

4: SBRC

RECIFE - 24 A 26 DE MARÇO 86

DIMAC: Um Protocolo de Acesso por Passagem de Token*

Clylton Galamba Fernandes
Jaelson Freire Brelaz de Castro
Paulo Roberto Freire Cunha

Grupo de Redes de Computadores e
Sistemas Distribuídos (REDIS)

Departamento de Informática da UFPe
Cidade Universitária
Recife-PE

Resumo

Este trabalho apresenta a especificação do protocolo de controle de acesso ao meio a ser utilizado na Rede Local do Departamento de Informática da UFPe. O protocolo é do tipo por passagem de token, tendo evoluído a partir do "Token Bus", o padrão IEEE 802.4. O protocolo é especificado utilizando-se Redes de Petri estendidas com parâmetros temporais e arcos inibidores.

(*) Este trabalho contou com auxílio parcial do CNPq.

1 Introdução

Entre as diversas topologias e métodos de acesso ao meio de comunicação (MAC) propostos para redes locais, optamos, para a rede local do Departamento de Informática da UFPE, por uma rede em barra com protocolo de acesso por passagem de token.

O padrão IEEE 802.4, o "Token Bus", é um método de acesso por passagem de token [1,2]. O protocolo de acesso DIMAC a ser descrito aqui evoluiu a partir do "Token Bus", no qual foram introduzidas várias modificações visando maior simplicidade em alguns aspectos e maior eficiência em outros.

No "Token Bus" as estações ativas na barra formam um anel lógico, ou seja, uma sequência ordenada de estações com a última seguida da primeira. Cada estação conhece o endereço da estação que a precede e o da estação que a sucede no anel lógico. Uma mensagem de controle, chamada token, é utilizada para controlar o acesso ao meio. Um único token é repassado continuamente de estação para estação na barra. Ao concluir sua transmissão, a estação detentora repassa o token para a sua sucessora. Assim, em regime permanente, a operação do DIMAC consiste na alternância de procedimento de transferência de dados e repasse do token.

Além do funcionamento em regime permanente, o "Token Bus" cuida da inicialização do anel e da gerência de situações transitórias. A inicialização se faz necessária quando as estações são primeiramente ligadas. Já as situações transitórias são causadas ou pela entrada de novas estações no anel (estações ligadas após a formação do anel lógico), ou pela saída de estações do anel (usuário deseja desligar sua estação), ou mesmo pela ocorrência de falhas que podem tornar o anel inoperante caso não haja procedimento de recuperação.

Enquanto que a 'entrada' de novas estações no anel lógico é sempre um procedimento sob o controle do protocolo de acesso, a 'saída' nem sempre é feita sob seu controle. Isto porque o usuário pode simplesmente desligar sua estação sem dar chance aos procedimentos de deleção atuar. Neste caso a 'saída' consiste em uma falha que o protocolo detecta e corrige. Outras falhas que podem ocorrer são token em duplicata e perda de token (a estação detentora do token é desligada) que causam a solução drástica de inicialização do anel.

Todos estes aspectos e funções são igualmente válidos para o DIMAC, que funciona exatamente como o "Token Bus" exceto por algumas diferenças. A primeira diferença entre o DIMAC e o "Token Bus" é a não existência de estações do tipo "Non Token Passing" [2]. No "Token Bus", estações "Non Token Passing" são aquelas que não participam do anel lógico, ou seja, só transmitem respondendo a "polls" e pedidos de confirmação sob comando do LLC (Logic Link Control) sem que para isso precisem estar de posse do token. No caso do DIMAC todas as estações participam do anel lógico, precisando portanto estar de posse do token mesmo para responder

a "polls" e pedidos de confirmação. Verificamos que a complexidade adicional a ser introduzida ao DIMAC para suportar estações do tipo "Non Token Passing" não se justifica quando o número de estações deste tipo é reduzido, como no nosso caso.

As diferenças mais básicas, no entanto estão na gerência dos processos de contenção, presente nas situações transitórias. No DIMAC por exemplo, as funções de inicialização do anel e adição de novas estações são realizadas por um procedimento único. Na realidade o procedimento de adição é idêntico ao procedimento de inicialização do "Token Bus" só que restrito às estações entre a detentora e sua sucessora.

2 Descrição do Protocolo de Acesso

As mensagens que transitam na barra podem ser mensagens de controle de acesso, de responsabilidade do DIMAC, ou mensagem do protocolo de controle de link (LLC) transmitidas pelas estações detentoras do TOKEN, conforme ilustrado na Figura 1, onde os campos estão especificados em bytes.

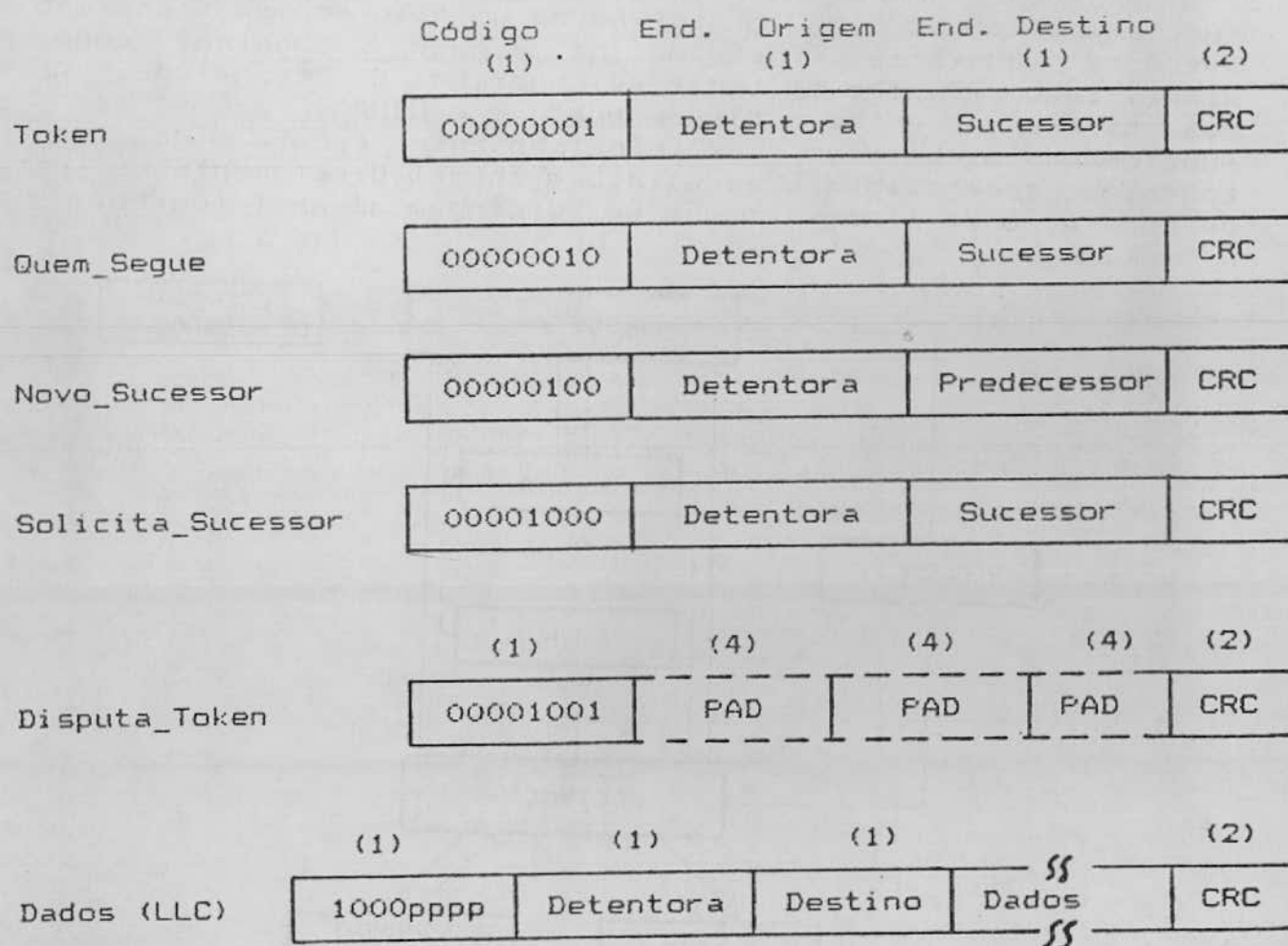


Figura 1

As mensagens do DIMAC transmitidas pela estação detentora do token são:

Token- Para repasse do controle da barra à sua sucessora.

Quem_Segue- Para identificar a estação que se sucede à sua sucessora no anel lógico.

Solicita_Sucessor- Para permitir as estações com endereço entre ela própria e sua sucessora a disputar a posse do token para entrar no anel lógico.

Novo_Sucessor- Para informar à sua predecessora que está se retirando do anel. Esta mensagem pode também ser recebida pela estação detentora vindo da estação que ganhou a disputa pela posse do token provocada pelo envio da mensagem Solicita_Sucessor. As mensagens de Dados são repassadas diretamente ao LLC para processamento.

A melhor maneira de se entender a estrutura global de funcionamento do DIMAC é através de um diagrama de transição de estados, conforme ilustrado na Figura 2 abaixo, onde por simplicidade estão omitidas as condições de transição. Apesar das diferenças entre o "Token Bus" e o DIMAC, ambos têm seu funcionamento baseado na transição entre escuta e detenção do token. Ao serem ligadas as estações entram diretamente no estado de Escuta, onde ficam até que se adicionem ao anel lógico.

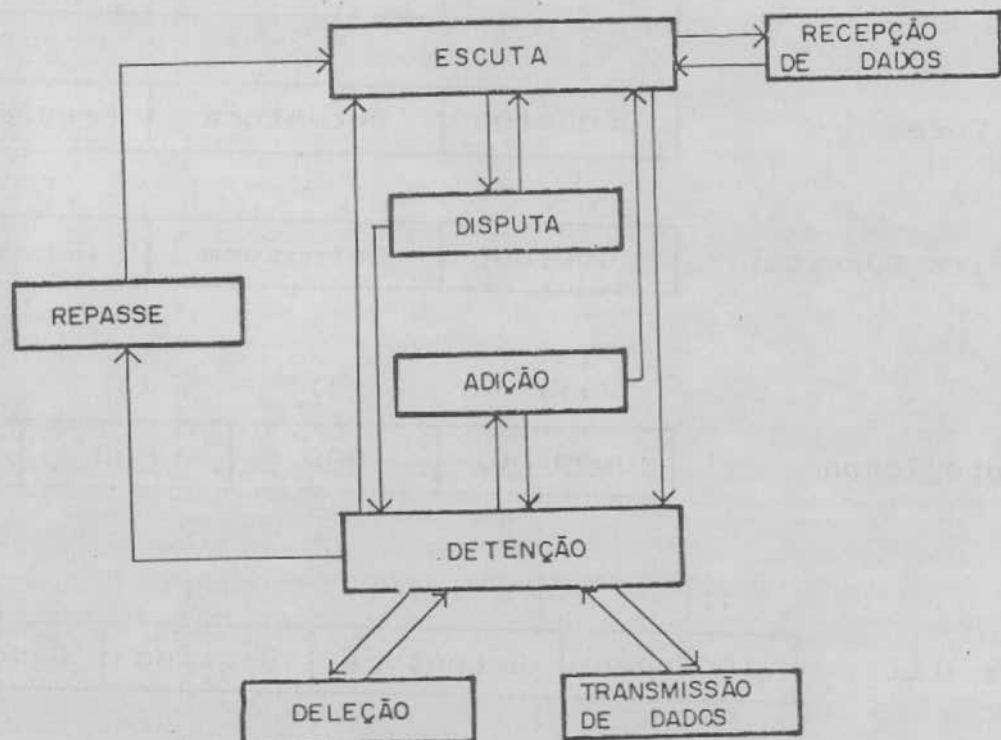


Figura 2

Enquanto no estado de Escuta as estações podem passar para o estado de:

-Recepção, caso seja endereçada com uma mensagem de controle de link, retornando em seguida à Escuta.

-Detenção do Token, caso seja endereçada por uma mensagem Token, ou por uma mensagem Quem_Segue, de onde poderá Transmitir Dados, antes de repassar o Token para a sua sucessora.

-Disputa pela Posse do Token, caso seja endereçada por uma mensagem Solicita_Sucessor e seu endereço esteja compreendido entre a estação Origem e a Destino no anel lógico, ou ocorra o Time-out de escuta devido à inatividade prolongada na barra. O primeiro caso trata-se de uma adição e o segundo de uma inicialização. O procedimento de Disputa garante que uma única estação ganhe a posse do token com as demais retornando ao estado de Escuta. Uma mensagem Novo_Sucessor é emitida pela estação ganhadora se identificando à estação que provocou a disputa, a qual atualiza o endereço de sua nova sucessora e passa ao estado de Escuta com a ganhadora passando ao estado de Detenção. No processo de Disputa as estações participantes transmitem concorrentemente mensagens Disputa-Token cujo comprimento é função do endereço das estações, conforme descrito mais adiante.

Enquanto no estado de Detenção uma estação poderá ir para o estado de :

-Transmissão de Mensagens do controle de link retornando em seguida ao estado de Detenção.

-Deleção, onde emite uma mensagem Novo_Sucessor informando a estação que a precede que está se retirando do anel lógico, e retorna ao estado de Detenção.

-Adição, onde envia uma mensagem Solicita_Sucessor permitindo as estações entre ela e sua sucessora a disputarem pela posse do token. Caso receba uma mensagem Novo_Sucessor vai para o estado de Escuta, caso contrário retorna ao estado de Detenção.

-Repasse do Token, onde a estação envia o Token para sua sucessora e retorna ao estado de Escuta. Caso haja insucesso no repasse (sucessora desligada, por exemplo) a estação tenta repassar o token para a estação seguinte no anel lógico enviando uma mensagem Quem_Segue. Caso receba uma mensagem Novo_Sucessor atualiza sua nova sucessora e vai para a Escuta. Caso contrário irá para a escuta provocando a inicialização do anel.

Ao entrar no estado de Detenção de Token, as estações primeiramente verificam a necessidade de transmissão do LLC, de adição e deleção antes de repassar o token. A adição de nova estação ao anel é provocada após um número fixo de entradas no estado de detenção.

3 Especificação do DIMAC usando Redes de Petri

Para a especificação do DIMAC procurou-se um método que fosse simples conceitualmente mas ao mesmo tempo poderoso na representação e análise do fluxo de controle e informação do protocolo, que envolva processamento de natureza assíncrona e concorrente. Dentre os muitos métodos utilizados para modelagem formal e verificação de sistemas concorrentes foi escolhido uma extensão das Redes de Petri [3,4,5], onde foram introduzidos parâmetros temporais e arcos inibidores [6,7,8].

Visando facilitar o entendimento e não sobrecarregar o leitor, o protocolo será especificado utilizando uma coleção de subgrafos, que correspondem aos procedimentos distintos do protocolo a saber :

- Escuta
- Transmissão de Dados
- Recepção de Dados
- Disputa pela Posse do Token
- Passagem de Token
- Adição
- Deleção
- Detenção do Token (posse)

A interação entre os subgrafos é obtida através dos lugares comuns destes subgrafos. O meio de transmissão é modelado através de um subgrafo de apenas um lugar, MSG. A presença de uma mensagem no meio é representada pela presença de uma ficha neste lugar.

3.1 Subgrafo Escuta

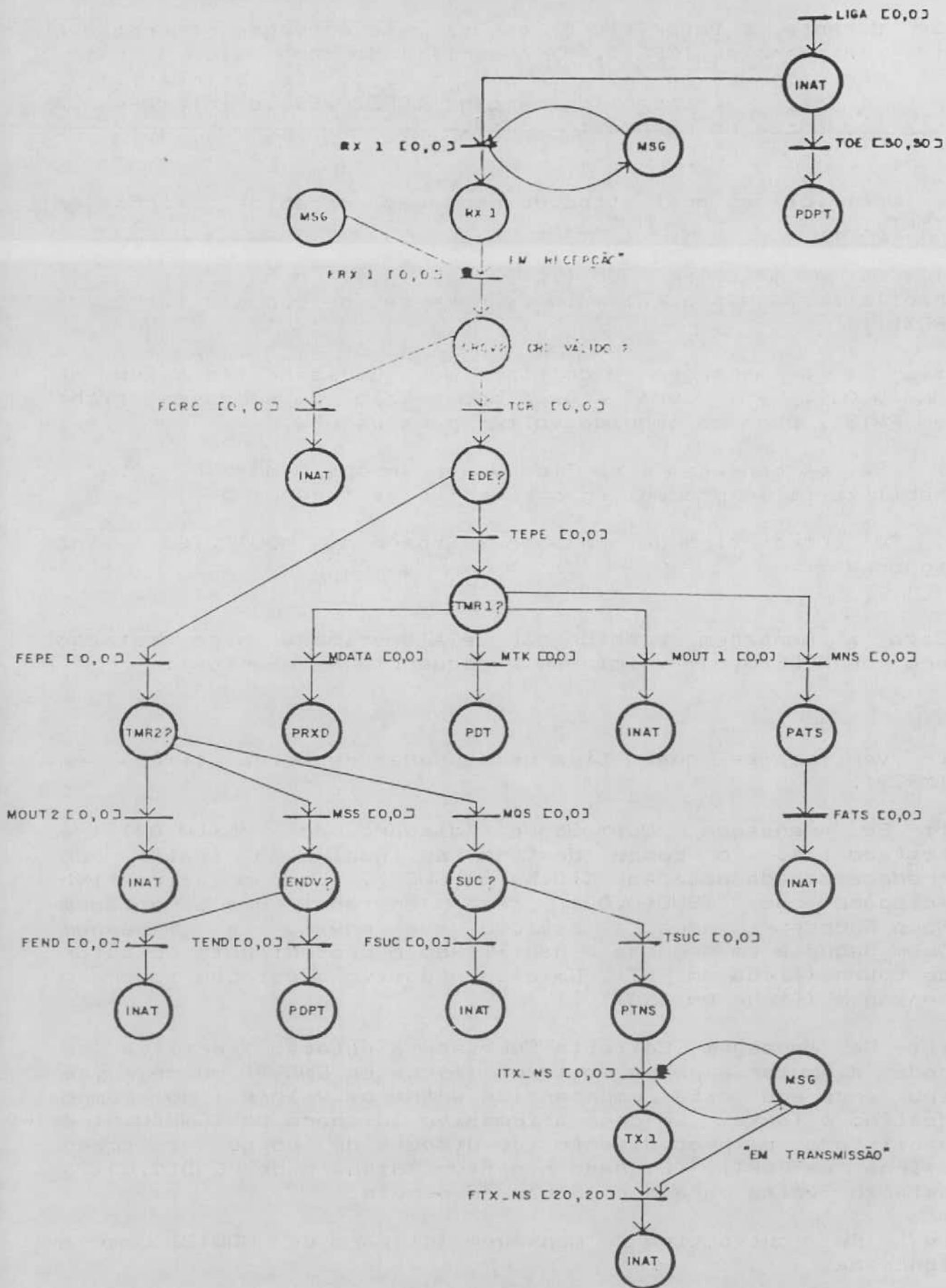
Durante o procedimento Escuta (Figura 3), iniciado pela presença de uma ficha em INAT, a estação aguarda a recepção de mensagens. Estas podem ter sido endereçadas para a estação ou não. Dependendo do tipo de mensagem recebida, uma determinada ação é tomada. A mensagem pode ser aguardada por um período máximo igual a T_{oe} (time-out de escuta). Esgotado este limite (disparo de $T_{oe}[50,50]$), a estação atualiza o seu endereço de predecessor e sucessor (que passa a ser o da própria estação) e habilita o procedimento de disputa da posse do token (ficha em PDPT).

Se durante o intervalo de escuta uma mensagem correta é recebida (disparo de TCRC[0,0]) é verificado se o valor do campo destino corresponde ao endereço da estação. Se afirmativo a mensagem é para a estação (disparo de TEPE[0,0]), e inicia-se a seguinte sequência de eventos:

- i- Verifica-se qual tipo de mensagem recebida (ficha em TMR?).
- ii- Se a mensagem é do tipo Dados (disparo de Mdata[0,0]) habilita-se o procedimento de Recepção de Dados (ficha em PRXD).
- iii- Se a mensagem é do tipo Novo_Sucessor (disparo de MNS[0,0]), será atualizado o valor do sucessor (ficha em PATS), para em seguida voltar para escuta.
- iv- Se a mensagem é do Tipo Token (disparo de MTK [0,0]), habilita-se o procedimento detentor de token (PDT).
- v- Se outro tipo de mensagem (disparo de MOUT[0,0]) será ignorada.

Caso a mensagem recebida não seja destinada para estação (disparo de NEPE[0,0]) inicia-se a sequência de eventos :

- i- Verifica-se qual tipo de mensagem recebida (ficha em TMR2?).
- ii- Se mensagem Quem_Segue (disparo de MOS[0,0]) é testado se o campo destino é igual ao valor do predecessor da estação (ficha em SUC?). Em caso afirmativo (disparo de TSUC[0,0]) será enviada uma mensagem Novo_Sucessor para a estação que enviou a mensagem Quem_Segue e em seguida é habilitado o procedimento detentor de token (ficha em PDT). Em caso negativo a estação ignora o mensagem (ficha em INAT).
- iii- Se mensagem Solicita-Sucessor a estação verifica se pode disputar a posse do token (ficha em ENDV?) ou seja, se seu endereço está compreendido entre os valores do campo destino e fonte. Em caso afirmativo (disparo de TEND[0,0]) é habilitado o procedimento de disputa de posse de token (ficha em PDPT). Em caso negativo (disparo de FEND[0,0]) a estação volta para a condição de escuta.
- iv - Se outro tipo de mensagem (disparo de MOUT2) será ignorada.



3.2 Subgrafo Detentor de Token

Durante o procedimento Detentor de Token (Figura 4) a estação possui o token podendo então habilitar um dos seguintes procedimentos: Repasse de Token, Adição, Deleção ou Transmissão de Dados.

i- Se a condição de passar token for válida (ficha em PAST) será habilitado o procedimento de Passagem de Token (disparo de IPPT[0,0]).

ii- Se a condição de fim de transmissão for válida (ausência de ficha em FTXD) será habilitado o procedimento de transmissão de dados bem como será incrementado o contador 1 (ficha em PTXD e CT1).

iii- Se a condição fim de transmissão de dados for válida (ficha em FTXD), será atualizado o endereço do predecessor com o valor do campo origem da mensagem de token recebida e em seguida é verificado se a estação pretende continuar no anel (ficha em FADP). Em caso negativo (disparo de FCA[0,0]) será habilitado o procedimento de Deleção (ficha em PDEL). Em caso afirmativo ir para passo iv.

iv-Verificar se é hora de adicionar novas estações no anel. Em caso afirmativo (disparo de TAD[0,0]) habilitar o procedimento de Adição (ficha em PAD). Em caso negativo (disparo de FAD[0,0]) habilitar procedimento de transferência de dados (PTXD).

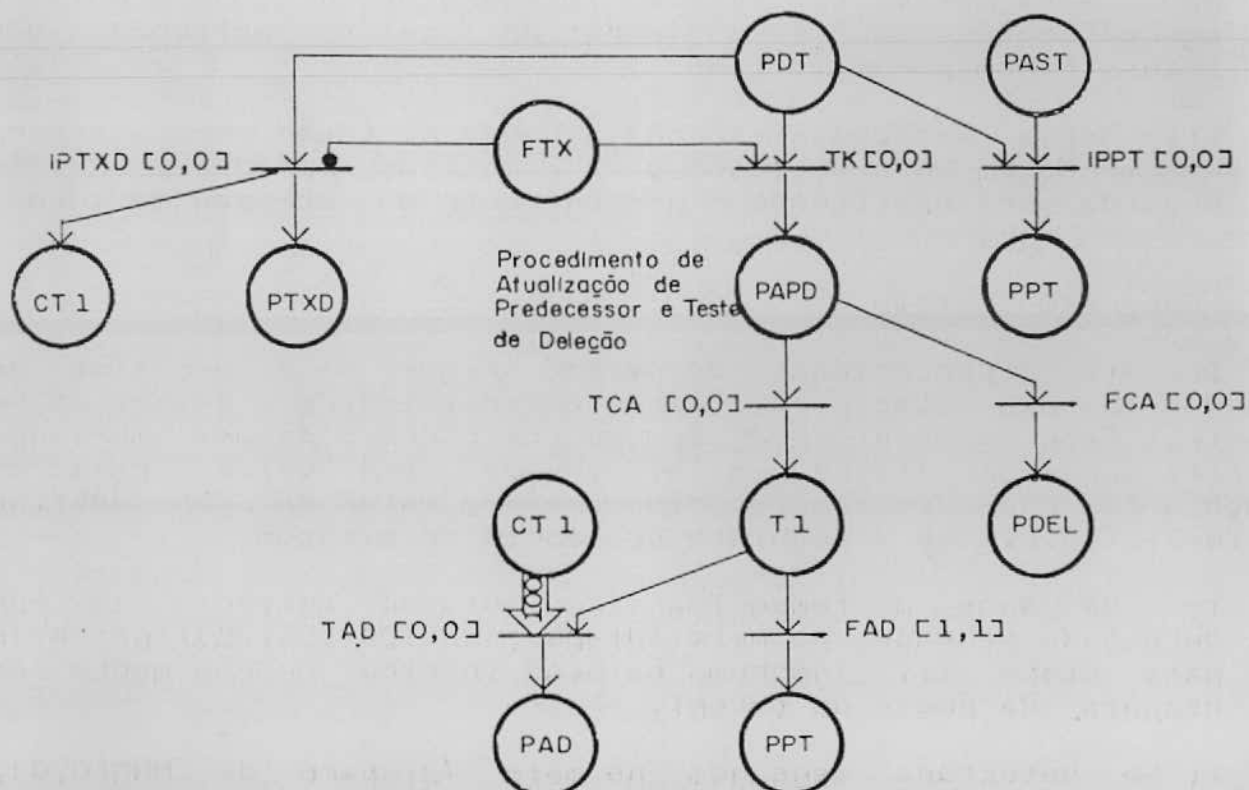


Figura 4. Subgrafo Detentor de Token

3.3 Subgrafo Disputa de Token

Este procedimento de Disputa da posse do Token (Figura 5) consiste de 4 fases onde em cada fase as estações endereçadas pela mensagem Solicita_Sucessor transmitem uma mensagem Disputa_Token.

Na primeira fase cada estação transmite sua mensagem Disputa_Token expandida de 0,1,2,ou 3 PADS, dependendo do primeiro par de bits do seu endereço. Na segunda fase a expansão depende do segundo par de bits e assim por diante.

Ao término de cada transmissão, a estação observa a barra e caso haja alguma estação ainda transmitindo (mensagem Disputa_Token mais longa) desiste da disputa, caso contrário passa à fase seguinte. A disputa tem como ganhadora a estação que ao término da quarta fase encontrar a barra silenciosa. Após a transmissão da mensagem Disputa_Token na primeira fase, a estação passa a escutar o meio (ficha em EM1). Inicia-se a sequência de eventos:

i- Se há mensagem no meio (disparo de TMNM[0,0]) a estação desiste da disputa do token, passando para o procedimento de escuta.

ii- Se não há mensagem no meio (disparo de FMNM[0,0]) será enviado uma nova mensagem Disputa_Token expandida (disparo de NMPT) baseada no próximo par de bits do seu endereço,. Ficarà com a posse do token a estação que usar todos os pares de bits do seu endereço e não escutar mensagem no meio após esta última transmissão (disparo de PT[0,0]), pois o procedimento só será repetido pelas estações que transmitiram por mais tempo.

iii- Se a estação ficar com a posse do token será enviada uma mensagem Novo_Sucessor para a estação predecessora. Em seguida será habilitado o procedimento de detenção de token.

3.4 Subgrafo Adição

Durante o procedimento de Adição (Figura 6) é permitido as estações que não estão participando do anel lógico a chance de se inserir. Este procedimento começa com o envio de uma mensagem Solicita_Sucessor (ficha em PTSS, que será válido para as estações com endereço compreendido entre o valor do campo destino e origem). Inicia-se a seguinte sequência de eventos.

i- Se após o tempo Toa (time-out de adição) não for detectada mensagem no meio (disparo de TOA [25,25]) passa-se para etapa iii (nenhuma estação iniciou procedimento de disputa da posse do token).

ii- Se detectada mensagem no meio (disparo de MNM[0,0], início da disputa pela posse do token) esperar um tempo

mínimo suficiente para o término da disputa de Time[12,12]). A estação passa a aguardar uma mensagem Novo_Sucessor (ficha em EMNS) daquela que ficou com a posse do token. Se esta mensagem não for recebida dentro de um tempo máximo Toad (disparo de TOAD[25,25]) seguir etapa iii. Caso contrário (disparo de TMNS[0,0]) será atualizado o sucessor (ficha em PATSD2) e em seguida a estação volta para escuta.

iii-Verifica-se se esta já foi a segunda tentativa de adição. Em caso afirmativo (disparo de TST1[0,0]) habilita-se o procedimento detentor de token com a condição de passagem de token (ficha em PDT e PAST).

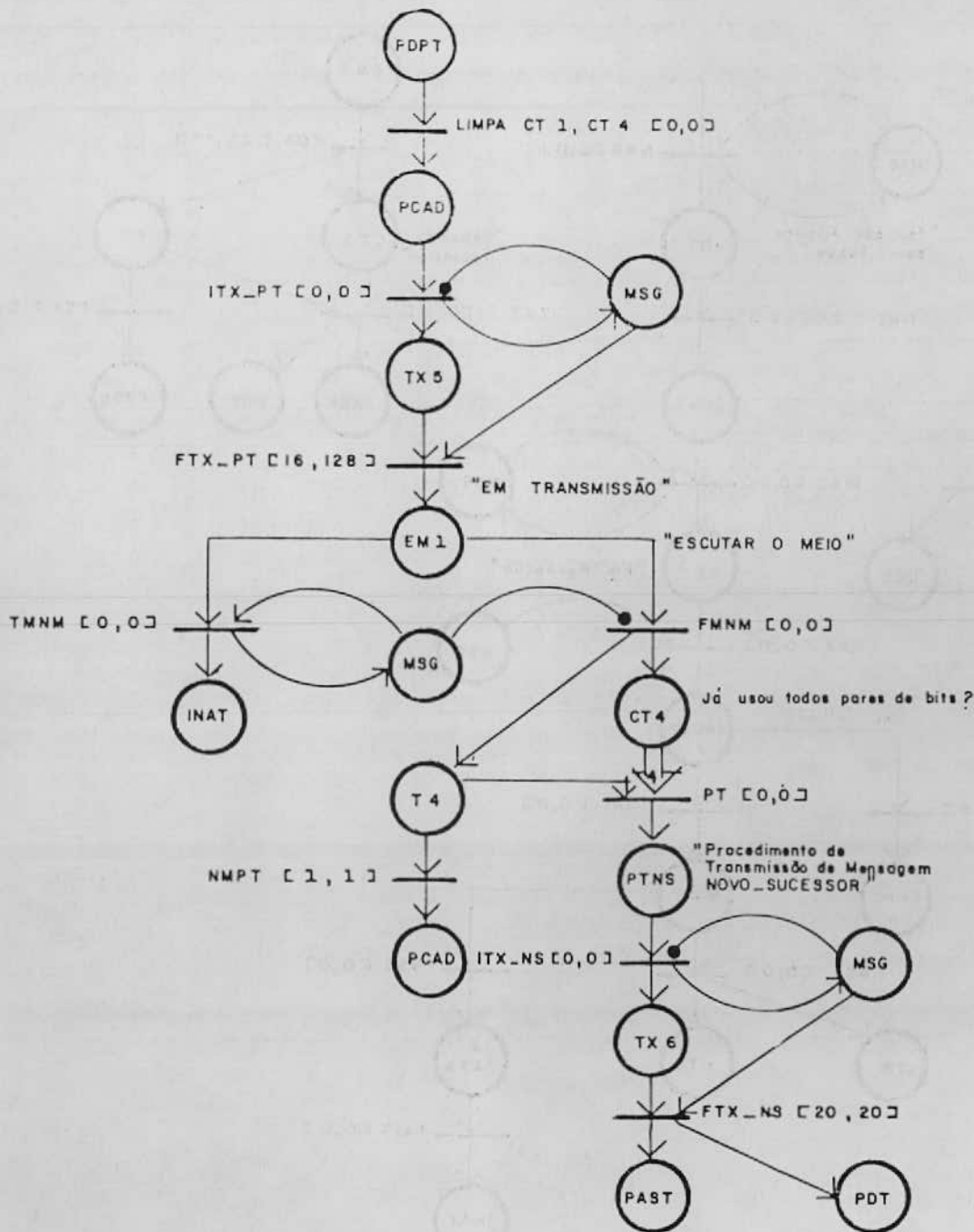


Figura 5. Subgrafo Disputa de Token

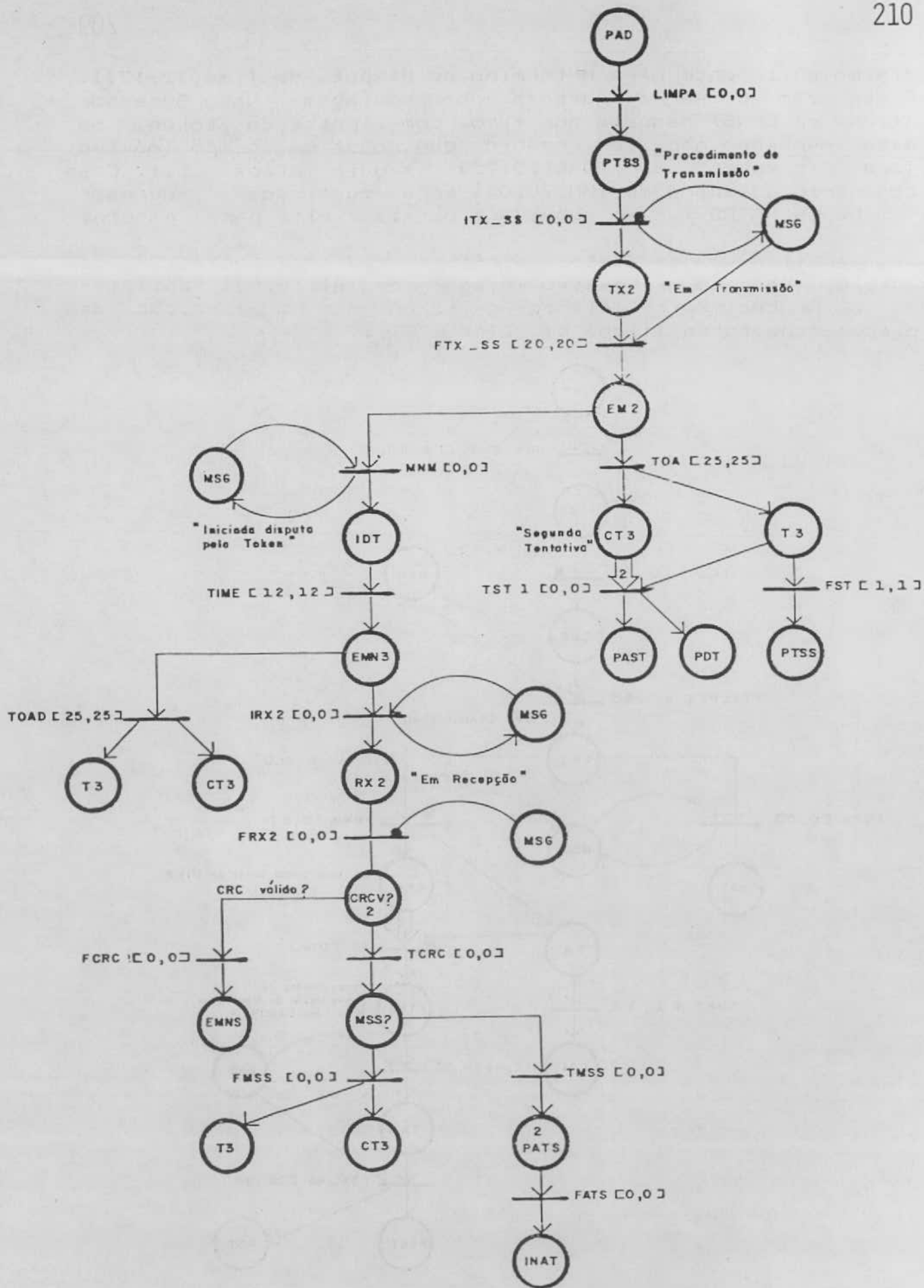


Figura 6. Subgrafo Adição

No procedimento de Recepção de Dados (Figura 7) a informação de dados recebida na mensagem é entregue ao LLC (Logic Link Control), através do procedimento de transferência para o LLC (ficha em PTLLC).

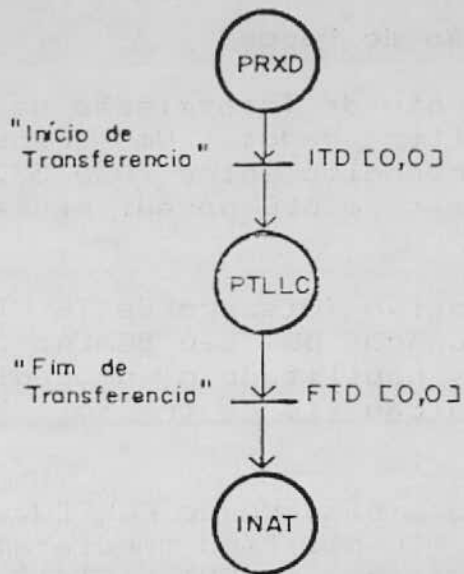


Figura 7. Subgrafo Recepção de Dados

3.6 Subgrafo Deleção

No procedimento de Deleção (Figura 8) a estação será retirada do anel. Inicialmente um mensagem Novo_Sucessor (disparo de ITX-NS[0,0]) é transmitida para a estação predecessora. Em seguida é habilitado o procedimento detentor de token com a condição passar token (ficha em PDT e PAST).

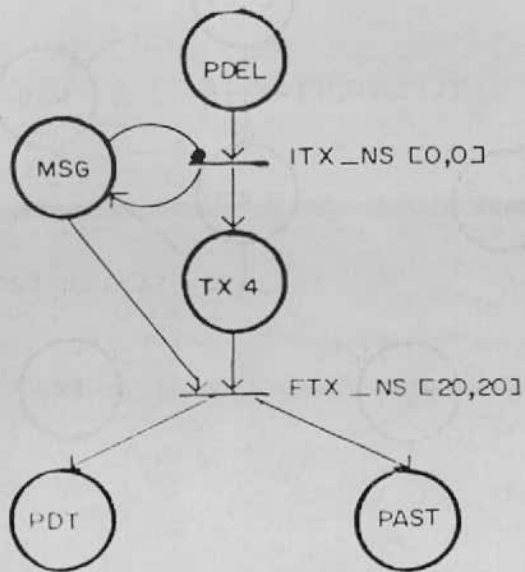


Figura 8. Subgrafo Deleção

3.7 Subgrafo Transmissão de Dados

Durante o procedimento de Transmissão de Dados (Figura 9) a estação poderá transmitir dados. Um transmissão consome um intervalo de tempo compreendido entre 20 a 532 unidades de tempo. Inicialmente verifica-se se o LLC possui mensagem para transmitir (ficha em LLC?).

i- Em caso afirmativo (disparo de TLLC[0,0]) será iniciada a transmissão. Quando do seu término (disparo de FTX-LLC[20,532]) será habilitado o procedimento detentor de token com a condição fim de transmissão (ficha em PDT e FTX).

ii- Caso contrário (disparo de FLLC[0,0]) o procedimento detentor de token é habilitado imediatamente com a condição fim de transmissão (ficha em PDT e FTX).

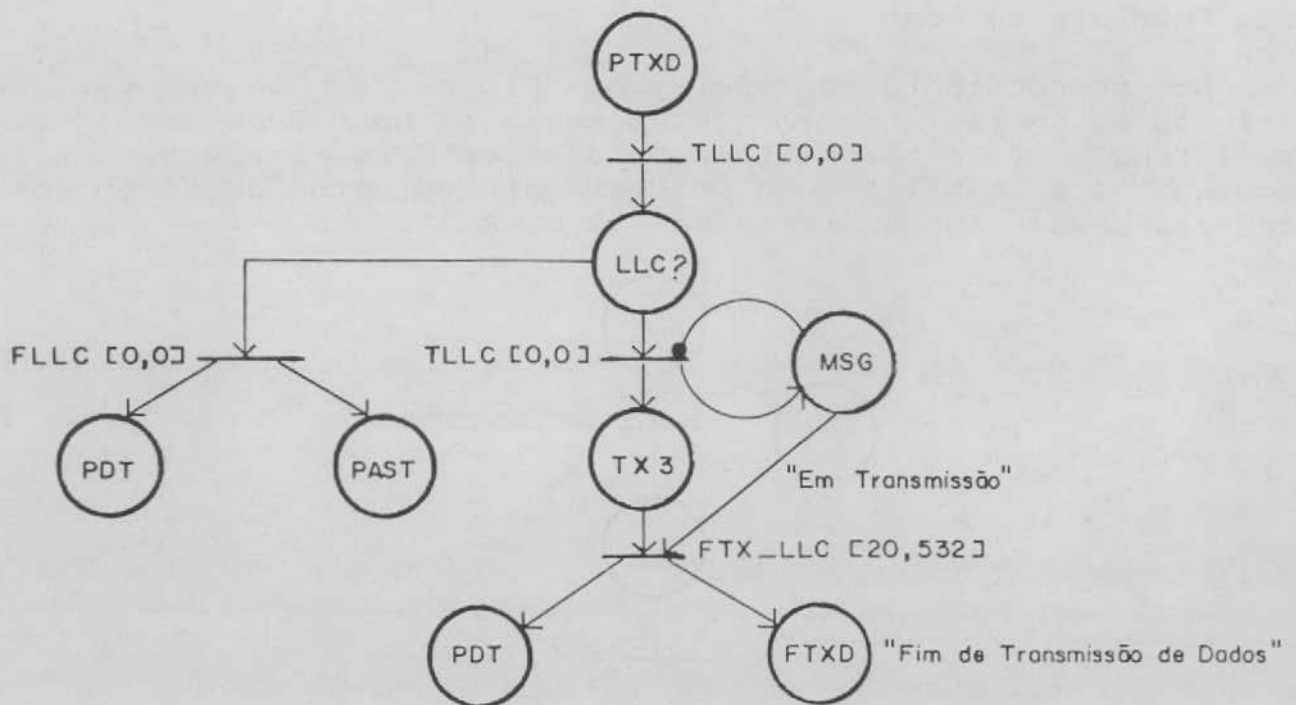


Figura 9. Subgrafo Transmissão de Dados

3.8 Subgrafo Passagem de Token

O procedimento de Passagem de Token (Figura 10) é iniciado com o envio de uma mensagem de token para a estação sucessora (ficha em PTK). Em seguida a Barra é escutada (ficha em EM4) para se ter certeza que ela está ativa, e inicia-se a seguinte sequência de eventos:

i- Se a estação sucessora estiver ativa, uma mensagem válida será escutada (disparo de TCRC4[0,0]). Assim a estação que enviou o token pode passar para o modo escuta.

ii- Se esgotado o time-out T01 não for escutada uma mensagem válida (disparo de T01[25,25]), será enviado mais uma vez o mensagem de token

iii- Após dois insucessos (disparo de TST[0,0]), é assumido que a estação sucessora falhou, sendo então enviado uma mensagem Quem_Segue (ficha em PT05) perguntando pela identidade da estação que segue aquela que falhou. Deverá ser recebida uma mensagem Novo_Sucessor enviado pela segunda estação na sequência. Em caso de recebimento (disparo de TNS[0,0]) é feita a atualização do sucessor e passa-se para escuta. Se esgotado time-out T03 (disparo de T03[25,25]) não for recebida uma resposta válida tenta-se mais uma vez (ficha em PT05).

iv- Após duas tentativas sem sucesso (disparo de TST[0,0]), a estação desiste da tática Quem_Segue e passa para a escuta o que provoca a inicialização do anel, uma vez que a barra ficará inativa.

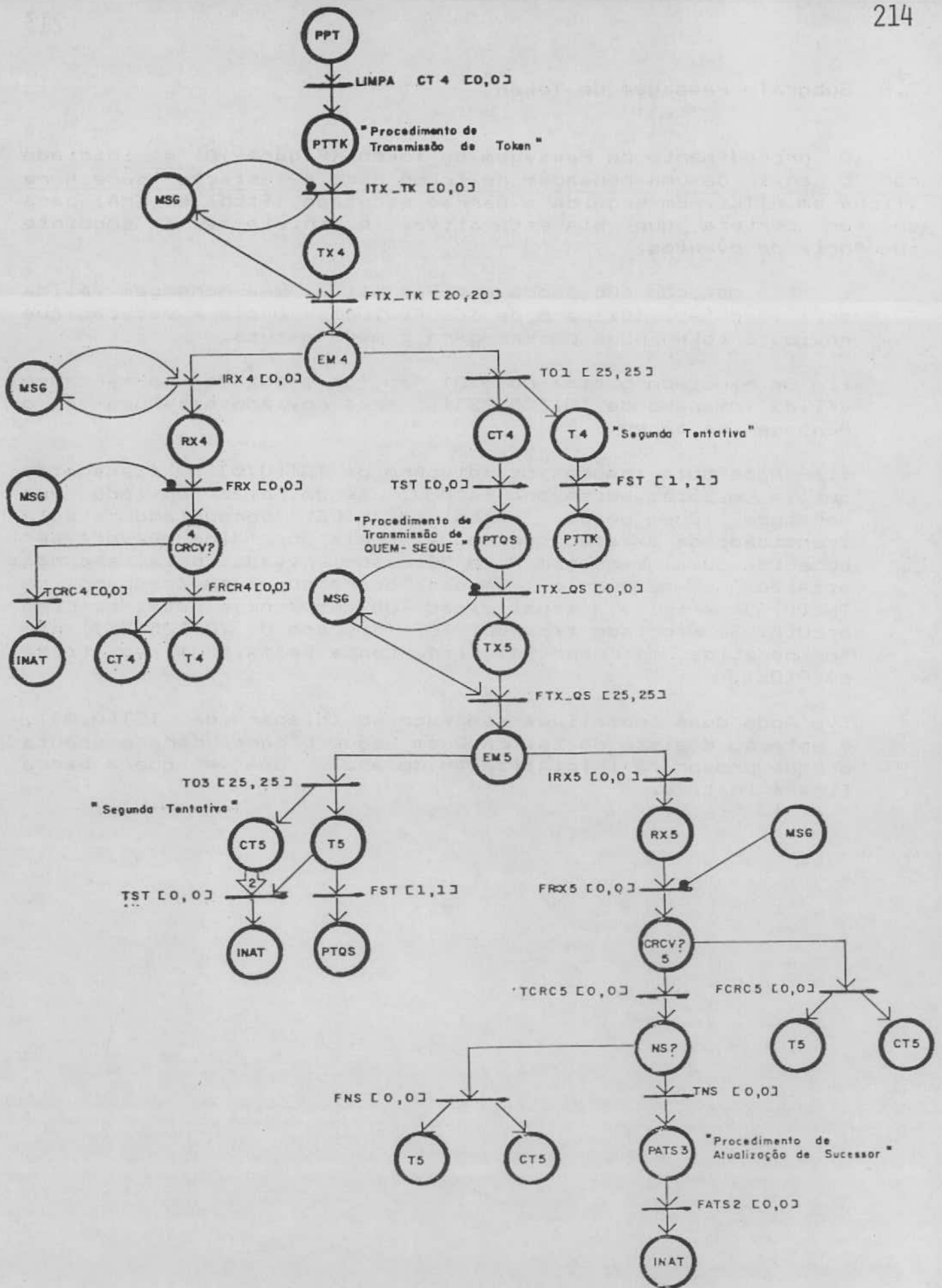


Figura 10. Subgrafo Passagem de Token

4 Conclusões

A partir do método de acesso por passagem de token "Token Bus" definido no padrão IEEE 802.4, desenvolvemos um protocolo de acesso (DIMAC) para a rede local do Departamento de Informática da UFFe onde foram introduzidas modificações visando maior simplicidade em alguns aspectos e maior eficiência em outros. O protocolo de acesso DIMAC foi especificado formalmente neste trabalho, utilizando-se Redes de Petri estendidas com parâmetros temporais e arcos inibidores.

A utilização de um procedimento único para as funções de Adição e Inicialização torna mais simples a implementação do protocolo, perdendo no entanto um pouco em eficiência em relação ao "Token Bus". Isto se deve ao fato do procedimento "Resolve Contention" utilizado no "Token Bus" ter uma duração que depende do número de estações que disputam o Token. Por outro lado o DIMAC evita transmissões desnecessárias de Token, como por exemplo no Procedimento de Adição onde a estação que se insere passa ser automaticamente a estação detentora do token.

Na continuação de nossos trabalhos ligados a este projeto, pretendemos elaborar uma metodologia para programação dos protocolos baseada em especificações formais usando Redes de Petri estruturadas. Nesta direção, o protocolo foi desenvolvido através da identificação de suas diversas funções (módulos) e recursos, e da definição das interfaces entre estes módulos e recursos.

Referências

- [1] Myers, W.: Toward a Local Network Standard; IEEE Micro, Vol. 2, no. 3, Agosto 1982.
- [2] Stalling, W.: Local Networks : An introduction; Mc Millan, 1984.
- [3] Peterson, J. L : Petri Nets. ; Computing Surveys, vol 9 no. 3, pp 223-252, setembro 1977.
- [4] Peterson, J. L : Petri Nets Theory and The Modelling of Systems; Prentice Hall, 1981.
- [5] Martins, C. O. : Redes de Petri e Análise de Sistemas Concorrentes; Anais do IX Semish- Ouro Preto 1982. pp 231-241.
- [6] Merlin ,P.: A study of the Recoverability of Computer Systems; Ph.D. Thesis , University of California, Irvine, 1974.
- [7] Menasche, M.: Introdução à Especificação e Verificação Formal de Protocolos de Comunicação Dependentes do Tempo por Redes de Petri com Temporização; Proc. XI SEMISH 1984, pp 31-44.
- [8] Agerwala, T. and Flynn, M. : Comments on Capabilities-Limitations and Correctness of Petri Nets ; Proceedings of the 1st Annual Symposium on Computer Architecture, dezembro de 1974.
- [9] Bux, W.: Performance Issues in Local Area Networks; IBM Systems Journal, vol 23, no. 24 ,1984.