMMY NODE *est_sai*, sem contabilizar tempos de propagação e de processamento da estação; isto ocorre quando o QUADRO está rodando no anel no modo curto-circuito;

- . Seguir para outra estação, entrando na ACTIVE QUEUE *prop*, contabilizando tempos de propagação e de processamento da estação, caso o QUADRO não esteja no modo curto-circuito e esteja sendo repetido no anel.

A FICHA, ao chegar a uma estação, pode ter dois encaminhamentos:

- . Caso existam QUADROS em qualquer estação do anel ($n_qd_anel > 0$), entrar na estação para atualizar as estatísticas de medidas de desempenho desejadas no estudo de simulação, e seguir para os SET NODES *t_rot_xx* que controlam a transmissão de QUADROS da estação;
- . Caso contrário ($n_qd_anel < 1$), entrar no SET NODE *fi-cha_off* para atribuir à GLOBAL VARIABLE <ficha_anel> o valor zero, informando que o anel permanecerá sem FICHA, e destruí-la no SINK NODE.

3.3.6 Estação de Controle da FICHA

A estação de controle da FICHA, cujos submodelos gráfico e textual encontram-se descritos, respectivamente, nas Figuras 3.22 e 3.23, tem o seu funcionamento resumido na obtenção de medidas de desempenho do *tempo de rotação médio* e *máximo* da FICHA no anel, e integração geral das estações do anel.

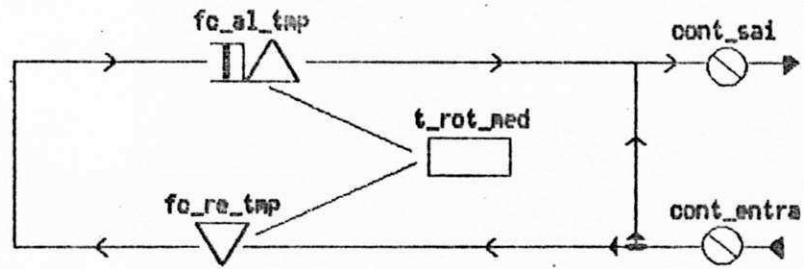


Figura 3.22 - Estação de Controle da Ficha.

Submodelo

```

SUBMODEL: ct_ficha
CHAIN PARAMETERS: cadeia
QUEUE: t_rot_med /*Controla o tempo maximo de rotacao da ficha*/
TYPE: passive
TOKENS: 2147483647 /*infinita*/
DSPL: fcfs
ALLOCATE NODE LIST: fc_al_tmp
    NUMBERS OF TOKENS TO ALLOCATE: 1
RELEASE NODE LIST: fc_re_tmp
DUMMY NODES: cont_entra cont_sai
CHAIN: cadeia
TYPE: external
INPUT: cont_entra
OUTPUT: cont_sai
:cont_entra -> fc_re_tmp cont_sai; if(jv(tipo)=ficha) if(t)
:fc_re_tmp -> fc_al_tmp -> cont_sai /*Inicializacao da ficha*/
END OF SUBMODEL CT_FICHA
    
```

Figura 3.23 - Seção do programa da Estação de Controle da Ficha.

CAPITULO 4

**ESTUDO DE DESEMPENHO
DO
PROTOCOLO FDDI**

4.1 INTRODUÇÃO

Uma das mais importantes utilizações de um simulador está na obtenção de medidas de desempenho que justifiquem os dimensionamentos efetuados em projetos e em possíveis modificações dos parâmetros de sistema, já durante o seu funcionamento. Com o simulador FDDI desenvolvido, pode-se efetuar um estudo do modelo em diversas configurações de variáveis e carga desejadas do anel, efetivando a sua utilização como ferramenta de auxílio no desenvolvimento de projetos e gerenciamento de redes FDDI.

O protocolo MAC FDDI já foi objeto de estudo de desempenho, via simulação através de várias ferramentas, por vários pesquisadores para suas aplicações específicas. Portanto, os resultados obtidos através do simulador FDDI implementado com a ferramenta RESQ2 podem tomar, por referência, resultados existentes na literatura.

Neste Capítulo serão destacados os aspectos mais importantes ligados ao estudo de desempenho de uma rede FDDI. Inicialmente, serão apresentados os elementos que compõem um sistema FDDI em termos de variáveis de sistema a serem parametrizadas e das medidas de desempenho que podem ser obtidas. A seguir apresenta-se alguns estudos já existentes na literatura e que servem de base para a validação dos resultados de desempenho obtidos, bem como um estudo de algumas configurações particulares a título de exemplo de utilização da ferramenta desenvolvida.

4.2 ELEMENTOS PARA O ESTUDO DE DESEMPENHO DO FDDI

De um modo geral, em um estudo de desempenho de modelos de sistemas, existe um universo amplo de medidas e parâmetros de simulação que estão disponíveis para análise do comportamento desses sistemas. Entretanto, no caso de modelos de redes de computadores pode-se destacar um subconjunto deste universo que permite um estudo mais adequado. Nesta seção, enfoca-se apenas as medidas de desempenho e os parâmetros de simulação específicos para o estudo de uma rede FDDI. Uma abordagem mais detalhada sobre parâmetros de simulação e medidas de desempenho de outros tipos de redes e sistemas pode ser encontrada em [SAUER,1985],[SOARES,1990].

Para um estudo de desempenho de uma rede FDDI precisam ser definidos basicamente os parâmetros do sistema para uma configuração específica e quais as medidas de desempenho que se deseja obter da simulação. A ferramenta RESQ2, por ser uma ferramenta de simulação de propósito geral, oferece um espectro muito grande de tipos de resultados acerca da simulação. Entretanto, para a obtenção de resultados globais da simulação via ferramenta RESQ2, alguns cálculos manuais precisam ser efetuados (anexo III).

Na modelagem de rede FDDI desenvolvida neste trabalho, são necessários parâmetros de simulação para caracterizar uma configuração de anel específica. Estes parâmetros são especificados no simulador FDDI/RESQ2 no arquivo de rede (Anexo II.2, II.3 e II.4). Para cada

configuração de anel, os seguintes parâmetros devem ser fornecidos:

- . Velocidade de transmissão;
- . Latência por estação;
- . Atenuação do meio de transmissão;
- . Tamanho do QUADRO e da FICHA;
- . Informações acerca do temporizador (*ttrt*);
- . Número de estações por tipo (SINCRONA ou ASSINCRONA);
- . Carga oferecida ("*offered load*") ao anel por tipo de estação (SINCRONA ou ASSINCRONA).

O padrão FDDI [ISO,1989b] especifica os seguintes valores:

- . Velocidade de transmissão : 100 Mbps
- . Latência máxima por estação : 1,2 μ s
- . Atenuação máxima do meio : 5,085 ns/m
- . Tamanho do Preâmbulo : 8 octetos
- . Tamanho máximo do QUADRO : 4.500 octetos
- . Tamanho da FICHA : 3 octetos
- . *ttrt* mínimo : 4 ms
- . *ttrt* máximo : 167 ms

Conforme descrito no Capítulo 3, as estações FDDI geram dois tipos de tráfegos específicos: SINCRONO e ASSINCRONO. Assim, em cada configuração de anel FDDI, o usuário define o número de estações SINCRONAS e ASSINCRONAS separadamente e, para cada tipo de estação, especifica qual

é a carga que ela oferece ao anel. Entende-se por *carga oferecida* ao anel, a quantidade de informações em bits/segundo que a estação insere no anel. Na modelagem do FDDI através da ferramenta RESQ2, para se obter a *carga oferecida* ao anel, divide-se o tamanho do QUADRO (mais preâmbulo) gerado pela estação pelo tempo de interchegada de QUADROS, ambos parâmetros definidos pelo usuário, e multiplica-se o resultado da divisão pelo número de estações daquele tipo. Por exemplo, seja o seguinte ambiente de simulação:

.Tamanho do QUADRO	: 4040 octetos
.Tempo de Interchegada de QUADROS SINCRONOS:	6×10^{-03} seg
.Número de estações SINCRONAS	: 20

A *carga oferecida* SINCRONA ao anel, neste exemplo, é de 108 Mbps, ou seja, 108%, tendo em vista que a velocidade de transmissão efetiva do anel é igual a 100 Mbps. Uma grande parte dos estudos de desempenho de redes utiliza a *carga oferecida* [JOHNSON,1988b] como um dos parâmetros de variação para se estudar o comportamento da rede.

Na literatura sobre estudos de desempenho de redes de computadores, as medidas de velocidade podem ter nomenclaturas diversas para uma mesma funcionalidade. Neste trabalho, as medidas de desempenho utilizadas para análise do comportamento do anel FDDI são as seguintes:

. *Vazão* ("throughput"): quantidade de bits por segundo trafegando pela rede [bits/segundo];

- . *Utilização* ou *Carga aplicada* ("utilization" ou "application load"): percentual de uso do meio de transmissão que pode ser obtido a partir da *vazão* [%];
- . *Eficiência*: relação entre a banda utilizável, máxima vazão possível sob alta carga em um determinado contexto, e a banda nominal, 100 Mbps para o FDDI; por exemplo, se, para um conjunto de parâmetros de rede e de carga oferecida, a banda utilizável não for nunca maior que 90 Mbps, a eficiência será 90% para aquele contexto [%];
- . *Atraso de acesso* ("access delay"): tempo que um QUADRO espera para obter a FICHA de permissão de acesso ao anel [segundos];
- . *Tempo de resposta* ("response time"): tempo entre a chegada de um QUADRO proveniente de protocolos superiores e a transmissão do seu último bit; como esse tempo inclui o atraso de acesso, esta é uma medida interessante apenas quando o anel não está saturado; quando a carga oferecida está próxima ou acima da capacidade do anel, o tempo de resposta aproxima-se de infinito e não oferece nenhuma informação; neste último caso o *atraso de acesso* é uma medida mais significativa [segundos];
- . *Atraso de QUADRO* ("frame delay"): tempo entre a "geração" do QUADRO na estação para ser transmitido e a

sua chegada na estação destino; esta medida inclui: atraso de acesso, tempo de transmissão do QUADRO e atraso inserido pela propagação do QUADRO no meio entre a estação origem e a estação destino [segundos];

- . *Equidade*: capacidade das estações terem a oportunidade de acesso à transmissão de forma equilibrada;
- . *Comprimento de filas*: esta análise visa identificar possíveis pontos de congestionamento do sistema;
- . *Tempo de rotação da Ficha*: é uma medida específica para protocolos que utilizam o controle temporizado de rotação da ficha (*trt - timed token rotation*).

No estudo de desempenho de uma rede pode-se utilizar ferramentas de simulação em duas etapas: no projeto inicial e durante o seu funcionamento. Para um projeto inicial de implantação de uma rede FDDI, todos os parâmetros de simulação podem ser variados, respeitando os seus limites, para a obtenção de uma infinidade de medidas de desempenho de interesse. Na análise de desempenho durante o funcionamento de uma rede já implantada, apenas variáveis como *ttrt* e *carga oferecida* ao anel, síncrona e assíncrona, podem ser modificadas. Contudo, nestas duas etapas, é importante discriminar aqueles parâmetros que efetivamente influenciam no desempenho do sistema, pois uma escolha criteriosa de variáveis pode ser necessária para a obtenção dos resultados desejados. Na seção a seguir serão

apresentados alguns trabalhos existentes na literatura que permitem demonstrar o grau de influência de variáveis típicas no comportamento de um anel FDDI.

4.3 OUTROS ESTUDOS DE DESEMPENHO DE REDES

Vários trabalhos foram desenvolvidos com o objetivo de avaliar o desempenho do protocolo de acesso ao meio especificado pelo padrão FDDI. Em [ULM,1982] é efetuada uma análise do uso do protocolo de ficha temporizada como eficiente mecanismo de controle de acesso ao meio. A utilização de uma rede FDDI em uma Estação Espacial desenvolvida pela NASA, gerou diversos estudos, avaliando analiticamente e através de simulações, o comportamento do tráfego síncrono/assíncrono num anel FDDI sob condições variadas [JOHNSON,1986b,1987,1988a,1988b,1989] e [SEVCIK,1987]. Resultados importantes do comportamento do protocolo MAC FDDI sob tráfego assíncrono com uma só prioridade e com múltiplas prioridades, obtidos via simulação e/ou ferramentas analíticas, podem ser encontrados em [DYKEMAN,1987,1988] e [TANGEMANN,1991]. A parametrização eficiente de redes FDDI operando apenas com tráfego assíncrono com uma só prioridade, é objeto de análise em [JAIN,1991] e [WELZEL,1990]. Além destes estudos específicos que serão abordados mais detalhadamente nesta seção, análises comparativas entre o protocolo MAC FDDI e outros protocolos de acesso ao meio podem ser encontradas também em [RODRIGUES,1990], [GOYAL,1987] e [MARSAN,1990]. A seguir, serão apresentados os principais resultados dos

estudos feitos, através da simulação e de ferramentas analíticas, sobre o protocolo de ficha temporizada (*trt*) e sobre o comportamento do protocolo FDDI sob diversos contextos

Em um estudo analítico pioneiro sobre o protocolo de ficha temporizada para controle de acesso ao meio em uma rede em anel [ULM,1982], obtém-se uma fórmula para o cálculo da *utilização* do anel, que é verificada em vários trabalhos posteriores. Neste trabalho pioneiro são ressaltadas características importantes do protocolo de ficha temporizada: a implementação simples e a possibilidade de se integrar em um anel diferentes classes de serviço.

Não ocorrendo falhas mecânicas no anel, o protocolo de ficha temporizada do FDDI deve apresentar as seguintes propriedades [ISO/FDDI,1989b]: (a) operar com um tempo médio de rotação da ficha igual ao *ttrt* e (b) o tempo máximo de rotação da ficha ser igual a 2 vezes o *ttrt*. Mostra-se, intuitivamente [ULM,1982], que as propriedades acima são atendidas pelo protocolo de ficha temporizada e, prova-se formalmente a propriedade (a) [SEVCIK,1987] e a propriedade (b) [JOHNSON,1987].

O *ttrt* padrão FDDI, definido pelas estações durante a inicialização do anel, pode ser muito restritivo, implicando em ineficiência do uso da banda disponível do anel. O máximo *ttrt* possível ($ttrt_{\text{ótimo}}$), de forma a preservar a desejada frequência de acesso do tráfego síncrono e aumentar a eficiência do uso do anel, é obtido

analiticamente em [JOHNSON,1988a]. Via simulação, em [JOHNSON,1988a], é mostrado que o valor da eficiência obtido com o t_{trt} pode ainda ser restritivo e que podem ser obtidos resultados mais eficientes ao utilizar-se o t_{trt} igual ao tempo de acesso síncrono desejado ao anel.

O protocolo de acesso ao meio especificado pelo padrão FDDI deve oferecer tempos de resposta pequenos e atraso no acesso máximo limitado para as funções em tempo real, enquanto assegura eventual acesso ao anel para o tráfego de natureza não crítica. Mostra-se, analiticamente e por simulação em [JOHNSON,1986b e 1988b] que o protocolo MAC FDDI, provê a todas estações do anel igual acesso ao meio para transmissão dos quadros assíncronos, em condições de tráfego homogeneamente distribuído. Contudo, em contexto específico, com tamanhos de QUADROS distintos por estações, mostra-se, [JOHNSON,1988b], que pode existir um desbalanceamento na oportunidade de acesso assíncrono ao anel. Problemas similares de desequilíbrio na oportunidade de acesso ao meio com protocolo MAC FDDI, envolvendo múltiplas prioridades de tráfego assíncrono, são detectados em [DYKEMAN,1987 e 1988]. Nestes trabalhos, mostra-se, por exemplo, que é possível que duas estações com a mesma prioridade e a mesma carga tenham diferentes vazões, dependendo de sua localização no anel.

Um estudo detalhado do comportamento do tráfego assíncrono no anel FDDI, através de ferramentas analíticas e de simulações, pode ser encontrado em [DYKEMAN,1987 e 1988]. Aqui, desenvolve-se uma fórmula geral para o cálculo

da *vazão* máxima no anel, e obtém-se resultados que tendem para os resultados pioneiros de [ULM,1982]. Em outro estudo significativo, medidas padrões de desempenho, tais como *atraso do quadro* médio, *utilização* do anel, *comprimento das filas* de transmissão e *tempo de rotação da ficha* médio no anel, foram realizadas considerando-se uma configuração particular de anel FDDI [JOHNSON,1988b].

Em uma rede FDDI, operando apenas com tráfego assíncrono de uma única prioridade, o *ttrt*, pela sua flexibilidade, torna-se um parâmetro difícil de ser determinado. Contextos de simulação específicos que visam oferecer critérios para a definição do valor do *ttrt*, são desenvolvidos em [JAIN,1991] e [WELZEL,1990].

O comportamento do protocolo MAC FDDI, sob tráfego assíncrono operando com múltiplas prioridades, é estudado em [DYKEMAN,1987 e 1988]. Neste estudo em particular, desenvolve-se um algoritmo que permite obter a *vazão* máxima total quando múltiplos níveis de prioridade assíncrona são usados, e a *vazão* para cada um dos níveis de prioridade, quando todas as estações ativas têm quadros para serem transmitidos continuamente. Um resultado particular desse estudo [DYKEMAN,1988] consiste no estabelecimento de critérios na atribuição de valores limites de transmissão para as variáveis *tht* do sistema de prioridades assíncronas, de tal maneira que sejam obtidas as *vazões* desejadas num sistema FDDI.

Um estudo de caso prático de utilização de uma rede FDDI em uma Estação Espacial desenvolvida pela NASA é

simulado por [JOHNSON,1989]. Neste estudo é feita uma análise dos diversos tipos de tráfego, sendo o tráfego de gerenciamento da Estação Espacial avaliado conjuntamente com o seu tráfego normal e o tráfego de emergência. Os diferentes níveis de prioridade de tráfego assíncrono, juntamente com o tráfego síncrono, são mapeados nos serviços de gerenciamento normal e de emergência da estação espacial.

Nos estudos descritos acima, a ferramenta de simulação foi amplamente utilizada, não só para validar resultados analíticos sobre desempenho de redes FDDI, como também para embasar definições mais eficientes de seus parâmetros. Por exemplo, foram utilizados: um simulador desenvolvido pela NASA chamado LANES ("*Local Area Network Simulator*") [JOHNSON,1986b,1987,1988a,1988b e 1989] e [SEVCIK,1987]; simuladores desenvolvidos especificamente para o FDDI [DYKEMAN,1987 e 1988], [JAIN,1991] e [TANGEMANN,1991] e, também, uma ferramenta de simulação de redes de filas de propósito geral denominada QNAP2 [WELZEL,1990].

4.4 RESULTADOS OBTIDOS COM O SIMULADOR FDDI/RESQ2

Os resultados apresentados nesta seção referem-se às variáveis mais significativas para o estudo de desempenho, principalmente, de uma rede FDDI já implantada. Todavia, a ferramenta de simulação implementada também permite a mudança de parâmetros que dizem respeito ao projeto de instalação de uma nova rede.

A modelagem FDDI/RESQ2 desenvolvida permite diversas aplicações: seja em configurações mistas, com estações síncronas e assíncronas, ou apenas com estações assíncronas, caso não haja tráfego com características críticas no tempo. Como ambiente de simulação usou-se o computador IBM 4381, modelo P20, com 16 Mbytes de memória RAM, do Campus II da Universidade Federal da Paraíba. As simulações realizadas basearam-se em cenários já investigados em outros trabalhos [JOHNSON,1988a,b] e [JAIN,1991], de modo a facilitar a validação do modelo do protocolo MAC FDDI em RESQ2.

As configurações estudadas neste trabalho operam com estações do tipo DAS com apenas uma MAC implementada e sem concentradores integrados ao anel. As simulações executadas a partir destes contextos utilizaram o método de estimação de intervalo de confiança SPECTRAL [SOARES,1991] oferecido pela própria ferramenta RESQ2 [SAUER,1982b], com nível de confiança de 90%. No Anexo III, encontra-se um detalhamento dos procedimentos operacionais para se efetuar simulação de uma configuração desejada e a forma de se obter uma síntese das estatísticas, a partir das listagens oferecidas pela ferramenta. As medidas de desempenho típicas selecionadas para validação são as seguintes: *tempo de resposta*, *atraso de quadro médio* e *atraso de acesso*.

Os resultados apresentados a seguir foram obtidos com o simulador do protocolo de acesso ao meio FDDI, implementado em RESQ2, sendo executado em condições de operação rotineira da máquina.

A partir de um contexto totalmente assíncrono [JOHNSON,1988b], sintetizado na Tabela 4.1 e detalhado no Anexo II.2, foram obtidos com o simulador FDDI/RESQ2 os resultados de desempenho mostrados nas Figuras 4.1 e 4.2.

Num. de Estações no anel	: 20 assíncronas
ttrt	: 40 ms
Tamanho do Quadro	: 32.330 bits

Tabela 4.1 - Parâmetros de simulação do exemplo 1.

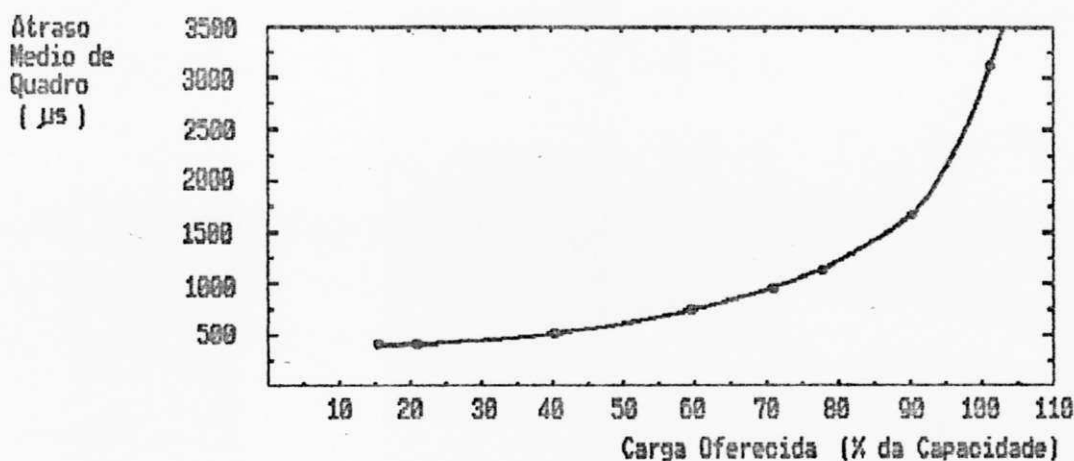


Figura 4.1 - Atraso de Quadro Médio x Carga Oferecida

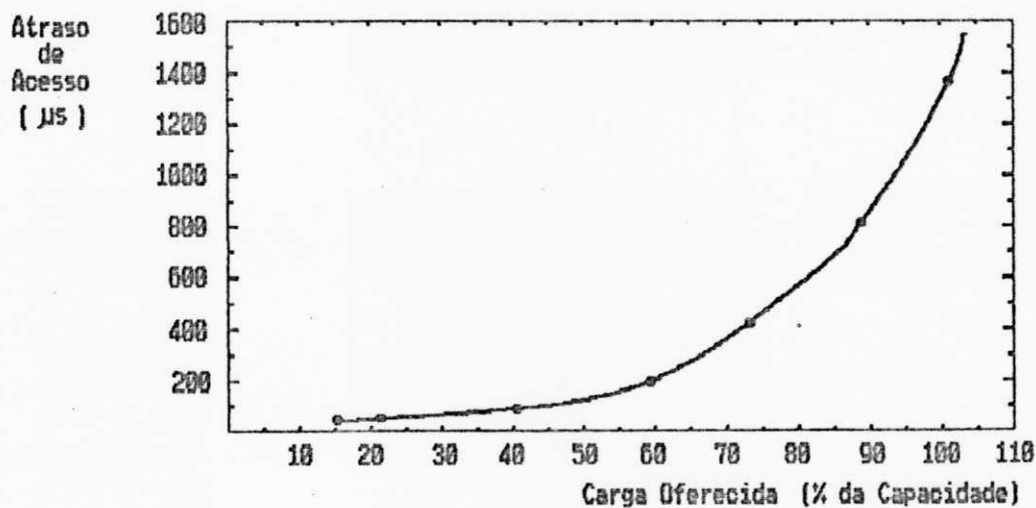


Figura 4.2 - Atraso de Acesso x Carga Oferecida

Como numa configuração completamente assíncrona não ocorrem tráfegos de natureza crítica, observa-se que, mesmo a 98% de *carga oferecida*, o *tempo de resposta* médio não chegou a alcançar metade do *ttrt*. Da mesma forma, os *tempos de acesso*, mesmo em cargas superiores a 95%, estiveram bem abaixo do *ttrt*.

A utilização do canal é também uma medida de desempenho de interesse na análise do comportamento do tráfego em um anel FDDI. Na Figura 4.3 é apresentado um gráfico, onde observa-se que a utilização do anel FDDI cresce linearmente com a carga oferecida até o anel ficar saturado. Este resultado concorda com os resultados analíticos obtidos por [ULM,1982].

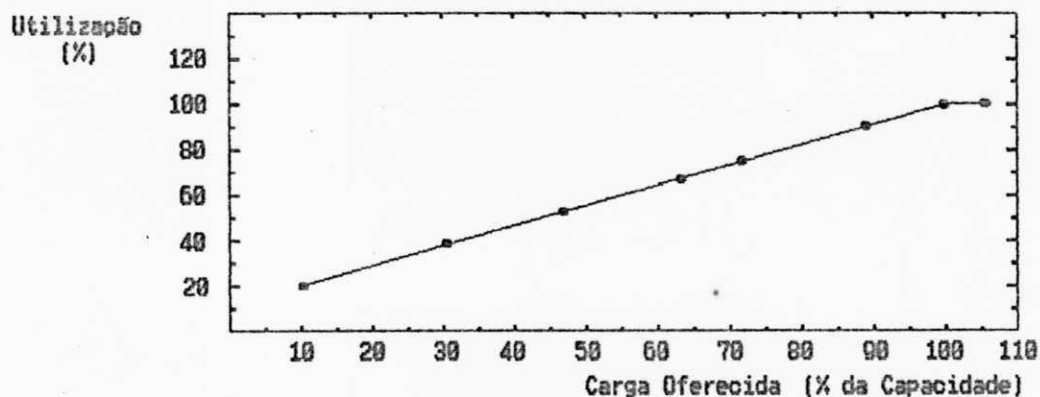


Figura 4.3 - Utilização do anel x Carga Oferecida

Através do processo de simulação, verificou-se que o número de quadros médio sendo processado pelo anel, por rotação de ficha, estava muito aquém da sua capacidade, mesmo quando a *carga oferecida* chegava a patamares de 120%. Foram feitas algumas incursões a *cargas oferecidas* em torno de 1,6 Gbits/seg e observou-se que o tempo médio de rotação

da ficha ainda permanecia nos 38 ms. Por outro lado, nestas condições, o *atraso de quadro* médio ficou em torno dos 60 ms, e o atraso máximo de quadro, entre 420 e 450 ms.

Dentro do cenário assíncrono [JAIN,1991] sintetizado na Tabela 4.2 e detalhado no Anexo II.3, as simulações com a ferramenta FDDI/RESQ2 a 90% de *carga oferecida*, obtiveram um tempo de resposta médio que permaneceu em torno de 1 ms, independente do *ttrt* aplicado.

Num. de Estações no anel	: 20 assíncronas
ttrt	: 4; 8; 16; 20; 165 ms
Tamanho do Quadro	: 36.000 bits

Tabela 4.2 - Parâmetros de simulação do exemplo 2.

Em aplicações críticas no tempo, que exigem tempo de resposta garantido (ou seja, com estações gerando tráfego síncrono), tem-se o *ttrt* definido pelo padrão [ISO/FDDI,1989b]. Contudo, podem ser feitas variações desta definição, visando uma otimização da banda passante disponível, conforme foi visto anteriormente (seção 4.3). Por exemplo, trabalhando-se com um *ttrt* igual ao tempo de acesso síncrono desejado por uma estação, numa configuração [JOHNSON,1988a] conforme sintetizado na Tabela 4.3 e detalhado no Anexo II.4, com as seguintes condições de carga oferecida:

Num. de Estações no anel	: 16; 12 (sinc.) e 4 (assinc.)
Freq. de acesso síncrono	: 8 ms
ttrt	: 8 ms
Tamanho do Quadro	: 32.330 bits

Tabela 4.3 - Parâmetros de simulação do exemplo 3.

- . carga nível 1 (6 estações síncronas ativas gerando 18 quadros por rotação da ficha),
- . carga nível 2 (todas estações ativas gerando 24 quadros por rotação de ficha) e
- . carga nível 3 (todas estações ativas gerando 28 quadros por rotação de ficha)

obteve-se os seguintes resultados via simulação FDDI-RESQ2:

- . com o anel sendo submetido à carga nível 1, a *carga aplicada* fica em torno de 75% e, tanto o tempo médio, como o tempo máximo de rotação de ficha, resultam bem inferiores que o *ttrt*, assim como o *atraso de quadros* síncronos médio; apesar de não haver uma correlação direta entre o *atraso médio* dos quadros síncronos e o tempo entre dois acessos consecutivos, desde que o *atraso de acesso* síncrono não exceda o *ttrt*, o anel estará operando como desejado;
- . sob a carga nível 2, o anel operando à 97% de *carga aplicada*, em média, a frequência de acesso síncrono oferecida pelo anel às estações permanece abaixo de 3 ms e o *atraso dos quadros* médio idem;
- . sob a carga nível 3, 113% de *carga aplicada*, o atraso médio dos quadros síncronos permanece abaixo de 5.000 μ s; o atraso de acesso médio por estação para carga de

113% pode ser melhor visualizado no gráfico da Figura 4.4.

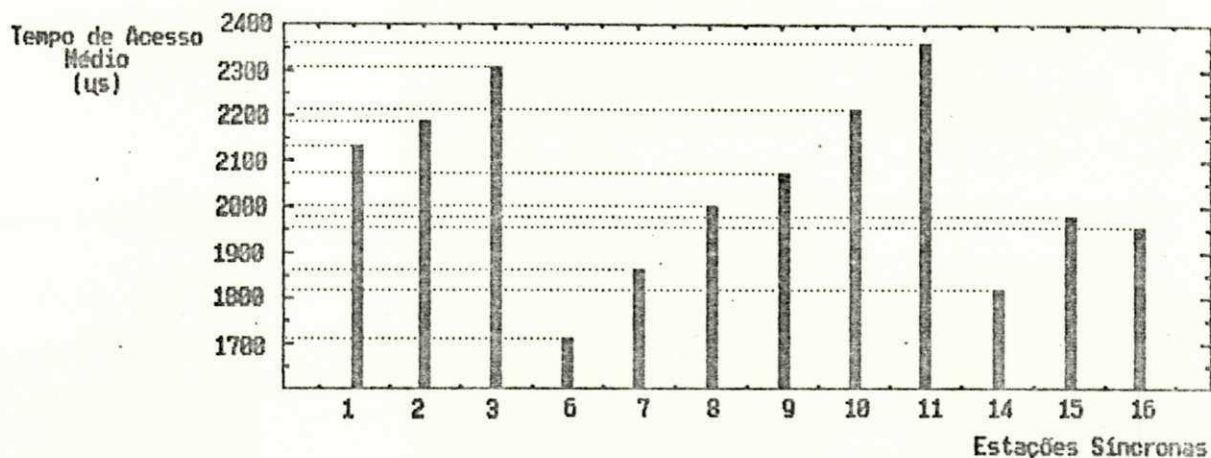


Figura 4.4 - Atraso de Acesso por estação para carga de 113% do exemplo 3.

A definição do t_{trt} igual ao tempo de acesso síncrono desejado tem implicações de perdas de quadros das estações síncronas, na configuração mista. A depender do grau de perdas compatível com a aplicação desejada, pode-se efetuar esta definição ou até definições mais ousadas do t_{trt} .

Pode-se observar a partir dos resultados obtidos na Figura 4.4 que, sob condições de tráfego com características de geração distintas, uma desigualdade de acesso ao anel. Assim, sugere-se que, estudos de desempenho detalhados levando em conta estas características de configuração do anel sejam executadas posteriormente, para uma melhor avaliação das consequências da integração de tipos de tráfegos diferentes em um mesmo anel. Na Figura 4.5, apresenta-se um gráfico com as perdas percentuais por

estação, referente à carga oferecida de 113% (carga nível 3 do Exemplo 3) ao anel.

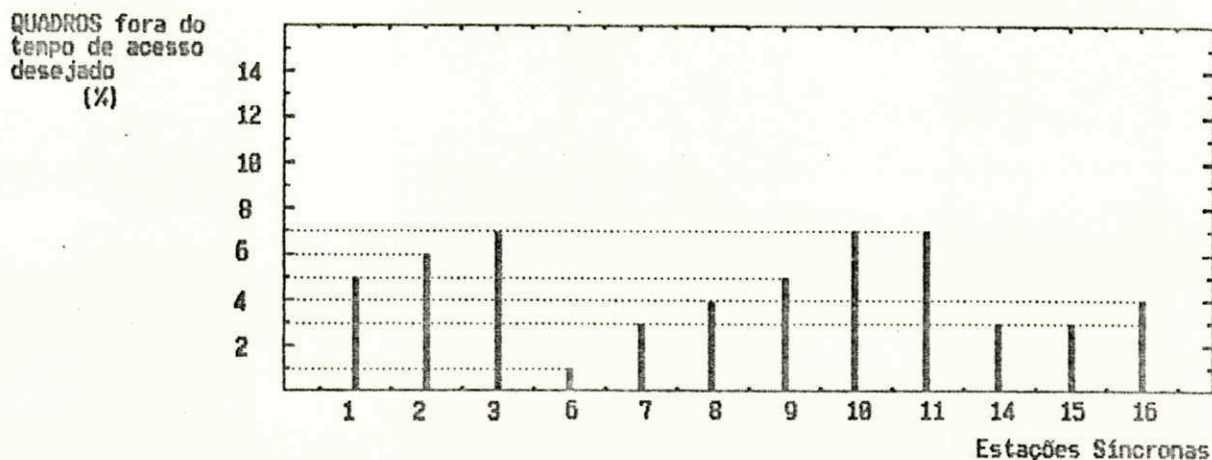


Figura 4.5 - Perda percentual de quadros por estação para carga de 113% do exemplo 3.

A utilização do simulador FDDI/RESQ2 desenvolvido em estudos mais detalhados, requer um ambiente computacional com capacidade de processamento maior que a disponível. A partir da análise dos intervalos de confiança do exemplo 3, por exemplo, pode ser observado que para um nível de confiança de 90%, os intervalos de confiança para simulação de duração de uma hora estabelecem-se em torno dos 30%, enquanto que ao executar-se durante 10 horas seguidas, os intervalos baixam para ordem de 7 a 10%. Os estudos de equidade de acesso ao meio e/ou estudos que envolvessem um crescimento do anel com um número de estações ativas muito superior ao apresentado, ficaram prejudicadas em razão dos recursos computacionais disponíveis.

Os resultados obtidos nos gráficos apresentados neste Capítulo, estão de acordo com os resultados encontrados na literatura indicada para validação. A obtenção de erros percentuais com relação aos resultados encontrados nas referências, não foi efetuada pois, estes erros, podem ser mascarados por duas imprecisões: grau de confiança usado neste trabalho diferente do utilizado pelas referências e, a obtenção de resultados pontuais constantes na literatura, precisa ser feita através de levantamento visual dos gráficos apresentados.

CAPITULO 5

CONCLUSÕES

O protocolo de controle de acesso ao meio FDDI pode ser modelado através do pacote de simulação de redes de filas de propósito geral RESQ2. O simulador desenvolvido permite inserir de forma simples modificações em um grande espectro de parâmetros de modelagem e obtenção de medidas de desempenho.

A utilização da ferramenta de simulação de redes de filas de propósito geral RESQ2 para simulação do protocolo FDDI, permitiu a obtenção de resultados de forma rápida e com menos envolvimento na ferramenta, comparando-se com o desenvolvimento de um simulador específico. Partindo-se apenas do conhecimento superficial da linguagem, em três meses efetuou-se a modelagem do sistema, a escrita e digitação do programa, os testes de conformidade e a otimização do modelo. Caso a utilização da ferramenta RESQ2 já seja dominada pelo usuário, estes tempos ainda podem diminuir sensivelmente. A ferramenta RESQ2 oferece resultados de desempenho específicos e com larga faixa de atuação, porém, necessita-se de uma macro linguagem, atualmente inexistente, para obtenção de resultados sintetizados (o RESQ2 permite o uso do PL/I para obtenção de gráficos, contudo a sua utilização é muito restritiva). Passada a dificuldade inicial de se raciocinar dentro da filosofia RESQ2, a abordagem gráfica oferecida para modelagem permite modificações do modelo em estudo com relativa facilidade. O programa final modelando o FDDI em RESQ2 permite a usuários que tenham interesse de pesquisar o funcionamento do protocolo ou desenvolver projetos de

implantação do FDDI, modificar variáveis de parametrização do anel e realizar estudos de interesse.

Outra observação importante a ser destacada é o comportamento da máquina IBM utilizada sob simulação da ferramenta RESQ2. Tanto quando a máquina estava em operação normal (vários usuários trabalhando) quanto, existiam apenas processos de simulação RESQ2 rodando sozinhos, os tempos de simulação foram similares. O uso da máquina sob operação normal implica em perdas de desempenho perceptíveis, apenas quando várias horas de simulação estão envolvidas no teste. Possivelmente um ganho de desempenho será conseguido com uma máquina com maior capacidade de processamento (MIPS) e de memória.

A análise dos resultados obtidos o FDDI é sensível à variação de uma série de parâmetros, a nível de projeto inicial ou já em funcionamento. As variações físicas no anel, como o aumento da extensão da fibra ou do número de estações físicas inativas conectadas ao anel, criam um impacto em termos de atenuação que se reflete apenas no cálculo do ttrt mínimo possível de ser calculado para aquela configuração específica [ISO/FDDI,1989b] de modo que a vazão do anel não seja muito baixa [DYKEMAN,1988]. Tanto nas aplicações com geração exclusiva de quadros assíncronos pelas estações, onde o ttrt é definido pelo usuário quanto nas aplicações mistas, onde o ttrt é estabelecido pelo padrão, a simulação mostra ser uma ferramenta bastante importante no auxílio da parametrização do ttrt. Com uma parametrização adequada é possível prover

um melhor aproveitamento da banda passante disponível e assegurar às medidas de desempenho desejadas. Como as ferramentas de simulação utilizadas nos outros trabalhos existentes não estão disponíveis de forma evidente para a maioria dos usuários, a modelagem do FDDI desenvolvida neste trabalho com uma ferramenta comercial permite ampliar as possibilidades de avaliação de desempenho de configurações e aplicações específicas de redes FDDI.

Este estudo inicial do protocolo FDDI com a ferramenta RESQ2 pode ser expandido em algumas vertentes como, por exemplo:

- . um detalhamento do comportamento dos métodos de estimação de intervalo de confiança embutidos no próprio RESQ2, avaliando melhor o comportamento dos intervalos de confiança em diversas configurações, conjuntamente com os tempos de simulação e os critérios de definição dos intervalos de amostragem das medidas em foco;
- . estudo da equidade de acesso ao meio das estações, conjuntamente com uma ampliação do número de estações ativas conectadas ao anel;
- . inserção no contexto do tráfego assíncrono com múltiplas prioridades, assim como a possibilidade de se conectar concentradores de estações ao anel FDDI;
- . enfim, pode-se partir para a modelagem do FDDI-II, incorporando o tráfego de voz por circuitos na rede operando com pacotes, permitindo assim, o estudo das redes integrando serviços.

**REFERENCIAS
BIBLIOGRAFICAS**

- [ANSI/FDDI,1987]
ANSI - "FDDI Token Ring Media Access Control (MAC)",
ANSI X3-139-1987.
- [ANSI/FDDI,1988]
ANSI - "FDDI Physical Layer Protocol (PHY)", ANSI X3-
148-1988.
- [ANSI/FDDI,1990a]
ANSI - "FDDI Station Management (SMT)", ASC X3T9.5,
Draft Proposed, Rev. 6.2, maio 1990.
- [ANSI/FDDI,1990b]
ANSI - "FDDI Enhanced Physical Layer Protocol (PHY-2)",
ASC X3T9.5, Draft Proposed, Rev. 3.1, maio 1990.
- [ANSI/FDDI,1990c]
ANSI - "FDDI Physical Layer Medium Dependent (PMD)",
ANSI X3-166-1990.
- [ANSI/FDDI,1990d]
ANSI - "SONET Physical Layer Mapping (SPM)", ASC
X3T9.5, Draft Proposed, Rev. 3.1, maio 1990.
- [ANSI/FDDI,1990e]
ANSI - "FDDI Hybrid Ring Control (HRC) ", ASC X3T9.5,
Draft Proposed, Rev. 6, maio 1990.
- [ANSI/FDDI,1990f]
ANSI - "FDDI Single Mode Fiber PMD (SMF-PMD)", ASC
X3T9.5, Draft Proposed, Rev. 4.2, maio 1990.
- [ANSI/FDDI,1990g]
ANSI - "FDDI Enhanced Media Access Control (MAC-2)",
ASC X3T9.5, Draft Proposed, Rev. 3.2, jun. 1990.
- [ARMBRUSTER,1987]
ARMBRUSTER, H. e ARNDT, G. - "Broadband Communica-
tion and Realization with Broadband ISDN". *IEEE Comm.
Magazine*, v. 25, n. 11, nov. 1987, p. 8-19.
- [BALLART,1989]
BALLART, R. e CHING, Y. C.- "SONET : Now It's the
Standard Optical Network". *IEEE Communication Magazine*,
v.29, maio 1989, p. 8-15.
- [BAPTISTELLA,1990]
BAPTISTELLA, L. F. B. e LOBO, M. P. - RDSI - Rede Digi-
tal de Servicos Integrados. McGraw-Hill, 1990.
- [BORGONOVO,1987]
BORGONOVO, F. - "ExpressMAN : exploiting traffic lo-
cality in ExpressNET". *IEEE Journal on Selected Areas
in*

- Communications*, v. SAC-5, n. 9, dez. 1987, p. 1436-1443.
- [BRANDWEIN,1990]
BRANDWEIN, R. e outros - "The IEEE 802.6 Physical Layer Convergence Procedures". *IEEE LCS Magazine*, n. 5, maio 1990, p. 26-34.
- [BRITO,1991]
BRITO, S. F. e GIOZZA, W. F. - "Redes Metropolitanas (MAN's)". In: *Anais do 9º SBRC*. Florianópolis, SC, maio 1991, p. 185-199.
- [BURR,1986]
BURR, W. E. - "The FDDI optical data link". *IEEE Comm. Magazine*, v. 24, n. 5, maio 1986, p. 18-23.
- [CODEN,1990]
CODEN, M. H. e outros - "Modular FDDI bridge and concentrator". In: *Proceedings of FDDI, Campus-Wide and Metropolitan Area Networks*, set. 1990, San Jose, California, p. 22-37.
- [DIETERICH,1986]
DIETERICH, A. L. B. et all - "Análise Qualitativa de um Anel de Token". In : *Anais do VI Cong. da Sociedade Brasileira de Comput.*. Recife, PE, jul. 1986, p. 27-37.
- [DYKEMAN,1987]
DYKEMAN, D. e BUX, W. - "An investigation of the FDDI media-access control protocol". In: *Proc. 5th European Fibre Opt. Commun. & Local Area Networks Conference*. Basel, Switzerland, jun. 1987, p. 229-236.
- [DYKEMAN,1988]
DYKEMAN, D. e BUX, W. - "Analysis on tuning of the FDDI Media Access Control Protocol". *IEEE Journal on Selected Areas in Communications*, v. 6, n. 6, jul. 1988, p. 997-1010.
- [GALDINO,1991]
GALDINO, J. F. e GIOZZA, W. F. - "Redes Locais de Altíssima Velocidade". In: *Anais do 9º SBT*. São Paulo, SP, set. 1991.
- [GIOZZA,1986]
GIOZZA, W. F.; ARAUJO, J. F. M.; MOURA, J. A. B. e SAUVÉ, J. P. - Redes Locais de Computadores - Tecnologia e Aplicações, McGraw-Hill, São Paulo, 1986.
- [GIOZZA,1991]
GIOZZA, W. F.; CONFORTI, E. e WALDMAN, H. - Fibras ópticas: Tecnologia e Projeto de Sistemas, McGraw-Hill, São Paulo, 1991.

[GOYAL,1987]

GOYAL, A. e DIAS, D. - "Performance of priority protocols on high-speed token ring networks". In: *Proc. 3rd International Conference Data Communications Systems and their Performance*. Rio de Janeiro, Brasil, jun. 1987, p. 25-34.

[HAMSTRA,1988]

HAMSTRA, J.. - "FDDI design tradeoffs". In: *13th Local Comp. Networks Conf.*. Out. 1988.

[HANSON,1990]

HANSON, D. C. - "Progress in Fiber Optic LAN and MAN Standards". *IEEE LCS Magazine*, n. 5, maio 1990, p. 17-25.

[HARTMANN,1989]

HARTMANN, M. J. - "FDDI Challenges interface designers". *LASER FOCUS WORLD*, maio 1989, p. 169-170.

[HEMRICK,1988]

HEMRICK, C. F. e outros - "Switched Multi-Megabit Data Service and early availability via MAN technology". *IEEE Communications Magazine*, v. 26, n. 4, abr. 1988, p. 9-14.

[HEYWOOD,1989]

HEYWOOD, P. - "The rise of METROPOLITAN AREA NETWORKS". *Data Communications International*, v. 18, n. 16, dez. 1989, p. 70-78.

[ISO,1984]

ISO - "Information Processing Systems - Open Systems Interconnection - Basic Reference Model", ISO 7498, 1984.

[ISO/FDDI,1989a]

ISO - "FDDI Physical Layer Protocol (PHY)", ISO 9314-1, 1989.

[ISO/FDDI,1989b]

ISO - "FDDI Token Ring Media Access Control (MAC)", ISO 9314-2, 1989.

[ISO/FDDI,1990a]

ISO - "FDDI Physical Layer Medium Dependent (PMD)", ISO 9314-3, 1990.

[ISO/FDDI,1990b]

ISO - "FDDI Single Mode Fiber PMD (SMF-PMD)", ASC X3T9.5, DP 9314-4, 1990.

[ISO/FDDI,1990c]

ISO - "FDDI Hybrid Ring Control (HRC) ", DP 9314-5, 1990.

[JAIN,1991]

JAIN, R. - "Performance Analysis of FDDI Token Ring Networks: Effect of Parameters and Guidelines for Setting TTRT". *IEEE LTS Magazine*, maio 1991, p. 16-22.

[JOHNSON,1986a]

JOHNSON, M. J. - "Reliability mechanisms of the FDDI high bandwidth token ring protocol". *Computer Networks and ISDN Systems*, v. 11, n. 2, fev. 1986, p. 121-131.

[JOHNSON,1986b]

JOHNSON, M. J. - "Fairness of channel access for non-time-critical traffic using the FDDI token ring protocol". In: *Proc. Seminar Real-Time Local Area Networks*. Bandol, France, abr. 1986, p. 145-157.

[JOHNSON,1987]

JOHNSON, M. J. - "Proof that timing requirements of the FDDI token ring protocol are satisfied". *IEEE Transactions on Communications*, v. COM-35, 1987, p. 620-625.

[JOHNSON,1988a]

JOHNSON, M. J. - "Analysis of FDDI synchronous traffic delays". In: *Proceedings of Systems Design and Networks Conference: Putting Local Area Networks to Work*, Santa Clara, CA, abr. 1988, p. 65-72.

[JOHNSON,1988b]

JOHNSON, M.J. - "Performance analysis of FDDI". In : *EFOC/LAN'88*, jun. 1988.

[JOHNSON,1989]

JOHNSON, M.J. - "Performance Issues in Management of the Space Station Information System". JOHNSON, M. J.. North Holland, 1989, p. 423-434.

[KLEINROCK,1975]

KLEINROCK, L. - Queueing System, Volume 1 e 2: Theory, John Wiley & Sons, 1975.

[MANTELMANN,1989]

MANTELMAN, L. - "Incompatible Bridges stymie use of FDDI as a LAN BACKBONE". *Data Comm.*, set. 1989, p. 39-44.

[MARSAN,1990]

MARSAN, M. A. e NERI, F. - "Performance Comparison of Four Mac Protocols for Integrated MAN's". International Telecommunications Symposium, ITS, 90, Rio de Janeiro, Setembro, 1990.

[MARTINI,1988]

MARTINI, P. e outros - "File Transfer in High-Speed Token Ring Networks : Performance Evaluation by

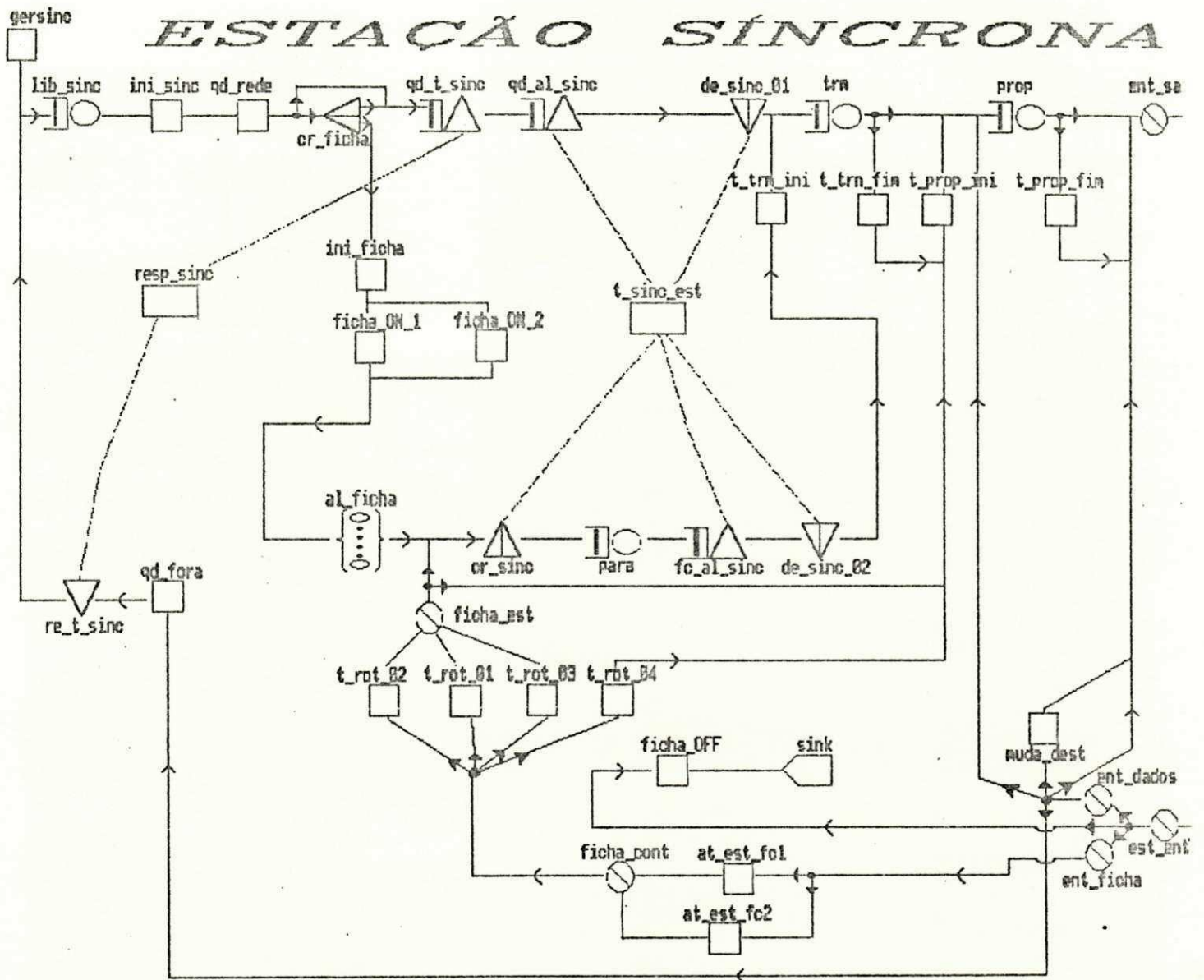
- approximate analysis and simulation". *IEEE Journal on Selected Areas in Communications*, v. 6, 987-996.
- [MCCOOL,1988]
MCCOOL, J. F. - "FDDI : Getting to know the inside of the ring". *Data Communications*, mar. 1988, p. 185-192.
- [MCQUILLAN,1990]
MCQUILLAN, J.M. - "Broadband Networks. The End of Distance?". *Data Comm. International*, jun. 1990, p. 62-74
- [MINZER,1989]
MINZER, S. E. - "Broadband ISDN and Asynchronous Transfer Mode (ATM)". *IEEE Communications Magazine*, v. 27, n. 9, set. 1989, p. 17-24.
- [MOLLENAUER,1988]
MOLLENAUER, J. F. - "Standards for METROPOLITAN AREA NETWORKS". *IEEE Comm. Mag.*, v. 26, n. 4, abr. 1988, p. 15-19.
- [MOLLENAUER,1989]
MOLLENAUER J. F. - "METROPOLITAN AREA NETWORK update : the global LAN is getting closer". *Data Communications International*, v. 18, n. 16, dez. 1989, p. 109-118.
- [MOURA,1986]
MOURA, J. A. B.; SAUVÉ, J. P. GIOZZA: W. F. e ARAUJO, J. F. M. - Protocolos de Alto Nível e Avaliação de Desempenho, McGraw-Hill, 1986.
- [NEWMAN,1988]
NEWMAN, R. M. e outros - "The QPSX MAN". *IEEE Communications Magazine*, v. 26, n. 4, abr. 1988, p. 20-28.
- [OHLHABER,1989]
OHLHABER, R. L. - "What FDDI says about hardware?". *LASER FOCUS WORLD*, maio 1989, p. 165-167.
- [PAPOULIS,1965]
PAPOULIS, A. - Probability, Randon Variables and Stochastic Processes, McGraw-Hill, 1965.
- [RODRIGUES,1990]
RODRIGUES, M. A. - "Evaluating Performance of High-Speed Multiaccess Networks". *IEEE Network Magazine*, v. 4, n. 3, maio 1990, p. 36-41.
- [ROSS,1986]
ROSS, F. E. - "FDDI-A Tutorial". *IEEE Communication Magazine*, v. 24, maio 1986, p. 10-17.

- [ROSS,1987]
ROSS, F. E. - "Rings Are Round For Good". *IEEE Network Magazine*, v. 1, n. 1, jan. 1987, p. 31-38.
- [ROSS,1989]
ROSS, F. E. - "An Overview of FDDI : The Fiber Distributed Data Interface". *IEEE Journal on Selected Area in Communications*, v. 7, n. 7, set. 1989, p. 1043-1051.
- [ROSS,1990]
ROSS, F.E. - "FDDI - A LAN among MAN's". *Computer Communications Review*, v. 20, n. 3, jul. 1990, p. 16-31.
- [RUIZ,1989]
RUIZ, A. e outros - "An Architecture and Implementation for an FDDI MAC Concentrator". IBM Report, 1989.
- [SAUER,1981]
SAUER, C.H. e CHANDY, K.M. - "Computer Systems Performance Modeling", Prentice-Hall,1981.
- [SAUER,1982a]
SAUER, C. H.: MACNAIR, E. A. e KUROSE, J. F. - "The Research Queueing Package Version 2: Introduction and Examples", IBM Research Report RA-138, Yorktown Heights, New York, 1982.
- [SAUER,1982b]
SAUER, C. H.: MACNAIR, E. A. e KUROSE, J. F. - "The Research Queueing Package Version 2: CMS Users Guide", IBM Research Report RA-139, Yorktown Heights, New York, 1982.
- [SAUER,1983]
SAUER, C. H. e MACNAIR, E. A. - "Simulation of Computer Communications Systems", Prentice-Hall, 1983.
- [SAUER,1985]
SAUER, C. H. e MACNAIR, E. A. - "Elements of Pratical Performance Modeling", Prentice-Hall, 1985.
- [SEVCIK,1987]
SEVCIK, K. C. e JOHNSON, M. J. - "Cycle time properties of the FDDI token ring protocol". *IEEE Transactions on Software Engineering*, v. SE-13, mar. 1987, p. 376-385.
- [SHAPIRO,1989]
SHAPIRO, S. F. - "FDDI Moves Closer to Fruiton as Standard Network Backbone". *Computer Design*, ago. 1989, p. 35-38.
- [SOARES,1990]
SOARES, L. F. G. - Modelagem e Simulação Discreta de Sistemas, VII Escola de Computação, São Paulo, 1990.

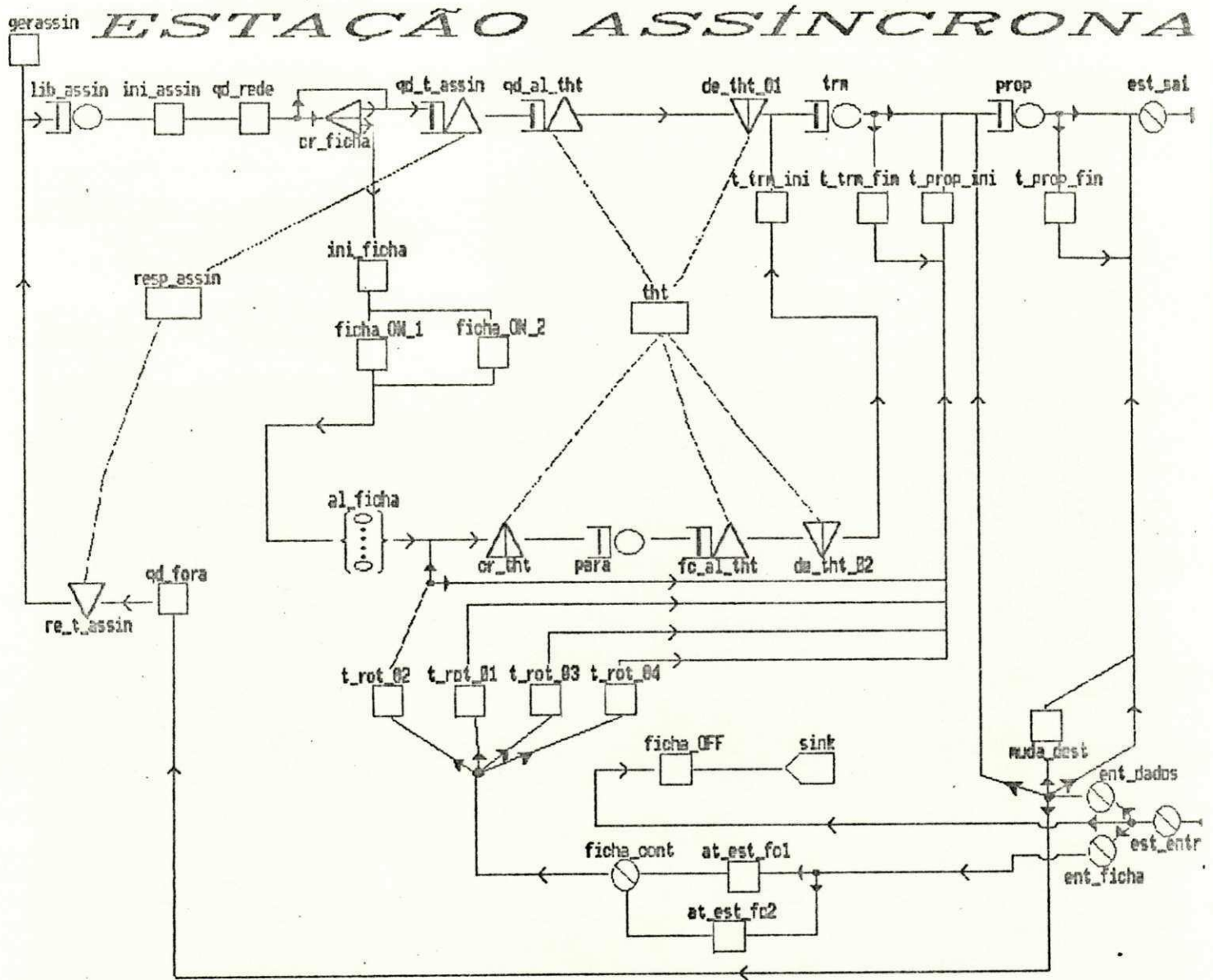
- [STALLINGS,1990]
STALLINGS, W. - "Local Network Performance". *IEEE Communications Magazine*, v. 22, n. 2, fev. 1984, p. 27-36.
- [STEVENS,1990]
STEVENS, R. S. - "FDDI Network Cabling". In: *Proceedings of FDDI, Campus-Wide and Metropolitan Area Networks*, set. 1990, San Jose, California, p. 101-114.
- [STRAUSS,1990]
STRAUSS, P. R. - "IBM committee drive GBIT's LAN's". *Data Communications International*, v. 19, n. 2, fev. 1990, p. 43.
- [STROHL,1991]
STROHL, M. J. - "High Performance Distributed Computing in FDDI Networks". *IEEE LTS*, maio 1991, p. 11-15.
- [TANGEMANN,1991]
TANGEMANN, M. e SAUER, K. - "Performance Analysis of the Timed Token Protocol of FDDI and FDDI-II". *IEEE Journal on Selected Area in Communications*, v. 9, n. 2, fev. 1991, p. 271-278.
- [TANENBAUM,1981]
TANENBAUM, A. S. - Computer Networks. Prentice-Hall, 2ª edição, 1988.
- [ULM,1982]
ULM, J. M. - "A timed token ring local area network and its performance characteristics". In: *Proc. IEEE 7th Local Comput. Networks Conference*. Mineapolis, MN, out. 1982. p. 50-56.
- [ZAFIROVIC-VUKOTIC,1988]
ZAFIROVIC-VUKOTIC, M. et all - "Performance Analysis of Slotted Ring Protocols in HSLAN's". *IEEE Journal on Selected Areas in Communications*, v. 6, n. 6, jul. 1988, p. 1011-1024.
- [WELZEL,1990]
WELZEL, T. - "Performance Comparison of Services Strategies in High Speed Optical Token Rings". *International Telecommunications Symposium, ITS, 90*, Rio de Janeiro, Setembro,1990.
- [WILSON,1991]
WILSON, R. - "FDDI chips struggle toward the Desktop". *Computer Design*, fev. 1991, p. 61-68.

ANEXOS

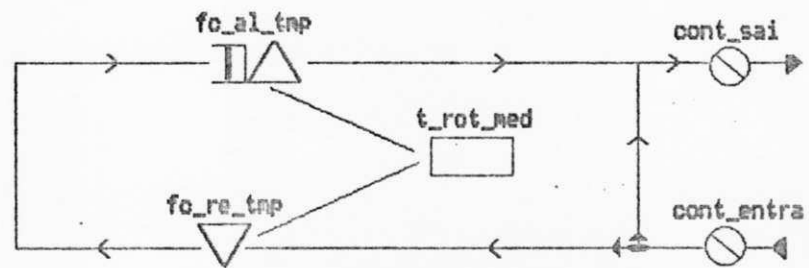
ANEXO I.1 Diagrama da Estação Síncrona



ANEXO I.2 Diagrama da Estação Assíncrona



ANEXO I.3 Diagrama da Estação de Controle da Ficha

CONTROLE DA FICHA

ANEXO II.1 Arquivo de Sub-modelos

/*Descrição da estações do anel FDDI */

```

SUBMODEL: ests_sinc
  NUMERIC PARAMETERS: j /*Indica a estacao atual*/
  CHAIN PARAMETERS: cadeia
  QUEUE: lib_sincq /* Controla liberacao dos qds para o anel*/
    TYPE: fcfs
    CLASS LIST: lib_sinc
      SERVICE TIMES: constant(tint_sinc*tmp_cv(j)) /*(seg/qd)*/
  QUEUE: al_ficha /* Controla liberacao dos qds para o anel*/
    TYPE: is
    CLASS LIST: al_ficha
      SERVICE TIMES: constant(alea_ficha)
  QUEUE: resp_sinc /*Controla o tempo de resposta sincrono*/
    TYPE: passive
    TOKENS: 2147483647 /*infinita*/
    DSPL: fcfs
    ALLOCATE NODE LIST: qd_t_sinc
      NUMBER OF TOKENS TO ALLOCATE: 1
    RELEASE NODE LIST: re_t_sinc
  QUEUE: t_sinc_est /*Controla banda de trm. sincrona*/
    TYPE: passive
    TOKENS: 0
    DSPL: fcfs
    ALLOCATE NODE LIST: qd_al_sinc fc_al_sinc
      NUMBERS OF TOKENS TO ALLOCATE: 1 ta(t_sinc_est)
    DESTROY NODE LIST: de_sinc_01 de_sinc_02
    CREATE NODE LIST: cr_sinc
      NUMBERS OF TOKENS TO CREATE: ceil(sinc_est/t_trm_qd)
  QUEUE: para_q /* Controla parada da fic para liberacao dos qds*/
    TYPE: fcfs
    CLASS LIST: para
      SERVICE TIMES: 0
  QUEUE: trm_q /*Controla atraso de trm. de quadros/fichas*/
    TYPE: fcfs
    CLASS LIST: trm
      SERVICE TIMES: constant(jv(tamanho)/cap_trm) /*(bits/bps)*/
  QUEUE: prop_q /*Controla atraso de prop. e est. quadros/fichas*/
    TYPE: fcfs
    CLASS LIST: prop
      SERVICE TIMES: constant((at_metro*dist_est)+at_estacao)
  SET NODES: gersinc /*Colocacao dos QUADROS sincronos*/
    ASSIGNMENT LIST: jv(tipo)=sinc
  SET NODES: ini_sinc /*Inicializa JV's dos QUADROS sincronos*/
    ASSIGNMENT LIST:
jv(tamanho)=tam_men ++
jv(origem)=j ++
jv(destino)=ceil(uniform(1,(jv(origem)-1),(jv(origem)-1)/(mx_est-1));(j++
v(origem)+1),mx_est,(mx_est-jv(origem))/(mx_est-1))) ++
tmp_cv(j)=1
  SET NODES: qd_rede /*Incrementa quadros rodando no anel*/
    ASSIGNMENT LIST: n_qd_anel=n_qd_anel+1 ++
n_qd_sin=n_qd_sin+1

```


ANEXO II.1 - (continuação)

```

SET NODES: qd_fora /*Decrementa quadros rodando no anel*/
ASSIGNMENT LIST: n_qd_anel=n_qd_anel-1
SET NODES: ini_ficha /*Inicializa JV's da nova FICHA criada*/
ASSIGNMENT LIST:jv(tipo)=ficha ++
jv(tamanho)=tam_fic ++
ficha_anel=1 ++
n_vt_vaz=((CLOCK-fim_trab)/((at_metro*dist_est*mx_est)+(at_estacao*mx_++
est))) ++
fc_n_est=fc_n_est+1 ++
fc_rotac=fc_rotac+n_vt_vaz ++
fc_t_rotmd=CLOCK/fc_rotac
SET NODES: ficha_ON_1 /*Atualiza inf. rel. a criacao FICHA */
ASSIGNMENT LIST: /* n_vt_vaz < 1 */ ++
alea_ficha=(CLOCK-fim_trab)
SET NODES: ficha_ON_2 /*Atualiza inf. rel. a criacao FICHA */
ASSIGNMENT LIST: /* n_vt_vaz >= 1 */ ++
alea_ficha=uniform(0,((at_metro*dist_est*mx_est)+(at_estacao*(mx_est-1++
))),1)
++
trt(*)=constant((at_metro*dist_est*mx_est)+(at_estacao*(mx_est-1)))
SET NODES: ficha_OFF /*Atualiza inf. rel. a destruicao da FICHA*/
ASSIGNMENT LIST: ficha_anel=0
SET NODES: t_trm_ini /*Inicia proc. atraso de trm. de qds e fc*/
ASSIGNMENT LIST: ini_trab=CLOCK
SET NODES: t_trm_fim /*Final. proc. atraso de trm. de qds e fc*/
ASSIGNMENT LIST: fim_trab=CLOCK ++
trt(*)=trt(*)+(fim_trab-ini_trab)
SET NODES: t_prop_ini /*Inicia proc. atraso de prop. de qds e fc*/
ASSIGNMENT LIST: ini_trab=CLOCK
SET NODES: t_prop_fim /*Final. proc. atraso de prop. de qds e fc*/
ASSIGNMENT LIST: fim_trab=CLOCK ++
trt(*)=trt(*)+(fim_trab-ini_trab)
SET NODES: muda_dest /*Muda dest. do QUADRO p/ est. de origem*/
ASSIGNMENT LIST: jv(destino)=jv(origem)
SET NODES: at_est_fc1 /* Atualiza Estats da Ficha < uma volta */
ASSIGNMENT LIST: fc_n_est=fc_n_est+1
SET NODES: at_est_fc2 /* Atualiza Estats da Ficha = uma volta */
ASSIGNMENT LIST: fc_n_est=1 ++
fc_rotac=fc_rotac+1 ++
fc_t_rotmd=CLOCK/fc_rotac
SET NODES: t_rot_01 /*Modifica GV's p/ trt>ttrt e late_ct=0 */
ASSIGNMENT LIST: /*Ficha atrasada e rotacao anterior OK */ ++
late_ct(j)=1 /* Avisa atraso rotacao atual */ ++
trt(j)=trt(j)-ttrt /* Acumula latencia */ ++
assin_est(j)=0 /* Nao transmite quadros assin. */
SET NODES: t_rot_02 /*Modifica GV's p/ trt<ttrt e late_ct=0 */
ASSIGNMENT LIST: /*Ficha em tempo e rotacao anterior OK */ ++
assin_est(j)=ttrt-trt(j) /* Transmite quadros assin. */ ++
trt(j)=0 /* trt atual e resetado*/
SET NODES: t_rot_03 /*Modifica GV's p/ trt<ttrt e late_ct=1 */
ASSIGNMENT LIST: /*Ficha em tempo e rotacao ant. atrasada*/ ++
late_ct(j)=0 /* Avisa rotacao atual OK */ ++
trt(j)=0 /* trt e resetado */ ++
assin_est(j)=0 /* Nao transmite quadros assin. */
SET NODES: t_rot_04 /*Modifica GV's p/ trt>ttrt e late_ct=1 */

```

ANEXO II.1 - (continuação)

```

ASSIGNMENT LIST: /*Ficha atrasada e rot. ant. atrasada */ ++
late_ct(j)=1      /* Avisa atraso rotacao atual */ ++
trt(j)=trt(j)-ttrt /* Acumula latencia */ ++
assin_est(j)=0    /* Nao transmite quadros assin. */ ++
ab_rede(j)=ab_rede(j)+1 /* E nao transmite quadros sincronos */

SPLIT NODES: cr_ficha
DUMMY NODES: est_entra est_sai ent_dados ent_ficha ficha_cont
DUMMY NODES: ficha_est
CHAIN: cadeia
TYPE: external
INPUT: est_entra
OUTPUT: est_sai
:gersinc -> lib_sinc -> ini_sinc -> qd_rede
:qd_rede -> qd_t_sinc cr_ficha; if(ficha_anel=1) if(t)
:cr_ficha -> qd_t_sinc ini_ficha; split
:ini_ficha -> ficha_ON_1 ficha_ON_2; if(n_vt_yaz<1) if(t)
:ficha_ON_1 ficha_ON_2 -> al_ficha -> cr_sinc
:qd_t_sinc -> qd_al_sinc -> de_sinc_01 -> trm
:trm -> t_trm_fim prop; if(jv(tipo)=ficha) if(t)
:t_trm_fim -> t_prop_ini -> prop
:prop -> t_prop_fim est_sai; if(jv(tipo)=ficha) if(t)
:t_prop_fim -> est_sai
:est_entra -> ent_ficha ficha_OFF ent_dados; ++
if(if(jv(tipo)=ficha) and if(n_qd_anel>0)) ++
if(if(jv(tipo)=ficha) and if(n_qd_anel<1)) if(t)
:ent_dados -> qd_fora muda_dest prop est_sai; ++
if(if(jv(origem)=jv(destino)) and if(jv(destino)=j)) ++
if(if(jv(origem)^=jv(destino)) and if(jv(destino)=j)) ++
if(if(jv(origem)^=jv(destino)) and if(jv(destino)^=j)) ++
if(if(jv(origem)=jv(destino)) and if(jv(destino)^=j))
:qd_fora -> re_t_sinc -> lib_sinc
:muda_dest -> est_sai
:ficha_OFF -> sink
:ent_ficha -> at_est_fc2 at_est_fc1; ++
if(fc_n_est)>(mx_est-1) if(t)
:at_est_fc2 at_est_fc1 -> ficha_cont
:ficha_cont -> t_rot_01 t_rot_02 t_rot_03 t_rot_04; ++
if(if(trt(j)>ttrt) and if(late_ct(j)=0)) /*F Atr. e ant. OK*/ ++
if(if(trt(j)<ttrt) and if(late_ct(j)=0)) /*F OK e ant. OK*/ ++
if(if(trt(j)<ttrt) and if(late_ct(j)=1)) /*F OK e ant. atr.*/ ++
if(if(trt(j)>ttrt) and if(late_ct(j)=1)) /*Aborta Rede*/
:t_rot_01 t_rot_02 t_rot_03 -> ficha_est
:ficha_est -> t_prop_ini cr_sinc;
if(ql(qd_al_sinc)=0) if(t)
:cr_sinc -> para -> fc_al_sinc
:fc_al_sinc -> de_sinc_02 -> t_trm_ini -> trm
:t_rot_04 -> t_prop_ini
END OF SUBMODEL ESTS_SINC
SUBMODEL: ests_assin
NUMERIC PARAMETERS: j /*Indica a estacao atual*/
CHAIN PARAMETERS: cadeia
QUEUE: lib_assing /* Controla liberacao dos qds para o anel*/
TYPE: fcfs
CLASS LIST: lib_assin

```

ANEXO II.1 - (continuação)

```

SERVICE TIMES: tint_assin /*[segundos/quadro]*/
QUEUE: al_ficha /* Controla liberacao dos qds para o anel*/
TYPE: is
CLASS LIST: al_ficha
SERVICE TIMES: constant(alea_ficha)
QUEUE: resp_assin /*Controla o tempo de resposta assincrono*/
TYPE: passive
TOKENS: 2147483647 /*infinita*/
DSPL: fcfs
ALLOCATE NODE LIST: qd_t_assin
NUMBER OF TOKENS TO ALLOCATE: 1
RELEASE NODE LIST: re_t_assin
QUEUE: tht /*Controla banda de trm. assincrona*/
TYPE: passive
TOKENS: 0
DSPL: fcfs
ALLOCATE NODE LIST: qd_al_tht fc_al_tht
NUMBERS OF TOKENS TO ALLOCATE: i ta(tht)
DESTROY NODE LIST: de_tht_01 de_tht_02
CREATE NODE LIST: cr_tht
NUMBERS OF TOKENS TO CREATE: ceil(assin_est(j)/t_trm_qd)
QUEUE: para_q /* Controla parada da fic para liberacao dos qds*/
TYPE: fcfs
CLASS LIST: para
SERVICE TIMES: 0
QUEUE: trm_q /*Controla atraso de trm. de quadros/fichas*/
TYPE: fcfs
CLASS LIST: trm
SERVICE TIMES: constant(jv(tamanho)/cap_trm) /*[bits/bps]*/
QUEUE: prop_q /*Controla atraso de prop. e est. quadros/fichas*/
TYPE: fcfs
CLASS LIST: prop
SERVICE TIMES: constant((at_metro*dist_est)+at_estacao)
SET NODES: gerassin /*Colocacao dos QUADROS assincronos*/
ASSIGNMENT LIST:jv(tipo)= assin
SET NODES: ini_assin /*Inicializa JV's dos QUADROS assincronos*/
ASSIGNMENT LIST:
jv(tamanho)=tam_men ++
jv(origem)=j ++
jv(destino)=ceil(uniform(1,(jv(origem)-1),(jv(origem)-1)/(mx_est-1);(j++
v(origem)+1),mx_est,(mx_est-jv(origem))/(mx_est-1))) ++
SET NODES: qd_rede /*Incrementa quadros rodando no anel*/
ASSIGNMENT LIST: n_qd_anel=n_qd_anel+1 ++
n_qd_ass=n_qd_ass+1
SET NODES: qd_fora /*Decrementa quadros rodando no anel*/
ASSIGNMENT LIST: n_qd_anel=n_qd_anel-1
SET NODES: ini_ficha /*Inicializa JV's da nova FICHA criada*/
ASSIGNMENT LIST:jv(tipo)=ficha ++
jv(tamanho)=tam_fic ++
ficha_anel=1 ++
n_vt_vaz=((CLOCK-fim_trab)/((at_metro*dist_est*mx_est)+(at_estacao*mx_++
est))) ++
fc_n_est=fc_n_est+1 ++
fc_rotac=fc_rotac+n_vt_vaz ++

```


ANEXO II.1 - (continuação)

```

fc_t_rotad=CLOCK/fc_rotac
  SET NODES: ficha_ON_1 /*Atualiza inf. rel. a criacao FICHA */
  ASSIGNMENT LIST: /* n_yt_vaz < 1 */ ++
alea_ficha=(CLOCK-fim_trab)
  SET NODES: ficha_ON_2 /*Atualiza inf. rel. a criacao FICHA */
  ASSIGNMENT LIST: /* n_yt_vaz >= 1 */ ++
alea_ficha=uniform(0,((at_metro*dist_est*mx_est)+(at_estacao*(mx_est-1+ ))),1)
++
trt(*)=constant((at_metro*dist_est*mx_est)+(at_estacao*(mx_est-1)))
  SET NODES: ficha_OFF /*Atualiza inf. rel. a destruicao da FICHA*/
  ASSIGNMENT LIST: ficha_anel=0
  SET NODES: t_trm_ini /*Inicia proc. atraso de trm. de qds e fc*/
  ASSIGNMENT LIST: ini_trab=CLOCK
  SET NODES: t_trm_fim /*Final. proc. atraso de trm. de qds e fc*/
  ASSIGNMENT LIST: fim_trab=CLOCK ++
trt(*)=trt(*)+(fim_trab-ini_trab)
  SET NODES: t_prop_ini /*Inicia proc. atraso de prop. de qds e fc*/
  ASSIGNMENT LIST: ini_trab=CLOCK
  SET NODES: t_prop_fim /*Final. proc. atraso de prop. de qds e fc*/
  ASSIGNMENT LIST: fim_trab=CLOCK ++
trt(*)=trt(*)+(fim_trab-ini_trab)
  SET NODES: muda_dest /*Muda dest. do QUADRO p/ est. de origem*/
  ASSIGNMENT LIST: jv(destino)=jv(origem)
  SET NODES: at_est_fc1 /* Atualiza Estats da Ficha < uma volta */
  ASSIGNMENT LIST: fc_n_est=fc_n_est+1
  SET NODES: at_est_fc2 /* Atualiza Estats da Ficha = uma volta */
  ASSIGNMENT LIST: fc_n_est=1 ++
fc_rotac=fc_rotac+1 ++
fc_t_rotad=CLOCK/fc_rotac
  SET NODES: t_rot_01 /*Modifica GV's p/ trt>ttrt e late_ct=0 */
  ASSIGNMENT LIST: /*Ficha atrasada e rotacao anterior OK */ ++
late_ct(j)=1 /* Avisa atraso rotacao atual */ ++
trt(j)=trt(j)-ttrt /* Acumula latencia */ ++
assin_est(j)=0 /* Nao transmite quadros assin. */
  SET NODES: t_rot_02 /*Modifica GV's p/ trt<=ttrt e late_ct=0 */
  ASSIGNMENT LIST: /*Ficha em tempo e rotacao anterior OK */ ++
assin_est(j)=ttrt-trt(j) /* Transmite quadros assin. */ ++
trt(j)=0 /* trt atual e resetado*/
  SET NODES: t_rot_03 /*Modifica GV's p/ trt<=ttrt e late_ct=1 */
  ASSIGNMENT LIST: /*Ficha em tempo e rotacao ant. atrasada*/ ++
late_ct(j)=0 /* Avisa rotacao atual OK */ ++
trt(j)=0 /* trt e resetado */ ++
assin_est(j)=0 /* Nao transmite quadros assin. */
  SET NODES: t_rot_04 /*Modifica GV's p/ trt>ttrt e late_ct=1 */
  ASSIGNMENT LIST: /*Ficha atrasada e rot. ant. atrasada */ ++
late_ct(j)=1 /* Avisa atraso rotacao atual */ ++
trt(j)=trt(j)-ttrt /* Acumula latencia */ ++
assin_est(j)=0 /* Nao transmite quadros assin. */ ++
ab_rede(j)=ab_rede(j)+1 /* E nao transaite quadros sincronos */
  SPLIT NODES: cr_ficha
  DUMMY NODES: est_entra est_sai ent_dados ent_ficha ficha_cont
  CHAIN: cadeia
  TYPE: external
  INPUT: est_entra

```

ANEXO II.1 - (continuação)

```

OUTPUT: est_sai
:gerassin -> lib_assin -> ini_assin -> qd_rede
:qd_rede -> qd_t_assin cr_ficha; if(ficha_anel=1) if(t)
:cr_ficha -> qd_t_assin ini_ficha; split
  :ini_ficha -> ficha_ON_1 ficha_ON_2; if(n_vt_vaz<1) if(t)
:ficha_ON_1 ficha_ON_2 -> al_ficha -> cr_tht
:qd_t_assin -> qd_al_tht -> de_tht_01 -> trm
:trm -> t_trm_fim prop; if(jv(tipo)=ficha) if(t)
:t_trm_fim -> t_prop_ini-> prop
:prop -> t_prop_fim est_sai; if(jv(tipo)=ficha) if(t)
:t_prop_fim -> est_sai
:est_entra -> ent_ficha ficha_OFF ent_dados; ++
if(if(jv(tipo)=ficha) and if(n_qd_anel>0)) ++
if(if(jv(tipo)=ficha) and if(n_qd_anel<1)) if(t)
  :ent_dados -> qd_fora muda_dest prop est_sai; ++
if(if(jv(origem)=jv(destino)) and if(jv(destino)=j)) ++
if(if(jv(origem)^=jv(destino)) and if(jv(destino)=j)) ++
if(if(jv(origem)^=jv(destino)) and if(jv(destino)^=j)) ++
if(if(jv(origem)=jv(destino)) and if(jv(destino)^=j))
  :qd_fora -> re_t_assin -> lib_assin
  :muda_dest -> est_sai
  :ficha_OFF -> sink
  :ent_ficha -> at_est_fc2 at_est_fc1; ++
if(fc_n_est)>(mx_est-1)) if(t)
  :at_est_fc2 at_est_fc1 -> ficha_cont
  :ficha_cont -> t_rot_01 t_rot_02 t_rot_03 t_rot_04; ++
if(if(trt(j)>ttrt) and if(late_ct(j)=0)) /*F Atr. e ant. OK*/ ++
if(if(trt(j)<=ttrt) and if(late_ct(j)=0))/*F OK e ant. OK*/ ++
if(if(trt(j)<=ttrt) and if(late_ct(j)=1))/*F OK e ant. atr.*/ ++
if(if(trt(j)>ttrt) and if(late_ct(j)=1)) /*Aborta Rede*/
  :t_rot_02 -> t_prop_ini cr_tht;
  if(q1(qd_al_tht)=0) if(t)
  :t_rot_01 t_rot_03 -> t_prop_ini
  :t_rot_04 -> t_prop_ini
:cr_tht -> para -> fc_al_tht -> de_tht_02
:de_tht_02 -> t_trm_ini -> trm
END OF SUBMODEL ESTS_ASSIN

SUBMODEL: ct_ficha
CHAIN PARAMETERS: cadeia
QUEUE: t_rot_med /*Controla o tempo maximo de rotacao da ficha*/
TYPE: passive
TOKENS: 2147483647 /*infinita*/
DSPL: fcfs
ALLOCATE NODE LIST: fc_al_tmp
NUMBERS OF TOKENS TO ALLOCATE: 1
RELEASE NODE LIST: fc_re_tmp
DUMMY NODES: cont_entra cont_sai
CHAIN: cadeia
TYPE: external
INPUT: cont_entra
OUTPUT: cont_sai
:cont_entra -> fc_re_tmp cont_sai; if(jv(tipo)=ficha) if(t)
:fc_re_tmp -> fc_al_tmp -> cont_sai /*Inicializacao da ficha*/
END OF SUBMODEL CT_FICHA

```


ANEXO II.2 Arquivo de rede com a configuração de simulação do exemplo 1

```

MODEL: FDDI_EX01
METHOD: Simulation
NUMERIC PARAMETERS: tint_assin ++
  /* VR [s] Tempo de intercheg 1 QD. ASSIN.*/
NUMERIC IDENTIFIERS: tipo ficha sinc assin
  tipo: 0 /* JV indicando tipo do QUADRO */
  ficha: 1 /* Se QUADRO=FICHA */
  sinc: 2 /* Se QUADRO=dados SINCRONOS */
  assin: 3 /* Se QUADRO=dados ASSINCRONOS */
NUMERIC IDENTIFIERS: tamanho tam_men tam_fic t_trm_qd
  tamanho: 1 /* JV indicando o tamanho do QUADRO */
  tam_men: 32330 /* VR (bits) Tamanho de cada QUADRO */
  tam_fic: 88 /* VR (bits) Tamanho de cada FICHA */
  t_trm_qd: 323.30E-06 /* VR (s) Tempo de trm. de 1 QUADRO */
NUMERIC IDENTIFIERS: origem destino
  origem: 2 /* JV indicando est. origem do QUADRO */
  destino: 3 /* JV indicando est. destino do QUADRO */
NUMERIC IDENTIFIERS: mx_est cap_trm dist_est at_metro at_estacao
  mx_est: 20 /* VR Número máximo de estações */
  cap_trm: 100.0E+06 /* VR (bps) Capacidade de trm. do meio */
  dist_est: 30 /* VR (m) Distancia entre estações */
  at_metro: 5.1E-09 /* VR (s) Atenuação por metro de cabo */
  at_estação: 1.2E-06 /* VR (s) Atenuação por estação */
NUMERIC IDENTIFIERS: tint_sinc
  tint_sinc: 6.75E-03 /* VR (s) Tempo de intercheg 1 QD. SINC.*/
NUMERIC IDENTIFIERS: ttrt sinc_est
  ttrt: 40.0E-03 /* VR (s) Tempo medio de rotação da ficha*/
  sinc_est: 318.00E-06 /* VR (s) Tempo de trm sinc. da estação */
GLOBAL VARIABLES: trt(mx_est) late_ct(mx_est) ab_rede(mx_est)
  trt: 0 /* (s) Tempo de rot. ficha p/ cada est. */
  late_ct: 0 /* (0,1) Controle de overflow de trt(j) */
  ab_rede: 0 /* Vezes que a rede foi abortada na est. */
GLOBAL VARIABLES: assin_est(mx_est)
  assin_est: 0 /* (s) Tempo de Trm. Assin. p/ Est. */
GLOBAL VARIABLES: tmp_cv(mx_est) clock
  tmp_cv: .2 .1 .3 .12 .25 .56 .43 .87 .65 .32 ++
          .23 .44 .12 .78 .92 .54 .77 .21 .84 .33
  clock: 0
GLOBAL VARIABLES: n_qd_anel ficha_anel ini_trab fim_trab alea_ficha
  n_qd_anel: 0 /*num. atual dos qds distribuidos no anel*/
  ficha_anel: 0 /*(1,0) indica existencia ou nao de FICHA*/
  ini_trab: 0 /*(s) indica valor do CLOCK inicio proc.*/
  fim_trab: 0 /*(s) indica valor do CLOCK fim process.*/
  alea_ficha: 0 /* Dist. para liberacao da ficha */
GLOBAL VARIABLES: fc_rotac fc_vazio n_vt_vaz fc_t_rotmd fc_n_est
  fc_rotac: 0 /*num. de rotacoes da FICHA no anel */
  fc_vazio: 0 /*(s) tempo que a FICHA rodou na vazio */
  n_vt_vaz: 0 /*num. de voltas parcial no vazio */
  fc_t_rotmd: 0 /*tempo medio de rotacao ficha no anel */
  fc_n_est: 0 /*num. atual de estacoes perc. pela FICHA*/
GLOBAL VARIABLES: n_qd_ass n_qd_sin
  n_qd_ass: 0 /* Num. de qds assin total oferec ao anel */

```

ANEXO II.2 - (continuação)

```
n_qd_sin: 0      /* Num. de qds sin total oferec ao anel */

MAX JV: 3

/* Definicao da concatenacao final para rodar o exemplo*/
/* EXEMPLO 01 */
INVOCATION: cont_ficha
  TYPE: ct_ficha
  CADEIA: an_FDDI_01
INVOCATION: est_as_01
  TYPE: ests_assin
  J: 1
  CADEIA: an_FDDI_01
INVOCATION: est_as_02
  TYPE: ests_assin
  J: 2
  CADEIA: an_FDDI_01
INVOCATION: est_as_03
  TYPE: ests_assin
  J: 3
  CADEIA: an_FDDI_01
INVOCATION: est_as_04
  TYPE: ests_assin
  J: 4
  CADEIA: an_FDDI_01
INVOCATION: est_as_05
  TYPE: ests_assin
  J: 5
  CADEIA: an_FDDI_01
INVOCATION: est_as_06
  TYPE: ests_assin
  J: 6
  CADEIA: an_FDDI_01
INVOCATION: est_as_07
  TYPE: ests_assin
  J: 7
  CADEIA: an_FDDI_01
INVOCATION: est_as_08
  TYPE: ests_assin
  J: 8
  CADEIA: an_FDDI_01
INVOCATION: est_as_09
  TYPE: ests_assin
  J: 9
  CADEIA: an_FDDI_01
INVOCATION: est_as_10
  TYPE: ests_assin
  J: 10
  CADEIA: an_FDDI_01
INVOCATION: est_as_11
  TYPE: ests_assin
  J: 11
  CADEIA: an_FDDI_01
INVOCATION: est_as_12
```

ANEXO II.2 - (continuação)

```
TYPE: ests_assin
J: 12
CADEIA: an_FDDI_01
INVOCATION: est_as_13
TYPE: ests_assin
J: 13
CADEIA: an_FDDI_01
INVOCATION: est_as_14
TYPE: ests_assin
J: 14
CADEIA: an_FDDI_01
INVOCATION: est_as_15
TYPE: ests_assin
J: 15
CADEIA: an_FDDI_01
INVOCATION: est_as_16
TYPE: ests_assin
J: 16
CADEIA: an_FDDI_01
INVOCATION: est_as_17
TYPE: ests_assin
J: 17
CADEIA: an_FDDI_01
INVOCATION: est_as_18
TYPE: ests_assin
J: 18
CADEIA: an_FDDI_01
INVOCATION: est_as_19
TYPE: ests_assin
J: 19
CADEIA: an_FDDI_01
INVOCATION: est_as_20
TYPE: ests_assin
J: 20
CADEIA: an_FDDI_01
CHAIN: an_FDDI_01
TYPE: open
:cont_ficha -> est_as_01 -> est_as_02 -> est_as_03
:est_as_03 -> est_as_04 -> est_as_05 -> est_as_06
:est_as_06 -> est_as_07 -> est_as_08 -> est_as_09
:est_as_09 -> est_as_10 -> est_as_11 -> est_as_12
:est_as_12 -> est_as_13 -> est_as_14 -> est_as_15
:est_as_15 -> est_as_16 -> est_as_17 -> est_as_18
:est_as_18 -> est_as_19 -> est_as_20 -> cont_ficha
CONFIDENCE INTERVAL METHOD: spectral
INITIAL STATE DEFINITIONS:
CHAIN: AN_FDDI_01
NODE LIST: est_as_01.gerassin
INIT POP: 10
NODE LIST: est_as_02.gerassin
INIT POP: 10
NODE LIST: est_as_03.gerassin
INIT POP: 10
NODE LIST: est_as_04.gerassin
```

ANEXO II.2 - (continuação)

```
INIT POP: 10
NODE LIST: est_as_05.gerassin
INIT POP: 10
NODE LIST: est_as_06.gerassin
INIT POP: 10
NODE LIST: est_as_07.gerassin
INIT POP: 10
NODE LIST: est_as_08.gerassin
INIT POP: 10
NODE LIST: est_as_09.gerassin
INIT POP: 10
NODE LIST: est_as_10.gerassin
INIT POP: 10
NODE LIST: est_as_11.gerassin
INIT POP: 10
NODE LIST: est_as_12.gerassin
INIT POP: 10
NODE LIST: est_as_13.gerassin
INIT POP: 10
NODE LIST: est_as_14.gerassin
INIT POP: 10
NODE LIST: est_as_15.gerassin
INIT POP: 10
NODE LIST: est_as_16.gerassin
INIT POP: 10
NODE LIST: est_as_17.gerassin
INIT POP: 10
NODE LIST: est_as_18.gerassin
INIT POP: 10
NODE LIST: est_as_19.gerassin
INIT POP: 10
NODE LIST: est_as_20.gerassin
INIT POP: 10
CONFIDENCE LEVEL: 90
SEQUENTIAL STOPPING RULE: no
CONFIDENCE INTERVAL QUEUES: cont_ficha.t_rot_med
MEASURES: qt
RUN LIMITS -
LIMIT - CP SECONDS: 2000
TRACE: no
END
```

ANEXO II.3 Arquivo de rede com a configuração de simulação do exemplo 2

```

MODEL: FDDI_02
METHOD: Simulation
NUMERIC PARAMETERS: tint_assin tint_sinc          ++
/* VR (s) Tempo de intercheg 1 QD. ASSIN.*/
NUMERIC IDENTIFIERS: tipo ficha sinc assin
tipo: 0          /* JV indicando tipo do QUADRO */
ficha: 1         /* Se QUADRO=FICHA */
sinc: 2         /* Se QUADRO=dados SINCRONOS */
assin: 3        /* Se QUADRO=dados ASSINCRONOS */
NUMERIC IDENTIFIERS: tamanho tam_men tam_fic t_trm_qd
tamanho: 1      /* JV indicando o tamanho do QUADRO */
tam_men: 32330 /* VR (bits) Tamanho de cada QUADRO */
tam_fic: 88    /* VR (bits) Tamanho de cada FICHA */
t_trm_qd: 323.30E-06 /* VR (s) Tempo de trm. de 1 QUADRO */
NUMERIC IDENTIFIERS: origem destino
origem: 2       /* JV indicando est. origem do QUADRO */
destino: 3     /* JV indicando est. destino do QUADRO */
NUMERIC IDENTIFIERS: mx_est cap_trm dist_est at_metro at_estacao
mx_est: 16     /* VR Número máximo de estações */
cap_trm: 100.00E+06 /* VR (bps) Capacidade de trm. do meio */
dist_est: 30  /* VR (m) Distancia entre estações */
at_metro: 5.1E-09 /* VR (s) Atenuação por metro de cabo */
at_estação: 1.2E-06 /* VR (s) Atenuação por estação */
NUMERIC IDENTIFIERS: ttrt sinc_est
ttrt: 8.00E-03 /* VR (s) Tempo medio de rotação da ficha*/
sinc_est: 350.00E-06 /* VR (s) Tempo de trm sinc. da estação */
GLOBAL VARIABLES: trt(mx_est) late_ct(mx_est) ab_rede(mx_est)
trt: 0 /* (s) Tempo de rot. ficha p/ cada est. */
late_ct: 0 /* (0,1) Controle de overflow de trt(j) */
ab_rede: 0 /* Vezes que a rede foi abortada na est. */
GLOBAL VARIABLES: assin_est(mx_est)
assin_est: 0 /* (s) Tempo de Trm. Assin. p/ Est. */
GLOBAL VARIABLES: tmp_cv(mx_est) clock
tmp_cv: .2 .1 .3 .12 .25 .56 .43 .87 .65 .32 ++
        .23 .44 .12 .78 .92 .54 .77 .21 .84 .33
clock: 0
GLOBAL VARIABLES: n_qd_anel ficha_anel ini_trab fim_trab alea_ficha
n_qd_anel: 0 /*num. atual dos qds distribuidos no anel*/
ficha_anel: 0 /*(1,0) indica existencia ou nao de FICHA*/
ini_trab: 0 /*(s) indica valor do CLOCK inicio proc.*/
fim_trab: 0 /*(s) indica valor do CLOCK fim process.*/
alea_ficha: 0 /* Dist. para liberacao da ficha */
GLOBAL VARIABLES: fc_rotac fc_vazio n_vt_vaz fc_t_rotmd fc_n_est
fc_rotac: 0 /*num. de rotacoes da FICHA no anel */
fc_vazio: 0 /*(s) tempo que a FICHA rodou na vazio */
n_vt_vaz: 0 /*num. de voltas parcial no vazio */
fc_t_rotmd: 0 /*tempo medio de rotacao ficha no anel */
fc_n_est: 0 /*num. atual de estacoes perc. pela FICHA*/
GLOBAL VARIABLES: n_qd_ass n_qd_sin
n_qd_ass: 0 /* Num. de qds assin total oferec ao anel */
n_qd_sin: 0 /* Num. de qds sin total oferec ao anel */

```


ANEXO II.3 - (continuação)

MAX JV: 3

```
/* Definicao da concatenacao final para rodar o exemplo*/
/* EXEMPLO 02 */
INVOCATION: cont_ficha
  TYPE: ct_ficha
  CADEIA: an_FDDI_01
INVOCATION: est_si_01
  TYPE: ests_sinc
  J: 1
  CADEIA: an_FDDI_01
INVOCATION: est_si_02
  TYPE: ests_sinc
  J: 2
  CADEIA: an_FDDI_01
INVOCATION: est_si_03
  TYPE: ests_sinc
  J: 3
  CADEIA: an_FDDI_01
INVOCATION: est_as_04
  TYPE: ests_assin
  J: 4
  CADEIA: an_FDDI_01
INVOCATION: est_as_05
  TYPE: ests_assin
  J: 5
  CADEIA: an_FDDI_01
INVOCATION: est_si_06
  TYPE: ests_sinc
  J: 6
  CADEIA: an_FDDI_01
INVOCATION: est_si_07
  TYPE: ests_sinc
  J: 7
  CADEIA: an_FDDI_01
INVOCATION: est_si_08
  TYPE: ests_sinc
  J: 8
  CADEIA: an_FDDI_01
INVOCATION: est_si_09
  TYPE: ests_sinc
  J: 9
  CADEIA: an_FDDI_01
INVOCATION: est_si_10
  TYPE: ests_assin
  J: 10
  CADEIA: an_FDDI_01
INVOCATION: est_si_11
  TYPE: ests_sinc
  J: 11
  CADEIA: an_FDDI_01
INVOCATION: est_as_12
  TYPE: ests_assin
  J: 12
```

ANEXO II.3 - (continuação)

```
CADEIA: an_FDDI_01
INVOCATION: est_as_13
TYPE: ests_assin
J: 13
CADEIA: an_FDDI_01
INVOCATION: est_si_14
TYPE: ests_sinc
J: 14
CADEIA: an_FDDI_01
INVOCATION: est_si_15
TYPE: ests_sinc
J: 15
CADEIA: an_FDDI_01
INVOCATION: est_si_16
TYPE: ests_sinc
J: 16
CADEIA: an_FDDI_01
CHAIN: an_FDDI_01
TYPE: open
:cont_ficha -> est_si_01 -> est_si_02 -> est_si_03
:est_si_03 -> est_as_04 -> est_as_05 -> est_si_06
:est_si_06 -> est_si_07 -> est_si_08 -> est_si_09
:est_si_09 -> est_si_10 -> est_si_11 -> est_as_12
:est_as_12 -> est_as_13 -> est_si_14 -> est_si_15
:est_si_15 -> est_si_16 -> cont_ficha
CONFIDENCE INTERVAL METHOD: spectral
INITIAL STATE DEFINITIONS:
CHAIN: AN_FDDI_01
NODE LIST: est_as_01.gersinc
INIT POP: 3
NODE LIST: est_as_02.gersinc
INIT POP: 3
NODE LIST: est_as_03.gersinc
INIT POP: 3
NODE LIST: est_si_04.gerassin
INIT POP: 20
NODE LIST: est_si_05.gerassin
INIT POP: 20
NODE LIST: est_si_06.gersinc
INIT POP: 3
NODE LIST: est_si_07.gersinc
INIT POP: 3
NODE LIST: est_si_08.gersinc
INIT POP: 3
NODE LIST: est_si_09.gersinc
INIT POP: 3
NODE LIST: est_as_10.gersinc
INIT POP: 3
NODE LIST: est_as_11.gersinc
INIT POP: 3
NODE LIST: est_si_12.gerassin
INIT POP: 20
NODE LIST: est_si_13.gerassin
INIT POP: 20
```

ANEXO II.3 - (continuação)

```
NODE LIST: est_si_14.gersinc
  INIT POP: 3
NODE LIST: est_si_15.gersinc
  INIT POP: 3
NODE LIST: est_si_16.gersinc
  INIT POP: 3
CONFIDENCE LEVEL: 90
SEQUENTIAL STOPPING RULE: no
CONFIDENCE INTERVAL QUEUES: cont_ficha.t_rot_med
  MEASURES: qt
RUN LIMITS -
LIMIT - CP SECONDS: 2000
TRACE: no
END
```

ANEXO II.4 Arquivo de rede com a configuração de simulação do exemplo 3

```

MODEL: FDDI_EX03
METHOD: Simulation
NUMERIC PARAMETERS: ttrt ++
NUMERIC IDENTIFIERS: tipo ficha sinc assin
    tipo: 0 /* JV indicando tipo do QUADRO */
    ficha: 1 /* Se QUADRO=FICHA */
    sinc: 2 /* Se QUADRO=dados SINCRONOS */
    assin: 3 /* Se QUADRO=dados ASSINCRONOS */
NUMERIC IDENTIFIERS: tamanho tam_men tam_fic t_trm_qd
    tamanho: 1 /* JV indicando o tamanho do QUADRO */
    tam_men: 36000 /* VR (bits) Tamanho de cada QUADRO */
    tam_fic: 80 /* VR (bits) Tamanho de cada FICHA */
    t_trm_qd: 360.00E-06 /* VR (s) Tempo de trm. de 1 QUADRO */
NUMERIC IDENTIFIERS: origem destino
    origem: 2 /* JV indicando est. origem do QUADRO */
    destino: 3 /* JV indicando est. destino do QUADRO */
NUMERIC IDENTIFIERS: mx_est cap_trm dist_est at_metro at_estacao
    mx_est: 20 /* VR Número máximo de estações */
    cap_trm: 100.00E+06 /* VR (bps) Capacidade de trm. do meio */
    dist_est: 30 /* VR (m) Distancia entre estações */
    at_metro: 5.1E-09 /* VR (s) Atenuação por metro de cabo */
    at_estacao: 1.2E-06 /* VR (s) Atenuação por estação */
NUMERIC IDENTIFIERS: tint_sinc
    tint_sinc: 6.75E-03 /* VR (s) Tempo de intercheg 1 QD. SINC*/
    tint_assin: 7.2E-03 /* VR (s) Tempo de intercheg 1 QD. SINC*/
NUMERIC IDENTIFIERS: ttrt sinc_est
    sinc_est: 318.00E-06 /* VR (s) Tempo de trm sinc. da estação */
GLOBAL VARIABLES: trt(mx_est) late_ct(mx_est) ab_rede(mx_est)
    trt: 0 /* (s) Tempo de rot. ficha p/ cada est. */
    late_ct: 0 /* (0,1) Controle de overflow de trt(j) */
    ab_rede: 0 /* Vezes que a rede foi abortada na est. */
GLOBAL VARIABLES: assin_est(mx_est)
    assin_est: 0 /* (s) Tempo de Trm. Assin. p/ Est. */
GLOBAL VARIABLES: tmp_cv(mx_est) clock
    tmp_cv: .2 .1 .3 .12 .25 .56 .43 .87 .65 .32 ++
            .23 .44 .12 .78 .92 .54 .77 .21 .04 .33
    clock: 0
GLOBAL VARIABLES: n_qd_anel ficha_anel ini_trab fim_trab alea_ficha
    n_qd_anel: 0 /*num. atual dos qds distribuidos no anel*/
    ficha_anel: 0 /*(1,0) indica existencia ou nao de FICHA*/
    ini_trab: 0 /*(s) indica valor do CLOCK inicio proc.*/
    fim_trab: 0 /*(s) indica valor do CLOCK fim process.*/
    alea_ficha: 0 /* Dist. para liberacao da ficha */
GLOBAL VARIABLES: fc_rotac fc_vazio n_yt_vaz fc_t_rotmd fc_n_est
    fc_rotac: 0 /*num. de rotacoes da FICHA no anel */
    fc_vazio: 0 /*(s) tempo que a FICHA rodou na vazio */
    n_yt_vaz: 0 /*num. de voltas parcial no vazio */
    fc_t_rotmd: 0 /*tempo medio de rotacao ficha no anel */
    fc_n_est: 0 /*num. atual de estacoes perc. pela FICHA*/
GLOBAL VARIABLES: n_qd_ass n_qd_sin
    n_qd_ass: 0 /* Num. de qds assin total oferec ao anel */
    n_qd_sin: 0 /* Num. de qds sin total oferec ao anel */

```

ANEXO II.4 - (continuação)

MAX JV: 3

```
/* Definicao da concatenacao final para rodar o exemplo*/
/* EXEMPLO 01 */
INVOCATION: cont_ficha
  TYPE: ct_ficha
  CADEIA: an_FDDI_01
INVOCATION: est_as_01
  TYPE: ests_assin
  J: 1
  CADEIA: an_FDDI_01
INVOCATION: est_as_02
  TYPE: ests_assin
  J: 2
  CADEIA: an_FDDI_01
INVOCATION: est_as_03
  TYPE: ests_assin
  J: 3
  CADEIA: an_FDDI_01
INVOCATION: est_as_04
  TYPE: ests_assin
  J: 4
  CADEIA: an_FDDI_01
INVOCATION: est_as_05
  TYPE: ests_assin
  J: 5
  CADEIA: an_FDDI_01
INVOCATION: est_as_06
  TYPE: ests_assin
  J: 6
  CADEIA: an_FDDI_01
INVOCATION: est_as_07
  TYPE: ests_assin
  J: 7
  CADEIA: an_FDDI_01
INVOCATION: est_as_08
  TYPE: ests_assin
  J: 8
  CADEIA: an_FDDI_01
INVOCATION: est_as_09
  TYPE: ests_assin
  J: 9
  CADEIA: an_FDDI_01
INVOCATION: est_as_10
  TYPE: ests_assin
  J: 10
  CADEIA: an_FDDI_01
INVOCATION: est_as_11
  TYPE: ests_assin
  J: 11
  CADEIA: an_FDDI_01
INVOCATION: est_as_12
  TYPE: ests_assin
  J: 12
```


ANEXO II.4 - (continuação)

```
CADEIA: an_FDDI_01
INVOCATION: est_as_13
TYPE: ests_assin
J: 13
CADEIA: an_FDDI_01
INVOCATION: est_as_14
TYPE: ests_assin
J: 14
CADEIA: an_FDDI_01
INVOCATION: est_as_15
TYPE: ests_assin
J: 15
CADEIA: an_FDDI_01
INVOCATION: est_as_16
TYPE: ests_assin
J: 16
CADEIA: an_FDDI_01
INVOCATION: est_as_17
TYPE: ests_assin
J: 17
CADEIA: an_FDDI_01
INVOCATION: est_as_18
TYPE: ests_assin
J: 18
CADEIA: an_FDDI_01
INVOCATION: est_as_19
TYPE: ests_assin
J: 19
CADEIA: an_FDDI_01
INVOCATION: est_as_20
TYPE: ests_assin
J: 20
CHAIN: an_FDDI_01
TYPE: open
:cont_ficha -> est_as_01 -> est_as_02 -> est_as_03
:est_as_03 -> est_as_04 -> est_as_05 -> est_as_06
:est_as_06 -> est_as_07 -> est_as_08 -> est_as_09
:est_as_09 -> est_as_10 -> est_as_11 -> est_as_12
:est_as_12 -> est_as_13 -> est_as_14 -> est_as_15
:est_as_15 -> est_as_16 -> est_as_17 -> est_as_18
:est_as_18 -> est_as_19 -> est_as_20 -> cont_ficha
CONFIDENCE INTERVAL METHOD: spectral
INITIAL STATE DEFINITIONS:
CHAIN: AN_FDDI_01
NODE LIST: est_as_01.gerassin
INIT POP: 10
NODE LIST: est_as_02.gerassin
INIT POP: 10
NODE LIST: est_as_03.gerassin
INIT POP: 10
NODE LIST: est_as_04.gerassin
INIT POP: 10
NODE LIST: est_as_05.gerassin
```

ANEXO II.4 - (continuação)

```
INIT POP: 10
NODE LIST: est_as_06.gerassin
INIT POP: 10
NODE LIST: est_as_07.gerassin
INIT POP: 10
NODE LIST: est_as_08.gerassin
INIT POP: 10
NODE LIST: est_as_09.gerassin
INIT POP: 10
NODE LIST: est_as_10.gerassin
INIT POP: 10
NODE LIST: est_as_11.gerassin
INIT POP: 10
NODE LIST: est_as_12.gerassin
INIT POP: 10
NODE LIST: est_as_13.gerassin
INIT POP: 10
NODE LIST: est_as_14.gerassin
INIT POP: 10
NODE LIST: est_as_15.gerassin
INIT POP: 10
NODE LIST: est_as_16.gerassin
INIT POP: 10
NODE LIST: est_as_17.gerassin
INIT POP: 10
NODE LIST: est_as_18.gerassin
INIT POP: 10
NODE LIST: est_as_19.gerassin
INIT POP: 10
NODE LIST: est_as_20.gerassin
INIT POP: 10
CONFIDENCE LEVEL: 90
SEQUENTIAL STOPPING RULE: no
CONFIDENCE INTERVAL QUEUES: cont_ficha.t_rot_med
MEASURES: qt
RUN LIMITS -
LIMIT - CP SECONDS: 2000
TRACE: no
END
```

ANEXO III.1 Passos para simulação de uma configuração.

1. Para se efetuar uma simulação é necessário integrar o arquivo de rede com o arquivo de sub-estações. Assim, nomeia-se o arquivo de rede com a configuração desejada para um <arquivo de simulação> com outro nome (comando "copy" do IBM).
2. Edita através do XEDIT o <arquivo de simulação> e dentro do editor executa os seguintes comandos:
 - . TOP
 - . CLOCATE /MAX JV/
 - . GET <arquivo de estações> RQ2INP A1
 - . FILE
3. Compila o <arquivo de simulação> concatenado através do comando SETUP.
4. Executa a simulação através do comando EVAL do RESQ2.

ANEXO III.2 Memorial de Cálculo para Obtenção das Estatísticas de forma resumida.

1. Número de Quadros Inseridos pelas estações ao anel:

. Estatística RESQ2: gv	
. Nome da Variável:	Tipo da Estação:
<n_qd_sinc>	síncrona
<n_qd_assin>	assíncrona

2. Tempo simulado total:

. Estatística RESQ2: SIMULATED TIME

3. Tempo médio de rotação da ficha:

. Estatística RESQ2: gv
 . Nome da Variável: <fc_t_rotmd>

4. Carga Aplicada ao Anel/Vazão/Utilização:

. Estatística RESQ2: nd
 . Nome da Variável: <est_xx_xx.ent_dados>

. Cálculos:

- .. Achar a média do nd de todas estações;
- .. Multiplicar pelo tamanho médio do quadro em bits;
- .. Dividir pelo tempo simulado total.

5. Tempo de Resposta:

5.1 Médio

. Estatística RESQ2: qt	
. Nome da Variável:	Tipo da Estação:
<est_si_xx.resp_sinc>	síncrona
<est_as_xx.resp_assin>	assíncrona

. Cálculos:

- .. Achar a média do valor por tipo de estação.

5.2 Máximo

. Estatística RESQ2: mxqt	
. Nome da Variável:	Tipo da Estação:
<est_si_xx.resp_sinc>	síncrona
<est_as_xx.resp_assin>	assíncrona

. Cálculos:

- .. Achar o maior valor de todos por tipo de estação.

ANEXO III.2 - (continuação)

6. Tempo de Acesso:

6.1 Médio

- . Estatística RESQ2: qt
- . Nome da Variável: Tipo da Estação:
 - <est_si_xx.t_sinc_est> síncrona
 - <est_as_xx.tht> assíncrona
- . Cálculos:
 - .. Achar a média do valor por tipo de estação.

5.2 Máximo

- . Estatística RESQ2: mxqt
- . Nome da Variável: Tipo da Estação:
 - <est_si_xx.t_sinc_est> síncrona
 - <est_as_xx.tht> assíncrona
- . Cálculos:
 - .. Achar o maior valor de todos por tipo de estação.