

EXECUTIVO PARA IMPLANTAÇÃO DE REDES EM AMBIENTE MS-DOS

Joberto S. B. Martins
J. H. F. Cavalcanti

Grupo de Redes de Computadores
DEE - CCT
Campina Grande - PB

RESUMO

Devido as características de hardware e software dos microcomputadores tipo IBM PC, diversas instituições estão desenvolvendo microcomputadores compatíveis com estes, bem como redes locais baseadas nestes microcomputadores. Geralmente, os softwares desenvolvidos para esses microcomputadores rodam sob o sistema operacional MS-DOS ou equivalente, que é um sistema mono-usuário, não adequado para essa aplicação. Neste trabalho, apresentamos o projeto e implementação parcial de um executivo em tempo real que tornará possível a implantação de redes de computadores para máquinas compatíveis aos IBM PC's, rodando o sistema operacional MS-DOS, ou equivalente.

1-Introdução.

A partir do lançamento de microcomputadores pessoais pela IBM (IBM PC), no início dos anos 80, microcomputadores pessoais estão sendo empregados massivamente em aplicações científicas e comerciais. Isto se deve ao relativo baixo custo e boa capacidade de processamento, bem como disponibilidade de software desses microcomputadores. Com a difusão de IBM PC's e compatíveis, foram desenvolvidas, em outros países, diversas redes locais para esses microcomputadores [HEA-84]. No Brasil, com a nossa indústria de computação construindo microcomputadores compatíveis com IBM PC, a necessidade do desenvolvimento das nossas próprias redes locais se coloca normalmente.

O sistema operacional para PC's, sendo normalmente mono-usuário, não é propício para aplicações do tipo rede que necessita execução de multitarefas com prioridade e mecanismo de temporização. A solução geralmente adotada é o desenvolvimento de uma extensão, geralmente chamada de executivo em tempo real, para permitir a programação multitarefa e uso de prioridade e temporização.

Neste trabalho, apresentamos o projeto e implementação parcial de um executivo em tempo real, denominado EXECNET, para aplicação em redes locais de microcomputadores. Consideramos as seguintes características básicas para a arquitetura do EXECNET:

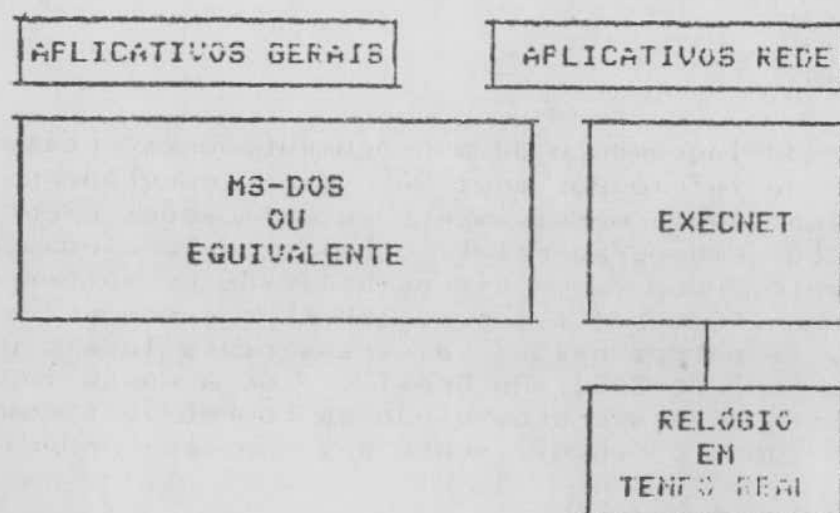
- a) transparência em relação aos programas aplicativos;
- b) acesso ao EXECNET a partir de programação em alto nível ou "assembly";
- c) tarefas "preemptivas" com mecanismos de prioridade [GON-77];
- d) primitivas de sincronização [HEA-77] e temporização direcionadas para aplicações tipo rede.

a) implantação do executivo em linguagem C e em linguagem montadora ("assembly") onde a questão do desempenho é crítica (núcleo do executivo).

No parágrafo 2 mostraremos as características básicas do EXECNET. No parágrafo 3 apresentaremos as primitivas desenvolvidas e suas representações na linguagem C. No parágrafo 4 faremos uma descrição do funcionamento do EXECNET. No parágrafo 5 apresentaremos comentários e conclusões sobre a implementação do EXECNET.

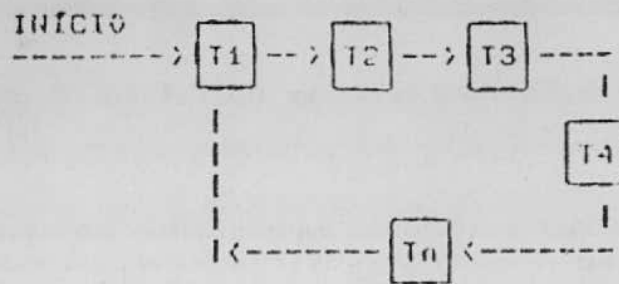
2- Características básicas do EXECNET.

A figura 1 mostra o esquema geral da implantação do EXECNET no MS-DOS com transparência para os aplicativos já disponíveis a este sistema operacional.



0) EXECNET e o Sistema Operacional
figura 1

Geralmente, executivos em tempo real possuem um núcleo para atualização e acionamento das tarefas. No EXECNET (figura 2), cada tarefa é representada por um "descriptor" com todas as informações sobre o seu estado. Os descriptors são ligados em anel, onde a posição no anel determina a prioridade (a primeira do anel tem a maior prioridade). Para garantir um melhor desempenho do executivo, que é um aspecto importante para as suas aplicações, o núcleo do escalonador foi implantado em linguagem "assembly" e todas as primitivas de sincronização, temporização e auxiliares são implantadas em linguagem C.



Onde prioridade de $T1 > T2 > T3 \dots > Tn$

Representação das tarefas em anel do EXECNET

figura 2

Os descritores das tarefas são objetos do tipo estrutura na linguagem C. Os descritores possuem os seguintes campos:

- id-identificador da tarefa;
- apont-apontador de pilha;
- ender-apontador da tarefa (código);
- st-estado da tarefa (executando, pronta para executar e bloqueada);
- freq-frequência de execução (especificada em número de interrupções do relógio);
- conta-contador das interrupções;
- liga-apontador para a próxima tarefa no anel.

Todos os apontadores utilizados endereçam um mesmo segmento de memória de 64 kbytes.

3-Primitivas do EXECNET.

As primitivas do EXECNET são na realidade um conjunto de funções escritas na linguagem C que manipulam os descritores das tarefas apresentados anteriormente. A parametrização dessas funções permite a criação de uma biblioteca de funções que pode ser posta à disposição para um implantador de um pacote de redes. Evidentemente, o acesso direto as primitivas listadas abaixo é sempre possível. As funções, ou primitivas básicas são apresentadas a seguir:

a)cria(id)

O usuário adiciona tarefas para controle pelo EXECNET. A tarefa é posta no final do anel (menor prioridade). O identificador (id) é definido pelo usuário.

b)destroi(id)

O usuário retira a tarefa com identificador "id" do anel de descritores. O anel é reordenado.

c)senal(id)

O usuário aciona a tarefa com identificador "id" no anel. A tarefa passará do estado "bloqueada" para o estado "pronta para

executar".

d(esperadid,tempo,estado)

Usada para sincronização das tarefas do EXECNET, onde:

id: identificador da tarefa;

tempo: temporização em unidades de ciclos de interrupções do relógio;

estado: especifica o estado (campo st do descritor) que deverá ficar a tarefa após a execução da função.

4-Funcionamento do EXECNET.

O funcionamento do EXECNET exige a utilização de uma linha de interrupção para o mecanismo de temporização, correspondente a um relógio em tempo real, para interromper a UCF periodicamente.

O núcleo do EXECNET é residente e, para tal, faz-se uso das facilidades do sistema operacional em questão. Juntamente com o núcleo, existe pelo menos uma tarefa (tarefa T1) encarregada da gestão da base de tempo. Assim sendo, após o carregamento e acionamento do EXECNET, existirá a tarefa relógio (tarefa T1), permitindo ao usuário a geração de novas tarefas (usando a função "cria"). No seu funcionamento em regime, a ocorrência de uma interrupção do seu relógio, causa a seguinte sequência de ações:

- a)atualização dos descritores das tarefas criadas (contagem dos ciclos do relógio);
- b)acionamento da tarefa de maior prioridade;
- c)retorno de interrupção.

5-Conclusões.

O EXECNET apresentado é uma solução simples que coloca, à disposição do conceitor de software de redes, um conjunto de primitivas e mecanismos imprescindíveis, que são disponíveis em S.O. multi-usuário e multitarefa. Ele é portanto uma ferramenta de suporte a programação que, de maneira transparente ao S.O., incorpora facilidades com um mínimo de código de máquina residente.

O núcleo do EXECNET foi implementado em "assembly" do microprocessador 8088 (212 bytes) num microcomputador do tipo IBM PC. Ele foi testado com várias tarefas usando uma simulação do relógio em tempo real.

6-Bibliografia.

[HAU-84]-Haugdahl, J. S., "Local-area networks for the IBM PC", revista BYTE, dezembro de 1984, pg.147-174.

[GON-77]-Gonzalez, M. J., "Deterministic processor scheduling", ACM Comp. Surveys, 1977, vol. 9-N3, pg.173-204.

[HEA-84]-Heath, W. S. "A system executive for real-time microcomputer programs", IEEE-MICRO, vol.4 número 3, junho de 1984, pg.20-32.