

Marcus de Aguiar Dias

Controlador Programável a Multimicroprocessadores para
Controle Hierárquico de Robôs

9107/169/34

Dissertação apresentada à Faculdade de Engenharia Elétrica da Universidade Estadual de Campinas como parte dos requisitos exigidos para obtenção do título de MESTRE EM ENGENHARIA ELÉTRICA.

Este exemplar corresponde à redação final da tese defendida por Marcus de Aguiar Dias e aprovada pela Comissão Julgadora em 7/6/91.


Orientador

Orientador: Prof. Dr. Alvaro Geraldo Badan Palhares *ru*

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE SISTEMAS E CONTROLE DE ENERGIA

Campinas - SP
Maio 1991

UNICAMP
BIBLIOTECA CENTRAL

A G R A D E C I M E N T O S

Parece incrível a capacidade que a maioria dos homens tem de subestimar o trabalho necessário para a conclusão de tarefas com certo grau de complexidade. Com certeza a elaboração e conclusão de um trabalho de tese é uma tarefa que apresenta um grau de dificuldade muito maior do que se imagina inicialmente, sendo assim muitas são as tentações para um término prematuro sem alcançar os objetivos propostos.

Somente com muita determinação e com ajuda de pessoas interessadas e eficientes foi possível o término deste trabalho, pessoas estas que não poderia deixar de homenagear, pois foram de fundamental importância para a conclusão do Mestrado.

Gostaria de prestar uma homenagem especial ao amigo e professor Doutor Álvaro Geraldo Badan Palhares, orientador de programa e orientador da tese, pelo excelente acompanhamento, suporte e ótimo ambiente de trabalho.

Aos meus grandes amigos e colegas de laboratório, Marconi Kolm Madrid e Adilson Sakahi Ohfugi, pelo companheirismo, pelo trabalho em equipe, e pela grande ajuda em todas as fases do projeto.

A Kenia Pena de Souza pela leitura dos capítulos, pelas sugestões e principalmente pela ajuda durante as fases difíceis.

Este trabalho propiciou vários trabalhos de iniciação científica permitindo a participação de alguns alunos em partes específicas do trabalho resultando em boa ajuda na finalização do mesmo, gostaria de agradecer aos alunos de iniciação científica, Marcelo Cortizzo, Alcyr Silveira Franco, Geovanni Crestano e Richard Gatinho Marques.

Aos técnicos do Laboratório, Eduardo Bonani, Marcelo Carrapatoso e Yuri Amorim pelo auxílio nas atividades de suporte.

Ao CNPq Conselho Nacional de Pesquisa pela concessão da bolsa de Mestrado.

A FINEP e a Faculdade de Engenharia Elétrica da UNICAMP pelo financiamento do Projeto.

A IBM BRASIL pelo apoio a finalização desta tese, permitindo a utilização de recursos e períodos de tempo para a dedicação aos trabalhos de tese.

GOSTARIA DE HOMENAGEAR E DEDICAR ESTA TESE PARA:

EVANDRO E LÚCIA

PELA AMIZADE, PELO CARINHO, PELA AJUDA E PRINCIPALMENTE
POR SEREM REALMENTE PAIS.

PATRÍCIA, CÉSAR, LÚCIO E DÁCIO

PELO APOIO E ESTIMULO A CONCLUSÃO DESTE TRABALHO.

KENIA

PELA AMIZADE, COMPANHEIRISMO E CARINHO.

RESUMO

O objetivo principal deste trabalho é apresentar o desenvolvimento de um controlador a multimicroprocessadores destinado a controle de robôs manipuladores. São apresentados inicialmente dois exemplos de controladores disponíveis comercialmente, geralmente estes apresentam somente um microcomputador responsável por todo o processamento necessário ao controle do manipulador.

Mostra-se a seguir exemplos de controladores a multimicroprocessadores propostas na literatura. Conclui-se que este tipo de arquitetura tem por finalidade aumentar a capacidade de processamento pela utilização de uma arquitetura distribuída com vários microcomputadores dedicados a tarefas específicas.

A estrutura proposta para a tese é detalhadamente apresentada, dedicando-se microcomputadores para controle de juntas e um Micro Mestre para controlar os movimentos do manipulador. São apresentadas as características dos microcomputadores aplicados a controle, visando com isto, dar subsídios para a apresentação do microcomputador desenvolvido com finalidade de controlar cada junta do manipulador.

São detalhados os circuitos desenvolvidos, assim como todas as características necessárias para sua avaliação. Para atestar a operacionalidade do controlador foram desenvolvidas várias rotinas que também são apresentadas. Entre estas, uma Rotina de Controle PID desenvolvida para verificar o funcionamento do conjunto do CPR1.

Em sequência são apresentados dados experimentais obtidos do sistema montado, assim com as conclusões resultantes do seu desenvolvimento, montagem, testes e integração do sistema, com a apresentação dos problemas encontrados e suas respectivas soluções.

CAPÍTULO 1

1 INTRODUÇÃO

1.1 O MUNDO DA ROBÓTICA	01
1.2 IMPACTOS SOCIAIS DA ROBÓTICA.....	04
1.3 CARACTERÍSTICAS GERAIS DE ROBÔS MANIPULADORES.....	06
1.4 CONTEXTO DO TRABALHO DE TESE.....	12

CAPÍTULO 2

2 INTRODUÇÃO AOS SISTEMAS DE CONTROLE A MULTIMICROPROCESSADORES

2.1 MOTIVAÇÕES ARQUITETURAIS.....	17
2.2 PROCESSAMENTO DISTRIBUÍDO.....	19
2.3 ARQUITETURAS TÍPICAS DE CONTROLADORES PARA MANIPULADORES.....	20
2.4 ARQUITETURAS DISTRIBUÍDAS PROPOSTAS NA LITERATURA.....	26
2.5 ARQUITETURA PROPOSTA NESTE TRABALHO.....	33
2.5.2 ESTRUTURA DO BARRAMENTO.....	37
2.5.3 PLACAS MICROPROCESSADORAS DEDICADAS.....	39
2.6 MICROCOMPUTADORES APLICADOS EM CONTROLE DE PROCESSOS.....	42
2.7 VANTAGENS E LIMITAÇÕES DOS MICROPROCESSADORES.....	48
2.8 UTILIZAÇÕES DOS MICROCOMPUTADORES EM CONTROLE DE PROCESSOS... ..	52
2.8 METODOLOGIA DE PROJETO DE SISTEMAS DE CONTROLE DIGITAL.....	55

CAPÍTULO 3

3 CONTROLADOR PROGRAMÁVEL DE ROBOS 1

3.1 DESCRIÇÃO DO SISTEMA.....	59
3.2 MICROCOMPUTADOR MESTRE.....	61
3.3 PLACA DE INTERFACE DO PC (PIPC).....	63
3.4 BARRAMENTO AUXILIAR.....	73
3.5 PLACA DE AMPLIFICAÇÃO DO BARRAMENTO (PABO).....	73
3.6 BARRAMENTO PRINCIPAL.....	74
3.7 PLACA DIGITAL DE CONTROLE DISTRIBUÍDO (PCDD).....	77
3.8 PLACA DIGITAL DE ENTRADAS E SAÍDAS DIGITAIS.....	85

CAPÍTULO 4

4 PROGRAMAS DESENVOLVIDOS PARA O CPR1

4.1 INTRODUÇÃO	88
4.2 PROGRAMA DE COMUNICAÇÃO.....	90
4.3 PROGRAMA MONITOR DAS PCDD'S	97
4.4 PROGRAMA MONITOR DA PDES	100
4.5 PROGRAMA PID PARA CONTROLE DE JUNTAS	102
4.5.1 EQUAÇÃO RECURSIVA UTILIZADA.....	102
4.5.2 DESCRIÇÃO DO DIAGRAMA DE BLOCOS RESULTANTE	104
4.5.3 OBTENÇÃO DOS PARÂMETROS q_0 , q_1 , q_2	106
4.5.4 SOFTWARE DESENVOLVIDO.....	107

CAPÍTULO 5

5 ANÁLISE DE DESEMPENHO E CONCLUSÕES

5.1 CONSIDERAÇÕES INICIAIS.....	113
5.2 ROTINAS DE COMUNICAÇÃO.....	114
5.3 TEMPORIZAÇÕES RESULTANTES COM OS "JUMPERS" DE "WAIT STATE".....	116
5.4 TEMPORIZAÇÕES MÁXIMAS OBTIDAS.....	117
5.5 RUÍDOS OBSERVADOS.....	118
5.6 DESEMPENHO DO ALGORITMO DE CONTROLE.....	119
5.7 COMENTÁRIOS E CONCLUSÕES.....	121

BIBLIOGRAFIA.....	126
-------------------	-----

CAPÍTULO 1

INTRODUÇÃO

1.1 O MUNDO DA ROBÓTICA

Os robôs industriais começaram a ser utilizados há cerca de quatro décadas. Inicialmente eram imprecisos, com controladores simples e pouco versáteis, tendo evoluído consideravelmente e a tendência é que continuem cada vez mais a auxiliar o trabalho do homem em tarefas repetitivas, perigosas, insalubres, e a melhorar a produtividade e a qualidade das nossas fábricas.

Para podermos compreender o desenvolvimento da Robótica ao longo deste período, é importante apresentar a sua evolução não isoladamente, mas sim dentro do desenvolvimento da automação, ou ainda mais profundamente dentro da evolução da própria técnica da humanidade.

Segundo [Kirschner 1984], podemos definir a palavra Técnica como sendo o empenho para poupar esforços. Há cerca de 2.5 milhões de anos o homem primitivo aguçou um pedaço de pedra para fazer um machado, criando assim, um dos primeiros instrumentos da história da técnica. Este autor considera que o mundo do homem é um mundo inerentemente técnico, dizendo que a evolução humana na terra é ditada pela evolução da técnica. Historicamente notamos claramente que os grandes inventos têm mudado a face do mundo, provocado revoluções, enriquecido ou empobrecido povos, tornando alguns poderosos e outros fracos.

Desde os primórdios da humanidade quando o homem descobriu o poder do fogo, começaram a surgir os avanços da técnica. Porém a evolução vinha de um modo lento, até que nas últimas décadas a influência das técnicas científicas aumentou de um modo sem precedentes dentro do sistema "humanidade". Não existe atualmente na prática nenhuma esfera vital que não dependa direta ou indiretamente da técnica, sendo que o ritmo dessa evolução acelera-se com o passar do tempo.

Voltando a evolução da técnica que denominamos automação, notamos que o homem passou a dedicar esforços intelectuais para poupar esforço físico. A automação industrial trouxe o desenvolvimento das novas máquinas à vapor que eram um prenúncio da transição para a área industrial na Europa. Com o aparecimento destas máquinas criaram-se distúrbios sociais que tiveram início com trabalhadores que temiam perder seus empregos devido a mecanização dos processos de produção. Mais tarde apareceram máquinas, ferramentas e instrumentos que em maior ou menor grau passaram a substituir o trabalho manual do homem, sendo que uma das mais novas ferramentas surgidas foram os robôs que nada mais são, do que máquinas programáveis destinadas a realizar a quase totalidade dos trabalhos manuais anteriormente realizados pelo homem.

Os Robôs manipuladores surgiram como uma importante ferramenta para auxiliar o desenvolvimento da automação industrial, devendo ser considerado como uma peça importante para auxiliar na produção de bens de manufatura. O uso de Robôs manipuladores começou em meados da década de sessenta no Japão, Estados Unidos e em vários outros países desenvolvidos, sendo que atualmente começam a ser integrados a modernos conceitos de automação de manufatura tais como o CIM (Computador Integrado a Manufatura) que maximizam o seu aproveitamento.

Existe muita controvérsia nas vantagens decorrentes do uso de robôs industriais para auxiliar na manufatura. Por um lado é inegável que a sua larga utilização contribuiu para o sucesso industrial no Japão, porém em outros países como os EUA o benefício apresentado pelos fabricantes de robôs inicialmente não se verificou. No Brasil o uso de robôs é pouco difundido e restrito a poucos usuários, a explicação para este fato é que temos em geral um sistema de produção arcaico necessitando a aplicação de técnicas e conceitos simples e baratos de automação tais como:

- 1) Concepção de "Lay-out" adequado : Onde pela simples organização do fluxo produtivo, com entrada, saída, estoque de peças e equipamentos bem definidos e arranjados pode-se conseguir uma melhoria significativa dos tempos de produção dos bens manufaturados.

2) Controle de estoque : Onde através da utilização de ferramentas computacionais simples pode-se realizar técnicas de estoque mínimo evitando assim um grande inventário e com otimização do espaço de armazenagem.

Estas e outras técnicas e conceitos trazem resultados práticos satisfatórios com pouca necessidade de recursos.

Para exemplificar o que foi dito, vamos considerar o caso de uma empresa montadora de placas de circuitos eletrônicos. Pode-se automatizar a montagem de componentes nas placas com a utilização de robôs de inserção (por exemplo robô do tipo "SCARA"). Para a utilização de manipuladores são necessários grandes investimentos em equipamentos, pessoal especializado, ferramentas, etc.. A utilização de robôs reduziria o tempo de montagem de lotes em horas. Por outro lado existe o fato de que a maior parte do tempo gasto na montagem de um lote de placas consiste de armazenagem, preparação de componentes, etc., para resolver estes problemas pode-se implementar técnicas de estoque mínimo, controles, localização automática de peças e outros que podem reduzir o "Lead Time" de produção de um lote de cartões em dias resultando em grande economia e um curto período para amortização do investimento

Atualmente no Brasil esta realidade aos poucos vai mudando, de forma que as empresas começam a investir em tecnologias de automação, como um meio de sobrevivência face as crescentes pressões de um mercado consumidor cada vez mais exigente e de uma acirrada competitividade internacional. O Brasil precisa investir em novas tecnologias, em pesquisa, em qualidade e principalmente tornar-se competitivo, para com isto dominar a técnica e passar a não mais ser dependente de outros povos em razão da técnica. Neste contexto a utilização de robôs manipuladores deve ser difundida, tornando-se importante o desenvolvimento do "hardware" e "software" para o uso de robôs com tecnologia totalmente nacional.

1.2 IMPACTOS SOCIAIS DA ROBÓTICA

Este é um assunto muito controvertido pois o uso de robôs tem um aspecto social muito importante. A opinião geral é que o uso de robôs causa desemprego em massa, e que vem substituir o homem, principalmente nos grandes centros industrialmente desenvolvidos. Esta preocupação não deixa de ser pertinente, porém os robôs têm sido tradicionalmente introduzidos para tarefas repetitivas, monótonas, barulhentas e perigosas, e portanto nestas tarefas os robôs poderão realizar uma máxima contribuição para a humanização do trabalho.

Os robôs representam um menor índice de desemprego que outras tecnologias de automação como a informática por exemplo, mas são mais atrativos frente a opinião pública, devido ao fascínio que exerce sobre o homem que o vê pela ficção, como um semelhante seu a lhe fazer concorrência.

O temor que novos equipamentos provoquem desemprego não é de hoje, já nos fins do século XVIII, a máquina à vapor e o tear mecânico lançavam ao desemprego um grande número de operários artesanais, que reagiram através de um movimento irracional, o ludismo (nome dado em função do líder do movimento se chamar LUDD), que consistia em destruir as máquinas na calada da noite.

Nesta época esses operários trabalhavam uma média de 16 horas por dia, hoje graças a elas e a uma racionalização do trabalho, a jornada média de trabalho é de 8 horas.

Evidentemente uma das causas do sucesso e do avanço tecnologico no Japão é o extensivo uso da automação em todos os setores industriais.

O Japão é hoje um dos países mais ricos do mundo e obviamente isto é refletido em melhoria de qualidade de vida da sua população, com isto é inegável constatar que o uso extensivo de robôs trouxe benefícios para a nação e conseqüentemente para o todo o povo Japonês.

Um aspecto importante que pode ser analisado ao longo do tempo é que os avanços científicos e tecnológicos podem ser retardados mas nunca impedidos. Portanto não se deve lutar contra o uso de robôs, mas sim lutar por medidas que contraponham os seus possíveis impactos negativos sobre a sociedade.

1.3 CARACTERÍSTICAS GERAIS DE ROBÔS MANIPULADORES

Robôs manipuladores consistem de uma estrutura mecânica que suporta um corpo denominado órgão terminal ou efetuator a ser posicionado e orientado. A estrutura mecânica consiste de um conjunto de corpos ou elementos ("links") integrados por meio de ligações denominadas juntas. As juntas interligam dois elementos permitindo uma mobilidade relativa entre eles. Em mecânica existem vários tipos de ligações possíveis entre dois elementos, resultando em movimentos em um ou mais graus de liberdade. Para a realização dos movimentos são necessários sistemas atuadores para cada grau de liberdade. Eventualmente, utilizando-se um sistema de locomoção dá-se ao manipulador mobilidade.

Para controlar os movimentos da estrutura mecânica existem sistemas de controle dos atuadores que controlam e coordenam o movimento de cada junta isoladamente. O estudo do mecanismo e controle de robôs manipuladores não é uma ciência nova, é meramente uma coleção de tópicos advindos dos campos clássicos. Por exemplo as metodologias para o estudo do mecanismo em situações estáticas e dinâmicas são contribuições da engenharia mecânica, a matemática fornece as ferramentas para descrever os movimentos no espaço e os outros atributos dos manipuladores, a teoria de controle fornece a ferramenta para projetar e analisar o desempenho dos algoritmos para realizar os movimentos desejados, o desenvolvimento do "hardware" do controlador e o projeto de sensores e interfaces são contribuições da engenharia elétrica e a ciência da computação contribui com a programação destes dispositivos (desenvolvimento do "software").

DEFINIÇÃO :

Segundo o "Robot Institute of America" Um robô é um manipulador multifuncional reprogramável com controle de posição automático tendo vários eixos e capaz de manipular materiais, peças, ferramentas ou dispositivos especializados através de operações variáveis programadas a fim de desempenhar uma variedade de tarefas.

PRINCIPAIS COMPONENTES :

Segundo [Klafter 1989] um robô é composto de quatro componentes principais:

- 1) Manipulador
- 2) Sensores
- 3) Controlador ("hardware" e "software")
- 4) Elemento de Potência

A Figura 1.1 apresenta estes componentes interligados em um sistema. O manipulador é do tipo esférico com juntas rotacionais e seis graus de liberdade. O controlador comanda a Unidade de conversão de energia através da informação do posicionamento desejado e realimentações dos sensores internos distribuídos pela estrutura e de um sensor visual de posição do efetuator.

MANIPULADOR (ESTRUTURA MECÂNICA)

A estrutura mecânica ilustrada na Figura 1.1 consiste de um manipulador com elementos mecânicos interligados por meio de juntas, formando uma cadeia cinemática aberta.

Existem vários tipos de juntas utilizadas nos manipuladores, sendo que as duas principais são:

- Junta Rotacional - movimento rotacional
- Junta Prismática - movimento translacional

Cada junta do manipulador define um eixo de junta ao longo do qual o elemento ligado a ela se movimenta (rotação ou translação). Cada junta define um grau de liberdade de modo que o número de graus de liberdade dos manipuladores que utilizam somente juntas prismáticas e rotacionais seja igual ao número de juntas. Um manipulador elementar completo apresenta sete graus de liberdade, reservando três para posicionamento, três para orientação e um para preensão de objetos.

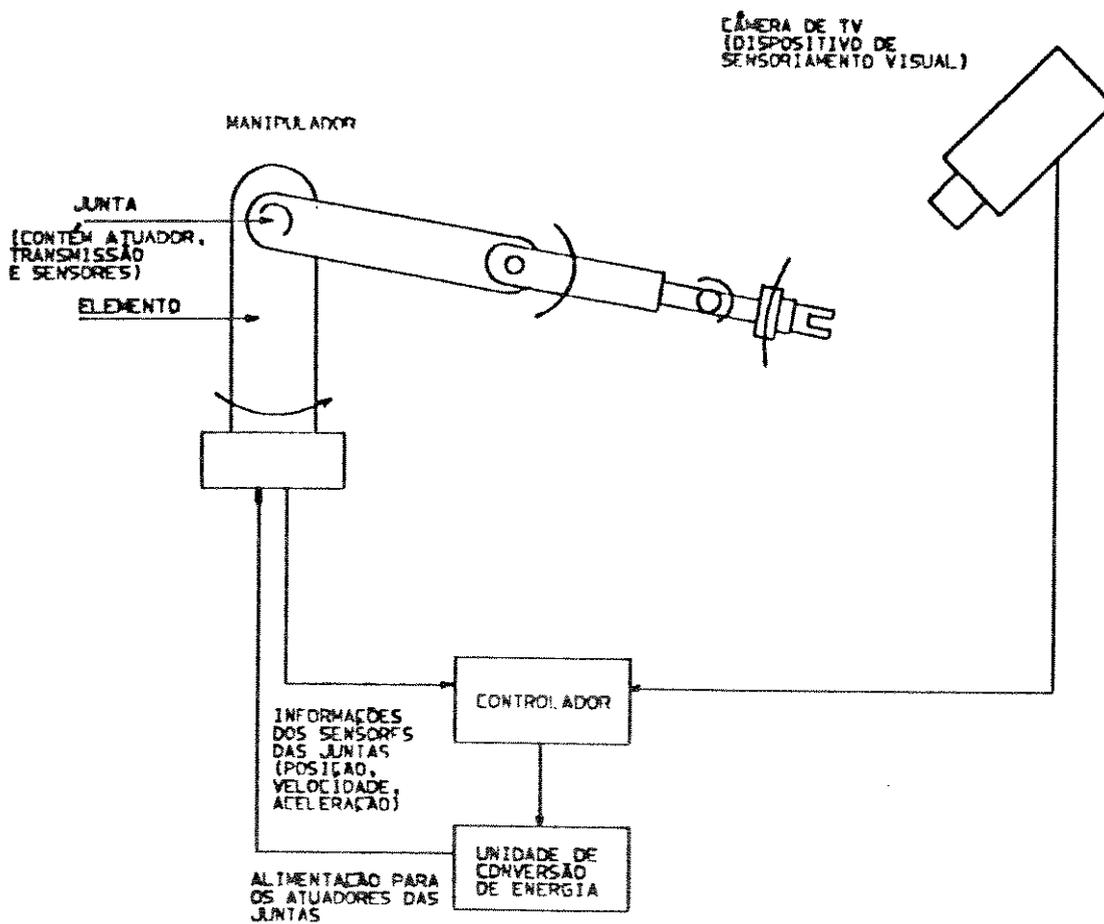


Figura 1.1 : Componentes básicos de um Manipulador

Os manipuladores constituídos por uma estrutura com elementos e juntas geralmente contêm três órgãos básicos com as suas respectivas funções:

- braço (posicionamento)
- punho (orientação)
- garra (preensão)

Além dos componentes mecânicos, os manipuladores contêm dispositivos para a realização do controle dos movimentos dos vários elementos e juntas mecânicas. Estes dispositivos são; os atuadores ou acionadores podendo ser de origem pneumática, hidráulica ou elétrica; os sensores de posição, rotação e torque; as chaves de fim de curso; os relês; e outros.

O uso de sensores tem por finalidade informar as condições do manipulador, obter informações sobre a aceleração, velocidade e posição das juntas. Através da realimentação destas variáveis para a unidade de controle pode-se controlar adequadamente a estrutura mecânica.

Os sensores utilizados nos manipuladores podem ser divididos em dois grupos:

- Sensores não visuais
- Sensores visuais

Os sensores não visuais incluem chaves limitadoras (sensor de proximidade, fotoelétrico ou mecânico), sensores de posição (encoders, potenciômetros, e resolvers), sensores de velocidade (tacômetros), sensores de força (células de carga).

Os sensores visuais consistem de câmeras CCD (Dispositivos de carga acoplada) integradas ao "Hardware e Software" de detecção de imagem. Estes sensores são utilizados para a realimentação da posição do efetuador, para reconhecimento de objetos e para a preensão de objetos.

CONTROLADOR:

Pode-se considerar um manipulador como sendo um dispositivo mecânico de posicionamento. Para que o manipulador execute tarefas de posicionamento do seu efetuator, dentro do espaço de trabalho, com rapidez e precisão, é necessário que este apresente um sistema de controle capaz de assegurar um desempenho rápido e preciso em qualquer condição de operação.

O controle dos movimentos dos manipuladores constitui um problema complexo por diversos motivos: por um lado, o movimento da estrutura mecânica se realiza através de movimentos de rotação e translação de suas juntas que devem ser controladas simultaneamente e cujo acoplamento dinâmico dificulta o controle independente das mesmas: por outro lado, o comportamento dinâmico da estrutura articulada é fortemente não linear e dependente das condições operativas. Condições estas que devem ser levadas em conta na estratégia de controle escolhida.

Finalmente, a trajetória desejada é definida pela, posição velocidade, aceleração e orientação do efetuator, tornando-se necessário então, efetuar transformações de coordenadas com tempos bem definidos e com grande complexidade de cálculos.

O desenvolvimento dos microprocessadores, cada vez mais versáteis, rápidos e precisos tem tornado possível a utilização de técnicas avançadas de controle de movimentos de manipuladores. Em muitos casos torna-se necessária a utilização de estruturas a multimicroprocessadores em forma hierárquizada para conseguir realizar o grande volume de cálculo que estas estratégias necessitam, e permitindo um processamento independente das juntas, caracterizando um processamento paralelo a partir da geração das referências para cada junta.

Muitas são as estruturas a multimicroprocessadores que podem ser utilizadas para controle de manipuladores. Uma estrutura muito utilizada é a que divide a tarefa de controle em subtarefas e aloca microprocessadores específicos para cada subtarefa.

UNIDADE DE CONVERSÃO DE ENERGIA

A função básica destes elementos é fornecer energia para os acionamentos do manipulador. Pode ter a forma de um amplificador de potência para o caso de utilização de motores elétricos, compressores para acionamentos pneumáticos e reguladores de pressão para acionamentos hidráulicos. Os acionamentos mais comumente utilizados são:

1) Elétrico

Utiliza algum tipo de elemento que converte energia elétrica em movimento rotacional ou translacional. São exemplos os motores de corrente contínua e os motores de passo. Não oferece uma boa relação peso desempenho, especialmente se considerarmos as reduções mecânicas necessárias.

2) Pneumático

Utiliza ar comprimido para realizar movimentos. São usados quase que exclusivamente em manipuladores pequenos para movimentação de pequenas peças entre lugares fixos.

3) Hidráulico

Utiliza líquido em alta pressão para realizar movimentos, sendo capaz de fornecer maior potência para um determinado volume, tem o inconveniente de poder apresentar vazamentos, exige bombas, reguladores de pressão e outros acessórios, embora estes sejam componentes industriais comuns.

Estas formas de acionamento não são mutuamente excludentes, mas é conveniente que pelo menos o acionamento básico do manipulador seja uniforme, reservando-se como opção a utilização de uma forma diferente para finalidades especiais, por exemplo, pode-se utilizar acionamento pneumático no efetuator independentemente da forma de acionamento utilizada para os outros movimentos.

1.4 CONTEXTO DO TRABALHO DE TESE

Este trabalho insere-se dentro de um projeto em desenvolvimento no Laboratório de Sistemas e Controle de Energia da Faculdade de Engenharia Elétrica da UNICAMP, denominado JECA II que tem por objetivo a construção de um manipulador mecânico com seis graus de liberdade. O diagrama de blocos do sistema Eletro-eletrônico está apresentado na Figura 1.2 e indica o fluxo de informações no sistema. As principais características do projeto são:

Manipulador

O manipulador desenvolvido apresenta 5 juntas rotacionais e uma prismática. O espaço operacional é do tipo esférico.

Acionamento

O acionamento escolhido é do tipo elétrico, segundo [Ferraté 1986] os motores elétricos permitem um controle antecipativo de posição e velocidade através de controladores eletrônicos e a sua conexão aos manipuladores é simples. Isto aliado ao fato que os atuadores para manipuladores devem possuir baixa inércia e uma capacidade de trabalhar a baixas velocidades, faz com que o atuador eletro-mecânico mais utilizado sejam os motores contínua de baixa inercia e os motores de passo.

Os motores de corrente contínua utilizados possuem imã permanente sendo controlado pela corrente de armadura. O acionamento é realizado por meio de Recortadores a transistores. As placas de controle de junta fornecem um palavra digital de 12 bits indicando o ângulo de comutação para o módulo gerador de MLP (Modulação por Largura de Pulso), este módulo converte esta palavra digital em um ângulo de comutação θ_c de acordo com a Figura 1.3. A frequência de trabalho utilizada foi fixada em 1 kHz, porém pode-se varia-la de 200 Hz a 5 kHz. O ângulo θ_c pode variar de 0 a 360 graus de acordo com a palavra dos controladores de juntas:

0000H ==> $\theta_c = 0^\circ$
7FFFH ==> $\theta_c = 180^\circ$
FFFFH ==> $\theta_c = 360^\circ$

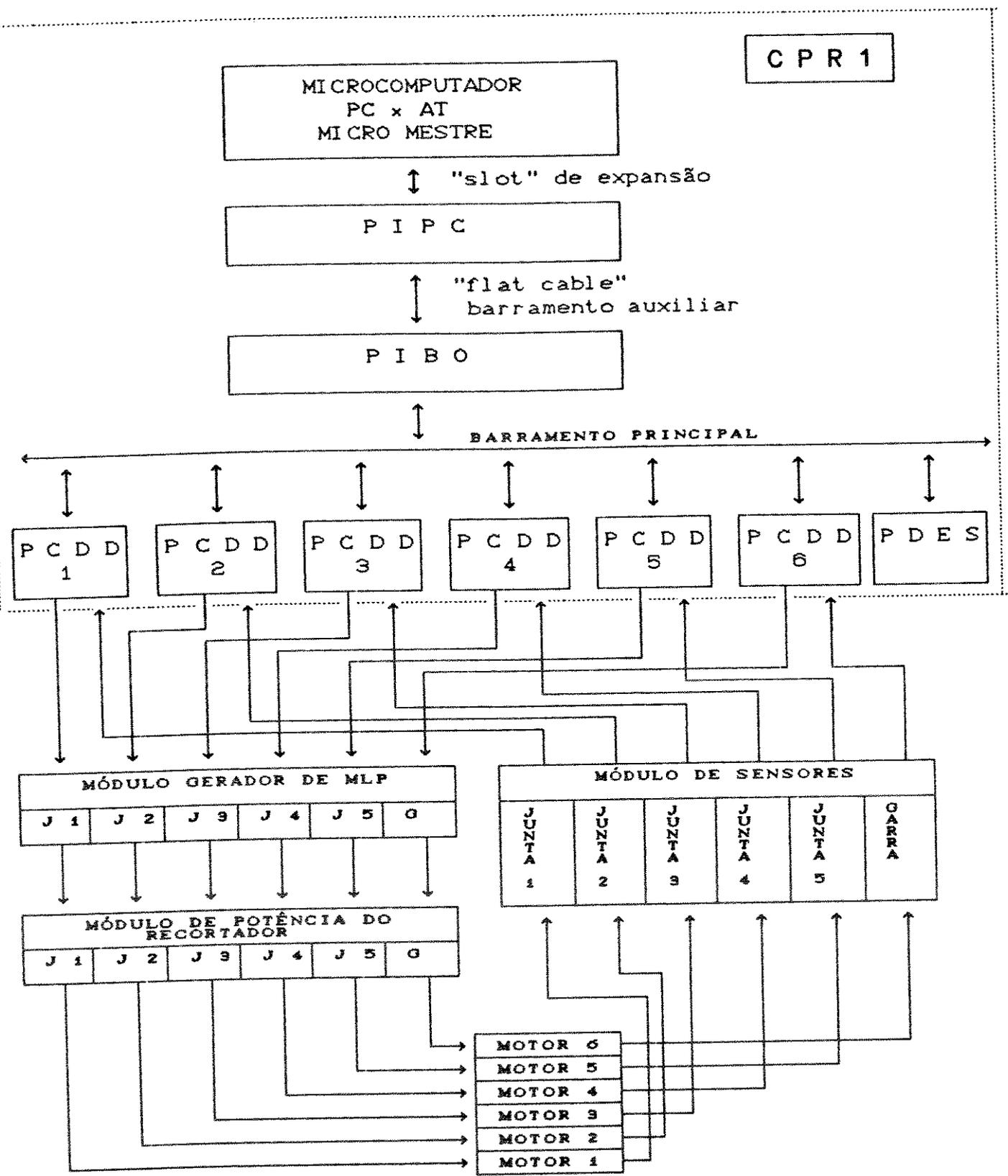


Figura 1.2 : Diagrama de blocos do Sistema Eletro-eletrônico

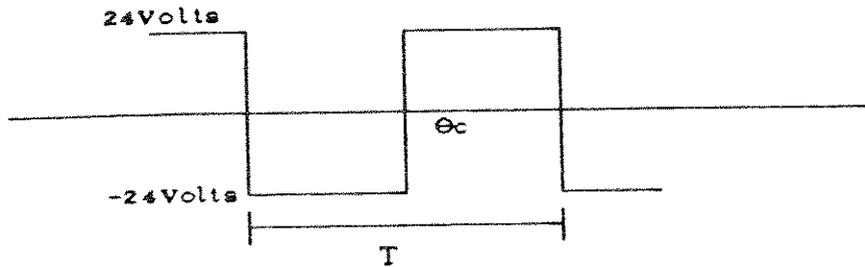


Figura 1.3 : Sinal de controle para o Recortador

O recortador utilizado funciona em quatro quadrantes, portanto ele regenera energia durante a frenagem. O recortador fornece um sinal de tensão nos bornes do estator dos motores que provoca o seu acionamento.

O recortador suporta 40 (A) de pico e 8 (A) continuamente. O valor de tensão varia de -24 (V) a + 24(V) de acordo com o ângulo de comutação.

Sensores

Básicamente os sensores utilizados consistem de potenciômetros e "encoders" para identificar a posição, tacômetros para velocidade e resistores série para corrente.

Cada junta apresenta um potenciometro linear sem fim de curso (10 kOhms) para medidas absolutas, um "encoder" optico (1000 pulsos por rotação) para incrementos, um tacômetro analógico (3(V)/1000 rpm) e um resistor "shunt" de 0.1 Ohms/5 Watts.

Os sinais são processados pelo módulo dos sensores de modo que o sinal fique compreendido entre +/- 10 Volts antes de serem enviados para as placas controladoras de juntas.

Controlador

O controlador apresenta uma estrutura hierárquica de controle apresentando um mestre cuja função é controle de trajetória e interface homem-máquina e placas microprocessadas dedicadas a controle de juntas. Esta configuração foi escolhida pois apresenta melhor desempenho quando comparado com os sistemas de controle com um único microprocessador. Com a configuração utilizada consegue-se uma boa capacidade computacional com a utilização de microprocessadores simples e de baixo custo. A idéia básica é utilizar um sistema a multimicroprocessadores para dividir a tarefa de controle e assim poder implementar algoritmos de controle mais sofisticados. O processamento dos algoritmos de cada junta é feito de forma paralela, permitindo à estrutura um desempenho dinâmico rápido e eficiente.

O objetivo desta tese é o projeto e implementação do controlador que vamos denominar CPR1 (Controlador programável de Robôs 1). Para apresentação o trabalho foi dividido nos seguintes capítulos:

No capítulo 2 apresenta-se uma introdução aos sistemas distribuídos a multimicroprocessadores citando algumas aplicações no controle de manipuladores. A seguir apresenta-se um estudo de microcomputadores aplicados a controle de servomecanismos, visando com isto dar subsídios ao projeto.

No capítulo 3 descreve-se a estrutura do controlador CPR1 projetado e construído, sendo apresentados diagramas de blocos do sistema global assim como dos componentes do CPR1.

No capítulo 4 apresenta-se a descrição do software básico de testes e comunicação desenvolvidos para o sistema. Foi também desenvolvido um software de controle de posição PID implementado pelo microcomputador Mestre e processado nos microcomputadores dedicados ao controle das juntas.

No capítulo 5 apresenta-se os resultados da análise de desempenho do controlador e conclusões obtidas no trabalho.

Dois apêndices foram agregados, formando a documentação do sistema, estes apêndices são referenciados no texto deste trabalho, porém se encontram em um bloco a parte.

No Apêndice A1 são mostrados os esquemáticos dos blocos funcionais apresentados no Capítulo 3.

No Apêndice A2 são mostradas as rotinas dos algoritmos apresentados no capítulo 4.

CAPÍTULO 2

INTRODUÇÃO AOS SISTEMAS DE CONTROLE A MULTIMICROPROCESSADORES

2.1 MOTIVAÇÕES ARQUITETURAIS

As últimas décadas têm se caracterizado por um grande aumento na complexidade dos circuitos integrados, ou seja, o número de transistores nos Circuitos Integrados (CI's) dobra a cada ano. Geralmente confunde-se este aumento na complexidade, com um equivalente aumento na capacidade de processamento que é geralmente medida em operações por segundo.

A Figura 2.1 mostra a evolução da complexidade dos CI's nas últimas quatro décadas, observa-se que a complexidade dos circuitos integrados aproximadamente dobra a cada ano. A linha reta na escala semilogaritmica implica na ocorrência de um crescimento exponencial nos últimos quarenta anos. Segundo [grahan 1987] a complexidade passou de 10 transistores por circuito integrado no começo da década de sessenta para mais de 1 milhão de transistores na década de 80.

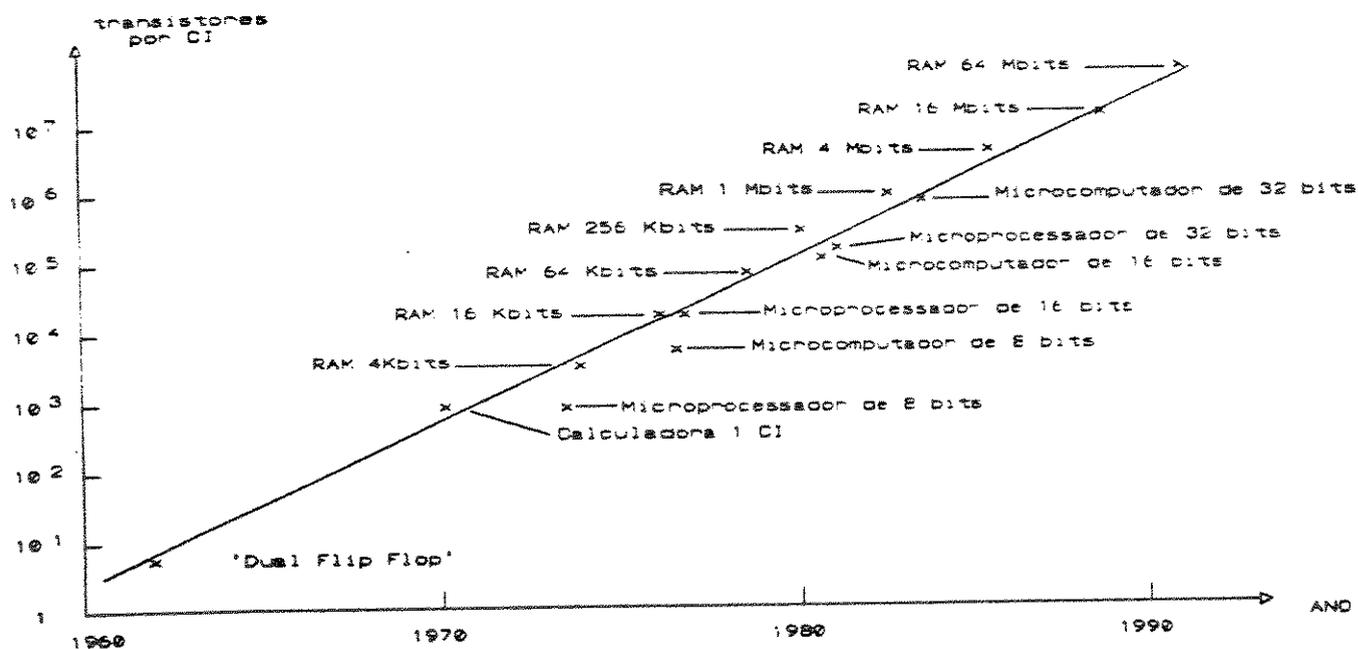


Figura 2.1 : Evolução dos Circuitos Integrados

Por outro lado a Figura 2.2 mostra o aumento na capacidade de processamento das diferentes classes de computadores nas últimas quatro décadas, ou seja, no mesmo período de tempo apresentado na Figura 2.1. As curvas resultantes são concavas, como a escala é semilogaritma estas curvas indicam que o crescimento na capacidade de processamento não é exponencial como no caso da complexidade dos circuitos integrados.

Uma das razões para este fato é que geralmente o atraso de processamento devido a comunicação aumenta substancialmente com o aumento na complexidade da arquitetura. Qualquer sistema de processamento consiste de várias unidades funcionais que precisam comunicar-se entre si, quando a complexidade do circuito aumenta, estas atividades de comunicação também aumentam.

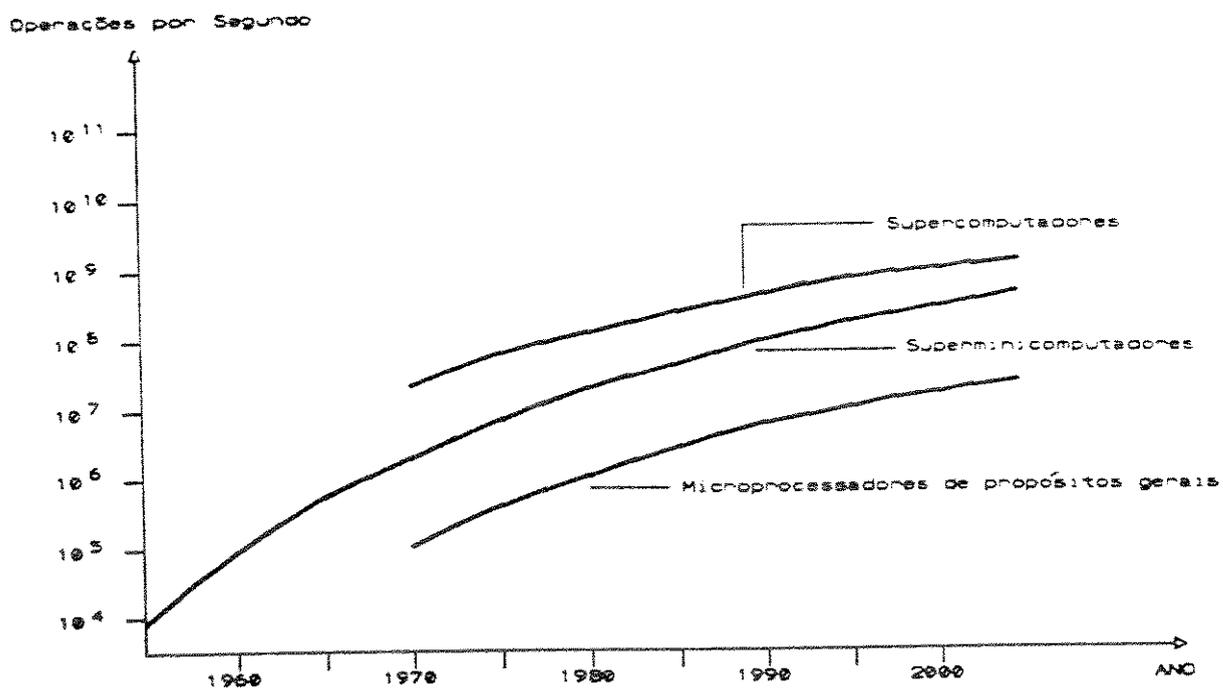


Figura 2.2 : Aumento da capacidade de processamento dos computadores

Segundo [Graham 1987] esta análise baseia-se em arquiteturas de computadores de propósitos gerais. Para computadores dedicados a aplicações especiais, uma forma de aumentar a capacidade de processamento está na utilização de arquiteturas paralelas especializadas.

2.2 PROCESSAMENTO DISTRIBUÍDO

O termo processamento distribuído pode ter dois significados:

- 1) Uma tarefa é distribuída entre vários processadores, ou seja, identifica-se as partes concorrentes de uma tarefa e sua alocação é repartida entre processadores interligados por algum meio, por exemplo, barramento comum ou memória compartilhada.
- 2) As tarefas são distribuídas geograficamente. Neste caso elas são normalmente independentes e os processadores, também geograficamente separados, são utilizados para executar estas tarefas. Os processadores comunicam-se entre si através de algum tipo de rede. A distância entre processadores pode variar desde poucos metros num escritório, a centenas de metros em um avião e a centenas de quilômetros em uma rede. Para descrever computadores localmente interconectados, isto é, sem o uso de comunicação pública, usamos o termo rede local.

Segundo [Parker 1983] as maiores motivações para a construção de sistemas a multiprocessadores são:

- a) - Melhoria do desempenho
- b) - Aumento da Confiabilidade
- c) - Conciliar necessidades de aplicações distribuídas
- d) - Construir arquiteturas especializadas
- e) - Costruir supercomputadores

Para o caso do desenvolvimento de controladores dedicados a robôs manipuladores a utilização de estruturas distribuídas é uma tentativa de melhorar o desempenho do sistema minimizando o tempo de processamento. Entretanto, a utilização desta arquitetura implica em que, a tarefa de controle permita divisões em subtarefas quase independentes e processadores dedicados são alocados para a execução de cada subtarefa.

O aumento na capacidade de processamento é geralmente necessário pois sendo o controle realizado em tempo real, torna-se assim o tempo de execução, um fator crítico para a elaboração dos algoritmos de controle.

A escolha do esquema de interligação e de comunicação entre os processadores é importante, pois existem muitas opções possíveis, dependendo do tamanho, da topologia, do desempenho desejado e dos microprocessadores utilizados.

2.3 ARQUITETURAS TÍPICAS DE CONTROLADORES PARA MANIPULADORES

A Figura 2.3 ilustra os componentes básicos de um controlador para um manipulador industrial típico. Além do controle de juntas, o controlador (com um único processador) também é responsável pela interface com o operador, inicialização, programação de tarefas, aquisição e armazenamento de dados, interface com equipamentos industriais, monitoração do espaço operacional, chaves de fim de curso, etc ...

Quando o manipulador estiver conectado a uma rede, o controlador é responsável pela comunicação com outros processadores na rede.

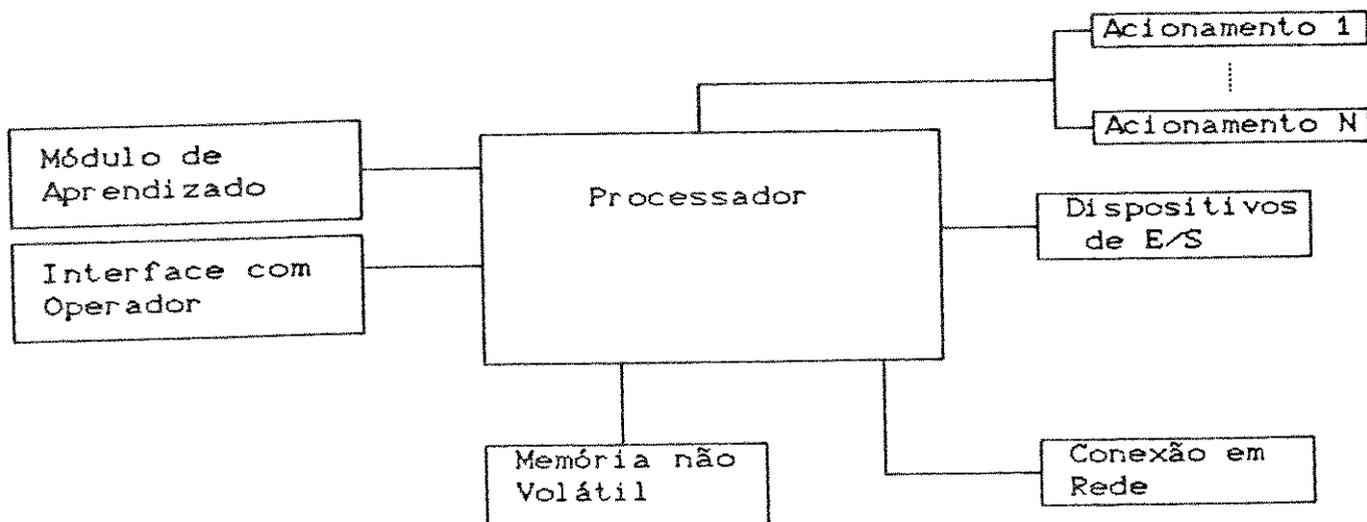


Figura 2.3 : Arquitetura do controlador de um manipulador típico

EXEMPLO DO MANIPULADOR 7576 (TIPO SCARA)

Este manipulador é composto basicamente [Technical Reference do 7576] de três subsistemas principais:

- Controlador
- Módulo de Potência
- Estrutura Mecânica (Planar com 4 graus de liberdade)

O controlador é responsável pela operação do manipulador, coordenando seus movimentos, monitorando sua velocidade e seu posicionamento. O controlador apresenta as seguintes características:

- 512 kbytes de memória
- Microprocessador 80286
- Coprocessador matemático 80287
- 48 Portas de Entradas/Saídas Digitais
- 4 portas RS-232 (Comunicação serial)
- Adaptador de disquete
- Fonte de alimentação (192 Watts)
- 2 Cartões de Monitoração de juntas

O diagrama de blocos pode ser observado na Figura 2.4. As portas de E/S digitais se destinam a controle de ferramentas, alimentadores, alarmes, válvulas, enfim do ambiente de trabalho do robô. Uma porta serial de comunicação RS 232 é utilizada para comunicação do microprocessador com o módulo de aprendizado e outra é utilizada quando o manipulador estiver conectado a um processador central.

Um programa aplicativo (sob controle do 80286) controla os movimentos da estrutura mecânica, ou seja, o cartão de monitoração das juntas recebe um sinal de referência e após verificar o valor realimentado, calcula o sinal de controle para o servo amplificador da junta.

A interface entre o controlador e a estrutura mecânica é denominada Módulo de Potência. Este módulo é composto por quatro recortadores responsáveis pelo acionamento dos motores de corrente contínua do manipulador (servos amplificadores das juntas).

Existem duas fontes de potência, uma para alimentar o controlador e outra para fornecer energia para os recortadores que acionam os motores.

Um único microprocessador comanda todas as operações do robô, exigindo grande atividade da CPU para realizar todas estas operações.

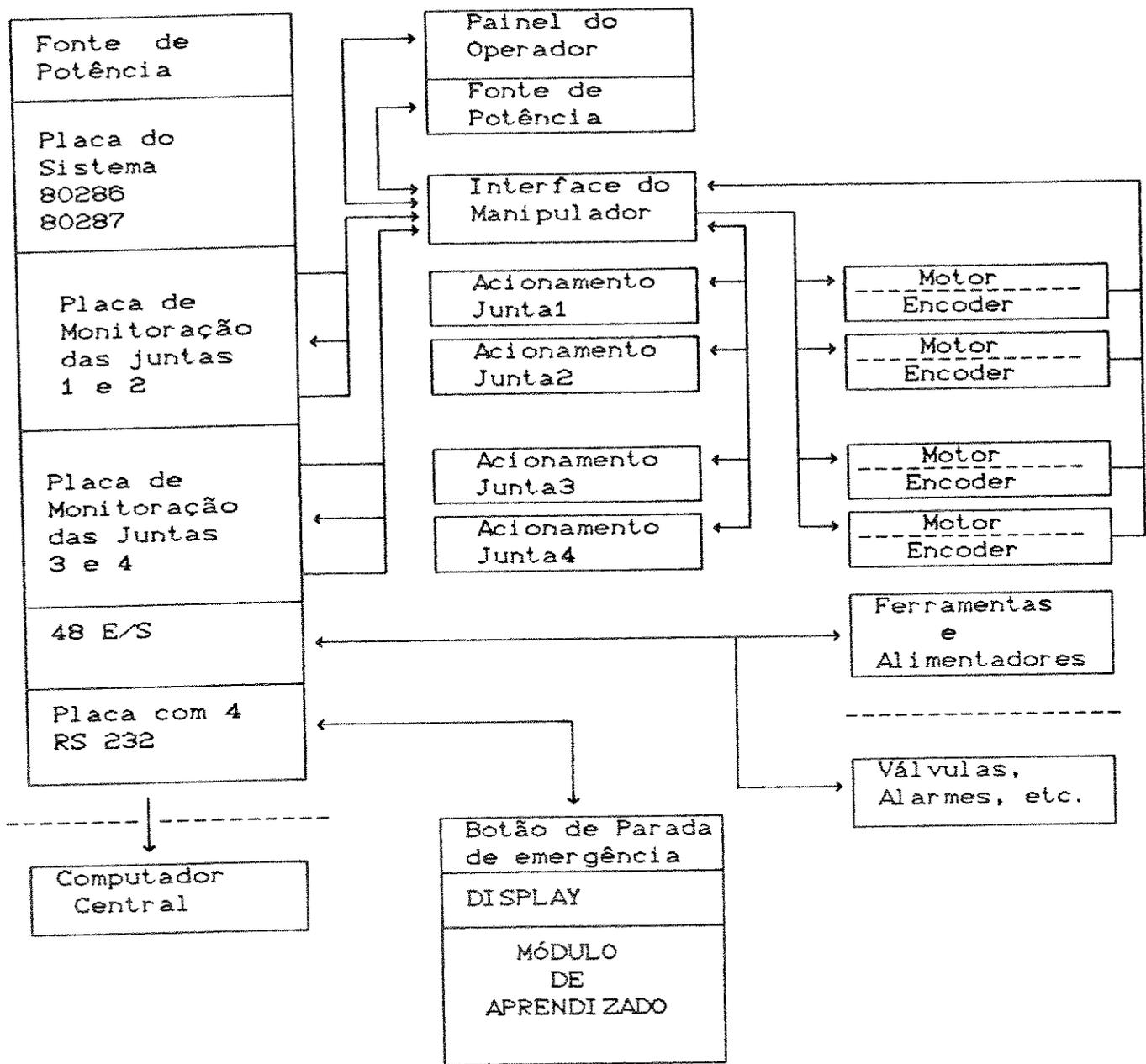


Figura 2.4 : Diagrama de Blocos do robô 7476

EXEMPLO DO ROBÔO SERVO HIDRÁULICO DA SERVUS (SR4)

O manipulador Servo-Hidráulico fabricado no Brasil pela SERVUS apresenta segundo o Folheto Técnico SF16 as seguintes características:

- Três servo atuadores para posicionamento espacial
- Três servo atuadores para orientação espacial
- Servo atuador para a garra

O diagrama de blocos apresentado na Figura 2.5 apresenta um microcomputador central que é responsável por todo o sistema, ou seja, memória, teclado, dispositivo programador de tarefas, unidades de armazenamento de massa, módulo de distribuição e aquisição de dados, monitor, o microcomputador também é responsável pelo "software", ou seja, interface homem-máquina, comunicação, geração de referências para as juntas e outros.

O controlador de certa forma distribui a tarefa de controle do manipulador pois utiliza para controle de juntas controladores PID analógicos, diminuindo assim o trabalho do processador central, porém, sem as vantagens inerentes do controle digital nas juntas (Ver item 2.7 "Vantagens e limitações dos processadores").

As apresentações destes robôs foram realizadas de forma superficial pois o objetivo é exemplificar sistemas de controladores de manipuladores atualmente disponíveis no mercado.

Estes sistemas utilizam um único processador, porém na literatura tem sido propostas arquiteturas de controle distribuído a multimicroprocessadores, algumas destas arquiteturas estão exemplificadas a seguir.

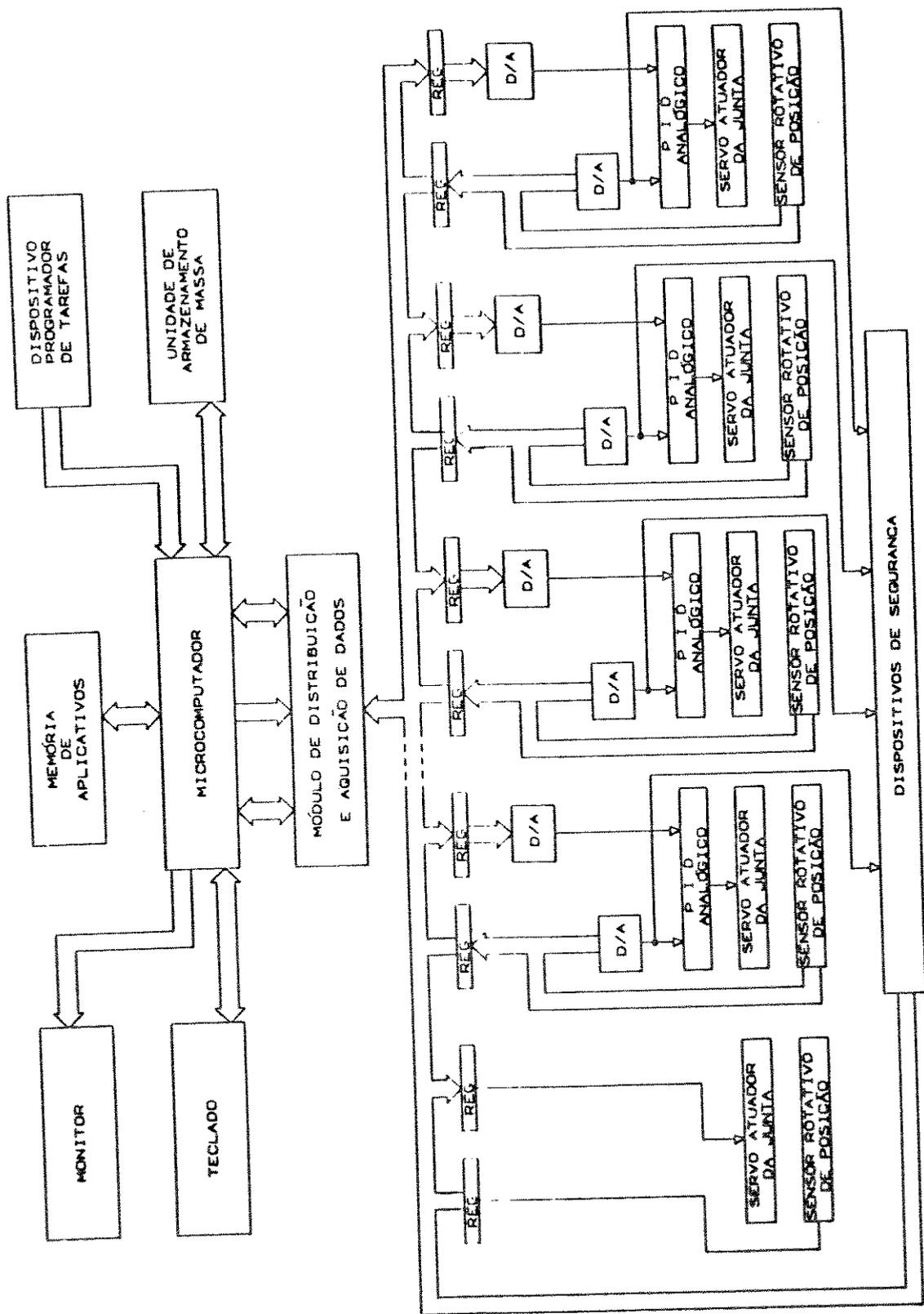


Figura 2.5 : Diagrama de Blocos do controlador SR4

2.4 ARQUITETURAS DISTRIBUÍDAS PROPOSTAS NA LITERATURA

Segundo [Klafter 1987] a utilização de controladores com processamento distribuído a multimicroprocessadores tem várias vantagens:

- Com o uso de processadores adequados para cada aplicação pode-se minimizar custo e complexidade, obtendo-se um desempenho ótimo.
- O "software" para cada processador pode ser projetado, codificado e testado independentemente dos outros processadores.
- O sistema pode ser projetado para ser modular, portanto a complexidade do controlador pode ser reduzida quando necessitar-se de menos recursos para uma aplicação específica.
- Pode-se distribuir as tarefas de modo que os defeitos ocorridos sejam de fácil detecção pois somente os módulos responsáveis pela tarefa defeituosa devem ser verificados.

Por outro lado as desvantagens do uso de multimicroprocessadores distribuídos são:

- Existe a dificuldade de comunicação entre os processadores, principalmente em sistemas cujo atraso devido a comunicação é prejudicial ao desempenho do sistema, como no caso do controle de processos (com constantes de tempo pequenas) em tempo real. Nestes sistemas, a comunicação deve ser rápida o suficiente para eliminar tais problemas.

- Para testar cada módulo separadamente devemos providenciar sistemas para simular os sinais e dados dos módulos externos, que podem ser complexos e difíceis de serem implementados.
- Se o sistema necessitar de testes de conjunto, deve-se ter disponível analisadores lógicos múltiplos ou equipamentos de teste especialmente projetados para esta finalidade.

ARQUITETURA PROPOSTA POR KLAFTER

A Figura 2.6 apresenta a estrutura do controlador. Esta arquitetura é citada como um exemplo onde assume-se que exista algum tipo de interface de comunicação de alta velocidade entre o controle central ou Mestre e os outros módulos de hardware.

Neste sistema são utilizados processadores separados para controlar cada junta associada com o manipulador. As únicas informações que estes módulos necessitam do sistema Mestre são os dados de referência e de sincronização, os quais são transmitidos através do barramento comum. Para sincronizar os processadores das juntas uma mensagem é enviada para todos os processadores simultaneamente, informando que os algoritmos devem utilizar novas referências.

Neste sistema o controle de cada junta é realizado por meio de um microprocessador, possibilitando a divisão de tarefas de controle e liberando o micro Mestre para o controle de tarefas e geração de referências para o controle de trajetórias, interface homem-máquina e outras atividades de sua responsabilidade.

Outra característica do sistema é a sua modularidade, ou seja, o barramento permite o acoplamento de placas diversas, por exemplo uma placa de aquisição de imagens, ou uma placa de controle da garra, uma placa para controle de esforços, etc.. Isto só é possível através da utilização de dispositivos de comunicação que possibilitem a integração dos diversos módulos.

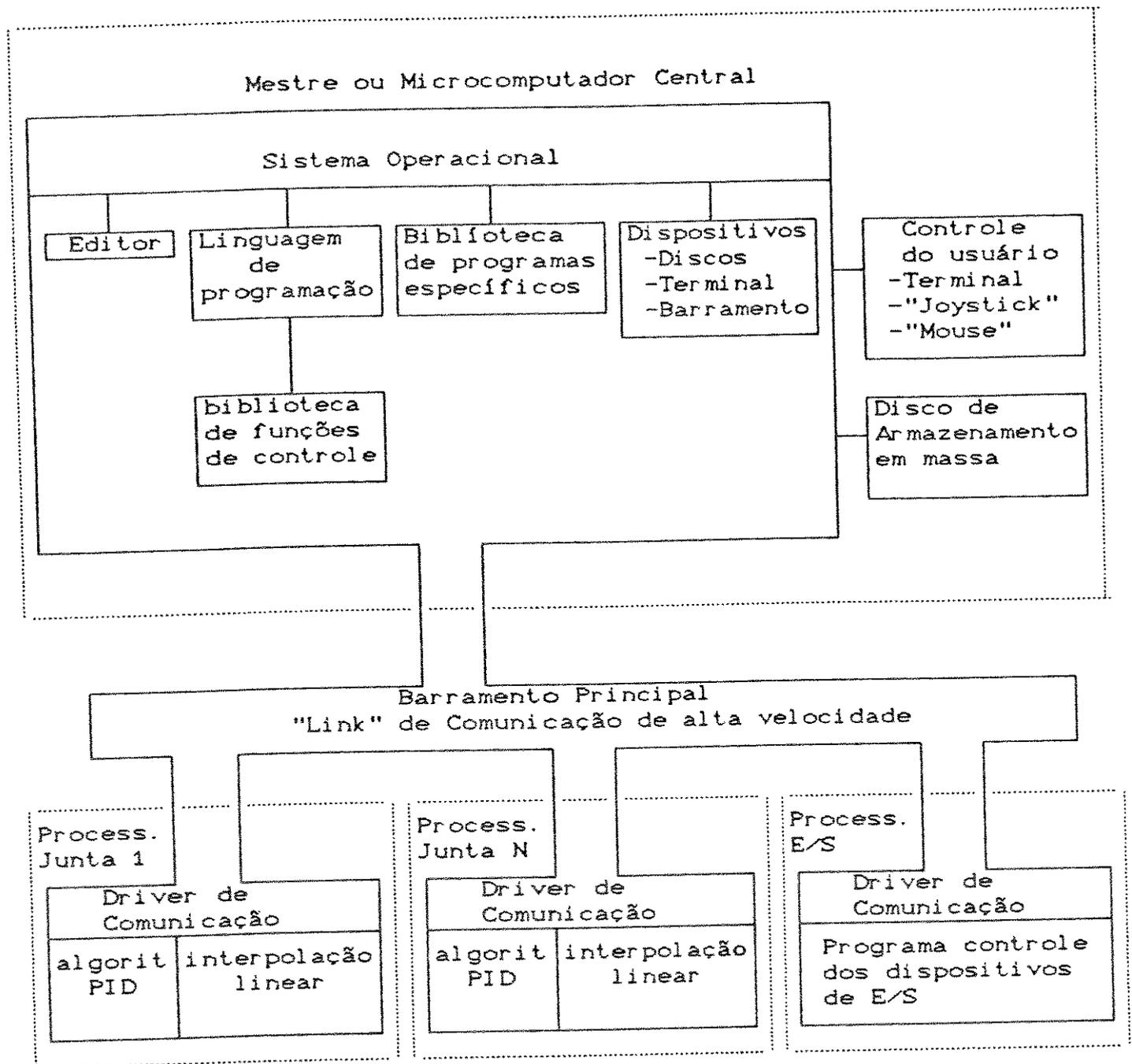


Figura 2.6 : Arquitetura do Controlador proposta por Klafter

ARQUITETURA PROPOSTA POR GRAHAM

A arquitetura proposta por [Graham 1987] está apresentada na Figura 2.7. Apresenta-se uma estrutura de controle hierárquica para controle da estrutura mecânica, consistindo basicamente de um sistema "Organizador" (Tem a função de comandar e coordenar a operação do sistema), um sistema de visão (Tem a função de realimentar a posição do efetuador), um sistema de sensoreamento (Tem a função de captar sinais das juntas), um coordenador do braço com dois controladores digitais microprocessados (Um para controlar a posição das juntas 1-3 e outro para controlar orientação através das juntas 4-6 e o efetuador).

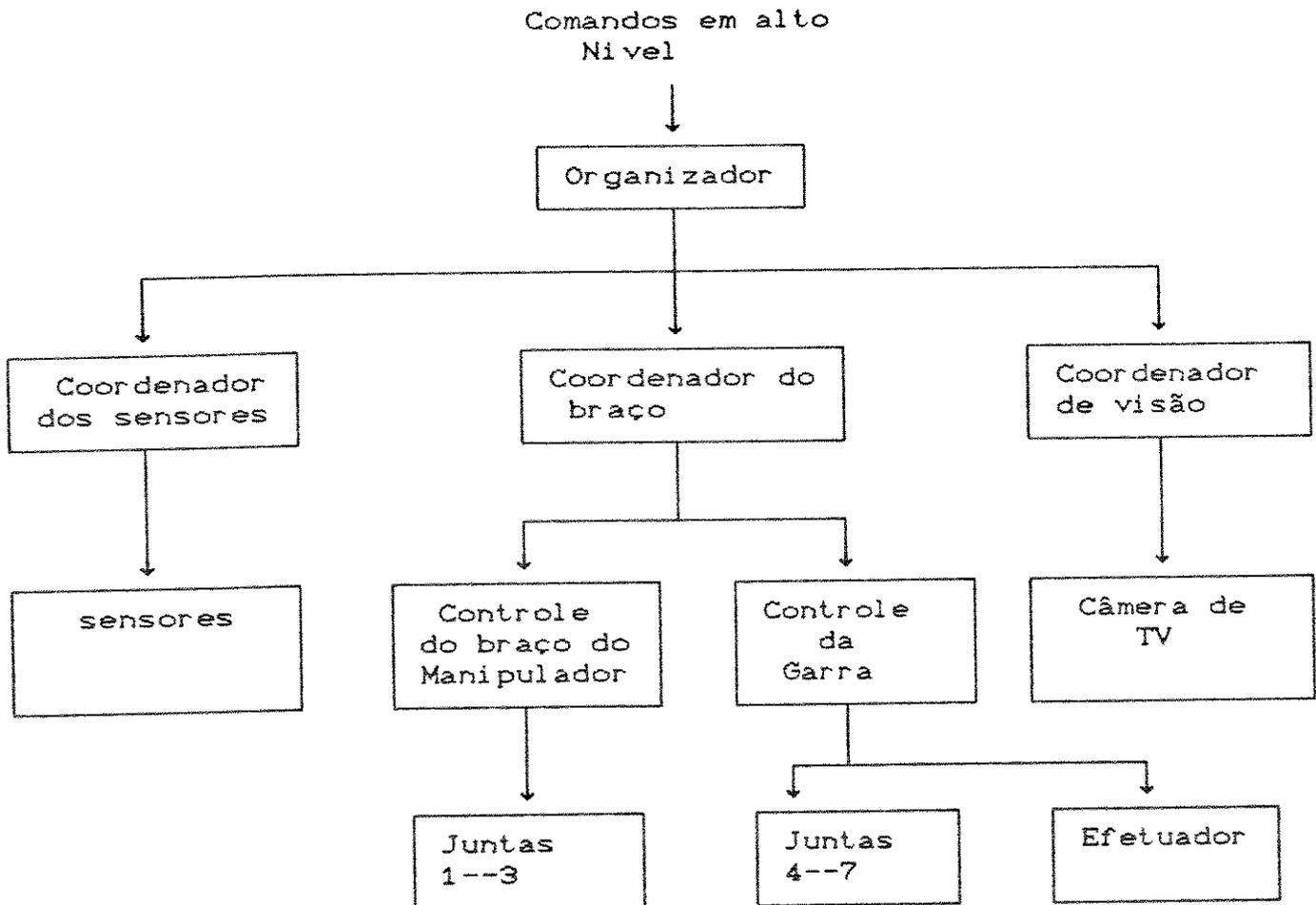


Figura 2.7 : Estrutura do controlador proposta por Graham

A capacidade de processamento necessária para o controle das juntas do manipulador depende das especificações de desempenho requeridas nas aplicações desejadas. Manipuladores para manuseio de materiais podem usar, por exemplo, algoritmos PID de controle de juntas. Estes algoritmos estão sujeitos a "overshoot" e oscilações quando se ultrapassar os limites de velocidade e carga nominais, também estão sujeitos a variação de parâmetros dos componentes das juntas. Com a estrutura paralela em muitos casos pode-se utilizar algoritmos mais sofisticados como controle adaptativo, ótimo e robusto.

As atividades de visão e sensoreamento são de fundamental importância se desejarmos aplicações avançadas para o manipulador. Quanto maior a sofisticação da aplicação geralmente é maior a integração necessária com o ambiente através de sensores e visão.

O processamento digital de imagens tem sido estudado em um grande número de aplicações com ênfase em automação e robótica, provavelmente por causa do alto grau de necessidade de realimentação visual para a realização das tarefas anteriormente planejadas e executadas por homens.

Sensores de força geralmente montados no punho, são essenciais para aplicações de preensão e montagens com controle de esforços.

ARQUITETURA PROPOSTA POR ZHENG

A arquitetura proposta por [Zheng 1986] está apresentada na Figura 2.8. O autor apresenta uma proposta de cálculo de torques aplicados em tempo real no controle dinâmico de um manipulador.

A proposta é utilizar um sistema a multimicroprocessadores composto de uma CPU central e um grupo de CPU's satélites para dividir o cálculo da dinâmica através da formulação de *Newton-Euler*. A tarefa de cada satélite é controlar uma junta calculando seus dados relativos. A tarefa da CPU central é calcular os torques a serem aplicados.

O "software" é o mesmo para todas as CPU's satélites. O esquema a multimicroprocessadores resultante é flexível e modular adaptando-se a grande variedade de configurações dinâmicas dos manipuladores.

[Zheng 1986] não apresenta detalhes de comunicação e nem sobre o ambiente de operação do robô. O esquema de comunicação com linhas entre todos os processadores pode ser muito difícil de ser implementado.

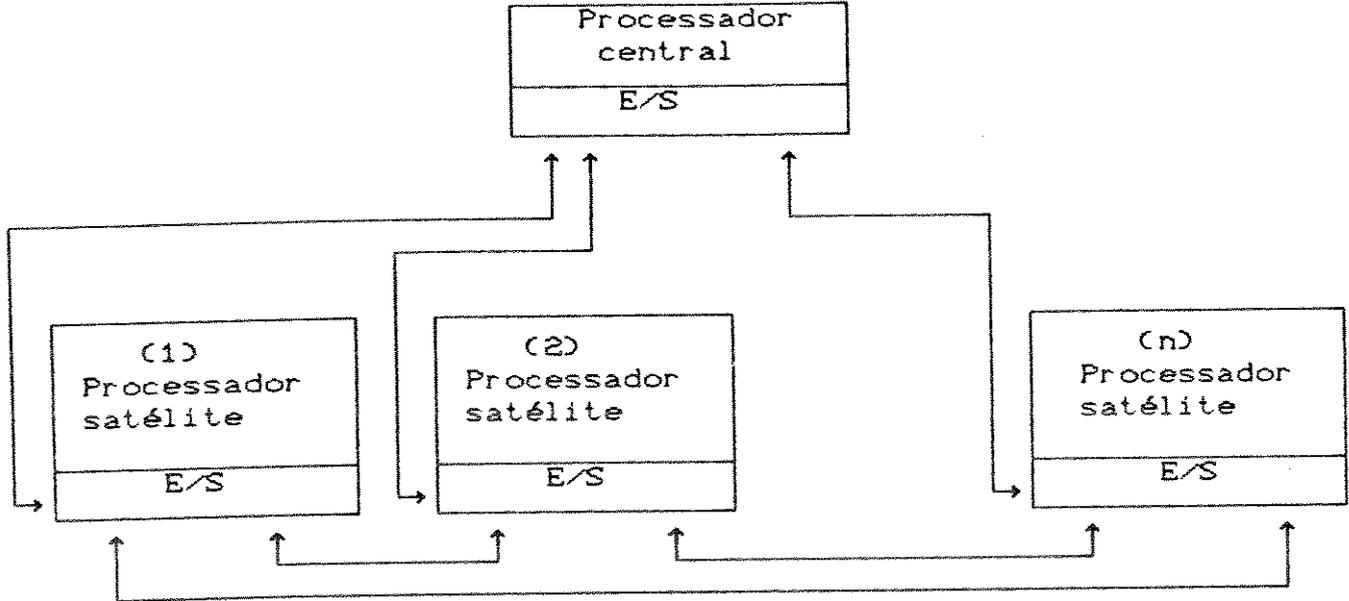


Figura 2.8 : Diagrama de Blocos de um sistema a Multimicroprocessadores

ARQUITETURA PROPOSTA POR BESTAOU

A metodologia de controle proposta por [Bestaoui 1989] utiliza controle hierárquico e observadores descentralizados. A estrutura proposta está apresentada na Figura 2.10, a metodologia utilizada é dividir o cálculo de dados das juntas necessários para as Equações de Newton-Euler e alocar processadores dedicados nas juntas para realizar a observação e controle destas juntas.

O microprocessador central sincroniza a operação de todos os microprocessadores e calcula os torques nominais usando as equações de Newton-Euler. O período de amostragem do processador central pode ser maior do que a dos processadores locais, permitindo-se que se consiga calcular o modelo matemático centralizado do manipulador, usando posições, velocidades e acelerações das juntas.

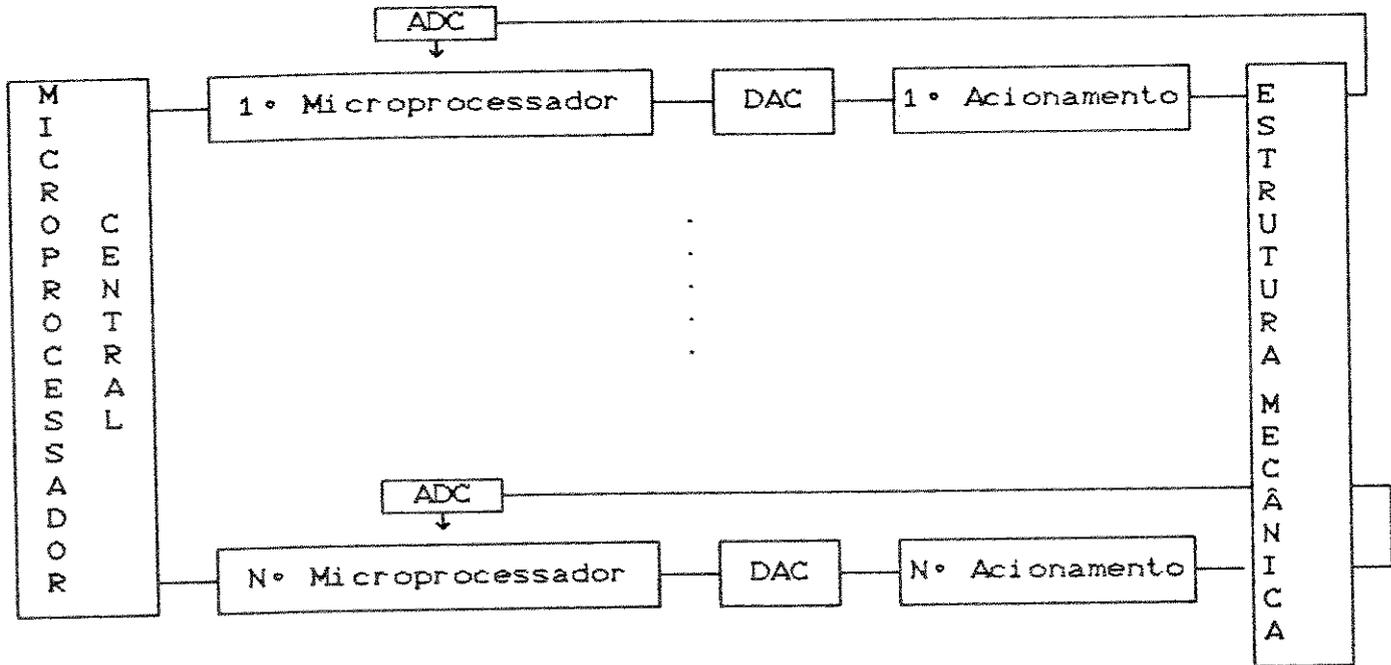


Figura 2.10 : Estrutura Proposta por Bestaoui

2.5 ARQUITETURA PROPOSTA NESTE TRABALHO

Nota-se nas arquiteturas apresentadas uma semelhança em alguns pontos:

- Proposição da arquitetura a multimicroprocessadores como um meio de aumentar a capacidade de processamento.
- Dividir a tarefa de controle do manipulador em subtarefas de modo a obter processamento paralelo.
- Utilizar processadores dedicados ao controle das juntas.

Antes de iniciarmos a apresentação da estrutura proposta para o controlador é importante citar algumas considerações sobre o sistema no qual este se insere.

A estrutura mecânica apresenta cinco graus de liberdade e um efetuator com um movimento de preensão de objetos com controle de esforços, utiliza somente juntas rotacionais. Os servo-acionadores são motores de corrente contínua a imã permanente acionados por meio de recortadores em quatro quadrantes, topologia em ponte H.

Foi previsto a possibilidade de acionamento destes recortadores pela utilização de sinais analógicos ou digitais. Os sensores das juntas não estão definitivamente definidos e podem ter saídas na forma digital (resolvers, encoders, etc..) como na forma analógica (tacômetros, potenciômetros, etc..).

O manipulador em questão tem finalidades didáticas (aulas de controle) e científicas (estudo de estruturas e controle de juntas, de tarefas e geração de referências para controle de trajetórias). Podendo, outrossim ser utilizado para operações industriais.

Com as premissas anteriores optou-se pela utilização de uma estrutura hierárquica de controle a multimicroprocessadores. A estrutura escolhida está apresentada na Figura 2.10. A idéia básica é alocar placas processadoras para o controle de cada junta do manipulador liberando ao máximo o micro Mestre para as tarefas de interface homem máquina e geração de referências para controle de trajetórias. Outra consideração importante é a possibilidade de se acoplar até dez placas microprocessadas no barramento, podendo-se com isto construir placas com propósitos específicos, por exemplo, para um sistema de visão (futura expansão).

O objetivo principal deste trabalho é projetar, construir e testar a estrutura do denominado CPR1 (Controlador Programável de Robôs 1), representado pelos blocos dentro das linhas pontilhadas na Figura 2.10.

O controlador pode ser subdividido em três partes principais, Sistema Mestre, Barramento Expandido e Placas microprocessadoras dedicadas com finalidades específicas.

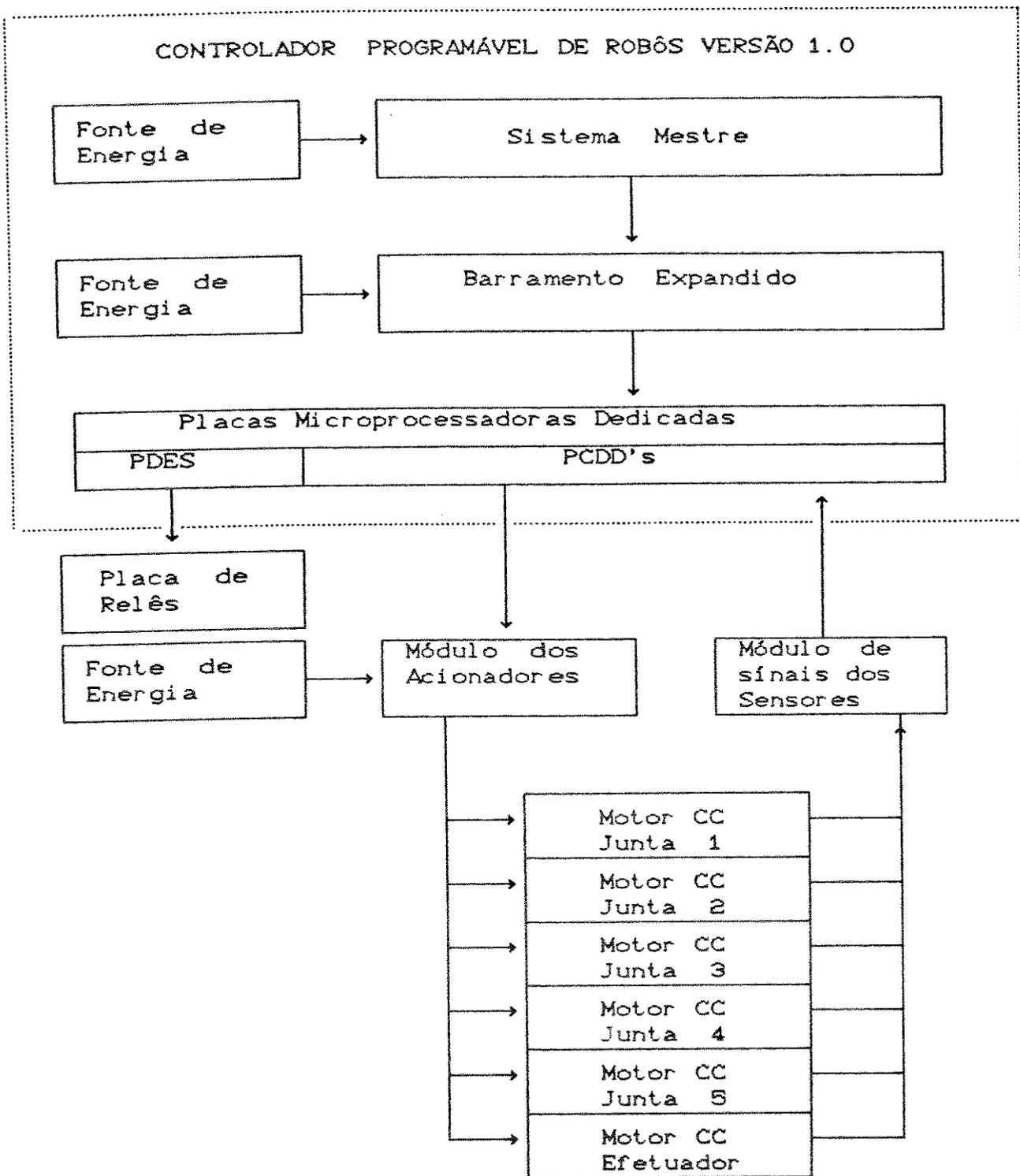


Figura 2.10 : Diagrama de Blocos do CPR1

ESTRUTURA DO MICRO MESTRE

O diagrama de blocos do Sistema Mestre está apresentado na Figura 2.11, este sistema é composto basicamente dos periféricos necessários à interface homem máquina :

- Unidades de disco rígido e flexível para armazenamento de arquivos e programas;
- Expansão de memória RAM para armazenamento de dados em massa de forma rápida;
- Placa de interface para geração e controle dos sinais do barramento.

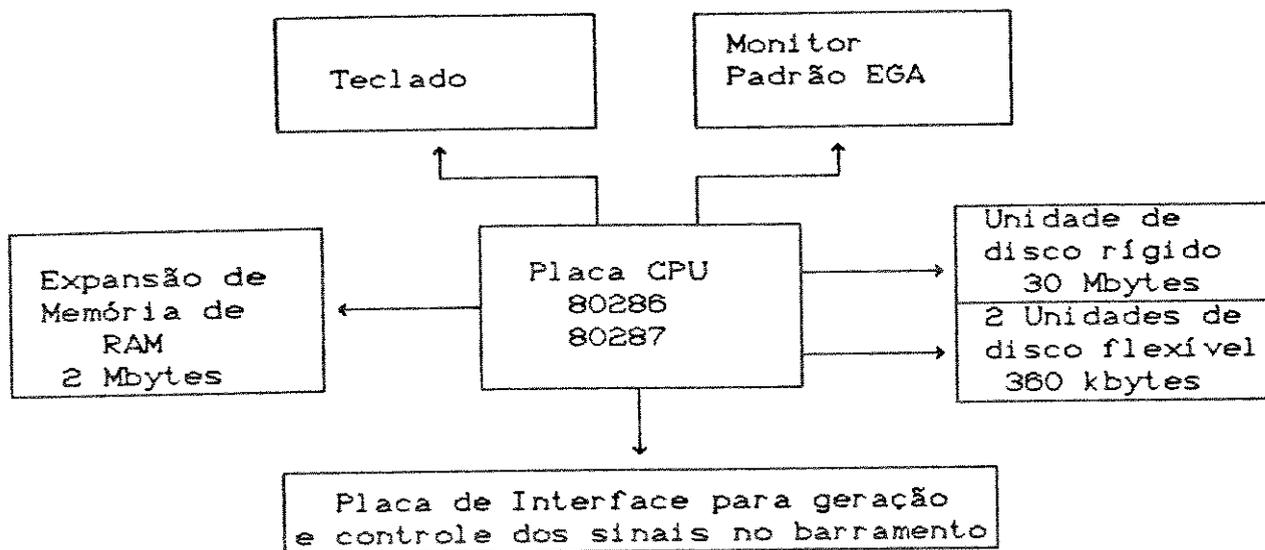


Figura 2.11 : Diagrama de Blocos do Micro Mestre

As funções básicas deste subsistema são:

- (1) Interface homem-máquina - Para que possamos utilizar efetivamente o manipulador devemos elaborar uma interface simples, amigável e eficiente para o operador.

- (2) Implementar e testar algoritmos de controle - Como este sistema tem fundamentalmente finalidades didáticas é importante a possibilidade de implementar automaticamente diferentes algoritmos de controle de juntas e ter ferramentas disponíveis para analisar seu desempenho. Para tanto o sistema apresenta a possibilidade de enviar programas desenvolvidos em linguagem assembler para os controladores de juntas e uma ferramenta para captar dados e traçar curvas.
- (3) Controle de tarefas - Em aplicações gerais é importante a geração de referências para controle de trajetórias no espaço operacional do manipulador. Este subsistema é responsável também pela geração e envio de sinais de referência para as placas de controle, coordenando assim a sua utilização.
- (4) Analisar o desempenho da tarefa efetuada - Após o movimento o Mestre capta os dados resultantes nas juntas e monta, pelo cálculo do modelo geométrico inverso, a tarefa efetuada para análise.

2.5.2 ESTRUTURA DO BARRAMENTO

O barramento é um componente importante no desenvolvimento de sistemas distribuídos, pois é através dele que os processadores comunicam-se entre si. Existem muitos barramentos padrões para interligação destes tipos de sistemas, porém com as ferramentas disponíveis a utilização de um barramento padrão não atenderia os requisitos de velocidade necessários para operar todo o sistema em tempo real.

A filosofia básica do sistema consiste em dar ao micro Mestre, total autonomia sobre o funcionamento das placas dedicadas. Existem sinais de requisição de barramento para cada placa acoplada a ele. Quando o sistema Mestre necessitar enviar dados ou programas para estas placas deve enviar um sinal de requisição de barramento ("BUSREQ") e imediatamente passar a acessar a memória e dispositivos de E/S da placa em questão.

Este barramento pode ser dividido em:

- Sinais de controle
- Sinais de endereços
- Sinais de dados
- Sinais de Alimentação

A Figura 2.12 apresenta os componentes do barramento. Os sinais de controle, de dados e de endereços provenientes do micro Mestre formando o barramento auxiliar, são transmitidos através de "Flat Cables" à placa de Amplificação de Barramento, esta placa quando acoplada corretamente no conector, transmite os sinais fortificados para o barramento principal.

Uma fonte de energia fornece os sinais de alimentação para os microcomputadores diretamente por linhas do barramento principal.

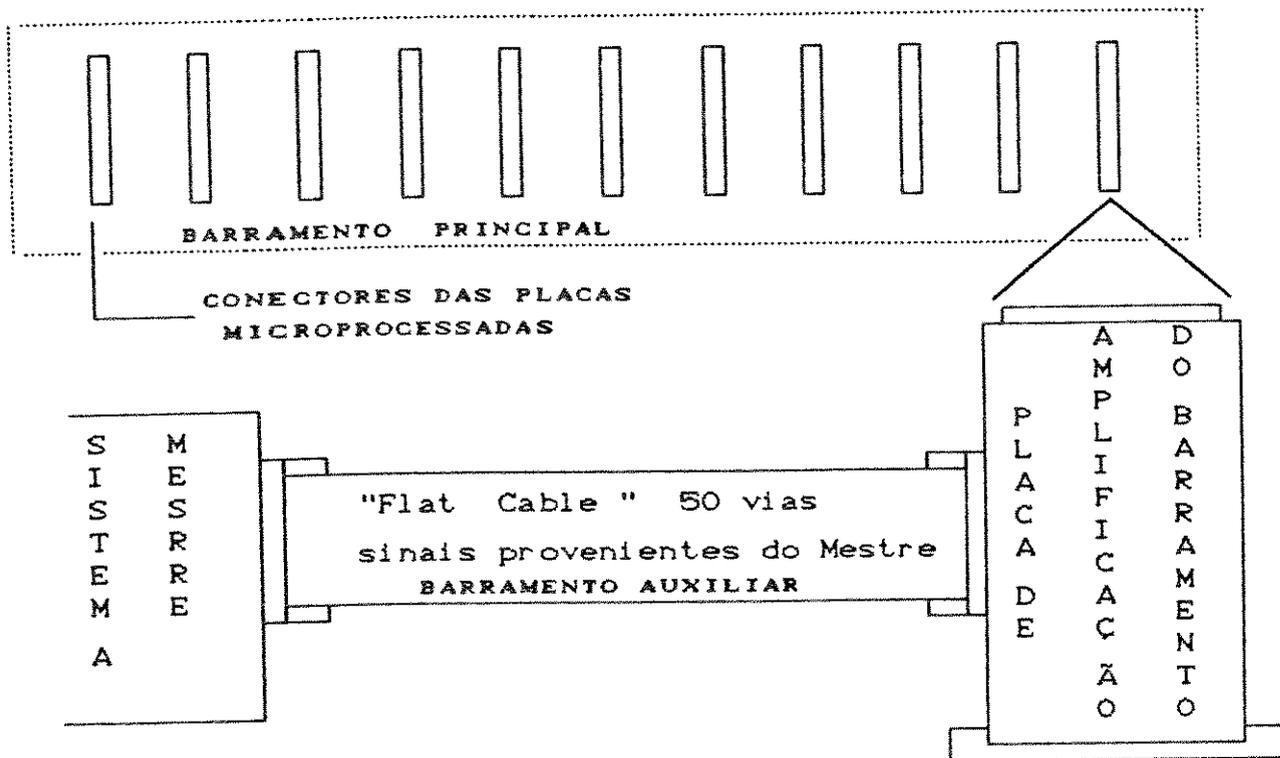


Figura 2.12 : Componentes principais do barramento

2.5.3 PLACAS MICROPROCESSADORAS DEDICADAS

Estas placas são acopladas ao barramento principal de acordo com o esquema apresentado na Figura 2.13.

Foram desenvolvidas duas placas:

- Placa de Controle Digital Direto (PCDD)
- Placa de Entradas e Saídas Diditais (PDES)

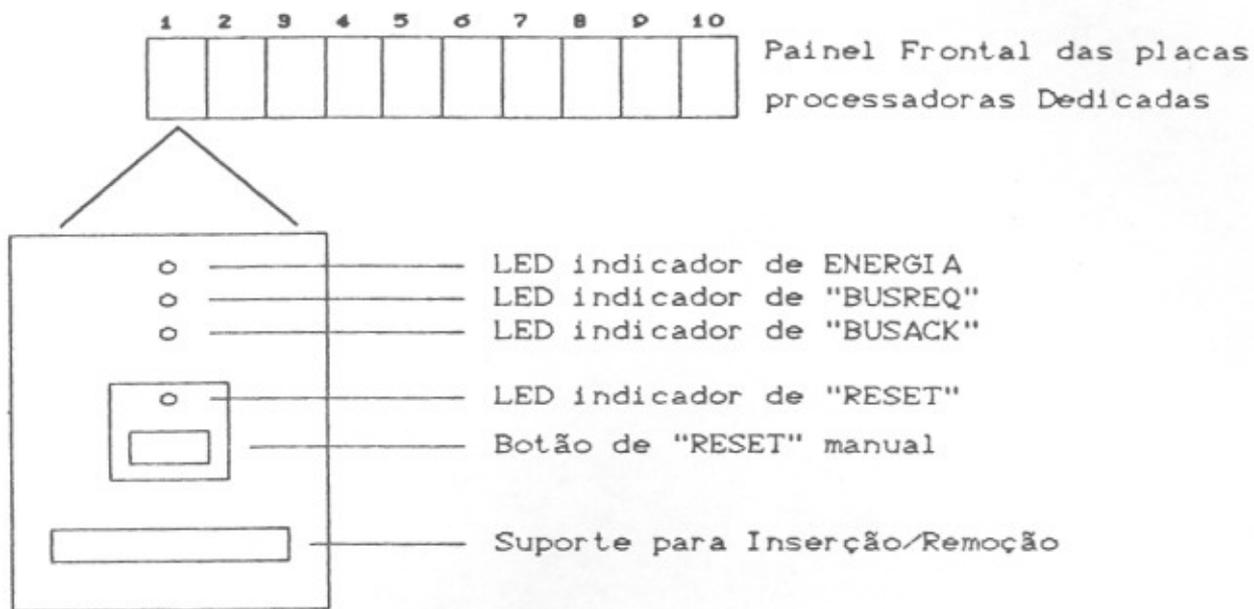


Figura 2.13 :Painel frontal das Placas processadoras

As funções das Placas de Controle Digital Direto são:

- Aceitar a programação e controlar a junta de acordo com sinais de referências do Micro Mestre através de um programa monitor e de acordo com o algoritmo escolhido para a junta;
- Armazenar variáveis de interesse para análise de desempenho;
- Converter a realimentação dos valores das variáveis das juntas da forma analógica para digital, quando necessário;
- Calcular o ângulo de comutação para os recortadores.

As funções da placa de Entradas e Saídas Digitais são:

- Através de um programa monitor aceitar programação via Micro Mestre;
- Controlar o ambiente do robô segundo algoritmo enviado, este programa é responsável pelo sistema de segurança do manipulador monitorando os sensores que indicam a invasão do espaço operacional do manipulador;
- Avisar o sistema mestre quando ocorrer problemas nas juntas ou no ambiente de trabalho.

2.6 MICROCOMPUTADORES APLICADOS EM CONTROLE DE PROCESSOS

A seguir vamos fazer um breve estudo do uso dos microcomputadores como controladores de processos em tempo real dirigido especificamente para o objetivo proposto no item 2.5 para em seguida utilizarmos estes conceitos no projeto do sistema proposto.

Com o advento dos microprocessadores no início da década de setenta iniciou-se uma nova era no campo de controle de processos, microcomputadores passaram a ser intensivamente utilizados em tarefas de controle de uma maneira geral, substituindo com grandes vantagens técnicas e econômicas os controladores analógicos até então utilizados em larga escala.

A primeira geração de microcomputadores usava uma arquitetura simples de 4 bits, que foram rapidamente substituídos pelos micros de 8 bits melhorando o desempenho de forma substancial, gradualmente apareceram os de 16 bits e atualmente já estão disponíveis os de 32 e 64 bits.

Nos sistemas de controle o microcomputador ou controlador digital é o responsável pelo processamento do sinal de erro para a geração do sinal de controle e tomada de decisões para melhoria do desempenho dos sistemas de proteção, segurança e controle de energia ("no break"). O elemento principal dos microcomputadores é a Unidade Central de Processamento "CPU" ou microprocessador. Além da "CPU" os microcomputadores aplicados ao controle de processos ou simplesmente microcontroladores digitais, geralmente apresentam os seguintes elementos:

- Gerador de Relógio
- Amplificadores de barramento de endereços e de dados
- Memória ROM ou EPROM
- Memória RAM
- Controlador de Interrupção
- Entradas e Saídas Digitais
- Entradas e Saídas Analógicas (D/A's e A/D's)
- Interface de Comunicação
- Temporizadores Programáveis

Na Figura 2.14 apresentamos um diagrama de blocos típico de um microcomputador. O microprocessador está ligado aos outros elementos através dos barramentos de endereços, de dados e do barramento de controle.

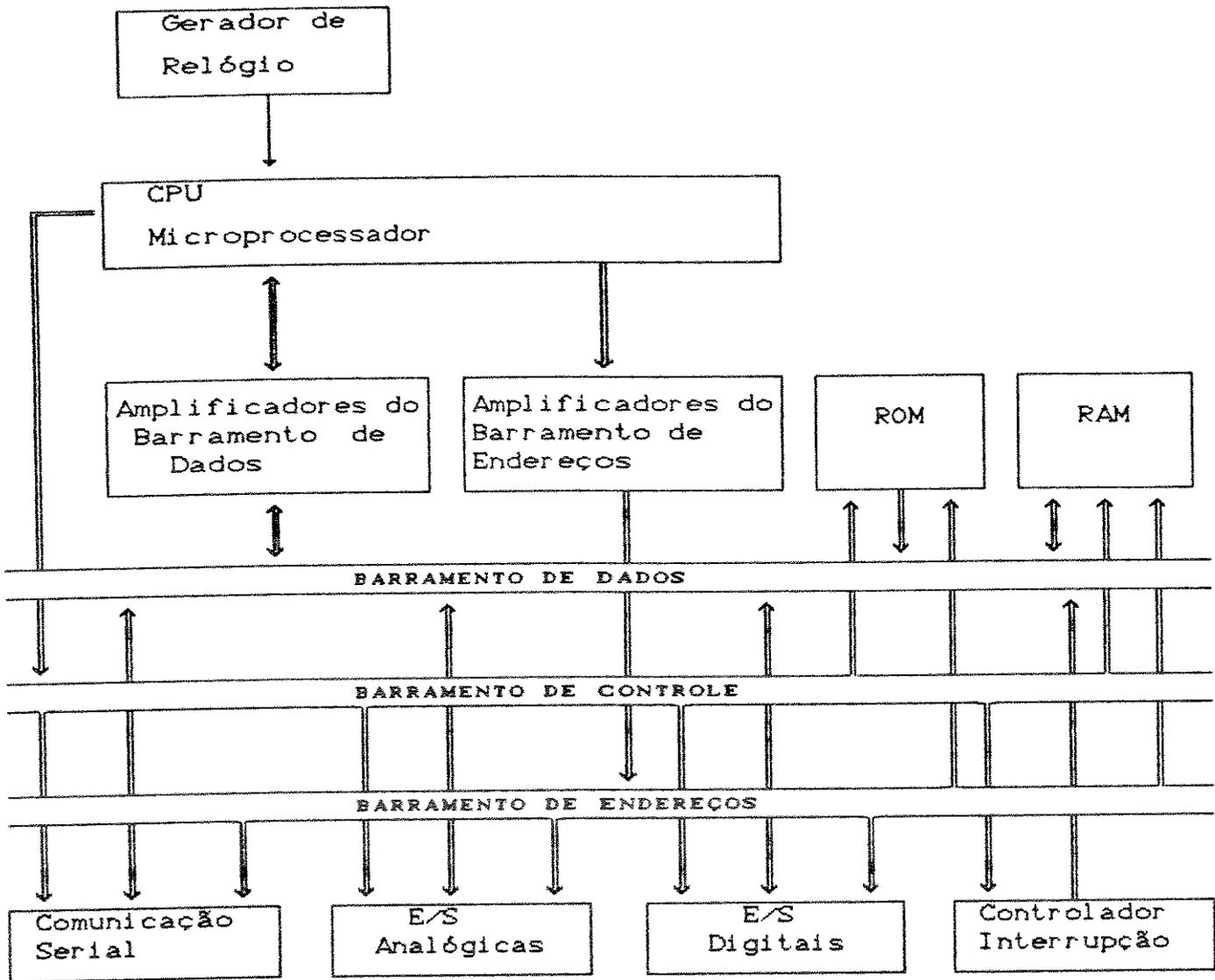


Figura 2.14 : Diagrama de Blocos de um microcomputador aplicado ao controle de processos

Algumas linhas de endereços são decodificadas para gerar a habilitação de elementos periféricos denominados dispositivos de Entrada e Saída, este bloco do CPR1 foi denominado de "Seletor de E/S".

Como os microprocessadores aplicados em controle geralmente se encontram em ambientes ruidosos (fábricas) é importante amplificar os sinais dos barramentos de endereços, de dados e de controle. No CPR1 este bloco foi denominado de "Amplificadores de Barramentos da CPU".

Os barramentos dos microprocessadores estão interligados ao Micro Mestre pelo barramento de comunicação, para evitar conflitos de dados nos barramentos, optou-se por desacoplar os sinais do barramento de comunicação por meio de amplificadores "tri-state" denominados de "Amplificadores de Barramento Externo". Portanto os sinais para os componentes gerais das placas processadoras são controlados por meio dos amplificadores de barramentos da CPU e externo. Existe um circuito lógico que evita que estes dois circuitos sejam acionados simultaneamente, gerando conflito entre o barramento do Micro Mestre e o barramento do microprocessador local.

O circuito "Gerador de Relógio" tem a função de sincronizar a operação do microprocessador, além de fornecer pulsos para temporizadores e conversores analógico-digitais. Este circuito define a velocidade de processamento da CPU, os dispositivos acoplados aos barramentos devem ter tempo de acesso compatíveis para que as operações do microprocessador possam ser realizadas com sucesso.

A memória PROM ou EPROM é utilizada para armazenar os programas a serem executados pelo microprocessador. Nos microcontroladores digitais dedicados (MDD) geralmente esta memória é pequena quando comparada com computadores de propósitos gerais. O CPR1 utiliza 16 kbytes de EPROM reservadas para as rotinas de interrupção, programa monitor e subrotinas básicas utilizadas pelo programa principal que é armazenado pelo Micro Mestre (MMD) na memória RAM dos processadores dedicados.

A memória RAM além de armazenar o programa principal reserva áreas de memória para dados enviados pelo Mestre e áreas de armazenamento de variáveis geradas pela própria junta (realimentação dos valores das variáveis dos sensores, sinal de erro, sinal de controle calculado para velocidade, posição e esforço, etc).

As realimentações das variáveis de estado do sistema e os algoritmos de controle devem ser cíclicos com períodos pré-estabelecidos. Para que isto aconteça pode ser necessária a inclusão de temporizadores programáveis na estrutura do microcontrolador digital dedicado. No CPR1 existem três canais de temporização programáveis internos e três externos gerados pelo Micro Mestre e que podem ser utilizados para sincronizar a operação de todo o sistema.

O controlador de Interrupção para a operação normal do processador e direciona-o para executar uma rotina que requer uma atenção urgente. No CPR1 temos oito canais de interrupção, três reservados para as temporizações internas e três para as temporizações externas, um para o conversor Analógico-Digital e um para expansão.

CPU UTILIZADA

A CPU é composta geralmente de um microprocessador, no caso do CPR1 foram utilizados microprocessadores Z80 da Zilog de 8 bits, sendo que as razões para esta escolha são seu baixo custo, simplicidade e facilidade de aquisição e com desempenho satisfatório para os objetivos propostos, entretanto pode-se utilizar outros processadores no projeto de novas placas desde que se preserve a filosofia de comunicação implementada.

MEMÓRIA SOMENTE DE LEITURA (ROM/EPROM)

Como mencionado anteriormente, a ROM armazena o programa a ser executado pelo microprocessador. Esta área de memória é eficaz para armazenar tabelas, coeficientes e valores que não são alterados durante a execução do programa, pois a principal característica deste tipo de memória é que ela é não volátil, ou seja, a memória não perde o seu conteúdo com a retirada do sinal de alimentação. Nos estágios iniciais de desenvolvimento um programa pode necessitar constantes alterações e portanto, deve-se utilizar um outro tipo de memória denominada EPROM, a qual tem as mesmas características da memória ROM com a diferença de poder ser apagada quando exposta a luz ultra violeta. As placas processadoras do CPR1 utilizam memória EPROM de 16 kbytes.

MEMÓRIA DE ACESSO ALEATÓRIO (RAM)

A memória de acesso aleatório normalmente armazena os dados gerados pela execução do programa, além desta utilização o CPR1 utiliza esta memória para carregar programas com algoritmos do controle de juntas via Micro Mestre (MMD).

DISPOSITIVOS DE ENTRADAS E SAÍDAS DIGITAIS

A comunicação entre o microprocessador e o mundo externo é realizada por meio de dispositivos periféricos de entrada e saída. Estas portas digitais são geralmente programáveis e acessadas através de instruções "IN" e "OUT". São reservadas três portas digitais programáveis de E/S em cada placa controladora de juntas do CPR1.

DISPOSITIVOS DE ENTRADA E SAÍDA ANALÓGICOS

O microprocessador pode receber ou enviar dados analógicos para o mundo externo, para tanto existem os Conversores Analógico-Digitais (ADC) e os Digital-Analógicos (DAC). O número de "BITS" da palavra utilizada pelos conversores determina a precisão do sinal resultante. Uma observação importante é que o tempo de conversão da ADC deve ser pequeno quando comparado com o período de amostragem do sinal. As placas de controle de juntas apresentam conversores A/D e D/A de 10 bits e tempo de conversão do ADC de 6 [us].

CONTROLADOR DE INTERRUPÇÃO

O controlador de interrupção gerencia os sinais de interrupção dos componentes externos. Quando o microprocessador recebe um sinal de interrupção, suspende a sequência normal de operação, executa a subrotina de interrupção e retorna ao programa principal. O controlador de interrupção pode gerenciar um número pré-determinado de pedidos baseando-se em uma lógica de prioridade. No CPR1 os temporizadores programáveis funcionam conectados ao controlador de interrupção, isto torna o sistema flexível pois, tem-se controle da temporização desejada e pode-se programar o nível de prioridade desta interrupção.

2.7 VANTAGENS E LIMITAÇÕES DOS MICROPROCESSADORES

Os sistemas de controle digital a microcomputadores, segundo [Bose 1984], tem as seguintes vantagens sobre seu correspondente sistema analógico:

- Redução do custo do "Hardware"

A simplificação do "Hardware" de controle e a correspondente redução do custo quando comparado com o sistema de controle analógico é a principal vantagem dos sistemas de controle digitais. Esta tendência aumenta com a melhoria da capacidade de processamento e com a integração de novas funções nos microprocessadores. A redução no tamanho e no consumo de energia são vantagens adicionais.

- Aumento de Confiabilidade

A confiabilidade do sistema ou o MTBF (Tempo médio entre falhas), é melhorada utilizando-se microcomputadores no controle de processos. Segundo ele a confiabilidade dos sistemas com circuitos integrados é consideravelmente maior que os circuitos analógicos que contém uma grande quantidade de componentes eletrônicos discretos. A confiabilidade pode ser melhorada com refrigeração ou pelo uso de componentes com especificação militar.

- Eliminação do "DRIFT"

O processamento digital de sinais diminui os efeitos de DRIFT (variação dos componentes com o tempo e condições ambientais) e a variação de parâmetros resultante. Este problema é relevante nos sistemas analógicos.

- **Compatibilidade com Controle Digital Hierárquico**

O controle de processos por microcomputadores permite compatibilidade com sistemas de mesmo nível de controle e com sistemas de controle de níveis superiores. Num ambiente de fábrica automatizada a comunicação entre microcomputadores de controle locais e o computador central torna-se importante para controle e diagnóstico.

- **"Hardware" Universal**

Um "Hardware" de uso geral pode ser projetado para uma certa classe de sistemas onde com modificações no software atende-se as especificações de uma sub-classe. A vantagem adicional do "software" de controle é que este pode ser alterado ou melhorado de acordo com a necessidade de desempenho.

- **Possibilidade de Diagnóstico**

Uma outra vantagem do controle por microcomputador é que pode-se projetar um software de diagnóstico, o qual pode ser utilizado por um técnico com conhecimentos limitados. Com o conhecimento do problema reduz-se o tempo de manutenção do sistema, além disso, podem ser utilizados aquisição de dados, monitoração de sinais, aviso de advertência, e "displays".

Algumas das desvantagens do uso de microcomputadores em controle de processos, são também apontadas por [Bose 1984] :

- **Atraso computacional**

O programa nos microcomputadores funciona de uma maneira serial o que implica em um tempo de processamento resultante considerável, diferentemente do "hardware" analógico que funciona de uma maneira paralela. Atrasos computacionais elevados causam uma deterioração no desempenho e podem causar problemas de instabilidade nos sistemas de controle em malha fechada. A capacidade computacional dos microcomputadores disponíveis vem aumentando com o passar do tempo e a tendência é que continue a melhorar. O uso de processamento paralelo pode melhorar consideravelmente a velocidade de execução.

- **Erro de Quantização**

Os sinais nos sistemas físicos são analógicos e quando interfaceados com os microcomputadores por meio de um conversor A/D ocorrem erros de quantização resultantes da conversão de uma grandeza contínua para uma grandeza descontínua. A distorção do sinal devido a quantização e o tempo de amostragem pode ser evidenciada quando um sinal de tensão na forma senoidal é aplicado no conversor A/D e imediatamente amostrado de volta em um conversor D/A. Pode-se minimizar os erros de quantização aumentando-se o tamanho da palavra utilizada pelos conversores.

- **Dificuldade de Acesso aos sinais digitais**

Nos sistemas analógicos pode-se através de um osciloscópio monitorar diversos sinais de controle. O controle digital por microcomputadores não fornece fácil acesso aos sinais de software para monitoração e análise de falhas. O sistema pode necessitar do desenvolvimento de instrumentos de diagnóstico e análise de falhas para esta finalidade.

- **Custo elevado do desenvolvimento de "Software"**

O desenvolvimento do "Software", especialmente em linguagem de máquina pode ser muito demorado, e portanto, com alto custo de desenvolvimento. O custo adicional pode ser justificado pelas vantagens apresentadas sobre os sistemas de controle analógicos.

2.8 UTILIZAÇÕES DOS MICROCOMPUTADORES EM CONTROLE DE PROCESSOS

São muitas as aplicações que podem ser efetuadas por meio de microcomputadores. Algumas das aplicações definidas para as placas processadoras do CPR1 são:

- Controle em malha fechada
- Geração do sinal de comutação para os recortadores
- Controle de falhas e proteção
- Monitoração de sinais e avisos de advertência
- Aquisição de dados

CONTROLE EM MALHA FECHADA

Dependendo da aplicação e das especificações de desempenho devemos implementar um controle em malha fechada, ou seja, uma ou mais variáveis do processo são realimentadas. Nos sistemas digitais de controle estas variáveis realimentadas são discretas no tempo ou contínuas porém amostradas para a efetivação da realimentação o que juntamente com o atraso computacional resultante do processamento do algoritmo de controle implica em atrasos de transportes, com consequências instabilizantes, além disso, os efeitos do chaveamento dos dispositivos semicondutores de potência podem requerer para o circuito uma análise não-linear do sistema, tanto por um modelo contínuo de variáveis médio, como por um modelo discreto.

Os efeitos da discretização no tempo podem ser desprezados se os intervalos de amostragem e chaveamento forem pequenos comparados com o tempo de resposta do sistema. Nestes casos uma análise de sistemas lineares baseado na Transformada de Laplace pode ser utilizada.

Para o projeto de microcomputadores aplicados em controle deve-se saber de antemão quais serão as variáveis realimentadas e a forma que estas se encontram (Analógica ou Digital). Por exemplo no controle de motores de corrente contínua pode-se ter três variáveis de estado para se efetuar as realimentações de :

- Posição (digital medida com detector ótico incremental);
- Velocidade (analógica medida com tacogerador);
- Corrente (analógica medida com sensor de efeito "Hall").

Entretanto se a velocidade fosse medida com detector ótico absoluto seria uma entrada digital. Para este exemplo devemos reservar portas digitais de entrada para a posição e canais de entrada analógicos para a velocidade e a corrente.

Outra informação prévia deve ser o tempo de resposta do sistema ao ser excitado, pois com esta informação pode-se calcular o tempo máximo de conversão, de processamento dos algoritmos de controle, e quando interligado em um sistema a multimicroprocessadores, o tempo máximo de comunicação com os outros processadores.

A precisão requerida torna-se importante para a escolha do tamanho da palavra dos conversores (ADC e DAC) e do tipo de numeração utilizado (ponto fixo ou ponto flutuante).

Geralmente os algoritmos de controle devem operar de maneira cíclica com períodos fixos, isto pode requerer que o microcomputador disponha de temporizadores programáveis acoplados em controladores de interrupção para controlar e coordenar o período de amostragem dos sinais.

GERAÇÃO DE SINAIS DE COMUTAÇÃO PARA OS RECORTADORES

Quando o acionamento utilizar Modulação por Largura de Pulso, a geração dos sinais comutação pode ser realizada pelo microcomputador de controle ou por um circuito digital dedicado onde uma palavra digital é transformada em um sinal pulsado. O CPR1 envia uma palavra de comando de 12 bits a um circuito digital que converte este sinal para os recortadores.

CONTROLE DE FALHAS E MONITORAÇÃO

A monitoração de parâmetros do sistema é conveniente pois pode-se prevenir e consertar falhas de operação. Por exemplo, os manipuladores em geral monitoram a posição das juntas para verificar se estas não excedem valores limites.

Se algumas variáveis do sistema começarem a ultrapassar um valor limite, sinais áudio-visuais de perigo podem ser gerados. Se o operador falhar na realização da ação apropriada o sistema automaticamente pode desligar-se.

AQUISIÇÃO DE DADOS

Aquisição de dados é uma função importante de um microcomputador num sistema de controle digital, podendo ser utilizada para as seguintes finalidades:

- Medição dos sinais de realimentação nos sistemas em malha fechada.
- Armazenamento de dados medidos para realizar análise dinâmica de desempenho do sistema com ajuda do microcomputador responsável pela interface homem máquina.
- Armazenamento de dados para uma avaliação futura.
- Armazenamento de dados para análise de falhas.
- Dados para propósitos de monitoração.

TESTE E DIAGNÓSTICO

Um microcomputador pode ser utilizado para uma grande variedade de testes, os dados capturados no sistema de aquisição de dados podem ser utilizados, por exemplo, para calcular curvas de desempenho, gráficos, ou mesmo para verificar a integridade de todo o sistema.

2.8 METODOLOGIA DE PROJETO DE SISTEMAS DE CONTROLE DIGITAL

Após as especificações de desempenho serem detalhadas pode-se iniciar o projeto do sistema de controle. Geralmente o desenvolvimento envolve as seguintes etapas:

- Projeto e análise do sistema global
- Projeto do "Hardware"
- Projeto do "Software"
- Integração, reformulação, correção de problemas, testes de desempenho

PROJETO E ANÁLISE DO SISTEMA GLOBAL

A estrutura do sistema de controle é construída de acordo com especificações de desempenho do sistema. Todas as funções de controle são identificadas, analisadas detalhadamente e finalmente projetadas.

Nos sistemas de acionamento por Recortadores, as funções incluem controle por realimentação, geração do sinal de comutação para o Recortador, captação dos sinais de realimentação, controle sequencial, monitoração de sinais, proteção e diagnóstico.

Nesta fase todos os subsistemas devem ser projetados e uma análise deve ser realizada para verificar a integração destes subsistemas. Por exemplo, se o sensor de posição tem uma faixa de medidas de ± 10 Volts, no projeto das placas de controle não se pode especificar canais de entrada de ± 5 Volts.

PROJETO DO "HARDWARE"

O "Hardware" global geralmente inclui um ou mais microcomputadores projetados com finalidades de controle de processos de acordo com as características da Figura 2.14.

O microprocessador a ser utilizado deve ser selecionado após a análise do sistema. Alguns dos critérios de seleção são:

- Tamanho da palavra
- Tempo de ciclo de instruções
- Capacidade de multiplicação/divisão
- Sistema de suporte de desenvolvimento

O tamanho da memória RAM e ROM devem ser escolhidas com uma margem de segurança e determinadas pelo tamanho do programa e banco de dados requerido.

Os sinais de entrada e saída que podem consistir de sinais analógicos, digitais e lógicos devem ser definidos. Deve-se eliminar ruídos dos sinais de entrada, e sinais lógicos de contadores e chaves devem ter um circuito de "debounce" para eliminar ruídos resultantes do chaveamento de um estado para outro.

Deve-se determinar o sistema de controle de interrupção, seus sinais de entrada e definir níveis de prioridade.

O projeto do CPR1 incluiu o projeto de diversos subsistemas, sistema Mestre, interface do barramento, barramento, placas de controle, placa de E/S digitais, recortadores, sensores, além de um detalhado estudo de integração destes subsistemas.

PROJETO DO "SOFTWARE"

Existem basicamente três categorias de "software" em sistemas de controle:

- (1) "Software" básico - Esta categoria é independente da aplicação, orientado para o uso eficiente dos recursos de computação, e geralmente é fornecido pelo fabricante. Por exemplo os sistemas monitores cuja função é interfacear os programas de aplicação do usuário e o "Hardware".

(2) "Software de Aplicação" - Esta categoria geralmente é desenvolvida pelo usuário e está relacionado diretamente com as especificações da aplicação.

(3) "Software" de Diagnóstico - Estes programas têm a função de testar todo o sistema e apresentar resultados com o diagnóstico dos problemas encontrados.

O desenvolvimento de programas envolve geralmente os seguintes passos:

Seleção da Linguagem

O controle de processos, especialmente as funções relacionadas com os algoritmos de controle em malha fechada são usualmente críticos em relação ao tempo de processamento, portanto, geralmente implementa-se estes programas em linguagem "Assembly". Esta linguagem é muito próxima da linguagem de máquina, tem um tempo de execução curto e ocupa um reduzido espaço na memória. Entretanto, o desenvolvimento de programas em "assembly" é muito demorado, tedioso e geralmente dispendioso.

O desenvolvimento de programas utilizando linguagens de alto nível como C, PASCAL e FORTRAN tem um curto período de desenvolvimento, entretanto a sua execução é lenta e ocupa mais memória.

Seleção do Tamanho da palavra

As especificações de resolução e o tamanho da palavra dos conversores A/D e D/A determinam o tamanho apropriado da palavra a ser utilizado.

Seleção do tempo de Amostragem

O microcomputador executa um programa de maneira sequencial, isto implica em um tempo finito para a execução dos programas, portanto, o período de amostragem deve ser especificado. O desempenho de um sistema digital se aproxima de um sistema analógico se o período de amostragem for curto e a resolução for alta.

Quando o período de amostragem for grande pode-se utilizar linguagem de alto nível, entretanto, isto causa um atraso maior e distorção no sistema de controle podendo causar instabilidade devido ao atraso de transporte inerente ao tempo de processamento.

A seleção do período de amostragem é determinada pela constante de tempo mais rápida do sistema. Como critério prático geralmente é utilizado um período de amostragem igual ou menor que $(\pi \cdot \tau)$, onde τ é a menor constante de tempo do sistema.

INTEGRAÇÃO, CORREÇÃO DE PROBLEMAS E TESTES DE DESEMPENHO

Após o desenvolvimento dos vários subsistemas é necessário esforços no sentido de integrá-los para obter o sistema global. A integração deve ser gradual com verificação de problemas e correção em cada subsistema acoplado. A correção de problemas no sistema global pode ser dificultada na fase de integração pela complexidade do sistema resultante, porém, após sua conclusão a correção de problemas fica facilitada justamente pela modularidade resultante.

Os testes de desempenho são importantes no sentido de verificar se o sistema atende as especificações anteriormente especificadas.

CAPÍTULO 3

CONTROLADOR PROGRAMÁVEL DE ROBÔS 1

3.1 DESCRIÇÃO DO SISTEMA

Como já dissemos o controlador projetado e construído para o manipulador foi denominado "Controlador Programável de Robôs 1 (CPR1)", e compreende todo o sistema computacional. Vale salientar porém que o CPR1 não compreende acionamentos e dispositivos de sensoreamento.

O CPR1 é um sistema de controle a multimicroprocessadores que apresenta um Mestre e placas processadoras acopladas ao sistema Mestre por meio de um barramento comum. O barramento suporta placas processadoras dedicadas ao controle das juntas e placas processadoras com finalidades gerais, por exemplo, placa dedicada a monitoração do espaço operacional e placa de gerenciamento de um sistema de visão. O sistema desenvolvido utiliza seis placas de controle distribuído e uma placa de interface de Entrada e Saída.

O diagrama de blocos deste sistema está apresentado na Figura 3.1. O sistema é composto de diversos módulos com finalidades específicas. Os principais módulos desenvolvidos são:

- Placa de Interface do PC (PIPC) - Tem por função a geração, recepção e coordenação dos sinais do barramento.
- Placa de Amplificação do Barramento (PABO) - Tem por função fortalecer os sinais provenientes do sistema Mestre, que são transmitidos por meio de "Flat Cables", na entrada do barramento principal.
- Placa de Controle Digital Distribuído (PCDD) - Tem por função o controle local das juntas.

- Placa Digital de Entrada e Saída (PDES) - Tem por função monitorar o espaço de trabalho e ferramentas do manipulador. Este módulo é composto por 96 portas programáveis de E/S.

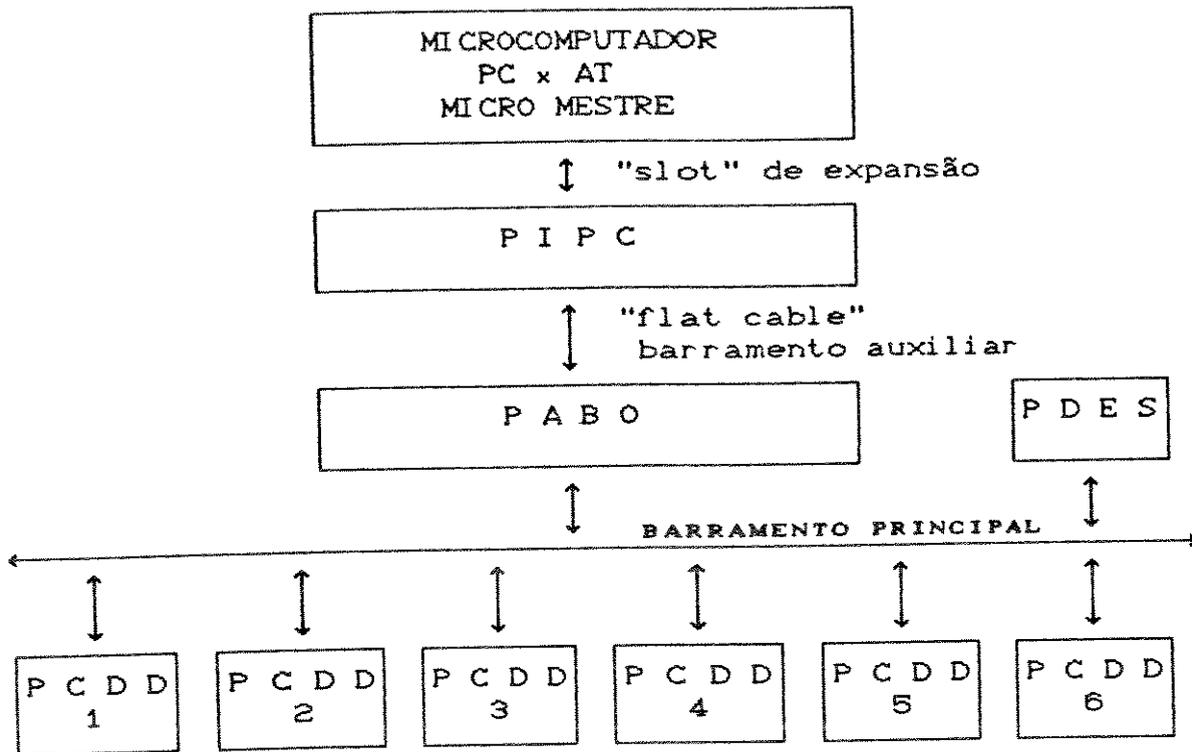


Figura 3.1 : Diagrama de blocos do CPR1

O Controlador divide-se nos seguintes módulos:

- Micro Mestre
- Placa de interface do PC (PIPC)
- Barramento auxiliar
- Placa de Amplificação do Barramento (PABO)
- Barramento Principal
- Placas de Controle Digital Distribuído (PCDD)
- Placa de Digital de Entradas e Saídas (PDES)

3.2 MICROCOMPUTADOR MESTRE

O microcomputador Mestre apresenta os seguintes componentes principais :

(1) Microprocessador : Utiliza o microprocessador 80286 INTEL. O 80286 apresenta 24 bits de endereços, 16 bits de dados, e dois modos de operação:

a) Modo de Endereçamento Real : Neste modo a memória física que pode ser acessada pelo microprocessador de 1 Mbyte. O microprocessador endereça a memória pela geração de 20 bits de endereços.

b) Modo Protegido : O modo de endereçamento protegido oferece aumento do espaço de endereçamento físico e de memória virtual, este modo de operação oferece 16 Mbytes de memória física.

CONFIGURAÇÃO DO MICRO MESTRE :

- Tipo de CPU - 80286
- Coprocessador - 80287
- Memória disponível para usuários- 640 kbytes
- Memória da placa EGA - 256 kbytes
- Memória estendida - 384 kbytes
- Expansão de memória - 2048 kbytes
- Saída serial e paralela
- Controladora de disco rígido e discos flexíveis
- controladora do monitor padrão EGA

MAPA DE MEMÓRIA :

000000	a	09FFFF	512 kbytes (placa do sistema)
000000	a	08FFFF	128 kbytes (placas processadoras)
0A0000	a	0DFFFF	256 kbytes placa EGA
0E0000	a	0FFFFFF	128 kbytes (ROM do sistema)
100000	a	15FFFF	384 kbytes memória estendida
15FFFF	a	45FFFF	2 Mbytes expansão de memória

(2) Monitor padrão EGA : Possibilita a elaboração da interface homem-máquina com a apresentação de telas, gráficos, configuração, etc..

(3) Impressora : Utilizada para imprimir variáveis, tabelas, arquivos, telas ou seja informações úteis com finalidade de armazenamento, análise, demonstrativos de desempenho, etc..

(4) Placa de Expansão de Memória : O microcomputador Mestre apresenta uma memória de expansão de 2 Mbytes, sendo utilizada para manusear grandes volumes de dados. Esta memória é de acesso rápido e portanto pode servir de banco de dados durante a execução de uma tarefa.

(5) Teclado : periférico utilizado para entrada de dados do operador.

(6) Discos : O sistema apresenta um disco rígido de 40 Mbytes e duas unidades de disco flexível.

3.3 PLACA DE INTERFACE DO PC (PIPC)

A placa de interface permite que o micro Mestre acesse totalmente a memória das PCDD's (32 kbytes), permitindo dois modos de operação:

- a) Modo real : Neste modo o micro Mestre opera no modo de endereçamento real do 80286, os endereços são formados pela junção do endereço da PCDD em questão com os 16 bits menos significativos do barramento de endereços. Neste modo deve-se utilizar somente posições de memória compreendidas entre 080000 e 09FFFF.
- b) Modo avançado : Neste modo o micro opera no modo de endereçamento protegido utilizando endereços entre 500000 e 5FFFFF.

Além do acesso a memória das PCDD's a placa de interface permite ao sistema Mestre acessar os dispositivos E/S das mesmas. Para tanto deve-se gerar o endereço da PCDD a qual deseja-se acessar, com isto o Mestre acessa o barramento da placa em questão, e realiza as operações de E/S como se fossem seus próprios dispositivos de E/S.

O diagrama de blocos da PIPC esta apresentado na Figura 3.2. A seguir passaremos a explicar cada módulo que compõe a PIPC.

BLOCO 1 : MÓDULO DECODIFICADOR DE ENDEREÇOS DE E/S

Este módulo é responsável pela geração dos sinais "IORD" e "IOWR" utilizados pelas PCDD's, assim como dos sinais de habilitação de seus próprios dispositivos de E/S.

Os sinais "IOWR" ou "IORD" (sinais de escrita e leitura para dispositivos de E/S das placas no barramento) são gerados quando houver, respectivamente, um sinal "IOW" ou "IOR", além de um endereço entre 120H e 13FH, no barramento do Micro Mestre. O esquemático deste módulo encontra-se na página A3 do Apêndice 1 da documentação do sistema CPR1.

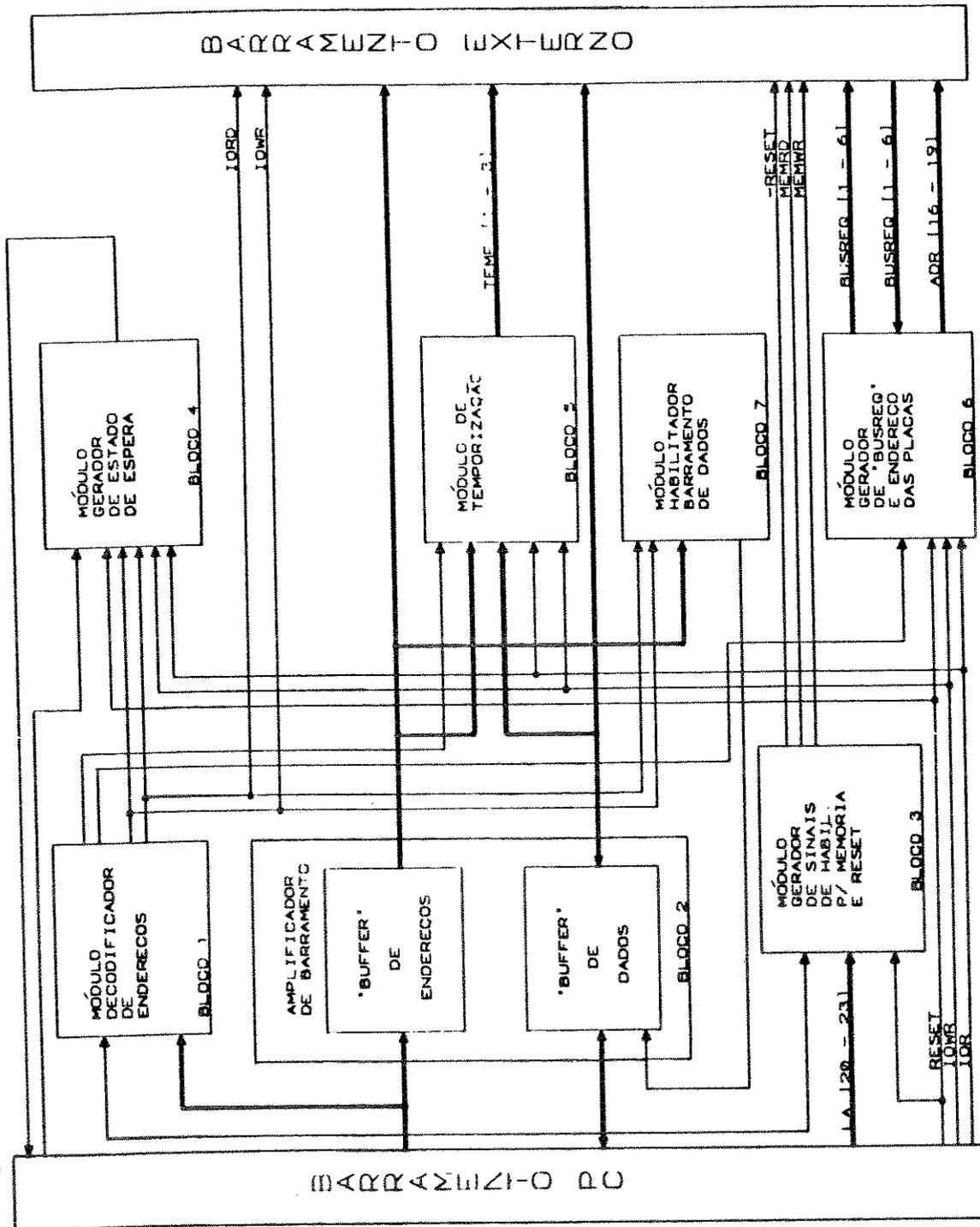


Figura 3.2 : Diagrama de Blocos da Placa de Interface do PC

BLOCO 2 : MÓDULO AMPLIFICADOR DE BARRAMENTO

Este módulo é composto por "Amplificadores tri-state" e destina-se a amplificação dos sinais de endereços e dados originados do barramento do Micro Mestre. Os sinais provenientes do barramento interno do PC são amplificados, pois além do fortalecimento dos sinais, protege-se o microcomputador de curtos elétricos e incidentes de uma maneira geral.

Os bits de endereços SA0 a SA15 são "bufferizados" através de "Amplificadores" permanentemente habilitados. Os bits SA16 a SA19 são habilitados através de um sinal gerado por "Software", estes bits formam o endereços das placas no barramento e somente podem ser transmitidos ao barramento interno da PIPC se o Micro Mestre estiver funcionando no modo de endereçamento protegido.

No modo de endereçamento real os bits SA16 a SA19 são gerados pela PIPC e somente são utilizados os endereços SA0 e SA15 do barramento do Micro Mestre.

Os bits de dados SD0 a SD7 são "amplificados" por um "buffer" bi-direcional. A direção do fluxo dos dados é controlada pelo sinal "IOR", e a habilitação pelo BLOCO 7 (módulo habilitador do barramento de dados).

O esquemático deste módulo encontra-se na página A4 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 3 : MÓDULO GERADOR DE HABILITAÇÃO DE MEMÓRIA E RESET

Este módulo é responsável pela geração dos sinais de "MEMRD" e "MEMWR" necessários para as operações de leitura e escrita nas placas processadoras acopladas ao barramento, e "RESET", a partir dos sinais de memória "SMEMR" e "SMEMW" e "RESET" enviados pelo Micro Mestre, quando o módulo estiver habilitado.

O sinal de "RESET" do Micro Mestre é ativo alto, enquanto que as PCDD's necessitam de um sinal ativo baixo. Portanto, torna-se necessária a complementação do sinal do Micro Mestre.

Para a geração dos sinais de "MEMRD" ou "MEMWR" existem duas situações a serem consideradas:

- a) Micro Mestre no modo de endereçamento real : Neste caso, os sinais são gerados quando o Micro Mestre enviar o sinal de "SMEMR" ou "SMEMW" e o sinal de habilitação do modo real, gerado na PIPC via software estiver ativo.
- b) Micro Mestre no modo de endereçamento protegido : Neste caso, é necessária a presença, no barramento do Micro Mestre, dos sinais "SMEMR" ou "SMEMW" e LA22 (vigésimo terceiro bit de endereço do Micro Mestre), em nível ativo, pois será utilizado o quinto megabyte do modo protegido da memória do Micro Mestre. Além disso, o sinal BALE (Habilitador do segurador de endereços) deve estar ativo e o sinal AEN (Habilitador de endereços) deve estar em nível lógico baixo indicando que o endereço atual é válido.

O esquemático deste módulo encontra-se na página A5 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 4 : MÓDULO GERADOR DE ESTADO DE ESPERA

Este módulo se destina a gerar sinais de "wait", que quando ativo, provoca o aumento do número de ciclos de máquina para as instruções de E/S, geradas pelo Micro Mestre. Tal procedimento permite a comunicação com dispositivos mais lentos. Este circuito é controlado por "Jumpers":

o o o o o o
o o o o o o
J1 J2 J3 J4 J5 J6

J1 ==> 1 ciclo adicional de relógio
J2 ==> 2 ciclos adicionais de relógio
J3 ==> 3 ciclos adicionais de relógio
J4 ==> 4 ciclos adicionais de relógio
J5 ==> 5 ciclos adicionais de relógio
J6 ==> 6 ciclos adicionais de relógio

O esquemático deste módulo encontra-se na página A6 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 5 : MÓDULO TEMPORIZADOR

A função deste módulo é a geração de sinais de temporização para o barramento. Com isto preve-se a possibilidade de sincronizar a operação de todas as placas processadoras. Este módulo pode ser dividido em três subsistemas:

- a) Gerador de Relógio : Como o temporizador programável utilizado não permite frequências de relógio acima de 2 MHz, torna-se necessária a geração de um sinal de relógio específico. Este sinal é gerado através de um circuito oscilador com um cristal de 4 MHz. Na saída do oscilador, o sinal tem sua frequência dividida por meio de um "flip flop JK" em modo "toggle".
- b) Temporizador : O temporizador programável é responsável pela geração dos sinais TEMP1 a TEMP3 (sinais de temporização para o barramento).

Endereços para programação do temporizador:

Contador 1	==>	0224H	(TEMP1)
Contador 2	==>	0225H	(TEMP2)
Contador 3	==>	0226H	(TEMP3)
Palavra de controle	==>	0227H	

Palavra de controle para o canal 0 ==> 34H
Palavra de controle para o canal 1 ==> 74H
Palavra de controle para o canal 2 ==> F4H

Para programar corretamente os canais de temporização deve-se proceder da seguinte forma:

- Escolher o canal desejado
- Programar a palavra de controle
- Carregar o byte menos significativo
- Carregar o byte mais significativo

c) Amplificador : É composto por um "Buffer" permanentemente habilitado. Sua função é amplificar os sinais de temporização para o barramento.

O esquemático do módulo está na página A7 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 6 : MÓDULO GERADOR DE "BUSREQ" E ENDEREÇOS DAS PLACAS

Este módulo é responsável pela geração dos sinais "BUSREQ1" a "BUSREQ10" (pedidos de requisição de barramento das placas processadoras), pela geração dos bits de endereço ADR16 a ADR19 (quando o Micro Mestre está no modo de endereçamento real) e dos sinais que indicam em qual modo o Micro Mestre está trabalhando. Os sinais de BUSREQ, e ADR16 a ADR19 são amplificados na entrada do barramento auxiliar.

A geração dos bits ADR16 a ADR19 neste módulo é necessária pois, no modo de endereçamento real, os sinais SA16 a SA19 são utilizados na decodificação de endereços válidos de instruções de memória do CPR1. Entretanto, o tamanho da memória que poderia ser acessada utilizando-se somente os endereços válidos (128 kbytes) é inferior ao necessário (10 placas apresentam 320 kbytes em conjunto). O artifício utilizado contorna este problema.

Endereços de E/S da PIA:

Porta A	==>	0220h
Porta B	==>	0221h
Porta C	==>	0222h
Palavra de Controle	==>	0223h

A porta A gera os sinais "BUSREQ1" a "BUSREQ6".

bit0	==>	"BUSREQ1"
bit1	==>	"BUSREQ2"
bit2	==>	"BUSREQ3"
bit3	==>	"BUSREQ4"
bit4	==>	"BUSREQ5"
bit5	==>	"BUSREQ6"
bit6	==>	NC
bit7	==>	NC

A porta B gera os sinais "BUSREQ7" a "BUSREQ10".

bit0	==>	"BUSREQ7"
bit1	==>	"BUSREQ8"
bit2	==>	"BUSREQ9"
bit3	==>	"BUSREQ10"
bit4	==>	NC
bit5	==>	NC
bit6	==>	NC
bit7	==>	NC

A porta C gera os sinais ADR16 e habilitação.

bit0	==>	ADR16
bit1	==>	ADR17
bit2	==>	ADR18
bit3	==>	ADR19
bit4	==>	NC
bit5	==>	NC
bit6	==>	Habilitação do modo de endereçamento
bit7	==>	Modo de endereçamento Protegido

O bit6 habilita os sinais de "MEMREQ" E "MEMWR" para as placas de controle.

bit6 = 0 Acesso a memória modo real

bit6 = 1 Acesso a memória modo protegido

O bit7 controla a geração dos bits ADR16 a ADR19.

bit7 = 0 geração via PIA.

bit7 = 1 geração pelo barramento PC.

Deve-se programar o dispositivo de E/S digital de modo que todas as portas funcionem como saídas.

O esquemático deste módulo está apresentado na página A8 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 7 : MÓDULO HABILITADOR DO BARRAMENTO DE DADOS

Este módulo é responsável pela geração do sinal de habilitação do barramento bidirecional de dados. Este sinal é ativo em nível lógico "0", e deve estar presente sempre que se deseja fazer um acesso aos endereços de memória ou E/S das placas processadoras.

O esquemático deste módulo encontra-se na página A9 do Apêndice 1 da documentação do sistema CPR1.

ENDEREÇOS DE E/S DAS PCDD's (ACESSO VIA MESTRE)

Para o acesso aos endereços de E/S das placas processadoras acopladas ao barramento, o Micro Mestre suspende a operação das CPU's das mesmas e assume esses endereços de E/S como se fossem seus próprios dispositivos. A seguir são apresentados os endereços dos dispositivos de E/S das placas processadoras para acesso via Micro Mestre.

ENDEREÇOS DE E/S DA PCDD

Os endereços de acesso aos dispositivos periféricos da Placa de Controle Digital Direto são:

Porta Digital A	==> 120H
Porta Digital B	==> 121H
Porta Digital C	==> 122H
Controle das Portas	==> 123H

Porta Saída D/A (D9 e D8)	==> 124H
Porta Saída D/A (D0 a D7)	==> 125H

Controle Canais Analógicos ==> 128H

Seleção	Endereço	Barramento de dados
Canal 0	128H	00H
Canal 1	128H	01H
Canal 2	128H	02H
Canal 3	128H	03H
Canal 4	128H	04H
Canal 5	128H	05H
Canal 6	128H	06H
Canal 7	128H	07H

Início de Conversão A/D	==> 129H
Leitura de A/D (D0 a D7)	==> 12AH
Leitura de A/D (D9 e D8)	==> 12BH

Canal 0 do temporizador	==> 12CH
Canal 1 do temporizador	==> 12DH
Canal 2 do temporizador	==> 12EH
Controle do temporizador	==> 12FH

Controle 1 de interrupção	==> 130H
Controle 2 de interrupção	==> 131H

ENDEREÇOS DE E/S DA PDES

Os endereços de acesso aos dispositivos periféricos da Placa de Entradas e Saídas Digitais são:

Porta Digital A	==> 120H
Porta Digital B	==> 121H
Porta Digital C	==> 122H
Controle das Portas	==> 123H

Porta Digital A	==> 124H
Porta Digital B	==> 125H
Porta Digital C	==> 126H
Controle das Portas	==> 127H

Porta Digital A	==> 128H
Porta Digital B	==> 129H
Porta Digital C	==> 12AH
Controle das Portas	==> 12BH

Porta Digital A	==> 12CH
Porta Digital B	==> 12DH
Porta Digital C	==> 12EH
Controle das Portas	==> 12FH

3.4 BARRAMENTO AUXILIAR

São os sinais do barramento principal excluídos as linhas de alimentação. Estes sinais são gerados pela PIPC e são transmitidos ao barramento principal por meio de linhas físicas especiais "Flat Cable".

Este barramento está especificado para aplicações em tempo real, foram utilizados critérios para diminuir atrasos decorrentes do envio e recepção de dados entre o sistema mestre e as placas escravas.

Um dos critérios utilizados para a elaboração do barramento foi considerar as placas processadoras como uma extensão do micro Mestre, ou seja, após o pedido de requisição de barramento o processador da placa termina a instrução corrente e entra em "Halt". Com isto o micro mestre pode acessar a memória do processador como se esta fizesse parte de sua memória básica.

Outro critério foi a utilização de "wait state" somente para os dispositivos de E/S. A geração de de "wait state" para a leitura e escrita na memória causaria atrasos, pois a comunicação entre o micro mestre e os Processadores é realizada por meio de "flags" na memória.

Maiores informações sobre este barramento podem ser obtidas no item 3.6 Barramento Principal.

3.5 PLACA DE AMPLIFICAÇÃO DO BARRAMENTO (PABO)

Este módulo tem por finalidade amplificar os sinais do barramento auxiliar, para com isto fortificar os sinais na entrada do barramento principal. Os sinais podem ser distorcidos pelo tamanho do "flat cable" (aproximadamente 2 metros), com esta amplificação garante-se a integridade dos sinais nos "slots" nos quais são inseridas as placas microprocessadoras.

O esquemático deste módulo encontra-se na página A11 do Apêndice 1 da documentação do sistema CPR1.

3.6 BARRAMENTO PRINCIPAL

Na Tabela 3.1 pode-se observar os sinais presentes no barramento principal, estes sinais são subdivididos em sinais de alimentação, controle, endereços, dados e de temporizações.

Sinais presentes no barramento :

- Sinais de alimentação
 - 8 linhas de GND
 - 4 linhas de 5V
 - 4 linhas de 12V
 - 4 linhas de -12V
 - 1 linha de GND micro Mestre

- 20 linhas de endereços (ADR0 a ADR19)

- 8 linhas de dados (DAD0 a DAD7)

- 12 linhas de requisição de barramento
"BUSREQ1" A "BUSREQ10"

- 3 linhas de temporização (TEMP1 A TEMP3)

- 1 linha de Relógio do micro Mestre

- 4 linhas de controle
 - IORD
 - IOWR
 - MEMRD
 - MEMWR

- 1 linha de "RESET"

Sinais de Alimentação	1--> GND 2--> 5V 3--> 12V 4--> -12V 5--> GND	36--> GND 37--> 5V 38--> 12V 39--> -12V 40--> GND
Sinais de Controle	6--> BUSREQ1 7--> BUSREQ3 8--> BUSREQ5 9--> BUSACK1 10--> BUSACK3 11--> BUSACK5 12--> -IORD 13--> -IOWR 14--> RESET	41--> BUSREQ2 42--> BUSREQ4 43--> BUSREQ6 44--> BUSACK2 45--> BUSACK4 46--> BUSACK6 47--> -MEMRD 48--> -MEMWR 49--> CLOCK PC AT
Sinais de Endereços	15--> ADR14 16--> ADR12 17--> ADR10 18--> ADR8 19--> ADR6 20--> ADR4 21--> ADR2 22--> ADR0	50--> ADR15 51--> ADR13 52--> ADR11 53--> ADR9 54--> ADR7 55--> ADR5 56--> ADR3 57--> ADR1
Sinais de Dados	23--> DAD6 24--> DAD4 25--> DAD2 26--> DAD0	58--> DAD7 59--> DAD5 60--> DAD3 61--> DAD1
Sinais de Endereços	27--> ADR18 28--> ADR16	62--> ADR19 63--> ADR17
Sinais de Temporização	29--> GND PC AT 30--> TEMP 3	64--> TEMP 1 65--> TEMP 2
Sinais de Alimentação	31--> GND 32--> -12V 33--> 12V 34--> 5V 35--> GND	66--> GND 67--> -12V 68--> 12V 69--> 5V 70--> GND

Tabela 3.1 : Barramento principal

SINAIS DO BARRAMENTO PRINCIPAL

ADR0 a ADR15 : Estes sinais são usados para endereçar memória e dispositivos E/S, estas linhas permitem acessar 64 kbytes de memória.

ADR16 a ADR19 : Estas linhas de endereços são utilizadas para decodificar o endereço da placa a ser acessada. Os endereços das placas processadoras variam de 0 a FH.

DAD0 a DAD7 : Estas linhas fornecem o barramento de dados entre os microprocessadores, memórias e dispositivos E/S.

BUSREQ1 a BUSREQ10 : São sinais para requisição do barramento das placas processadoras, estas requisições podem ser ativados independentemente. Estes sinais são ativados com nível lógico "0".

RESET : Este sinal é utilizado para inicializar e reinicializar o sistema quando este é ligado ou na presença de um problema que faça o Micro Mestre parar.

IOR : Este sinal indica aos dispositivos de E/S que devem colocar dados no barramento. É ativo em nível lógico "0".

IOW : Este sinal indica aos dispositivos de E/S a lerem dados do barramento. É ativo em nível lógico "0".

MEMRD : Este sinal indica aos dispositivos de memória que devem colocar dados no barramento. É ativo em nível lógico "0".

MEMWR : Este sinal indica aos dispositivos de memória a lerem dados do barramento. É ativo em nível lógico "0".

TEMP1 a TEMP3 : São sinais de temporização para as placas processadoras. A frequência destes sinais é controlada pelo operador.

3.7 PLACA DIGITAL DE CONTROLE DISTRIBUÍDO (PCDD)

Estas placas tem por finalidade o controle de processos de uma maneira geral, são microprocessadas e interligadas ao micro mestre pelo barramento principal. O diagrama de blocos do sistema está apresentado na Figura 3.3, cada bloco apresentado esta descrito suscitamente a seguir :

BLOCO 1 : MÓDULO DO MICROPROCESSADOR

Este bloco é composto do microprocessador Z80, tem por função o controle dos componentes da placa. O microprocessador entra em "HALT" (parada) quando recebe um sinal de BUSREQ, sinalizando o painel frontal com um sinal de BUSACK.

O esquemático encontra-se na página A13 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 2 : MÓDULO DO GERADOR DE RELÓGIO

Este bloco é responsável pela geração do Relógio para a CPU, para o conversor A/D (no caso da PCDDII) e para os temporizadores programáveis. A frequência de relógio para a CPU é de 4 MHz, os temporizadores programáveis e o conversor A/D são alimentados por uma frequência de Relógio de 2 MHz e 500 kHz respectivamente.

O esquemático encontra-se na página A14 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 3 : MÓDULO DE "RESET" MANUAL AUTOMÁTICO E EXTERNO

O "reset" para CPU e circuitos periféricos de E/S é gerado quando o sistema é energizado, de uma maneira manual, através de uma chave "Push botton" no painel frontal da placa, e quando se executa um "Reset" no micro Mestre.

O esquemático encontra-se na página A15 do Apêndice 1 da documentação do sistema CPR1.

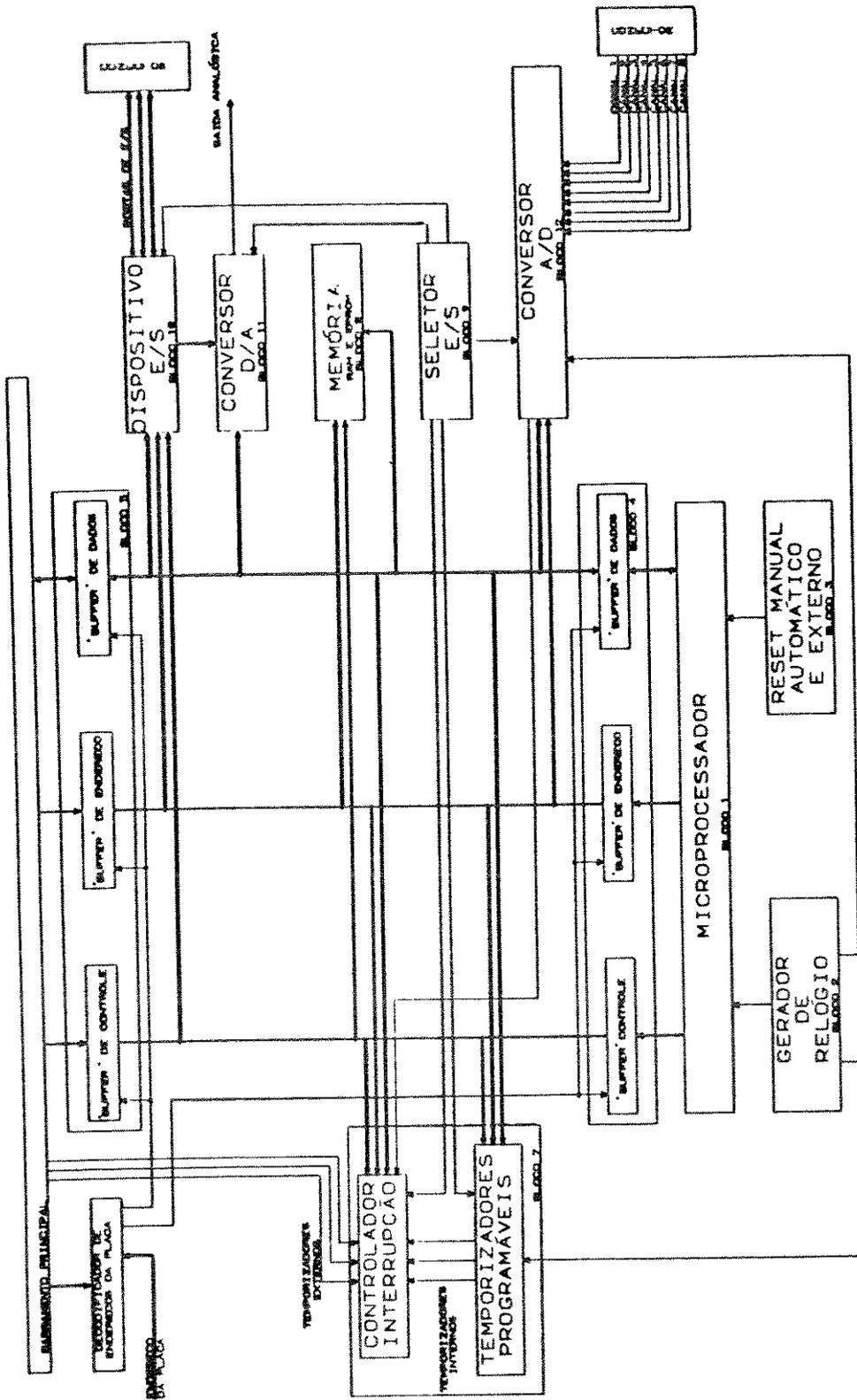


Figura 3.3 : Diagrama de Blocos das Placas de Controle

BLOCO 4 : MÓDULO AMPLIFICADOR DE BARRAMENTOS DA CPU

É composto de Amplificadores "tri-state" nos Barramentos de dados, endereços e de controle. A função deste módulo é desacoplar a CPU do resto do sistema quando o Micro Mestre requisitar o Barramento. A habilitação destes CI's está ligada diretamente no sinal de "BUSACK". Quando ocorre um "BUSREQ" a CPU termina a instrução corrente e ativa o sinal de "BUSACK" o qual desativa os Amplificadores da CPU. O sinal de habilitação depende também do BLOCO 6 que somente desativa os amplificadores da CPU quando a placa em questão estiver sendo endereçada pelos sinais de ADR16 a ADR19.

O esquemático deste módulo encontra-se na página A16 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 5 : MÓDULO AMPLIFICADOR DO BARRAMENTO PRINCIPAL

É composto por amplificadores "tri-state" nos barramentos de endereços dados e controle do barramento principal. A Habilitação deste CI's acontece quando ocorrer um sinal de "BUSACK" da CPU e quando a placa em questão estiver sendo endereçada pelos sinais de ADR16 a ADR19.

O esquemático deste módulo encontra-se na página A17 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 6 : MÓDULO DECODIFICADOR DE ENDEREÇO DA PLACA

Este módulo é responsável pela geração do sinal apropriado de habilitação para os amplificadores de CPU e amplificadores do barramento principal.

O esquemático deste módulo encontra-se na página A18 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 7 : MÓDULO DOS TEMPORIZADORES E DE INTERRUPÇÃO

Os algoritmos de controle podem ser divididos em tarefas, por exemplo, "loop" de controle de posição, "loop" de controle de velocidade, "loop" de controle de corrente, etc... . Algumas rotinas devem iniciar em períodos bem definidos de maneira cíclica. Para tanto existem 3 canais de temporização que estão acoplados a um controlador de interrupção. Portanto a função destes temporizadores é coordenar os ciclos de execução de rotinas do programa principal.

Endereços dos canais de temporização.

Canal 0 ==> 0CH

Canal 1 ==> 0DH

Canal 2 ==> 0EH

Controle ==> 0FH

Palavra de controle para o canal 1 ==> 34H

Palavra de controle para o canal 2 ==> 74H

Palavra de controle para o canal 3 ==> F4H

O Relógio para os contadores dos temporizadores é de 500 kHz, cada contador apresenta 16 bits, pode-se portanto dividir esta frequência por valores que vão desde 1 a 65536.

O controlador de interrupção utilizado apresenta oito canais de interrupção controlados por software. Os canais de interrupção IRQ3 a IRQ5 estão acoplados aos sinais de temporização do barramento, apresentando as mesmas funções dos temporizadores internos (IRQ0 a IRQ2). O canal 6 de interrupção está conectado ao conversor A/D (Sinal de final de conversão). O canal 7 esta reservada para expansão.

CANAIS DE INTERRUPTÃO

IRQ0 ==> TEMP1 (PCDDIII)
IRQ1 ==> TEMP2 (PCDDIII)
IRQ2 ==> TEMP3 (PCDDIII)
IRQ3 ==> TEMP1 (BARRAMENTO)
IRQ4 ==> TEMP2 (BARRAMENTO)
IRQ5 ==> TEMP3 (BARRAMENTO)
IRQ6 ==> Final de Conversão A/D
IRQ7 ==> Expansão

O esquemático deste módulo encontra-se na página A19 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 8 : MÓDULO DE MEMÓRIA

As placas processadoras apresentam 16 kbytes de memória RAM e 16 kbytes de memória EPROM. Na EPROM estão as rotinas de inicialização, rotinas de interrupção, rotinas utilizadas pelo programa principal, tabelas e programa monitor, cuja função é coordenar a recepção e execução do programa de controle enviado pelo Micro Mestre. A memória RAM reserva 3 kbytes para armazenamento do programa de controle, que é transmitido pelo Micro Mestre toda vez que se inicializa o manipulador, sendo que este algoritmo pode mudar conforme comandado pelo Micro Mestre, quatro áreas de 3 kbytes para armazenamento de variáveis durante a execução de uma tarefa e 1 kbytes para "flags".

MAPA DE MEMÓRIA

EPROM ==> 0000 A 3FFF
RAM ==> 4000 A 7FFF

O esquemático deste módulo encontra-se na página A20 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 9 : MÓDULO SELETOR DE E/S

A função deste bloco é selecionar os dispositivos de Entrada / Saída das Placas de acordo com a Tabela 3.2.

Dispositivo de E/S	Endereço
Porta Digital A	00H
Porta Digital B	01H
Porta Digital C	02H
Controle das Portas	03H
Porta Saída D/A (D9 e D8)	04H
Porta Saída D/A (D0 a D7)	05H
Controle Canais Analógicos	08H
Início de Conversão A/D	09H
Leitura de dados A/D (D0 a D7)	0AH
Leitura de dados A/D (D9 e D8)	0BH
Canal 0 do temporizador	0CH
Canal 1 do temporizador	0DH
Canal 2 do temporizador	0EH
Controle do temporizador	0FH
Controle 1 de interrupção	10H
Controle 2 de interrupção	11H

Tabela 3.2 : Endereços de E/S dos dispositivos de E/S

O esquemático deste módulo encontra-se na página A21 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 10 : MÓDULO DAS PORTAS DIGITAIS DE E/S

São reservadas três portas digitais de E/S com finalidades gerais, estas portas são programáveis através de uma palavra de controle, de acordo com a Tabela 3.3.

Porta A	Porta B	Porta C	Controle
E	E	E	9Bh
E	E	S	92h
E	S	E	99h
E	S	S	90h
S	E	E	8Bh
S	E	S	82h
S	S	E	89h
S	S	S	80h

Tabela 3.3 : Palavras de programação para as portas

O esquemático deste módulo encontra-se na página A22 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 11 MÓDULO DO CONVERSOR DIGITAL / ANALÓGICO

As placas de controle apresentam um conversor Digital Analógico de 10 bits.

ENDEREÇOS D/A:

Porta de saída de dados (A9 e A8) ==> 04h

Porta de saída de dados (A7 a A0) ==> 05h

O esquemático deste módulo encontra-se na página A23 do Apêndice 1 da documentação do sistema CPR1.

BLOCO 12 : MÓDULO DO CONVERSOR ANALÓGICO DIGITAL

As placas de controle utilizam conversor Analógico Digital de 10 bits com um tempo médio de conversão de 6 us.

ENDEREÇOS DO A/D :

Início de conversão ==> 09h

Leitura de dados (byte inferior) ==> 0Ah

Leitura de dados (byte superior) ==> 0Bh

Antes de utilizar o conversor A/D deve-se seleccionar o canal de entrada desejado, para tanto deve-se sair com o dado conveniente pela porta 08H, segundo a Tabela 3.4.

Canal Analógico	Barramento de dados
Canal 0	00H
Canal 1	01H
Canal 2	02H
Canal 3	03H
Canal 4	04H
Canal 5	05H
Canal 6	06H
Canal 7	07H

Tabela 3.4 : Ativação dos canais analógicos multiplexados

O esquemático deste módulo encontra-se na página A24 do Apêndice 1 da documentação do sistema CPR1.

3.8 PLACA DIGITAL DE ENTRADAS E SAÍDAS DIGITAIS

A função principal desta placa é monitorar e atuar sobre os dispositivos periféricos do manipulador, contém 12 portas digitais programáveis de Entrada e Saída, podendo atuar sobre os seguintes elementos auxiliares de controle do manipulador :

- Chaves de fim de curso
- Botões de emergência
- Invasão do espaço de trabalho
- Sensores diversos
- Relês de proteção

O microprocessador da PDES monitora as portas de entrada, processa as informações e atua nas portas de saída de modo a tratar os possíveis problemas que venham a ocorrer. Pode gerar uma interrupção não mascarável ao PC quando o evento necessitar de atenção urgente.

O diagrama de blocos da PEDS esta apresentado na Figura 3.4. A placa foi dividida em doze blocos cujos esquemáticos estão no Apêndice 1 (páginas A25 a A37) da documentação do sistema CPR1. Esta placa é derivada das placas de controle (PCDD) e a descrição dos blocos é idêntica aos já realizados no item 3.7 (Placa Digital de Controle Distribuído).

As diferenças entre a PDES e as PCDD's está no fato dela não apresentar conversores A/D e D/A, temporizadores e controlador de interrupção. Por outro lado a PDES apresenta doze portas digitais de E/S totalizando 96 bits de programação.

CARACTERÍSTICAS TÉCNICAS:

- Microprocessador : Z80
- Clock : 4 MHz
- Memória : RAM de 16 kbytes
- Memória : EPROM de 16 kbytes
- 12 Portas digitais de E/S
- Reset manual, externo e automático

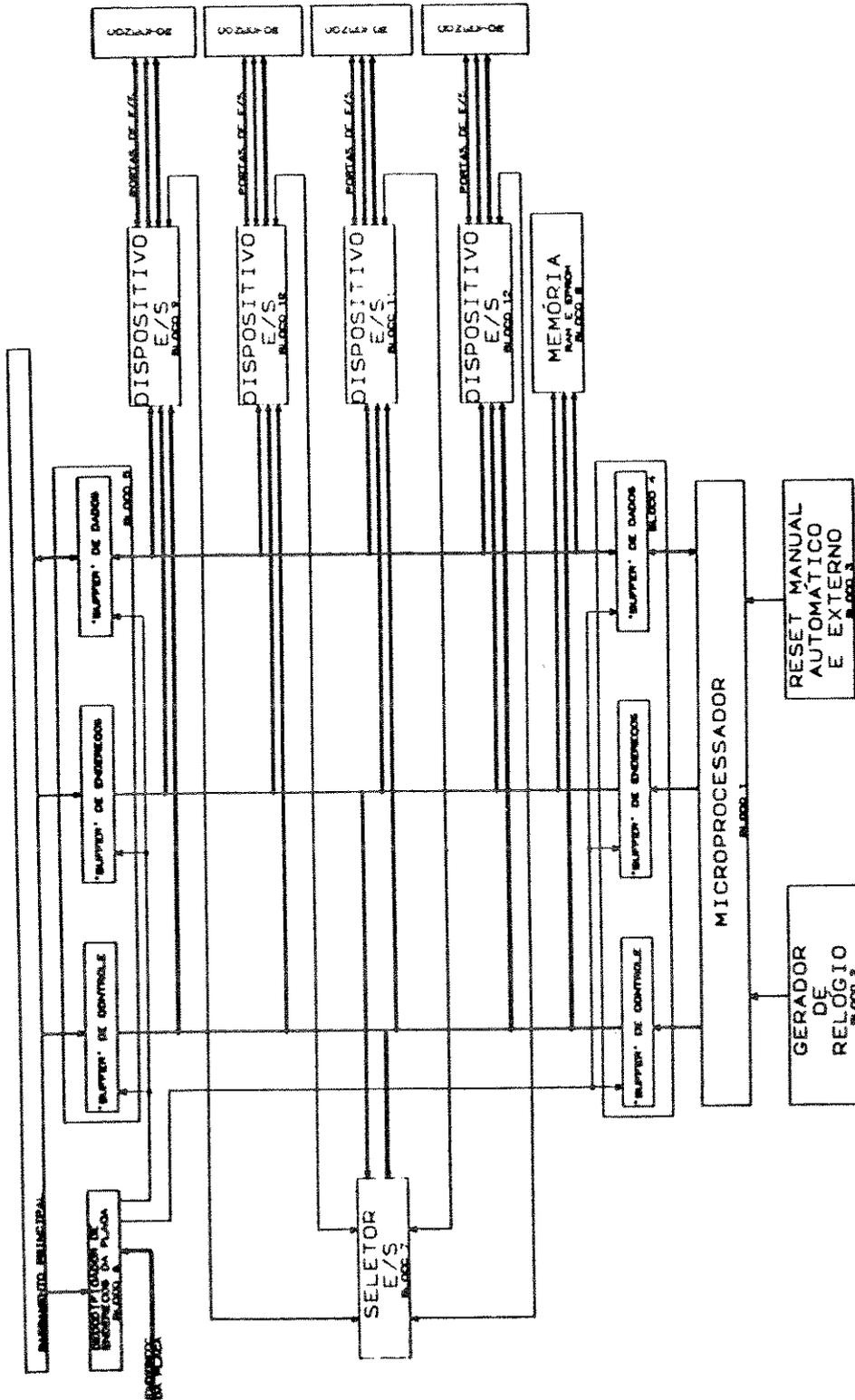


Figura 3.4 : Diagrama de Blocos da Placa Digital de E/S

BLOCOS 9 A 12 : MÓDULO DAS PORTAS DIGITAIS DE E/S

A PDES possui doze portas digitais de E/S. Estas portas são programáveis através de uma palavra de controle conforme a Tabela 3.3, sendo que a Tabela 3.5 apresenta os endereços de E/S das portas digitais.

Dispositivos de E/S	End reço
Porta A	00
Porta B	01
Porta C	02
Registro de controle (A B C)	03
Porta D	04
Porta E	05
Porta F	06
Registro de controle (D E F)	07
Porta G	08
Porta H	09
Porta I	0A
Registro de controle (G H I)	0B
Porta J	0C
Porta K	0D
Porta L	0E
Registro de controle (J K L)	0F

Tabela 3.5 : Endereços dos dispositivos de E/S da PDES

CAPITULO 4

DESENVOLVIMENTO DOS PROGRAMAS DO NÍVEL DE CONTROLE DE JUNTAS

4.1 INTRODUÇÃO

Em paralelo com a implementação do "Hardware" tornou-se necessário estudar a estrutura básica do "Software" a ser utilizado no projeto. Verificou-se que um esquema hierárquico que, a partir de um movimento especificado para o manipulador, pudesse chegar a valores de tensões aplicados nos motores, após a execução de três níveis básicos de controle, deveria ser adotado pela sua simplicidade e viabilidade técnica nesta primeira etapa que se deseja atingir.

A Figura 4.1 mostra um diagrama simplificado do esquema hierárquico adotado:

Como já foi dito no capítulo 3 este trabalho se refere basicamente ao projeto, análise e implementação das placas de controle digital direto, além das placas de interface e de Entradas e Saídas, portanto o "Software" a ser desenvolvido para viabilizar o seu funcionamento é o "Software" restrito ao nível de controle de juntas e consiste dos seguintes programas :

- Programa de Comunicação
- Programa Monitor Junta (PCDD)
- Programa Monitor da PDES
- Programa de Controle de Junta

O "Software" necessário para a execução dos níveis de definição de tarefas e controle de trajetória está sendo desenvolvido na tese de Doutorado do aluno Marconi Kolm Madrid.

A seguir fazemos uma descrição da finalidade dos programas relacionados anteriormente.

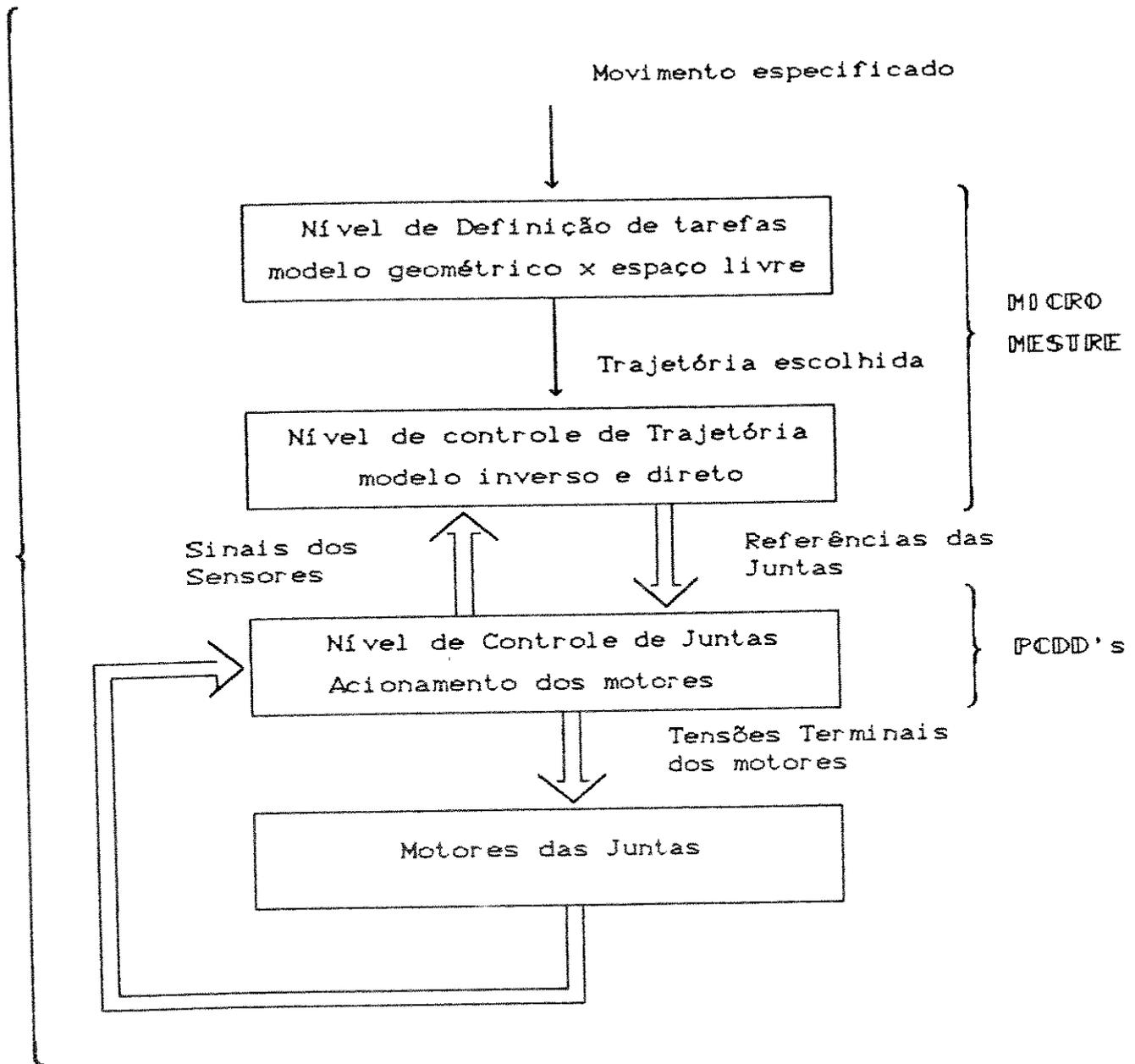


Figura 4.1 : Estrutura Hierárquica de Controle do Manipulador

4.2 PROGRAMA DE COMUNICAÇÃO

Para testar a comunicação entre o Micro Mestre e as placas processadoras foi desenvolvido um programa em linguagem "C" que transfere arquivos em linguagem de máquina para as placas processadoras.

O sistema Mestre transfere os programas desenvolvidos no Micro Mestre em linguagem de máquina para posições de memória RAM das placas processadoras. Isto é realizado para que tenhamos uma flexibilidade no uso de diferentes algoritmos de controle de juntas, dependendo do desempenho desejado, precisão ou mesmo o tipo de junta (garra, punho ou braço).

Os arquivos a serem transmitidos devem ser desenvolvidos em linguagem "assembly", compilados e "linkados" por meio de alguma ferramenta computacional, e o arquivo resultante deve estar no formato INTEL intellec 8/MDS, cujo formato de armazenamento está apresentado na Figura 4.2.

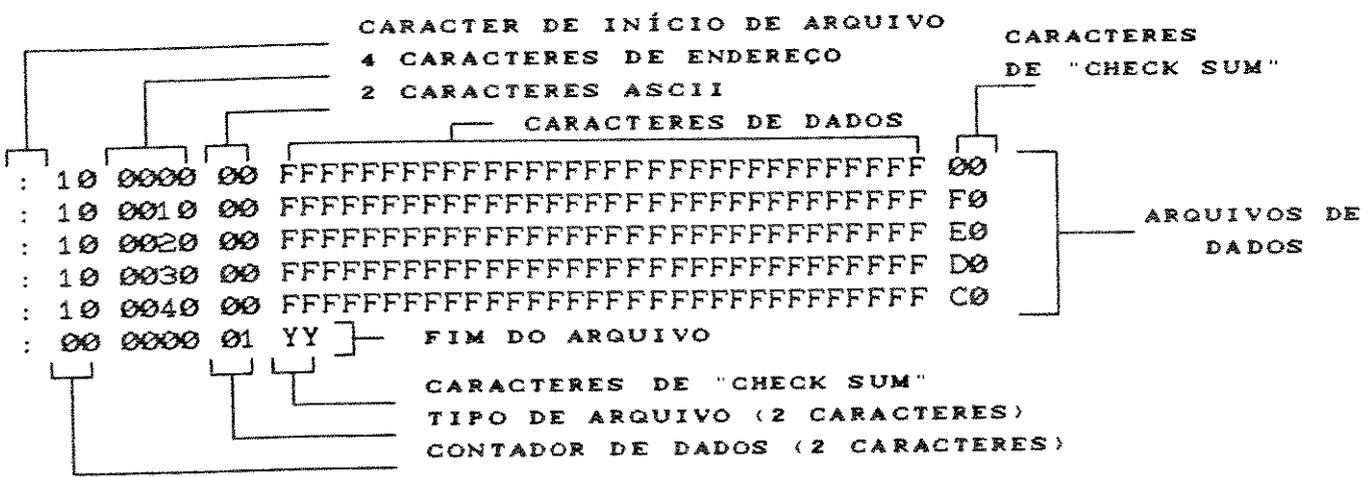


Figura 4.2 : Formato INTEL INTELLEC 8/MDS

Como pode ser observado na Figura 4.2 o arquivo é composto por linhas contendo as seguintes informações em sequência:

- Caracter de início de arquivo ":" - Caracter que define o início de uma linha do arquivo.
- Contador de dados - Define o número de caracteres de dados presentes na linha (2 Caracteres).
- Dois caracteres de endereços - Define o endereço de memória no qual os dados devem ser alocados, em ordem sequencial na memória (4 caracteres).
- tipo de arquivo na linha - O valor "01" indica que a linha em questão é a última linha do arquivo.
- Caracteres de dados - Este é o arquivo a ser transmitido propriamente dito, estes caracteres devem ser carregados na memória especificada pelos caracteres de endereços.
- Caracteres de "Check Sum" - Tem a finalidade de verificar a integridade dos dados no arquivo. Este valor é a soma de todos os caracteres anteriores em complemento de um.

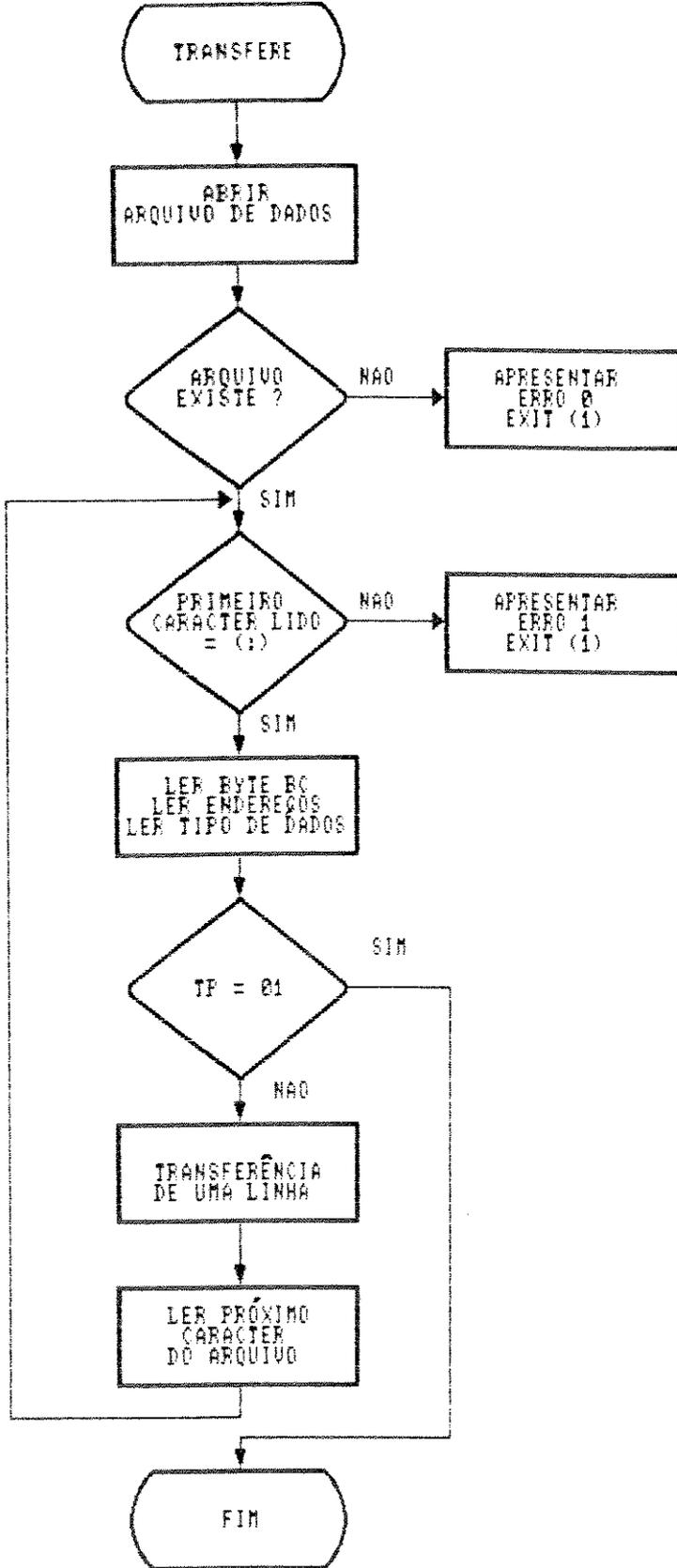
O Fluxograma deste programa está apresentado na Figura 4.3.

PASSO 1 :

É composto basicamente da inicialização dos temporizadores e das portas digitais de E/S.

PASSO 2 :

Verifica-se se o arquivo a ser transmitido existe ou não, se o arquivo for inexistente o programa é abortado e o erro 0 (arquivo inexistente) é apresentado na tela.



OBSERVAÇÕES:

1) ERRO 0 - ARQUIVO DE DADOS INEXISTENTE.

2) ERRO 1 - ARQUIVO DE DADOS NO FORMATO ERRADO.

3) O CARACTER "(" DEVE SER SEMPRE O PRIMEIRO CARACTER DA LINHA.

4) BC = CONTADOR DE DADOS, INDICA QUAL É O NÚMERO DE BYTES DE DADOS EXISTENTES NA LINHA CORRENTE.

5) TP = TIPO DE ARQUIVO, TP=01 INDICA QUE A LINHA CORRENTE É A ÚLTIMA LINHA DO ARQUIVO.

Figura 4.3 : Fluxograma do algoritmo de comunicação

PASSO 3 :

Verifica-se o primeiro caracter do arquivo é " : ", (caracter de início de arquivo), em caso contrário o programa é abortado e o erro 01 (arquivo no formato errado) é apresentado.

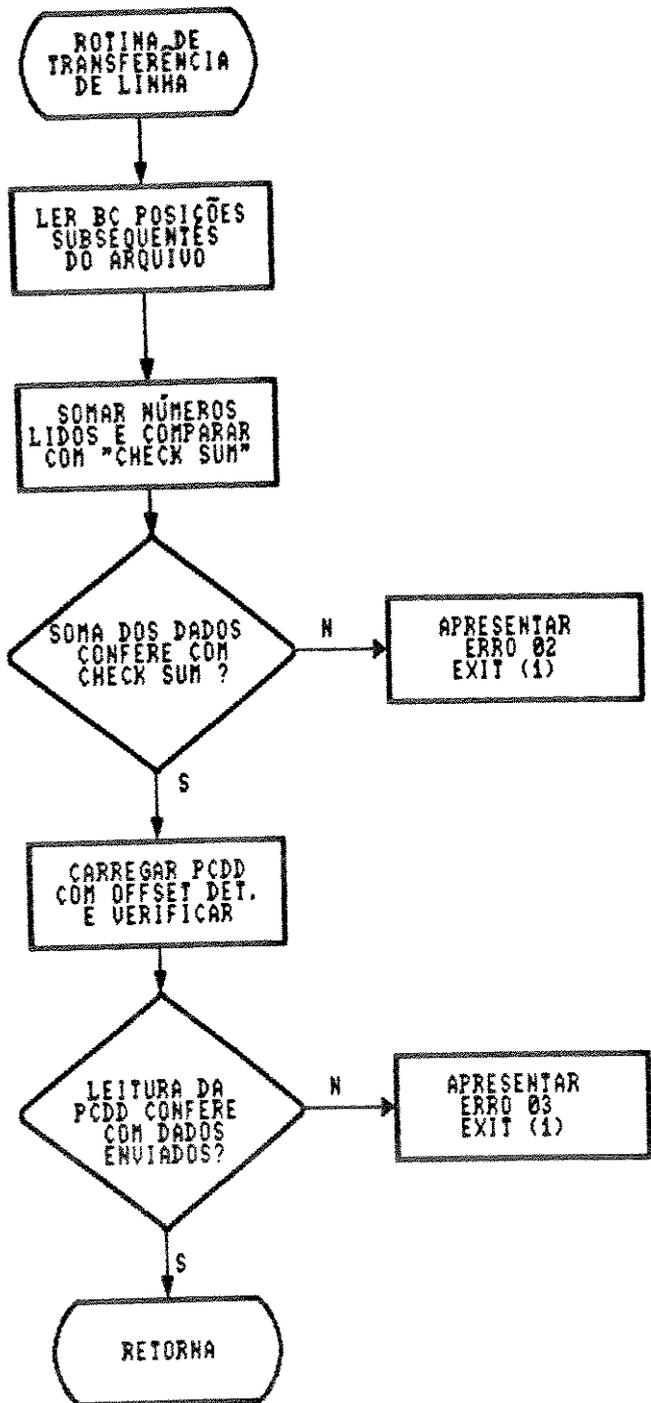
PASSO 4 :

Este passo corresponde ao fluxograma apresentado na Figura 4.4. Consiste em realizar a leitura dos caracteres ASC II de dados da linha e convertê-los para o hexadecimal correspondente através das rotinas convb (converte 2 bytes em ASCII para 1 byte Hexadecimal) e conv2b (converte 4 bytes em ASCII para 2 bytes em hexadecimal).

Neste ponto é conveniente esclarecer a geração de sinais de memória no Micro Mestre. Os endereços são formados através da utilização de um registrador de segmento e um endereço de deslocamento ("offset"):

- Registrador de segmento - 16 bytes que indicam um bloco de 64 Kbytes de memória)
- Endereço de deslocamento - 16 bytes que localizam um dado dentro do bloco determinado pelo registrador de segmento.

O processador constrói o endereço de dados com 20 bits, somando o endereço de deslocamento ao conteúdo do registrador de segmento com quatro zeros acrescentados na posição menos significativa, como mostrado na Figura 4.5.



OBSERVAÇÕES :

1) LEITURA DOS BC CARACTERES DE DADOS DA LINHA CORRENTE.

2) "CHECK SUM" É A SOMA DOS BYTES DA LINHA EM COMPLEMENTO DE UM, UTILIZADO PARA VERIFICAR A CONSISTÊNCIA DOS DADOS LIDOS.

3) ERRO 02 - ERRO DE CHECK SUM.

4) O ENDEREÇO DA PLACA É DETERMINADO PELO ENDEREÇO 8000H (REGISTRADOR DE SEGMENTO) E O "OFFSET" (VALOR ABSOLUTO DA MEMÓRIA DAS PLACAS PROCESSADORAS).

5) ERRO 03 - ERRO NA TRANSMISSÃO DOS DADOS LIDOS PARA AS PLACAS DEDICADAS.

Figura 4.4 : Fluxograma do algoritmo de transferência de linha

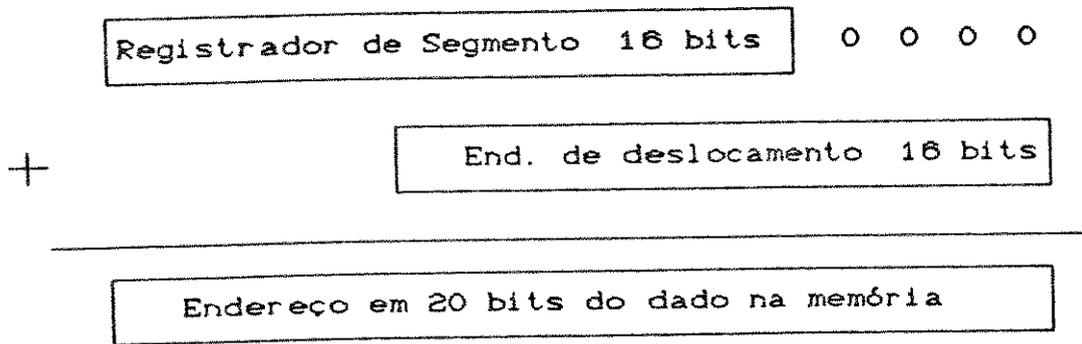


Figura 4.5 : Formação de endereços de memória

O Micro Mestre acessa as memórias das placas processadoras como se estas fizessem parte da sua própria memória. O endereço utilizado pelo Micro Mestre encontra-se entre ~~80000H~~ e 8FFFFH, entretanto somente são utilizados 32 kbytes de memória, pois o endereço das placas é gerado pela PIPC.

Para acessar toda a memória das 10 placas possíveis no barramento seriam necessários 320 Kbytes de endereçamento real. A Figura 4.6 mostra a formação dos endereços pelo Micro Mestre:

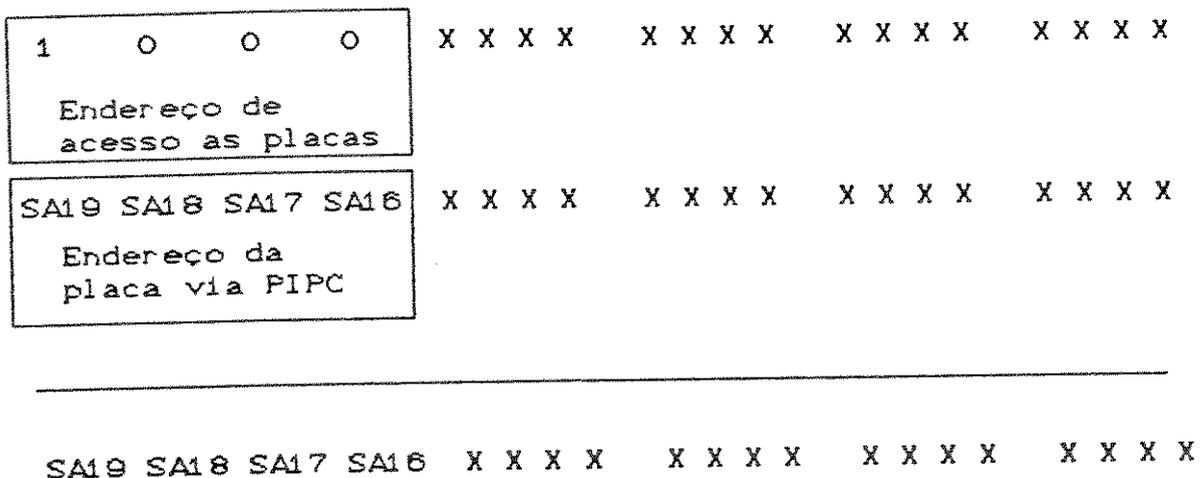


Figura 4.6 : Formação do endereço de acesso as placas processadoras

O endereço de segmento foi escolhido como 8000H. O endereço de deslocamento "Offset" é determinado pelo programa a ser transmitido e varia de 0000 7FFFH.

PASSO 5 :

Este passo faz parte do fluxograma da Figura 4.4. Após a criação do conjunto de dados da linha, realiza-se a soma em complemento de um de todos os dados da linha e sua comparação com o "check sum", em caso de divergência é apresentado o erro 2 (leitura incorreta). O programa então passa a enviar os dados para a placa processadora gerando-se um pedido de requisição de barramento e endereçando a placa em questão. O envio de dados para as placas é realizado através de instruções "pokeb" e "peekb" do "C",

A instrução "Pokeb" (segmento, offset, dado) carrega o dado na posição determinada pelo segmento e offset.

A instrução "Peekb" (segmento, offset), lê o dado da posição determinada pelo segmento e offset.

O algoritmo estabelece a comunicação, envia um dado, realiza sua leitura, compara e apresenta o erro 3 (erro na gravação da memória) caso os dados sejam incompatíveis.

PASSO 6 :

Após o envio da linha corrente do arquivo o programa passa a linha seguinte, realiza as mesmas operações anteriores e termina quando encontrar um caracter de fim de arquivo.

A listagem deste programa encontra-se na página 2 do Apêndice A2, da documentação do sistema.

4.3 PROGRAMA MONITOR DAS PCDD'S

O programa monitor das PCDD's é responsável pela sua inicialização, tratamento de interrupções e pela determinação do seu modo de operação. O modo de operação consiste em determinar se o algoritmo de controle armazenado na memória RAM deve ser processado ou inibido, isto é importante pois devemos desabilitar o processamento das juntas quando necessitarmos transferir outro algoritmo de controle para a junta ou quando necessitarmos ler um banco de dados das PCDD's.

O fluxograma deste programa esta apresentado na Figura 4.7 e é composto basicamente dos seguintes módulos:

INICIALIZAÇÃO DOS COMPONENTES DA PLACA

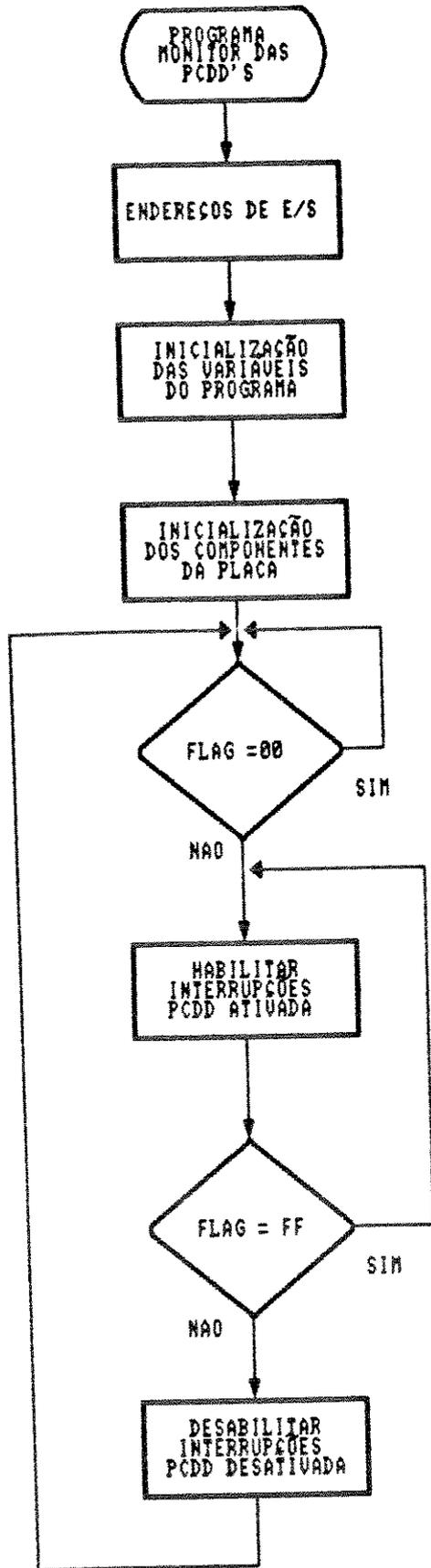
No endereço de "RESET" (0000H), inicializa-se ponteiros, área de variáveis, periféricos de entradas e saídas digitais, controlador de interrupção e temporizadores.

Todas as portas digitais de E/S estão programadas como saídas. O controlador de interrupção fica desabilitado e o temporizador programável com 5 [ms] de ciclo.

A frequência definida pela programação dos temporizadores determina o período de amostragem do algoritmo de controle.

ROTINA PRINCIPAL

O programa monitor passa a esperar um "Flag" de liberação a ser enviado pelo Micro Mestre para habilitar o algoritmo de controle. Em caso contrário envia um sinal 80H para o Módulo Gerador de MLP. (O angulo de 180 graus resultante força a parada do motor e corresponde a $e_c = 0$ graus).



OBSERVAÇÕES :

1) FLAG = FF INDICA QUE O PROGRAMA MONITOR DEVE PERMITIR A EXECUÇÃO DA ROTINA DE CONTROLE

2) FLAG = 00 INDICA QUE O PROGRAMA MONITOR NÃO DEVE PERMITIR A EXECUÇÃO DA ROTINA DE CONTROLE.

Figura 4.7 : Fluxograma do programa monitor da PCDD

ROTINA DE TRATAMENTO DE INTERRUPÇÕES

Na ocorrência de uma interrupção o microprocessador para o seu processamento normal e dá um salto para o endereço 0038H, indica ao controlador de interrupção a aceitação do pedido e lê um vetor que indica qual foi o canal que gerou a interrupção. O canal 0 indica que é o momento de processarmos o algoritmo de controle, O canal 6 indica o final de conversão do conversor A/D.

MAPA DE MEMÓRIA DA EPROM

0000H- 01FFH- Tratamento de interrupções
0200H- 0FFFH- Rotina de inicialização e rotina principal
1000H- 3FFFH- Subrotinas auxiliares

MAPA DE MEMÓRIA DA RAM

4000H- 47FFH- Área de algoritmos de controle
4800H- 53FFH- Tabela de armazenamento 1
5400H- 5FFFH- Tabela de armazenamento 2
6000H- 6BFFH- Tabela de armazenamento 3
6C00H- 77FFH- Tabela de armazenamento 4
7800H- 7EFFH- Área para variáveis do programa
7F00H- 7FFFH- Área de "STACK POINTER"

A listagem deste programa encontra-se na página 8 do Apêndice A2, da documentação do sistema.

4.4 PROGRAMA MONITOR DA PDES

O programa Monitor da PDES tem as funções de inicialização e habilitação do programa enviado pelo Micro Mestre. Os programas enviados pelo Micro Mestre têm a finalidade de monitorar os sinais de E/S do espaço operacional do manipulador. A apresentação do fluxograma está na Figura 4.8. basicamente esta rotina é semelhante ao programa monitor da PCDD, a diferença consiste na inicialização (Não apresenta temporizadores, controlador de interrupção, conversor D/A, conversor A/D) e na rotina principal.

A rotina está apresentada na página 15 do Apêndice A2, da documentação do sistema.

OBSERVAÇÃO :

1) SE FLAG # 00 O MICRO MESTRE ESTÁ INDICANDO A PLACA PROCESSADORA QUE O MANIPULADOR NÃO VAI INICIAR UM MOVIMENTO.

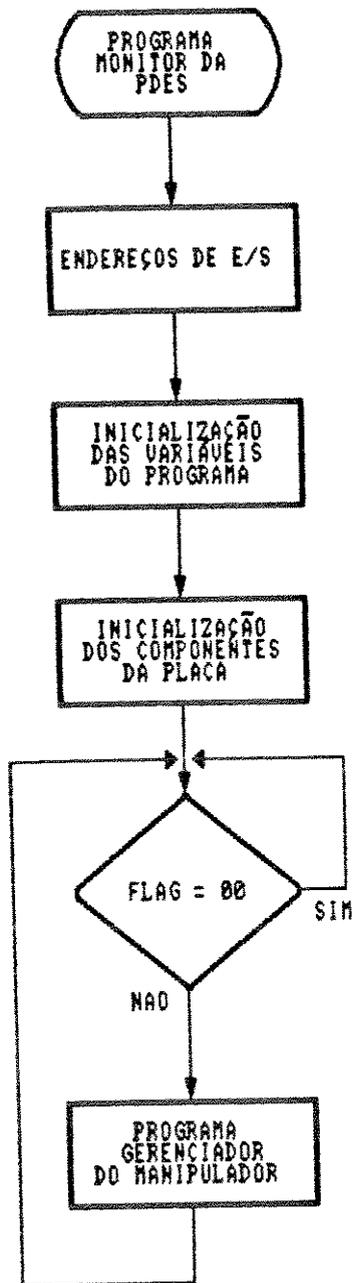


Figura 4.8 : Fluxograma do programa monitor da PDES

4.5 PROGRAMA PID PARA CONTROLE DE JUNTAS

Para podermos analisar o desempenho do sistema foi implementado um algoritmo PID de controle de juntas.

4.5.1 EQUAÇÃO RECURSIVA UTILIZADA

A equação do controlador PID analógico é dada pela seguinte equação:

$$u(t) = k \left[e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau + T_D \frac{de(t)}{dt} \right] \quad (4.1)$$

onde:

$u(t)$: função de controle

$e(t)$: erro

k : ganho

T_I : tempo integral

T_D : tempo derivativo

Para pequenos intervalos de amostragem T_0 esta equação pode ser discretizada substituindo-se a derivada por uma diferença de primeira ordem e a integral por um somatório, seguindo a integração retangular [Isermann 81], teremos:

$$u(k) = k \left[e(k) + \frac{T_0}{T_I} \sum_{i=0}^k e(i-1) + \frac{T_D}{T_0} (e(k) - e(k-1)) \right] \quad (4.2)$$

Para facilitar a implementação de algoritmos recursivos, é conveniente utilizar a equação mostrada a seguir:

$$\Delta u(k) = u(k) - u(k-1) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \quad (4.3)$$

Onde os parâmetros q_0 , q_1 e q_2 são definidos por :

$$q_0 = k \left(1 + \frac{T_D}{T} \right) \quad (4.4)$$

$$q_1 = -k \left(1 + 2 \frac{T_D}{T_0} - \frac{T_0}{T_i} \right) \quad (4.5)$$

$$q_2 = k \frac{T_D}{T_0} \quad (4.6)$$

Utilizando-se a integração trapezoidal, teremos :

$$u(k) = k \left[e(k) + \frac{T_0}{T_i} \frac{(e(k) - e(0))}{2} + \sum_{i=0}^{k-1} e(i) + \frac{T_D}{T_0} (e(k) - e(k-1)) \right] \quad (4.7)$$

Para a aproximação trapezoidal os parâmetros q_0 , q_1 e q_2 são definidos por :

$$q_0 = k \left(1 + \frac{T_0}{2T_i} + \frac{T_D}{T_0} \right) \quad (4.8)$$

$$q_1 = -k \left(1 + \frac{2T_D}{T_0} - \frac{T_0}{2T_i} \right) \quad (4.9)$$

$$q_2 = k \frac{T_D}{T_0} \quad (4.10)$$

Aplicando-se a transformada Z na equação 4.3 obtemos a seguinte função de transferência para o regulador PID digital:

$$G_c(z) = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{1 - z^{-1}} \quad (4.11)$$

Para grandes intervalos de amostragem [Isermann 81] encontra a seguinte função de transferência para um algoritmo de v^{th} ordem :

$$G_c(z) = \frac{q_0 + q_1 z^{-1} + \dots + q_v z^{-v}}{1 - z^{-1}} \quad (4.12)$$

Para $v = 2$ teremos um controlador PID. A função de transferência resultante é igual a da equação (4.11). Com a diferença que os parâmetros q_0 , q_1 e q_2 não são encontrados utilizando-se as equações 4.4 a 4.6 e 4.8 a 4.10.

O algoritmo implementado utiliza a equação a diferenças 4.3, sendo que os parâmetros q_0 , q_1 e q_2 são enviadas pelo Micro Mestre.

4.5.2 DESCRIÇÃO DO DIAGRAMA DE BLOCOS RESULTANTE

O diagrama de blocos do controle de posição em malha fechada está apresentado na Figura 4.9.

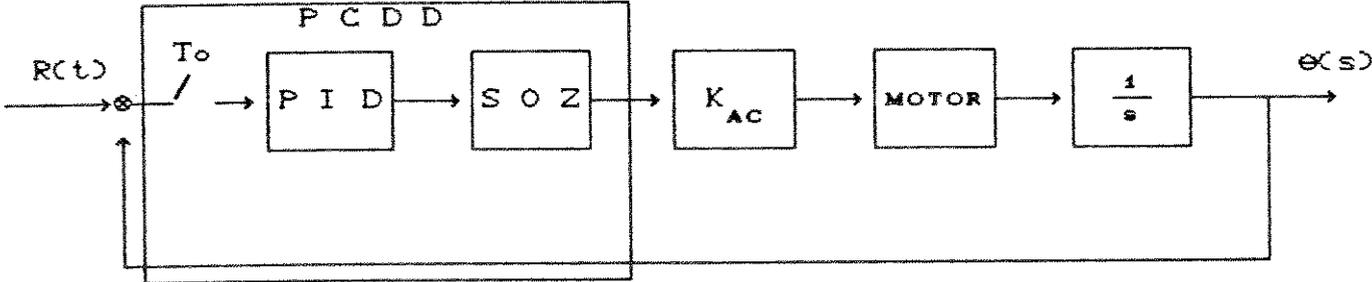


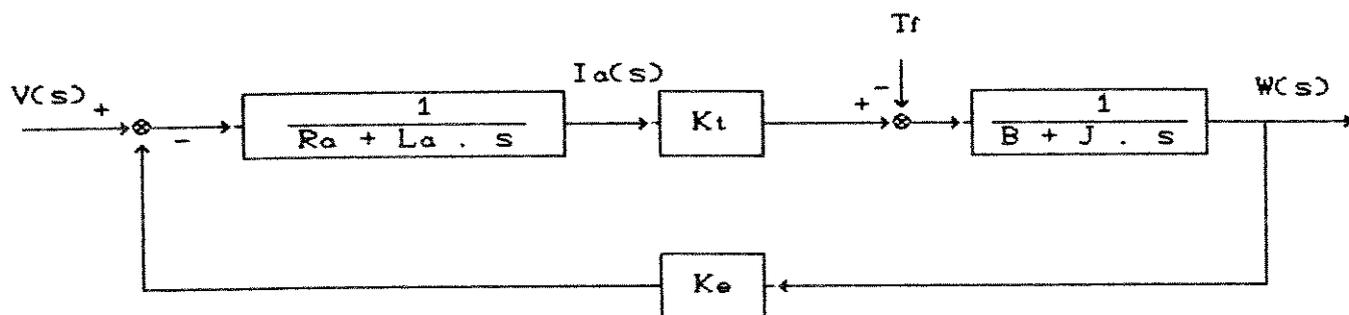
Figura 4.9 : Diagrama de blocos do Sistema em Malha Fechada

A PCDD está representada acima pelo amostrador (A/D), pelo regulador PID e pelo segurador de ordem zero (Saída para o Recortador).

O bloco K_{ac} representa a função de transferência do recortador para valores adequados de frequência, quando se garante a estabilidade do regime periódico. A relação entre o ângulo de comutação de entrada e a tensão média de saída é linear o que resulta em um ganho K_{ac} no diagrama de blocos.

O motor utilizado é um motor de corrente contínua a imã permanente e seu diagrama de blocos é apresentado segundo [Eletrocraft 1973] na Figura 4.10, tendo como variável de saída a velocidade angular, portanto devemos integrar a sua saída para obtermos a posição angular, esta integração é representada pelo bloco $(1/s)$ na Figura 4.9.

Para fechar a malha existe uma realimentação de posição com ganho unitário.



Onde:

- R_a : Resistência de armadura
- L_a : Indutância de armadura
- B : Coeficiente de atrito viscoso
- J : Momento de inércia
- K_t : Constante de torque
- K_e : Constante de Força contra-eletromotriz

Figura 4.10 : Modelo do motor de corrente contínua a imã permanente

4.4.3 OBTENÇÃO DOS PARÂMETROS q_0 , q_1 , q_2

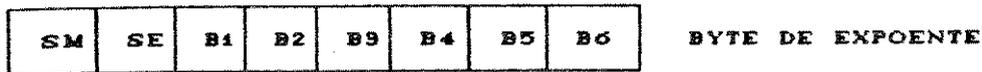
O cálculo dos parâmetros pode ser realizado por vários métodos [Isermann 1981], por exemplo :

- Ziegler Nichols
- Análise em frequência
- Tentativa e erro "Heurístico"

O cálculo destes parâmetros foge ao escopo deste trabalho de tese não sendo portanto apresentado. Estes parâmetros são gerados heurísticamente pelo operador e enviados pelo Micro Mestre através de um programa monitor protótipo.

4.5.4 SOFTWARE DESENVOLVIDO

O programa de controle foi desenvolvido em ponto flutuante reservando-se 4 bytes para a mantissa e um byte de expoente e sinal da mantissa:



Onde:

SM = Bit sinal da mantissa

SE = Bit sinal do expoente

B1 a B5 = bits do expoente

Utilizando-se este sistema de numeração podemos obter números na seguinte faixa:

. F F F F 2^{+64} (Máximo número positivo)
. -F F F F 2^{+64} (Máximo número negativo)
. F F F F 2^{-64} (Mínimo número positivo)
. -F F F F 2^{-64} (Mínimo número negativo)

ALGORITMO PID

PASSO 1 :

O Fluxograma deste algoritmo está apresentado na FIGURA 4.11. O algoritmo de controle lê a posição atual através do conversor A/D. A saída deste conversor é em "offset binário", ou seja:

F F B 0 (+ 5V)
8 0 0 0 (zero)
0 0 0 0 (- 5V)

PASSO 2 :

A seguir converte o valor obtido em "offset" binário na notação utilizada para ponto flutuante.

PASSO 3 :

O valor de referência é obtido de uma posição específica na memória e é enviado pelo micro Mestre. Com o valor de referência e o valor real realimentado, calcula-se então o valor do erro resultante.

PASSO 4 :

O erro é submetido a um regulador PID cuja função esta apresentada na equação 4.3. As constantes do regulador são obtidas de posições específicas de memória e são fornecidas pelo micro Mestre.

PASSO 5 :

A saída do regulador é convertida para ponto fixo e submetida ao módulo gerador de MLP.

PASSO 6 :

Realocar as variáveis utilizadas pelo regulador e esperar a próxima interrupção, ou seja, $e(k)$ passa a ser $e(k-1)$ e $e(k-1)$ passa a ser $e(k-2)$, preparando o algoritmo para a próxima execução da rotina.

A listagem do programa PID encontra-se na página 19 do Apêndice A2, da documentação do sistema CPR1.

SUBROTINAS MATEMÁTICAS

A subrotina de multiplicação está apresentada na Figura 4.12, sua listagem encontra-se na página 37 do Apêndice A2, da documentação do sistema CPR1. A subrotina de adição é também utilizada para subtração pois aceita operandos negativos. A subrotina de adição está apresentada na Figura 4.13, sua listagem está no Apêndice A2 página 28.

A subrotina IGUALA faz parte da subrotina de adição, sendo utilizada para igualar os expoentes dos números a serem somados.

A subrotina obtém a diferença entre os expoentes e roda a mantissa de menor expoente para direita tantas vezes quanto for a diferença dos expoentes. O fluxograma desta subrotina está apresentada na Figura 4.14 .

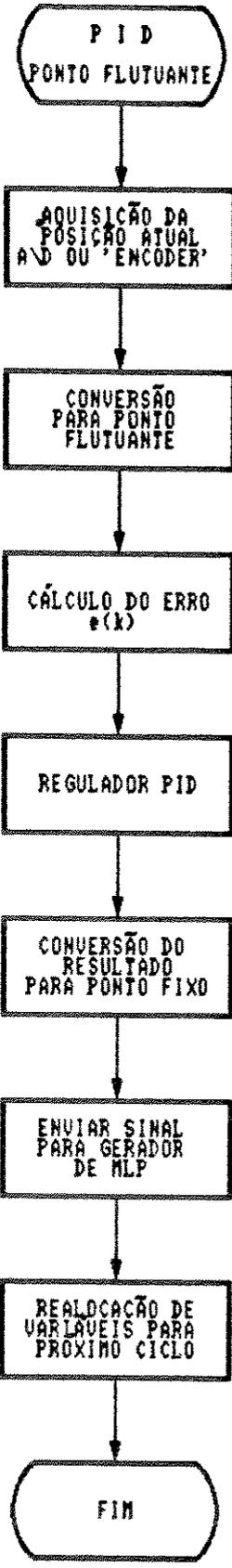


Figura 4.11: Fluxograma do algoritmo PID de controle de junta

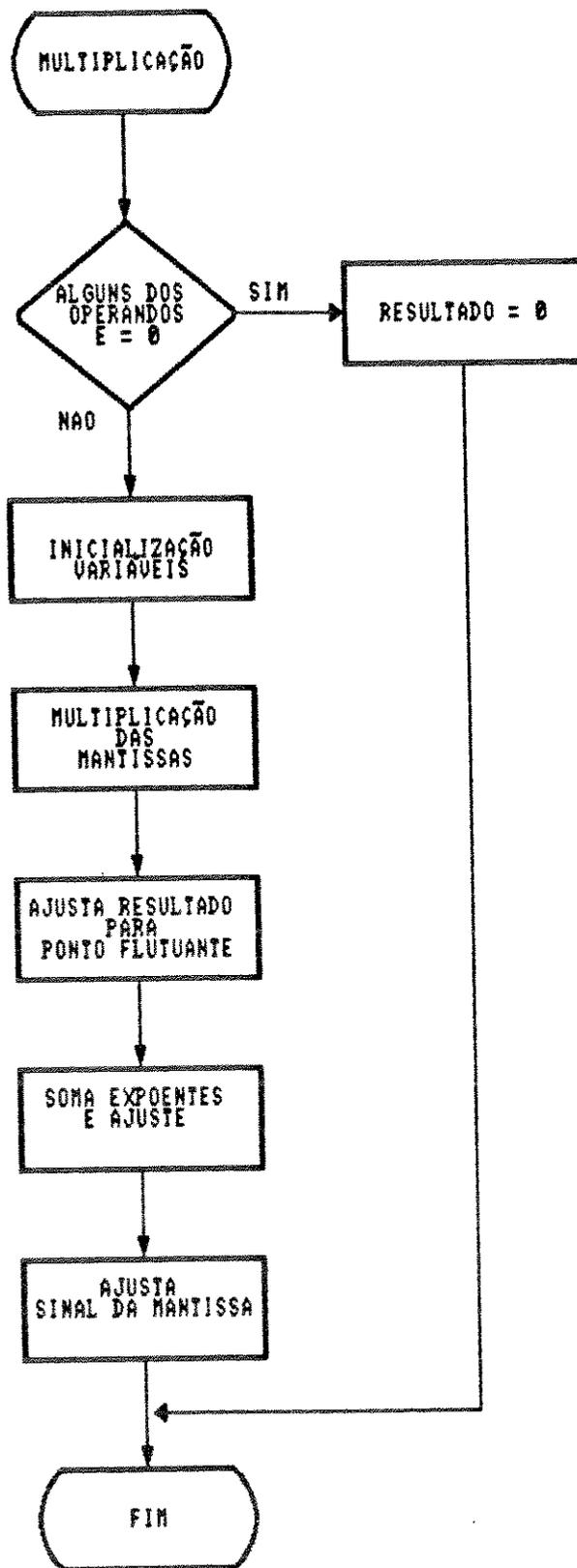


Figura 4.12 : Fluxograma da subrotina de multiplicação

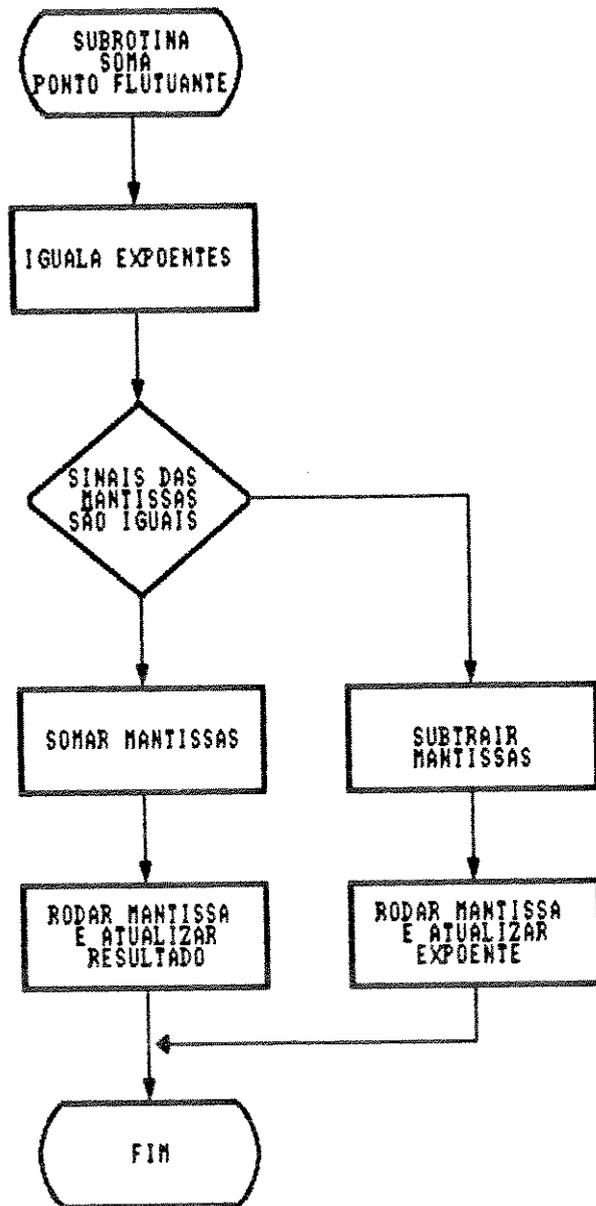


Figura 4.13 : Fluxograma da subrotina de adiçao

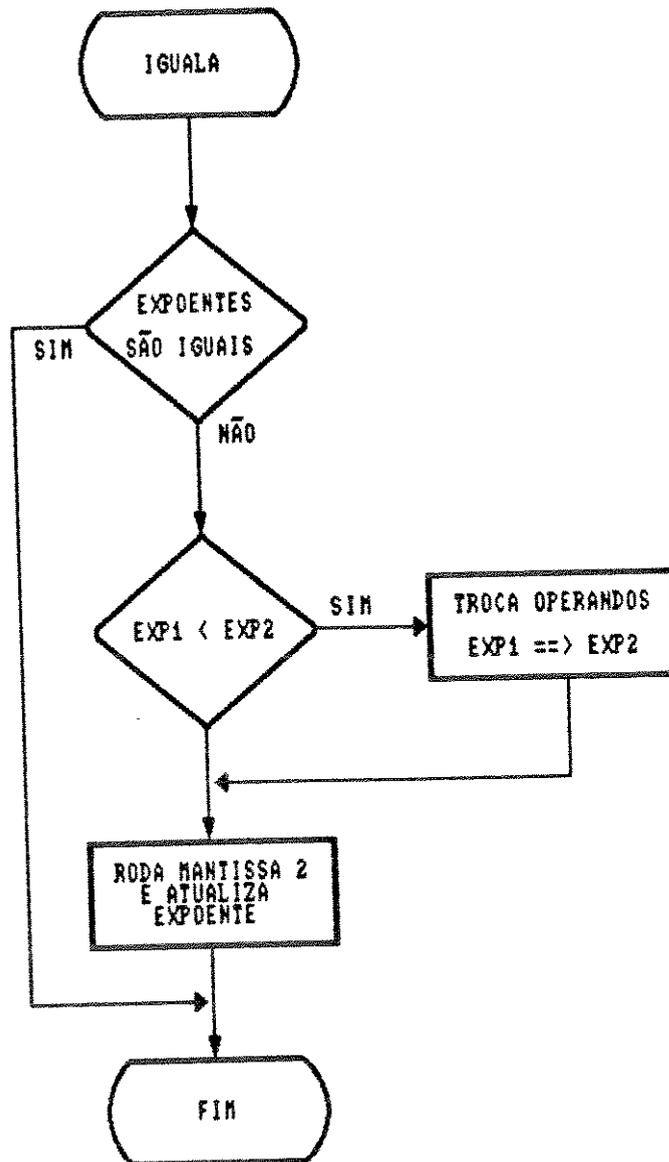


Figura 4.14 : Fluxograma da Subrotina IGUALA

CAPÍTULO 5

ANÁLISE DE DESEMPENHO E CONCLUSÕES

5.1 CONSIDERAÇÕES INICIAIS

O desenvolvimento de robôs manipuladores é sem dúvida uma tarefa complicada, pois envolve uma integração de vários campos de pesquisa em diversas áreas do conhecimento. Esta tese foi direcionada a projetar e implementar o "Hardware" para um controlador genérico destinado a robôs manipuladores, inserindo-se dentro de um projeto denominado JECA II que prevê a construção de um manipulador completo, estando em fase final de construção, onde todos os subsistemas já foram projetados, montados e estão em fase de testes e integração, os quais estão relacionados a seguir:

- Estrutura Mecânica;
- Placa de Acionamento dos Recortadores;
- Placas de Tratamento dos Sinais dos Sensores;
- Recortadores;
- Fontes.

Para realizar os testes de desempenho do CPR1 foi montado um circuito de acionamento de uma junta, especificamente o acionamento do movimento de preensão da garra.

Com este sistema montado esperávamos verificar se todas as funções do "Hardware" estavam funcionando e operando de acordo com as especificações do projeto. O algoritmo de controle implementado a nível de junta visa testar somente o "Hardware", não sendo para esta tese estudados algoritmos de controle especiais para a junta. O algoritmo implementado está de acordo com o item 4.4 (Programa PID para a junta), com a geração aleatória pelo operador dos parâmetros para o algoritmo e transmissão para as placas processadoras através do Micro Mestre.

Além do teste de conjunto foram levantados alguns dados do sistema, principalmente relativos a tempos gastos para comunicação entre o Micro Mestre e as placas dedicadas, pois esta foi a maior preocupação durante a fase de projeto, uma vez que este sistema foi designado para operar em tempo real.

Estes dados obtidos experimentalmente serão utilizados para analisarmos o desempenho do CPR1. Os tópicos a serem abordados são:

- Tempos das rotinas de comunicação;
- Temporizações com os "Jumpers" de "Wait State";
- Temporizações máximas obtidas com os temporizadores;
- Ruídos observados;
- Desempenho do algoritmo de controle;
- Barramento.;

5.2 ROTINAS DE COMUNICAÇÃO

Um fator muito importante na análise do desempenho do sistema é o tempo gasto para o Micro Mestre enviar e receber dados para as placas processadoras dedicadas. Iremos apresentar o tempo gasto para enviar lotes de dados.

- Tempo gasto para enviar / receber um dado
T_{médio} = 23,5 [μs]
- Tempo gasto para enviar / receber 6 dados
T_{médio} = 115 [μs]

- Tempo gasto para enviar / receber 1 kbytes de dados
Tmédio = 18 [ms]
- Tempo gasto para enviar / receber 4 kbytes de dados
Tmédio = 68 [ms]
- Tempo gasto para enviar / receber 16 kbytes dados
Tmédio = 268 [ms]

A Rotina de comunicação está apresentada nos fluxogramas das Figuras 4.3 e 4.4. Como existe um procedimento de verificação se o dado é consistente ou não, os números obtidos acima são valores médios, desde que no máximo a rotina em caso de erro realiza até dez tentativas de acerto na transmissão e recepção de dados. Os testes indicam que a quantidade de falhas ocorridas durante a transmissão / recepção está na ordem de 5%.

Observou-se com medidas em osciloscópio que existe um pouco de ruído nos sinais de requisição de barramento. Com a melhoria da placa de amplificação de barramento e com o aterramento da estrutura este nível de problemas tende a diminuir e conseqüentemente reduzir o nível de erros.

Para analisarmos o atraso de transporte devido ao tempo perdido para o Micro Mestre enviar os sinais de referência e receber os sinais dos sensores de todas as seis placas de controle, devemos considerar a equação 5.1 que nos mostra a formação do atraso de transporte resultante no sistema.

$$t_d = t_c + t_p \quad (5.1)$$

Onde:

- t_d : tempo de atraso de transporte
- t_c : tempo necessário para comunicação
- t_p : tempo de processamento do algoritmo de controle

Para o algoritmo implementado temos um $t_p = 4$ ms e um tempo de comunicação de aproximadamente 0,690 ms, com isto o tempo de atraso de transporte é de aproximadamente 4,7 ms, portanto o tempo de comunicação entre o micro mestre e todas as seis juntas processadoras é da ordem de 15% do tempo de atraso de transporte.

Como a menor constante de tempo de todas as juntas é $\tau = 50$ ms. O algoritmo implementado tem um tempo de atraso de transporte menor do que 10% da menor constante de tempo mecânica do sistema.

O sistema é totalmente flexível, permitindo a programação via sistema Mestre do período de amostragem. Com isto pode-se testar a influência do período de amostragem em relação ao tempo de atraso de transporte no desempenho dos algoritmos de controle implementados a nível de juntas.

5.3 TEMPORIZAÇÕES RESULTANTES COM OS "JUMPERS" DE "WAIT STATE"

Os dispositivos de E/S das placas escravas são lentos quando comparados com a velocidade de processamento do Micro Mestre, para evitar problemas de funcionamento os sinais para estes dispositivos são acrescentados de ciclos de máquina, as temporizações resultantes foram:

- J1 ==> 83 [us]
- J2 ==> 166 [us]
- J3 ==> 249 [us]
- J4 ==> 332 [us]
- J5 ==> 415 [us]
- J6 ==> 498 [us]

Os testes indicaram um funcionamento correto dos dispositivos de E/S a partir de J2.

5.4 TEMPORIZAÇÕES MÁXIMAS OBTIDAS

As temporizações são efetuadas por meio de um circuito contador de 16 bits programável, ou seja, o circuito conta com uma frequência definida (500 kHz) até atingir o valor programado. Pode-se portanto dividir esta frequência por valores que vão desde 1 até 65536.

Com isto os algoritmos de controle podem ter um tempo de operação variando de 2 (us) (1 ciclo) até aproximadamente 131,1 [ms] (65536 ciclos).

Para operação com o algoritmo de controle implementado utilizou-se um canal de temporização ajustado para 40 [ms], uma vez que se verificou que este algoritmo tem um $t_d = 4$ [ms].

As placas dedicadas a controle permitem até três ciclos temporizados para alocação de rotinas, possibilitando um algoritmo de controle com até três rotinas independentes, por exemplo uma rotina de controle de corrente, uma rotina de controle de velocidade e uma rotina para controle de posição.

Os resultados obtidos com o controle de uma junta indicam que o conjunto, temporizadores mais controlador de interrupção, opera corretamente. O sistema também foi testado para interrupções aninhadas (interrupção dentro de outra interrupção), pois o algoritmo de controle da junta, que é ativado utilizando-se um canal de interrupção, apresenta uma conversão analógica para digital que somente é ativada por meio de uma interrupção indicando o final de conversão.

5.5 RUÍDOS OBSERVADOS

Uma das principais dificuldades encontradas durante a fase de implementação do sistema foi a necessidade de diminuição do nível de ruído nos sinais digitais e principalmente nos circuitos conversores analógico digitais. No sistema digital foram utilizados capacitores de desacoplamento de 100 nF entre VCC (5 V) e terra em todos os circuitos integrados utilizados. Além disso foram utilizados circuitos de filtragem em toda a extensão do barramento principal entre os sinais de alimentação e terra, o que possibilitou reduzir o nível de ruído inicialmente verificado nestes sinais de 0.4 (V) para cerca de 100 (mV).

Nos circuitos analógicos o ruído se manteve na casa dos 10 (mV) o que define a perda de um bits de informação, sendo portanto ainda muito elevado.

Às novas placas em fase de montagem foram introduzidas várias melhorias em relação a placa protótipo, tais como:

- separação física dos circuitos analógico e digital
- separação da malha de terra, com junção somente nos terminais da fonte de alimentação.
- construção de um suporte metalizado aterrado entre as placas digitais.
- aterramento de todo o "RACK"
- confecção das placas em circuito impresso, o protótipo foi montado em "Wire-Wrapp".

O ruído é um fator muito importante para o fracasso ou sucesso da construção de sistema digitais aplicados a controle, uma vez que geralmente estes são utilizados em ambientes ruidosos, que podem causar problemas operativos. Para exemplificar, durante a implementação do CPR1 notou-se que os sinais de requisição de barramento, mesmo amplificados, causavam um funcionamento inadequado do microprocessador, devido ao nível de ruído encontrados nestes sinais.

5.6 DESEMPENHO DO ALGORITMO DE CONTROLE

O algoritmo de controle apresentado no Capítulo 4 foi implementado, sendo que seus parâmetros são enviados pelo Micro mestre através de um programa monitor protótipo desenvolvido em linguagem "C" por outro membro da equipe. Com isto foi montado um circuito de acionamento de uma junta.

Com este sistema montado foi possível testar várias características do sistema tais como:

- Transmissão do programa de controle
- Programa Monitor da Junta
- "Hardware" de acionamento
- Placa Digital de Controle Distribuído
- Verificação dos movimentos efetuados
- Integração do Micro Mestre e placas processadoras
- Filosofia de Comunicação
- Envio de parâmetros pelo operador

O algoritmo de controle implementado apresenta um ciclo com aproximadamente 4 [ms], Como o algoritmo da junta pode chegar a 15 [ms] pode-se implementar algoritmos muito mais poderosos a nível de junta.

Os resultados obtidos foram muito promissores, todos os circuitos funcionam corretamente com um desempenho satisfatório.

O programa monitor protótipo implementado, permite que os sinais de realimentação de posição, previamente armazenados pelo programa de controle da junta, sejam plotados na tela, mostrando assim, a curva resultante do movimento. O circuito de acionamento da garra desenvolvido, permite movimentos entre 0 mm (garra fechada), e 48 mm (garra aberta). A seguir são apresentadas duas curvas resultantes de movimentos efetuados nas juntas. Os movimentos são efetuados de 15 mm para 40 mm, variando-se heurísticamente os parâmetros q_0 , q_1 , q_2 .

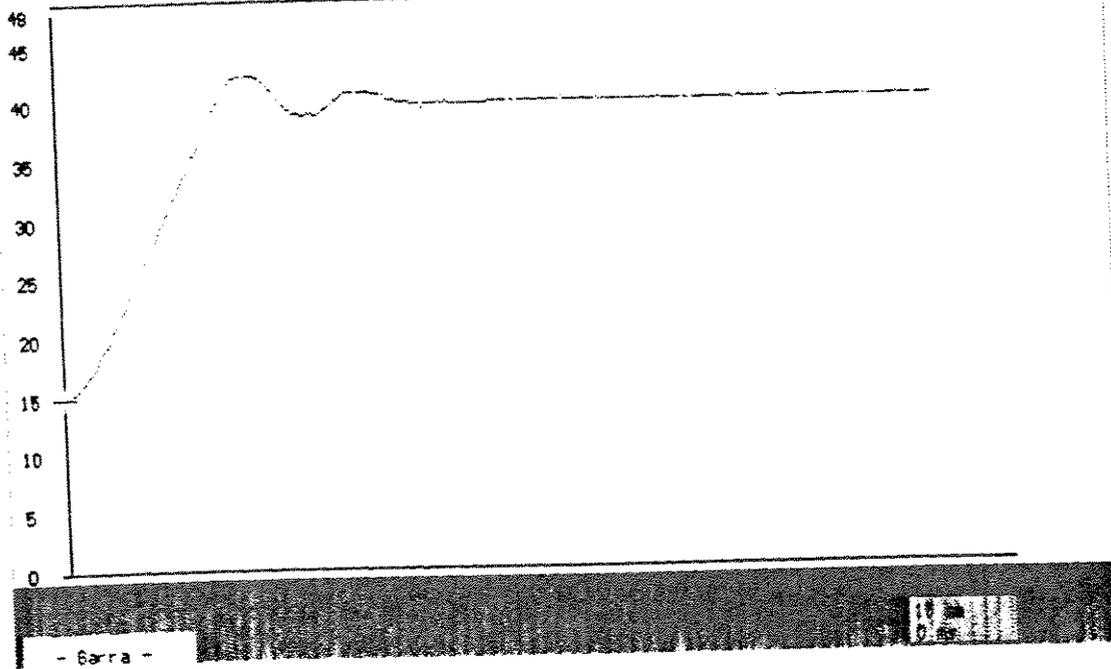


Figura 5.1 : Exemplo de movimento efetuado na junta

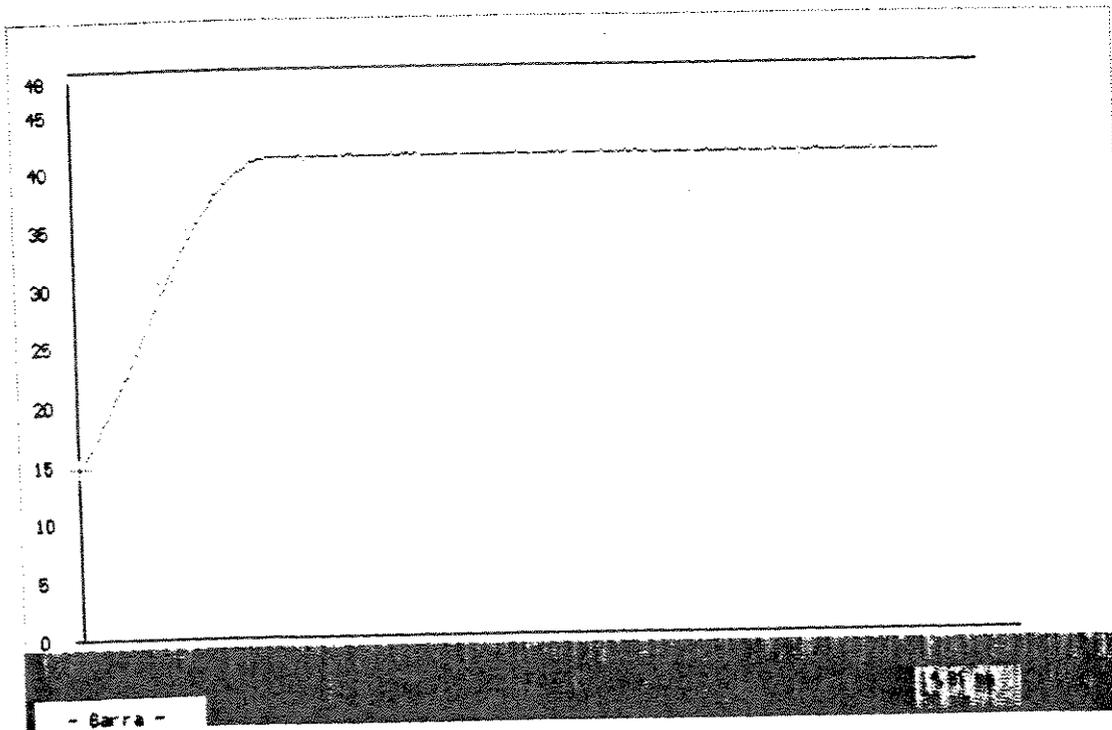


Figura 5.2 : Exemplo de movimento efetuado na junta

5.7 COMENTÁRIOS E CONCLUSÕES

O projeto do sistema digital de controle a multimicroprocessadores foi realizado com um grau de dificuldade maior do que se imaginava inicialmente, com escolhas tais como, filosofia de comunicação, microprocessadores das juntas, processador Mestre, e outras, que não poderíamos anteciper com certeza o seu sucesso. Algumas etapas iniciais do projeto apresentaram problemas e tiveram que ser reformuladas, gerando por exemplo, duas versões da placa de interface e da placa de controle digital direto. A seguir são comentados algumas escolhas e mudanças ocorridas durante as fases de projeto, testes e integração do Controlador Programável de Robos 1.

Em relação ao Micro Mestre optou-se por utilizar um microcomputador disponível comercialmente, pois, uma das características principais deste subsistema é a necessidade de várias interfaces já existentes e disponíveis no mercado, tais como:

- Monitor de Vídeo;
- Unidade de Disco Rígido;
- Unidade de Disco Flexível;
- Expansão de Memória RAM.

Foi escolhido um microcomputador compatível com o IBM PCxAT (80286 e 80287), com todos os periféricos acima citados incorporados a sua estrutura.

Para a interligação do Micro Mestre com as placas processadoras acopladas ao barramento, tornou-se necessária a elaboração de uma placa de interface para o PCxAT (Micro Mestre), com finalidades de geração, recepção e controle dos sinais para o barramento. Esta interface foi implementada utilizando-se um dos "slots" de expansão.

Outra escolha importante foi o tipo do barramento a ser utilizado, pensou-se inicialmente na utilização do padrão MULTIBUS que apresenta uma filosofia de interligação de microprocessadores por meio de um barramento comum com a utilização de sinais e filosofia de comunicação já definidos. Entretanto esta filosofia foi abandonada, optando-se por um mecanismo de memória comum, onde a memória das placas processadoras acopladas ao barramento é compartilhada com o Micro mestre, ou seja, quando for necessário o Micro Mestre força que o microprocessador da junta entre em "Halt" com uma subsequente liberação do seu barramento interno para o MM, permitindo o acesso a memória e aos dispositivos de E/S como se estes fossem pertencentes a ele próprio. Para um compartilhamento de memória com todas as placas processadoras (Máximo de dez) de uma forma imediata dentro do campo de endereçamento real, seriam necessários 320 kbytes de endereços de memória disponíveis no mapa de memória do Micro Mestre. No item 3.2 MAPA DE MEMÓRIA nota-se que este espaço de memória não está disponível.

Optou-se então pelo projeto de uma interface que utilizasse instruções de E/S para acesso a memória das Placas processadoras, pois o PCxAT apresenta 64 kbytes destes endereços. Cada placa tem 32 kbytes de memória portanto ainda assim esta faixa não seria adequada, isto foi solucionado pela geração do endereço da placa (4 bits) pela interface criada. Esta interface foi montada e nos testes ela funcionou muito bem, porém como existem muitos periféricos disponíveis para o PC, que utilizam endereços de E/S (O PC reserva 300 bytes para endereços de E/S). Nesta situação existia o problema das placas periféricas com decodificação parcial destes endereços, que poderia causar problemas operativos no futuro.

Esta filosofia foi abandonada e o projeto refeito utilizando-se endereços de memória no modo de endereçamento real do Micro Mestre. Isto foi conseguido pela desativação de parte da sua área de memória, diminuindo sua memória RAM de 640 kbytes para 512 kbytes, porém isto ainda não resolvia o problema dos 360 kbytes necessários para acessar todas as placas. Para resolver este problema foi mantido o circuito para geração de endereço da placa na interface. Para os endereços E/S das placas dedicadas foram escolhidos endereços em lacunas reservados para expansão do Micro Mestre.

Com isto a filosofia original de comunicação foi mantida, ou seja, quando o Micro Mestre necessitar receber ou enviar dados ele pede o barramento de uma ou mais placas e acessa a memória destas como se fossem uma extensão da sua própria, minimizando com isto o tempo necessário para a comunicação. Esta filosofia de barramento permite também que os dispositivos de E/S das placas escravas sejam acessados pelo Micro Mestre.

As placas processadoras são acopladas a um barramento situado em um "RACK" distante 2 metros. Os sinais do barramento são transmitidos por meio de "Flat Cables", como esta distância pode distorcer o sinal na entrada do barramento foi projetada uma placa, cuja função é restabelecer os sinais oriundos do Micro Mestre na entrada das placas dedicadas, para evitar problemas com ruídos e distorções nos sinais presentes no barramento.

No projeto das placas dedicadas a controle das juntas foram determinados várias grandezas tais como, tamanho da memória, palavra dos conversores A/D e D/A. A primeira versão desta placa apresentava uma memória RAM de 2 kbytes, EPROM de 2 kbytes, conversores de 8 bits, um canal de interrupção mascarável e um de interrupção não mascarável. Nos testes esta placa funcionou perfeitamente, porém foram encontrados algumas problemas tais como:

- Pouca memória RAM e EPROM;
- Conversores com pouca precisão (8 bits);
- Falta de temporizadores;
- Falta de um controlador de interrupção.

Para solucionarmos estes problemas optou-se pelo projeto de uma nova placa de controle onde a memória foi aumentada para 16 kbytes de RAM e 16 kbytes de EPROM, os conversores de 8 bits foram substituídos pelos de 10 bits, foram incluídos 3 canais programáveis de temporização, e um controlador de interrupção com oito canais. Esta placa demonstrou ter um desempenho satisfatório para a aplicação desejada.

Para testar todas estas placas foram desenvolvidas rotinas especiais de testes e no caso específico das placas de controle um "hardware" de teste com portas digitais de saídas com LEDS para visualização, e chaves digitais de entrada.

Rotinas especiais de teste de "hardware"

- Memória RAM da PCDD e PDES;
- Controlador de interrupção;
- Canais de temporização;
- Conversor D/A;
- Conversor A/D;
- Periférico de entradas e saídas digitais (PIA);
- Placa de Interface do Micro Mestre.

Estas rotinas foram desenvolvidas com a finalidade de garantir a integridade das placas antes de serem inseridas no barramento principal. Além de garantirmos também o correto funcionamento da placa de interface antes da interligação do sistema. Portanto, quando a integração do sistema foi efetuada garantimos que os problemas resultantes eram oriundos da filosofia utilizada para comunicação. A integração deste sistema foi um sucesso, os testes iniciais indicam que o sistema atende os requisitos de desempenho designados anteriormente.

Como o sistema JECA II não está completamente concluído torna-se difícil analisarmos o desempenho resultante do CPR1 no manipulador. Testes obtidos a nível de juntas indicam uma precisão de décimos de milímetros com o algoritmo de controle utilizado. Somente após a conclusão do programa Monitor do Micro Mestre e de todo o "Hardware" de Acionamento é que poderemos analisar o desempenho em termos de geração de trajetória, precisão e repetibilidade do manipulador.

É importante ressaltar que o principal objetivo inicialmente proposto, a versatilidade, foi alcançado, pois, foi implementado um programa Monitor provisório no qual foi possível transferir programas de controle para as juntas, fazendo com que o controlador da junta operasse em memória RAM, e captando dados resultantes armazenados pelo próprio programa de controle em memória RAM. Com isto pode-se efetivamente transferir programas genéricos para as juntas, realizar movimentos e após captar dados para análise do desempenho do movimento efetuado.

Em termos de "Hardware" do controlador, foram verificados e estão funcionando perfeitamente toda a filosofia de comunicação assim como as placas dedicadas, faltando somente o desenvolvimento do "Software" e a expansão do sistema.

BIBLIOGRAFIA

- Isermann R.; "Digital Control systems", Springer-Verlag Berlin, Heidelberg (1981).
- Graham J. H.; "Computer Architectures for Robotics and Automation", Gordon and Breach Science Publishers, New York (1987).
- Klafter R. D., Chmielewski T. A., Negin M.; "Robotic Engineering an Integrated Approach", Prentice Hall, Englewood Cliffs (1989).
- Paker Y.; "Multimicroprocessor System", Academic Press Inc, Orlando (1983).
- Zheng Y. F., Hermani H.; "Computation of Multibody System Dynamics by a Multiprocessor scheme", IEEE Trans. on Systems, Man and Cybernetics, Vol. SMC-16, N^o 1, pp. 102-110 (1986).
- Ben-Ari M.; "Principles of concurrent Programming", Prentice Hall international, Englewood Cliffs (1984).
- Electro-Craft Corp.; "DC Motor Speed Controls servo systems", Minnesota (1973).
- Kuo B. C.; "Sistemas de Controle Automático", Prentice Hall do Brasil, Rio de Janeiro (1985).
- Ogata K.; "Engenharia de Controle Moderno", Prentice Hall do Brasil, Rio de Janeiro (1982).
- Kose B. K., "Introduction to Microcomputer Control" IEEE Proceedings, Vol 132, N^o 5, (1984).

Mestaoui Y. D. M.; "Decentralised PD and PID Robotic Regulators", IEE Proceedings, vol 136, N^o 4, pp. 133 - 145, (1989).

Pinder E. E.; Herzog J. H.; " Distributed Computer Architecture and Fast Parallel Algorithms in Real Time Robot Control", IEEE trans. on system, man, and cybernetics, vol. SMC-16, N^o 4, pp. 543-549, (1986).

Kabuka M. R., Glaskowsky P. N., Miranda J.; " Microcontroller Based Architecture for Control of a Six Joints Robot Arm", IEEE Trans. on Industrial Electronics, vol. 35, N^o 2, pp 217-221, (1988).

Cypriano, L. B.; " Z80 Software", Editora Erica, São Paulo (1984).

Magalhães I. P., Tozzi C. L.; " Microprocessadores em Controle de Processos", Anais do 4^o Congresso Brasileiro de Automática.

Robot 7576; "Technical Specifications".

Servus Tecnologia e Informática, Folheto Técnico SF16.

Ferraté G.; " Robotica Industrial", Marcombo Bixareu editores, Barcelona (1986).

FTauile, J. R.; " O Desenvolvimento Internacional da Robótica, Dados e reflexões", Anais SOBRACON (1984).

Rohrer, G. H.; " Tendências Futuras da Robótica", Anais SOBRACON, Simpósio sobre Robótica, (1985).

Feldman, R. P.; " Como Evitar o Desemprego com o Advento de Robôs", Anais SOBRACON (1984).

Kirschner U.; " Técnica Progresso Possibilidades e Perigos", Círculo do Livro, (1984).