

ARQUITETURA DE UM MICROCOMPUTADOR
PARA CONTROLE DE PROCESSOS
CONCORRENTES EM TEMPO REAL

concurrent processes requires a more complex microcomputer
and the ones usually found in other kinds of applications. The
basic microcomputer was designed to allow a good throughput, an
adaptation to asynchronous events and an optimized data flow.

Graduação em Engenharia de Eletrônica (UNICAMP, 1979); arquitetura de microcomputadores, aplicações
em tempo real, controle de processos; P&D Systems Distribuição S/A.
Rua H. Domingos Cazotti, 176 - Jardim Santa Helena - Campinas, SP -

Graduação em Engenharia de Eletrônica (ITA, 1967); Mestrado em Engenharia de Eletrônica (UNICAMP, 1974); Ph.D. in Computer Science (University of Illinois, 1977);
arquitetura de software, arquitetura de processos industriais, controle de processos.
Rua H. Domingos Cazotti, 176 - Jardim Santa Helena - Campinas, SP - 13080.

ARQUITETURA DE UM MICROCOMPUTADOR PARA CONTROLE DE PROCESSOS CONCORRENTES EM TEMPO REAL

P. L. GEUS * , M. JINO **

SUMÁRIO

O controle de processos concorrentes em tempo real requer uma arquitetura de microcomputador mais elaborada do que a usualmente utilizada em outros tipos de aplicações. A arquitetura de um microcomputador básico foi modificada para permitir uma boa taxa de processamento, um tempo de resposta adequado para eventos assíncronos e uma otimização do fluxo de dados.

ABSTRACT

Real time control of concurrent processes requires a more complex microcomputer architecture than the ones usually found in other kinds of applications. The architecture of a basic microcomputer was improved to allow a good throughput, an adequate response time to asynchronous events and an optimized data flow.

*Engenheiro de eletrônica (UNICAMP, 1979); arquitetura de microcomputadores, aplicações de microprocessadores, controle de processos; P&D Sistemas Eletrônicos S/A, Desenvolvimento, R. Domingos Cazotti, 176 - Jardim Santa Genebra - Campinas, SP - 13100.

**Engenheiro de eletrônica (ITA, 1967); Mestrado em Engenharia Elétrica (UNICAMP, 1974); Ph.D. in Computer Science (University of Illinois, 1978); bancos de dados, engenharia de software, automação de processos industriais; DEE/FEC/UNICAMP, c.p. 6166 - Cid. Univ. Zeferino Vaz - Campinas - SP - 13100.

1. INTRODUÇÃO

Este trabalho, desenvolvido junto à P&D Sistemas Eletrônicos S/A de Campinas, apresenta um conjunto de aperfeiçoamentos acrescentados ao hardware de um microcomputador a fim de torná-lo capaz de realizar processamentos mais complexos, requisitados por uma aplicação específica. A arquitetura resultante pode ser utilizada em outras aplicações mais genéricas em controle de processos, em unidades concentradoras, ou em unidades de processamento multi-usuário.

A idéia básica é permitir à CPU uma boa taxa de processamento, atenuando os efeitos negativos de um grande número de interfaces de entrada/saída que se necessite. Ainda dentro deste conceito, a degradação no desempenho gerada pelo tratamento de memória física maior que o espaço lógico da CPU também deve ser atenuada. Transferências via DMA são utilizadas ao máximo, otimizando o fluxo de dados com o exterior e liberando a CPU para o tratamento de outras tarefas.

Naturalmente é necessário um sistema operacional multi-tarefas para poder fazer uso destas características; de outra forma elas não melhorariam significativamente a capacidade de processamento do sistema. O projeto citado na seção seguinte utiliza este microcomputador em dois níveis hierárquicos de controle, com diferentes necessidades de processamento, mas devido à sua modularidade basta reconfigurar as placas do sistema para se obter o controlador desejado.

A seção 2 relata um histórico do projeto, apresentando o equipamento do qual faz parte o microcomputador objeto do trabalho e situando o problema cujas soluções serão apresentadas na seção 3.

A seção 3 inicia-se por uma breve descrição do barramento padronizado do microcomputador, discutindo a seguir as diversas modificações de arquitetura e operação introduzidas.

A seção 4 apresenta alguns dados de carga e desempenho do sistema, tanto quanto é possível até o momento (apenas algumas unidades mais simples já estão em testes de campo).

2. HISTÓRICO [4]

O SITASU (Sistema de Taxação e Supervisão de Centrais Telefônicas) é um sistema que fornece automaticamente informações para as gerências de manutenção, de tráfego e de redes, orientando e agilizando a solução dos problemas encontrados nestes processos. Utilizando microcomputadores estruturados hierarquicamente, o SITASU prevê a utilização de três níveis: no nível superior a UCS (Unidade Centralizadora de Supervisão), no nível intermediário a UCR (Unidade de Controle Regional) e no nível inferior a USCE (Unidade de Supervisão e Controle de Estação).

As USCE's estão fisicamente conectadas aos pontos supervisionados nas centrais telefônicas, adquirindo todos os dados e efetuando um primeiro processamento, podendo emitir local ou remotamente relatórios sobre os aspectos críticos para a manutenção das cen

trais. A UCR consolida e armazena estruturadamente os dados e relatórios recebidos periodicamente das USCE's, processa e emite relatórios para a gerência de manutenção e tráfego de uma região e pode transmitir para a UCS os dados para a gerência de rede, permitindo atualização dos dados de tráfego e manutenção de rede.

As instalações realizadas anteriormente implementavam apenas este nível inferior, de interação direta com o processo, e ainda assim apenas um subconjunto dos algoritmos previstos para este nível. O número de pontos supervisionáveis era limitado a aproximadamente 10.000, devido à limitação de memória (64 kbytes) do microcomputador utilizado originariamente. O equipamento incluía o software aplicativo já desenvolvido e um hardware específico para aquisição de dados de processos típicos de centrais telefônicas (sinais digitais). Este hardware é provido de certa autonomia, fazendo a varredura sobre os pontos de aquisição automaticamente. Uma placa no microcomputador engloba esta lógica e faz a interface com as gavetas de aquisição de dados, realizando as transferências de dados via DMA. A taxa de aquisição oscila tipicamente (dependendo da central telefônica) ao redor de 100 Kbytes/s.

Do equipamento completo sendo instalado agora (três níveis) estamos implementando sistemas no nível inferior (USCE), de supervisão direta dos processos, e no nível intermediário (UCR), de concentração regional.

A época da definição do projeto ficou muito clara a necessidade de aumentar a capacidade de processamento do microcomputador até então utilizado no sistema, em vista de características adicionais requeridas e do novo tipo de processamento a ser realizado no nível intermediário.

Nosso microcomputador (semelhante ao utilizado originalmente no projeto) dispunha de uma arquitetura simples, capaz de rodar um sistema operacional tipo CP/M: CPU 8085 ou Z80, 64 Kbytes de memória RAM, 16 Kbytes de memória EPROM, 2 canais de interface série, controlador de diskette. O aprimoramento do microcomputador era extremamente facilitado pelo barramento de conexão entre suas placas, de 16 bits de dados e 20 bits de endereço (1 Mbytes), além de as placas periféricas serem compatíveis com CPU's de 8 ou 16 bits.

A seção seguinte mostrará como se conseguiu esse aprimoramento.

3. ARQUITETURA DESENVOLVIDA

Esta seção apresenta primeiramente uma breve descrição do barramento padronizado do microcomputador, relatando em seguida a definição do projeto, com as alternativas possíveis, e a descrição das modificações realizadas.

3.1. O Microcomputador Básico (SME)

O SME é um sistema microcomputador constituído por um conjunto de placas que se comunicam através de um barramento padronizado de comunicação, denominado "via SME".

O tamanho de placas adotado (160 x 160 mm) juntamente com o barramento padronizado, permitem modularidade e expansibilidade ao sistema, facilitando a implementação de no-

vas funções.

A via SME utiliza conectores de 100 pinos, dos quais 22 não pertencem ao barramento, mas são utilizados na conexão de sinais de placas do SME com dispositivos externos ao sistema e de sinais extra barramento entre placas do SME. As 100 vias do conector podem ser agrupadas nas seguintes categorias de sinais:

- 20 linhas de endereço
- 16 linhas de dados
- 8 linhas de interrupção
- 14 linhas de controle
- 20 linhas de alimentação
- 22 linhas de uso geral

A via SME é um barramento ²² síncrono apto a realizar transferências de dados a uma taxa de 3 Mwords/s.

3.2. Definição do Projeto [4]

Três caminhos possíveis poderiam ser seguidos para aumentarmos o desempenho do sistema:

Arquitetura multi-processada

Com várias CPU's dedicadas a tarefas específicas a carga de processamento por CPU diminui bastante, aumentando a capacidade do sistema como um todo. Requisitos básicos para esta arquitetura seriam um barramento de comunicação de alta velocidade, além de placas com CPU's autônomas (memória e I/O incorporados). Nosso barramento teria que ser modificado para um bom gerenciamento de vários processadores (controladores de barramento); além disso, o tamanho de placas até agora utilizado, que fornece grande flexibilidade ao sistema, torna difícil a colocação de uma CPU autônoma. Em outras palavras, teria de ser projetado todo um sistema novo.

CPU de 16 bits

Uma CPU mais poderosa poderia cumprir os requisitos de processamento, mas o software aplicativo existente para o processo a ser supervisionado teria que ser reescrito e otimizado na linguagem de máquina da nova CPU.

Arquitetura Existente Reformulada

Surgiu como a hipótese mais viável por permitir a utilização do software aplicativo já desenvolvido e do hardware de barramento existente. Foi realizada uma redefinição de algumas linhas do barramento não utilizadas até então de forma a implementar uma cadeia de prioridades para interrupções vetorizadas no barramento. As outras modificações introduzidas no sistema se realizaram a nível de placas, introduzindo maiores capacidades,

desempenho e flexibilidade. As placas que sofreram modificações foram: a CPU, com a implementação de uma unidade de gerenciamento de memória e de um esquema de interrupções vetorizadas pelo barramento; a placa controladora de DMA, com a ampliação de 4 para 11 canais independentes e a ampliação da capacidade de endereçamento para o espaço total do microcomputador; a placa de interfaces série, com a ampliação de 2 para 4 canais, capacidade de vetorizar interrupções e interface para DMA disponível para os quatro canais. (figura 1).

3.3. Características Acrescentadas ao Microcomputador Básico

A CPU Z80 endereça diretamente apenas 64 Kbytes de memória, e as necessidades típicas nos sistemas em implantação no momento estão se situando na faixa de 192 - 256 Kbytes. O esquema de gerenciamento de memória deveria então ser suficientemente rápido e não provocar perdas significativas no desempenho, para que o sistema operacional multi-tarefas suportasse o gerenciamento da aquisição de dados em tempo real, dos algoritmos de processamento e das tarefas de comunicação. Manter a CPU fazendo "polling" em interfaces de comunicação seria inviável; o esquema de interrupções vetorizadas, somado aos "chips" de interface série Z80 - SIO fazendo pleno uso destas características, liberariam quase completamente a CPU para outros processamentos; habilitando todos os canais de interface série a utilizarem transferência via DMA, a CPU não teria de dedicar tempo às frequentes rotinas de interrupção para tratamento de I/O, característica particularmente vantajosa na comunicação de grandes blocos de informação; para suportar todos os dispositivos em condições de utilizá-la (controlador de diskette, aquisição de dados do processo, congelamento de memória, interfaces série, possivelmente em grande número), a placa controladora de DMA não poderia ser restrita a apenas os 4 canais do chip 8237, e nem poderia ser muito útil endereçando apenas 64 Kbytes. Foram implementados 11 canais, um dos quais com capacidade de transferências memória a memória, todos eles com registros de segmentação para o endereçamento completo de 1 Mbytes.

3.4. MMU - Unidade de Gerenciamento de Memória [1], [5], [6], [7]

O esquema adotado não utiliza registros para seleção de bancos, cujo mecanismo causaria grande "overhead" à CPU, mas sim implementa tabelas de paginação, sem provocar atrasos nos ciclos de acesso à memória e garantindo o desempenho nominal da CPU. O circuito mantém presentes 64 tabelas de paginação, cada uma com 16 entradas que mapeiam páginas lógicas de 4 Kbytes da CPU Z80 em qualquer região do espaço físico de 1 Mbytes do SME. Devem ser respeitadas as restrições de módulo das páginas, i.e., o endereço inicial das páginas (lógicas ou físicas) deve ser um múltiplo de 4.096 (decimal). A mudança de um contexto (tabela) é extremamente rápida (uma instrução apenas, 2,7 μ s), agilizando bastante o gerenciamento multi-tarefas. A figura 2 traz um diagrama do mecanismo.

Funcionamento do Circuito

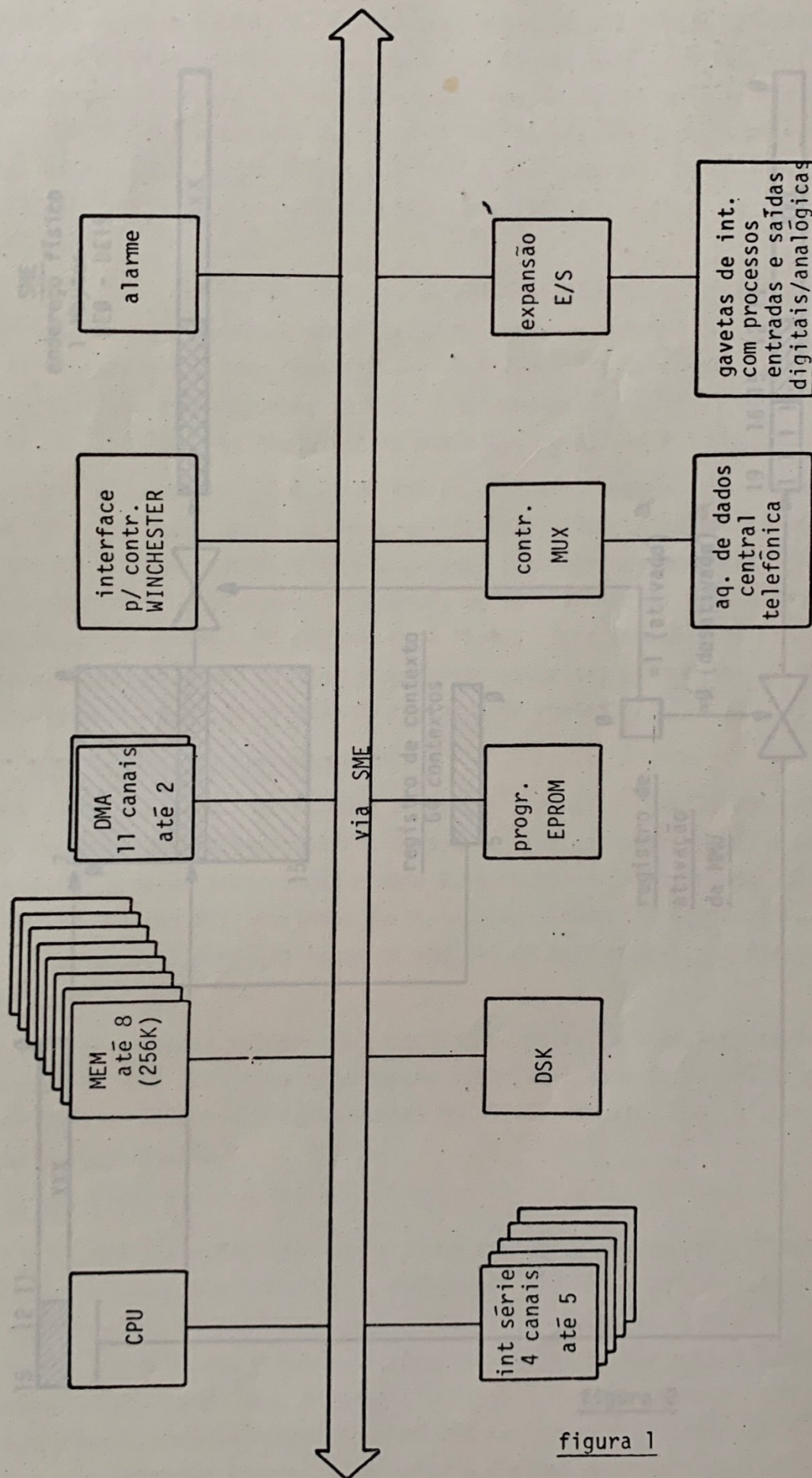


figura 1

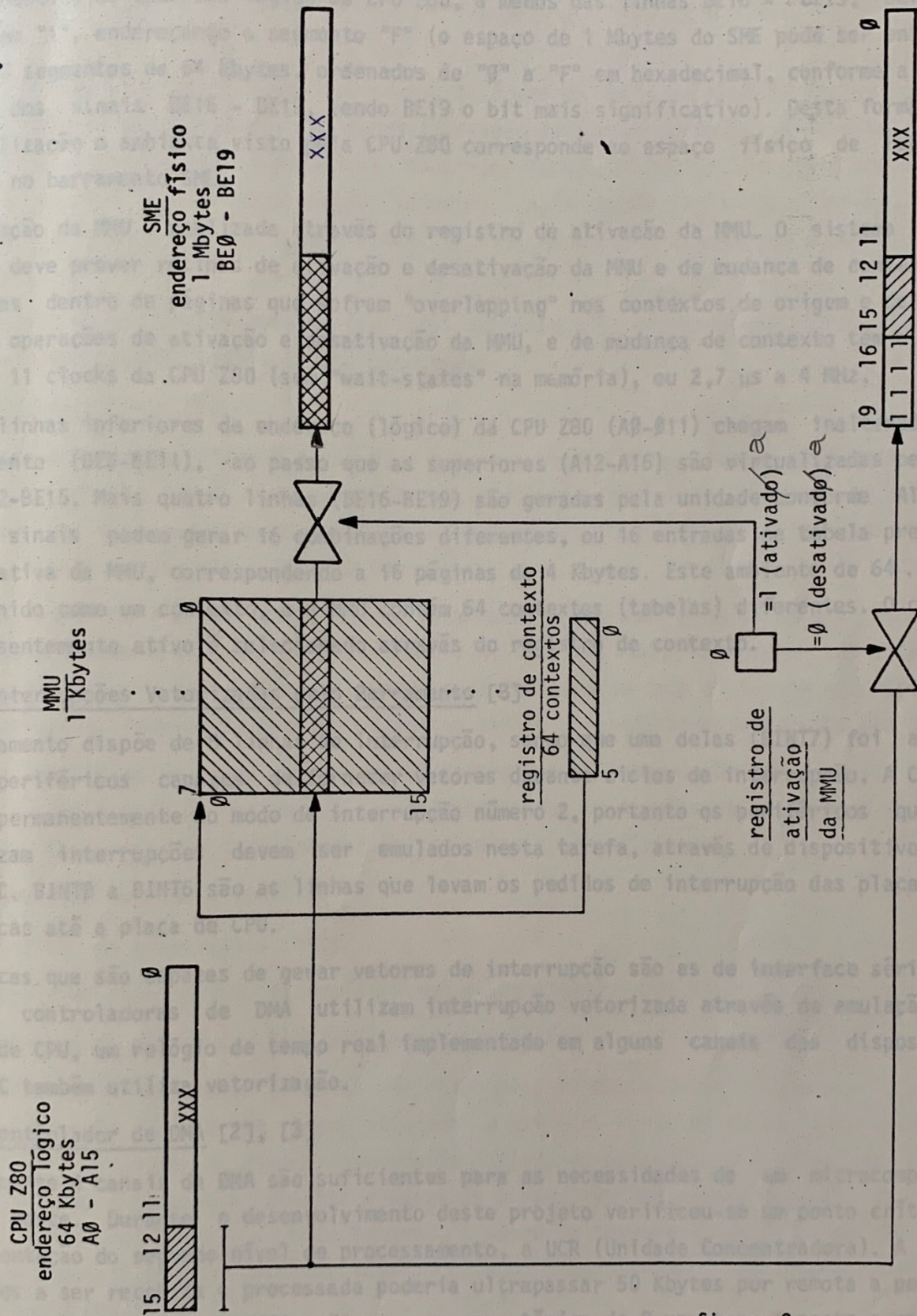


figura 2

Após a inicialização do sistema a MMU está desativada, e o endereço físico no barramento será idêntico ao endereço lógico da CPU Z80, a menos das linhas BE16 - BE19, definidas todas em "1", endereçando o segmento "F" (o espaço de 1 Mbytes do SME pode ser entendido como 16 segmentos de 64 Kbytes, ordenados de "0" a "F" em hexadecimal, conforme a combinação dos sinais BE16 - BE19, sendo BE19 o bit mais significativo). Desta forma, após inicialização o ambiente visto pela CPU Z80 corresponde ao espaço físico de F0000H a FFFFFH no barramento SME.

A ativação da MMU é realizada através do registro de ativação da MMU. O sistema operacional deve prover rotinas de ativação e desativação da MMU e de mudança de contexto residentes dentro de páginas que sofram "overlapping" nos contextos de origem e de destino. As operações de ativação e desativação da MMU, e de mudança de contexto têm a duração de 11 clocks da CPU Z80 (sem "wait-states" na memória), ou 2,7 μ s a 4 MHz.

As 12 linhas inferiores de endereço (lógico) da CPU Z80 (A0-A11) chegam inalteradas ao barramento (BE0-BE11), ao passo que as superiores (A12-A15) são virtualizadas pela MMU em BE12-BE15. Mais quatro linhas (BE16-BE19) são geradas pela unidade conforme A12-A15. Estes sinais podem gerar 16 combinações diferentes, ou 16 entradas na tabela presente-mente ativa da MMU, correspondendo a 16 páginas de 4 Kbytes. Este ambiente de 64 Kbytes é definido como um contexto, e a MMU contém 64 contextos (tabelas) diferentes. O contexto presentemente ativo é selecionado através do registro de contexto.

3.5. Interrupções Vetorizadas pelo Barramento [8]

O barramento dispõe de 8 linhas de interrupção, sendo que uma delas (BINT7) foi alocada para periféricos capazes de fornecer vetores durante ciclos de interrupção. A CPU Z80 opera permanentemente no modo de interrupção número 2, portanto os periféricos que não vetorizam interrupções devem ser emulados nesta tarefa, através de dispositivos tipo Z80-CTC. BINT0 a BINT6 são as linhas que levam os pedidos de interrupção das placas periféricas até a placa de CPU.

As placas que são capazes de gerar vetores de interrupção são as de interface série (as placas controladoras de DMA utilizam interrupção vetorizada através de emulação). Na placa de CPU, um relógio de tempo real implementado em alguns canais dos dispositivos Z80-CTC também utiliza vetorização.

3.6. Controlador de DMA [2], [3]

Normalmente 4 canais de DMA são suficientes para as necessidades de um microcomputador deste porte. Durante o desenvolvimento deste projeto verificou-se um ponto crítico na implementação do segundo nível de processamento, a UCR (Unidade Concentradora). A massa de dados a ser recebida e processada poderia ultrapassar 50 Kbytes por remota a períodos de 20 a 30 minutos, significando que, no caso típico de 8 remotas, somente a recepção destes dados por interrupção poderia tomar cerca de 80% do tempo disponível da CPU. A solução adotada para esta situação consistiu em dotar o sistema de canais de DMA su

ficientes (11 por placa, máximo de 22) para as interfaces envolvidas com grandes massas de dados, além dos periféricos que normalmente utilizam DMA, como disco flexível e disco rígido.

Outra característica adicionada foi a possibilidade de realizar transferências do tipo memória à memória, poupando a CPU na movimentação de blocos de dados. Além disso, o dispositivo utilizado tem capacidade de endereçar apenas 64 Kbytes, tornando necessária a inclusão de registros de segmentação. Todos os canais mantêm registros próprios, implementados através de memória super-rápida (45 ns), mantendo o desempenho original do dispositivo.

Um circuito de arbitragem resolve conflitos entre os três dispositivos LSI utilizados e a outra placa de DMA, possivelmente conectada.

3.7. Interface Série [8]

O LSI empregado nesta placa é o Z80-SIO, que se ambienta otimamente no esquema de interrupções da CPU Z80. Cada CI dispõe de 2 canais de interface série, operando em modo assíncrono ou síncrono, implementando os protocolos mais comuns orientados por byte (BSC-1, BSC-2) ou por bit (SDLC, HDLC).

A placa dispõe de 4 canais (2 CI's) com interfaces preparadas para terminais de vídeo, impressoras, modems síncronos e assíncronos. A interface com o barramento habilita o envio à placa de CPU dos vetores de interrupção internos durante ciclos de reconhecimento de interrupção. Pode-se também ter os quatro canais operando simultaneamente com DMA.

Os CI's Z80-SIO podem gerar até 4 vetores diferentes de interrupção por canal, facilitando e agilizando as rotinas de serviço associadas às tarefas de tratamento de entrada/saída. Esses vetores correspondem às seguintes ocorrências:

- caracter recebido disponível
- "buffer" de transmissão vazio
- condições especiais de recepção (erros e anomalias)
- mudanças de estado das linhas externas (controle)

As transferências de blocos de dados são otimizadas fazendo-se uma junção das operações via interrupção e via DMA. Exemplificando: o primeiro caracter de um bloco de dados é recebido via interrupção, assincronamente às tarefas em execução; a rotina de serviço associada deve armazenar o caracter e preparar o canal de DMA para a recepção do restante do bloco de dados. Com este esquema, a perda do desempenho da CPU fica reduzida a aproximadamente 1,5µs por caracter (0,12% do tempo disponível no barramento por canal operando a 9.600 baud).

4. RESULTADOS E CONCLUSÕES

Presentemente temos alguns sistemas do nível hierárquico inferior (USCE) em testes de campo, e o do nível intermediário (UCR) em testes de integração.

O sistema USCE está dimensionado para supervisionar até 16 Kpontos de central telefônica, com 64 Kbytes de software e até 192 Kbytes de memória de trabalho. O sistema operacional gerencia 51 tarefas "on-line", sendo 28 para comunicação com TTY local e com o sistema UCR. Estão sendo utilizados 59 contextos (tabelas de paginação) dos 64 possíveis na unidade de gerenciamento de memória (MMU) da CPU.

O sistema UCR está dimensionado para controlar 16 sistemas USCE, utilizando diskette e disco rígido como memória de massa e 192 Kbytes de memória de trabalho. O sistema operacional gerencia 28 tarefas "on-line", sendo que 3 delas são de apoio ao executivo, uma delas é um sistema compatível com CP/M e o restante tarefas aplicativos. Utiliza 15 contextos da MMU.

O microcomputador com as características adotadas suporta as implementações descritas acima, conseguindo realizar as tarefas pertinentes em tempo real. Até o momento não foi possível coletar dados sobre a porcentagem de tempo em que a CPU permanece inativa no núcleo dos executivos, mas os dois tipos de sistemas já estão rodando tarefas de teste representando carga semelhante à situação real.

(5) MILLER, J. Microprocessor manager: virtual memory for handling large programs.

Byte, 12(12):199-205, October 1981.

(6) SMITH, S. Virtual memory for microcomputers. Byte, 8(4):218-22, April 1983.

(7) SPRY, Larry W. The management of extended memory for concurrent processes.

In: T.I. microprocessor Pascal. In: Westcon/81 Electronic Show & Convention, September 15-17, 1981. San Francisco, Electronic Conventions, Inc., 1981. Section 9/1, p. 1-10.

(8) ZILG. 1982/83 Data book. Campbell, Zilog, 1981. p. 5-26, 81-100.

5. BIBLIOGRAFIA

- (1) DILLON, B. Extended addressing for the Z80. Wireless World, 88(1559):60, August 1982.
- (2) EL-KATEEB, A. Hardware switch for DMA to augment CPU efficiency. Microprocessors and Microsystems, 7(2):117-20, April 1983.
- (3) INTEL. Microprocessor and peripherals handbook. Santa Clara, INTEL, 1982. p.2-88, 2-102.
- (4) P&D SISTEMAS ELETRÔNICOS S/A. Especificações funcionais para o Sistema Telesp. Campinas, P&D Sistemas Eletrônicos, 1983. 97p. il.
- (5) PHILLIPS, D. Microprocessor manages virtual memory for handling large programs. Electronic Design, 26(22):199-205, October 1983.
- (6) SCHMITT, S. Virtual memory for microcomputers. Byte, 8(4):210-38, April 1983.
- (7) SPRY, Larry W. The management of extended memory for concurrent processes in T.I. microprocessor Pascal. In: Wescon/81 Electronic Show & Convention, September 15-17, 1981. San Francisco, Electronic Conventions, Inc., 1981. Session 9/1, p. 1-10.
- (8) ZILOG. 1982/83 Data book. Campbell, Zilog, 1981. p. 5-26, 81-108.

M I N U T A

ESTIMATIVA DA CARGA DA CPU/DMA DA USCE PARA EXECUÇÃO DE TAREFA

Seja X_i a v.a. tempo de execução da tarefa i na ativação de ordem n_i e seja X a v.a. soma dos tempos de retenção da CPU/DMA para executar o conjunto das S tarefas.

$$X = \sum_{i=1}^S k_i X_i$$

$$K_i = \frac{\tau}{\Delta t_i} \quad \begin{matrix} \tau - \text{tempo de observação} \\ \Delta t_i - \text{base de tempo da tarefa} \end{matrix}$$

$$EX = \sum_{i=1}^S k_i E X_i$$

A USCE tem oito tarefas de aquisição de dados com base de tempo menor ou igual a 50 segundos. No dia 24 ABR 85 foram realizadas medidas de tempo de duração das tarefas cujos resultados finais permitem as seguintes estimações de $E X_i$ e $E X$.

TAREFA	i	k_i	$\bar{X}_i$	$K_i \bar{X}_i$ (s)	$\bar{X}_{\text{real}}$
TAR41	1	$1,44 \times 10^5$	10,1	1454,4	10,1
TAR42	2	$1,44 \times 10^4$	8,0	115,2	8,0
TAR43	3	$3,6 \times 10^3$	23,5 - 10,1	84,6	13,4
TAR44	4	$1,8 \times 10^3$	65,1 - 20,2	117,2	44,9
TAR45	5	720	79,8 - 30,3	57,5	49,5
TAR46	6	180	163,2 - 50,5	29,4	112,7
TAR47	7	72	137,7	9,9	
TAREN	8	180	154,6	27,8	
				1811,4	

Taxa de ocupação da CPU/DMA para executar as tarefas de aquisição de dados = 50,3%

CONCLUSÃO:

A carga estimada acima foi verificada na USCE II de Campinas Centro, com 11800 pontos de supervisão. Considerando que as demais tarefas, bloqueio, tabela de horários e comunicação com a UCR devem consumir da ordem de 720 segundos por hora na situação mais crítica (relatório de resumo), a taxa de ocupação acima deve elevar-se a **70,3%**.

Considerando que a presente avaliação refere-se a uma particular USCE de 11800 pontos, considerando que parte da carga é partilhada com DMA e que a avaliação desta partilha, recomendamos limitar o número de pontos das USCE a 16384. A ultrapassagem deste limite poderá ser decidida após avaliação técnica rigorosa.

É possível que, a continuação do estudo de carga, se revele a possibilidade de elevação do limite ora recomendado.